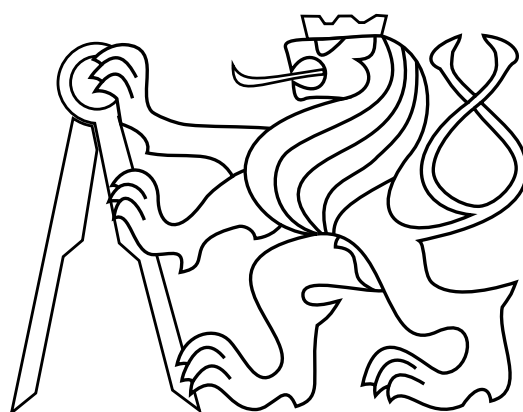


České vysoké učení technické v Praze

Fakulta elektrotechnická

Katedra elektromagnetického pole



Diplomová práce

Měřič magnetizace pro defektoskopii

Autor: Bc. Pavel Staněk

Vedoucí práce: Prof. Ing. Zbyněk Škvor, CSc.

květen 2017

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Staněk** Jméno: **Pavel** Osobní číslo: **406072**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávající katedra/ústav: **Katedra elektromagnetického pole**
Studijní program: **Komunikace, multimédia a elektronika**
Studijní obor: **Elektronika**

II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

Měřič magnetizace pro defektoskopii

Název diplomové práce anglicky:

Fluxmeter for Defectoscopy Applications

Pokyny pro vypracování:

1. Prostudujte požadavky kladené na měření dosažené magnetizace při defektoskopických zkouškách trubek magnetickou metodou práškovou.
2. Navrhněte a experimentálně ověřte měřič magnetizace vhodný ke kontrole funkce magnetizérů pro magnetickou metodu práškovou v defektoskopii železných kovů.
3. Měřič by měl měřit tangenciální složku intenzity magnetického pole těsně nad povrchem kovu v rozsahu alespoň 500 A/m až 5000 A/m, a to ve dvou vzájemně kolmých směrech. Zároveň musí měřič vyhodnotit nejen velikost, ale i vzájemný fázový posun dílčích magnetizací v obou směrech, a to pro frekvence do 100 Hz.
4. Přenosný měřič má být řízen vhodným procesorem řady STM32 a ovládán dotykovým displejem.
5. Zhodnotte dosažené výsledky.

Seznam doporučené literatury:

- [1] Punčochář, J.: Operační zesilovače v elektronice, Ben, Praha 2002
- [2] Yiu, J.: The definitive guide to ARM CORTEX M-3 and CORTEX M-4 processors, Elsevier, Oxford 2014
- [3] Sadiku, M.N.O.: Elements of Electromagnetics. Saunders College Publishing. London, 1994

Jméno a pracoviště vedoucí(ho) diplomové práce:

prof. Ing. Zbyněk Škvor CSc., katedra elektromagnetického pole FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

Datum zadání diplomové práce: **16.02.2017**

Termín odevzdání diplomové práce: _____

Platnost zadání diplomové práce: _____

Podpis vedoucí(ho) práce

Podpis vedoucí(ho) ústavu/katedry

Podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Diplomant bere na vědomí, že je povinen vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v diplomové práci.

Datum převzetí zadání

Podpis studenta

Abstrakt

Tato práce se zabývá problematikou návrhu měřiče magnetizace pro magnetickou metodu nedestruktivního testování materiálů. První část obsahuje teoretický popis magnetické práškové metody a z ní vyplývající nároky na měření magnetického pole. Druhá část je zaměřena na teoretický rozbor návrhu a jsou zde popsány jednotlivé bloky měřiče. V praktické části je realizován prototyp měřiče a jsou zde uvedeny výsledky měření na magnetizačním zařízení. Práce také obsahuje řídicí program pro procesor řady STM32, kterým je měřič ovládán.

Abstract

This thesis deals with the design of fluxmeter for magnetic non-destructive defectoscopy applications. The theoretical part contains a description of magnetic particle testing as well as magnetic field properties to be measured. The second part covers description of potential solutions, its benefits and drawbacks. In the third practical part an experimental vector fluxmeter is designed and tested. The fluxmeter is driven by mikroprocessor STM32 and cotrolled via a touchscreen. Last chapter also contains first measured results.

Poděkování

Rád bych zde poděkoval prof. Ing. Zbyňku Škvorovi, CSc., vedoucímu práce, za ochotu, trpělivost a cenné odborné rady.

Prohlášení

Prohlašuji, že jsem diplomovou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v příloženém seznamu v souladu s metodickým pokynem O dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze 26. května 2017

.....

Obsah

Úvod	2
1 Magnetická metoda prášková	4
1.1 Princip magnetické metody práškové	4
1.2 Podmínky úspěšné detekce defektů	5
1.3 Magnetizační techniky	6
2 Návrh měřiče magnetizace	8
2.1 Blokové schéma	8
2.2 Sensory magnetického pole	9
2.2.1 Sensory GMR	10
2.2.2 Cívka jako senzor	10
2.2.3 Sensory s Hallovým jevem	11
2.2.4 3D sensory s Hallovým jevem	11
2.2.5 3D integrované sensory s digitálním zpracováním signálu	12
2.3 Zesilovač	12
2.4 Procesor	15
2.5 Napájení	16
2.6 Software	16
2.6.1 Měření	17
2.6.2 Zobrazení bodů a časového průběhu	17
2.6.3 Aproximace elipsou	18
2.6.4 Uložení dat	19
3 Realizace a měření parametrů	21
3.1 Uchycení sond	21
3.2 Realizace zesilovače	22
3.3 Kalibrace	23
3.4 Mechanické provedení přístroje	24
3.5 Výsledky měření pole magnetizéru	24
Závěr	28

Úvod

Jednou z možných metod nedestruktivního testování je magnetická metoda prášková. Tato metoda je založená na vložení zkoušeného dílu do magnetického pole a následné detekci rozptylového pole způsobeného defekty. Vyhodnocení probíhá vizuálně za pomoci feromagnetického prášku, který se koncentruje v místě vad. Pro úspěšnou detekci je nutné, aby byl vektor intenzity magnetického pole ideálně kolmý k této vadě. Druhým požadavkem je dostatečná intenzita. Při konstrukci a testování magnetizérů pro tuto metodu je nutné měřit časový průběh vektoru magnetického pole pro zajištění dostatečné intenzity ve všech směrech. Dostupné způsoby ověření magnetického pole jsou standardizované měřky či gaussmetry. Měřky jsou schopné detekovat směr magnetického pole a ukázat, zda je intenzita dostatečná či není. Běžně prodávané gaussmetry s vyhovujícím měřicím rozsahem jsou schopné měřit jednotlivé složky magnetického pole, ale nedisponují možností zobrazení časového průběhu a polarizace pole. Tato práce si klade za cíl navrhnout a realizovat zařízení schopné vektorově měřit magnetické pole s možností zobrazení polarizace, časových průběhů a fázového posunu jednotlivých složek.

Navrhovaný měřič by měl být schopen měření intenzity magnetického pole v rozsahu, který pokrývá intenzity pole běžně používaného při magnetické metodě nedestruktivního testování. Minimální intenzita potřebná k úspěšnému testu je zhruba 2000 Am^{-1} těsně nad povrchem materiálu. Požadovaný rozsah byl tedy stanoven od 500 Am^{-1} do 5000 Am^{-1} . Dalším požadavkem je měření této intenzity ve dvou na sebe kolmých směrech. Vyhodnocována by měla být nejenom amplituda intenzity, ale i vzájemný fázový posun polí v obou směrech pro frekvence do 100 Hz. Měřič má být řízen vhodným procesorem řady STM 32 a ovládán dotykovým displejem.

Tato práce je členěna do tří hlavních kapitol. První kapitola se věnuje popisu magnetické metody práškové pro nedestruktivní testování. Jsou zde diskutovány požadavky na průběhy magnetického pole pro úspěšné provedení zkoušky a z nich vyplývající parametry měřiče. V druhé kapitole je uveden teoretický rozbor možných řešení a argumenty vedoucí k volbě jednoho z nich. Ve třetí kapitole je popsán postup realizace, dále jsou zde uvedeny a diskutovány výsledky měření na magnetizačním

zařízení.

V práci je dále používána terminologie běžná v oboru nedestruktivního testování. Pojem magnetizace, který se v teorii elektromagnetického pole obvykle definuje jako $M = \frac{1}{\mu_0}B - H$, je zde používán v odlišném významu, pokud to není explicitně uvedeno. Na následujících stranách je magnetizací označován průběh magnetické intenzity na povrchu testovaného předmětu, který je stěžejní pro hledání materiálových vad. [13]

1 Magnetická metoda prášková

Magnetická metoda prášková je jednou z metod nedestruktivního zkoušení materiálů. Využívá se v různých odvětvích průmyslu při závěrečné fázi výroby. Uplatnění nalézá v automobilovém a leteckém průmyslu, stejně tak jako v energetice či dopravě. Touto metodou je možné bez poškození dílu či výrobku zjišťovat povrchové a podpovrchové vady jako jsou trhliny, praskliny či póry feromagnetických materiálů.

1.1 Princip magnetické metody práškové

V průběhu zkoušky je testovaný předmět vhodným způsobem zmagnetován a na povrch je nanesen feromagnetický prášek. Potenciální vada má jiné magnetické vlastnosti než testovaný materiál, čímž dochází k deformaci pole a jeho vystoupení z materiálu. Vytvoří se takzvaný rozptylový magnetický tok, který nese informaci o místě defektu. Částičky feromagnetického prášku jsou tímto tokem přitahovány a seskupí se v místě vady. Vyhodnocení je provedeno na základě barevného vizuálního vjemu nebo jasového kontrastu obvykle při osvětlení dílu UV lampou. Obrázek 1.1 ukazuje příklad indikace defektu.



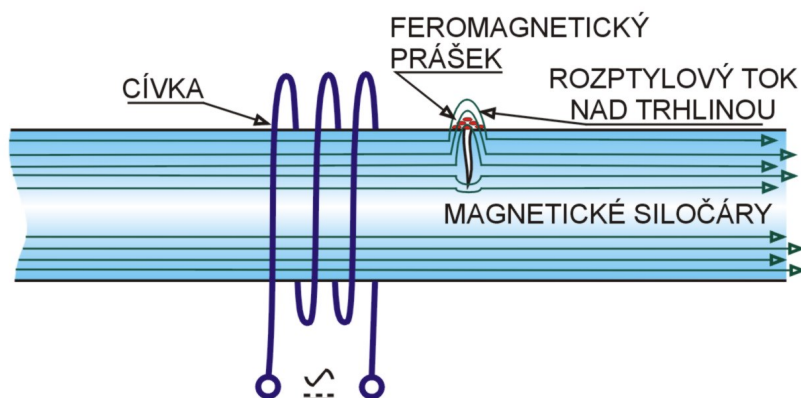
Obrázek 1.1: Indikace defektu magnetickou metodou práškovou[1]

Tuto metodu lze použít pro detekci vad feromagnetických materiálů. Největší citlivost vykazuje pro povrchové vady. Lze ji však aplikovat i k detekci objemových

podpovrchových vad. Citlivost však výrazně klesá směrem od povrchu materiálu. Minimální rozměr vady detekovatelný touto metodou začíná na tisícinách milimetru. [2]

1.2 Podmínky úspěšné detekce defektů

Princip magnetické metody testování je založen na řádově odlišných hodnotách permeability feromagnetických materiálů a hledaných vad. Běžné oceli dosahují $\mu_r \cong 100$ některé dokonce $\mu_r \cong 100\,000$. Permeabilita defektu zaplněného vzduchem je $\mu_r \cong 1$. Podmínkou vzniku výše zmíněného rozptylového toku je magnetické nasycení testovaného materiálu, to znamená, že všechny magnetické domény jsou souhlasně orientovány. V případě výskytu vady vystupují siločáry na povrch materiálu do vzduchu. Siločáry mají ve vzduchu menší hustotu než v testovaném materiálu a dojde k vytvoření magnetických pólů na hranách trhliny. Princip metody je zachycen na obrázku 1.2.



Obrázek 1.2: Princip magnetické metody práškové [2]

Podmínkou úspěšného odhalení defektů je tedy magnetické pole dostatečné intenzity ideálně kolmé k defektu. Experimentálně lze ukázat, že k určité indikaci dochází i v případě, že trhlina v materiálu a vektor intenzity svírají úhel přibližně 45° . Pro nalezení defektů ve všech směrech je tedy nutné vytvořit pole s rotujícím vektorem intenzity rovnoběžně s povrchem zkoušeného dílu. Ideálně by měl vektor intenzity rotovat po kružnici s konstantní úhlovou rychlostí. Této polarizace lze dosáhnout v případě použití dvou navzájem kolmých magnetizací s harmonickými průběhy se stejnou frekvencí. Jako minimální dostatečná intenzita pro úspěšné provedení zkoušky se uvádí přibližně 2 kAm^{-1} tangenciální složky pole na povrchu testovaného

dílu.

Při návrhu magnetizačních zařízení je nutné směr a velikost intenzity magnetického pole ověřovat, aby bylo dosaženo dostatečné magnetizace ve všech směrech a na celém povrchu dílu. Existuje několik možností, jak ověřit, zda je průběh magnetického pole vhodný k detekci defektů. Lze použít standardizované měrky (např. Betholdova měrka) či gaussmetr. Měrky jsou schopné detekovat pouze směr pole a vyhodnotit, zda je intenzita dostatečná pro provedení zkoušky. Dostupné gaussmetry s vyhovujícím rozsahem jsou schopné měřit intenzitu magnetického pole v jednom či více směrech, ale nedisponují funkcí zobrazení časového průběhu či polarizace pole. [12]

1.3 Magnetizační techniky

Magnetizaci lze provádět dvěma způsoby: přímo a nepřímo. Přímá magnetizace je založena na průchodu elektrického proudu testovaným materiálem. Nepřímá magnetizace využívá vnějších zdrojů magnetického pole.

Přímá magnetizace se provádí hrotovými elektrodami, které se přitisknou na zkoušený předmět. Proud protékající elektrodami se uzavírá přes zkoušený díl. Kolem elektrod vzniká kruhové magnetické pole. Detekovatelné vady leží ve směru spojnice elektrod, proto je potřeba test opakovat s otočením elektrod o 90° , aby byly detekovány vady ve všech směrech. Pro otestování celého povrchu je nutné překrývání zkoušených oblastí.

