

Vysoká škola báňská – Technická univerzita Ostrava  
Fakulta elektrotechniky a informatiky  
Katedra informatiky

# **Meteostanice s výstupem dat na webové rozhraní a SNMP**

## **Weather Station with Web Interface and SNMP Support**

2014

Jan Šimeček

## Zadání diplomové práce

Student: **Bc. Jan Šimeček**

Studijní program: N2647 Informační a komunikační technologie

Studijní obor: 2612T025 Informatika a výpočetní technika

Téma: **Meteostanice s výstupem dat na webové rozhraní a SNMP  
Weather Station with Web Interface and SNMP Support**

### Zásady pro vypracování:

Cílem práce je navrhnout a realizovat meteostanici, která bude poskytovat data o naměřených meteorologických veličinách. Ta budou dostupná skrze webovém rozhraní a bude je možné získávat prostřednictvím protokolu SNMP.

1. Prostudujte metody měření meteorologických veličin.
2. Vyberte vhodná čidla pro měření některých meteorologických veličin: teplota, vlhkost, atmosférický tlak, případně další.
3. Navrhněte a realizujte vhodný systém, který bude získávat data z čidel a následně prezentovat. Realizujte propojení vybraných čidel se systémem.
4. Implementujte software, který bude poskytovat data z meteostanice prostřednictvím protokolu SNMP.
5. Aktuální naměřená data prezentujte také jako webovou stránku, zvažte možnost prezentace i historie měření.

### Seznam doporučené odborné literatury:

- [1] MAURO, Douglas R a Kevin J SCHMIDT. Essential SNMP. 2nd ed. Beijing: O'Reilly, 2005, 442 s. ISBN 05-960-0840-6
- [2] SKOČOVSKÝ, Luděk a Scott JERNIGAN. Linux: dokumentační projekt. 4., aktualiz. vyd. Překlad Ivo Fořt, David Krásenský. Brno: Computer Press, 2007, 1334 s. ISBN 978-80-251-1525-1

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **Ing. David Seidl, Ph.D.**

Datum zadání: 01.09.2013

Datum odevzdání: 07.05.2014



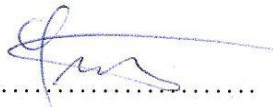
doc. Dr. Ing. Eduard Sojka  
vedoucí katedry



prof. RNDr. Václav Snášel, CSc.  
děkan fakulty

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 3. května 2014



.....

Rád bych na tomto místě poděkoval svému vedoucímu Ing. Davidu Seidlovi, Ph.D. za odborné vedení práce a cenné rady, které mi pomohly tuto práci zkompletovat. Dále bych rád poděkoval panu prof. Aleši Dudáčkovi z FBI-VŠB za pomoc při kalibraci digitálních čidel. Také bych velmi rád poděkoval své rodině za vytvoření příjemného domácího prostředí, bez kterého by tato práce nebyla s vysokou pravděpodobností již hotova.

## Abstrakt

Meteorologie je věda zabývající se studiem počasí. Lidé využívají poznatky o počasí již několik tisíc let. Zařízení měřící meteorologické veličiny se nazývají meteostanice. Na trhu je k dostání velké množství komerčních meteostanice, většina z nich však nemá otevřený software. Cílem diplomové práce je teoretický návrh a následná realizace meteostanice. Výstupní data z meteostanice jsou napojena na síťový protokol SNMP a vizualizována pomocí webového rozhraní.

Na výrobu meteostanice je potřeba pořídit meteorologická čidla a základní počítač. V první části práce se nachází popis a důvod výběru jednotlivých součástek. Komunikace všech prvků mezi sebou je popsána v dalších kapitolách práce. Následující část práce se věnuje popisu a napojením na protokol SNMP a webové rozhraní Cacti. Na konci práce je několik stránek věnovaných konstrukci a vzhledu meteostanice.

**Klíčová slova:** meteostanice, meteorologie, Raspberry Pi, Arduino, SNMP, Cacti, teplota, tlak, vlhkost, vítr, SHT15, BMP085, WH1080

## Abstract

Meteorology is science that studies weather. For thousands of years people have been using knowledge of weather. Device that measure meteorological quantities is called meteorological station. There are many commercial meteorological stations available on the market, but they usually do not have open source software. The aim of the thesis is theoretical layout and subsequent realization of a meteorological station. The outcome data from the meteorological station are connected to network protocol SNMP and visualized via web interface.

For the construction of the meteorological station it is necessary to obtain meteorological sensors and basic computer. The first part of the thesis deals with description and reasons for choosing components. Communication between all of the parts is described in the next chapters of the thesis. The following part of the paper is dedicated to description and connection to the SNMP protocol and web interface Cacti. The last several pages of the thesis are devoted to construction and the design of the meteorological station.

**Keywords:** meteostation, meteorology, Raspberry Pi, Arduino, SNMP, Cacti, temperature, pressure, humidity, wind, SHT15, BMP085, WH1080

## Seznam použitých zkratk a symbolů

ASN.1	– Abstract syntax notation one - standard popisující strukturu pro reprezentaci, kódování, dekodování a posílání dat v telekomunikačních a počítačových sítích
A/D převodník	– analogově-digitální převodník
API	– Application Programming Interface – rozhraní pro programování aplikací
ARM	– architektura procesorů vyráběných firmou ARM Limited
Composite RCA	– cinch konektor pro audio/video přenos signálů
DHCP	– Dynamic Host Configuration Protocol – protokol pro přidělení IP adresy zařízením v síti
DSI	– Display Serial Interface – rozhraní pro posílání video signálu
E <sup>2</sup> PROM	– Electrically Erasable Programmable Read-Only Memory – elektricky programovatelná paměť, nižší počet přepisů než FLASH
FLASH	– elektricky programovatelná paměť
GNU	– rekurzivní zkratka – Gnu is Not Unix, jinak typ operačního systému
GPIO	– General-purpose input/output – vstup-výstupní piny
GUI	– grafické uživatelské prostředí
HDMI	– High-Definition Multimedia Interface – rozhraní pro přenos nezkomprimovaného videa
I <sup>2</sup> C	– Inter-Integrated Circuit
IRQ	– Interrupt ReQuest – žádost o přerušení
ISP	– Internet Service Provider – poskytovatel internetového připojení
KB	– KiloByte – datová jednotka skládající se z 1024 bajtů
MCU	– MicroController Unit – mikrokontrolér
MMC	– MultiMediaCard – multimediální karta
MPEG	– Moving Picture Experts Group – množina standardů pro audio/video kompresní formát
MRTG	– The Multi Router Traffic Grapher – nástroj pro tvorbu grafů z routerů, podpora SNMP
MySQL	– databázový systém společnosti Sun Microsystems
NMS	– Network Management System – systém pro správu sítě

OpenGL	- Open Graphics Library – standard specifikující API pro tvorbu aplikací počítačové grafiky
PHP	- PHP: Hypertext Preprocessor – hypertextový procesor
PWM	- Pulse-Width Modulation – pulsní šířková modulace
RAM	- Random-Access Memory – počítačová paměť
RFC	- request for comment – dokument popisující síťový protokol
RJ45/RJ11	- modulární ethernetový konektor
RRDTool	- round-robin database tool – nástroj pro zpracování a ukládání časově závislých dat
SD	- Secure Digital memory card – zabezpečená paměťová karta
SGMP	- Simple Gateway Monitoring Protocol – protokol pro zjišťování stavů síťových bran
SMI	- Structure of Management Information – definuje moduly a jejich vzájemné vztahy v MIB
SNMP	- Simple network management protocol
SPI	- Serial peripheral interface
SRAM	- Static Random Access Memory – paměť, která nepotřebuje periodickou obnovu dat, je statická
SSH	- Secure Shell – zabezpečený komunikační protokol nad TCP/IP
TCP/IP	- Transmission Control Protocol/Internet Protocol – sada protokolů pro komunikaci v počítačové síti
TRAP	- past na události
USB	- Universal Serial Bus – sériová sběrnice

# Obsah

Úvod	6
<b>1 Meteorologie dnes</b>	<b>7</b>
1.1 Historie meteorologie v ČR	7
1.2 Dnešní meteostanice	7
1.3 Teoretické řešení výroby meteostanice	8
1.4 Měření meteorologických veličin	8
1.4.1 Teplota	9
1.4.2 Vlhkost	9
1.4.3 Tlak	10
1.4.4 Vítr	11
1.4.5 Srážky	11
1.4.6 Sluneční svit	11
<b>2 Výběr součástek</b>	<b>12</b>
2.1 Specifikace Raspberry Pi a Arduino	12
2.2 Výběr čidel	14
2.2.1 SHT15	16
2.2.2 BMP085-breakout	19
2.2.3 Analogová čidla	22
2.2.4 Kalibrace čidel	24
<b>3 Komunikace</b>	<b>25</b>
3.1 Analogová čidla – Arduino	25
3.2 Digitální čidla – Raspberry Pi	27
3.2.1 SHT15 – Raspberry Pi	27
3.2.2 BMP085 – Raspberry Pi	29
3.3 Zpracování dat na Arduinu a Raspberry Pi	31
3.3.1 Arduino – Raspberry Pi	31
3.3.2 Raspberry Pi - SNMP	31
3.3.3 Raspberry Pi - Cacti	31
3.3.4 Nullmailer - mobilní zařízení	32
3.4 Stručný výpis programů	33
3.4.1 AgentX	33
3.4.2 Data z analogových čidel	33
3.4.3 SHT15	34
3.4.4 BMP085	34
3.4.5 Příjem dat z Arduina	34



<b>4</b>	<b>SNMP</b>	<b>35</b>
4.1	Historie SNMP . . . . .	35
4.2	Architektura SNMP . . . . .	36
4.2.1	MIB . . . . .	36
4.2.2	Net-SNMP . . . . .	38
4.2.3	AgentX . . . . .	39
4.2.4	Konfigurační soubory . . . . .	40
<b>5</b>	<b>Cacti</b>	<b>41</b>
5.1	Historie Cacti . . . . .	41
5.2	Vlastnosti Cacti . . . . .	41
5.3	Instalace na Raspberry Pi . . . . .	42
5.4	Vytváření pluginu . . . . .	43
5.4.1	Implementace pluginu . . . . .	44
5.4.2	Instalace pluginu . . . . .	45
5.4.3	Nastavení sítě . . . . .	45
<b>6</b>	<b>Konstrukce meteostanice</b>	<b>46</b>
6.1	Upevnění mechanických dílů . . . . .	46
6.2	Krabičky pro Arduino a Raspberry Pi . . . . .	46
6.2.1	Výběr konektorů . . . . .	47
	<b>Závěr</b>	<b>48</b>
	<b>Literatura</b>	<b>50</b>
	<b>Přílohy</b>	<b>51</b>
<b>A</b>	<b>Schémata připojení analogových čidel</b>	<b>52</b>
<b>B</b>	<b>Konstrukce meteostanice – fotografie</b>	<b>53</b>
<b>C</b>	<b>Příloha na CD</b>	<b>57</b>

### Seznam tabulek

2.1	Tabulka vlastností teplotních a vlhkostních čidel . . . . .	15
2.2	Tabulka vlastností tlakových čidel . . . . .	15
2.3	Funkce pinů teplotního a vlhkostního čidla SHT15 [8] . . . . .	17
2.4	Závislost hodnot koeficientů na napětí [8] . . . . .	18
2.5	Závislost hodnot koeficientů na typu převodu [8] . . . . .	18
2.6	Koeficienty pro výpočet vlhkosti [8] . . . . .	19
2.7	Funkce pinů teplotního a tlakového čidla BMP085-breakout [2] . . . . .	20
2.8	Výstupy směrové růžice při různém natočení [3] . . . . .	23
2.9	Maximální odchylky mých čidel a čidel stanice VantagePro2+ . . . . .	24
3.1	Hodnoty kontrolních registrů u různých nastavení senzoru [2] . . . . .	30
5.1	Tabulka znázorňující informace o zvoleném zařízení. . . . .	44

### Seznam obrázků

2.1	Počítač Raspberry Pi, ver. 2 [25] . . . . .	12
2.2	Deska platformy Arduino Leonardo [23] . . . . .	13
2.3	Maximální přesnost teploty různých typů senzorů SHTxx [8] . . . . .	16
2.4	Maximální přesnost vlhkosti různých typů senzorů SHTxx [8] . . . . .	17
2.5	Funkce pinů teplotního a vlhkostního čidla SHT15 [8] . . . . .	18
2.6	Funkce pinů teplotního a tlakového čidla BMP085 [2] . . . . .	20
2.7	Závislost nadmořské výšky na tlaku [2] . . . . .	21
2.8	Schéma zapojení anemometru a směrové růžice. Směr natočení je označen anglickými písmeny [26]. . . . .	23
3.1	Blokové schéma zapojení prvků meteostanice . . . . .	26
3.2	Standardní schéma SPI s jedním slave zařízením [27] . . . . .	27
3.3	Časový diagram komunikace s SHTx. SCK je hodinový signál, DATA symbolizují přenos dat v binární formě [8]. . . . .	28
3.4	Časový diagram I <sup>2</sup> C senzoru BMP085 [2] . . . . .	29
3.5	Časový diagram měření teploty a vlhkosti senzoru BMP085 [2] . . . . .	29
3.6	Časový diagram měření teploty a vlhkosti senzoru BMP085 [2] . . . . .	30
4.1	SMI stromová struktura protokolu SNMP [4] . . . . .	37
4.2	Architektura balíku Net-SNMP [28] . . . . .	38
5.1	Ukázka stránky zobrazující údaje o meteostanici (plugin). . . . .	45
6.1	DIN – vidlice [25] . . . . .	47
6.2	DIN – do panelu [25] . . . . .	47
6.3	Jack – vidlice – stereo/mono [29] . . . . .	47
6.4	Jack – do panelu [25] . . . . .	47
A.1	Schéma zapojení anemometru a směrové růžice [30] . . . . .	52
A.2	Schéma zapojení srážkoměru [30] . . . . .	52
B.1	Přední stěna krabičky pro Arduino. Pohled zvenku. . . . .	53
B.2	Přední stěna krabičky pro Arduino. Pohled zevnitř. . . . .	54
B.3	Připojení analogových čidel přes konektory do nepájivého pole a Arduina. . . . .	54
B.4	Přední stěna krabičky Raspberry Pi. Pohled zvenku. . . . .	55
B.5	Raspberry Pi upevněné v krabičce. . . . .	55
B.6	Celá meteostanice. . . . .	56

### Seznam výpisů zdrojového kódu

1	Ukázka kódu nastavení přerušení a dalších pinů na Arduino . . . . .	25
2	Ukázka MIB souboru vytvořeného pro meteostanici . . . . .	36
3	Nastavení skriptů pro zpracování TRAPů různých zařízení. . . . .	40

## Úvod

S meteorologií, jakožto vědou zabývající se atmosférickými jevy, jejich vlastnostmi, stavy a složením, se můžeme setkat už od pradávna. První výskyt použití slova meteorologie spadá až hluboko ke kořenům samotné lidské existence. Při tzv. zemědělské revoluci kolem roku 5 000 př. n. l. se lidé poprvé začali zajímat o souvislosti mezi počasím a pěstováním plodin. Člověk se s měřením meteorologických veličin potýká už nějaký čas, a tak se i v dnešní době meteorologie podílí velkou měrou na zásadních rozhodnutích lidstva. Ať už na poli vědy při startu raketoplánů nebo běžného denního rozhodování zda si vzít ven deštník či nikoliv. K měření hodnot fyzikálních veličin týkajících se počasí se využívají meteostanice. Dnes se můžeme na trhu setkat s obrovským množstvím meteostanic, většinou připojených s výstupem na LCD displej.

Tato práce je zaměřená na navrhnutí a sestavení meteorologické stanice a následné zpracování dat meteorologických veličin. Měřenými veličinami jsou: teplota, tlak, vlhkost, směr a rychlost větru, srážky a intenzita slunečního svitu. Výstupním subjektem je databáze, do které se logují všechny záznamy a umožňuje napojení více meteostanic najednou. Uživatelsky přívětivějším výstupem je webové rozhraní Cacti s vlastním implementovaným pluginem konkrétně pro meteostanici. V poslední době je trendem propojení složitějších meteostanic se síťovým protokolem SNMP, aby byla data dostupná z jakéhokoliv přístroje připojitelného na internet. Proto je součástí diplomové práce také napojení meteostanice na tento protokol. V práci je postupně popsán momentální stav v oblasti meteorologie a dostupnost, případně konstrukce meteostanic. Následující kapitoly se věnují implementaci algoritmů pro vyčítání dat z čidel a komunikaci mezi jednodeskovými počítači. V dalších částech práce je popsána implementace webového uživatelského rozhraní s tvorbou pluginu a skriptů, které spolupracují s protokolem SNMP.

Téma diplomové práce jsem si zvolil, jelikož mě vždy zajímal z části software i hardware. Práce také určitým způsobem navazuje na bakalářskou práci, ve které bylo hlavním cílem vytvoření jednoduché meteostanice na bázi plošného spoje s připojením na PC.

### 1 Meteorologie dnes

Dnešní moderní, dynamicky se rozvíjející společnost by nemohla bez meteorologie existovat. Tato skutečnost se promítá do celého spektra lidských činností, v historii lidé využívali znalosti o počasí pro pěstování plodin a chov dobytka [1]. Dnes je tato vědní disciplína potřebná např. u řízení letadlového provozu, včasného varování před přírodními katastrofami nebo také u předpovědi týkajících se různých sportovních klání. Atmosférické jevy se měří různými přístroji, od amatérských meteostanic až po profesionální přístroje, které byly například umístěny na vozítku Curiosity pro sledování povrchu planety Mars.

#### 1.1 Historie meteorologie v ČR

Počátky meteorologie se u nás datují k roku 1775, kdy se v pražské jezuitské koleji v Klementinu poprvé objevují pokusy o souvislé měření teploty. V roce 1804 je toto měření obohaceno o měření srážek. Dalším důležitým milníkem je bezesporu nejnižší dosud naměřená teplota v ČR, její hodnota byla  $-42.2(^{\circ}\text{C})$  a váže se k 11.únoru roku 1929 a obci Litvínovice u Českých Budějovic.

Ve 20. století našeho letopočtu se obrovský vývoj vědy projevil i v měření meteorologických jevů. V roce 1971 byla zahájena pravidelná sondáž atmosféry v Praze na Libuši. Následně v roce 1993 proběhla digitalizace radiolokačního měření na tom samém místě. První superpočítač byl pro meteorologické účely využit v roce 1998. Před třemi lety se ČR stává členem EUMETSATu - organizace pro využívání meteorologických družic. V roce 2012 se počet všech meteorologických stanic v ČR zvýšil na celkových 735 [18].

