

**ZÁPADOČESKÁ UNIVERZITA V PLZNI**

**FAKULTA ELEKTROTECHNICKÁ**

**KATEDRA ELEKTROMECHANIKY A VÝKONOVÉ ELEKTRONIKY**

**BAKALÁŘSKÁ PRÁCE**

**Elektronická prezentace experimentu**

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Oldřich HOLÝ**  
Osobní číslo: **E12B0026K**  
Studijní program: **B2644 Aplikovaná elektrotechnika**  
Studijní obor: **Aplikovaná elektrotechnika**  
Název tématu: **Elektronická prezentace experimentu**  
Zadávající katedra: **Katedra elektromechaniky a výkonové elektroniky**

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

Cílem je vytvoření části informačního systému, přebírajícího aktuální data z reálného technického experimentu.

1. Prostudujte rozhraní pro komunikaci s přístroji.
2. Vytvořte jeho programovou obsluhu s ukládáním dat do databáze.
3. Prostudujte možnosti zobrazení dat na webové stránce, výstupy tabulkové s filtrací, grafické, download dat, atd.
4. Vytvořte program pro efektivní prezentaci dat z experimentu pro webový server.
5. Popište vlastnosti vytvořeného systému, prostudujte a sepište jeho slabá místa a možné důvody poruchy.

Rozsah grafických prací: **podle doporučení vedoucího**

Rozsah pracovní zprávy: **20 - 30 stran**

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

Seznam odborné literatury:

1. Dostupné manuály k programům, jazykům a systémům BASH, Python, MySQL, PHP a HTML.
2. Dokumentace k referenční aplikaci informačního systému.
3. Další vhodnou literaturu si student vyhledá v dostupných pramenech podle doporučení vedoucího práce.

Vedoucí bakalářské práce:

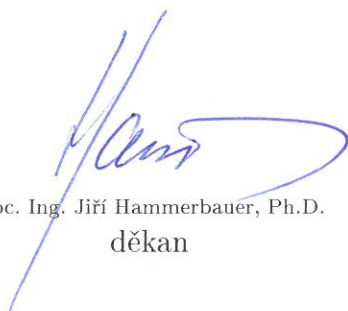
**Ing. Richard Linhart, Ph.D.**

Katedra aplikované elektroniky a telekomunikací

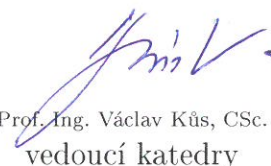
Datum zadání bakalářské práce: **15. října 2014**

Termín odevzdání bakalářské práce: **8. června 2015**

Doc. Ing. Jiří Hammerbauer, Ph.D.  
děkan



Prof. Ing. Václav Kůs, CSc.  
vedoucí katedry



V Plzni dne 15. října 2014

## **Abstrakt**

Práce se zabývá vývojem aplikace pro měření, zpracování a elektronickou prezentaci dat magnetometru. Téma vzniklo za účelem prohloubení znalostí se senzory pro měření magnetického pole Země typu AMR. Tyto senzory složené z tenkých odporových pásků slitiny niklu a železa (Permalloy) fungují na principu změny odporu vlivem vnějšího magnetického pole. Použitým senzorem v přístroji je Honeywell HMC5843. Měření je definováno sedmi parametry magnetického pole Země a teplotou. Pro komunikaci a práci se zařízením připojeným na sériový port je vytvořena sada skriptů v jazyce Python. Základem je modul pro nastavení sériové komunikace, dále skripty pro inicializaci a kalibraci zařízení vycházející ze specifikací výrobce. Měřicí algoritmus zpracovává naměřená data a ukládá je do databáze na lokálním úložišti a do databáze platformy ThingSpeak, určené k vizualizaci výsledků měření. Prezentace dat je rozčleněna na tři části – zobrazení aktuálních dat, souboru hodnot v průběhu dne a aktuálního měsíce, využívající přehledových tabulek dat a grafů vytvořených pomocí uvedené platformy. Starší data jsou archivována pro další využití.

## **Klíčová slova**

AMR senzor, měření magnetického pole, model magnetického pole Země, anizotropická magnetorezistivita, měřicí aplikace, elektronická prezentace, ThingSpeak.

## **Abstract**

This thesis deals with the development of an application for measuring, data processing and electronic presentation for use with the magnetometer. The theme was established for gaining deeper knowledge in Earth's magnetic field measuring by sensors type AMR. These sensors composed of thin strips made of nickel and iron alloy also called Permalloy works on the principle of change in resistance due to the external magnetic field. Sensor used in the device is Honeywell HMC5843. The measuring is defined by seven parameters of Earth's magnetic field and temperature. The set of scripts written in Python is used for communication and work with the device connected to the serial port. The base is a module for serial communication settings, scripts for device initialization and calibration, based on manufacturer specifications. The measuring algorithm processes the data and stores them into the local and ThingSpeak platform database, which is intended for measuring results visualization. Data presentation is divided into three parts – displaying the actual value, the data sets of values during the day and during the current month, using surveillance data tables and graphs created with the given platform. Older data are archived for further use.

## **Keywords**

AMR sensor, magnetic field measuring, The World Magnetic Model, anisotropic magnetoresistivity, measuring application, electronic presentation, ThingSpeak

## **Prohlášení**

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně, s použitím odborné literatury a pramenů uvedených v seznamu, který je součástí této bakalářské práce.

Dále prohlašuji, že veškerý software, použitý při řešení této bakalářské práce, je legální.

.....

podpis

V Plzni dne 4.6.2015

Oldřich Holý

## **Poděkování**

Rád bych poděkoval svému vedoucímu práce Ing. Richardu Linhartovi, Ph.D., za motivaci k dobrým výsledkům práce, odborné konzultace a hodnocení.

Dále děkuji Bc. Janu Širokému DiS. za podíl na pokračování práce, vývojem přístroje s novějším typem AMR senzoru Honeywell HMC5883L s pracovním názvem EMIL – Earth's Magnetic field Investigator (Low-power).

Závěrem děkuji Bc. Janě Široké za projevenou podporu, péči a trpělivost.

# Obsah

<b>OBSAH .....</b>	<b>7</b>
<b>ÚVOD.....</b>	<b>8</b>
<b>SEZNAM SYMBOLŮ A ZKRATEK.....</b>	<b>9</b>
<b>1 MĚŘENÍ MAGNETICKÉHO POLE.....</b>	<b>10</b>
1.1 PRINCIP MAGNETOREZISTIVNÍCH SENZORŮ AMR .....	11
1.2 POUŽITÝ ČIP MAGNETOMETRU HONEYWELL AMR HMC5843 .....	11
1.3 BLOKOVÉ SCHÉMA MAGNETOMETRU.....	14
1.4 MĚŘENÉ VELIČINY .....	15
<b>2 METODY VIZUALIZACE NAMĚŘENÝCH DAT .....</b>	<b>16</b>
2.1 THINGSPEAK - PORTÁL PRO ZPRACOVÁNÍ A VIZUALIZACI MĚŘENÝCH DAT .....	16
2.2 GD - GRAFICKÁ KNIHOVNA PRO PHP .....	17
2.3 D3.JS – GRAFICKÁ KNIHOVNA PRO JAVASCRIPT .....	17
<b>3 KOMUNIKACE S PŘÍSTROJEM A FUNKČNÍ SKRIPTY.....</b>	<b>18</b>
3.1 MODUL PRO NASTAVENÍ SÉRIOVÉ KOMUNIKACE PORTCFG.PY .....	19
3.2 INICIALIZAČNÍ SKRIPT MGME-INIT.PY .....	20
3.3 KALIBRAČNÍ SKRIPT MGME-CALIB.PY.....	21
3.4 TESTOVACÍ SKRIPT MGME-TEST.PY .....	21
3.5 MODUL PRO PŘÍSTUP NA SERVER THINGSPEAK.COM .....	21
3.6 STRUKTURA DATABÁZE MYSQL PRO UKLÁDÁNÍ NAMĚŘENÝCH HODNOT .....	22
3.7 MĚŘÍCÍ SKRIPT MGME-MEAS.PY .....	23
3.8 ARCHIVACE A ZPRACOVÁNÍ DENNÍCH A MĚSÍČNÍCH SOUBORŮ DAT .....	25
<b>4 VIZUALIZACE DAT NA WEBU .....</b>	<b>26</b>
4.1 VIZUALIZACE DENNÍHO SOUBORU HODNOT.....	27
4.2 VIZUALIZACE MĚSÍČNÍHO SOUBORU HODNOT .....	29
4.3 VIZUALIZACE AKTUÁLNÍ HODNOTY MĚŘENÍ .....	30
<b>HODNOCENÍ A ZÁVĚR .....</b>	<b>32</b>
<b>SEZNAM LITERATURY A DALŠÍCH INFORMAČNÍCH ZDROJŮ .....</b>	<b>33</b>
<b>PŘÍLOHY.....</b>	<b>35</b>



## Úvod

Předkládaná práce se zabývá vývojem aplikace pro zpracování měřených dat a jejich elektronické prezentace. Téma práce vzniklo zejména za účelem testování magnetorezistivních senzorů AMR, použitých pro měření magnetického pole Země. Očekávaným výstupem experimentu je získat hlubší zkušenosti s těmito senzory testováním jejich parametrů, sběrem a vyhodnocováním dat z měření a porovnáváním jejich výsledků.

Vytvořená aplikace musí mít takové vlastnosti, aby jejím prostřednictvím bylo možné s přístrojem snadno komunikovat, získávat soubory dat z měření, které mohou být použity pro další zpracování či testování a porovnávání.

Následná vizualizace dat slouží k efektivní prezentaci dat, tak aby bylo možné data snadno číst a hodnotit, nejen pro obsluhu magnetometru, ale i pro širší odbornou veřejnost, případně další studenty, kteří se budou zabývat stejnou problematikou. S přibývajícimi zkušenostmi s těmito senzory tak bude možné je aplikovat do dalších projektů či rozšířit ten stávající na větší úrovně chápání a použití.