Magnetizační zařízení pro nepřímou metodu lze rozdělit do dvou skupin: zařízení s permanentními magnety a s elektromagnety. Permanentní magnety na rozdíl od elektromagnetů nevyžadují zdroj energie, ale nelze měnit intenzitu jejich magnetického pole, která může navíc s časem slábnout. Nemožnost měnit intenzitu je podstatnou nevýhodou při procesu demagnetizace, kde je žádoucí ji plynule snižovat. Demagnetizace je nutnou součástí procesu testování, jelikož zbytkový magnetismus může způsobovat problémy v průběhu následného sváření či obrábění materiálu.

V praxi se častěji používají zařízení s elektromagnety zejména kvůli možnosti regulace intenzity magnetického pole. Celkovou magnetizaci malých dílů lze provádět například jejich vložením do cívky. Dalším běžně používaným magnetizačním zařízením je elektromagnetické jho. Jedná se o magnetický obvod ve tvaru podkovy buzený cívkou. V praxi výhodnou konfigurací je křížové jho. Skládá se ze čtveřice cívek nasazených na společném jádře. V případě, že je tato čtveřice cívek napájena proudy s vhodnou amplitudou a fází, lze v prostoru mezi nimi vytvořit magnetické pole s kruhovou polarizací. Tím je zajištěna detekce defektů ve všech směrech bez nutnosti otáčení magnetizačního zařízení. Obrázek 1.3 ukazuje obě zmíněná jha.



Obrázek 1.3: Elektromagnetické jho a křížové elektromagnetické jho [2] [3]

Magnetizaci za pomoci elektromagnetů je možné provádět různými časovými průběhy proudu. Stejnoseměrný proud se obvykle používá u ručních magnetizérů. Výhodou je absence skin efektu a tím pádem rovnoměrnější zmagnetování v celém objemu testovaného dílu a možnost detekce podpovrchových vad ve větší hloubce. Při použití střídavého proudu obvykle na síťové frekvenci 50 Hz se již začíná projevovat skin efekt a magnetizace proběhne pouze v povrchové vrstvě materiálu. Výhodou je dobrá zjištělnost povrchových vad a snadné odsmagnetování. Dále lze také použít dvoucestně usměrněný jedno či vícefázový proud. Existují i zařízení využívající pulzní proudy.

V případě nutnosti odsmagnetování se používají odsmagnetovací tunely, stažení jha napájeného střídavým proudem či oddalování jha napájeného stejnosměrným proudem při stálém otáčení o 180° . [12]

2 Návrh měřiče magnetizace

Na trhu existuje řada běžně dostupných měřičů s různými rozsahy parametrů. Důležitými parametry jsou zejména: měřicí rozsah, frekvenční rozsah, přesnost, způsob vyhodnocení a zobrazení magnetického pole a počet navzájem kolmých směrů měření. Vůči těmto existujícím měřicím přístrojům je nutné se vymežit a zdůvodnit vznik návrhu v rámci této práce.

výrobce	označení	rozsah (mT)	frekvence (Hz)	osy (-)
GMW	THM1176	20 000	DC–1 000	3
F.W.BELL	5180	3 000	DC–20 000	1
Hirst Magnetics	GM07	3 000	15–10 000	1
MAGSYS	HGM09s	4 500	DC–5 000	1

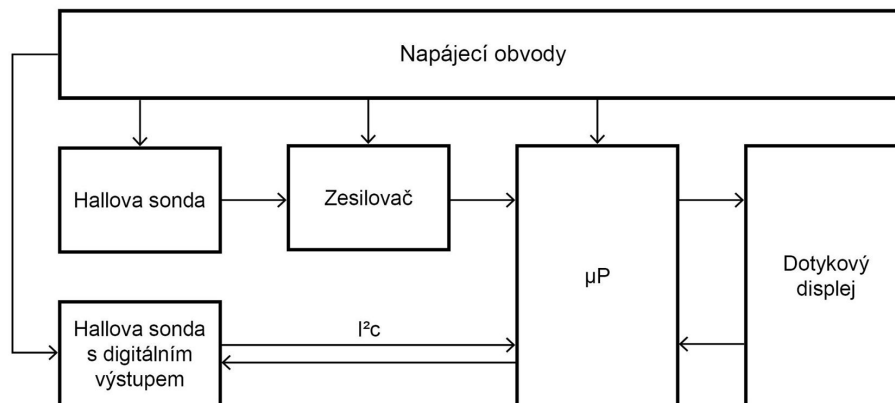
Tabulka 2.1: Porovnání dostupných měřičů magnetizace

Tabulka 2.1 ukazuje srovnání parametrů existujících přístrojů. Tyto přístroje vyhovují svými rozsahy zadání, nicméně nejsou schopny zobrazit časový průběh a polarizaci měřeného pole. Unikátnost návrhu tedy spočívá zejména v možnosti dvouosého vektorového měření na frekvencích až do 100 Hz a následné možnosti zobrazení polarizační elipsy, časových průběhů a fázového posunu mezi jednotlivými směry.

2.1 Blokové schéma

Obrázek 2.1 ukazuje blokové schéma zapojení měřiče. Je zde počítáno s možností připojení dvou senzorů magnetického pole s různými typy výstupů. První použitý senzor disponuje napěťovým výstupem úměrným magnetickému poli. Zde je nutné zařadit navazující blok, který se stará o zesílení signálu a přesunu stejnosměrné složky tak, aby byl využit celý rozsah AD převodníků procesoru. Druhým typem připojovaných senzorů, jsou sondy obsahující digitální obvody pro zpracování signálu komunikující přímo s procesorem po datové sběrnici I²C. Dalším blokem je mikroprocesor řady STM32, který se stará o zpracování signálů a o jejich zobrazení

na dotykovém displeji, který zároveň slouží k ovládání celého zařízení. Technické řešení jednotlivých bloků je diskutováno v následujících podkapitolách.



Obrázek 2.1: Blokové schéma zapojení

2.2 Senzory magnetického pole

Senzor magnetického pole je stěžejní komponentou zařízení a je potřeba ho volit tak, aby svými parametry splňoval požadavky na měřené pole. Zejména se jedná o vhodný rozsah, citlivost, linearitu a dostatečně vysokou vzorkovací frekvenci.

Při požadovaném měřicím rozsahu intenzity magnetického pole od 500 Am^{-1} do 5000 Am^{-1} dostáváme dle vztahu

$$B = \mu_0 \mu_r H$$

ve vzduchu indukci v rozsahu od $0,628 \text{ mT}$ do $6,28 \text{ mT}$. Je tedy nutné volit senzor, který splňuje minimálně tento rozsah. Vzhledem k tomu, že má být měřeno nízkofrekvenční magnetické pole o frekvenci maximálně 100 Hz a požadavku zobrazení jeho časového průběhu, je nutné dle Shannon-Nyquistova teorému vzorkovat na frekvenci alespoň 200 Hz , což odpovídá vzorkovací periodě maximálně 5 ms . Ze zadání dále plyne požadavek na měření ve dvou navzájem kolmých směrech, proto je nutné zohlednit možnosti uchycení senzorů v navzájem kolmé poloze. Možným řešením je také použití víceosých senzorů integrovaných do jednoho pouzdra, které umožňují

měření ve všech třech na sebe kolmých směrech.

2.2.1 Senzory GMR

Tento typ senzorů je založen na jevu zvaném *Giant Magnetoresistance*. Jedná se o změnu velikosti elektrického odporu feromagnetických slitin v závislosti na působícím magnetickém poli. Konkrétní hodnoty změny elektrického odporu se pohybují v rozmezí 10 % až 50 % od hodnoty odporu při nulovém magnetickém poli. Po dosažení maximální změny dojde k saturaci a hodnota se již příliš nemění. [4]

Senzory na tomto principu vyrábí například firma *NVE Corporation*. Maximální dostupná hodnota lineárního rozsahu je 0,7 mT. Vzhledem k výše uvedeným hodnotám tyto senzory nevyhovují zadání. Jsou vhodné pro měření polí s menší intenzitou.

2.2.2 Cívka jako senzor

Pro detekci změny magnetického toku lze využít cívku. Můžeme vyjít z Faradayova indukčního zákona

$$u_i = \frac{d\Phi_c}{dt},$$

kde u_i je indukované napětí na vývodech cívky a Φ_c cívkový tok. Po integraci dostáváme

$$\int_{t_1}^{t_2} u_i dt = \int_{\Phi_{c1}}^{\Phi_{c2}} d\Phi_c = \Delta\Phi_c.$$

V případě, že předpokládáme homogenní magnetické pole v cívce, pak

$$\Delta B = \frac{\Delta\Phi_c}{SN} = \frac{1}{SN} \int_{t_1}^{t_2} u_i dt,$$

kde B je složka magnetické indukce kolmá k ploše cívky S s N závity. Uvedený integrální vztah lze realizovat např. elektronickým integrátorem. [10]

Pro praktickou realizaci by bylo nutné zakoupit, popřípadě navinout dvě cívky, co nejvíce shodné a zajistit jejich kolmé upevnění. Tato varianta by byla zatížena chybou výrobního procesu a následnou realizací kolmého upevnění. Pravděpodobně by bylo také nutné použití cívek s více závity, což je nežádoucí, z hlediska větších fyzických rozměrů, a tím pádem odchýlení od měření ideálně všech kolmých směrů pole v jednom bodě.

2.2.3 Senzory s Hallovým jevem

Další skupinou senzorů jsou senzory založené na principu Hallova jevu. V případě, že polovodičovou destičkou protéká proud a současně kolmo působí magnetické pole s indukcí B , začne na pohybující se nosiče náboje působit síla ve směru kolmém k indukci. Náboj se hromadí na stranách destičky a vytváří tzv. Hallovo napětí, které můžeme popsat vztahem

$$u_H = R_H i \frac{B}{d},$$

kde R_H je Hallova konstanta použitého materiálu a d je tloušťka destičky ve směru indukce B . Tento typ senzorů umožňuje měření indukce až do jednotek T, tedy v rozsahu zadání.[10]

Senzorů využívajících tento jev je dostupná celá řada od různých výrobců. Nejjednodušší integrované obvody mají obvykle tři vývody: napájecí napětí, zem a výstupní napětí. Výstupní napětí je při nulové magnetické indukci B stabilizováno na polovině napětí napájecího. V případě nenulové indukce B dojde ke změně výstupního napětí. Nárůst či pokles napětí je dán směrem magnetického pole. Většina senzorů tohoto typu je ve velkém rozsahu lineární s definovaným rozmezím citlivosti, která se může měnit s velikostí napájecího napětí. Obvykle je v rámci senzoru integrován i filtr typu dolní propust se zlomovým kmitočtem kolem 1 kHz. Tabulka 2.2 nabízí porovnání dostupných senzorů tohoto typu. Vzhledem k dostupnosti byl volen senzor od *Allegro Microsystems* A1302.

Tento senzor lze napájet v rozmezí 4,5–6 V. Citlivost je dle katalogového listu rostoucí lineární funkcí napájecího napětí a pro uvedený napájecí rozsah se pohybuje v rozmezí 1,2–1,6 mVG⁻¹. Výstupní napětí je také mírně teplotně závislé.

výrobce	označení	citlivost (mVG ⁻¹)	rozsah (G)
Allegro MicroSystems	A1302	1,3	-
Allegro MicroSystems	A1302	2,5	-
Honeywell	SS495A	3,125	±670
Honeywell	SS496B	2,5	±840

Tabulka 2.2: Porovnání dostupných senzorů s Hallovým jevem

2.2.4 3D senzory s Hallovým jevem

Zajímavou variantou je také senzor HE444 od firmy *Advanced Sensor Technology*, který integruje tři navzájem kolmé snímací elementy. Dle katalogového listu disponuje

lineárním výstupem až do hodnoty 1 T a umožňuje měření s přesností na 6 cifer. Výše uvedené parametry jsou podmíněné stabilitou vnějšího zdroje proudu, který senzor pro svou činnost vyžaduje. Realizace měřícího zařízení s tímto senzorem by vyžadovala relativně složitý návrh analogových napájecích obvodů, aby mohlo být dosaženo uvedené přesnosti.

2.2.5 3D integrované senzory s digitálním zpracováním signálu

Tato skupina senzorů obsahuje tři navzájem kolmé snímací elementy s Hallovým jevem uvnitř jednoho integrovaného obvodu. Dále disponují obvody pro zpracování signálu, zejména AD převodníky a teplotním čidlem. Komunikují pomocí rozhraní SPI či I²C.

výrobce	označení	rozsah (mT)	rozhraní	vzorkovací frek. (Hz)
Infineon	TLV493D-A1B6	±130	I ² C	33 000
NXP	MAG3110	±1	I ² C	80
Melexis	MLX90393	-	I ² C, SPI	790

Tabulka 2.3: Porovnání dostupných 3D senzorů

Výhodou této skupiny senzorů je kolmé sesazení snímacích elementů výrobcem a jejich umístění do maximální blízkosti. Lze tedy dosáhnout dobrého přiblížení k měření složek pole v jednom bodě. Nevýhodou je neznalost přesné struktury vnitřních obvodů a jejich časových parametrů. Senzory uvedené v tabulce 2.3 disponují pouze jedním AD převodníkem a získávání hodnot ve směru jednotlivých os je řešeno časovým multiplexem. Nelze tedy odebírat vzorky ve směru všech os najednou, ale pouze s určitým časovým odstupem. Vzorkovací frekvence a rychlost AD převodu se mění podle stav vnitřních registrů senzoru. Z těchto senzorů byl pro měření vybrán TLV493D-A1B6, jelikož dosahuje výrazně větší vzorkovací frekvence než ostatní senzory.