#### 1.2 Dnešní meteostanice

V České republice se můžeme setkat s několika typy meteostanic. Nejrozšířenějším typem jsou stanice klimatologické, které mají za úkol shromažďovat data o vývoji klimatu 3x denně. Meteostanice lze rozdělit jak podle účelu měření, tak podle technické vyspělosti stanice.

Podle účelu měření:

- Synoptické stanice - měření probíhá v hodinových intervalech, ze zjištěných hodnot se vytváří synoptické mapy
- Klimatologické stanice - měření se provádí 3x denně, slouží pro sledování vývoje klimatu
- Srážkoměrné stanice - provádí se měření množství spadlých srážek každý den v 7 hodin ráno [5]
- Letecké stanice - sledování počasí pro letecké účely
- Silniční stanice - slouží pro řízení silniční dopravy a poskytování informací účastníkům silničního provozu

## 1 METEOROLOGIE DNES

---

- Amatérské stanice - jsou provozovány amatérskými nadšenci pro meteorologii, většinou jsou data dostupná na internetu nebo jinak elektronicky uložena

Podle technické vyspělosti:

- Podomácku vyrobené - jednoduché konstrukce, velmi nepřesné měření, spíše pro demonstrativní účely
- Poloprofesionální - měření pomocí specializovaných kalibrovaných čidel, využití elektronických součástek v čele s mikrokontroléry, relativně přesné měření, s několika výstupy - LCD displej, webové rozhraní, síť (LAN, WAN), cena se pohybuje v řádech tisíců Kč
- Profesionální - spolehlivé a přesné přístroje, mnoho funkcí, měření podle norem a standardů, cena v řádech desítek tisíců Kč

### 1.3 Teoretické řešení výroby meteostanice

Při návrhu vytvoření nové meteostanice připojitelné např. k počítači mě napadly tři základní varianty.

První varianta je koupě celé nové meteostanice. To sebou nese problémy v podobě vysokých nákladů a případných modifikací stanice, rozšíření její funkčnosti a přidávání dalších modulů.

Druhá varianta je vytvoření meteostanice zcela svými silami, nicméně v tomto případě je nutno počítat s výrobou měřící techniky, návrhu plošného spoje a promyšlení komunikace s počítačem či sítí. Tento postup je velmi zdlouhavý a náročný na čas, proto jsem se rozhodl pro zlatou střední cestu – 3. variantu.

Jako základní stavební prvek meteostanice jsem využil dva jednodeskové počítače. K těmto počítačům jsem připojil kalibrovaná digitální a analogová čidla. Čidla pochopitelně nejsou vlastní výroby, ale koupené. Analogová čidla jsou prodejná jako mechanické díly meteostanice, která je volně dostupná v kamenných i internetových obchodech. Všechna čidla jsou připojitelná k počítačům přes vstup-výstupní piny a příslušná rozhraní. Tímto způsobem jsem docílil relativně levného zařízení, které je možno rozšiřovat dle libosti. Záleží pouze na programátorských a elektrotechnických schopnostech konstruktéra.

### 1.4 Měření meteorologických veličin

Základními meteorologickými veličinami, které slouží k předpovědi počasí, jsou teplota, tlak a vlhkost. Každá veličina se měří jinými metodami. Ke zpracování naměřených dat se využívají meteostanice. V ČR se můžeme setkat s nejvyšším počtem klimatologických meteorologických stanic. V následujících odstavcích jsou shrnuty metody měření jednotlivých veličin elektrotechnickými senzory, z nichž některé byly použity při výrobě meteostanice.

## 1 METEOROLOGIE DNES

---

### 1.4.1 Teplota

Teplota je skalární fyzikální veličina související s kinetickou energií částic. Čím vyšší tato energie je, tím je vyšší teplota. Přímo ovlivňuje všechny živé organismy na Zemi, a proto se řadí k nejdůležitějším meteorologickým veličinám. Teplota je také důležitým indikátorem stavu počasí, a tudíž je pro člověka výhodné, aby ji měřil a zaznamenával údaje (např. pro následnou předpověď počasí) [6]. Základními jednotkami pro měření teploty jsou stupeň Celsia (°C), Kelvin (K) a stupeň Fahrenheita (°F). Teplota se měří pomocí teploměrů, které můžeme rozdělit na několik typů:

- dilatační teploměry – Využívají roztažnosti látek při změně teploty (např. rtuť či líh).
- elektrické teploměry
  1. odporové snímače teploty – Jsou založeny na teplotní závislosti elektrického odporu kovů nebo polovodičů.
  2. kovové odporové snímače teploty – Využívají změny elektrolytické vodivosti.
  3. termoelektrické snímače teploty – Při různé teplotě konců vodiče se na každém konci objeví jiný potenciál.
  4. PN snímače teploty – Využívají závislosti teploty na napětí u PN<sup>1</sup>přechodu v propustném směru.
- barevné teplotní indikátory – Určují přibližnou teplotu povrchu tělesa. Při styku s povrchem tělesa dojde k chemické reakci a změně barvy indikátoru.
- bezdotykové teploměry – Jsou založeny na zachycení elektromagnetického záření, které tělesa vyzařují.

Meteostanice měří teplotu pomocí termoelektrického snímače teploty.

### 1.4.2 Vlhkost

Vlhkost udává, jaké množství vodní páry obsahuje dané množství vzduchu. Relativní vlhkost vzduchu udává poměr mezi okamžitým množstvím vodních par ve vzduchu a množstvím vodních par, jaké by měl nasycený vzduch při stejném tlaku a teplotě [7]. K měření vlhkosti můžeme použít tyto metody:

- váhová – Je přesná a spočívá v porovnání hmotnosti vzduchu před a po jeho vysušení.
- kondenzační – Stanovení teploty rosného bodu pomocí ochlazování měřicí plošky, v okamžiku orosení je teplota na plošce rovna teplotě rosného bodu, poté se určí vlhkost.

---

<sup>1</sup>PN přechod je rozhraní polovodiče typu P a N, propouští elektrický proud pouze jedním směrem



## 1 METEOROLOGIE DNES

---

- infračervená – Vodní pára ve vzduchu absorbuje infračervené záření - čím méně ho projde na detektor, tím je větší vlhkost.
- hydroskopická – Hydroskopické látky mění při absorbování vlhkosti ze vzduchu své geometrické vlastnosti (např. lidský vlas).
- elektrické
  1. kapacitní – Změna kapacity kondenzátoru, jehož dielektrikum je vytvořeno ze speciálního polymeru, změnou vlhkosti se mění vlastnosti polymeru.
  2. odporová – Využívá změny elektrolytické vodivosti.

Meteostanice měří vlhkost pomocí kapacitní metody. Důležitým pojmem při měření vlhkosti je tzv. „rosný bod“. Rosný bod je teplota, při které má vzduch relativní vlhkost 100 % a je tudíž maximálně nasycen vodními parami.

### 1.4.3 Tlak

Tlak je fyzikální veličina vyjadřující poměr síly, která působí kolmo na rovinnou plochu. Základní jednotkou je Pascal (Pa). Z meteorologického hlediska je tlak důležitý, jelikož se podle něj dá zjistit směr a rychlost proudění vzduchu [7]. Tlak se měří například těmito způsoby:

- barometr – skleněná trubička, která otevřeným koncem přechází do baňky
- aneroid – ručička na ciferníku ukazuje tlak pomocí deformace krabíčky, na kterou je připojena
- elektrické
  1. Wheatstonův můstek – složen ze čtyř odporových cest, které mění svůj odpor se změnou svých rozměrů vlivem tlaku na membránu
  2. využitím křemíkových piezorezistorů, které při změně tlaku samy generují napětí na elektrických ploškách

V diplomové práci je použito čidlo, které využívá křemíkových piezorezistorů.

## 1 METEOROLOGIE DNES

---

### 1.4.4 Vítr

Vítr je horizontální proudění vzduchu v atmosféře [5]. Je vyvolaný rozdíly v tlaku vzduchu a rotací Země. Při jeho popisu nás hlavně zajímá jeho směr a rychlost. Rychlost větru se měří pomocí anemometru. Směr větru se měří pomocí směrové růžice, která je většinou součástí anemometru. Pro tyto účely se využívá těchto přístrojů:

- rychlost – anemometry
  1. mechanické – energie větru se přenáší na konstrukci, využit tzv. Robinsonův kříž
  2. aerodynamické – tlak proudícího vzduchu je přenášen a porovnáván s tlakem statickým
  3. akustické – akustické anemometry měří nebo odvozují rychlost, ale i směr větru, ze změn šíření zvuku v atmosféře
- směr – směrovky
  1. využití odporového potenciometru – se změnou natočení se změní hodnota výstupního napětí
  2. využití zabudovaného enkodéru, který snímá úhel natočení magnetického pole

### 1.4.5 Srážky

Atmosferické srážky jsou vodní kapky (částice vody) nebo ledové částice, které vznikly z vodní páry. Tyto částice následně vlivem zemské přitažlivosti padají z oblaků k zemi. Na meteorologických stanicích se měří pomocí srážkoměru nebo ombrografu. Úhrn srážek se udává v milimetrech ( $1 \text{ mm} = 1 \text{ l/m}^2$ ). Sníh se nechává roztát. Pro měření srážek se nejčastěji používá:

- ombrograf – padající srážky jsou sváděny do nádoby uvnitř válce, přelitím do kalibrovaného odměrného válce se zjistí úhrn srážek
- člunkový srážkoměr – modernější způsob měření, srážky jsou sváděny na překlápěcí člunky, měří se počet překlápění člunků za určitou dobu

### 1.4.6 Sluneční svit

Délka slunečního svitu udává počet hodin za den, měsíc nebo rok, po které přímé sluneční záření dosahovalo zemského povrchu. Doba slunečního svitu udává časový interval, po který svítilo slunce. Slunečního svit se měří pomocí slunoměrů (heliografů) těchto typů:

- Campbellův-Stokesův slunoměr – paprsky jsou soustředěny koulí na registrační pásek, který je paprsky propalován
- Marvinův slunoměr – využití registračního kontaktního elektrického teploměru
- Jordanův slunoměr – exponuje sluneční paprsky na speciální fotografický papír

### 2 Výběr součástek

Nejdůležitějším krokem při výběru součástek byl výběr základní platformy, na které mají běžet všechny podpůrné programy. Tento malý počítač by měl umět komunikovat s ostatními počítači pomocí nějakého rozhraní, být připojitelný na síť pomocí ethernetu a měl by také umět komunikovat s některými čidly. Nejpřístupnějším počítačem v této kategorii se mi jevil Raspberry Pi, jelikož je malý, je osazeno mikroprocesorem, disponuje množstvím vstup-výstupních rozhraní a běží na něm operační systém na bázi Linuxu. Jelikož je použito více čidel, výborně se hodí i další malý počítač, který je schopen číst data z analogových čidel a posílat je po sériové lince. Jeho jméno je Arduino, má velmi nízkou spotřebu energie a je osazen programovatelným mikrokontrolérem ATmega32U4.

Pro měření teploty, vlhkosti a tlaku jsem vybral digitální čidla, jelikož při komunikaci s mikrokontrolérem využívají uživatelsky přívětivých rozhraní SPI a I<sup>2</sup>C. O měření ostatních veličin se starají čidla s analogovým výstupem připojená na Arduino. Tyto senzory jsou vlastně pouze mechanickými díly poloprofesionální meteostanice WH1080.

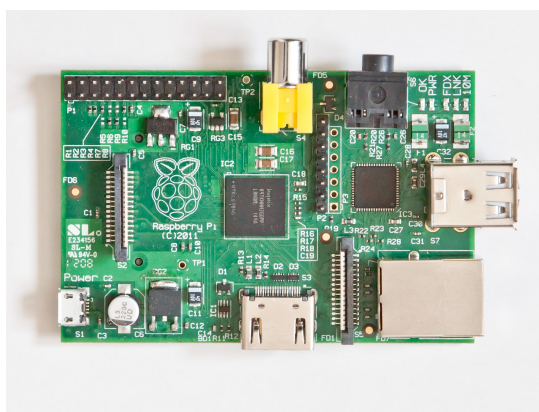
#### 2.1 Specifikace Raspberry Pi a Arduino

Obě dvě zařízení se dají považovat za malé počítače, které se vyznačují nízkou spotřebou a obsahují několik vstup-výstupních pinů. Na Raspberry Pi běží plnohodnotný operační systém, a proto je možné na něj nainstalovat velké množství podpůrného softwaru. Arduino je naproti tomu vybaveno sadou knihoven pro práci s mikrokontrolérem.

**2.1.0.1 Raspberry Pi** Raspberry Pi se dá považovat za malý plnohodnotný počítač, jeho součástí je totiž mikroprocesor, grafický procesor, USB, ethernetový konektor a mnoho dalších rozhraní. Jeho rozměry se podobají například rozměrům kreditní karty [21].

##### 2.1.0.1.1 Specifikace počítače

- procesor ARM1176JZF-S z rodiny ARM11 taktovaný na 700 Mhz
- grafický procesor VideoCore IV, podporující OpenGL ES 2.0, 1080p30, MPEG-4
- 256 MB (u vyšších verzí 512 MB) RAM sdílených s grafickou kartou
- jeden až dva USB porty
- Obrazový výstup Composite RCA, HDMI, DSI
- Zvukový výstup přes 3,5 mm konektor, HDMI



Obrázek 2.1: Počítač Raspberry Pi, ver. 2 [25]

## 2 VÝBĚR SOUČÁSTEK

- slot pro SD nebo MMC kartu
- v dražší verzi ethernetový adaptér 10/100 s konektorem RJ45
- 8xGPIO, UART, I<sup>2</sup>C, sběrnice SPI
- cena kolem 1200 Kč (GM Computers Ostrava)

Jednou z největších výhod Raspberry Pi je přítomnost operačního systému, konkrétně distribuce zvané Raspbian, což je odnož Debianu založeném na Linuxu. Počítač je napájen přes mikro-USB port a hodnota napětí je 5.0 V [19].

**2.1.0.2 Arduino** Arduino je open-source platforma založená na mikrokontroléru AT-Mega od firmy Atmel. Hodí se na programování periodicky se opakujících operací, čtení dat, kontroly jiných zařízení atd. Arduino se prodává hned v několika provedeních, každé provedení se liší svými technickými detaily. V době výroby meteorostanice k této diplomové práci bylo nejlépe dostupné Arduino Leonardo, které sice nedisponuje tak velkým obsahem paměti jako například Arduino DUE nebo MEGA, nicméně plně postačuje ke zpracování signálu analogových čidel, které jsou na něj připojeny.

### 2.1.0.2.1 Specifikace počítače

- MCU: ATmega32u4
- Pracovní napětí: 5 V
- I/O Piny: 20 (7 použitelných jako PWM výstup)
- Analogové vstupy: 12
- Flash paměť: 32 KB (ATmega328) 4 KB použito pro bootloader
- SRAM 2,5 KB (ATmega328)
- EEPROM 1 KB (ATmega328)
- Krystal: 16 MHz
- cena: kolem 650 Kč (GM computers Ostrava)



Obrázek 2.2: Deska platformy Arduino Leonardo [23]

Na rozdíl od Raspberry Pi Arduino neobsahuje operační systém a jedinou možností, jak s ním pracovat, je využití mikrokontroléru. Po naprogramování se může mikrokontrolér chovat tak, že provádí nějakou opakující se činnost, nebo čeká na signály přerušení, které následně zpracuje. Desku počítače lze napájet skrze externí napájecí napětí, jehož doporučený rozsah je 7–12 V. Počítač může být také napájen přes mikro-USB port, čehož lze využít při připojení na jiná zařízení. Hodnota napětí je pak 5.0 V. Mikro-USB

## 2 VÝBĚR SOUČÁSTEK

---

port slouží také pro komunikaci s ostatními zařízeními a posílá data pomocí sériového rozhraní [23].

### 2.2 Výběr čidel

Při výběru měřících čidel jsem se potýkal s několika omezeními. U digitálních čidel byla prvním omezujícím kritériem velikost čidla, jelikož ta v některých případech značně překračovala můj stanovený plán, a to umístění čidel do krabičky. Dalším podstatným kritériem je rozsah a přesnost měření, převážně u teploty, vlhkosti a tlaku se tyto veličiny značně lišily. V neposlední řadě byla důležitá také cena, jelikož nemá smysl kupovat přehnaně přesné přístroje, pokud na to nejsou dostatečné prostředky.

V tabulkách 2.1 a 2.2 se nalézají kladné a záporné body, ze kterých jsem vycházel při výběru čidel. Ceny jsou pouze orientační.

Jako teplotní a vlhkostní čidlo jsem si vybral SHT15, jelikož je přesné, dokáže měřit obě veličiny a mohl jsem ho využít z bakalářské práce [24], tím pádem nebylo potřeba čidlo kupovat. U tlakového čidla jsem se rozhodl pro verzi čidla BMP085-breakout, protože součástí čidla je i redukce vývodů na malý plošný spoj s několika potřebnými součástkami.