Řešení práce je rozděleno do čtyř částí. V první části práce je popsán princip magnetorezistivních senzorů AMR, jejich aplikace v použitém přístroji a popis přístroje pro zjištění způsobu komunikace. Rovněž je zde uveden vlastní princip měření magnetického pole Země, tedy jeho parametry a požadovaný výstup měřených veličin.

V druhé části jsou prozkoumány metody vizualizace dat, kde jsou rozebrány požadavky na výslednou podobu vizualizace a porovnány s funkcionalitami daných metod.

Třetí část práce se věnuje samotné komunikaci s přístrojem a realizaci aplikace pro konfiguraci, měření, zpracování a archivaci dat.

Poslední část práce je věnována vybrané metodě vizualizace dat po zhodnocení předchozích variant a popsán způsob finální prezentace dat na webu.

## Seznam symbolů a zkratk

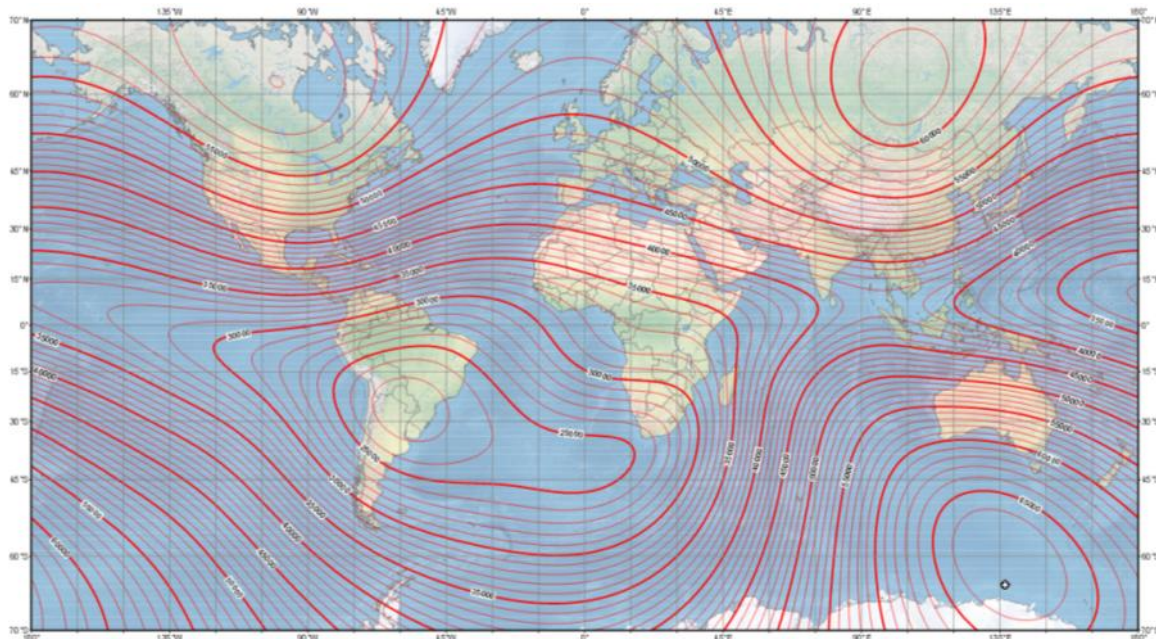
ZKRATKA	ANGLICKY	ČESKÝ VÝZNAM
AMR	anisotropic magnetoresistivity	anizotropická magnetorezistivita
ASCII	American standard code for information interchange	znaková sada amerického standardu pro výměnu informací
CSS	cascading style sheets	kaskádové styly pro webové stránky
DOM	document object model	objektový model dokumentu
GPS	global positioning system	globální polohovací systém
HTML	hypertext markup language	značkovací jazyk pro tvorbu webových stránek
HTTP	hypertext transfer protocol	přenosový protokol hypertextových dokumentů na internetu
I2C	inter-integrated circuit	sériová počítačová sběrnice
MySQL	my - structured query language	databázový systém
NOAA	national oceanic and atmospheric administration	instituce národní oceánské a atmosférické administrace
PHP	hypertext preprocessor	skriptovací jazyk pro tvorbu webových stránek
SVG	scalable vector graphics	škálovatelná vektorová grafika
UART	universal asynchronous receiver/transmitter	univerzální asynchronní sériové rozhraní
WMM	world magnetic model	světový model magnetického pole
$B_h$	[ $\mu\text{T}$ ]	horizontální složka magnetického pole
$B_t$	[ $\mu\text{T}$ ]	celková velikost magnetického pole
$B_x$	[ $\mu\text{T}$ ]	severní složka magnetického pole
$B_y$	[ $\mu\text{T}$ ]	východní složka magnetického pole
$B_z$	[ $\mu\text{T}$ ]	vertikální složka magnetického pole
$D$	[ $^\circ$ ' '' ]	magnetická deklinace
$\mathbf{H}$	[-]	vektor vnějšího magnetického pole
$I$	[ $^\circ$ ' '' ]	magnetická inklinace
$\mathbf{I}$	[-]	vektor proudu
$\mathbf{M}$	[-]	vektor magnetizace
$\varphi$	[ $^\circ$ ]	úhel vektoru $\mathbf{M}$ od osy anizotropie

## 1 Měření magnetického pole

Model magnetického pole Země je využíván v mnoha oborech lidské činnosti, např. ve vojenství, světovými ministerstvy obrany či v navigačních systémech na moři i na souši. Velikost magnetického pole se neustále mění, proto je důležité jej neustále měřit a aktualizovat hodnoty v každém místě. Oficiální světový model magnetického pole Země, „The World Magnetic Model” (aktuální podoba modelu viz Obr. 1), je zpracováván a aktualizován institucí NOAA (National Oceanic And Atmospheric Administration). Velikost magnetického pole Země nabývá hodnot v rozmezí 25 až 65  $\mu\text{T}$ , pro Českou Republiku je to hodnota okolo 49  $\mu\text{T}$  [14].

Cílem této bakalářské práce je vytvořit aplikaci pro zpracování a vizualizaci dat z použitého AMR magnetometru, aby bylo možné dále s daty pracovat, porovnávat naměřené hodnoty s modelem NOAA WMM, a hodnotit dosažené výsledky pro získání hlubších zkušeností s tímto typem senzorů.

Před vlastní realizací práce bylo nutné zhodnotit dostupné metody pro zpracování naměřených dat pomocí AMR magnetometru, zvolit vhodný způsob komunikace s přístrojem a práce s daty, rovněž určit periodu měření a množství dat pro ukládání do databáze. V další části bylo úkolem pro takto vzniklý soubor naměřených dat zvolit efektivní způsob vizualizace dat na webu pro prezentaci a další práci s daty.

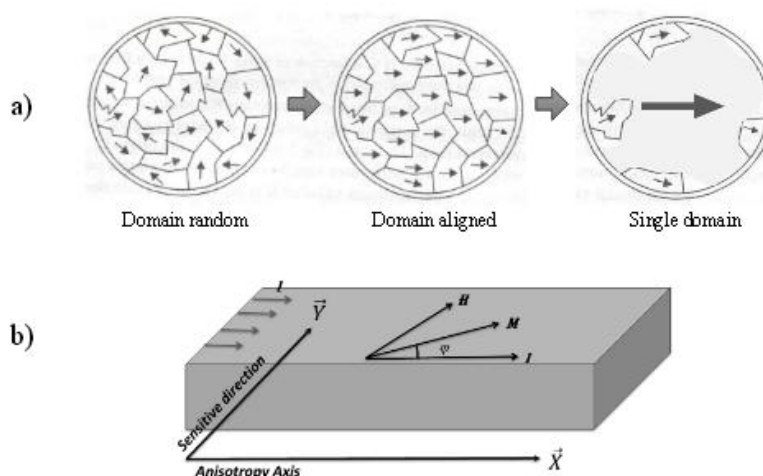


Obr. 1 - Světový model magnetického pole Země - The World Magnetic Model

## 1.1 Princip magnetorezistivních senzorů AMR

Princip anizotropní magnetorezistivity popsal v roce 1857 William Thompson ve své práci nazvané v anglickém originále Effect of magnetization on the electric conductivity of nickel and iron, kde uvádí tvrzení, že změny odporu byly odlišné v různých směrech v porovnání se směrem magnetizace pole. Trvalo dalších sto let, než mohla být technologie tenkých odporových pásek ze slitiny niklu a železa (Permalloy) použita v praktických aplikacích jako senzory [8].

Permalloy pásek je zmagetován ve směru jeho delší osy, která se nazývá osou snadné magnetizace neboli anizotropní osou. Měrný odpor je největší, pokud je vektor  $\mathbf{M}$  ve fázi s vektorem protékajícího proudu  $\mathbf{I}$ , nejmenší naopak ve směru kolmém. Vlivem vnějšího magnetického pole  $\mathbf{H}$ , působícího v citlivém směru, dochází k pootočení vektoru  $\mathbf{M}$ , a tedy ke změně úhlu  $\varphi$ . Výstup senzoru tedy závisí na úhlu  $\varphi$  mezi magnetizačním vektorem  $\mathbf{M}$  a osou anizotropie, který se mění vlivem vnějšího magnetického pole (viz Obr. 2). Problematika je dále podrobně popsána v [8] a [10].