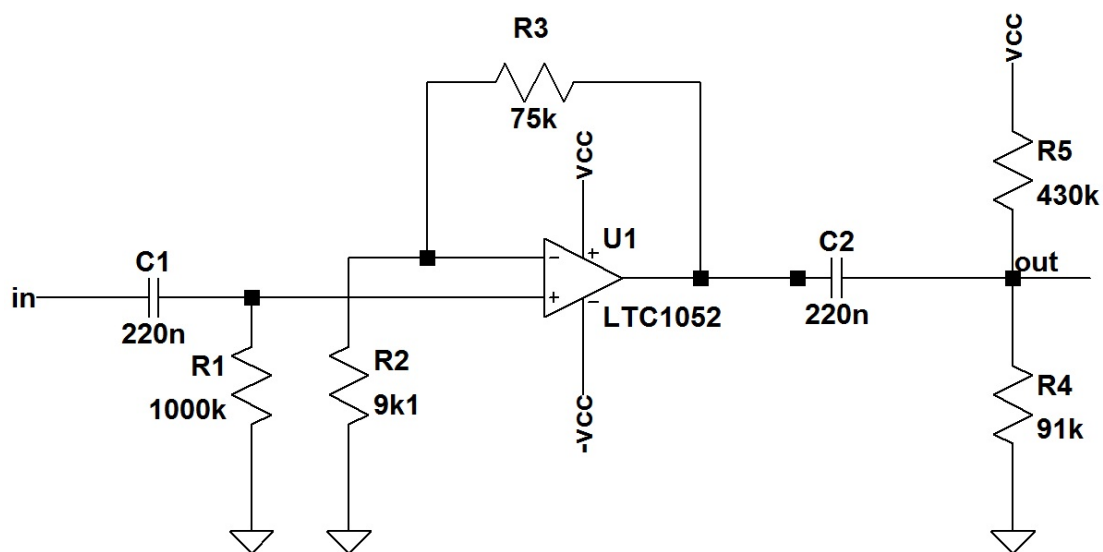
2.3 Zesilovač

Blok zesilovače zajišťuje zesílení napěťového výstupu z Hallovy sondy a přesun stejnosměrné složky. Jak již bylo zmíněno, napětí na výstupu sondy je rovno polovině napájecího napětí v případě nulového magnetického pole, tj. 2,5 V při napájení 5 V. Při citlivosti 1,3 mVG⁻¹ pro zadaný měřící rozsah dostáváme napěťovou změnu 2500 ± 82 mV. Vstupní rozsah AD převodníků procesoru STM32 je 0–1,8 V při použití vnitřního referenčního napětí. Je tedy vhodné signál zesílit tak, aby byl využit celý

rozsah převodníků a aby napětí na výstupu zesilovače bylo 0,9 V při vstupním napětí 2,5 V.

Principiálně se nabízí dvě varianty řešení. První varianta je založená na rozdílovém zesilovači a zachování možnosti měřit stacionární magnetické pole. Druhou možností je použití jednoduššího zapojení neinvertujícího zesilovače a oddělení vstupu a výstupu kondenzátory.

První varianta by spočívala v použití diferenčního zesilovače s OZ a zesílení napěťového rozdílu mezi výstupem sondy a referenčním napětím 2,5 V. Pro dosažení větší přesnosti by bylo možné použít složitější zapojení tzv. přístrojového zesilovače. Toto zapojení se skládá ze tří OZ a disponuje velkým vstupním odporem a nízkou napěťovou nesymetrií. Dále by bylo nutné provést přesun střední hodnoty sečtením zesíleného rozdílu a referenčního napětí 0,9 V. Bylo by tedy potřeba čtyř OZ na jeden zesilovací kanál a nutnost vytvořit dvě referenční napětí. Nevýhodou je nepřesné dělení napájecího napětí sondami při nulovém magnetickém poli a jeho závislost na napájecím napětí a teplotě. Obtížné by také bylo vytvoření přesných referenčních napětí. Výhodou je možnost měření stacionárního magnetického pole. Pro přesné měření by bylo nutné toto zapojení trimovat pomocí potenciometrů, proto byla pro realizaci zvolena druhá možnost.

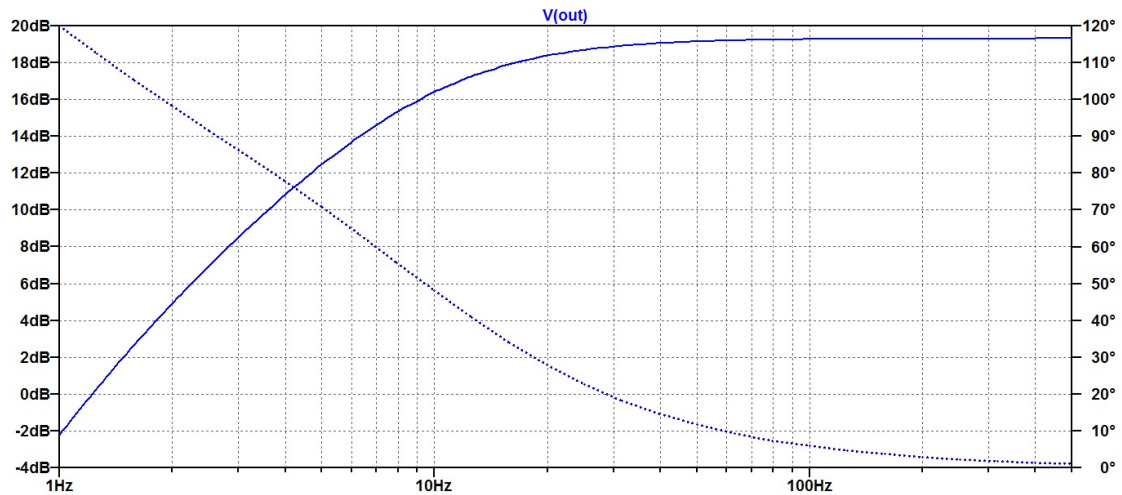


Obrázek 2.2: Schéma zapojení zesilovače

Schéma použitého zapojení ukazuje obrázek 2.2. Zesílení je realizováno neinvertujícím zesilovačem. Stejnoseměrné oddělení zajišťují CR články, které tvoří filtr typu horní

propust se zlomovým kmitočtem

$$f_c = \frac{1}{2\pi RC}. \quad (2.1)$$



Obrázek 2.3: AC analýza zesilovače

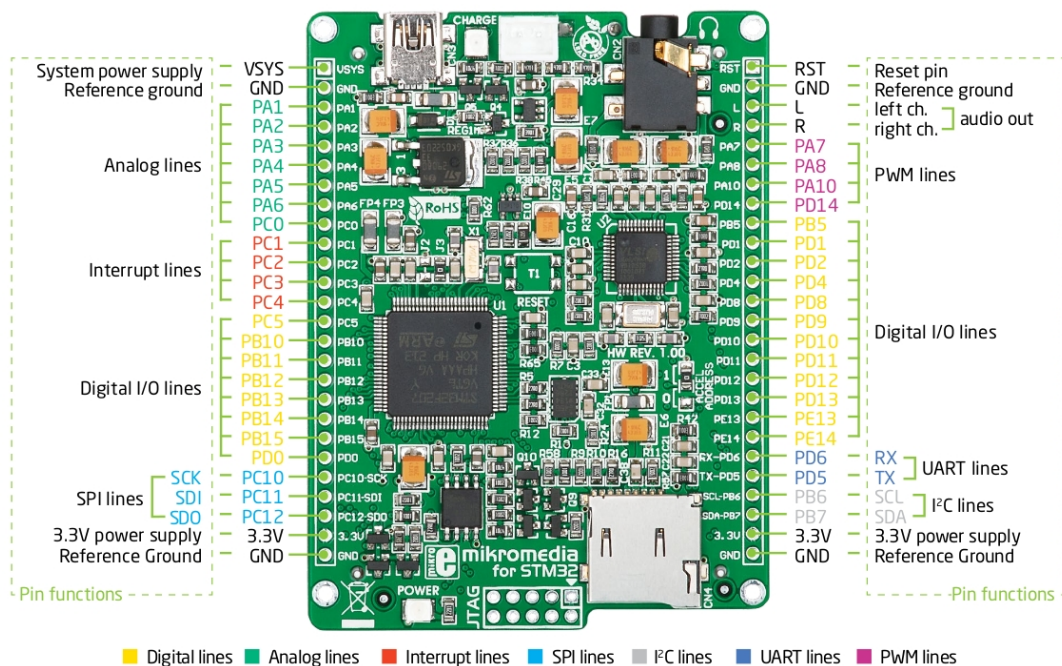
Kapacita kondenzátorů C1 a C2 byla volena jako největší běžně dostupná kapacita keramických kondenzátorů, tj. 220 nF. Při AC analýze se rezistory R4 a R5 jeví jako paralelně zapojené na zem. Při volbě zlomového kmitočtu $f_c = 10$ Hz vychází dle vztahu 2.1 odpor paralelního zapojení rezistorů R4 a R5 na 72 k Ω . Tyto hodnoty odporu je potřeba určit tak, aby na výstupu zesilovače bylo stejnosměrné napětí 0,9 V při nulovém rozkmitu na vstupu. Vstupní CR člunek je tvořen kondenzátorem C1 a paralelním zapojením rezistoru R1 a vstupního odporu OZ. Rezistor R1 je zařazen do zapojení, aby bylo možné rychlejší vybití kondenzátoru C1. Obvyklý vstupní odpor OZ je větší než 1 M Ω , proto bude mezní kmitočet vstupního CR člunku menší než 10 Hz, a proto není nutné se jím dále zabývat. Zisk zesilovače je nastaven poměrem odporů R3 a R2 dle vztahu

$$G = 1 + \frac{R_3}{R_2} = 1 + \frac{75\,000}{9\,100} = 9,2. \quad (2.2)$$

Tím je docíleno využití celého rozsahu AD převodníků. Obrázek 2.3 ukazuje AC analýzu zapojení s OZ LTC1052 provedenou v simulátoru *LT Spice*. Zesilovač vykazuje charakter horní propusti a zesílení na 50 Hz je přibližně 19,1 dB. Výhodou tohoto řešení je přítomnost pouze jednoho referenčního napětí a použití pouze jednoho OZ na každý kanál. Nevýhodou je nemožnost měření stacionárních magnetických polí. [14]

2.4 Procesor

Pro řízení měřiče byl volen procesor STM32F407VG na 32-bitové architektuře *ARM Cortex M4*. Maximální hodinový kmitočet tohoto procesoru je 168 MHz. Disponuje třemi AD převodníky s dvacetičtyřmi přepínatelnými vstupy, sedmnácti časovači a umožňuje komunikaci pomocí rozhraní USB, SPI, CAN a I²C.



Obrázek 2.4: Deska *mikromedia for STM32* [5]

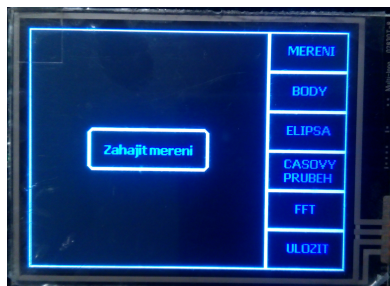
Měřič byl realizován na kompaktní vývojové desce *mikromedia for STM32* od firmy *MikroElektronika*. Tato deska obsahuje výše uvedený procesor, dotykový displej s rozlišením 240 x 320 a řadu dalších funkcí a komunikačních rozhraní. Desku lze propojit s PC přes USB rozhraní, které zajišťuje napájení a umožňuje nahrání softwaru do procesoru. Dále je zde k dispozici slot na SD kartu, akcelerometr, 8 Mbit flash paměť, stereo mp3 dekodér, konektor pro napájení z baterie a její možnost dobíjení přes USB a 52 I/O pinů připojených na vstupy a výstupy procesoru. Deska s procesorem je na obrázku 2.4.

2.5 Napájení

Jednotlivé části měřiče vyžadují ke své činnosti různá napájecí napětí. Zařízení je koncipováno jako přenosné, proto je nutné ho napájet bateriově. Hallovy senzory s napěťovým výstupem A1302 vyžadují napájecí napětí v rozmezí 4,5–6 V, senzor TLV493D komunikující po sběrnici I²C 2,8–3,5 V, desku s procesorem lze napájet 5 V či 3,3 V a zesilovače s OZ vyžadují symetrické napájení alespoň ± 5 V. Napájení bylo řešeno z dvou 9 V baterií pro vytvoření symetrického napájecího napětí. Úprava na ± 5 V je provedena pomocí lineárních regulátorů 7805 a 79L05. Deska s procesorem obsahuje lineární regulátor snižující napětí na 3,3 V pro napájení procesoru. Toto napětí je použito k napájení sondy TLV493D.

2.6 Software

Po obslužném programu se požaduje zpracování signálů ze sond, jejich zobrazení a ovládání měřiče pomocí dotykového displeje. Vývoj programu probíhal v prostředí *mikroC* od firmy *MikroElektronika* v jazyce C a jeho ladění na vývojové desce *EasyMx PRO v7 for STM32*. Program realizuje následující funkce: měření, zobrazení bodů, zobrazení časového průběhu, aproximace naměřených dat elipsou a uložení dat na SD kartu.



Obrázek 2.5: Úvodní obrazovka měřiče

Dotykový displej o rozlišení 240 x 320 je rozdělen na čtvercovou část 240 x 240, která je vyhrazena pro zobrazení výsledků a zbývající obdélníkovou část, ve které jsou zobrazena tlačítka sloužící k obsluze. Po zapnutí měřiče se zobrazí úvodní obrazovka, viz obrázek 2.5. Poté je nutné zahájit měření. Procesor si odebere předem stanovený počet vzorků z AD převodníků a zobrazí naměřené hodnoty formou bodů. Poté lze měnit způsoby zobrazení dat, zvolit jinou dostupnou operaci nebo zahájit nové měření. Kompletní kód programu je uveden v příloze.