## 2 VÝBĚR SOUČÁSTEK

Čidlo	Rozsah	Přesnost	Cena	Plusy	Mínusy
DS18B20	-55-125 °C	0.5 °C	45 Kč	nízká cena, digitální výstup	pouze teplota
HCH-1000-001	10-95 %	±2 %	50 Kč	nízká cena, min. 500 kusů	pouze vlhkost, výstupem je hodnota kapacitance
HIH-4000-004	0-100 %	±2 %	420 Kč	přesnost	pouze vlhkost, cena
DHT-11	0-50 °C, 20-90 %	±0.5 °C, ±5 %	95 Kč	přesnost, cena	malý rozsah
SHT-15	-40-124 °C, 0-100 %	±0.3 °C, ±0.2 %	460 Kč	přesnost, digitální výstup	cena

Tabulka 2.1: Tabulka vlastností teplotních a vlhkostních čidel

Čidlo	Rozsah	Přesnost	Cena	Plusy	Mínusy
MPXH6115AC6U	15-115 KPa	±1.5 %	675 Kč	rozsah	analogový výstup, cena
MPX4115A	15-115 KPa	±1.5 %	867 Kč	rozsah	analogový výstup, cena
LPS331AP	26-126 KPa	±1 %	12 \$	přesnost, digitální výstup (I <sup>2</sup> C, SPI)	nedostupnost v ČR
BMP085 Breakout	30-110 KPa	±1 %	500 Kč	přesnost, digitální výstup (I <sup>2</sup> C), cena	

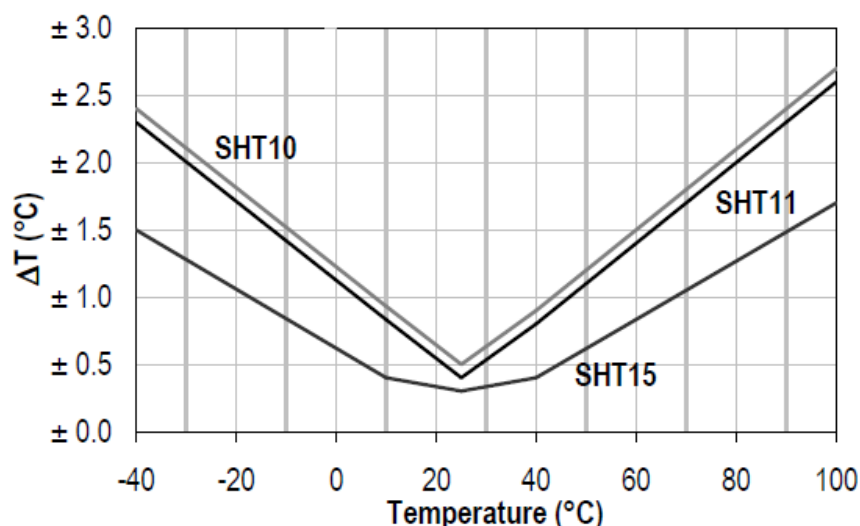
Tabulka 2.2: Tabulka vlastností tlakových čidel

## 2 VÝBĚR SOUČÁSTEK

### 2.2.1 SHT15

Toto čidlo měří teplotu i vlhkost. Řešení je vhodnější než měření obou veličin vlastními čidly, jelikož nedochází k tak velké odchylce chyb díky přesnějšímu měření [8]. Je snadno dostupné v obchodech v ČR, prodává se v SMD provedení a má digitální výstup. Má v sobě zabudované senzory pro měření teploty a vlhkosti a poskytuje plně kalibrovaný digitální výstup. A/D převodník je čtrnáctibitový, tím pádem poskytuje relativně přesný výsledek. Ochrana čidla je vylepšena o odstínění vnějšího rušení. Čip je individuálně kalibrován v komoře, která je určena speciálně pro získání referenční počáteční hodnoty vlhkosti. Sériové rozhraní a vnitřní regulace napětí umožňuje snadný a rychlý systém integrace. Čidlo jsem si vybral, protože dokáže měřit dvě veličiny najednou a v poměru cena/výkon/náročnost realizace se jeví jako dobré řešení.

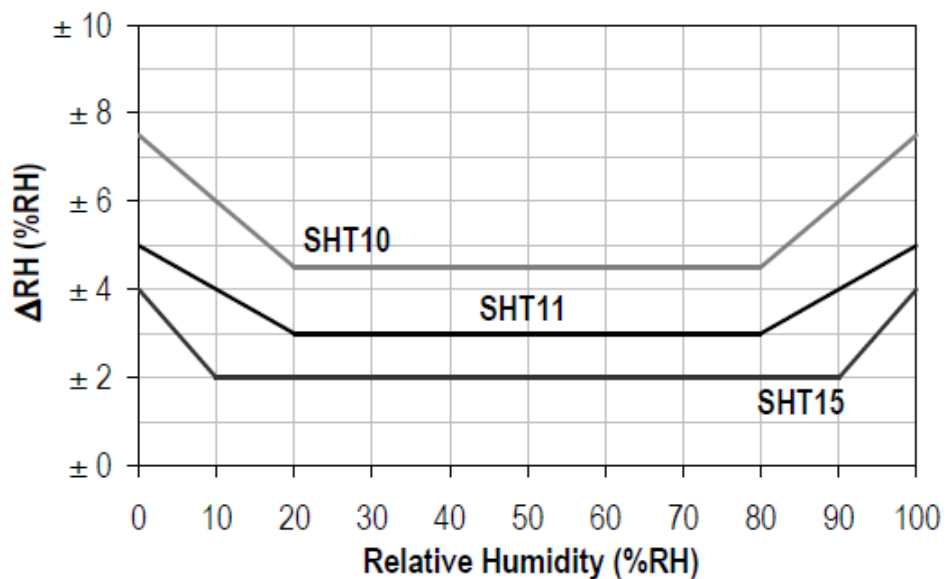
Čidlo SHT15 se mírně liší od svých předchůdců kvalitou měření teploty i vlhkosti. Grafy 2.3 a 2.4 ukazují maximální přesnosti jednotlivých typů senzorů.



Obrázek 2.3: Maximální přesnost teploty různých typů senzorů SHTxx [8]

**2.2.1.1 Rozhraní čidla** V tabulce 2.3 a na obrázku 2.5 jsou uvedeny jednotlivé funkce pinů čidla. Vstupní napětí čidla SHT15 musí být v rozmezí 2.4–5.5 V, doporučené napětí je 3.3 V. Datový pin (DATA) zprostředkovává čtení/zápis čidla. Při zápisu dat do čidla jsou platná data při nástupné hraně hodinového signálu (SCK) a musí zůstat stabilní dokud je hodinový signál nastaven na 1. Po detekci sestupné hrany hodinového signálu mohou být data změněna. Pro odstranění problémů odesílání neplatných dat musí mikrokontrolér řídit datový vodič pouze při jeho hodnotě rovné nule. Pro zlepšení udržení logické hodnoty datového vodiče v 1 se využívá tzv „pull-up“ rezistor<sup>2</sup>. „Pull-up“ rezistor

## 2 VÝBĚR SOUČÁSTEK



Obrázek 2.4: Maximální přesnost vlhkosti různých typů senzorů SHTxx [8]

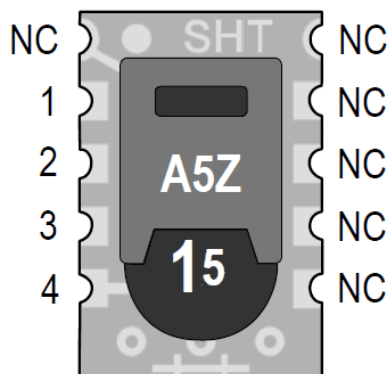
může být využit buď z mikrokontroléru (pokud je jeho součástí), nebo připojen externě. Doporučená velikost hodnoty rezistoru je 10 kΩ.

PIN	Jméno	Komentář
1	GND	zem
2	DATA	sériová data, obousměrná
3	SCK	sériové hodiny, pouze vstup
4	VDD	napájecí napětí
NC	NC	musí být nepřipojeno

Tabulka 2.3: Funkce pinů teplotního a vlhkostního čidla SHT15 [8]

<sup>2</sup>Pull up rezistor slouží k udržení logické hodnoty „1“





Obrázek 2.5: Funkce pinů teplotního a vlhkostního čidla SHT15 [8]

**2.2.1.2 Výpočet teploty a vlhkosti** Pro výpočet teploty se použije vzorec 2.1. Tabulky 2.4 a 2.5 ukazují hodnoty koeficientů použité v rovnici 2.1.

$$T = d_1 + d_2 \cdot SO_T \quad (2.1)$$

Kde:

$T$  – je výsledná teplota ve  $^{\circ}C$

$d_1$  – je teplotní koeficient

$d_2$  – je teplotní koeficient

$SO_T$  – je digitální výstup čidla

$VDD$	$d_1(^{\circ}C)$	$d_1(^{\circ}F)$
5	-40.1	-40.2
4	-39.8	-39.6
3.5	-39.7	-39.5
3	-39.6	-39.3
2.5	-39.4	-38.9

Tabulka 2.4: Závislost hodnot koeficientů na napětí [8]

$SO_T$ konverze	$d_2(^{\circ}C)$	$d_2(^{\circ}F)$
14 bitů	0.01	0.018
12 bitů	0.04	0.072

Tabulka 2.5: Závislost hodnot koeficientů na typu převodu [8]

## 2 VÝBĚR SOUČÁSTEK

---

Pro výpočet vlhkosti se použije vzorec 2.2, kde  $c_1$ ,  $c_2$  a  $c_3$  jsou koeficienty převzaté z tabulky 2.6.

$$RH_{linear} = c_1 + c_2 \cdot SO_{RH} + c_3 \cdot SO_{RH}^2 (\%RH) \quad (2.2)$$

Kde:

$RH_{linear}$  – je relativní vlhkost ve °C

$c_1$  – je vlhkostní koeficient

$c_2$  – je vlhkostní koeficient

$c_3$  – je vlhkostní koeficient

$SO_{RH}$  – je hodnota digitálního výstupu čidla

$SO_{RH}$	$c_1$	$c_2$	$c_3$
12 bitů	-2.0468	0.0367	$-1.5955E^{-6}$
8 bitů	-2.0468	0.5872	$-4.0845E^{-4}$

Tabulka 2.6: Koeficienty pro výpočet vlhkosti [8]

### 2.2.2 BMP085–breakout

Toto čidlo se vyznačuje velmi přesným měřením tlaku. Od výrobce je senzor již kalibrován, nicméně zákazník má možnost využít zabudovaného teplotního senzoru (umístěného uvnitř) pro výpočet ještě přesnější hodnoty tlaku.

Na trhu je k dostání ve dvou variantách:

**BMP085** Jedná se o samostatné čidlo. Tato varianta je levná, nicméně k připojení čidla je potřeba obstarat další potřebné součástky a rozumně je propojit s čidlem.

**BMP085–breakout** Toto řešení je vhodnější pro bezproblémové připojení, nevyžaduje totiž nic jiného, než přímé propojení čidla na piny mikrokontroléru bez nutnosti přidávat další součástky. Ty jsou totiž zabudované na malém plošném spoji spolu s čidlem. Bohužel jediné tato varianta čidla se prodává v našich končinách a je několikanásobně dražší.

Následníkem BMP085 je BMP180, které má téměř stejné parametry i rozhraní jako jeho předchůdce. Je však menší, levnější, dá se koupit již s přídatnými součástkami na plošném spoji, ale na trhu je pouze krátkou dobu. Proto není součástí této práce.

## 2 VÝBĚR SOUČÁSTEK

### 2.2.2.1 Specifikace čidla

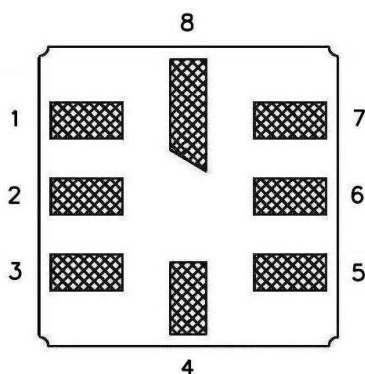
- rozsah - 300–1100 hPa (–500–9000 m nad mořem)
- největší rozlišení je 0.03 hPa
- operační rozsah teplot: –40–85 °C, ±2 °C
- I<sup>2</sup>C rozhraní s výstupem na 2 piny
- napájecí napětí 3.3–5.0 V

**2.2.2.2 Rozhraní čidla** Komunikace čidla s mikrokontrolérem probíhá přes dvoudrátové rozhraní I<sup>2</sup>C. Frekvence hodinového signálu přitom může dosahovat až 3.4 Mbit/s. SCL a SDA piny potřebují pull-up rezistory, typicky 4.7 kOhm napojené na VDD pin. I<sup>2</sup>C sběrnice je použita pro kontrolování senzoru, čtení kalibrovaných dat z E<sup>2</sup>PROM a čtení naměřených hodnot, pokud A/D převodník již ukončil převod. BMP085 obsahuje také XCLR pin pro reset čidla. Jinak je čidlo resetováno automaticky při připojení napájecího napětí.

V tabulce 2.7 a na obrázku 2.6 jsou uvedeny jednotlivé funkce pinů čidla.

PIN	Jméno	Funkce	Typ
1	GND	zem	napájení
2	EOC	konec konverze	digitální výstup
3	V <sub>DDA</sub>	napájecí napětí	napájení
4	V <sub>DDD</sub>	digitální napájecí napětí	napájení
5	NC	–	
6	SCL	čítač sériové sběrnice I <sup>2</sup> C - vstup	digitální vstup
7	SDA	data sériové sběrnice I <sup>2</sup> C	digitální - obousměrný

Tabulka 2.7: Funkce pinů teplotního a tlakového čidla BMP085-breakout [2]



Obrázek 2.6: Funkce pinů teplotního a tlakového čidla BMP085 [2]

## 2 VÝBĚR SOUČÁSTEK

**2.2.2.3 Výpočet nadmořské výšky** Pomocí naměřeného tlaku  $p$  a hodnoty tlaku u hladiny moře  $p_0$ , tj. 1013.25 hPa, se počítá nadmořská výška podle následujícího vzorce 2.3. Vzorec se používá jako mezinárodní standard pro výpočet nadmořské výšky pomocí tlaku [2].

$$nv = 44330 \times \left(1 - \left(\frac{p}{p_0}\right)^{\frac{1}{5.255}}\right) \quad (2.3)$$

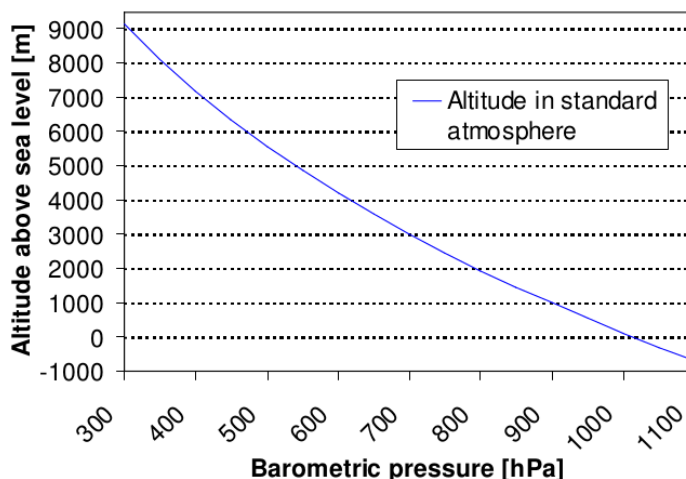
Kde:

$nv$  – je nadmořská výška (m)

$p$  – je absolutní tlak (Pa)

$p_0$  – je tlak vztažený k hladině moře (Pa)

Z toho plyne, že změna 1 hPa odpovídá zhruba změně 8.43 m v nadmořské výšce. Na grafu 2.7 lze vidět závislost nadmořské výšky na tlaku.



Obrázek 2.7: Závislost nadmořské výšky na tlaku [2]

**2.2.2.4 Výpočet tlaku** Podle změřené hodnoty tlaku a zjištěné nadmořské výšky se dá udělat přepočítání na tlak vztažený ke hladině moře podle následujícího vzorce. Je to vlastně pouze vyjádření  $p_0$  z rovnice 2.3.

$$p_0 = \frac{p}{\left(1 - \frac{nv}{44330}\right)^{5.255}} \quad (2.4)$$

Kde:

$nv$  – je nadmořská výška (m)

$p$  – je absolutní tlak (Pa)

$p_0$  – je tlak vztažený k hladině moře (Pa)

## 2 VÝBĚR SOUČÁSTEK

---

### 2.2.3 Analogová čidla

Tato čidla jsou převzata z meteostanice WH1080 [3]. Měřím pomocí nich směr a rychlost větru a srážky. Díly jsou složeny z plastových částí, aby vydržely i v nepříznivém počasí. Ve dvou dílech (srážkoměru a anemometru) se nacházejí elektromagnetické relé pro vybuzení elektrického impulsu.

**2.2.3.1 Srážkoměr** Srážkoměr funguje na principu dvou samosevyprazdňujících misek. Každých cca 0.28 mm srážek způsobí uzavření kontaktu spínače a vyslání signálu, což se dá měřit pomocí přerušování mikrokontroléru na Arduinu. Spínač je napojen dvěma drátky na dvě prostřední drážky výstupního konektoru RJ11.

**2.2.3.2 Miskový anemometr** Anemometr je miskového typu a funguje na stejném principu jako srážkoměr. Při rotaci lopatek se sepne magnetický spínač, který vyšle signál do mikrokontroléru. Signál se dá opět zaznamenat pomocí přerušování (IRQ) na Arduinu. Čím větší je počet signálů zpracovaných za jednotku času, tím větší je také rychlost větru. Každé další sepnutí spínače za dobu jedné vteřiny signalizuje zvýšení rychlosti větru o 2.4 km/h.

Signál anemometru je veden pomocí dvou drátků, které jsou připojeny na konektor RJ11. Tento konektor je sdílen se směrovou růžicí.

**2.2.3.3 Směrová růžice** Směrová růžice je nejsložitější ze tří analogových senzorů. Obsahuje osm spínačů, každý připojený na jiný rezistor. Má schopnost uzavřít až dva spínače najednou a tím pádem dokáže indikovat až šestnáct různých možných stavů natočení. U zapojení je využito externího 10 k $\Omega$  rezistoru.

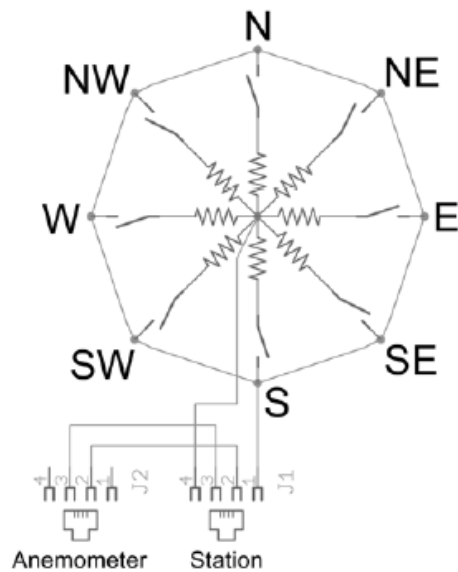
Hodnoty natočení růžice, příslušného odporu a výstupního napětí jsou uvedeny v tabulce 2.8.

Na obrázku 2.8 lze vidět, jak je připojena směrová růžice s anemometrem a jejich propojení přes konektor RJ11 do meteostanice (Arduina).

## 2 VÝBĚR SOUČÁSTEK

Směr (°)	Odpor (Ω)	Napětí (V, U = 5 V, R = 10k Ω)
0	33 k	3.84
22.5	6.57 k	1.98
45	8.2 k	2.25
67.5	891	0.41
90	1 k	0.45
112.5	688	0.32
135	2.2 k	0.90
157.5	1.41 k	0.62
180	3.9 k	1.40
202.5	3.14 k	1.19
225	16 k	3.08
247.5	14.12 k	2.93
270	120 k	4.62
292.5	42.12 k	4.04
315	64.9 k	4.78
337.5	21.88 k	3.43

Tabulka 2.8: Výstupy směrové růžice při různém natočení [3]



Obrázek 2.8: Schéma zapojení anemometru a směrové růžice. Směr natočení je označen anglickými písmeny [26].