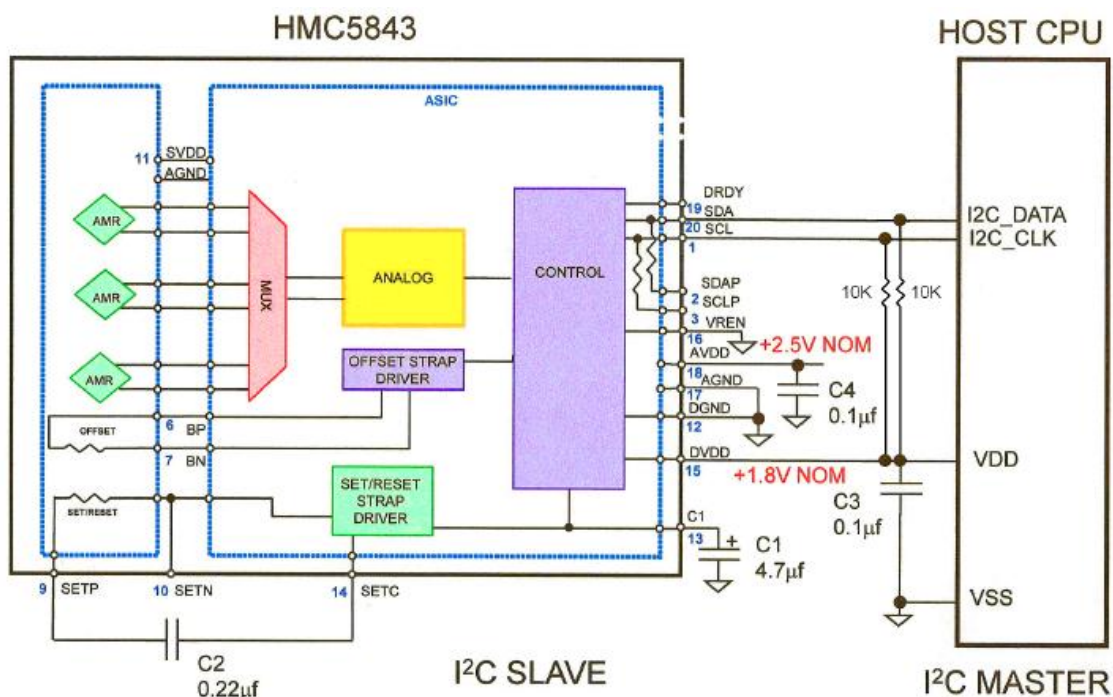


Obr. 2 - Magnetizační vektor a magnetické pole v AMR senzor [8]

## 1.2 Použitý čip magnetometru Honeywell AMR HMC5843

Přístroj používaný v průběhu vypracování této práce byl osazen čipem Honeywell AMR HMC5843 (blokové schéma viz Obr. 3). Veškeré skripty měřící aplikace jsou navrženy a testovány na tomto čipu. Do budoucna se předpokládá přechod na novější čip magnetometru s označením HMC5883L, který se liší v některých parametrech, zejména má větší citlivost, rozsah a rychlost měření. Nevýhodou je, že není možné provést prostou výměnu čipů, protože patice čipu není kompatibilní s původním čipem, je tedy nutné provést kompletní revizi plošného spoje, na který je čip osazován [5].

AMR senzory se vyznačují dobrou citlivostí měření, jsou nízkonákladové a nacházejí široké využití díky své konstrukci jako integrovaného obvodu pro osobní i automobilové navigace, mobilní telefony a další spotřební elektroniku [4].



Obr. 3 - Blokové schéma senzoru HMC5843 a sběrnice I<sup>2</sup>C k mikroprocesoru [4]

### 3osé AMR senzory (3-Axis AMR Sensors)

Oba zmíněné magnetometry jsou osazeny třemi senzory pro určení velikosti magnetického pole ve směru tří os. Jednotlivé senzory jsou odporové můstky, vytvořené pomocí tenkých odporových pásků, vyrobených ze slitiny niklu a železa (Permalloy). V přítomnosti magnetického pole dochází v takovém odporovém můstku ke změně napětí na jeho výstupech vlivem změny odporů pásků, jak je popsáno v předchozí části kapitoly [4].

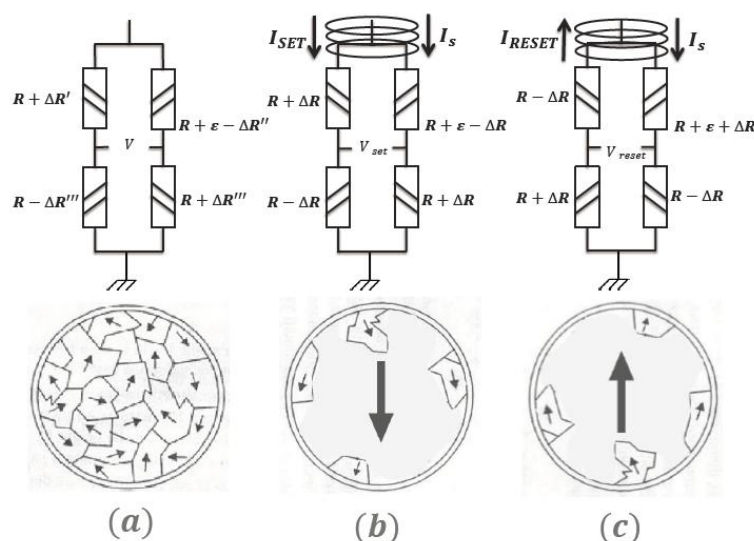
### Obvody cívek Set/Reset a Offset (Set/Reset and Offset Strap Circuit)

Senzory jsou vybaveny Offset a Set/Reset cívkami, které slouží pro demagnetizaci senzorových pásků, kompenzaci jejich offsetu a ke kalibraci senzorů.

Poté, co jsou senzorové pásky zmagetovány vlivem vnějšího magnetického pole, musí dojít k jejich opětovné demagnetizaci, resp. k jejich namagnetování ve směru procházejícího impulsu proudu, jinak by docházelo s ohledem na měření k trvalé magnetizaci pásků vlivem rušivých vnějších polí, tím ke ztrátě citlivosti magnetometru. Takové obnovení citlivosti je zajišťováno proudovými pulzy na cívku Set/Reset, která je integrována přímo do čipu HMC5843 [4] [10].

### Kompence offsetu metodou flipping

Permalloy pásy použité v senzorech nemají vždy absolutní přesnost a nejsou všechny naprosto shodných parametrů, a proto je pro přesné měření nutná kompenzace jejich offsetu. Pro tento účel se využívá metody „flipping“. Jedná se o využití pulzů na cívku Set/Reset, kdy jsou pásy senzoru nejdříve magnetovány pulzem Set v jednom směru, následně pak pulzem Reset ve směru opačném (viz Obr. 4). Výsledný rozdíl potom udává chybu, kterou přístroj kompenzuje [8].



Obr. 4 - Princip Set/Reset pulzů metody flipping [8]

### Test a kalibrace magnetometru

Offsetová cívka tedy není použita přímo pro vlastní kompenzaci offsetu odporových pásek, ale slouží pro kalibraci zařízení, resp. ke kontrole správnosti funkce magnetometru. Proces probíhá nejdříve měřením vnějšího magnetického pole, kterému předchází pulz Set. Následně je offsetovou cívkou protékán známý proud (5,5 mA) k vytvoření testovacího pole o velikosti asi 55  $\mu\text{T}$  (0,55 Gauss), které se sčítá s polem externím. První získaná hodnota se odečte od druhé a výsledná hodnota se zapíše do výstupních registrů. Výsledné hodnoty jsou rovněž kontrolou, zda již magnetometr nevykazuje známky poškození [4].

Magnetometr má možnost volit mezi několika úrovněmi nastavení zisku zesilovačů, viz Tab. 1. Z této tabulky je zřejmé, že se zvýšením zisku se zvyšuje rovněž citlivost měření, proto je nutné po každé změně zisku provést opětovnou kalibraci zařízení, k zajištění dostatečné přesnosti měření.

Tab. 1 - Přehled nastavení zisku a jeho závislost na citlivost měření [4]

GN2	GN1	GN0	Rozsah vstupního pole senzoru	Zisk [counts/Gauss]
0	0	0	+/- 0,7 Ga	1620
0	0	1	+/- 1,0 Ga	1300 (výchozí)
0	1	0	+/- 1,5 Ga	970
0	1	1	+/- 2,0 Ga	780
1	0	0	+/- 3,2 Ga	530
1	0	1	+/- 3,8 Ga	460
1	1	0	+/- 4,5 Ga	390
1	1	1	+/- 6,5 Ga (nedoporučuje se)	280

### Nastavení rychlosti převodu

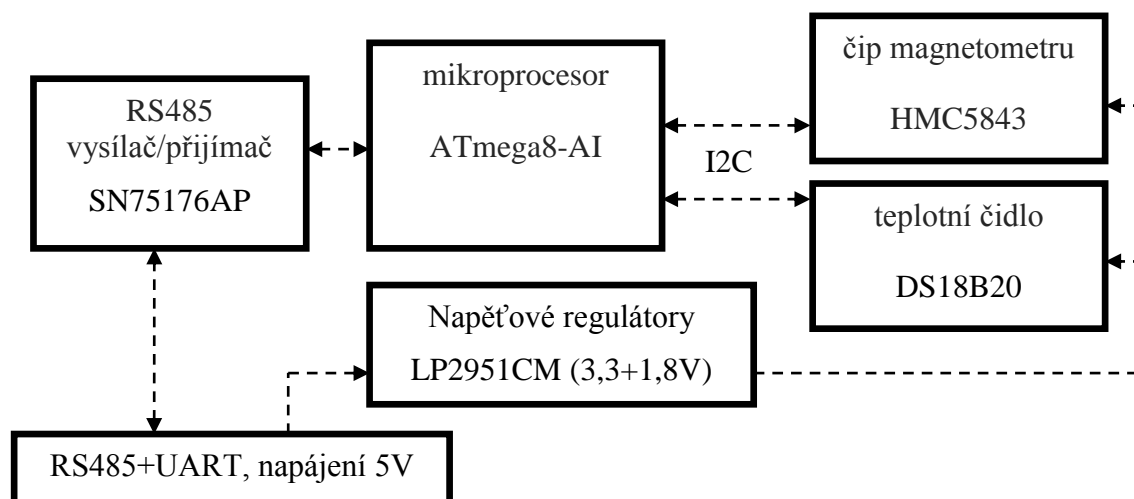
Rychlost převodu udává rychlost měření, tedy s jakou frekvencí mohou být data ze senzorů získávána. Možná nastavení jsou uvedeny v Tab. 2 - Nastavení rychlosti výstupu dat, jak je uvádí výrobce čipu [4].

