TÉCNICO LISBOA

Development of a Magnetic Camera for Barcode and QR Magnetic Identification Tag Readout

Alberto Nicolicea

Thesis to obtain the Master of Science Degree in

Engineering Physics

Supervisor(s): Prof. Susana Isabel Pinheiro Cardoso de Freitas

Examination Committee

Chairperson: Prof. Carlos Manuel Dos Santos Rodrigues da Cruz
Supervisor: Prof. Susana Isabel Pinheiro Cardoso de Freitas
Members of the Committee: Prof. Diogo Miguel Bárbara Coroas Prista Caetano

December 2021

Acknowledgments

Special Acknowledgments to the institution of INESC Microsystems and Nanotechnologies at the Instituto Superior Técnico, University of Lisbon, Portugal, for providing the equipment and environment necessary for the development of this thesis and to the teacher and supervisor Susana Isabel Cardoso Pinheiro de Freitas for supervising the thesis and giving much needed help and support during the process.

The design and nano-fabrication of the sensors was done by the author of this thesis, but the experimental work was done with the help of the colleagues from INESC-MN that have provided essential help during this process: João Pereira, Daniela Pereira, Maria Manuela, Rita Soares, Marília Silva, Sofia Abrunhosa. The process engineers such as Virginia Soares and José Bernardo also provided valuable assistance, especially with the lithography process.

The design of the PCB Board, the ordering of the components and assembly was also done by the author of the thesis. In the development of the electronics the help from Fabian Näf from INESC-ID was essential in indicating the software tools and the amplification circuit to be used.

Resumo

Os códigos QR têm ganho popularidade nos anos recented e são muito usados nos dias de hoje nas atividades comerciais. A tecnologia atual para deteção é baseada em dispositivos ópticos. A inovadora ideia de desenvolver um leitor de códigos QR impressos em tinta magnética usando sensores TMR é explorada nesta tese.

A motivação para o desenvolvimento de códigos QR Magnéticos vem do facto de fornecer uma possível forma de transmitir dados secretos que podem vir a ter aplicações práticas em sistemas de segurança e de prevenção de falsificações. Por exemplo, as notas bancárias podias tem um código QR magnético impresso que permitisse identificar as notas reais das falsas.

A tese começa com um estudo teórico do efeito que a distãncia de mediçãp e a direção de magnetização tem no mapeamento dos códigos QR. Para tal fim, simulações foram produzidas usando o modelo do dipolo a longo alcance e métodos finitos. As simulações foram de seguida verificadas experimentalmente.

Um sistema é então proposto, constituido por camadas sobrepostas de sensores magnéticos, em conjunto com a correspondente eletrónica de multiplexamento e amplificação. O sistema é então fabricado, montado e os sensores são testados e caracterizados.

Devido à falta de tempo, um produto final funcional não foi desenvolvido no periodo de tempo alocado à tese. No entanto, o conceito foi provado, e sensores funcionais foram produzidos.

Keywords: Códigos QR, Tinta Magnética, Junções de Tuneling Magnético, Câmara Magnética, Simulações de Padrões Magnéticos 2D

Abstract

QR Codes have gained popularity in recent years and are very used nowadays in commercial activities. The current technology for detection employed is based on optical devices. The novel idea of developing a QR Code reader based on TMR sensors is explored in this thesis in order to read codes printed with magnetic ink.

The reason for this is that magnetic QR Codes would provide the advantage of transmitting secret concealed information since the magnetic ink used could be invisible or printed beneath the surface. This would ave huge advantages for security systems and for counterfeiting measures. For example, in the case of monetary banknotes, they could contain magnetic QR Codes, which would help identify the real ones from the counterfeited.

The thesis begins by a theoretical study of the effect of the reading distance and magnetization direction on the QR Code mapping. In order to achieve this, simulations are produced using the long-range dipole model and the finite elements method. The results are afterwards experimentally confirmed.

A system was then proposed, consisting of layered sensor stack, together with the corresponding multiplexing and amplification electronics. The system was then fabricated, assembled and the sensors tested and characterized.

Due to the lack of time, a final functional product was not achieved within the time-frame of the thesis. However, the concept was proved and functional sensors were produced.