## 2 VÝBĚR SOUČÁSTEK

---

### 2.2.4 Kalibrace čidel

Poté, co jsem měl hotovou softwarovou část meteostanice, nacházel jsem se ve stavu, kdy jsem zpracovával její výstupní data. Nevěděl jsem však, jestli jsou mé výsledky správné a zda je vše nastaveno tak, jak má. Proto jsem začal pátrat po možnostech kalibrace čidel. Nejschůdnějším řešením byla kalibrace čidel u nejbližší téměř profesionální meteostanice na Fakultě bezpečnostního inženýrství VŠB (FBI). Fakulta sídlí ve Ostravě-Výškovicích a nachází se kousek od mého bydliště. Na této fakultě je nainstalována meteostanice VantagePro2+ společnosti DAVIS. Stanice měří směr a rychlost větru, teplotu a vlhkost ve výšce 2 m nad zemí a ve výšce 0,6 m nad zemí, UV záření, solární radiaci a srážky. Data naměřená meteostanicí jsou dostupná na webových stránkách fakulty.

Jelikož se anemometr a směrová růžice nachází na střeše fakulty a není k ní vhodný přístup, byl jsem nucen kalibrovat pouze digitální čidla. Nejprve jsem se domluvil s panem prof. Dudáčkem a domluvil si přístup do dvoru fakulty, kde se meteostanice nachází. Den kalibrace nastal v sobotu 15.5.2014.

Abych mohl spravovat data z čidel na Raspberry Pi, nainstaloval jsem si program „Tiny DHCP server“ na netbook. Tím jsem byl schopen dostat se na něj přes protokol SSH a program Putty a případně WinSCP. Kalibroval jsem teplotu, vlhkost a tlak a jako referenční čidla jsem uvažoval čidla meteostanice VantagePro2+ ve výšce 0.6 m nad zemí, jelikož se v podobné výšce vyskytovala také má čidla. Zpočátku se výsledky výrazně lišily, nicméně to bylo způsobeno špatnou rovnicí na mé straně u přepočtu atmosférického tlaku. Rozdíly hodnot stanice VantagePro2+ a mých čidel jsou uvedeny v následující tabulce:

Měřená veličina	max. rozdíl hodnot
teplota	0.5 °C
vlhkost	1 %
tlak	2 hPa

**Tabulka 2.9:** Maximální odchylky mých čidel a čidel stanice VantagePro2+

Měření probíhalo cca 3 hodiny. Z tabulky 2.9 je zřetelná přesnost digitálních čidel připojených na Raspberry Pi v tomto krátkém časovém intervalu. Díky špatnému počasí a zvyšující se intenzitě srážek jsem byl nucen opustit stanoviště kalibrace, aby Raspberry Pi, netbook a čidla zůstala funkční. Pan prof. Dudáček mi sice nabídl test funkčnosti tlakového čidla a srovnání výsledků v přetlakové komoře, ale srovnání výsledků s jejich meteostanicí jsem považoval za dostatečně přesné.

### 3 Komunikace

Jádrem celé diplomové práce a také její stěžejní částí je komunikace mezi jednotlivými částmi meteostanice. V následující kapitole bude popsána komunikace mezi všemi prvky meteostanice, kterými jsou čidla a počítače, databáze, síťový protokol SNMP a webové rozhraní Cacti. Na obrázku 3.1 je vidět blokové schéma a propojení všech prvků systému meteostanice.

#### 3.1 Analogová čidla – Arduino

Komunikace s analogovými čidly je poměrně jednoduchá, jelikož čidla sama o sobě nepodporují žádné sofistikovanější komunikační rozhraní. Čidla anemometru a srážkoměru komunikují s počítačem na principu posílání signálu do mikrokontroléru. Signály se dají odchytnout buď odchytem pomocí digitálního čítače nebo pomocí přerušení mikrokontroléru. Rozhodl jsem se pro implementaci druhé varianty pomocí přerušení, protože Arduino tuto variantu podporuje a má pro ni implementovanou knihovnu. Obě dvě přerušení jsou nastaveny na zachycení sestupné hrany signálu.

---

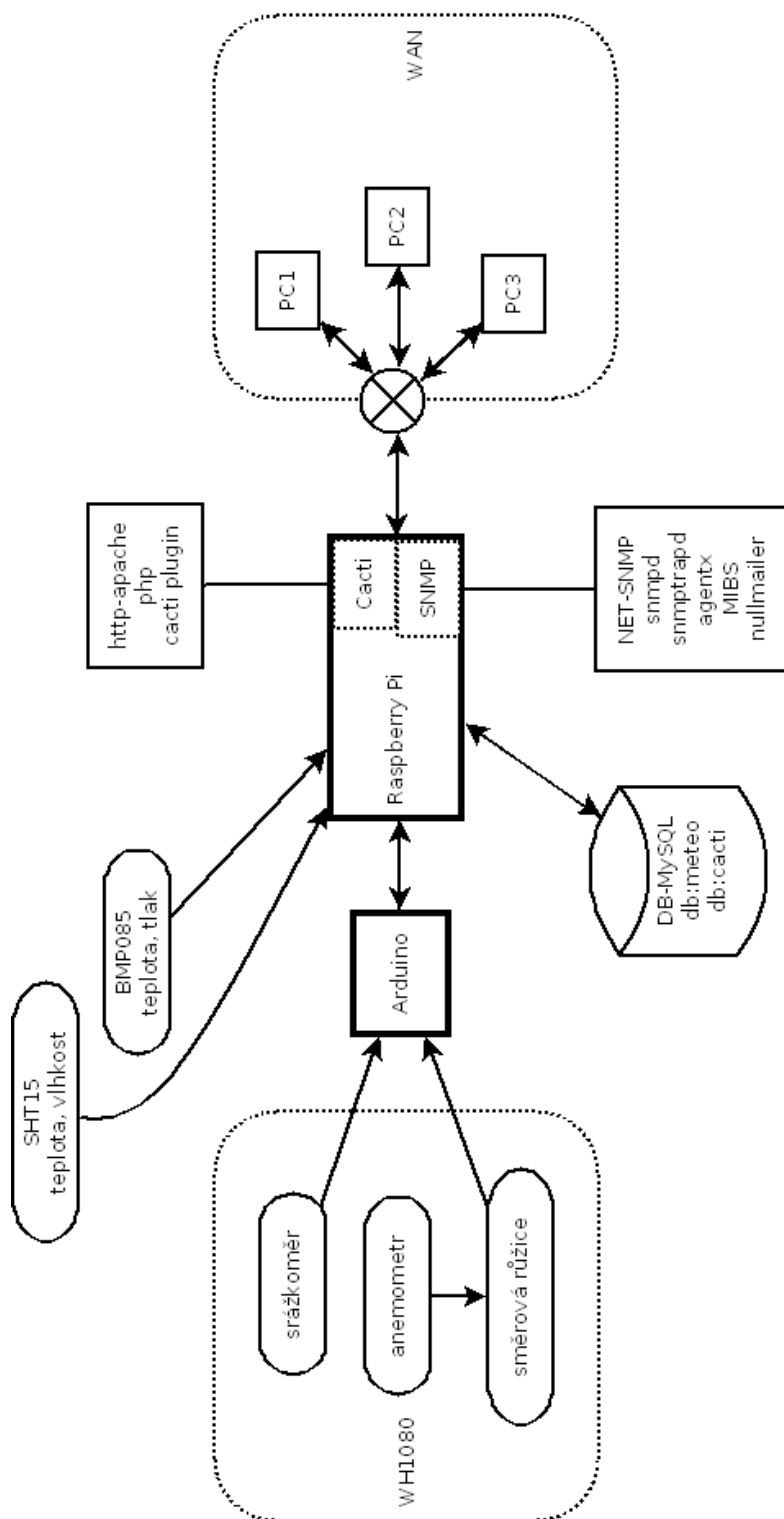
```
void setupWeatherInts()
{
  pinMode(ANEMOMETER_PIN,INPUT);
  digitalWrite (ANEMOMETER_PIN,HIGH); // Turn on the internal Pull Up Resistor
  pinMode(RAIN_GAUGE_PIN,INPUT);
  digitalWrite (RAIN_GAUGE_PIN,HIGH); // Turn on the internal Pull Up Resistor
  pinMode(VANE_PWR,OUTPUT);
  digitalWrite (VANE_PWR,LOW);
  attachInterrupt (ANEMOMETER_INT,anemometerClick,FALLING); //Set interrupt for anemometer
  attachInterrupt (RAIN_GAUGE_INT,rainGageClick,FALLING); //Set interrupt for rain_gauge
  interrupts ();
}
```

---

**Výpis 1:** Ukázka kódu nastavení přerušení a dalších pinů na Arduino

Směrová růžice upravuje výstupní napětí podle směru natočení a zapojení příslušných rezistorů. Její výstupní napětí je vyvedeno na analogový pin A1, ze kterého jsou čteny výsledné hodnoty napětí. Pin A1 se tedy chová jako vstupní pin. Hodnota napětí se z tohoto pinu převede pomocí 10bitového převodníku na číselné hodnoty, jejichž rozmezí je 0–1024. Tím se dají rozlišit rozdíly hodnot vstupního napětí až 4.9 mV. Tyto číselné hodnoty jsou namapovány na hodnoty směru natočení. Délka čtení hodnoty analogového pinu je 100 ms. Schéma zapojení je k vidění v příloze A [11].





Obrázek 3.1: Blokové schéma zapojení prvků meteorostanice

## 3 KOMUNIKACE

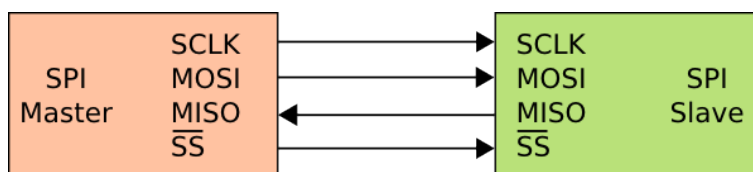
### 3.2 Digitální čidla – Raspberry Pi

Komunikace s digitálními čidly není až tak triviální jako komunikace s analogovými čidly, umožňuje však více možností nastavení. Je také lépe ovladatelná přes rozhraní, která jsou k tomuto účelu vytvořena – SPI a I<sup>2</sup>C. Obě čidla se také připojují přes několik pinů, také proto bylo nutné připojit některá čidla na Arduino a jiná na Raspberry Pi, jelikož počet vstup-výstupních pinů je na obou zařízeních omezený.

#### 3.2.1 SHT15 – Raspberry Pi

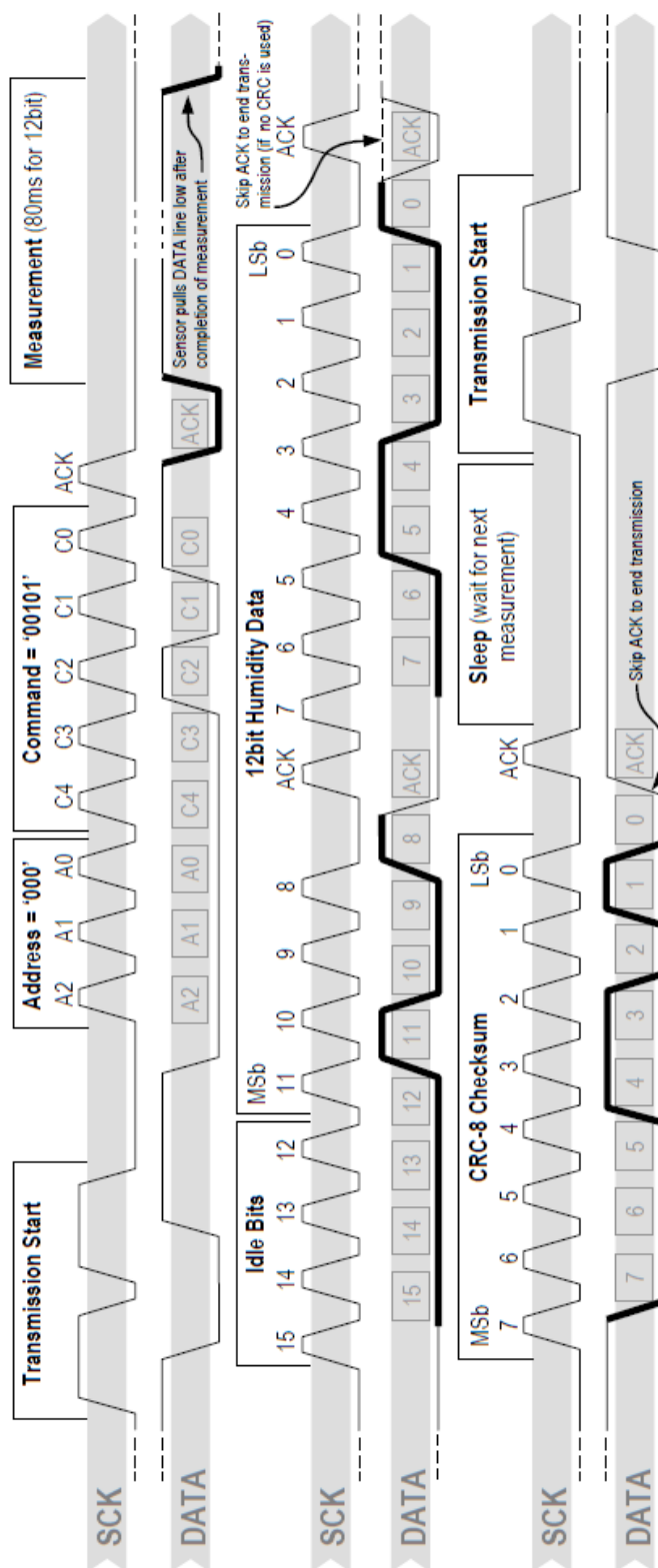
Čidlo SHT15 používá pro komunikaci SPI. Komunikace je realizována pomocí stejné sběrnice. SPI se vyznačuje tím, že využívá minimálně 4 piny.

- SCLK – serial clock - signál využívaný pro synchronizaci spojení
- MOSI – Master Output, Slave Input – příjem dat mikrokontroléru
- MISO – Master Input, Slave Output – příjem dat čidla
- $\overline{SS}$  - výběr zařízení, v mém případě pouze jedno čidlo



Obrázek 3.2: Standardní schéma SPI s jedním slave zařízením [27]

Na obrázku 3.3 lze vidět časový diagram komunikace mikrokontroléru s čidlem SHTx. Z obrázku je patrné, jak se posílají příkazy za sebou. Jako první se zahájí přenos dat, což je znázorněno blokem „Transmission Start“. Jako další následuje sekvence bitů nastavující adresu - „Address“ a příkaz čidlu, který může nabývat dvou hodnot - měření teploty nebo vlhkosti. Poté je již čidlo připraveno k měření a po úspěšném nastavení ACK bitu následuje fáze, kdy mikrokontrolér musí čekat, až v čidlu proběhne měření. Po této sekvenci se přečtou 4 nepodstatné bity, po nichž následuje čtení hodnot z čidla. Velikost dat záleží na nastavené kvalitě měření, čím přesnější, tím se posílá více dat. Po datech se posílá volitelný CRC součet, který ale nemusí být implementován například pro zrychlení procesu. Po těchto operacích se čidlo dostane do tzv. „spacího režimu“, aby nespotřebovávalo zbytečně elektrickou energii. Kód pro toto čidlo je napsán v jazyce C, využívající knihovnu pro SPI.

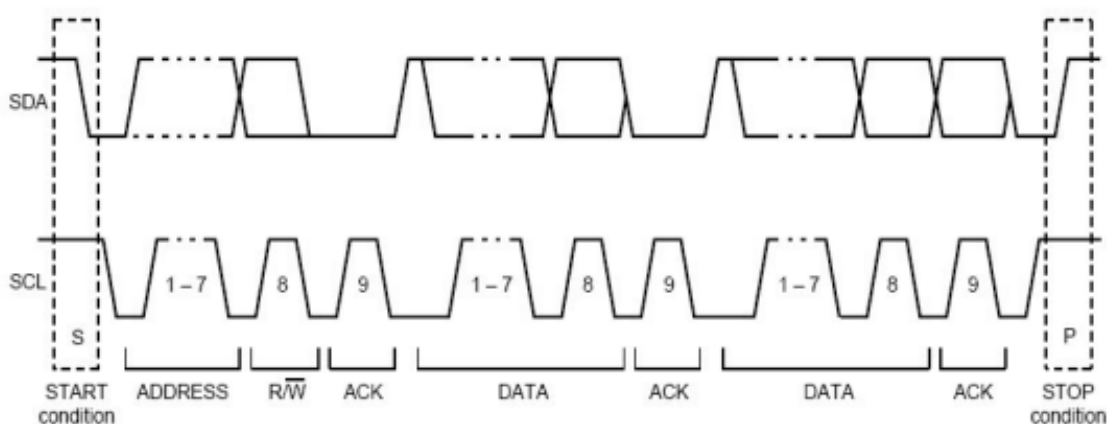


Obrázek 3.3: Časový diagram komunikace s SHTx. SCK je hodinový signál, DATA symbolizují přenos dat v binární formě [8].

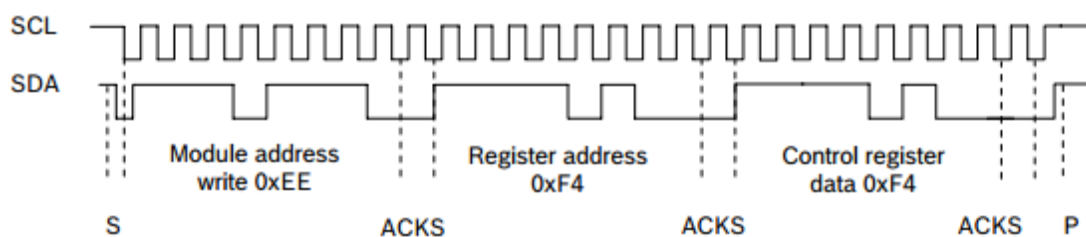
### 3 KOMUNIKACE

#### 3.2.2 BMP085 – Raspberry Pi

Tento senzor se připojuje pomocí rozhraní I<sup>2</sup>C. Adresa senzoru je 0x77. I<sup>2</sup>C rozhraní má speciální počáteční a ukončující podmínky pro přenos dat. Pro start je využito nastavení hodinového signálu na 1, zatímco datový signál se změní z 1 na 0. Následuje posílání adresy o 7 bitech a typu operace - čtení nebo zápis dat. Po této operaci by mělo „slave“ zařízení rozpoznat a potvrdit komunikaci tím, že datový signál nastaví na 0 v devátého cyklu hodinového signálu. Ukončující podmínka je nastavení hodinového signálu na 1 a změna datového signálu z 0 na 1. V této fázi musí data na datovém vodiči zůstat neměnná, změna dat může probíhat pouze tehdy, jestliže je SCL signál nastaven na 0. Časový diagram rozhraní I<sup>2</sup>C je znázorněn na obrázku 3.4.