Tab. 2 - Nastavení rychlosti výstupu dat

DO2	DO1	DO0	Minimální frekvence výstupu dat [Hz]
0	0	0	0,5
0	0	1	1
0	1	0	2
0	1	1	5
1	0	0	10 (výchozí)
1	0	1	20
1	1	0	50
1	1	1	nepoužito

### 1.3 Blokové schéma magnetometru

Čip magnetometru je osazen na desce kompletního přístroje, jak je naznačeno zjednodušeným blokovým schéma na Obr. 5. K čipu magnetometru je ještě přidáno teplotní čidlo, jeho význam je popsán v další části práce. Funkce senzorů je řízena pomocí mikroprocesoru ATmega8-AI protokolem I<sup>2</sup>C a vstupní linka k mikroprocesoru je zprostředkována přes UART a rozhraní RS485.



Obr. 5 - Blokové schéma použitého přístroje

#### 1.4 Měřené veličiny

Principem experimentu je měření a vyhodnocování měření magnetického pole Země v daném místě měření. Měření tedy musí probíhat v průběhu celého dne a z denních hodnot jsou potom vypočítány průměrné hodnoty pro daný den měření, aby bylo možné porovnávat hodnoty jak v průběhu dne, tak v průběhu celého měsíce.

Měření magnetického pole v každém místě je definováno sedmi parametry. Těmito parametry jsou [14]:

- magnetická inklinace ( $I$  [ $^{\circ}$  ' ']);
- magnetická deklinace ( $D$  [ $^{\circ}$  ' ']);
- horizontální složka mg. pole ( $B_h$  [ $\mu\text{T}$ ]);
- severní složka ( $B_x$  [ $\mu\text{T}$ ]);
- východní složka ( $B_y$  [ $\mu\text{T}$ ]);
- vertikální složka ( $B_z$  [ $\mu\text{T}$ ]);
- celková velikost ( $B_t$  [ $\mu\text{T}$ ]).

Jak již bylo předesláno v popisu blokového schéma přístroje, navíc je měřena teplota ( $T$  [ $^{\circ}\text{C}$ ]), a to kvůli testování závislosti teploty na měření magnetického pole.



## 2 Metody vizualizace naměřených dat

Jedním z hlavních cílů práce byla efektivní vizualizace získaných naměřených a vypočtených dat z magnetometru. Uvažováno bylo několik způsobů, při výběru byla zhodnocena kritéria pro výslednou podobu a potřebné funkcionality, které dané metody poskytují. Výchozími požadavky bylo zejména použití interaktivních grafů pro efektivní a přehledné zobrazení souborů denních a měsíčních naměřených hodnot magnetického pole a teploty. Výhodou je dynamická vizualizace v reálném čase, která umožňuje zobrazení nových naměřených hodnot. V této kapitole jsou popsány uvažované metody vizualizace.

### 2.1 ThingSpeak - portál pro zpracování a vizualizaci měřených dat

ThingSpeak je open-source aplikační platforma navržená pro Internet Of Things („Internet věcí“), která umožňuje práci s daty získanými z různých typů senzorů či akčních členů. V reálném čase je možné data zasílat a ukládat do databáze, následně je zpracovávat a zobrazovat. Kromě vizualizace naměřených dat je možné sledovat stavy akčních členů, jako jsou světla, termostat či čerpadla. Reálným příkladem použití je např. spínání termostatu pro ohřev vody na základě polohy uživatele pomocí GPS [12].

Pro práci s platformou je sice nutná, ale doposud bezplatná registrace a následné vytvoření vlastního kanálu „ThingSpeak Channel“, který slouží jako identifikace prostoru v databázi, kam může uživatel ukládat data ze svých senzorů nebo aktualizovat stavy akčních členů. Pro aktualizaci kanálu se zasílá HTTP požadavek GET, pro získání dat z kanálu požadavek POST. K dispozici je až 8 polí pro data v numerickém nebo alfanumerickém formátu, dále údaje o poloze a údaj o aktuálním stavu. Všechna tato data lze číst, nebo zapisovat [13].

Aby byl zajištěn plynulý a stabilní provoz platformy, je aktualizace dat zaslaných do databáze kanálu omezena na interval 15 sekund. Omezení je rovněž provedeno pro požadavky na výpis většího množství uložených výsledků, tedy pokud je požadováno více jak 100 výsledků v jednom výpisu, jsou tato data uložena do mezipaměti na 5 minut. Výpisy se 100 a méně výsledky do mezipaměti ukládána nejsou, což je vhodné pro využití v aplikacích zobrazující data v reálném čase [13].

Pro vlastní vizualizaci dat je možné využít přednastavených grafů, pro které lze nastavit základní parametry, jako jsou název grafu, názvy os  $x$  a  $y$ , barva datové řady, barva pozadí, výška či šířka. Typ zobrazeného grafu lze zvolit jako spojnicový (line),

sloupcový (column), pruhový (bar) nebo spojnicový s proloženou křivkou (spline) [13].

V případě, kdy je nevhodné, nebo nedostačující zobrazení dat pomocí základních grafů, nabízí platforma sekci „Plugins“ (zásuvné moduly), kde může uživatel zapisovat kód pomocí HTML, CSS a JavaScript. K dispozici je bohužel jen omezená nabídka již naprogramovaných modulů, jako je číselníkový graf „Google Gauge“, či graf s více datovými řadami. Součástí platformy je ale rovněž uživatelské fórum, kde uživatel může najít mnoho návodů při práci se zásuvnými moduly a jejich tvorbou se specifickými parametry [13].

## 2.2 GD - grafická knihovna pro PHP

GD knihovna je rozšíření programovacího jazyka PHP pro tvorbu a manipulaci s grafickými objekty. Obecně lze říci, že GD knihovna umí pracovat s formáty JPEG, PNG, GIF, XBM, XPM, WBMP, WebP. Je ovšem třeba vést v patrnosti, že kompatibilita s různými formáty obrázků může být závislá na konkrétní verzi samotné GD knihovny, ale i platformy PHP, a proto je někdy pro správnou funkci potřeba instalovat dodatečné knihovny [9].

Nevýhodou je, že pomocí GD knihovny v PHP nelze vytvářet interaktivní grafy, např. není možné zobrazovat vysvětlivku k datovým bodům (tzv. tooltip).

Výhodou je samotná implementace GD knihovny do kódu PHP skriptů, které následně generují HTML kód, takže není nutné při tvorbě webové stránky kombinovat PHP s dalším jiným jazykem.

## 2.3 D3.js – grafická knihovna pro JavaScript

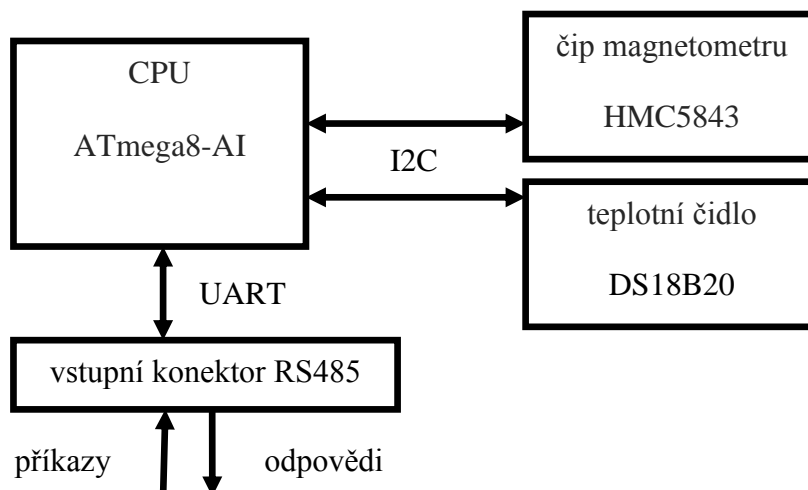
D3.js je knihovna napsaná v jazyce JavaScript, která slouží pro širší využití v oblasti vizualizace dat. Umožňuje svázat data do Document Object Model (DOM) a aplikovat jejich transformace k vytvoření tabulky nebo interaktivního grafu [2].

D3 není uzavřená platforma, umožňuje propojení se standardy HTML, SVG či CSS, a nabízí širokou škálu možností vizualizace [2].

Výhodou shledávám velké množství „hotových“ kódů pro nejrůznější způsoby vizualizace, od klasických grafů až po méně standardní prezentaci dat, která je ovšem v mnoha případech nejefektivnějším způsobem vizualizace v dané problematice.

### 3 Komunikace s přístrojem a funkční skripty

Vlastní komunikace s přístrojem probíhá pomocí příkazů v podobě prostých textových řetězců v ASCII, které jsou po fyzické vrstvě RS485+UART přenášeny do mikroprocesoru, kde jsou zpracovávány a převáděny na úroveň protokolu I<sup>2</sup>C, pomocí kterého komunikuje mikroprocesor s čipem magnetometru a teplotním čidlem. Stejným způsobem funguje komunikace opačným směrem, odpovědi na zasláné příkazy jsou tedy opět uživateli zobrazovány ve formě textových řetězců (názorně viz Obr. 6).



Obr. 6 - Zjednodušené blokové schéma komunikace s přístrojem

Základní příkazy pro zprostředkování komunikace mezi vstupní fyzickou vrstvou, mikroprocesorem, čipem magnetometru a teplotním čidlem jsou napsány v jazyce C. Přehled vlastních příkazů a základní popis jejich funkce, kterou vykonávají, je pro představu uveden v Tab. 3. Příkazy jsou základem pro tvorbu dalších měřících, kalibračních, konfiguračních a archivačních skriptů, které jsou popsány v následujících částech kapitoly.

Vlastní skripty jsou napsány v programovacím jazyce Python, ve verzi 2.7.3, jenž se s výhodou nabízí, díky své jednoduché struktuře objektového programování. Python je z řady vyšších programovacích jazyků, podobně jako C, C++, Perl, a Java [3].

Jedná se o snadno pochopitelný, objektově orientovaný jazyk, fungující na mnoha platformách. Pro vytváření skriptů jsem zvolil tento programovací jazyk zejména pro jeho jednoduchou syntaxi, výsledný kód je na pohled jednodušeji uspořádaný a velice rychle pochopitelný, je tedy snazší vytvořené skripty modifikovat. Další výhodou je snadná podpora pro zpracování dat a obsluhu periférií. Zdrojové kódy vytvořené v jazyce Python není nutné kompilovat díky interpreteru.