Keywords: QR Codes, Magnetic Ink, Magnetic Tunnel Junctions, Magnetic Camera, 2D Magnetic Pattern Simulation

Contents

List of Tables

List of Figures

Nomenclature

AFM	Anti-Ferromagnetic
FM	Ferromagnetic
GMR	Giant Magnetoresistance
TMR	Tunnel Magnetoresistance
SV	Spin Valve
MR	Magnetoresistance
MTJ	Magnetic Tunnel Junction
PCB	Printed Circuit Board
PM	Permanent Magnets
PP	Pole-Pitch
SAF	Synthetic Anti-Ferromagnetic
SNR	Signal-to-Noise Ratio
VSM	Vibrating Sample Magnetometer
RA	Resistance Area
QR	Quick Response
BCH	Bose–Chaudhuri–Hocquenghem
1D	One-Dimensional
2D	Two-Dimensional
TR	Threshold
GND	Ground
VCC	Voltage Common Collector
AMP	Amplifier
VOUT	Voltage Output
MUX	Multiplexer
DWL	Direct Write Laser
DAC	Digital to Analog Converter

Chapter 1

Introduction

1.1 Barcodes

Figure 1.1: Typical example of a UPC-A Type Barcode (12-digit barcodes used in product tracking in stores and commerce - typical dimensions $3.73cm \times 2.59cm^2$)

A Barcode is an optical, one-dimensional, machine- readable, linear code represented some data in a pattern of parallel lines of varying widths and spaces. The data normally describes the object with which the barcode is attached. A typical example is shown in figure 1.1.

Linear one-dimensional barcodes using parallel lines with varying width and spacing between them were the first ones developed and are widely used in commerce and trade, with the information stored along one direction only, as shown in the figure. The reading is normally done using optical barcode readers using a narrow beam of light, along its length. [1]

1.1.1 Historical Background

The barcode patented in 1951 by Norman Joseph Woodland and Bernard Silver and had a linear and a circular design, but only became commercially wide-spread in the 1970s when they became used in automating supermarket checkouts. The barcode design developed by George Laurer in 1973 with vertical linear bars became the standard. With time they became used more generally in Automatic Identification and Data Capture (AIDC). [1]

1.1.2 Commercial and Industrial

Barcodes have a variety of uses worldwide. UPC barcodes are pre-printed on most items available in stores and supermarkets enabling quick check-outs and also allows track items. Barcodes encoding book's ISBN are also widely pre-printed on books, journals and other printed materials. Membership

cards in commerce also use barcodes to identify customers, allowing product discounts or marketing offers to specific clients. [1]

They are also used in the healthcare and hospitals, allowing patient identification in order to access patient data and medical history, and also for medication management. They can also be used for indexing of documents, tracking the organization of species in biology, and identifying being weighed in a conveyor line for data collection, for example.

Besides direct applications, they can also be used to keep track of objects and people, for example rental cars, airline luggage, nuclear waste, registered mail, express mail and parcels. Barcoded tickets enable access to sports arenas, cinemas, theatres, fairgrounds, and transportation, and are used to record the entrance and departure of vehicles from rental facilities and are also in scanning work orders and tracking the time spent on a job.

In industrial environments, barcodes are also used in non-contact 1D and 2D position sensors, allowing the tracking of items and parts during production or storage. Besides sales and inventory tracking, barcodes are very useful in logistics and supply chain management.

1.2 QR Codes

Figure 1.2: Typical QR Code Example - Level 3 (29×29 pixels) - Typical size of $2 \times 2cm^2$ that is readable using smartphone cameras

QR Code - Quick Response Code - is a two-dimensional, machine-readable, matrix barcode where data is encoded in both vertical and horizontal directions in the form of an image, consisting of black and white modules which represent the encoded data. It is the most popular 2D barcode commercially used.

Matrix Codes or 2D Barcodes are analogous to 1D barcodes, with the advantage of storing a much larger amount of information, proportional to the surface instead of the length as in the one-dimensional case. A typical example is shown in figure 1.2.

For example, when comparing a typical $Level3$ (841 pixels) QR Code of $2 \times 2cm^2$ that can be scanned using a smartphone camera (as shown in figure 1.2), with a typical $3.73cm \times 2.59cm^2$ barcode with 12 digits (as shown in figure 1.1), the pixel density for the QR Code is of $210.25pixels/cm^2$, while the information density of the barcode is of $1.24digits/cm^2$.

Nowadays, QR codes normally contain data for a locator, identifier, or tracker that points to a website or application, but can also be used in industry or commerce for the tracking of products and packages. It uses four standardized encoding modes - numeric, alphanumeric, binary and kanji to store data.

This system of transmitting information became widespread because it allows fast readability just like a normal barcode, but contains a much larger amount of information than 1D UPC barcodes.