Obrázek 3.4: Časový diagram I<sup>2</sup>C senzoru BMP085 [2]



Obrázek 3.5: Časový diagram měření teploty a vlhkosti senzoru BMP085 [2]

Kde:

S – je start

P – je stop

ACKS – acknowledge by slave - potvrzení od zařízení „slave“

### 3 KOMUNIKACE

Diagram měření teploty a vlhkosti je znázorněn na obrázku 3.5. „Master“ zařízení posílá data v tomto pořadí: adresa modulu, adresa registrů, data kontrolních registrů.

Měření	Hodnota kontrolního registru 0xF4	Maximální doba převodu [ms]
Teplota	0x2E	4.5
Tlak	0x34	4.5
Tlak	0x74	7.5
Tlak	0xB4	13.5
Tlak	0xF4	25.5

Tabulka 3.1: Hodnoty kontrolních registrů u různých nastavení senzoru [2]

**3.2.2.1 Čtení výsledku A/D převodníku nebo E<sup>2</sup>PROM dat** Čtení datového slova teploty (16 bitů) nebo datového slova tlaku (16 až 19 bitů) probíhá následovně.

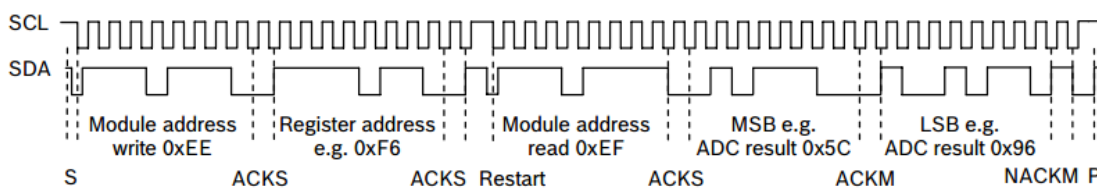
Po splnění startovní podmínky posílá „master“ zařízení požadavek na zápis adresovaný modulu. Poté pošle adresu registrů. Tím se realizuje výběr ze dvou registrů:

- E<sup>2</sup>PROM datový registr - 0xAA až 0xBF
- hodnota teploty nebo tlaku - 0xF6 (MSB), 0xF7 (LSB), volitelně 0xF8 (XLSB)

Poté „master“ zařízení pošle příkaz pro restart následovaný čtením z adresy modulu, který je následně potvrzen čidlem. Čidlo posílá prvních 8 MSB bitů, „master“ zařízení potvrzuje. Dále se pošle 8 LSB bitů a „master“ odpoví NACK.

Volitelně může být navíc čten XLSB registr pro zvětšení přesnosti. Pak se ale délka přenášených dat změní z 16 na 19 bitů.

Časový diagram čtení výsledných 16 bitů A/D převodníku lze vidět na obrázku 3.6.



Obrázek 3.6: Časový diagram měření teploty a vlhkosti senzoru BMP085 [2]

Kde:

S – je start

P – je stop

ACKS – acknowledge by slave - potvrzení od zařízení „slave“

ACKM – acknowledge by master - potvrzení od zařízení „master“

NACKM – not acknowledge by master - nepotvrzeno od zařízení „master“

### 3.3 Zpracování dat na Arduinu a Raspberry Pi

To, že meteorostanice umí vyčítat data z analogových a digitálních čidel, je její základní funkcí. Aby bylo možné s těmito daty dále pracovat, je však potřeba propojit data s několika dalšími prvky a službami. V této sekci bude popsáno, jak funguje zpracování meteorologických dat a jejich distribuce do SNMP a Cacti.

#### 3.3.1 Arduino – Raspberry Pi

Arduino podporuje posílání dat pomocí sériového rozhraní, které je konvertováno na připojení přes USB konektor. Toho jsem využil při komunikaci s Raspberry Pi. Arduino má na totu formu komunikace dobře zpracovanou knihovnu, a proto stačí pouze otevřít sériovou linku a pomocí funkce *Serial.println()* poslat požadovaná data. Takto je vyřešena komunikace na straně Arduina.

U Raspberry Pi bylo potřeba naprogramovat v nějakém jazyce příjem dat. Využil jsem proto jednoduchou knihovnu v Pythonu [20] a naprogramoval skript, který čeká na sériovém portu od Arduina (/dev/ACM0). Po zaplnění sériové linky minimálně 25 bajty se přečte vše, co je připraveno ke zpracování. Počet 25 bajtů není volen náhodně, jelikož Arduino posílá v pravidelných 3 vteřinových intervalech právě  $25 \pm 2$  bajty. Pokud není připraven dostatečný počet dat na sériovém rozhraní, skript se na vteřinu uspí. Po příjmu dat se data pošlou do MySQL databáze [22], ukládá se kromě typu čidla a jeho hodnoty také časová značka, aby bylo možné vyčíst nejaktuálnější data. Tím je řešena komunikace na straně Raspberry Pi.

#### 3.3.2 Raspberry Pi - SNMP

Přenesení aktuálních dat z databáze na SNMP rozhraní se děje pomocí dalšího Python skriptu. Ten je spuštěn samostatně a běží na pozadí, chová se tudíž podobně jako tzv. „démon“<sup>3</sup> na operačním systému Linux. Skript se periodicky dotazuje databáze a získává z nich aktuální data ze všech čidel. Ty pak posílá pomocí rozšiřujícího protokolu od SNMP – AgentX. Využil jsem již napsané knihovny AgentX v Pythonu. Skript se vlastně chová jako subagent vůči hlavnímu agentovi snmpd. Skript si na začátku svého běhu zaregistruje proměnné v meteo MIB<sup>4</sup> souboru a periodicky do nich ukládá aktuální data. Tím jsou data dostupná v SNMP hierarchii.

#### 3.3.3 Raspberry Pi - Cacti

Cacti je open source webové rozhraní tvořené v jazyce php. Využívá se pro vizualizaci dat a já jsem ho využil pro tvorbu grafů z hodnot jednotlivých meteorologických čidel. Cacti se pyšní velkým počtem šablon pro zpracování dat, nicméně já jsem využil pouze jednu – šablonu pro SNMP. Zjednodušeně má Cacti nastaveno periodické dotazování na část OID<sup>5</sup>

---

<sup>3</sup>démon - program běžící na pozadí operačního systému

<sup>4</sup>MIB - soubor pro správu a definici SNMP objektů - tzv. „management information base“

<sup>5</sup>OID - číselná reprezentace datové struktury SNMP protokolu

### 3 KOMUNIKACE

---

podstromu od SNMP, ve kterém se vyskytuje MIB<sup>4</sup> meteostanice. Tím se dostanou data ze SNMP do Cacti a automaticky se zpracovávají do grafů. Jediným problémem u posílání dat z SNMP do Cacti jsou datové typy. SNMP verze 2 nepodporuje čísla s desetinnou čárkou. Tento problém se dá řešit několika způsoby:

- zaokrouhlení hodnoty na celé číslo, což je velmi nevýhodné např. u teploty, u které jsou důležité i desetiny stupně Celsia
- převedení čísla na datový typ „STRING“ - není možné, protože Cacti nepodporuje „STRING“
- záměrné vynásobení hodnoty konstantou  $10^k$ , kde  $k$  je počet desetinných míst, které chceme uchovat

Byl jsem nucen vybrat si poslední volbu jako řešení a to pak reflektovat ve zpracování dat ve skriptu, který se stará o komunikaci SNMP a Cacti.

#### 3.3.4 Nullmailer - mobilní zařízení

V sekci komunikace nesmím opomenout metodu posílání varovných zpráv na můj mobilní telefon. Posílání se dá samozřejmě přenastavit pro příjem na jakékoliv jiné mobilní zařízení. Stačí mít dobře nastavené posílání chybových zpráv z SNMP, mail, zaplé zařízení pro práci s SMS (mobil, GSM modul...) a mít aktivovanou SIM kartu. Při návrhu realizace meteostanice jsem se dostal do situace, kdy jsem potřeboval vědět, jestli funguje zpracování všech dat tak, jak má. Může se stát, že čidlo špatně zpracuje signál, pošle špatná data nebo třeba vypadne drátek ze vstup-výstupního pinu na jednom z počítačů. Ve všech těchto případech se hodí vědět o těchto problémech co nejdříve. Proto jsem vytvořil řešení, které automaticky posílá chybné zprávy na můj mobilní telefon ihned po detekci nějaké chyby. Posílání varovných zpráv jsem realizoval takto:

1. Nastavení snmptrap – po detekci problému se spustí skript, který pošle mail
2. Instalace a nastavení programu nullmailer
3. Přeposlání zprávy na školní mail
4. Přeposlání zprávy na mobilní telefon

Systém funguje tak, že démon<sup>4</sup> pro odchycení chybových zpráv (snmptrapd) zachytí nějaký problém. Potom předá zprávu jako řetězec znaků jednoduchému skriptu napsanému v bashi<sup>5</sup>. Tento skript dále spolupracuje s programem nullmailer, který zprávu předá dál a pošle na nastavený SMTP mailový server. K dispozici může být např. účet na gmail, jelikož jsou k němu k dispozici informace pro připojení - port, IP adresa, poslání přihlašovacích údajů, podpora šifrování spojení atd. Z tohoto serveru se zpráva pošle rovnou na mobilní telefon anebo přepošle na jiný server, který tuto službu podporuje.

---

<sup>5</sup>Bash je interpret příkazového řádku na unixových strojích.

### 3 KOMUNIKACE

---

Detekce chyb je nastavena na dva případy:

- absolutně mylná výsledná hodnota čidla – neodpovídá požadovanému rozsahu (např. rychlost větru 100 m/s v našich podmínkách)
- nečinnost čidla, rozpojení, ztráta signálu, neaktuální data – vyhodnocení jako chyba po 10 hodinách stále stejné hodnoty – všechna čidla kromě srážkoměru

#### 3.4 Stručný výpis programů

Na následujících řádcích bude stručně popsána funkčnost všech programů/skriptů podílejících se na komunikaci. Všechny programy se vztahují k sekci 3. Programy jsou spustitelné zvlášť, nicméně pro správnou funkci všech programů je potřeba mít spuštěné všechny a také mít na stroji aktivované příslušné služby (MySQL, httpd, snmp). Některé předem neznámé pojmy budou vysvětleny zvlášť v nadcházejících kapitolách 4 a 5, aby nezabíraly v sekci „Komunikace“ příliš mnoho místa, jelikož logicky nepřímo souvisí s textem.

##### 3.4.1 AgentX

S protokolem AgentX je spojen skript „meteo-agentx.py“. Na začátku skript načítá informace z konfiguračního souboru „meteo.conf“. Poté se zaregistruje příslušná MIB a hlavní funkce a předá se běh programu knihovně AgentX, která se periodicky dotazuje na data. To funguje tím způsobem, že se v hlavní funkci skript dotazuje přes MySQL knihovnu na databázi, ze které čerpá aktuální data a posílá je přes AgentX do MIB. Tím je splněna fáze plnění SNMP a to tak, že se v něm vyskytují pouze aktuální data.

Další funkcí skriptu je posílání upozornění na neobvyklé události (TRAP). To probíhá tak, že skript kontroluje požadovaná data v hlavní funkci a pokud nastane problém, spustí terminálový příkaz pro vyvolání TRAPů a jako argument pošle hodnotu TRAPu. Jedná se o standardní příkaz Net-SNMP a jde o „snmptrap“.

##### 3.4.2 Data z analogových čidel

Program pro sběr dat z analogových čidel se jmenuje „analog\_senzors“ a je psaný v čistém C. Běží na Arduinu a proto ani v jiném jazyce nemohl být napsán. Některé části zdrojového textu lze najít na internetu, nicméně nikde jsem nenašel celý funkční kód a navíc většina kódu byla pro typ MEGA<sup>6</sup> namísto mého typu Leonardo. Proto jsem musel zdrojový kód upravovat a dodělat. Po úpravách program funguje správně.

Program využívá přerušení dostupných na Arduinu. Ve funkcích zpracovávajících přerušení se pouze přepočte počet přerušení za jednotku času a pomocí vzorečku vypočte výsledná rychlost větru a počet srážek. Směr větru se získá přečtením analogové hodnoty

---

<sup>6</sup>typ počítače Arduino MEGA a Leonardo se liší hlavně typem mikrokontroléru, dále pak počtem vstup-výstupních pinů



## 3 KOMUNIKACE

---

napětí na pinu A1 a její přiřazení hodnotě ve stupních. Podobné je to se slunečním svitem, u kterého pouze přečtu hodnotu napětí a vyhodnotím, zda slunce svítí, či nikoliv.

Funkce pro čtení dat jsou volány v hlavní funkci „main“ za sebou a následně jsou odeslány přes sériové rozhraní napojené na USB.

### 3.4.3 SHT15

Program pro digitální čidla je složitější a pro SHT15 je rozdělen na 2 soubory napsané v C a jejich hlavičkových souborech. V prvním souboru „Raspberry Pi\_SHT1x.c“ se vyskytují tyto funkce:

- výpočet teploty a vlhkosti
- kontrola crc
- inicializace pinů
- reset
- počátek komunikace
- čtení a zápis jednoho bajtu
- počátek měření
- čtení naměřené hodnoty

Tato část programu je myšlena jako knihovna pro práci s SHT15 na Raspberry Pi. Dalším souborem je „testSHT1x.c“. V tomto souboru se nachází hlavní funkce „main“ a další dvě funkce. Jedna na čtení výsledné teploty a vlhkosti a další pro zápis dat do databáze.

### 3.4.4 BMP085

Zdrojový kód pro toto čidlo se nachází v souboru „bmp085.c“ a vyskytují se v něm všechny potřebné funkce. Opět je zde hlavní funkce „main“, funkce pro ukládání dat do databáze a další funkce pro:

- kalibraci čidla po spuštění
- čtení a zápis jednoho bajtu
- otevření zařízení pro komunikaci
- čtení teploty a čtení tlaku

### 3.4.5 Příjem dat z Arduina

Posledním skriptem je ten, který se zabývá příjmem dat z Arduina a jmenuje se „meteo-analog-fill-db.py“. Je psán v Pythonu, jak jde vidět z koncovky souboru. Po spuštění skript načte informace z konfiguračního souboru. Poté se inicializuje spojení s databází a se sériovým portem. Nakonec se spustí funkce se smyčkou. Tato funkce jednoduše čeká na otevřeném sériovém portu, přijme data, pokud jich čeká dostatek, a ty následně uloží do databáze. Takhle je vyřešeno ukládání všech měřených veličin.

### 4 SNMP

SNMP je zkratka pro Simple Network Management Protocol. Původně se využíval pro řízení a sběr dat v počítačových sítích. Dnes se tento protokol hojně rozšířil a uplatňuje se při řízení velké množiny nejen síťových prvků. Jeho cílem je minimalizace náročnosti správy sítě a jejích prvků. Mezi další hlavní cíle SNMP patří:

- jednoduchost, nenáročnost a nízká cena řídicího softwaru
- zvyšující se podpora vzdálené správy protokolu a volání funkcí z internetové sítě
- důraz na jednoduchou strukturu
- zjednodušení funkcionality pro jednoduché pochopení dalších vývojářů internetových aplikací

Vedlejšími cíli jsou rozšíření funkcionálního paradigmatu tak, aby splňoval stále novější požadavky technologického vývoje sítí a nezávislost protokolu na provozní architektuře a platformě.

Protokol SNMP je asynchronní protokol založený na modelu klient/server [9]. Klientskou stranou může být webové rozhraní, složitý NMS systém či pouze skript s výpisem dat na standardní výstup. Jako server běží většinou na nějakém počítači – tzv. „snmp agent“, který odpovídá na dotazy klientovi. Jedinou výjimkou je tzv. „snmp trap“, což je neočekávaná událost, na kterou trapový agent reaguje tím, že pošle klientovi asynchronní zprávu ihned po zaznamenání problému. Pro přenos dat se používá protokol UDP (vyšší verze i TCP) a standardně běží na portu 161, posílání TRAPů pak na portu 162.

#### 4.1 Historie SNMP

Protokol vznikl na konci 80. let pro správu síťových prvků. Vyvinul se z protokolu SGMP. V roce 1989 byl protokol formalizován a od té doby slouží jako standard pro správu síťových prvků jako jsou směrovače, přepínače, tiskárny, telefonní ústředny atd. V současné době existují tři varianty SNMP protokolu - SNMPv1, SNMPv2 a SNMPv3. Důvodem pro vyvinutí dalších verzí protokolu byla hlavně bezpečnost a kvalitnější zpracování dat. Přesný popis protokolu se nachází v sadě dokumentů – tzv. „RFC“, tyto dokumenty jsou číslovány a je v nich popsána většina existujících síťových protokolů, se kterými se lze dnes setkat. Jenom samotný SNMP protokol a jeho všechny verze jsou popsány v přibližně 35 RFC dokumentech [10].

### 4.2 Architektura SNMP

V komunikaci zařízení pomocí SNMP se vyskytuje několik základních prvků:

- management station - stanice pro správu systému, vyhodnocování výsledků, GUI
- agent (démon) - realizuje komunikaci, čeká na dotazy, zasílá odpovědi nebo TRAPy
- zařízení - komunikuje s agentem, posílá mu data a informace o sobě

#### 4.2.1 MIB

MIB je zkratka pro management information base, což je datová hierarchická struktura, ve které se ukládají definice objektů pro popis zařízení. Je to vlastně soubor v textovém formátu, ve kterém jsou definovány objekty (proměnné), které obsahují informace o daném zařízení. MIB struktura je popsána pomocí SMI, což je podmnožina jazyka ASN.1 [12]. V MIB jsou pouze definice objektů, které agent po startu vytvoří a ukládá do nich data.