Tab. 3 - Funkční příkazy magnetometru

Příkaz	Funkce příkazu
<b>IDN?</b>	Vypíše identifikační řetězec
<b>CHID?</b>	Vypíše identifikaci čipu
<b>M?</b>	Změří a vypíše měřené hodnoty
<b>LOOP</b>	Spustí měření ve smyčce (počet iterací přednastaven)
<b>BIAS<math>x</math></b>	Zapnutí kalibračního magnetického pole, $x \in \langle 0;3 \rangle$
<b>GAIN<math>x</math></b>	Nastavení zisku zesilovačů, $x \in \langle 0;7 \rangle$ (0 - největší zisk, citlivost)
<b>RATE<math>x</math></b>	Nastavení rychlosti převodu, $x \in \langle 0;6 \rangle$
<b>DEMO</b>	Spustí demonstrační režim
<b>KONST?</b>	Vypíše kalibrační konstanty
<b>KALIB</b>	Spustí interní kalibrační algoritmus
<b>FORM<math>x</math></b>	Volba výstupního formátu dat $x \in (0 - counts, 1 - pravoúhlé, 2 - polární, 3 - citlivosti)$
<b>T?</b>	Změří a vypíše aktuální teplotu

Důležitou knihovnou používanou pro připojení na sériový port, na kterém je připojen přístroj, a konfiguraci portu uvnitř skriptů Python, je knihovna pySerial, verze 2.7. Její implementace v kódu je velmi jednoduchá, před otevřením sériového portu je však vždy nutné nejdříve nastavit parametry komunikace.

Veškeré dále popisované zdrojové kódy jsou v kompletní podobě v příloze na přiloženém DVD.

### 3.1 Modul pro nastavení sériové komunikace portcfg.py

Tento modul byl vytvořen pro snadnou konfiguraci přístupového portu magnetometru. Cílem bylo zajistit konfiguraci parametrů pro připojení z jednoho zdrojového souboru, aby se tak předešlo nevýhodné konfiguraci portu v každém skriptu zvlášť, což by mohlo vést k chybám při změně parametrů připojení a složité aktualizaci v každém skriptu přistupujícímu k zařízení.

Rovněž samotné nastavení parametrů portu probíhá mimo konfigurační modul. Zdrojem parametrů je konfigurační soubor *config.cfg*, snadno upravovatelný v nativním textovém editoru operačního systému.

```

1  port=/dev/ttyS0
2  baudrate=19200
3  timeout=1
4  bytesize=serial.EIGHTBITS
5  parity=serial.PARITY_NONE
6  stopbits=serial.STOPBITS_ONE

```

*Struktura konfiguračního souboru config.cfg*

Proces načtení a konfigurace portu v každém skriptu, který komunikaci se sériovým portem využívá, probíhá importováním modulu portcfg.py, spuštěním konfiguračního skriptu a vytvořením sériového objektu pro další práci se sériovým portem.

```
1 import portcfg
2
3 # Serial port object configured by portcfg module
4 portcfg.set_port()
5 ser = portcfg.ser
```

*Volání pro přístup na sériový port a nastavení parametrů*

### **Funkce set\_port()**

Volání této funkce spustí vlastní nastavení parametrů sériového portu. Skript načte data ze souboru *config.cfg* a testuje jeho základní strukturu. Pro jednotlivá nastavení ještě testuje povolené hodnoty nastavení. Pokud se data neshodují, skript skončí chybou. Povolené hodnoty jsou v souladu s dokumentací pro knihovnu pySerial [7].

```
1 # Available options
2 options = {0: port,
3           1: baudrate,
4           2: timeout,
5           3: bytesize,
6           4: parity,
7           5: stopbits,
8           }
```

*Testování struktury konfiguračního souboru probíhá kontrolou povolených voleb*

Následující část zdrojového kódu je pouze příkladem testování povolených vstupních hodnot pro předchozí volby.

```
1 # Setting baudrate
2 def baudrate(n):
3     baudrates = [50, 75, 110, 134, 150, 200, 300, 600, 1200, 1800,
4                 2400, 4800, 9600, 19200, 38400, 57600, 115200]
5     for num in range(len(baudrates)):
6         if cval[n] == baudrates[num]:
7             ser.baudrate = cval[n]
8             break
9     elif num == len(baudrates)-1:
10        print "Not allowed baudrate."
11        os._exit(1)
```

*Povolené parametry volby baudrate a testování vstupních hodnot konfiguračního souboru*

## **3.2 Inicializační skript mgme-init.py**

Inicializační skript provádí nastavení magnetometru. Po konfiguraci sériového portu volá postupně příkazy RATE, GAIN a FORM viz kapitola 3.

Názvy i povolené hodnoty příkazů RATE a GAIN jsou přímo převzaty od výrobce čipu magnetometru [4] a jsou implementovány do aplikace z důvodů důsledného testování těchto parametrů. Význam těchto parametrů jsou uvedeny v kapitole 1.2, včetně významů jednotlivých hodnot nastavení.

Po úspěšné inicializaci se potvrzení ve formě textu s časovou značkou zapíše do souboru *init.dat*, který slouží pro zpětnou informaci, kdy došlo k inicializaci magnetometru. K této inicializaci dochází zpravidla po restartu počítače, ke kterému je přístroj připojen. Uživatel tak může zpětně zjistit, kdy došlo k opětovné inicializaci zařízení, např. z důvodu, kdy systém přestane fungovat a musí být restartován nebo při výpadku napájení, apod.

### 3.3 Kalibrační skript mgme-calib.py

Význam a princip kalibrace zařízení je popsán v kapitole 1.2. Kalibrační skript se spouští vždy po posledním měření daného dne. Je rovněž možné nastavit, aby se kalibrace zařízení prováděla před každým měřením, jak je popsáno dále.

Kalibrace zařízení je prováděna příkazem KALIB, po vlastním provedení interní kalibrace zařízení jsou kalibrační konstanty včetně časové značky uloženy do souboru *calibration.dat*, aby mohl uživatel zpětně rozpoznat případnou vadu čipu magnetometru.

### 3.4 Testovací skript mgme-test.py

Pro ruční kontrolu funkce, nebo pro ukázkou funkce magnetometru slouží testovací skript, který se připojí k zařízení, přečte a vypíše aktuální hodnoty z měření na obrazovku. Je možné tak ověřit správné nastavení a formát výstupu dat, např. při sestavování zařízení po odstávce.

### 3.5 Modul pro přístup na server ThingSpeak.com

Se stejným cílem jako u výše popisovaného modulu pro sériovou komunikaci byl vytvořen modul pro komunikaci se serverem ThingSpeak.com, aby mohl být snadno modifikován a používán ve skriptech. Modul obsahuje dvě funkce pro zasílání dat do polí databáze určené pro data, z kterých se generuje denní a měsíční graf hodnot.

Funkce obsahují vstupní parametry, které jsou ve zdrojovém kódu příslušného skriptu nahrazeny hodnotami, které chceme zaslat na určené pozice v databázi kanálu a tyto pozice jsou označeny parametrem *field#*. Vstupními daty jsou celková velikost magnetické indukce  $B_t$  a teplota  $T$ . Pro parametr *key* musí být uveden vstupní klíč

vygenerovaný v aplikaci ThingSpeak pro daný kanál [1].

```

1  # Pushing data to ThingSpeak.com
2  import httplib, urllib
3
4  # Sending values for daily tab
5  def send_daily(bttotal, temp):
6      params = urllib.urlencode({'field1': bttotal, 'field2': temp,
7                                'key': 'A1B2C3D4E5F6G7H8G'})
8      headers = {"Content-type": "application/x-www-form-
9                urlencoded", "Accept": "text/plain"}
10     conn = httplib.HTTPConnection("api.thingspeak.com:80")
11     conn.request("POST", "/update", params, headers)
12     response = conn.getresponse()
13     print response.status, response.reason
14     data = response.read()
15     conn.close()

```

#### *Funkce pro zasilání dat do denní databáze kanálu*

Stejně jako lze data zasílat, je možné data ze serveru získávat zpět. V ovládacích skriptech tato funkcionality zatím uplatněna nebyla.

### 3.6 Struktura databáze MySQL pro ukládání naměřených hodnot

Před tím, než byl vytvořen skript pro měření hodnot, musela být vytvořena databáze ve vhodném formátu, kam se následně data ukládají a odkud lze data opět načíst.

Použita je databáze MySQL, kde jsou vytvořeny dvě oddělené tabulky pro ukládání hodnot z průběžného denního měření a pro hodnoty denních průměrů.

Struktura databáze byla zvolena tak, aby bylo možné ukládat všechny měřené veličiny v souladu s kapitolou 1.4 a každé měření je opatřeno časovou značkou měření, jak je uvedeno v Tab. 4.

*Tab. 4 - Struktura tabulek v databázi MySQL*

#	Název pole	Typ pole	Popis
1	tstmp	timestamp	časová značka nahrání dat
2	b_x	float	velikost komponenty Bx
3	b_y	float	velikost komponenty By
4	b_z	float	velikost komponenty Bz
5	total	float	celková velikost mg. pole Bt
6	horiz	float	velikost horizontální složky Bh
7	decl	float	hodnota magnetická deklinace D
8	incl	float	hodnota magnetické inklinace I
9	tpr	float	teplota T

### 3.7 Měřicí skript mgme-meas.py

Hlavní měřicí skript vykonává měřicí funkci magnetometru, tedy měření velikosti magnetického pole ve třech směrech os, tedy složkách  $B_x$ ,  $B_y$ ,  $B_z$  a data z teplotního čidla T. Skript je spouštěn každou hodinu během dne. Jako úložiště a zdroj dat pro další zpracování je použita databáze MySQL, s jejíž obsluhou souvisí nutný import příslušné knihovny do hlavičky kódu. Před vlastním měřením je možné spustit kalibraci, v základním nastavení skriptu je tato funkce vypnuta.

```
1 # If true, the calibration will be performed before every
  measurement.
2 calib = False;
```

#### *Nastavení parametru kalibrace*

Vlastní měření dále probíhá voláním funkce LOOP. Tato funkce provede dvacet iterací měření, data jsou kontrolována, zda obsahují správnou strukturu dat a hodnoty se před dalším zpracováním ukládají do proměnných typu pole.

```
3 # Performing the measurement.
4 bxa = []; bya = []; bza = []
5 ser.write("loop\n")
6 for i in range(0, 60):
7     try:
8         rep = ser.readline()
9     except ser.SerialTimeoutException:
10        print "Timeout on 'LOOP' command."
11        os._exit(1)
12    rep = rep.strip()
13    arep = rep.split(" ")
14    if len(arep) != 4 or rep[0] != "B" or arep[1] != "=":
15        print "Error on 'LOOP' command."
16        os._exit(1)
17    a = float(arep[2])
18    if rep[2] == "X":
19        bxa.append(a)
20    if rep[2] == "Y":
21        bya.append(a)
22    if rep[2] == "Z":
23        bza.append(a)
```

#### *Prováděcí smyčka měření*

V závislosti na aktuální poloze magnetometru je provedena transformace os dle aktuální polohy magnetometru. Tato transformace os je do skriptu zanesena ručně, je tedy nutné po změně polohy přístroje ručně upravit tuto transformaci.