2.6.1 Měření

Měření probíhá odebráním 256 vzorků ze signálu dvou Halloových sond na sebe kolmých směrech, tak aby byla pokryta alespoň jedna perioda signálu o frekvenci 100 Hz. Vzorkování signálu zajišťuje následující kód:

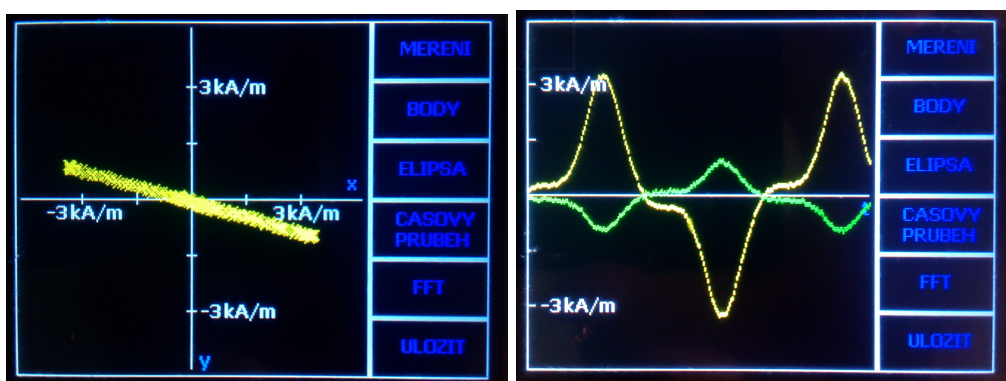
```
ADC_Set_Input_Channel(_ADC_CHANNEL_3 | _ADC_CHANNEL_4);
ADC1_Init(); ADC2_Init();

for(i = 0; i < 256; i++){
    xSamples[i] = ADC1_Get_Sample(3);
    ySamples[i] = ADC2_Get_Sample(4);
    Delay_us(10);
}
```

Jsou zde použité knihovny obsažené ve výše zmíněném vývojovém prostředí. Pro inicializaci převodníků a pro odebrání jednotlivých vzorků. Výstupy ze zesilovačů jsou přivedeny na analogové vstupy procesoru číslo tři a čtyři. Vzorkování z jednotlivých vstupů probíhá sériově. V ideálním případě by mělo probíhat vzorkování obou kanálů současně, ale vzhledem k měření polí do 100 Hz a rychlosti převodu včetně přiřazení hodnoty do příslušné proměnné v řádech desítek μs vzniká pouze malá časová odchylka. Pro dodržení vzorkovacího teorému je potřeba odebírat vzorky s periodou maximálně 5 ms. Zpoždění mezi odběry dvou vzorků z různých kanálů je o dva řády menší. Tento přístup by byl problematický pouze případě velkého zastoupení vyšších harmonických v signálu, kdy by se zpoždění odebrání vzorků blížilo k vzorkovací periodě. Kmitočet vyšších harmonických složek je limitována integrovaným výstupním filtrem sondy typu dolní propust na kmitočtu 1 kHz.

2.6.2 Zobrazení bodů a časového průběhu

Po provedení měření jsou hodnoty automaticky zobrazeny formou bodů do souřadnicového systému. Hodnoty ve směru os x a y reprezentují intenzitu pole ve dvou kolmých směrech. Bod v počátku souřadnicového systému reprezentuje stav s nulovými hodnotami magnetického pole v obou měřených směrech. Další možností zobrazení výsledků je průběh v čase. V případě volby této možnosti procesor vykreslí na displej časovou závislost amplitudy magnetického pole v obou směrech. Ukázka těchto dvou zobrazení je na obrázku 2.6. Průběhy ukazují naměřené pole v okolí trafopájkky.



Obrázek 2.6: Zobrazení časových průběhů a polarizace pole v okolí trafopáčky

2.6.3 Aproximace elipsou

V případě, že uvažujeme harmonický průběh dvou na sebe kolmých složek magnetického pole s různou amplitudou a stejnou frekvencí, vznikne jejich součtem elipticky polarizované magnetické pole. Naměřené body lze tedy aproximovat elipsou a za výše uvedených podmínek získat názornou grafickou interpretaci vektoru intenzity. Elipsa je jednou z kuželoseček jednoznačně popsanou pěti parametry: hlavní a vedlejší poloosou, souřadnicemi středu a úhlem natočení v rovině. Znalost těchto parametrů je výhodná z hlediska posouzení magnetického pole. Například hodnota vedlejší poloosy ukazuje minimální hodnotu intenzity a lze tedy posoudit zda je intenzita pole ve všech směrech dostatečná pro provedení magnetické zkoušky.

Pro zjištění parametrů elipsy je potřeba aproximovat naměřená data. Jako kritérium optimality bylo použito standardní kritérium nejmenšího součtu kvadrátů odchylek vzdáleností jednotlivých bodů od elipsy. Při aproximaci lze vyjít z parametrických rovnic elipsy

$$\begin{aligned} x(t) &= x_0 + \cos(\varphi) a \cos(t) - \sin(\varphi) b \sin(t) \\ y(t) &= y_0 + \sin(\varphi) a \cos(t) + \cos(\varphi) b \sin(t), \end{aligned} \quad (2.3)$$

kde x_0, y_0 jsou souřadnice středu, φ je úhel natočení a t je parametr nebo z obecné rovnice

$$Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0, \quad (2.4)$$

kde A, B, C, D, E, F jsou koeficienty popisující tvar obecné kuželosečky. V případě, že vyjdeme z parametrického vyjádření jsme postaveni před úlohu řešení soustavy nelineárních rovnic s goniometrickými funkcemi. Výhodou je přímý výpočet všech pěti

koeficientů majících jasnou grafickou interpretaci. V druhém případě vycházíme z obecné rovnice popisující obecně všechny kuželosečky a je tedy nutné odhadovat celkem šest parametrů a výsledkem nemusí být elipsa. Elipsu dostáváme pouze při splnění nerovnosti

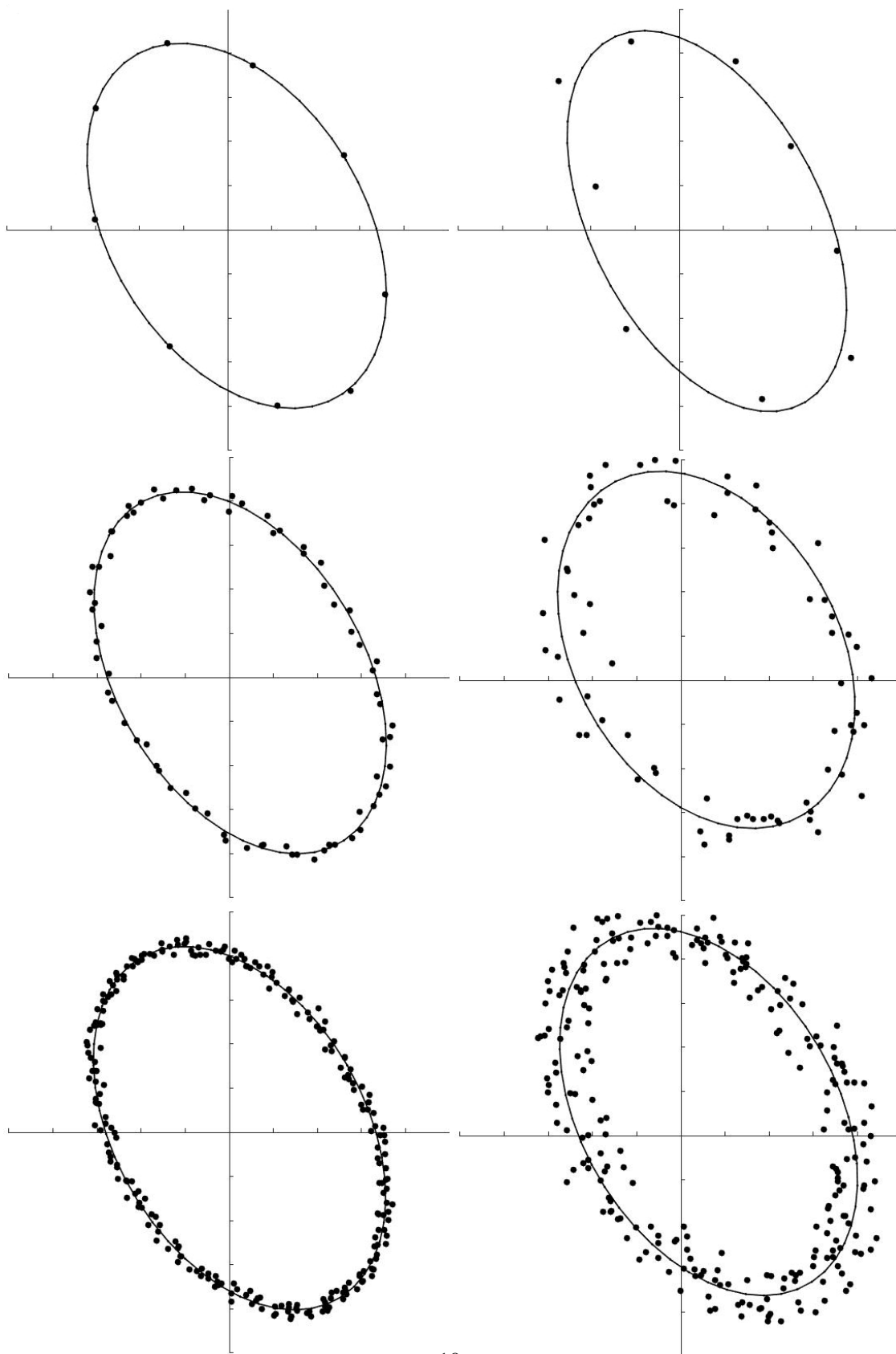
$$b^2 - 4ac < 0.$$

Tuto podmínku je nutné v aproximačních rovnicích zohlednit. Pro aproximaci byl implementován algoritmus popsáný v [11], který vychází z obecné rovnice, je numericky stabilní a výsledkem aproximace je vždy elipsa. Tento algoritmus vyžaduje implementace základních maticových operací do procesoru (součet, násobení, transpozice, inverze a nalezení vlastních čísel a vektorů), nicméně operuje vždy s maticemi stejných rozměrů bez ohledu na rozsah vstupních dat. Po výpočtu koeficientů obecné rovnice je pro zobrazení elipsy nutný přepočtení na parametrické koeficienty. Tento přepočtení probíhal pomocí algoritmu uvedeného v [9].

Pro vizuální posouzení kvality aproximace byl algoritmus nejprve implementován v matematickém programu *Matlab*. Script nejprve vygeneruje body ležící na elipse se zadanými parametry. Poté je k souřadnicím těchto bodů přičten náhodný šum a poté proběhne aproximace a přepočtení koeficientů obecné rovnice na koeficienty parametrické. Následující obrázek 2.7 ukazuje příklady aproximace popsáným algoritmem s různými hodnotami šumu a počtu bodů.

2.6.4 Uložení dat

Měřič umožňuje uložení dat na SD kartu. Před uložením procesor kartu naformátuje souborovým formátem FAT16. Každá naměřená sekvence dat se ukládá do samostatného souboru s příponou *.txt* označeného pořadovým číslem měření. Každý řádek souboru představuje jedno odebrání vzorků z výstupního signálu sond, tj. obsahuje dvě hodnoty reprezentující intenzity magnetického pole ve dvou měřených směrech. Tyto hodnoty jsou odděleny čárkou. Na kartu lze uložit až 100 souborů.

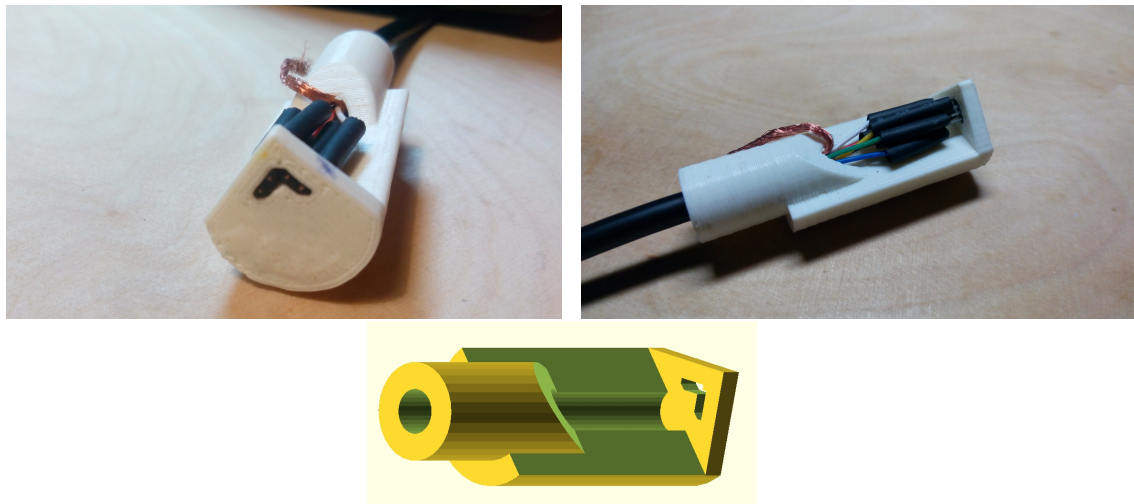


Obrázek 2.7: Aproximace elipsou pro různý počet bodů a intenzitu šumu

3 Realizace a měření parametrů

Tato kapitola popisuje průběh fyzické realizace měřice, jeho kalibraci a uvádí výsledky měření magnetického pole magnetizéru.

3.1 Uchycení sond



Obrázek 3.1: Držák sond

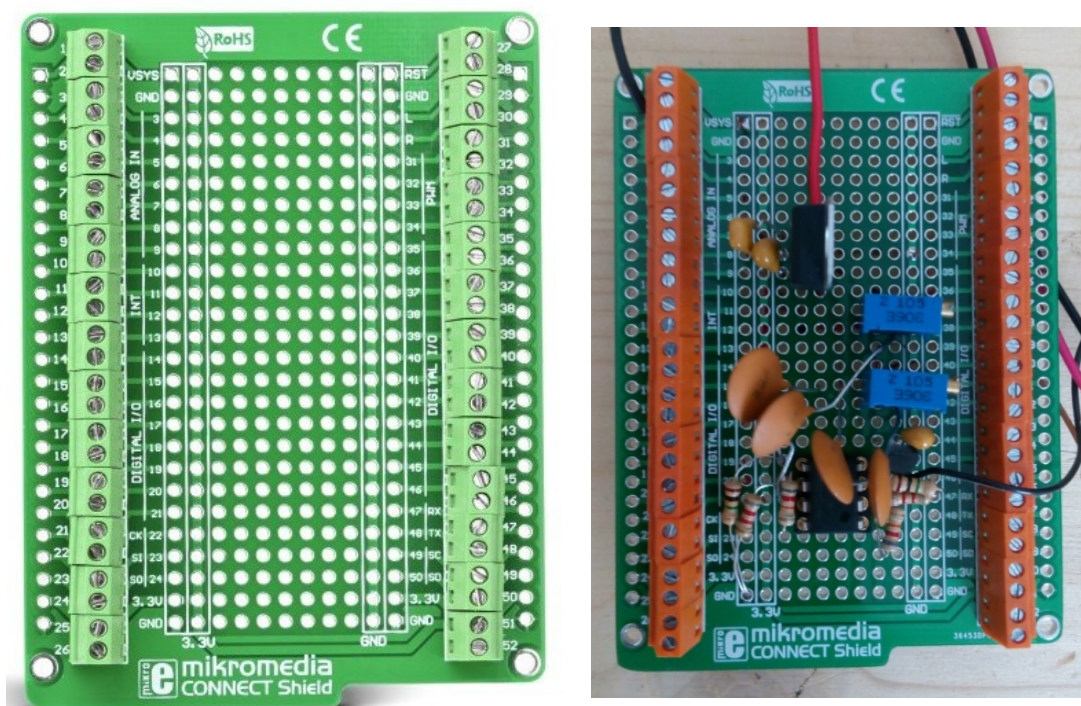
Uchycení sond v navzájem kolmé pozici bylo řešeno návrhem držáku a jeho následným vytištěním na 3D tiskárně. Držák se sondami navržený v programu OpenSCAD je na obrázku 3.1. Při návrhu bylo nutné zohlednit jejich těsné sesazení a umístění k okraji, aby bylo možné při měření kontrolovat vizuálně pozici sond. Připojení sond bylo realizováno stíněným kabelem za účelem minimalizace indukovaní napětí do napájecích a signálových vodičů. Průměr držáků byl volen tak, aby ho bylo možné nasadit na standardní vodovodní trubku PPR 20x3.4 PN20 pro pohodlnější měření. Přívodní kabel lze vést vnitřkem trubky. Zdrojový soubor návrhu držáku je součástí přílohy.