---

```
METEOSTATION-MIB DEFINITIONS ::= BEGIN
-- imports
IMPORTS
    MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE, enterprises, snmpModules
    FROM SNMPv2-SMI;

-- description and update information
meteostationMIB MODULE-IDENTITY
    LAST-UPDATED "201402270000Z"
    ORGANIZATION "N/A"
    CONTACT-INFO "web: http://www.vsb.cz"
    DESCRIPTION "A simple mib to demonstrate features of meteostation."
    ::= { snmpModules 90 }

windDirection OBJECT-TYPE
    SYNTAX INTEGER
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
    "Actual direction of wind."
    DEFVAL { 0 }
    ::= { meteostationMIB 0 }

windSpeed OBJECT-TYPE
    SYNTAX INTEGER
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
    "Actual speed of wind."
    DEFVAL { 0 }
    ::= { meteostationMIB 1 }
```

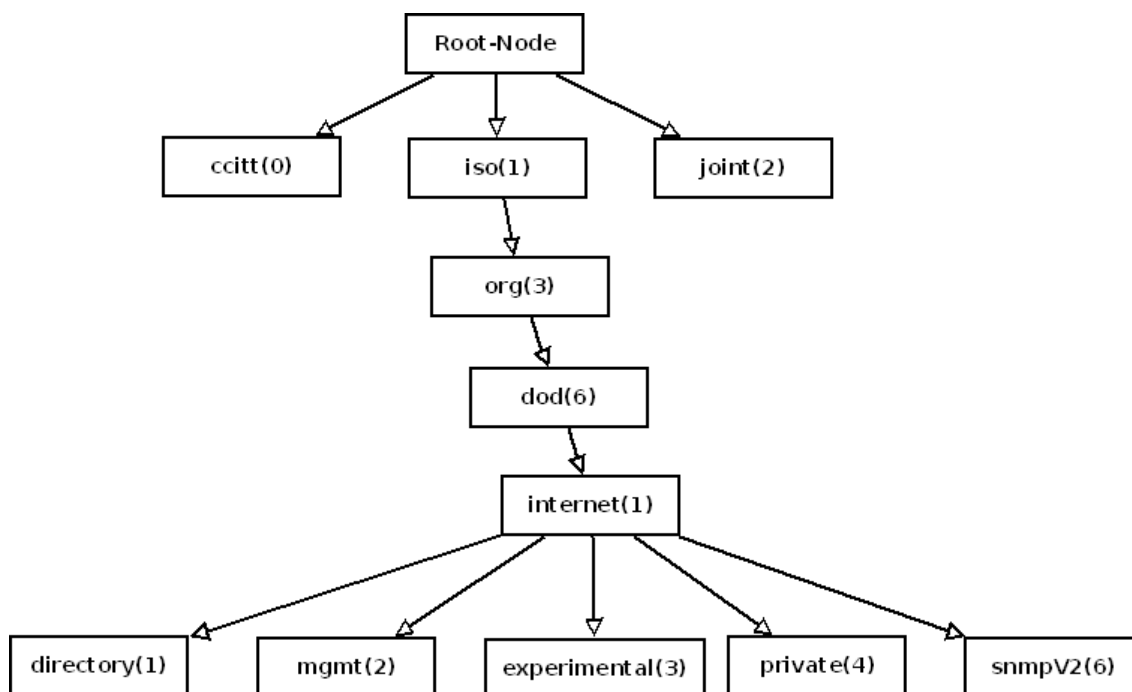
---

Výpis 2: Ukázka MIB souboru vytvořeného pro meteostanici

**4.2.1.1 OID** Každý objekt je v SNMP označen pomocí objektového identifikátoru (OID). Hodnota OID je tvořena posloupností čísel oddělených tečkou. OID tvoří stromovou strukturu. Hodnota vznikne tak, že se sepíší všechny hodnoty průchodu stromem od kořene až k dané MIB a oddělí se tečkami. Pro vytvoření vlastní meteo-MIB jsem se rozhodl, že využiji OID podstrom končící MIB snmpModules. Popis snmpModules je definován v IETF RFC 2578. Cesta od kořene k meteoMIB je následující:

iso(1) → identified-organization(3) → dod(6) → internet(1) → snmpV2(6) → snmpModules(3) → meteoMIB(90)

Na obrázku 4.1 je vidět základní struktura SMI SNMP podstromu. Jak je vidět, k uzlu se jménem „internet“ a jeho dalším následovníkům je možné přidávat vlastní MIB, čehož jsem využil právě při výrobě meteostanice.



Obrázek 4.1: SMI stromová struktura protokolu SNMP [4]

### 4.2.2 Net-SNMP

Jelikož na Raspberry Pi běží operační systém Raspbian, potřeboval jsem na něho nainstalovat specifický balík, který by dokázal spravovat SNMP. Jako nejvhodnější se mi jevil široce rozšířený balík Net-SNMP. Tento balík je volně dostupný z debianových repozitářů a výborně slouží pro správu meteorologických dat. Dokáže pracovat se všemi třemi verzemi SNMP protokolu a umí si poradit také s IPv4 a IPv6. Balík obsahuje tyto části:

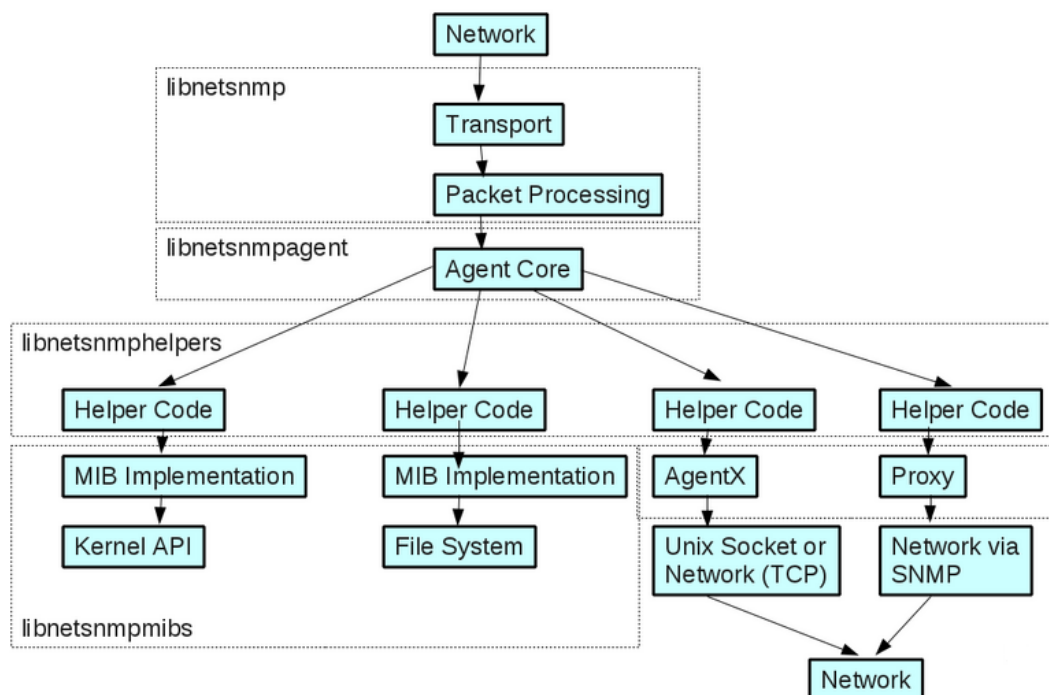
**Aplikace příkazového řádku** Aplikace využívající příkazů `snmpget`, `snmpwalk`, `snmptable`, `snmpset`, `snmptranslate` atd.

**Grafický prohlížeč MIB souborů** Prohlížeč, který používá Tk/Perl jako grafické rozhraní.

**Snmpttrapd** Démon pro zpracování upozornění (TRAP), které mohou být přesměrovány dále.

**Snmppd** Agent starající se o zaslání odpovědí na dotazy a komunikaci s ostatními subagenty a dalšími částmi SNMP. Agent může být rozšířen pomocí modulů, externími skripty a dvěma protokoly – SMUX a AgentX.

**Knihovna** Knihovna je vytvořena pro vývojáře a obsahuje API pro C i Perl.



Obrázek 4.2: Architektura balíku Net-SNMP [28]

### 4.2.3 AgentX

Využití SNMP protokolu zcela závisí na plnění jeho MIB objektů nějakými daty. Toho lze docílit několika způsoby, bohužel pouze jeden způsob umožňoval poskytování informací agentovi podle mých představ:

**Manuální vkládání** Jedná se o využití vkládání přímo z konzole pomocí příkazu `snmpset`.

**Perl rozšíření** Jelikož Net-SNMP obsahuje rozšiřující balíček pro napojení agenta `snmpd` na Perl, toto je další varianta. Jednoduše se vytvoří Perl skript, ve kterém se využije API pro práci s MIB a komunikaci se `snmpd` a pomocí předdefinovaných funkcí se posílají ze skriptu data do MIB objektů.

**Makro extend v `snmpd.conf`** Pomocí makra `extend` se dá nastavit, jaký skript se má spustit a jaké informace vrací. Bohužel s touto variantou počítá SNMP pouze tak, že data vrácená ze skriptu ukládá do obecné „NET-SNMP-EXTEND-MIB“. Z této MIB se dají data zobrazit např. pomocí příkazu „`snmptable`“, jelikož jsou v ní data uložena ve formě tabulky.

**Rozšíření pomocí protokolů** SMUX protokol je rozšíření, které bylo použito u vytváření UCD-SNMP<sup>7</sup> frameworku. SMUX funguje tak, že se přihlásí běžícímu `snmpd` agentovi pomocí identifikačního čísla a hesla. Poté si zaregistruje nějaké své MIB, které chce spravovat. Když `snmpd` dostane žádost o vrácení dat z této MIB, deleguje žádost na SMUX skript. Naproti tomu AgentX běží jako subagent a může být použit také jako tzv. „master agent“<sup>8</sup>.

Jak je vidět na obrázku 4.2, SNMP je rozšiřitelné pomocí protokolu AgentX. Právě AgentX protokol mě zaujal, jelikož jsem s ním měl zkušenosti již v minulosti a proto jsem se vydal touto cestou. Potřeboval jsem totiž, aby můj skript poskytoval informace o meteostanici nějaké konkrétní meteo-MIB a ne „NET-SNMP-EXTEND-MIB“, do které se ukládají všechny další výstupy skriptů makra `extend`. Knihovna pro AgentX je naimplementována jak v C, tak v Pythonu. V jazyce C je nicméně složitější, zatímco v Pythonu je přímočařejší a jednodušší. Proto jsem se rozhodl pro Python.

Zdrojové kódy knihovny AgentX jsou volně dostupné z internetových stránek `sourceforge.net`. Implementace celého AgentX je obsažena v jednom souboru, který je rozdělen na příslušné třídy. Knihovna využívá podpůrných knihoven z jazyka C pro správný převod datových typů. Souhrně se tyto knihovny nazývají „`ctypes`“. Jejich použití je nezbytně nutné, jelikož Net-SNMP je implementováno v čistém C a AgentX v Pythonu. AgentX je implementován tak, že vlastní řízení celého procesu registrace a posílání dat do MIB. Z tohoto důvodu není možné využít např. asynchronního zpracování událostí ve vlastním skriptu.

---

<sup>7</sup>Net-SNMP původně vzniklo na univerzitě v Kalifornii ve městě Davis, název projektu byl přejmenován až v roce 2000.

<sup>8</sup>Standardně se chová `snmpd` jako „master“, nicméně dá se nastavit i tak, aby se skript choval jako „master“ a `snmpd` jako „slave“.

### 4.2.4 Konfigurační soubory

Důležitou částí nastavení SNMP jsou konfigurační soubory. Základní konfigurační soubory jsou dva – `snmpd.conf` a `snmptrapd.conf`. Jak už název napovídá, první je zaměřen na nastavení agenta zpracovávajícího žádosti o poskytnutí dat. V tomto souboru se také nalézají informace o zabezpečení, vytvoření skupin a jejich přístupových práv, dále IP adresa a port běžícího agenta atd. Ve druhém souboru se vyskytují makra pro zachycení TRAPů a nastavení přeposlání této události dalším zařízením nebo zpracování pomocí spuštění skriptů.

Na následujícím obrázku je vidět ukázka nastavení odchyty a přeposlání TRAPů příslušným skriptům.

---

```
#set security community and privileges
authCommunity log,execute,net public

# the generic traps
traphandle SNMPv2-MIB::coldStart /home/nba/bin/traps cold
traphandle SNMPv2-MIB::warmStart /home/nba/bin/traps warm
traphandle IF-MIB::linkDown /home/nba/bin/traps down
traphandle IF-MIB::linkUp /home/nba/bin/traps up
traphandle SNMPv2-MIB::authenticationFailure /home/nba/bin/traps auth
# this one is deprecated
traphandle .1.3.6.1.6.3.1.1.5.6 /home/nba/bin/traps egg-neighbor-loss

# enterprise specific traps
traphandle UCD-TRAP-TEST-MIB::demoTrap /home/nba/bin/traps demo-trap
#trap handling meteostation warnings and errors
traphandle METEOSTATION-MIB::meteostationTrap /home/nba/bin/mail
```

---

**Výpis 3:** Nastavení skriptů pro zpracování TRAPů různých zařízení.

### 5 Cacti

Cacti je kompletní webové řešení pro tvorbu různých grafů. Je založené na tzv. „nástroji pro round robin databázi“ (RRDTool)<sup>9</sup>. Cacti poskytuje řešení pro rychlé dotazování aktuálních dat, pokročilé grafické šablony, několik možností zpracování dat a také správu uživatelů. Snaží se uživateli usnadnit práci pomocí intuitivního grafického rozhraní a mělo by zvládnout monitorování v řádu až stovek zařízení. Toto webové rozhraní jsem si vybral, protože jsem cítil potřebu nejenom zaznamenávat meteorologické údaje, nýbrž také zpracovávat a analyzovat tyto údaje a poskytnout je široké veřejnosti. Jedním z jednodušších způsobů, jak to vyřešit, je poskytování dat online pomocí internetu. V následujících sekcích práce bude popsáno rozhraní Cacti, jeho instalace, napojení na SNMP od meteostanice a tvorba vlastního pluginu do něj.

#### 5.1 Historie Cacti

Historie se datuje k roku 2001, kdy americký student střední školy Ian Berry pracoval pro malého ISP, zatímco se učil PHP a MySQL. Jeho hlavní myšlenkou bylo usnadnit uživatelům práci s RRDtool a MRTG. V roce 2004 byla vydána první stabilnější verze s pořadovým číslem 0.8.6. S touto verzí se do projektu přidalo mnohem více vývojářů a dodnes se snaží o zrychlení systému a o zlepšení zpracování stále většího množství dat. Až v roce 2013 byla vydána verze 0.8.8b se spoustou vyřešených bugů a zabezpečovacích problémů.

#### 5.2 Vlastnosti Cacti

Celé Cacti rozhraní je napsáno v PHP a pro práci s grafy, jejich šablonami a datovými zdroji využívá MySQL databázi [14]. Instalace na unix-like systémech by neměla být problém, hlavně na operačních systémech využívajících balíčkového systému. Cacti by mělo být totiž dostupné v základních repozitářích. Jelikož Cacti data vizualizuje, musí je nejprve od někud brát. K tomu slouží dvě techniky dotazování na aktuální data. Obě dvě techniky využívají periodického dotazování na data (polling). Prvním způsobem je vyžití PHP skriptu „cmd.php“, který je starší a pomalejší, ale stačí pro monitorování několika desítek zařízení [16]. Pokud se odpověď nedostaví v čase do 300 s, což je nejmenší možná perioda dotazu, je lepší využít druhé techniky – Spine (Cactid). Spine je dotazovací nástroj napsaný v C, podporuje práci s vlákny a SNMP a proto je mnohem rychlejší.

Má meteostanice využívá 8 čidel (2x teplota) na zaznamenání meteorologických dat. Z toho je jasné, že mi stačí využívat starší „cmd.php“ techniku pro dotazování. Bohužel nejmenší normální perioda dotazu je 300 s, nicméně to se mi podařilo vyřešit nainstalováním pluginu „realtime“. Meteostanice ukládá data v intervalech 5 vteřin a přesně toto plugin umí – vizualizovat data v reálném čase a v 5 vteřinových intervalech.

---

<sup>9</sup>jedná se o oficiální název – zkratku – pro open source nástroj vydaný pod licencí GNU



Mezi další vlastnosti Cacti patří [14]:

**Grafy** Možnost přidat jakékoliv množství grafů, upravit hodnoty v grafu pomocí funkcí (CDEF), shlukování grafů do skupin. CDEF funkce jsou definovány globálně a dají se využít pro jakýkoliv graf. Podpora množství RRDTool typů - AREA, STACK, LINE ...

**Datové zdroje** Umožňuje sbírat data jak z lokálních, tak ze vzdálených úložišť přes síť. Umožňuje použít více, než jeden datový zdroj na graf, ukládání RRD souboru kdekoli na disku. Ukládání dat z různého časového rozmezí pro nestandardní grafy.

**Sběr dat** Sběr dat ze skriptů a podpora SNMP - php-snmp, ucd-snmp nebo net-snmp. PHP dotazovací nástroj umí získat data ze skriptů a SNMP a umí aktualizovat RRD soubory [15].

**Šablony** Shlukování podobných grafů, datových zdrojů a hostů pomocí šablon.

**Zobrazení grafů** Několik typů zobrazení grafů - stromová struktura s příslušnou hierarchií, seznam, „preview“.

**Správa uživatelů** Uživatelé mají vlastní přístupová práva, jména a hesla a mohou vlastnit své grafy. Uživatelé mají své nastavení vlastností grafů.

### 5.3 Instalace na Raspberry Pi

Pro správný chod všech komponent bylo potřeba na Raspberry Pi nainstalovat několik balíčků, jejichž výčet je v následujícím seznamu [13]:

- apache2 - http démon (web server) pro přístup v prohlížeči
- php5 - programovací jazyk pro Cacti, bylo nutné doinstalovat moduly php5-mysql a php5-snmp pro práci s databází a snmp
- mysql-client mysql-server - klientská a serverová strana databáze
- snmp a snmpd - klientská (příkazy a dotazování z terminálu) a serverová strana snmp
- rrdtool - balíček pro práci s RRD soubory

Po nainstalování a nastavení příslušných balíčků bylo potřeba ještě vytvořit databázi a přístupová práva pro uživatele „cacti“. Poté jsem stáhl zdrojové soubory a nakopíroval je na příslušné místo na disku (/var/html/www) [13]. Také bylo potřeba pomocí webového Cacti průvodce naplnit tabulky základními údaji o počítači a nastavit práva pro přístup přes http.