A QR code consists of black squares arranged in a square grid on a white background, and is normally read by an optical device - optical sensor or smartphone camera, and processed using Reed–Solomon error correction until the image can be appropriately interpreted. Information is extracted from patterns that are present in both horizontal and vertical components of the QR code. [2]

1.2.1 Historical Background

It was initially invented in 1994 for the automotive industry in Japan by Masahiro Hara for the company Denso Wave to track vehicles and components during manufacturing.

1.2.2 Commercial and Industrial Applications

QR Codes have widespread applications nowadays, such as product tracking, item identification, time tracking, transport ticketing, document management and general marketing. [2]

QR code generating soft-wares are widely available nowadays since the QR codes have become one of the most popular type of 2D barcodes and can be found in most smartphones nowadays. Millions of scans of QR codes or barcodes are done every month by smartphone users at home or at stores.

Some examples of applications of QR Codes are:

- **Commercial tracking applications for smartphones** - mobile tagging - by using the smartphone camera as an optical reader. It allows the user to visualize text of the information contained within the code, or to open a web-page or to add a vCard contact to the smartphone, to open a Uniform Resource Identifier (URI), to connect to a wireless network, or to compose an email or text message or to display multimedia content in video or audio format.

- **Advertising** - Allows the user to scan the code with a smartphone camera giving direct access to a website with information or a discount for a product or a store, or to order online in a restaurant for example. This is an area where the utilization of QR Codes is rapidly increasing, and can be found nowadays even in magazines, street advertisement signs, buses, business cards, and so on, allowing the user to order products from ”Virtual Stores” directly to their homes. It can also be used to track where the code has been scanned for marketing statistics.

- **Currency** - In 2011 the Royal Dutch Mint issued the world’s first commemorative coin with a QR code that allows access to a website with information about it. In 2014 the Central Bank of Nigeria produced the first centennial banknote with an incorporated QR code. Magnetic QR codes could provide a counterfeiting for banknotes in the future since they are not optically detectable.

- **Credit Card Applications** - In September 2016 the Reserve Bank of India the Bharat QR that allows the capability of accepting payments on the unified payments interface (UPI) platform. Systems such as this provide a safe way of performing mobile payments. In 2013 the European Payment Council defined the guidelines for electronic payments using QR codes within the Eurozone.

- **Augmented Reality Systems** - Some of these systems use QR Codes to obtain the 3D position of objects.

- **Security** - QR are being used by companies and governments to allow users to verify the authenticity of the products and avoid counterfeiting, although optical QR Codes can easily be reproduced and counterfeited. This is a field where Magnetic QR Codes show promising applications since they cannot be easily detected or scanned.

- **Product Tracking and Packaging** - Just as typical Barcodes, QR Codes can be used during the industrial fabrication or assembly processes to keep track of the components. Magnetic QR Codes can be used in this field in the case of state-classified items to keep certain information secret.

1.2.3 QR Code Analytics

QR Code Analytics is termed as the overall data statistics determined through tracking of QR Codes to identify the device from which the QR Code has been decoded, geographic location and also the time during which it was scanned. It also demonstrates the traffic generated i.e., the number of visitors and requests. [2]

1.3 Objectives

The objective of this project is the development of a magnetic camera constituted by a matrix of magnetoresistive sensors, with the objective of scanning magnetic barcodes and QR codes, developing and optimizing the multiplexing electronics system and the user interface and testing on samples printed with magnetic ink.

The project starts with the analytical simulation of magnetic fields produced by the magnetic ink media. The next step consists of designing and fabricating magnetic sensors capable of scanning barcodes and QR codes, together with the corresponding electronics. The pitch-size distance between individual sensors and the magnetic interference between pixels limits the minimum size of readable magnetic patterns.

The system is to be tested in laboratory environment. The real time magnetic field mapping is to be done by integrating the sensor matrix within a scanning system or by taking a singular snapshot of the sample. The scanning process will be tested and optimized using a system of permanent magnets attached to the scanning head in order to magnetize the sample printed with magnetic ink. Software reading is to be done using the Matlab Interface via a micro-controller used to control the electronics linked to the sensor setup.

1.4 Motivation

The main motivation behind developing a magnetic camera and using magnetic ink as opposed to the mainstream available optical systems is the capacity that this technology would have in transmitting secret data only detectable via magnetic sensors. Another possible utility is in the fact that in an industrial environment with particles present in the air, if the surface of a magnetic QR Code were to be contaminated with dirt, it would not affect the reading, unlike the case of the optical QR Codes.