Ideálně bychom chtěli měřit charakteristiky pole v daném místě bez přítomnosti sondy.

To však není prakticky možné, jelikož je nutné měřící sondu do pole vložit. V závislosti na materiálu a fyzických rozměrech dojde k deformaci pole snímačem. Analýza narušení pole vložení sondy je nad rámec této práce. Předpokládáme, že deformace pole držákem z ABS plastu s dvěma sondami je malá.

3.2 Realizace zesilovače

Zesilovač byl realizován dle schématu 2.2 na univerzálním plošném spoji *mikromedia connect shield* kompatibilním s deskou s procesorem umožňující snadné propojení připájením 2x26 pinů. K těmto pinům je možné připojit pomocí svorkovnic periferie. Tento plošný spoj byl zvolen z důvodů univerzálnosti a jednoduchosti realizace zesilovače z diskretních součástek. Obrázek 3.2 ukazuje desku před a po osazení.



Obrázek 3.2: Univerzální deska *mikromedia connect shield* [6]

3.3 Kalibrace

Cílem kalibrace je určení jednoznačného převodního vztahu mezi výstupními hodnotami AD převodníků a měřenou magnetickou indukcí, aby bylo možné tyto hodnoty korektně zobrazit. Kalibraci prototypu zařízení lze provést jednorázovým měřením známého magnetického pole.

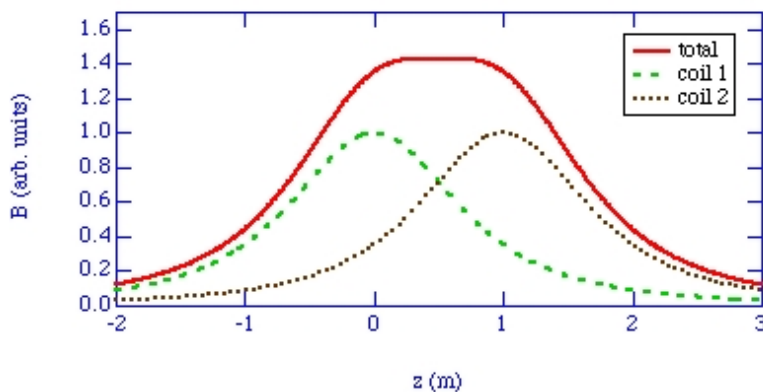
Známe pole se například tvoří kolem vodiče protékaného proudem. V okolí dlouhého rovného vodiče lze odvodit následující vztah pro magnetickou indukci v bodě:

$$B = \frac{\mu_0}{2\pi} \frac{I}{R}, \quad (3.1)$$

kde I je proud protékající vodičem a R je vzdálenost měřená kolmo k vodiči. Při znalosti proudu a vzdálenosti od vodiče jsme schopni učit magnetickou indukci. Další možností kalibrace jsou Helmholtzovy cívky. Jedná se o dvě cívky na společné ose ve vzdálenosti, která je rovna poloměru cívek. Mezi cívkami se vytvoří téměř homogenní magnetické pole. Magnetická indukce na ose uprostřed mezi cívkami je dána vztahem

$$B = \left(\frac{4}{5}\right)^{\frac{3}{2}} \frac{\mu_0 N I}{R}, \quad (3.2)$$

kde I je proud protékající cívkami, N je počet závitů cívek a R je poloměr cívek. Rozložení pole v ose cívek je na obrázku 3.3. [7]



Obrázek 3.3: Magnetické pole v ose Helmholtzových cívek [8]

Ke kalibraci přístroje byly použity Helmholtzovy cívky. Konkrétně se jednalo o dvě dlouhé cívky s podélným rozměrem $h = 70$ mm, průměrem $a = 40$ mm počtem závitů $N = 40$ a vzájemnou vzdáleností 20 mm. Amplituda magnetické intenzity intenzity v ose

cívek je dána vztahem

$$H_{max} = \frac{IN}{h} \sqrt{2} \left(\frac{\frac{h}{2} + x}{\sqrt{a^2 + (\frac{h}{2} + x)^2}} + \frac{\frac{h}{2} - x}{\sqrt{a^2 + (\frac{h}{2} - x)^2}} \right), \quad (3.3)$$

kde x je vzdálenost od středu cívky. Pro efektivní hodnotu proudu $I = 3,5$ A dostáváme intenzitu magnetického pole $1,2 \text{ kAm}^{-1}$. Tato hodnota byla ověřena vložením cívky se známými parametry do kalibračního pole. Následně byla odečtena efektivní hodnota napětí na vývodech kontrolní cívky a vypočítána intenzita magnetického pole. Kalibrace probíhala vložením sondy doprostřed Helmholtzových cívek a záznamu výstupních hodnot z AD převodníků obou kanálů. Poté bylo nalezeno minimum a maximum pro oba kanály. Vzhledem ke znalosti amplitudy jsme schopni jednoznačně stanovit převodní koeficient mezi výstupními hodnotami AD převodníků a magnetickým polem. Uprostřed mezi maximem se nachází hodnota pro nulové pole. Takto naměřené hodnoty byly zaneseny do řídicího programu.

3.4 Mechanické provedení přístroje

Pro měřič byla v programu *OpenSCAD* navržena a následně na 3D tiskárně vytištěna plastová krabice. Sondy jsou připojené pomocí šroubovacího konektoru. Vedle konektoru je umístěn posuvný spínač sloužící k zapnutí a vypnutí přístroje. Uvnitř krabice se nachází deska s procesorem a dotykovým displejem, deska se zesilovačem a napájecími obvody a baterie. Výsledná realizace prototypu je na obrázku 3.4.

3.5 Výsledky měření pole magnetizéru

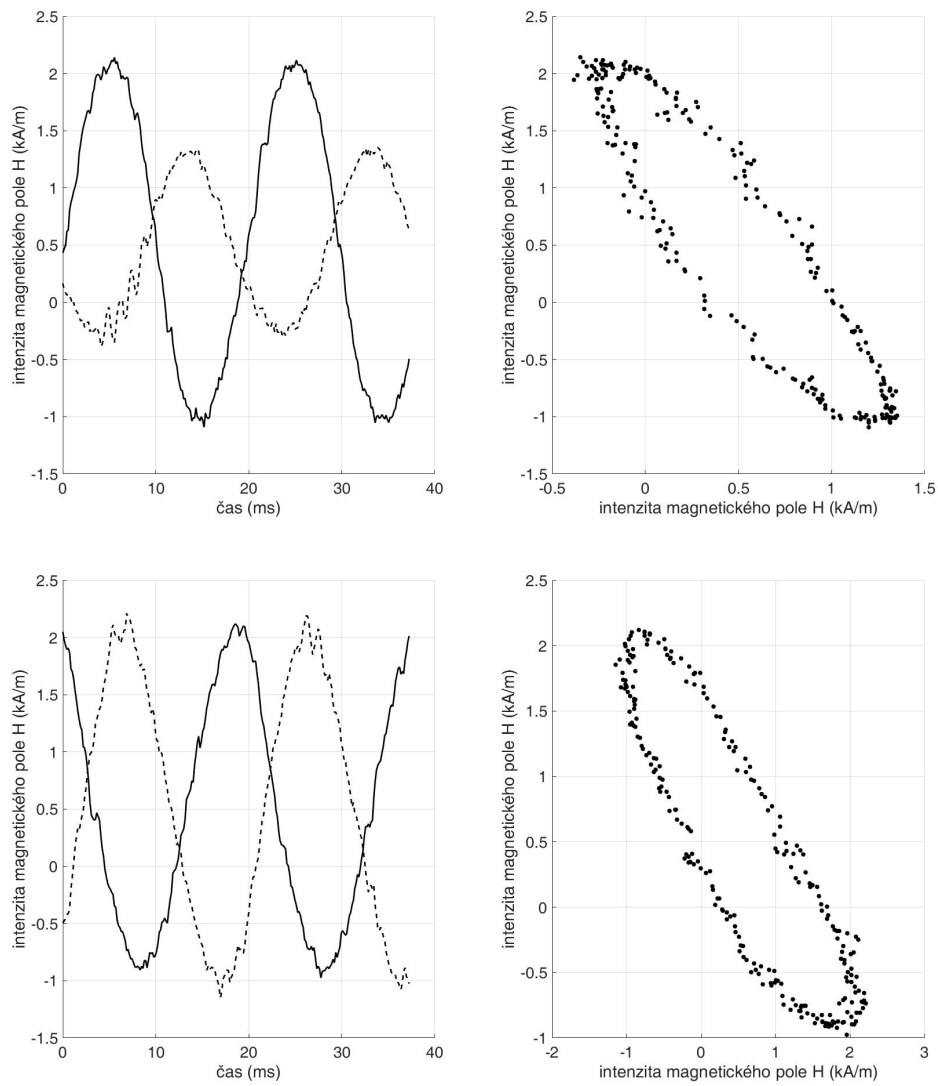
S prototypem bylo měřeno pole v okolí magnetizačního zařízení určeného pro testování ocelových trubek velkých průměrů ve firmě *ATG s.r.o.*, která se touto problematikou zabývá. Následující obrázky 3.5 a 3.6 ukazují naměřená data. Výsledky byly ukládány na paměťovou kartu a poté zpracovány v programu *Matlab* kvůli lepší možnosti zobrazení než poskytuje použitý dotykový displej malých rozměrů.

Magnetizační zařízení, které bylo v době měření ve fázi vývoje, se skládá ze čtyř cívek v konfiguraci křížového jha. Tyto cívky jsou napájeny fázově posunutými střídavými proudy s frekvencí 50 Hz. Výsledné pole je tedy polarizováno elipticky, čemuž odpovídají i výsledky měření provedeného v několika bodech v prostoru mezi cívkami. Amplituda se pohybuje v řádech jednotek kAm^{-1} . Zároveň je na výsledcích patrná

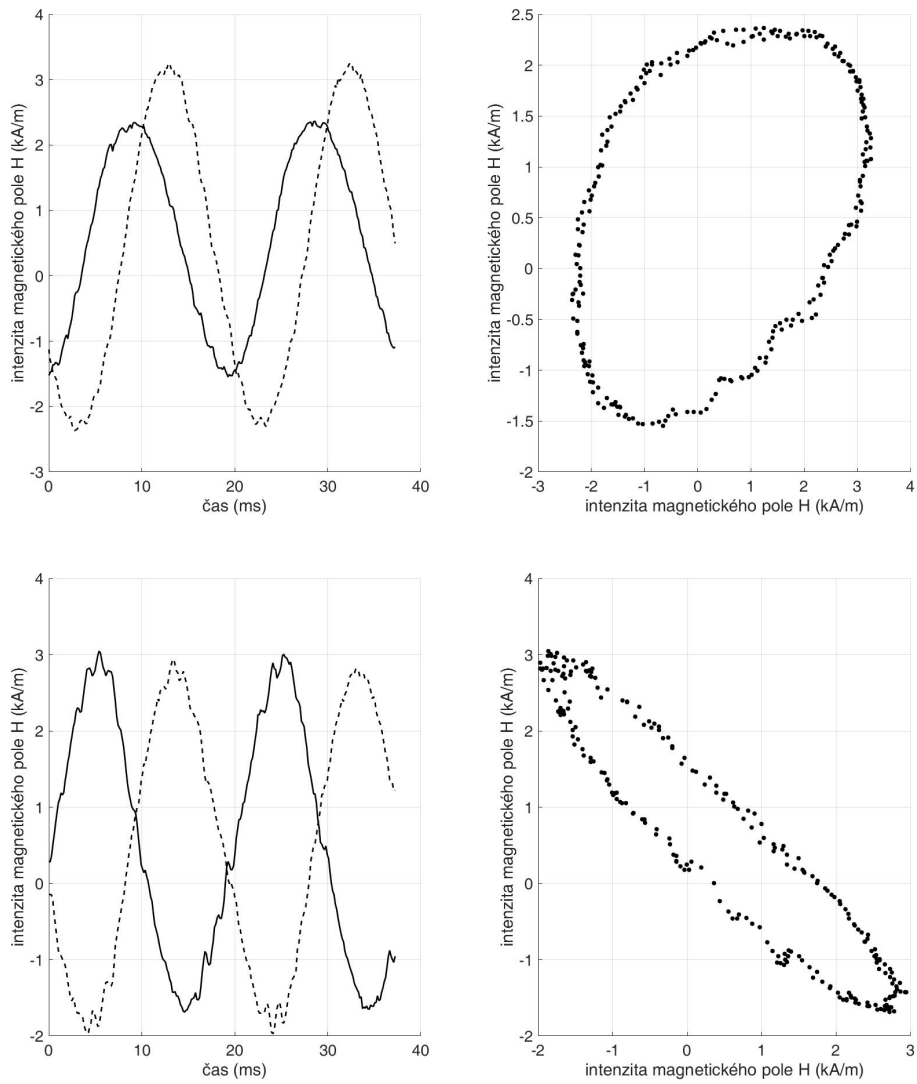


Obrázek 3.4: Výsledná realizace prototypu

přítomnost vyšších harmonických složek v budícím napětí a nemalé rušení způsobené pravděpodobně množstvím výkonových spínaných zdrojů v průmyslovém objektu.