### 5.4 Vytváření pluginu

Projekt Cacti má poměrně rozsáhlou komunitu vývojářů a nadšenců, kteří se snaží projekt neustále vylepšovat a zdokonalovat. Pro takové lidi byla vytvořena možnost vytvoření vlastního pluginu, který by byl volně dostupný ke stažení a open source. Jelikož jsem nechtěl mít v Cacti pouze zobrazené jednotlivé grafy meteorologických veličin, ale také něco navíc, vytvoření vlastního meteo pluginu pro mě byla jasná volba. Plugin má totiž tu výhodu, že je možné si udělat vlastní webovou stránku či více stránek přesně na míru s pomocí Cacti dokumentace.

Všechny zdrojové soubory od Cacti se vyskytují v adresáři `/var/html/www`. Složka pluginu musí být obsažena v adresáři `/var/html/www/plugins/PLUGINNAME/`, kde `PLUGINNAME` je jméno pluginu, v mém případě `meteostation`. V tomto adresáři musí být také přítomen „`setup.php`“ soubor, podle kterého Cacti pozná, že se jedná o plugin. V tomto souboru se musí vyskytovat pouze funkce, žádný samostatný spustitelný kód. V následujícím výčtu se nacházejí základní funkce, které musí být obsaženy v souboru „`setup.php`“. Funkce nepřijímají žádný argument, pouze provádí potřebné operace a vracejí výsledné hodnoty.

- `plugin_meteostation_install` - první funkce, která se volá při instalaci pluginu
- `plugin_meteostation_uninstal` - volá se při odinstalaci pluginu
- `plugin_meteostation_version` - poskytuje dodatečné informace o pluginu - verze, autor, email, domovská adresa
- `plugin_meteostation_check_config` - kontroluje, zda je plugin po instalaci připraven k použití, vrací `TRUE` po úspěšné instalaci a nastavení

Další důležitou funkcí je `api_plugin_register_hook`, která obsahuje 4 argumenty:

- `PLUGINNAME` - jméno pluginu
- `HOOKNAME` - název prvku/události, na kterou se má funkce připojit, Cacti obsahuje velké množství těchto událostí
- `CALLBACKFUNCTION` - funkce, která se má zavolat po spuštění události
- `FILENAME` - název souboru, ve kterém se vyskytuje `CALLBACKFUNCTION`

Pro omezení přístupu k pluginu se dá využít funkce `api_plugin_register_realm`, ve které mají argumenty následující význam:

- `PLUGINNAME` - jméno pluginu
- `FILENAMETORESTRICT` - jméno souboru/souborů, kterým je přiřazeno omezení
- `DISPLAYTEXT` - text, který bude zobrazen ve formuláři pro oprávnění přístupu
- čtvrtým argumentem je hodnota 1, pokud má být automaticky přiděleno oprávnění uživateli „`admin`“, jinak 0

### 5.4.1 Implementace pluginu

U implementace pluginu jsem vycházel z již hotového příkladu, pluginu se jménem „pl-gexample“. Ten bylo potřeba nejprve nainstalovat a přepsat veškeré proměnné obsahující název pluginu na „meteostation“. Poté jsem potřeboval vytvořit stránku, na které bych mohl prezentovat data s meteostanic. Využil jsem toho, že Cacti podporuje napojení na libovolný počet zařízení. Proto jsem na stránce nechal nejprve zobrazit formulář pro výběr zařízení. Po výběru zařízení se uživateli zobrazí několik tabulek s informacemi týkajících se zvoleného zařízení a případně připojené meteostanice. Meteostanic může být na Cacti připojen větší počet na různých zařízeních, a proto je důležité vybrat zařízení, na kterém je nějaká meteostanice připojena.

**5.4.1.1 Webová stránka** Po kliknutí na odkaz pluginu v horní liště záložek Cacti lze na webové stránce spatřit pouze výběrový formulář. Po výběru se na stránce objeví několik tabulek. V první tabulce se vyskytují informace ohledně zvoleného zařízení, jak je vidět v tabulce 5.1.

Název sloupce	Popis
description	popis zařízení (název)
hostname	IP adresa nebo hostname zařízení
snmp_version	verze SNMP protokolu
snmp_port	port SNMP protokolu
availability	dostupnost v procentech
meteostation_enabled	je připojena a běžící meteostanice?
physical quantities	seznam měřených meteorologických veličin

**Tabulka 5.1:** Tabulka znázorňující informace o zvoleném zařízení.

V podobném stylu se na stránce objeví také druhá tabulka, ve které se nachází aktuální hodnoty naměřené na meteostanici. Spolu s názvem veličiny a hodnotou lze vidět také její jednotku, aby nedocházelo k nedorozuměním mezi uživateli připojujícími se ze vzdálených krajů. Ve třetí tabulce jsou vidět všechny grafy, které jsou k daným meteorologickým veličinám přiřazeny a uloženy v Cacti. V názvu těchto grafů se musí vyskytovat název zařízení, na kterém meteostanice běží, aby se mohly tyto grafy svázat se stejným zařízením. Příkladem takového názvu zařízení a grafu je např. zařízení „Raspberry Pi-meteostation“ a graf „|Raspberry Pi-meteostation| - windDirection“. V poslední tabulce se vyskytuje pouze popis vysvětlující hodnoty ve sloupcích v tabulce první.

**5.4.1.2 Menu** Jako další problém k řešení byla tvorba a registrace záložky „meteostation“ v horní liště Cacti. Naštěstí stačilo ikonu pouze upravit v grafickém editoru a záložku následně registrovat pomocí funkce `api_plugin_register_hook`. Tím bylo dosaženo potřebného odkazu v hlavním menu Cacti na meteostation plugin.

## 5 CACTI

### 5.4.2 Instalace pluginu

Instalace jakéhokoliv pluginu do Cacti je jednoduchá. Stačí stáhnout příslušný plugin, rozbalit ho a nakopírovat do složky „plugins“. Následně je ještě zapotřebí kliknout na instalaci pluginu ve webovém rozhraní a povolit ho. Tím je instalace dokončena.

### 5.4.3 Nastavení sítě

Meteostanice je připojená na domácí router. Z tohoto důvodu bylo potřeba udělat několik úprav:

1. nastavit http port na jiný než 80
2. přesměrovat porty na vnějším routeru
3. přesměrovat porty na domácím (vnitřním) portu

Pomocí předchozího postupu se mi podařilo nastavit webové rozhraní Cacti tak, aby bylo dostupné pro jakékoliv zařízení připojené k internetu. Konkrétní adresa Cacti je 109.81.189.30:222/cacti. Port 222 byl zvolen, jelikož port 80 se používá jako webové rozhraní vnějšího routeru. Na Raspberry Pi bylo nutné dodatečně povolit přístup na http i z internetu, jelikož implicitně je tato volba zakázána.

The screenshot shows the Cacti web interface for the 'Meteostation' plugin. The page title is 'Meteostation: Ostrava, Zabreh, Jiskricek 2'. The interface includes a navigation bar with 'console', 'graphs', 'Meteostation', and 'settings' tabs. The main content area is divided into several sections:

- Select Host (Cacti Hosts):** A dropdown menu showing 'Localhost' and 'RPI-meteostation'.
- Information about selected host:** A table with columns: description, hostname, snmp\_version, snmp\_port, availability, meteostation enabled, and physical quantities.
- Actual meteodata:** A table with columns: meteorologic value, value, and unit.
- Meteo graphs in Cacti for this host:** A table with columns: id, local\_graph\_id, title, and export.

description	hostname	snmp_version	snmp_port	availability	meteostation enabled	physical quantities
RPI-meteostation	localhost	2	161	99.96287	YES	windDirection, windSpeed, rain, sunshine, temperature, humidity, pressure

meteorologic value	value	unit
windDirection	90	°
windSpeed	0	m/s
rain	0	mm/h
sunshine	22	V
temperature	28	°C
humidity	30	%
pressure	1018	hPa

id	local_graph_id	title	export
52	9	RPI-meteostation  - windDirection	on
49	6	RPI-meteostation  - windSpeed	on
50	7	RPI-meteostation  - rain	on
51	8	RPI-meteostation  - sunshine	on
53	10	RPI-meteostation  - temperature	on
54	11	RPI-meteostation  - humidity	on
55	12	RPI-meteostation  - pressure	on

Obrázek 5.1: Ukázka stránky zobrazující údaje o meteostanici (plugin).

### 6 Konstrukce meteostanice

U konstrukce meteostanice jsem vycházel z nutnosti ukotvení analogových čidel k zemi. Analogová čidla jsou totiž mechanickými díly poloprofesionální meteostanice WH1080 a při jejich koupi jsem obdržel také několik upevňovacích trubek. Nicméně nebylo toho mnoho a potřeboval jsem najít lepší řešení např. pro postavení meteostanice na zem. Dalším problémem bylo vytvoření smysluplné krabičky pro Arduino a Raspberry Pi, která by fungovala jednak jako ochrana proti vnějším vlivům a také jako obal pro lepší manipulování s počítači.

#### 6.1 Upevnění mechanických dílů

Doporučení pro sestavení těchto dílů a jejich plnohodnotné měření je takové, že by se alespoň anemometr a směrová růžice měly vyskytovat na nejvyšším bodě ve výšce kolem 10 m. Meteostanice by měla být postavena v otevřeném prostoru, nejlépe co nejdále budov. Mechanické díly se dají připevnit dvěma způsoby:

- připevnění pouze pomocí tyčí na okraj budovy či jiné konstrukce, která splňuje doporučenou výšku a vzdálenost od ostatních objektů
- upevnění na vlastní podstavec, který by bylo možné položit kamkoliv do terénu

Chtěl jsem vytvořit nějaký podstavec, aby bylo možné meteostanici přenášet. Využil jsem volného květináče, který jsem vyplnil sádrou, aby byl podstavec stabilní a nepřevrátil se např. působením větru. Doprostřed květináče jsem umístil hlavní tyč držící mechanické díly. Vše jsem vyrovnával vodováhou, aby čidla byla v rovnováze a neudávala nepřesné údaje. Výsledek tohoto snažení lze vidět v příloze ve formě několika fotografií.

#### 6.2 Krabičky pro Arduino a Raspberry Pi

Potřeboval jsem vytvořit krabičky pro oba počítače, protože k oběma vede několik kabelů (ethernet, usb). Jelikož je řešení meteostanice drátové, nejvhodněji se mi jevílo řešení obě krabičky mít co nejbližší u sebe spolu s mechanickými díly. Raspberry Pi by mělo být venku a blízko mechanických dílů i z důvodu napojení na digitální čidla, tzn. že digitální čidla jsou v krabičce spolu s Raspberry Pi. Takhle je možné mít všechny 3 hlavní díly pohromadě (Arduino, Raspberry Pi, mechanické díly) a vést pouze 2 hlavní kabely k meteostanici - napájecí kabel USB a síťový ethernetový kabel.

Mechanické díly byly po koupi opatřeny vývody zakončenými malým ethernetovým konektorem (RJ11). Bohužel jediný ethernet, který byl na malých počítačích dostupný, byl na Raspberry Pi a ten je využíván na zapojení do LAN. Proto jsem musel mechanické díly připojit jinak. Udělal jsem to tak, že jsem je připojil na nové konektory. Tyto konektory mají vždy 2 díly – vidlici a kus do panelu (kus a protikus). Vidlice jsem napájel na dráty vedoucí signály od analogových čidel (včetně fotodiody). Protikusy patřící do panelu jsem převrtal do předem připravených krabiček a díry vyhladil šmirglem. V krabičce od Arduina jsem připevnil nepájivé pole, na které jsou napájeny z jedné strany drátky

## 6 KONSTRUKCE METEOSTANICE

z konektorů a z druhé strany jsou připojeny vstupní piny Arduina. Tím bylo dosaženo vytvoření základny pro oba počítače. Výsledný design krabiček a napájených konektorů lze zhlédnout v příloze v sekci B.

### 6.2.1 Výběr konektorů

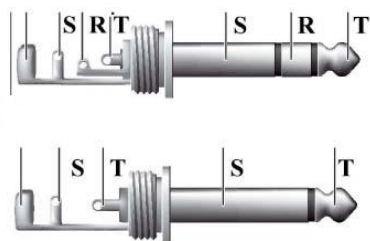
Pro přenos signálu z analogových čidel jsem potřeboval napojení do krabičky s Arduinem pomocí nějakých konektorů. Konektorů je na trhu spousta, proto jsem se snažil vybrat nějaké hojně se vyskytující a využívané. Pro přenos signálu z anemometru a směrové růžice jsem se rozhodl pro konektor DIN, jelikož jsem potřeboval připojit 4 drátky. Srážkoměru stačily pouze 2, a proto jsem využil stereo jacku, který se hojně využívá u audio zařízení. Stačil by i mono jack, nicméně jsem nechtěl nechat nic na náhodě a v případě např. špatného napájení mít jeden pin jako rezervu. Ukázka konektorů je k vidění na obrázcích 6.1–6.4.



Obrázek 6.1: DIN – vidlice [25]



Obrázek 6.2: DIN – do panelu [25]



Obrázek 6.3: Jack – vidlice – stereo/mono [29]



Obrázek 6.4: Jack – do panelu [25]

Při pájení jsem narazil na jeden závažný problém. Po koupi nejlevnějšího konektoru DIN jsem očekával jeho alespoň několikaměsíční životnost, tato teorie se však ukázala jako velmi nepřesná. Při pájení a mírném zahřátí konektoru se začal plastový kryt rozpadat a byl jsem nucen tento konektor vyměnit za nový. Tato situace se opakovala i u jednoho konektoru typu „jack“, kdy jsem byl nucen koupit kvalitnější a dražší díly. S těmito díly jsem však neměl problém a pájení všech částí se nakonec povedlo.

### Závěr

Cílem diplomové práce bylo navrhnout a následně sestavit meteostanici s výstupem dat na webové rozhraní a síťový protokol SNMP. Meteostanice je logicky úzce spjata s vědou, která se nazývá meteorologie. Začátek práce je věnovaný historii meteorologie, rychlému přehledu meteostanic dostupných v obchodech a teoretickému řešení výroby meteostanice. V první kapitole je popsán také význam meteorologie a jejich základních veličin jako jsou: teplota, vlhkost, tlak, směr a rychlost větru a sluneční svit je. Spolu s popisem jsou vylíčeny také metody, jak a čím se dají jednotlivé veličiny měřit.

Při sestavování vlastní meteostanice je potřeba pořídit si vhodné součástky. Na měření meteorologických veličin se používají analogová nebo digitální čidla. Jelikož měla být meteostanice propojená se softwarem, bylo nutné obstarat zařízení, které je možno programovat. Ideálním řešením se jeví dva minipočítače – Raspberry Pi a Arduino. Na prvním z počítačů běží operační systém na bázi Linuxu a proto je vhodný pro zapojení do sítě. Je také připojen na webové rozhraní Cacti a síťový protokol SNMP. Arduino naproti tomu obsahuje programovatelný mikrokontrolér, který se hodil na periodické zpracování dat z analogových čidel. Pro měření teploty a vlhkosti jsem vybral digitální čidlo SHT15, pro teplotu a tlak pak BMP085 breakout. Srážky, směr a rychlost větru se měří pomocí analogových čidel, konkrétně jsou to mechanické díly meteostanice WH1080. Důvod výběru čidel a počítačů, jejich specifikace a popis je uveden ve druhé části práce.

Po nákupu čidel a počítačů bylo potřeba zprovoznit komunikaci mezi všemi prvky meteostanice. Obě digitální čidla (SHT15, BMP085) jsou připojena na vstup-výstupní piny Raspberry Pi a jsou k nim napsány programy v jazyku C. Analogová čidla jsou připojena na Arduino. Mikrokontrolér Arduina je naprogramován tak, aby periodicky četl data z čidel a posílal je pomocí sériového protokolu (výstup v podobě USB) na Raspberry Pi. Takto je řešeno posílání dat na straně Arduina. Na Raspberry Pi běží skript, který čeká na sériovém rozhraní a data po obdržení uloží do databáze. O posílání aktuálních dat do SNMP se stará další Python skript, který využívá rozšiřujícího protokolu AgentX. Webové rozhraní (Cacti) přijímá data jednoduše tak, že se dotazuje na SNMP. Detailnější popis komunikace a stručný výpis programů je k vidění v sekci 3.

Další část se věnuje rozebráním síťového protokolu SNMP. Zakomponoval jsem ho do práce z toho důvodu, protože většina komerčních meteostanic, dostupných na trhu, má zabudované právě napojení na tento protokol. Této problematice jsem se věnoval hlavně z hlediska architektury protokolu a napojení na jeho rozšiřující části (AgentX). Protokol AgentX jsem využil u posílání dat z databáze do SNMP, poněvadž mi to umožnilo poskytovat data do vlastního MIB souboru.

Předposlední kapitolou je webové rozhraní Cacti a tvorba pluginu do něj. Tam je také popsána krátká historie a struktura projektu. Cacti jsem vybral, protože poskytuje kvalitní řešení vizualizace dat měnících se v čase. Základním prvkem v Cacti je nástroj RRDTool, pomocí kterého lze data vizualizovat v různém časovém rozmezí a po určitém datu rotovat databázi. To se hodí např. u dlouhodobého měření počasí. Cacti poskytuje

možnost implementace vlastního pluginu, čehož jsem využil. Implementace a vzhled pluginu je popsán také v této části.

Poslední sekcí je konstrukce meteostanice. Ta je rozebrána pouze na dvě stránky, nicméně je důležitou součástí celého projektu. Meteostanice by měla být umístěna ve volném prostoru a proto bylo nutné, aby odolala alespoň silnějšímu větru a srážkám. Proto jsem se rozhodl zabudovat Raspberry Pi a Arduino do malých plastových krabiček, ke kterým jsem do stěn připevnil konektory. Takto jsou oba počítače chráněny proti vnějším vlivům a mohou být zároveň připojeny na anemometr, směrovou růžici a srážkoměr. Pro přenášení meteostanice jsem vyrobil stabilizační podstavec.

Když jsem meteostanici vyráběl, napadlo mě hned několik myšlenek, jak tuto práci rozšířit. Jelikož může být meteostanice někde venku, daleko od elektrického vedení a připojení k síti, mohla by být meteostanice napájena samostatně např. pomocí fotovoltaických článků. Taková zařízení se občas nazývají „ostrovní systémy“. Připojení k síti by se dalo řešit např. pomocí napojení na GSM modul jednoho z místních mobilních operátorů. Trochu jiná varianta rozšíření by mohla počítat s tím, že pouze Arduino bude venku a na něj budou napojená všechna čidla (pokud bude dostatek pinů). Raspberry Pi by bylo v budově připojené k LAN a běžel by na něm SNMP a Cacti. U této varianty by bylo možné využít bezdrátového přenosu dat např. pomocí modulů RFM01 a RFM02. Dalšími možnostmi by bylo třeba zdokonalení designu krabiček, aby se počítače nepoškodily v extrémnějších podmínkách (zvýšení stupně krytí IP), nebo vylepšení Cacti pluginu.