Z naměřených hodnot jsou pomocí matematických výpočtů provedeny výpočty zbylých složek magnetického pole, jak jsou uvedeny v kapitole 1.4.



```
1 # Compute Earth's magnetic field properties and averaging.
2 bx = 0; by = 0; bz = 0; bt = 0;
3 bh = 0; dek = 0; ink = 0;
4 cnt = len(bxa)
5 for i in range(0, cnt):
6     bx = bx + bxa[i]
7     by = by + bya[i]
8     bz = bz + bza[i]
9     bt = bt + sqrt(bxa[i] * bxa[i] + bya[i] * bya[i] + bza[i] *
    bza[i])
10    bht = sqrt(bxa[i] * bxa[i] + bya[i] * bya[i])
11    bh = bh + bht
12    dek = dek + 180 / pi * atan2(bya[i], bxa[i])
13    ink = ink + 180 / pi * atan2(bza[i], bht)
14 bx = bx / cnt
15 by = by / cnt
16 bz = bz / cnt
17 bt = bt / cnt
18 bh = bh / cnt
19 dek = dek / cnt
20 ink = ink / cnt
21 print "Values taken: ", bx, by, bz, " uT"
```

### *Provádění výpočtu a průměrování složek magnetického pole*

K aktuálnímu měření magnetické indukce je provedeno rovněž měření teploty pomocí teplotního čidla pro účely porovnání závislosti změny okolní teploty na měřených hodnotách indukce. Pro měření teploty je použit příkaz T?

```
1 # Performing the temperature measuring
2 tpr = 0
3 ser.write("T?\n")
4 try:
5     rep = ser.readline()
6 except ser.SerialTimeoutException:
7     print "Timeout on 'LOOP' command."
8     os._exit(1)
9 rep = rep.strip()
10 arep = rep.split(" ")
11 if len(arep) != 4 or rep[0] != "T" or arep[1] != "=":
12     print "Error on 'T?' command."
13     os._exit(1)
14 tpr = float(arep[2])
```

### *Skript pro měření teploty*

Změřená a zpracovaná data jsou následně zaslána importovaným modulem na server ThingSpeak.com do pole pro zaznamenávání denního souboru hodnot a poté uložena do lokální databáze MySQL v tabulce denních dat pro další možnosti práce s daty.

Záznamy v databázi jsou na závěr skriptu testována na čas uložení a data starší než 48 hodin jsou smazána. Tímto dojde pouze k uvolnění databáze, smazaná data však nejsou ztracena, protože vždy dochází k archivaci dat ve skriptech popsanych dále.

### 3.8 Archivace a zpracování denních a měsíčních souborů dat

Jak již bylo poznamenáno výše, data jsou průběžně archivována z databáze do souborů *daily.dat* a *monthly.dat* a starší data jsou z databáze smazána.

Po posledním měření daného dne je spuštěn skript *mgme-midnight.py*, který archivuje a zpracovává naměřená data za posledních 24 hodin. Nejdříve jsou z lokální databáze MySQL načtena data za posledních 24 hodin měření a provedena jejich archivace do souboru *daily.dat*. Následně jsou ze souboru denních hodnot opět vypočítány všechny složky magnetického pole, jako je tomu v případě hlavního měřicího skriptu.

Změřená a zpracovaná data jsou následně zaslána pomocí importovaného modulu do databáze serveru ThingSpeak.com do pole pro zaznamenávání měsíčního souboru denních průměrných hodnot a poté uložena do lokální databáze MySQL v tabulce měsíčních dat. Data starší než 24 hodin jsou smazána na základě parametru *dailydel*.

Poslední ovládací skript *mgme-clean-monthly.py* slouží k archivaci měsíčního souboru denních průměrných hodnot složek magnetického pole. Data jsou z databáze přečtena a uložena do souboru *monthly.dat* a zároveň dojde k vyčištění databáze o již archivovaná data.

Data uložená do archivačních souborů jsou uložena ve formátu prostého textu a slouží zejména zájemcům o další výzkum.

## 4 Vizualizace dat na webu

Pro výběr nejvhodnějšího způsobu vizualizace dat byla nejdříve posouzena kritéria očekávaná ve výsledné prezentaci dat.

Výstupem prezentace dat z měření magnetometru je možnost porovnat měřená data v průběhu času. Jak je popsáno v kapitole 1.4, k dispozici jsou denní data měřená v hodinovém intervalu a data měsíční, tedy denní průměry hodnot v průběhu měsíce. Výstupem je aktuální hodnota, soubor denních a měsíčních hodnot z měření magnetického pole, jejich vypočítaných složek a okolní teploty.

K zobrazení souborů hodnot je zvoleno zobrazení v klasickém liniovém grafu a přehledové tabulce. Nejsou zde kladeny velké požadavky na metodu programování, ale pro snadné porovnávání dat mezi tabulkou a grafem jsem shledal důležité, aby graf byl interaktivní, tedy zejména s možností zobrazení popisky k aktuálně prohledávané hodnotě grafu. Jak je uvedeno v kapitole 2.2, není možné k této funkci využít PHP knihovnu GD.

K zobrazení aktuální hodnoty byla kritéria taková, aby bylo možné zobrazit aktuální hodnotu v pásmu minimální a maximální hodnoty daného měřeného období, vedoucí k tomu, že uživateli bude na první pohled zřejmé, jak se aktuální hodnota jeví v porovnání s celým souborem dat.

Po zhodnocení těchto kritérií bylo upuštěno od varianty, kdy by bylo možné kompletní vizualizaci provést na platformě PHP a její knihovny GD. Po prozkoumání dalších variant vizualizace mě svými možnostmi nejvíce zaujala platforma serveru ThingSpeak.com, která nabízí jednoduchý způsob vizualizace souboru dat pomocí interaktivních grafů v JavaScript. Není nutné ale psát celý kód, způsob zobrazení základních grafů je totiž možné nastavit přímo ve webové aplikaci ThingSpeak a hotový graf jednoduše vložit na svou webovou stránku kliknutím na funkci pro generování *Chart Iframe* kódu, který zachová v aplikaci nastavené parametry grafu, které lze později upravit přímo v kódu.

```
<iframe width="450" height="260" style="border: 1px solid #cccccc;"
  src="http://api.thingspeak.com/channels/17573/charts/1?width=450&he
  ight=260&timescale=10&results=24&dynamic=true&color=blue&yaxis=Bt%2
  0%5BnT%5D&xaxis=time&title=Magnetic%20Field" ></iframe>
```

*Chart Iframe - generovaný kód aplikací ThingSpeak*

#### 4.1 Vizualizace denního souboru hodnot

Jak jsem již naznačil výše, pro vizualizaci naměřených souborů dat za 24 hodin a denních průměrů za měsíc jsem zvolil způsob zobrazení dat v přehledové tabulce a interaktivních grafech zobrazující umístěnými nad tabulkou, pomocí kterých je možné data vizuálně porovnat.

Samotnou tabulku (viz Obr. 7) dat generuji pomocí PHP kódu, zapsaném v souboru *daytab.php*. Tabulka zobrazuje devět sloupců hodnot časovou známku, kdy byla data naměřena, hodnoty naměřené zvlášť pro jednotlivé osy magnetometru, jejich geometrický průměr celkové magnetické indukce, vypočítané parametry horizontální složku, deklinaci, inklinaci a naměřenou teplotu.

Timestamp	Bx [uT]	By [uT]	Bz [uT]	Total [uT]	Horizontal [uT]	Declination	Inclination	Temperature [°C]
2015-05-02 18:10:14	26.80	-23.01	39.96	53.33	35.32	-40°38'59"	48°31'20"	24.4
2015-05-02 17:10:15	26.89	-23.05	39.90	53.35	35.41	-40°36'15"	48°24'33"	24.4
2015-05-02 16:10:14	26.82	-23.04	39.93	53.33	35.36	-40°39'55"	48°28'25"	24.3
2015-05-02 15:10:15	26.75	-23.05	39.99	53.34	35.31	-40°44'53"	48°33'32"	24.3
2015-05-02 14:10:15	26.82	-23.14	39.96	53.40	35.43	-40°47'23"	48°26'34"	24.2
2015-05-02 13:10:15	26.78	-23.07	39.89	53.30	35.35	-40°44'32"	48°27'16"	24.3
2015-05-02 12:10:15	26.81	-23.09	39.98	53.39	35.38	-40°44'8"	48°29'43"	24.1
2015-05-02 11:10:15	26.87	-23.20	39.96	53.45	35.50	-40°47'57"	48°22'50"	23.9
2015-05-02 10:10:15	26.95	-23.20	40.07	53.57	35.56	-40°43'5"	48°24'42"	23.6
2015-05-02 09:10:15	26.81	-23.14	40.03	53.45	35.41	-40°47'42"	48°30'9"	23.3
2015-05-02 08:10:15	26.81	-23.15	40.18	53.57	35.42	-40°48'38"	48°35'55"	23.1
2015-05-02 07:10:15	26.96	-23.09	39.95	53.44	35.50	-40°34'48"	48°22'24"	23.1
2015-05-02 06:10:14	26.88	-23.10	39.93	53.38	35.44	-40°40'20"	48°24'32"	23.1
2015-05-02 05:10:14	26.84	-22.98	40.18	53.51	35.33	-40°33'40"	48°40'28"	23.1
2015-05-02 04:10:14	26.92	-23.16	40.11	53.57	35.51	-40°42'15"	48°28'44"	23.2
2015-05-02 03:10:14	26.84	-23.31	40.09	53.58	35.55	-40°58'28"	48°26'9"	23.3
2015-05-02 02:10:14	26.90	-23.09	39.99	53.44	35.45	-40°38'8"	48°26'19"	23.3
2015-05-02 01:10:15	26.89	-23.22	40.04	53.53	35.53	-40°48'51"	48°24'36"	23.4
2015-05-02 00:10:14	26.91	-23.11	40.16	53.58	35.47	-40°38'55"	48°32'37"	23.4
2015-05-01 23:10:15	26.89	-23.20	40.10	53.57	35.52	-40°47'34"	48°28'7"	23.4
2015-05-01 22:10:15	26.87	-23.33	40.17	53.66	35.58	-40°57'56"	48°27'48"	23.5
2015-05-01 21:10:14	26.84	-23.37	40.14	53.65	35.59	-41°2'15"	48°26'11"	23.6
2015-05-01 20:10:14	26.66	-23.25	39.97	53.38	35.38	-41°5'14"	48°29'20"	23.7
2015-05-01 19:10:15	26.77	-23.21	40.08	53.50	35.43	-40°55'16"	48°31'24"	23.9