Obrázek 3.5: Časový průběh a polarizace měřeného pole 1



Obrázek 3.6: Časový průběh a polarizace měřeného pole 2

Závěr

Cílem této práce bylo seznámit se s práškovou magnetickou metodou pro nedestruktivní testování materiálů a s požadavky na úspěšné provedení testu. Následně navrhnout a realizovat měřič magnetického pole, který bude schopen měřit intenzitu pole ve dvou na sebe kolmých směrech v rozsahu od 500 Am^{-1} do 5000 Am^{-1} . Dalším požadavkem bylo vyhodnocení nejenom intenzity, ale i vzájemného fázového posuvu složek pole a to pro frekvence do 100 Hz.

V první části byla prostudována prášková magnetická metoda pro nedestruktivní testování materiálů. Pro úspěšné provedení testu musí být velikost tangenciální složky intenzity pole alespoň 2000 Am^{-1} těsně nad povrchem materiálu. Dalším požadavkem je kolmost vektoru intenzity magnetického pole na případný defekt. Aby byly odhaleny defekty ve všech směrech měla by být polarizace pole ideálně kruhová.

Teoretický návrh uvažuje použití dvou druhů sond s Hallovým jevem. První využívá dvě navzájem kolmo sesazené sondy s napěťovým výstupem úměrným magnetickému poli v příslušném směru. Signál ze sond je zesílen zesilovačem a je upravena jeho střední hodnota tak byl využit celý rozsah AD převodníků procesoru. Druhá varianta počítá s připojením 3D senzoru s integrovanými digitálními obvody komunikující s procesorem přímo po sběrnici I²C. Pro zpracování a zobrazení dat je počítáno s procesorem řady STM32 a dotykovým displejem.

V praktické části je realizován prototyp se sondami s napěťovým výstupem a zesilovač dle návrhu uvedeném v teoretické části. Pro vyhodnocení signálu je použita deska s dotykovým displejem a procesorem STM32F407VG. Dále byl vytvořen obslužný program, který umožňuje chod a ovládání měřiče. Jsou implementovány možnosti zobrazení naměřených dat formou bodů, jejichž souřadnice reprezentují amplitudy pole v jednotlivých směrech. Další možností grafického zobrazení je aproximační elipsa určená metodou nejmenších čtverců. Program také umí zobrazit časové průběhy dat z obou kanálů a uložit výsledky měření na SD kartu.

Výsledky kalibrace ukazují, že měřicí rozsah dle zadání byl s určitou rezervou splněn. Dvanáctibitové převodníky procesoru disponují dostatečným rozlišením pro případné rozšíření rozsahu snížením zisku zesilovače. Měřič je schopen vyhodnocení polí přibližně

od 25 Hz, kdy útlum CR článku začíná být zanedbatelný a nedochází ke zkreslení amplitudy výsledků. Horní frekvenční limit je zhruba 350 Hz. Pro tuto frekvenci je časové zpoždění mezi odebráním vzorků z jednotlivých kanálů menší než 1 % ze vzorkovací periody. Navýšení frekvenčního rozsahu by bylo možné konfigurací AD převodníků do paralelního režimu ve strojovém jazyce procesoru. Frekvenční rozsah by vzrostl až do horního kmitočtu sondy tj. do 1 kHz. Požadované vyhodnocení fázového posunu lze provést z časových průběhů. Varianta se senzorem komunikujícím I²C po sběrnici nebyla v době odevzdání práce uvedena do provozu. Do budoucna je v plánu přidání výpočtu spektra naměřených polí, aby bylo možné posuzovat relevanci aproximace elipsou.

Seznam použitých zkratek a symbolů

Φ	magnetický tok
I	konstantní elektrický proud
\mathbf{B}	magnetická indukce
f	frekvence
S	plocha
L	vlastní indukčnost
t	čas
\mathbf{r}	polohový vektor
i	časově proměnný elektrický proud
R	elektrický odpor
u_e	elektromotorické napětí
M	vzájemná indukčnost
Z	impedance
OZ	operační zesilovač
AD	analog digitální
f_c	zlomový kmitočet filtru
G	zisk
u_H	Hallovo napětí
R_H	Hallova konstanta
μ	permeabilita
ε	permitivita

Seznam obrázků

1.1	Indikace defektu magnetickou metodou	4
1.2	Princip magnetické metody práškové	5
1.3	Elektromagnetické jho a křížové elektromagnetické jho	7
2.1	Blokové schéma zapojení	9
2.2	Schéma zapojení zesilovače	13
2.3	AC analýza zesilovače	14
2.4	Deska <i>mikromedia for STM32</i>	15
2.5	Úvodní obrazovka měřiče	16
2.6	Zobrazení časových průběhů a polarizace pole v okolí trafopájky	18
2.7	Aproximace elipsou pro různý počet bodů a intenzitu šumu	20
3.1	Držák sond	21
3.2	Univerzální deska <i>mikromedia connect shield</i>	22
3.3	Magnetické pole v ose Helmholtzových cívek	23
3.4	Výsledná realizace prototypu	25
3.5	Časový průběh a polarizace měřeného pole 1	26
3.6	Časový průběh a polarizace měřeného pole 2	27

Seznam tabulek

2.1	Porovnání dostupných měřičů magnetizace	8
2.2	Porovnání dostupných senzorů s Halloovým jevem	11
2.3	Porovnání dostupných 3D senzorů	12

Seznam příloh

- Příloha A** Algoritmus aproximace elipsou (Matlab)
- Příloha B** Návrh krabičky (OpenSCAD)
- Příloha C** Návrh držáků sond (OpenSCAD)
- Příloha D** Obslužný program (MikroC)

Literatura

- [1] Quality Magazine, 2013. *Magnetic Particle Powder Performance* [online]. [cit. 3.05.2017]. Dostupné z: <http://www.qualitymag.com/articles/90976-magnetic-particle-powder-performance>.
- [2] ATG s.r.o. *Magnetic Particle Testing (MT)* [online]. [cit. 15.5.2017]. Dostupné z: <https://www.atg.cz/>.
- [3] Quality System. *Hand magnetizing devices* [online]. [cit. 16.05.2017]. Dostupné z: <http://qs-ltd.com/en/ndt/magnetic-particle-testing/hand-magnetizing-devices/kmu-8-42-cross-yoke/>.
- [4] Antonín Vojáček, 2011. *GMR senzory mag. pole - 1. díl - princip a struktura* [online]. [cit. 10.04.2017]. Dostupné z: <http://automatizace.hw.cz/gmr-senzory-mag-pole-1-dil-princip-a-struktura> .
- [5] MikroElektronika d.o.o. *Smart displays* [online]. [cit. 26.04.2017]. Dostupné z: <https://shop.mikroe.com/smart-displays>.
- [6] MikroElektronika d.o.o. *Mikromedia connect shield* [online]. [cit. 20.04.2017]. Dostupné z: <https://shop.mikroe.com/smart-displays/mikromedia/mikromedia-3/connect-shield>.
- [7] Eric Dennison, 2015. *On-Axis Field of an Ideal Helmholtz Coil* [online]. [cit. 21.05.2017]. Dostupné z: <http://tiggerntatie.github.io/emagnet/helmholtz/idealmholtz.htm> .
- [8] Frank Wolfs, 2013. *THE MAGNETIC FORCE AND FIELD* [online]. [cit. 21.05.2017]. Dostupné z: http://teacher.nsrj.rochester.edu/phy122/Lecture_Notes/Chapter30/chapter30.html .
- [9] GANDER, W., STREBEL, R., AND GOLUB, G. H. Fitting of circles and ellipses least squares solution. (1994).
- [10] HAASZ, VLADIMÍR A SEDLÁČEK, MILOŠ. *Elektrická měření přístroje a metody*. Praha: Vydavatelství ČVUT, 2005. ISBN 80-01-02731-7.
- [11] HALIR, RADIM A FLUSSER, J. Numerically stable direct least squares fitting of ellipses. In *Proc. 6th International Conference in Central Europe on Computer Graphics and Visualization. WSCG* (1998), vol. 98, Citeseer, pp. 125–132.

- [12] KREJČA, JOSEF A ROXER, MIROSLAV. *Magnetická metoda prášková stupeň I, II podklady pro školení dle ISO 9712*. 2013.
- [13] NOVOTNÝ, KAREL. *Teorie elektromagnetického pole I*. Praha: Vydavatelství ČVUT, 1998. ISBN 80-01-01774-5.
- [14] PUNČOCHÁŘ, JOSEF. *Operační zesilovače v elektronice*. Praha: BEN – technická literatura, 2002. ISBN 80-730-0059-8.

Příloha A

Algoritmus aproximace elipsou (Matlab)

```
clear
close all
% generovani elipsy se sumem
elipsaPuvodni = calculateEllipse(1523, 2456, 463, 896, -30, 70);
for i =1:length(elipsaPuvodni)-1
    x(i) = round(elipsaPuvodni(i,1) + 50*rand);
    y(i) = round(elipsaPuvodni(i,2) + 50*rand);
end
%% aproximace
tic
for i = 1:length(x)
    D1(i,1) = x(i)^2;
    D1(i,2) = x(i)*y(i);
    D1(i,3) = y(i)^2;
end

for i = 1:length(x)
    D2(i,1) = x(i);
    D2(i,2) = y(i);
    D2(i,3) = 1;
end

S1 = D1' * D1;
S2 = D1' * D2;
S3 = D2' * D2;
T = - inv(S3) * S2';
M = S1 + S2 * T;
M = [M(3, :) ./ 2; - M(2, :); M(1, :) ./ 2];
[eval, evec] = eig(M);
cond = 4 * evec(1, :) .* evec(3, :) - evec(2, :).^ 2;
a1 = evec(:, find(cond > 0));
reseni = [a1; T * a1];

[stred, polosaA, polosaB, rotace err] = ellipse_params (reseni, 0);
rotace= rotace * 180 / pi;
rotace = 180 - rotace;
aproxElipsa = calculateEllipse(stred(1), stred(2), polosaA, polosaB, rotace, 50);
toc

%zobrazeni vysledku
iptsetpref('ImshowBorder','tight');
figure(1)
hold on
```

```

plot(aproxElipsa(:,1), aproxElipsa(:,2),'black.-', 'LineWidth', 1.25), axis equal
scatter(x,y,'black','o','filled', 'LineWidth',2)
set(gca,'xticklabel', [], 'yticklabel', [])
ax = gca;
ax.XAxisLocation = 'origin';
ax.YAxisLocation = 'origin';
hold off

```

```

function [X,Y] = calculateEllipse(x, y, a, b, angle, steps)
    %# This functions returns points to draw an ellipse
    narginchk(5, 6);
    if nargin<6, steps = 36;
    end

    beta = -angle * (pi / 180);
    sinbeta = sin(beta);
    cosbeta = cos(beta);

    alpha = linspace(0, 360, steps)' .* (pi / 180);
    sinalpha = sin(alpha);
    cosalpha = cos(alpha);

    X = x + (a * cosalpha * cosbeta - b * sinalpha * sinbeta);
    Y = y + (a * cosalpha * sinbeta + b * sinalpha * cosbeta);

    if nargin==1, X = [X Y];
    end
end

```

```

function [z, a, b, alpha, err] = ellipse_params (u, show);
%ELLIPSE_PARAMS Get ellipse params from algebraic equation
if (nargin < 2) show = 0; end;
err = 0;
A = [u(1) u(2)/2; u(2)/2 u(3)];
bb = [u(4); u(5)];
c = u(6);
[Q D] = eig(A);
det = D(1,1)*D(2,2);
if (det <= 0),
    err = 1;
if (show == 1), drawconic (u); end;
z = [0;0];
a = 1; b = 1; alpha = 0;
else
    bs = Q'*bb;
    alpha = atan2(Q(2,1), Q(1,1));
    zs = -(2*D)\bs;
    z = Q*zs;
    h = -bs'*zs/2-c;
    a = sqrt(h/D(1,1));
    b = sqrt(h/D(2,2));
end
end

```

Příloha B

Návrh krabičky (OpenSCAD)

```
tloustkaSteny = 2;
hloubkaZarezu = 1;
kvadrX = 60.45;
kvadrY = 1.65;
kvadrZ = 81.15;
vnitrniVyska = 45;
sirkaDispleje = 50;

difference() {
  cube(size = [kvadrX + 2*tloustkaSteny, vnitrniVyska + 2*tloustkaSteny,
  kvadrZ + 2*tloustkaSteny+3.5], center = false);

  //horni zarez deska
  translate([tloustkaSteny, 4, tloustkaSteny+hloubkaZarezu]){
    cube(size = [kvadrX, kvadrY, kvadrZ+6], center = false);
  }

  translate([tloustkaSteny+1, 3, tloustkaSteny+hloubkaZarezu]){
    cube(size = [kvadrX-2, 1, kvadrZ+6], center = false);
  }

  //dolni zarez deska
  translate([tloustkaSteny, 4 + 11, tloustkaSteny+hloubkaZarezu]){
    cube(size = [kvadrX, kvadrY, kvadrZ+6], center = false);
  }

  //zarez displej
  translate([(kvadrX + 2*tloustkaSteny - 50.2)/2, 0, tloustkaSteny+hloubkaZarezu + 4]){
    cube(size = [50.2, 4, 69.3+15], center = false);
  }

  //zarez reset
  translate([tloustkaSteny + 25, 0, tloustkaSteny+hloubkaZarezu]){
    cube(size = [15, 4, 4], center = false);
  }

  //zbytek vnitru
  translate([tloustkaSteny+hloubkaZarezu, tloustkaSteny+hloubkaZarezu,
  tloustkaSteny+hloubkaZarezu]){
    cube(size = [kvadrX - 2*hloubkaZarezu, vnitrniVyska-1, kvadrZ+6], center = false);
  }

  //konektor
```

```

translate([kvadrX - 12, vnitřniVyska - 8, 0]){
  cylinder(h=10, r=7.5, center=true);
}

//spinac
translate([12, 31.5, 0]){
  cube(size = [13, 12, 3], center = false);
}

//drážka
translate([2, 0, kvadrZ +2*tloušťkaSteny- 0.5]){
  cube(size = [kvadrX, vnitřniVyska+2, 2], center = false);
}

//uriznout rohy
translate([3, 0, kvadrZ +2*tloušťkaSteny+1.5]){
  cube(size = [kvadrX-2, 3, 2], center = false);
}

//skosení hran pro výrobu
translate([2, 0, kvadrZ +2*tloušťkaSteny+1.5]){
  rotate(a=[0,45,0]){
    cube(size = [2, vnitřniVyska+2, 2], center = false);
  }
}

translate([kvadrX + 2*tloušťkaSteny-2, 0, kvadrZ +2*tloušťkaSteny+1.5]){
  rotate(a=[0,-135,0]){
    cube(size = [2, vnitřniVyska+2, 2], center = false);
  }
}

//dvirka
rotate(a=[0,180,0]){
  translate([25, 0, 0]){
    cube(size = [kvadrX-0.5, 45+2, 2], center = false);
    translate([5, 0, -8.35]){
      cube(size = [50.2, 4, 8.35], center = false);
    }
  }
}
}