Meteostanice byla navržena a realizována. Čidla byla zkalibrována. Napojení na SNMP a webové rozhraní bylo implementováno. Meteostanice vyrobená v této práci se může přesností a funkčností rovnat stanicím poloprofesionálním jako jsou stanice WH1080 nebo TFA 35.1075 NEXUS. Nespornou výhodou však je dostupnost jejího volně dostupného softwaru, rozšiřitelnost a hlavně nízká cena. Všechny součástky použité v práci nepřesáhly cenu 3 000 Kč, přičemž poloprofesionální stanice se dají pořídit i několikanásobně draž.

Jan Šimeček



### Literatura

- [1] Podborský, Vladimír, *Dějiny pravěku a rané doby dějinné* Brno: Masarykova Univerzita, 2006 [cit. 2014-03-21].
- [2] BOSCH, *Datasheet BMP085 [online]*. 2008. [cit. 2014-03-22]. <<http://dlnmh9ip6v2uc.cloudfront.net/datasheets/Sensors/Pressure/BMP180.pdf>>
- [3] Argent Data Systems, *Weather Sensor Assembly p/n 80422 [online]*. <<http://ebookbrowse.net/weather-sensor-assembly-pdf-d23118026>>, 2014. [cit. 2014-03-27].
- [4] Douglas Mauro, Kevin Schmidt, *Essential SNMP O'REILLY* <<http://oreilly.com/catalog/esnmp/chapter/ch02.html>>, 2001 [cit. 2014-04-09].
- [5] Krška Karel, Šamaj Martin *Dějiny meteorologie v českých zemích a na Slovensku* Karolinum, 2001. 565 s. ISBN: 8071849510
- [6] F.W. Henkel *Weather Science: An Elementary Introduction to Meteorology [online]*. T. F. Unwin, 1911. 368 s. ISBN: B002WUTT4 <<http://www.e-booksdirectory.com/details.php?ebook=8899>>
- [7] Dr. Michael Pidwirny, Scott Jones *FUNDAMENTALS OF PHYSICAL GEOGRAPHY [online]*. University of British Columbia Okanagan, 2010. [cit. 2014-04-17] <<http://www.physicalgeography.net/fundamentals/8c.html>>
- [8] SENSIRION *Datasheet SHT15 [online]*, 2011. [cit. 2014-04-18]. <[http://www.sensirion.com/fileadmin/user\\_upload/customers/sensirion/Dokumente/Humidity/Sensirion`Humidity`SHT1x`Datasheet`V5.pdf](http://www.sensirion.com/fileadmin/user_upload/customers/sensirion/Dokumente/Humidity/Sensirion`Humidity`SHT1x`Datasheet`V5.pdf)>
- [9] Pipošiar Jiří *RMI rozhraní pro monitorování počítačové sítě s využitím protokolu SNMP* Plzeň. 2007. 37 s. Bakalářská práce na fakultě Aplikovaných věd Západočeské univerzity v Plzni na katedře informatiky a výpočetní techniky.
- [10] *Secure Internet Management and SNMP* SNMP Research International, Inc. <<http://www.snmp.com/>> [cit. 2014-04-19]
- [11] Štěpán Berka *Elektrotechnická schémata a zapojení 2 BEN* - technická literatura, Praha 2010. ISBN 978-80-7300-254-1
- [12] David T. Perkins *Understanding SNMP MIBs [online]*. c1993 [cit.2014-04-21]. <<http://www.dsc.ufcg.edu.br/~jacques/cursos/gr/recursos/outros/perkins.pdf>>
- [13] Ian Berry aj. *The Cacti Manual [online]*. The Cacti Group, 2012. [cit.2014-04-21]. <<http://www.cacti.net/downloads/docs/pdf/manual.pdf>>
- [14] The Cacti Group *Cacti – The complete rrdtool-based graphing solution [online]*. c2012. Ver. 0.8.8.b [cit.2014-04-10]. <<http://cacti.net/>>

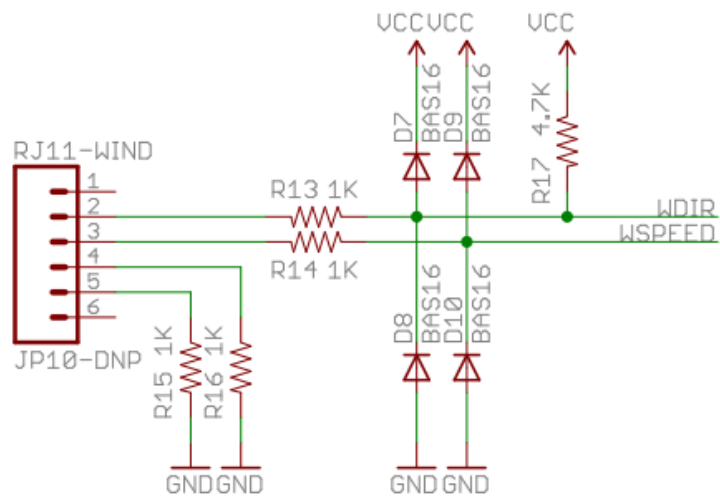
## LITERATURA

---

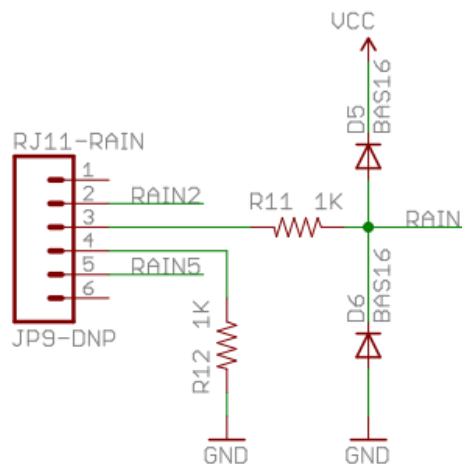
- [15] Dinangkur Kundu, S. M. Ibrahim Lavlu *Cacti 0.8 Network Monitoring [online]*. c2009. [cit.2014-04-09]. <<http://www.academia.edu/4018280/Cacti'Network' Monitoring?login=&email'was'taken=true>>
- [16] Thomas Urban *Cacti 0.8 Beginner's Guide* Packt Publishing, 2011. 348 s. ISBN: 1849513929
- [17] Tobias Oetiker *OETIKER+PARTNER [online]*. poslední úpravy 2013-06-26 [cit.2014-04-12]. <<http://oss.oetiker.ch/rrdtool/>>
- [18] ČHMÚ *HISTORIE METEOROLOGIE A KLIMATOLOGIE V ČESKU [online]*. [cit.2014-03-20]. <<http://portal.chmi.cz/files/portal/docs/ruzne/vystava/METEO/1.pdf>>
- [19] Brendan Holan *Practical Raspberry Pi* apress, 2013. 272 s. ISBN: 1430249714
- [20] Matt Richardson, Shawn Wallace *Getting Started with Raspberry Pi* O'Reilly Media, 2012. 180 s. ISBN: 978-1-4493-4421-4
- [21] Maik Schmidt *Raspberry Pi A Quick-Start Guide* The Pragmatic Programmers, c2012. 112 s.
- [22] Andrew K. Dennis *Raspberry Pi Home Automation with Arduino* Packt Publishing, 2013. 176 s. ISBN: 978-1-84969-586-2
- [23] Arduino *Arduino Leonardo [online]*. 2014. [cit.2014-03-20]. <<http://arduino.cc/en/Main/arduinoBoardLeonardo>>
- [24] Jan Šimeček *Meteostanice s připojením na PC* Brno. 2012. 42 s. Bakalářská práce na Fakultě informačních technologií Vysokého učení technického v Brně.
- [25] GM electronic *GM electronic [online]*. 2014. [cit.2014-03-19]. <[www.gme.cz](http://www.gme.cz)>
- [26] Argent Data Systems *Argent Data Systems [online]*. 2014. [cit.2014-03-19]. <[www.argentdata.com](http://www.argentdata.com)>
- [27] *Neuroscience and Robotics Lab [online]*. Northwestern University. [cit. 2014-03-24]. <<http://nrx.northwestern.edu/>>
- [28] *Net-SNMP [online]*. poslední revize 26.3.2013 [cit. 2014-03-24]. <<http://www.net-snmp.org/>>
- [29] *Hardware.fr [online]*. 2014. [cit. 2014-04-37]. <<http://www.hardware.fr/>>
- [30] Kassen Lloyd, Jordan VanOsdol, Dr. Donald Trisel *Digital Honey Bee experiment [online]*. Fairmont State University, poslední úpravy 11.8.2013 [cit. 2014-04-37]. <<http://www.digitalbees.info/>>

## A SCHÉMATA PŘIPOJENÍ ANALOGOVÝCH ČIDEL

### A Schémata připojení analogových čidel

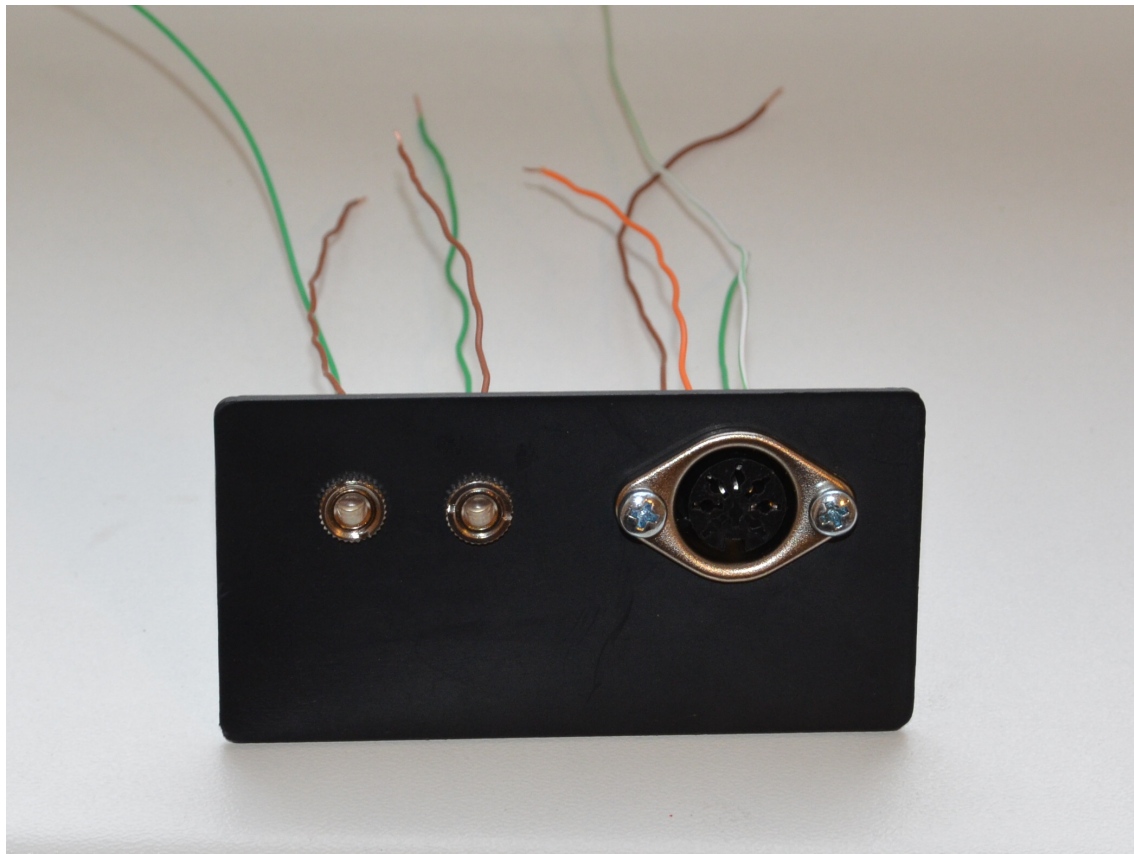


Obrázek A.1: Schéma zapojení anemometru a směrové růžice [30]



Obrázek A.2: Schéma zapojení srážkoměru [30]

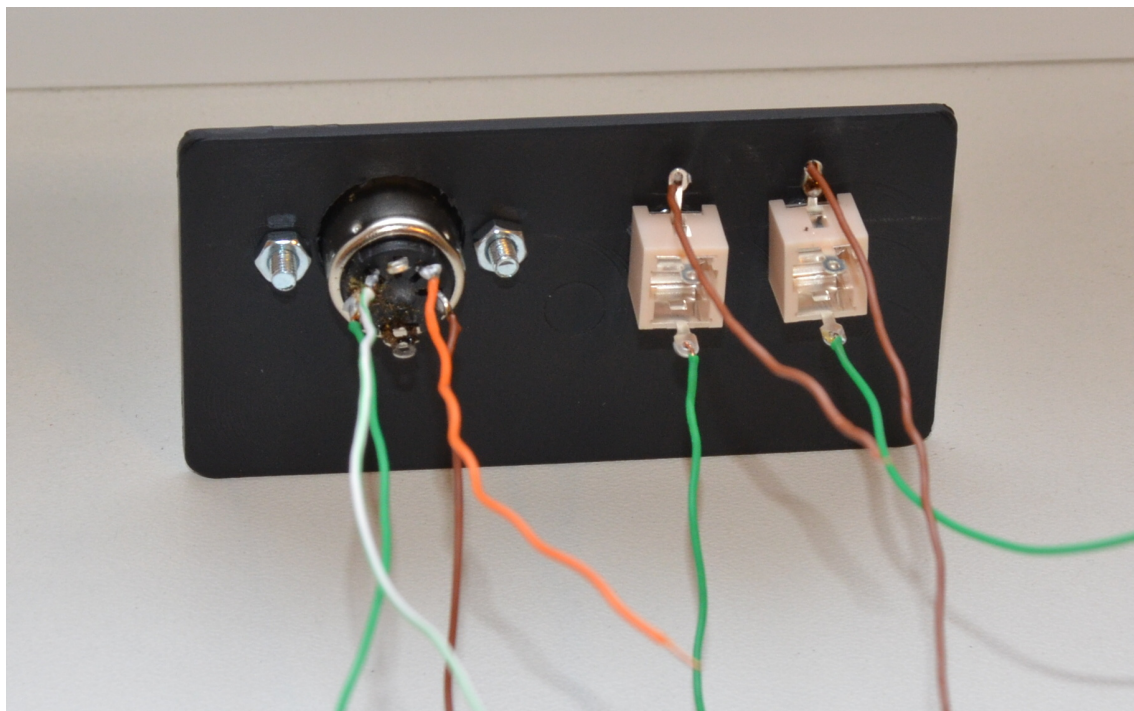
**B Konstrukce meteostanice – fotografie**



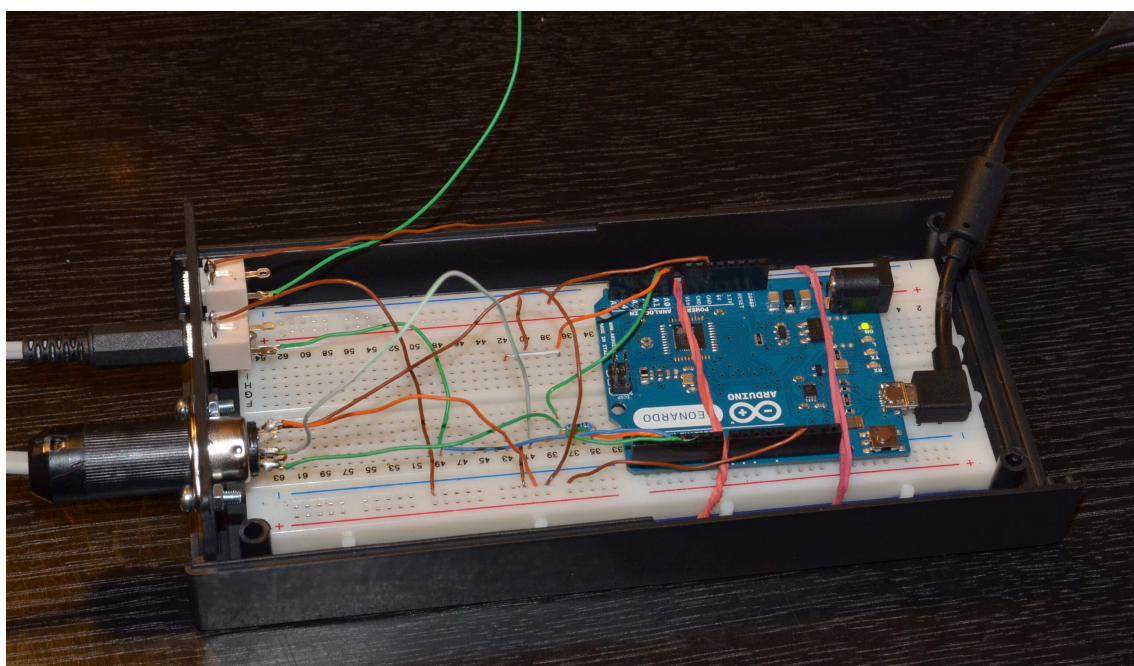
**Obrázek B.1:** Přední stěna krabičky pro Arduino. Pohled zvenku.

## B KONSTRUKCE METEOSTANICE – FOTOGRAFIE

---



Obrázek B.2: Přední stěna krabičky pro Arduino. Pohled zevnitř.



Obrázek B.3: Připojení analogových čidel přes konektory do nepájivého pole a Arduino.



## B KONSTRUKCE METEOSTANICE – FOTOGRAFIE

---



Obrázek B.4: Přední stěna krabičky Raspberry Pi. Pohled zvenku.



Obrázek B.5: Raspberry Pi upevněné v krabičce.

## B KONSTRUKCE METEOSTANICE – FOTOGRAFIE

---



Obrázek B.6: Celá meteostanice.

## C Příloha na CD

**meteostanice-dp-SIM526.pdf** Text diplomové práce – tento dokument.

**Zadani1.jpg** Naskenované okolkované zadání diplomové práce v elektronické formě.

**source** Složka obsahující všechny zdrojové kódy (skripty a programy) potřebné ke spuštění systému.

**manual.pdf** Manuál pro práci s meteostanicí na zařízeních Raspberry Pi a Arduino. Obsahuje také popis instalace nutného software.