Obr. 7 - Tabulka denních hodnot

Pro správné zobrazení parametrů inklinace a deklinace ve stupních, minutách a sekundách, bylo nutné provést jejich převod z desetinného formátu čísla. Pro tento účel slouží další samostatný PHP skript *libmag.php*, který je následně importován do skriptů vyžadující převod hodnot.

```

1 function dec2dms($dec)
2 {
3     $vars = explode(".", $dec);
4     $deg = $vars[0];
5     $tempma = "0." . $vars[1];
6
7     $tempma = $tempma * 3600;
8     $min = floor($tempma / 60);
9     $sec = number_format($tempma - ($min*60), 0);
10
11     return($deg . "°" . $min . "'" . $sec . "'");
12 }

```

*PHP skript pro převod desetinného čísla na stupně, minuty a sekundy*

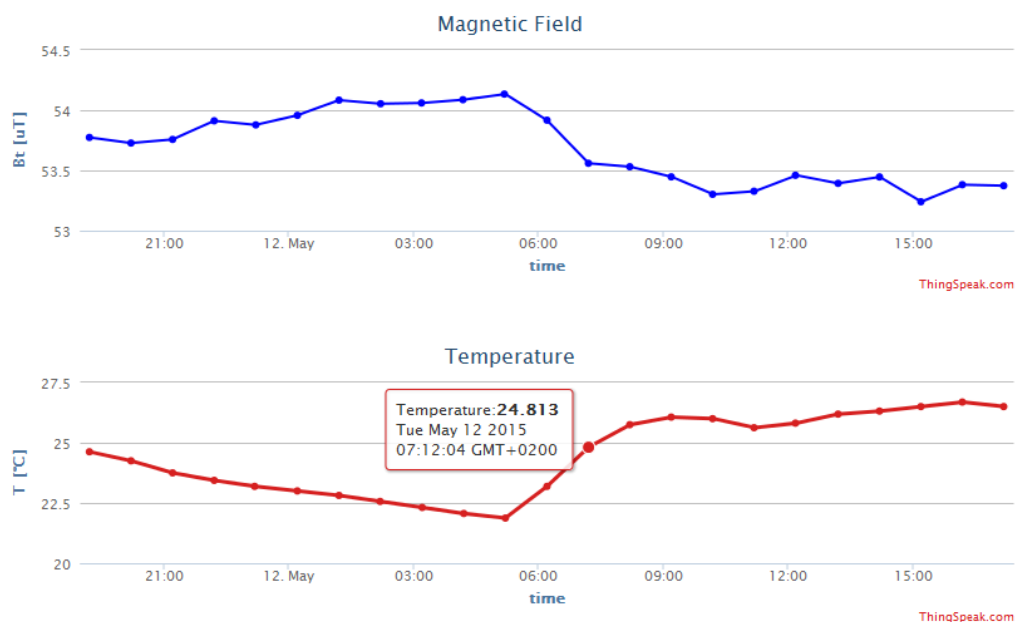
Data jsou následně do vytvořené PHP tabulky nahrána z MySQL databáze výběrem dat z posledních 24 hodin seřazených sestupně.

Grafy hodnot nad tabulkou zobrazují průběhy hodnot celkové velikosti magnetické indukce  $B_t$  [ $\mu\text{T}$ ] okolní teploty  $T$  [ $^{\circ}\text{C}$ ]. Samotné vložení grafů do kódu je realizováno pomocí generovaného kódu Chart Iframe ze serveru ThingSpeak.com, jak jsem již popsal výše. Byla pouze upravena velikost grafu a odebrán rámeček kolem grafu.

Data z měření jsou zasílána z měřících skriptů do příslušných polí kanálu, vytvořeném v aplikaci ThingSpeak, plnění grafu daty tedy probíhá automaticky. Nastavení zobrazovaného úseku dat bylo již předem nastaveno ve webové aplikaci, jak je zřejmé z Obr. 8.

Obr. 8 - Nastavení požadovaného zobrazení grafu v aplikaci ThingSpeak [11]

Nastaveno je zobrazení 24 hodnot, v intervalu měření po hodině se tedy jedná o zobrazení 24 hodinového úseku měření. Minimum a maximum osy y je přiřazováno automaticky dle rozsahu zobrazovaných hodnot, není tedy pevně nastaveno. Výsledné zobrazení na Obr. 9 znázorňuje rovněž popisku datové řady při zkoumání grafu.

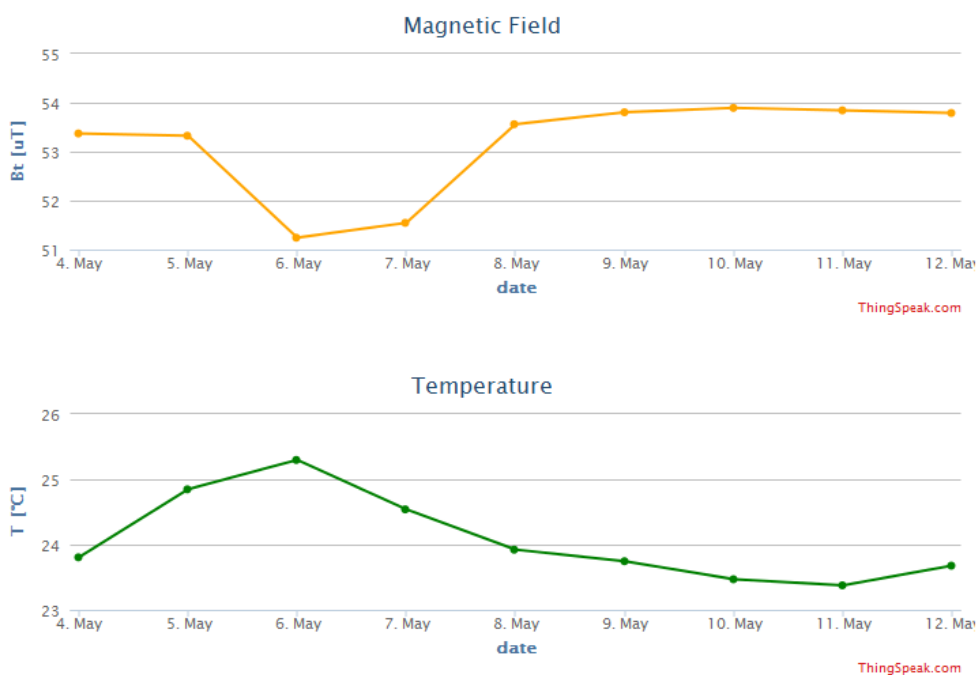


Obr. 9 - Výsledné zobrazení denních grafů a popiska datové řady

## 4.2 Vizualizace měsíčního souboru hodnot

Vizualizace měsíčního souboru dat probíhá obdobně jako u denního zobrazení dat, pomocí PHP kódu uloženého v souboru *monthtab.php*. Zobrazena je tedy přehledová tabulka stejné struktury, pouze časová známka je omezena na zobrazení ve formátu *yyyy-mm-dd*, není zde adekvátní zobrazovat čas uložení měření do databáze, jelikož se jedná o denní vypočítané průměry hodnot. Tabulka zobrazuje 30 denní data.

Nad tabulkou jsou opět zobrazovány grafy z platformy ThingSpeak, (viz Obr. 10).



Obr. 10 - Grafy zobrazující měsíční soubor dat z denních průměrů

V tomto případě byly grafy vytvořeny jako plugin s upraveným kódem, protože nebylo možné ve standardním nastavení upravit časovou osu tak, její popisky korespondovaly se zobrazenými daty. Hodnoty jsou sestaveny a nahrávány do databáze před půlnocí každého dne, automatický graf tedy hodnotu z daného dne zobrazoval až u popisky dne následujícího.

### 4.3 Vizualizace aktuální hodnoty měření

Pro zobrazení aktuální hodnoty měření bylo nutné zvolit odlišný způsob vizualizace, než je tomu u předešlých zobrazení souborů denních a měsíčních hodnot. Jako u předešlých případů jsou naměřené a spočtené hodnoty z aktuálního měření zobrazena v přehledové tabulce vytvořené v PHP skriptu, viz Obr. 11.