```

Příloha C

Návrh držáků sond (OpenSCAD)

```
$fn = 30;

polomerTrubky = 10;
hranaKvadru = 10*sqrt(2);
vyskaKvadru = 3;
vzdalenostSondOdKraje = 2.5;
vyskaValce = 40;
polomerValce = 10;
vnitriPolomer = 3;
vyskaMalehoValce = 30;

//rotate(45, 0, 0) {
difference() {
    cube(size = [hranaKvadru, hranaKvadru, vyskaKvadru], center = false);

    translate ([vzdalenostSondOdKraje, vzdalenostSondOdKraje, 0]) {
        linear_extrude(height = vyskaKvadru){
            polygon(points=[[0,0.68], [0,4.72], [0.68,4.72], [1.47,3.93],
                [1.47,1.47], [3.93,1.47], [4.72,0.78], [4.72,0],
                [0.68,0], [0.68,0.68]]);
        }
    }
}

difference() {
    translate ([hranaKvadru/2, hranaKvadru/2, 0]) {
        cylinder(vyskaValce, polomerTrubky, polomerTrubky, false);
    }

    rotate([0, 0, 45]) {
        translate ([0, -polomerTrubky, 0]) {
            cube([polomerTrubky, 2*polomerTrubky, vyskaValce], false);
        }
    }

    translate ([hranaKvadru/2, hranaKvadru/2, 0]) {
        cylinder(vyskaValce, vnitriPolomer, vnitriPolomer, false);
    }
}

difference() {
```

```
translate ([hranaKvadru/2, hranaKvadru/2, vyskaValce - (vyskaMalehoValce/2)]) {
  cylinder(vyskaMalehoValce, 6.6, 6.6, false);
}

translate ([hranaKvadru/2, hranaKvadru/2, vyskaValce - (vyskaMalehoValce/2)]) {
  cylinder(vyskaMalehoValce, vnitriPolomer, vnitriPolomer, false);
}

translate ([hranaKvadru, 0, vyskaValce - (vyskaMalehoValce/2)]) {
  rotate([0, 240, 45]) {
    cube([25, 25, 10], false);
  }
}
}
```

Příloha D

Obslužný program (MikroC)

```
#include "stdint.h"
#include "__Dsp.h"

// TFT pripojeni
unsigned int TFT_DataPort at GPIOE_ODR;
sbit TFT_RST at GPIOE_ODR.B8;
sbit TFT_RS at GPIOE_ODR.B12;
sbit TFT_CS at GPIOE_ODR.B15;
sbit TFT_RD at GPIOE_ODR.B10;
sbit TFT_WR at GPIOE_ODR.B11;
sbit TFT_BLED at GPIOE_ODR.B9;

// TP pripojeni
sbit DriveX_Left at GPIOB_ODR.B1;
sbit DriveX_Right at GPIOB_ODR.B8;
sbit DriveY_Up at GPIOB_ODR.B9;
sbit DriveY_Down at GPIOB_ODR.B0;

// MMC Chip Select
sbit Mmc_Chip_Select at GPIOD_ODR.B3;

//globalni promenne
const NUMBER_OF_SAMPLES = 256;
unsigned int xSamples[NUMBER_OF_SAMPLES], ySamples[NUMBER_OF_SAMPLES];
unsigned int i, xCoord, yCoord;
unsigned short screen = 0;
unsigned short cardInit = 0;
unsigned short fileNumber = 0;
char filename[13] = "mereni00.txt";
char out[7];
double elipseParams[5];
unsigned int backgroundColor, textColor, axisColor,
pointsColor, traceColor1, traceColor2, buttonBackground;

void colorConversion() {
    backgroundColor = TFT_RGBToColor16bit(10, 10, 10);
    textColor = TFT_RGBToColor16bit(59, 105, 213);
    axisColor = TFT_RGBToColor16bit(220, 220, 220);
    pointsColor = TFT_RGBToColor16bit(255, 255, 0);
    traceColor1 = TFT_RGBToColor16bit(255, 255, 0);
    traceColor2 = TFT_RGBToColor16bit(0, 204, 0);
    buttonBackground = TFT_RGBToColor16bit(15, 25, 40);
}
```



```

void drawEllipse(double *params) {
    const numPoints = 360;
    double x[numPoints];
    double y[numPoints];
    double sin0, cos0, Xo, Yo, a, b, o;
    unsigned int newValueX, newValueY;
    Xo = params[0];
    Yo = params[1];
    a = params[2];
    b = params[3];
    o = params[4];

    o = o * 0.01745;
    sin0 = sin(o);
    cos0 = cos(o);

    for(i=0; i < numPoints; i++) {
        x[i] = Xo + (a*cos0*cos(i * 0.01745) - b*sin0*sin(i * 0.01745)); //pi/180=0.01745
        y[i] = Yo + (a*sin0*cos(i * 0.01745) + b*cos0*sin(i * 0.01745));
    }

    for(i = 0; i < numPoints; i++) {
        newValueX = (x[i] - 381) * 0.05615; // 230/4096=0.05615
        newValueY = (y[i] - 333) * 0.05615; // 381 a 333 z kalibrace posun nuly
        TFT_Dot((4 + newValueX), (234 - newValueY), traceColor1);
    }
}

void ellipseFit() {
    int i,j,k;
    double sum = 0;
    double s1[3][3];
    double s2[3][3];
    double s3[3][3];
    double T[3][3];
    double M[3][3];
    double s3inv[3][3];
    double det;
    long double a = 0;
    double eigVal[3];
    double eigVect[3][3];
    double cond[3];

    //S1(0,0)
    for(i = 0; i < NUMBER_OF_SAMPLES; i++) {
        sum = sum + xSamples[i] * xSamples[i] * xSamples[i] * xSamples[i];
    }
    s1[0][0] = sum;

    //S1(0,1) a S1(1,0)
    sum = 0;
    for(i = 0; i < NUMBER_OF_SAMPLES; i++) {
        sum = sum + xSamples[i] * xSamples[i] * xSamples[i] * ySamples[i];
    }
}

```

```

s1[0][1] = sum;
s1[1][0] = sum;

//S1(0,2) a S1(2,0)
sum = 0;
for(i = 0; i < NUMBER_OF_SAMPLES; i++) {
    sum = sum + xSamples[i] * xSamples[i] * ySamples[i] * ySamples[i];
}
s1[0][2] = sum;
s1[2][0] = sum;

//S1(1,1)
sum = 0;
for(i = 0; i < NUMBER_OF_SAMPLES; i++) {
    sum = sum + xSamples[i] * ySamples[i] * xSamples[i] * ySamples[i];
}
s1[1][1] = sum;

//S1(1,2) a S1(2,1)
sum = 0;
for(i = 0; i < NUMBER_OF_SAMPLES; i++) {
    sum = sum + xSamples[i] * ySamples[i] * ySamples[i] * ySamples[i];
}
s1[1][2] = sum;
s1[2][1] = sum;

//S1(2,2)
sum = 0;
for(i = 0; i < NUMBER_OF_SAMPLES; i++) {
    sum = sum + ySamples[i] * ySamples[i] * ySamples[i] * ySamples[i];
}
s1[2][2] += sum;

//S2(0,0)
sum = 0;
for(i = 0; i < NUMBER_OF_SAMPLES; i++) {
    sum = sum + xSamples[i] * xSamples[i] * xSamples[i];
}
s2[0][0] = sum;

//S2(0,1) a S2(1,0)
sum = 0;
for(i = 0; i < NUMBER_OF_SAMPLES; i++) {
    sum = sum + xSamples[i] * xSamples[i] * ySamples[i];
}
s2[0][1] = sum;
s2[1][0] = sum;

//S2(0,2)
sum = 0;
for(i = 0; i < NUMBER_OF_SAMPLES; i++) {

```

```

    sum = sum + xSamples[i] * xSamples[i];
}
s2[0][2] = sum;

//S2(1,1)
sum = 0;
for(i = 0; i < NUMBER_OF_SAMPLES; i++) {
    sum = sum + xSamples[i] * ySamples[i] * ySamples[i];
}
s2[1][1] = sum;

//S2(1,2)
sum = 0;
for(i = 0; i < NUMBER_OF_SAMPLES; i++) {
    sum = sum + xSamples[i] * ySamples[i];
}
s2[1][2] = sum;

//S2(2,0)
sum = 0;
for(i = 0; i < NUMBER_OF_SAMPLES; i++) {
    sum = sum + ySamples[i] * ySamples[i] * xSamples[i];
}
s2[2][0] = sum;

//S2(2,1)
sum = 0;
for(i = 0; i < NUMBER_OF_SAMPLES; i++) {
    sum = sum + ySamples[i] * ySamples[i] * ySamples[i];
}
s2[2][1] = sum;

//S1(2,2)
sum = 0;
for(i = 0; i < NUMBER_OF_SAMPLES; i++) {
    sum = sum + ySamples[i] * ySamples[i];
}
s2[2][2] = sum;

//S3(0,0)
s3[0][0] = s2[0][2];

//S3(0,1) a S3(1,0)
sum = 0;
for(i = 0; i < NUMBER_OF_SAMPLES; i++) {
    sum = sum + xSamples[i] * ySamples[i];
}
s3[0][1] = sum;
s3[1][0] = sum;

//S3(0,2) a S3(2,0)
sum = 0;
for(i = 0; i < NUMBER_OF_SAMPLES; i++) {

```

```

        sum = sum + xSamples[i];
    }
    s3[0][2] = sum;
    s3[2][0] = sum;

    //S3(1,1)
    s3[1][1] = s2[2][2];

    //S3(1,2) a S(2,1)
    sum = 0;
    for(i = 0; i < NUMBER_OF_SAMPLES; i++) {
        sum = sum + ySamples[i];
    }
    s3[1][2] = sum;
    s3[2][1] = sum;

    s3[2][2] = NUMBER_OF_SAMPLES;

// T = -inv(s3) * s2T
det = s3[0][0]*((s3[1][1]*s3[2][2]) - (s3[2][1]*s3[1][2]))
      -s3[0][1]*(s3[1][0]*s3[2][2] - s3[2][0]*s3[1][2]) + s3[0][2]*
      (s3[1][0]*s3[2][1] - s3[2][0]*s3[1][1]);
det = 1/det;
s3inv[0][0] = -((s3[1][1]*s3[2][2] - s3[1][2]*s3[2][1]) * det);
s3inv[0][1] = -((s3[0][2]*s3[2][1] - s3[0][1]*s3[2][2]) * det);
s3inv[0][2] = -((s3[0][1]*s3[1][2] - s3[0][2]*s3[1][1]) * det);
s3inv[1][0] = -((s3[1][2]*s3[2][0] - s3[1][0]*s3[2][2]) * det);
s3inv[1][1] = -((s3[0][0]*s3[2][2] - s3[0][2]*s3[2][0]) * det);
s3inv[1][2] = -((s3[0][2]*s3[1][0] - s3[0][0]*s3[1][2]) * det);
s3inv[2][0] = -((s3[1][0]*s3[2][1] - s3[1][1]*s3[2][0]) * det);
s3inv[2][1] = -((s3[0][1]*s3[2][0] - s3[0][0]*s3[2][1]) * det);
s3inv[2][2] = -((s3[0][0]*s3[1][1] - s3[0][1]*s3[1][0]) * det);

for (i = 0; i <= 2; i++) {
    for (j = 0; j <= 2; j++) {
        det = 0;
        for (k = 0; k <= 2; k++) {
            det = det - s3inv[i][k] * s2[j][k];
        }
        T[i][j] = det;
    }
}

//m = s1 + s2*T

for (i = 0; i <= 2; i++) {
    for (j = 0; j <= 2; j++) {
        det = 0;
        for (k = 0; k <= 2; k++) {
            det = det + s2[i][k] * T[k][j];
        }
    }
}

```

```

        M[i][j] = det;
    }
}

for (i = 0; i <= 2; i++) {
    for (j = 0; j <= 2; j++) {
        M[i][j] = M[i][j] + s2[i][j];
    }
}

// m*inv(C1)
for(i = 0; i <= 2; i++) {
    M[0][i] = M[1][i] / 2;
    M[1][i] = M[2][i] * (-1);
    M[2][i] = M[2][i] / 2;
}

// jacobi(M,3,eigVal,eigVect,i);

cond[0] = 4 * eigVect[0][0] * eigVect[2][0] - eigVect[1][0] * eigVect[1][0];
cond[1] = 4 * eigVect[0][1] * eigVect[2][1] - eigVect[1][1] * eigVect[1][1];
cond[2] = 4 * eigVect[0][2] * eigVect[2][2] - eigVect[1][2] * eigVect[1][2];

for(i = 0; i < 3; i++) {
    if (cond[i] > 0) {
        det = i;
        break;
    }
}

for(i = 0; i < 3; i++) {
    ellipseParams[i] = eigVect[i][det];
}

for(i = 0; i < 3; i++) {
    for(j = 0; j < 3; j++) {
        ellipseParams[i+3] = ellipseParams[i+3] + T[i][j] * ellipseParams[j];
    }
}
}

//zapis na kartu
void writeFileToCard() {
    char newLine[3], stringToCard[7];

    char c = (fileNumber % 10) + '0';
    filename[7] = c;
    c = ((fileNumber - (fileNumber % 10)) / 10) + '0';
    filename[6] = c;

    Mmc_Fat_Assign(&filename, 0xA0); // Find existing file or create a new one
    Mmc_Fat_Rewrite(); // To clear file and start with new data
    newLine[0] = 0xD;
}