Taken at 2015-05-10 09:10:15

Bx:	27.30	uT
By:	-22.96	uT
Bz:	40.12	uT
B Total:	53.69	uT
B Horizontal:	35.67	uT
Declination:	-40°3'53"	deg
Inclination:	48°21'34"	deg
Temperature:	23.7	°C

Obr. 11 - Zobrazení dat z aktuálního měření

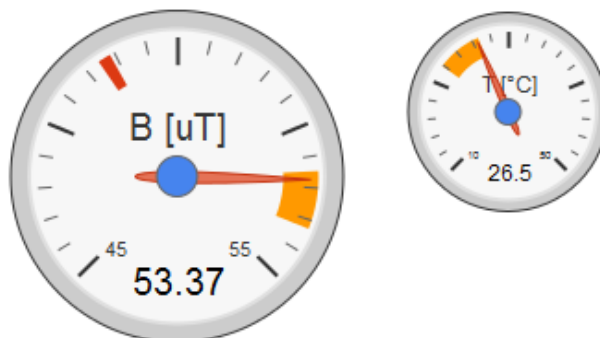
Pro grafické zobrazení aktuálních hodnot byl zvolen způsob zobrazení pomocí ThingSpeak plugin Google Gauge, který je v základních parametrech již naprogramován, bylo však nutné upravit kód tak, aby vhodně odpovídal jednak rozsahu měřených hodnot a cílenému zobrazení aktuálních hodnot v pásmu hodnot z celého souboru dat, jak jsem předeslal v úvodu kapitoly 4.

Google Gauge je zobrazení dat připomínající analogové měřicí přístroje. Data jsou zobrazena v pásmu analogové stupnice, kde aktuální hodnota je zobrazena výchylkou červeného ukazatele. Výsledné zobrazení je na Obr. 12.

Podobně jako u zobrazení denních a měsíčních hodnot jsem zvolil pro vizualizaci aktuální hodnoty celkovou velikost magnetické indukce  $B_t$  [uT] a okolní teplotu  $T$  [°C].

Pro lepší čitelnost dat jsem zvolil rozsahy hodnot tak, že počátek a konec stupnic nezačíná v nule nebo v jiném obvyklém bodě, ale zobrazuje pouze pásmo obvyklých

hodnot, které jsou dosahovány v daných podmínkách.



Obr. 12 - Vizualizace aktuálních dat pomocí Google Gauge plugin

- 1 options = {width: 200, height: 200, redFrom: red\_value-0.1, redTo: red\_value+0.1, yellowFrom: yellow\_min, yellowTo: yellow\_max, min: 45, max: 55, minorTicks: 5};

### Nastavení parametrů grafu Google Gauge

Nastavení rozsahů stupnice je nastaveno na konstantní hodnoty. V úvaze byla i dynamická změna rozsahů stupnic pro případ, kdy se změřená hodnota pohybuje mimo rozsah stupnice. Po zhodnocení situace, kdy by chybná hodnota mohla značně znepráhlednit vizualizaci měření, bylo však od této možnosti upuštěno.

Červené pásmo na stupnici magnetické indukce je konstanta, označující hodnotu magnetické indukce pro dané souřadnice měření, dle NOAA World Magnetic Model. Uživatel tak může sledovat rozdíl od této hodnoty způsobený např. okolním rušením a dalšími chybami měření.

V konkrétním případě tohoto experimentu, je dle Obr. 12 rozdíl mezi měřenými hodnotami a hodnotami dle NOAA způsoben zejména provizorním umístěním v budově, kde je mnoho feromagnetických prvků a zdrojů magnetizace, čímž dochází ke zkreslení měření magnetického pole.

Žluté pásmo pak označuje rozsah mezi minimem a maximem z naměřených hodnot magnetické indukce a teploty během 24 hodinového úseku měření. Uživatel tak může sledovat rozdíl mezi aktuální, minimální a maximální naměřenou hodnotou za posledních 24 hodin.

Přesná hodnota v číslicové podobě je zobrazena ve spodní části, mezi počátkem a koncem stupnice.



## Hodnocení a závěr

Výstupem předkládané práce je funkční aplikace pro použitý magnetometr, která zajišťuje komunikaci s přístrojem, měření, zpracování, archivaci a elektronickou prezentaci dat na webu. V první části práce byly prověřeny možnosti komunikace s přístrojem, kde jsou popsány parametry a dostupná konfigurace čipu AMR magnetometru Honeywell HMC5843. Na tuto část práce bylo úspěšně navázáno při tvorbě obslužné aplikace pro práci s přístrojem.

Pomocí programovacího jazyka Python byla vytvořena sada skriptů pro navázání komunikace a inicializaci přístroje připojeného na sériový port, kalibraci a test správné funkce zařízení. Zde se podařilo provést efektivní nastavení parametrů konfigurace pro přístup k zařízení vytvořeným modulem, který lze aplikovat do všech skriptů, ve kterých je zapotřebí komunikace s přístrojem, čímž je zajištěna snadná modifikace těchto parametrů.

Dále vytvořený měřicí algoritmus se stará o měření a výpočet všech předem definovaných složek magnetického pole a tyto hodnoty jsou ukládány do databází pro další zpracování, archivaci a prezentaci dat. Nevýhodou je nutná transformace výsledků z hlediska aktuální polohy přístroje, která je prováděna přímo ve zdrojovém kódu, takže je nutné po změně polohy přístroje transformaci upravit. Zde je možný potenciál dalšího vylepšení použitím akcelerometru, kterým by se výsledky měření daly automaticky kompenzovat s aktuální polohou magnetometru, a tím rovněž zpřesnit měření.

Prezentace dat v elektronické podobě na webu splnila očekávané výsledky, a to zejména díky využití profesionálně navržené platformě ThingSpeak, která nabízí mnoho funkcionalit pro práci se senzory a akčními členy. Tuto platformu lze doporučit dalším zájemcům pro tvorbu aplikací na poli Internetu věcí.

Díky implementaci dat z teplotního čidla do souběžných grafů s měřením magnetického pole lze velice zřetelně sledovat, jak změna teploty ovlivňuje výsledky měření. Tímto sledováním lze odvodit podnět pro další práci, a to implementaci algoritmu pro korekci výsledků měření v závislosti na okolní teplotě.

Výstup práce umožňuje další zkoumání výsledků a rozšíření zkušeností s AMR senzory. Vhodným umístěním přístroje mimo zdroje rušivých vlivů a s dalšími vylepšeními, na které tato práce poukazuje, by dalším cílem mohlo být zapojení magnetometru do mezinárodní sítě měření magnetického pole Země.

## Seznam literatury a dalších informačních zdrojů

- [1] Australian Robotics. How to Talk To ThingSpeak With Python (A Memory/CPU Monitor). [online]. Chris, 2011. [Cit. 25.4.2015]. Dostupné z: <http://www.australianrobotics.com.au/news/how-to-talk-to-thingspeak-with-python-a-memory-cpu-monitor>
- [2] BOSTOCK Mike. D3.js - Data-Driven Documents. [online]. ©2013 [Cit. 20.5.2015]. Dostupné z: <http://d3js.org/>
- [3] DOWNEY Allen. Think Python. [online]. Green Tea Press, 2012. [Cit. 23.5.2015]. Dostupné z: <http://www.greenteapress.com/thinkpython/thinkpython.pdf>
- [4] Honeywell. Datasheet. [online]. ©2010 [Cit. 25.5.2015]. Dostupné z: [http://www51.honeywell.com/aero/common/documents/myaerospacatalog-documents/Defense\\_Brochures-documents/HMC5843.pdf](http://www51.honeywell.com/aero/common/documents/myaerospacatalog-documents/Defense_Brochures-documents/HMC5843.pdf)
- [5] Honeywell. Datasheet. [online]. ©2010 [Cit. 25.5.2015]. Dostupné z: [http://www51.honeywell.com/aero/common/documents/myaerospacatalog-documents/Defense\\_Brochures-documents/HMC5883L\\_3-Axis\\_Digital\\_Compass\\_IC.pdf](http://www51.honeywell.com/aero/common/documents/myaerospacatalog-documents/Defense_Brochures-documents/HMC5883L_3-Axis_Digital_Compass_IC.pdf)
- [6] Honeywell. Features. [online]. ©2010 [Cit. 25.5.2015]. Dostupné z: [forum.arduino.cc/index.php?action=dlattach;topic=55159.0;attach=2675](http://forum.arduino.cc/index.php?action=dlattach;topic=55159.0;attach=2675)
- [7] LIECHTI Chris. pySerial API — pySerial 2.7 documentation. [online]. ©2001-2013. [Cit. 23.5.2015]. Dostupné z: [http://pyserial.sourceforge.net/pyserial\\_api.html#](http://pyserial.sourceforge.net/pyserial_api.html#)
- [8] MOHAMADABADI, Kaveh. Anisotropic Magnetoresistance Magnetometer for inertial navigation systems. [online]. Ecole Polytechnique X, 2013. Dostupné z: <https://tel.archives-ouvertes.fr/file/index/docid/946970/filename/Thesis.pdf>
- [9] PHP Group. PHP: Introduction - Manual. [online]. ©2001-2015 [Cit. 18.5.2015]. Dostupné z: <http://php.net/manual/en/intro.image.php>
- [10] RIPKA, Pavel. Magnetic sensors and magnetometers. Massachusetts: Artech House, 2001. 494 s. ISBN 1-58053-057-5.

- [11] ThingSpeak. Internet Of Things - ThingSpeak. [online]. ©2015 [Cit. 26.5.2015]. Dostupné z: <https://thingspeak.com/>
- [12] ThingSpeak. Introduction to the “Internet of Things” and ThingSpeak. [online]. ©2014 [Cit. 16.5.2015]. Dostupné z: <http://community.thingspeak.com/tutorials/introduction-to-the-%E2%80%9Cinternet-of-things%E2%80%9D-and-thingspeak/>
- [13] ThingSpeak. Support - ThingSpeak. [online]. ©2015 [Cit. 16.5.2015]. Dostupné z: <https://thingspeak.com/docs>
- [14] US Department of Commerce, NOAA Satellite and Information Service, National Centers for Environmental Information. NOAA National Centers for Environmental Information (NCEI). [online]. ©2012 [Cit. 31.5.2015]. Dostupné z: <http://www.ngdc.noaa.gov/>

## **Přílohy**

### **Přiložený nosič DVD-R – obsah:**

mgme.zip – soubor měřících a ovládacích skriptů magnetometru

prezentace.zip – soubor zdrojových kódů pro webovou prezentaci

thingspeak.zip – soubor JavaScript kódů z platformy ThinkSpeak