```

```

newLine[1] = 0xA;

for(i = 0; i <= NUMBER_OF_SAMPLES; i++) {
    IntToStrWithZeros(xSamples[i], stringToCard);
    Mmc_Fat_Write(stringToCard, 6);
    Mmc_Fat_Write(",", 1);
    IntToStrWithZeros(ySamples[i], stringToCard);
    Mmc_Fat_Write(stringToCard, 6);
    Mmc_Fat_Write(",", 1);

    Mmc_Fat_Write(newLine, 2);
}
fileNumber++;
}

//inicializace karty
void saveToCard() {
    if(cardInit == 0) {
        TFT_Set_Font(&TFT_defaultFont, textColor, FO_HORIZONTAL);
        TFT_Write_Text("Card initialization ...", 10, 10);

        //inicializece SDKarty
        if (Mmc_Fat_QuickFormat("SDCard") == 0) {
            cardInit = 1;
            TFT_Set_Font(&TFT_defaultFont, backgroundColor, FO_HORIZONTAL);
            TFT_Write_Text("Card initialization ...", 10, 10);

            TFT_Set_Font(&TFT_defaultFont, textColor, FO_HORIZONTAL);
            TFT_Write_Text("Card initialization successful !", 10, 10);

            //rychlejsi SPI
            SPI3_Init_Advanced(_SPI_FPCLK_DIV2, _SPI_MASTER | _SPI_8_BIT |
                _SPI_CLK_IDLE_LOW | _SPI_FIRST_CLK_EDGE_TRANSITION |
                _SPI_MSB_FIRST | _SPI_SS_DISABLE | _SPI_SSM_ENABLE |
                _SPI_SSI_1,
                &_GPIO_MODULE_SPI3_PC10_11_12);

            TFT_Set_Font(&TFT_defaultFont, backgroundColor, FO_HORIZONTAL);
            TFT_Write_Text("Card initialization successful !", 10, 10);
        }else{
            TFT_Set_Font(&TFT_defaultFont, textColor, FO_HORIZONTAL);
            TFT_Write_Text("Card initialization failed", 10, 10);
            Delay_ms(200);
            TFT_Set_Font(&TFT_defaultFont, backgroundColor, FO_HORIZONTAL);
            TFT_Write_Text("Card initialization failed", 10, 10);
            return;
        }
    }
}

TFT_Set_Font(&TFT_defaultFont, textColor, FO_HORIZONTAL);
TFT_Write_Text("Creating file...", 10, 10);

//zapis dat

```

```

writeFileToCard();

TFT_Set_Font(&TFT_defaultFont, backgroundColor, FO_HORIZONTAL);
TFT_Write_Text("Creating file...", 10, 10);
TFT_Set_Font(&TFT_defaultFont, textColor, FO_HORIZONTAL);
TFT_Write_Text("File created!", 10, 10);
Delay_ms(200);
TFT_Set_Font(&TFT_defaultFont, backgroundColor, FO_HORIZONTAL);
TFT_Write_Text("File created!", 10, 10);
}

void neni() {
    TFT_Set_Font(&TFT_defaultFont, textColor, FO_HORIZONTAL);
    TFT_Write_Text("Tato funkce zatim nebyla", 10, 100);
    TFT_Write_Text("implementovana", 10, 110);
}

//smaze levou cast TP
void deleteLeftPartOfTP() {
    TFT_Set_Pen(backgroundColor, 120);
    TFT_Rectangle(61, 61, 179, 179);
}

//urci macknute tlacitko
void checkButtons() {
    if(xCoord > 240) {
        if(yCoord < 39) {
            screen = 1;
            return;
        }
        if(yCoord < 79) {
            screen = 2;
            return;
        }
        if(yCoord < 119) {
            screen = 3;
            return;
        }
        if(yCoord < 159) {
            screen = 4;
            return;
        }
        if(yCoord < 199) {
            screen = 5;
            return;
        }
        screen = 6;
        return;
    }
    screen = 255;
}

```

```

// testuje jestli neni zmacknuto
void checkTP() {
    if (TP_TFT_Press_Detect() {
        if(TP_TFT_Get_Coordinates(&xCoord, &yCoord) == 0){

            checkButtons();

            TFT_Set_Font(TFT_defaultFont, textColor, FO_HORIZONTAL);
            TFT_Write_Text("pressed", 10, 10);
            IntToStr(screen, out);
            TFT_Set_Font(TFT_defaultFont, textColor, FO_HORIZONTAL);
            TFT_Write_Text(out, 60, 10);
            Delay_ms(50);

            TFT_Set_Font(TFT_defaultFont, backgroundColor, FO_HORIZONTAL);
            TFT_Write_Text("pressed", 10, 10);
            IntToStr(screen, out);
            TFT_Set_Font(TFT_defaultFont, backgroundColor, FO_HORIZONTAL);
            TFT_Write_Text(out, 60, 10);
        }
    }
}

// najde maximum pole
unsigned int findMax(unsigned int *array, unsigned int arrayLength ) {
    unsigned int max = 0;
    for(i = 0; i < arrayLength; i++) {
        if(array[i] > max) {
            max = array[i];
        }
    }
    return max;
}

// nakresli krizek an obrazovku
void drawCross (unsigned long centerX, unsigned long centerY,
unsigned long length, unsigned int color) {
    //nakresli krizek na souradnicich s danou velikosti
    unsigned long a;
    TFT_Dot(centerX, centerY, color);
    for (a = 1; a <= length; a++ ) {
        TFT_Dot(centerX - a, centerY - a, color);
        TFT_Dot(centerX - a, centerY + a, color);
        TFT_Dot(centerX + a, centerY - a, color);
        TFT_Dot(centerX + a, centerY + a, color);
    }
}

void drawCosses(unsigned short crossLength) {
    unsigned int newX, newY, maxX, maxY;
    //nakresli osy
    TFT_Set_Pen(axisColor, 1);
    TFT_V_Line(4, 234, 119);
}

```



```

TFT_H_Line(4, 234, 119);
TFT_H_Line(116, 122, 119+38);
TFT_H_Line(116, 122, 119+76);
TFT_H_Line(116, 122, 119-38);
TFT_H_Line(116, 122, 119-76);
TFT_V_Line(116, 122, 119+37);
TFT_V_Line(116, 122, 119+74);
TFT_V_Line(116, 122, 119-37);
TFT_V_Line(116, 122, 119-74);
TFT_Set_Font(TFT_defaultFont, textColor, FO_HORIZONTAL);
TFT_Write_Text("x", 225, 100);
TFT_Write_Text("y", 125, 220);
TFT_Set_Font(TFT_defaultFont, axisColor, FO_HORIZONTAL);
TFT_Write_Text("3 kA/m", 125, 35);
TFT_Write_Text("-3 kA/m", 22, 120);
TFT_Write_Text("-3 kA/m", 126, 187);
TFT_Write_Text("3 kA/m", 175, 120);

//preskaluje a nakresli body dle mereni
for(i = 0; i < NUMBER_OF_SAMPLES; i++) {

    newX = (xSamples[i] - 381)* 0.05615; // 230/4096=0.05615
    newY = (ySamples[i] - 333)* 0.05615; //381 a 333 z kalibrace posun nuly

    newX = 4 + newX;
    newY = 235 - newY;

    drawCross(newX, newY, crossLength, pointsColor);
}

// zobrazeni vzorku v case
void samplesVsTime() {
    unsigned int newValueX, newValueY;

    //osy
    TFT_Set_Pen(axisColor, 1);
    TFT_H_Line(1, 234, 119);
    TFT_H_Line(1, 5, 119+38);
    TFT_H_Line(1, 5, 119+76);
    TFT_H_Line(1, 5, 119-38);
    TFT_H_Line(1, 5, 119-76);
    TFT_Set_Font(TFT_defaultFont, textColor, FO_HORIZONTAL);
    TFT_Write_Text("t", 230, 120);
    TFT_Set_Font(TFT_defaultFont, axisColor, FO_HORIZONTAL);
    TFT_Write_Text("3 kA/m", 10, 35);
    TFT_Write_Text("-3 kA/m", 10, 187);

    for(i = 0; i < NUMBER_OF_SAMPLES; i++) {
        if(i > 234) {
            return;
        }
    }
}

```

```

        newValueX = (xSamples[i] - 381) * 0.05615; // 230/4096=0.05615
        newValueY = (ySamples[i] - 333) * 0.05615; // 381 a 333 z kalibrace posun nuly
        TFT_Dot(i + 1, (234 - newValueX), traceColor1);
        TFT_Dot(i + 1, (234 - newValueX + 1), traceColor1);
        TFT_Dot(i + 1, (234 - newValueX - 1), traceColor1);
        TFT_Dot(i + 1, (234 - newValueY), traceColor2);
        TFT_Dot(i + 1, (234 - newValueY + 1), traceColor2);
        TFT_Dot(i + 1, (234 - newValueY - 1), traceColor2);
    }
}

void getSamples(){
    for(i = 0; i < NUMBER_OF_SAMPLES; i++){
        xSamples[i] = ADC1_Get_Sample(1);
        ySamples[i] = ADC2_Get_Sample(2);
        Delay_us(50); //10 pro 1,5 peridy na 50hz na F107
    }
}

void deleteMeasureScreen(){
    //smaze text mereni
    TFT_Set_Font(&TFT_defaultFont, backgroundColor, FO_HORIZONTAL);
    TFT_Write_Text("Merim", 10, 10);
}

void measureScreen(){
    TFT_Set_Font(&TFT_defaultFont, textColor, FO_HORIZONTAL);
    TFT_Write_Text("Merim", 10, 10);
}

void firstScreen() {
    //barva pod tlacitkama
    TFT_Set_Pen(buttonBackground, 80);
    TFT_V_Line(0, 239, 279);

    //svisla cara odelujici tlacitka a grafy a
    //vodorovne cary odelujici talcitka
    TFT_Set_Pen(axisColor, 3);
    TFT_V_Line(0, 239, 240);
    TFT_H_Line(239, 319, 39);
    TFT_H_Line(239, 319, 79);
    TFT_H_Line(239, 319, 119);
    TFT_H_Line(239, 319, 159);
    TFT_H_Line(239, 319, 199);
    //po okraji
    TFT_Set_Pen(axisColor, 1);
    TFT_H_Line(0, 319, 0);
    TFT_H_Line(0, 319, 239);
    TFT_V_Line(0, 239, 0);
    TFT_V_Line(0, 239, 319);

    TFT_Set_Font(&TFT_defaultFont, textColor, FO_HORIZONTAL);
    TFT_Write_Text("MERENI", 260, 10);
}

```

```

TFT_Write_Text("BODY", 265, 50);
TFT_Write_Text("ELIPSA", 260, 90);
TFT_Write_Text("CASOVY", 258, 124);
TFT_Write_Text("PRUBEH", 258, 138);
TFT_Write_Text("FFT", 270, 170);
TFT_Write_Text("ULOZIT", 262, 210);

//tlacitko uprosted k zahajeni mereni
TFT_Set_Pen(axisColor, 3);
TFT_Rectangle_Round_Edges(59, 99, 179, 139, 5);
TFT_Set_Font(&TFT_defaultFont, textColor, FO_HORIZONTAL);
TFT_Write_Text("Zahajit mereni", 75, 112);
}

void deleteFirstScreen() {
//smaze tlacitko uprosted k zahajeni mereni
TFT_Set_Pen(backgroundColor, 3);
TFT_Rectangle_Round_Edges(59, 99, 179, 139, 5);
TFT_Set_Font(&TFT_defaultFont, backgroundColor, FO_HORIZONTAL);
TFT_Write_Text("Zahajit mereni", 75, 112);
}

void inicialization() {
// inicializace TFT TP
GPIO_Config(&GPIOE_BASE, _GPIO_PINMASK_9, _GPIO_CFG_DIGITAL_OUTPUT);
TFT_Set_Default_Mode();
TP_TFT_Set_Default_Mode();

//inicializace ADC, channel 3 a 4 pro sondy, channel 8 a 9 pro TP
ADC_Set_Input_Channel(_ADC_CHANNEL_1 | _ADC_CHANNEL_2 |
    _ADC_CHANNEL_8 | _ADC_CHANNEL_9);
ADC1_Init();
ADC2_Init();
Delay_ms(100);

// inicializace TFT TP rozsvitit kalibrovat
TFT_Set_Default_Mode();
TFT_Init_ILI9341_8bit(320, 240);

TP_TFT_Init(320, 240, 8, 9);
TP_TFT_Set_ADC_Threshold(250);

SPI3_Init_Advanced(_SPI_FPCLK_DIV64, _SPI_MASTER | _SPI_8_BIT |
    _SPI_CLK_IDLE_LOW | _SPI_FIRST_CLK_EDGE_TRANSITION |
    _SPI_MSB_FIRST | _SPI_SS_DISABLE | _SPI_SSM_ENABLE |
    _SPI_SSI_1, &_GPIO_MODULE_SPI3_PC10_11_12);

TFT_BLED = 1;

colorConversion();

TP_TFT_Set_Calibration_Consts(325, 3600, 250, 3700);
TFT_Fill_Screen(backgroundColor);

```

```

}

void main() {
    // inicializace TP TFT SPI ADC
    inicialization();
    // pocatecni obrazovka
    firstScreen();
    //cekaci smycka pro zahajeni mereni
    while(TP_TFT_Press_Detect() == 0) {
        Delay_ms(10);
    }

    //smazat pocatecni obrazovku
    deleteFirstScreen();

    //ziskani samplu z ADC
    getSamples();
    Delay_ms(10);

    //vyjresli namerene body
    drawCosses(3);

    //smycka pro kontrolu doteku TP
    while(1) {
        checkTP();
        //akce podle stinuteho tlacitka
        switch (screen) {
            case 1: measureScreen();
                    getSamples();
                    deleteMeasureScreen();

            case 2: deleteLeftPartOfTP();
                    drawCosses(3);
                    screen = 255;
                    break;

            case 3: deleteLeftPartOfTP();
                    ellipseFit();
                    drawEllipse(ellipseParams);
                    screen = 255;
                    break;

            case 4: deleteLeftPartOfTP();
                    samplesVsTime();
                    screen = 255;
                    break;

            case 5: deleteLeftPartOfTP();
                    neni();
                    screen = 255;
                    break;

            case 6: saveToCard();

```

```
        screen = 255;
        break;

    default:break;
}
Delay_ms(5);
}
```