Visible superficial or in-depth invisible magnetic elements printed on paper are used in security paper or banknotes for security reasons to avoid counterfeiting and can be detected using static magnetic sensors. The type of letter used and pattern of magnetic bits are subjected to the ISO standards. This allows the introduction of new security elements that do not overlap with the previous ones. There is a search and need in the global market for novel secret magnetic marking solutions.

This is a novel technology and would also bring prestige to the hosting institution - INESC-MN - as being one of the pioneers in the development of this technology. This project is also done in cooperation with industrial partners of INESC-MN that are applying this technology in the tracking of packages or mass production industry.

1.5 State of the Art

1.5.1 Optical Scanners

The current devices widely used in industry are based on optical scanners. These optical systems consist of a light source, a focusing lens and an optical sensor that detects the reflected image. They also contain electronics that decode the information and send it as a digital output. [1]

There are several types of readers, such as pen-type readers that are used to scan the samples manually, or industrial scanners using lasers as light sources. The detection in these devices is normally done using a photo-diode that detects the lights reflected by the bar-code. More recent technologies employ CCD sensors that read the whole bar-code all at once. The smartphones may also be used nowadays to scan bar-codes and QR Codes, since their camera can be used to take pictures of the QR Codes and software is used to detect the code within the picture via image analysis.

1.5.2 Magnetic Scanners

Magnetoresistive sensors - Spin-Valves(SV) employing the Giant Magnetoresistive (GMR) Effect or Magnetic Tunnel Junctions(MTJ) employing the Tunneling Magnetoresistive (TMR) Effect - are used in the electronics industry, in magnetic storage systems such hard-disks, in security systems and in biomedical devices. These type of sensors are very sensible, capable of high resolution and detecting very weak magnetic fields created by magnetic ink.

The controlling factor in the sensitivity of these sensors is the demagnetization parameters of the detecting layers and linearization can be done using exposure to external bias fields during the fabrication or annealing processes, or during the detection process itself. The spatial resolution is related to the pitch-size distance between individual sensors.

Within the context of the INESC-MN Institute, technologies capable of scanning 1-Dimensional Barcodes have already been developed, as shown in the reference ([3]). The technologies developed employ Tunneling Magnetoresistive Sensors (TMR) and electronics developed by industrial partners. Permanent magnets attached to the scanning head are used to magnetize the samples printed in magnetic ink.

Simulations of the magnetic fields produces by 1D Barcodes are also presented in the referred thesis. These simulations assume the approximation that the bars within the barcodes are infinite along their height, thus producing a simplification in the calculations due to geometrical symmetry.

Chapter 2

Theoretical Background

In this Chapter, the general concepts involving magnetism in vacuum and magnetic materials are described, with special attention to ferromagnetic substances such as the particles present in magnetic ink. The technology involved in Magnetic Sensors is then presented. The several types of sensors are described and the TMR sensors used in this thesis are explored in greater detail.

Finally the structure of QR Codes and the data screaming and encoding methods are described, together with the Solomon-Reed Algorithm used for error correction.

2.1 Magnetic Fields in Vacuum

Magnetic Fields in vacuum are produced by moving charges. ([4] , [5]) The non-relativistic expression of the magnetic field vector $\vec{B}$ is given by:

$$\vec{B}(t) = \frac{\mu_0}{4\pi} \frac{q\vec{v} \times \vec{e_r}}{r^2(t)} \qquad (A \cdot m^2) \tag{2.1}$$

- μ_0 - Magnetic Permeability q - Charge Value (Coulomb) $\vec{v}$ - Velocity Vector $\vec{e_r}$ - Unitary vector source-field $r(t)$ - Distance source-field

From this expression, the magnetic field created by an element containing electric current can be obtained:

$$d\vec{B}(t) = \frac{\mu_0}{4\pi} \frac{I\vec{dl} \times \vec{e_r}}{r^2} \tag{2.2}$$

- μ_0 - Magnetic Permeability of Vacuum I - Current (Ampere) $\vec{dl}$ - Unitary Length Vector $\vec{e_r}$ - Unitary source-field vector $r(t)$ - Distance source-field

The expression above is known as the Biot-Savart Law. This results in magnetic field perpendicular to the plane formed by the unitary length vector and the unitary source-field vector and with an intensity inversely proportional to the distance, in a circular closed loop. By integrating along the path, the total magnetic field can be obtained:

$$\vec{B} = \int d\vec{B} = \frac{\mu_0 I}{4\pi} \int \frac{\vec{dl} \times \vec{e_r}}{r^2} \qquad (A \cdot m^2) \tag{2.3}$$

This is the Integral Biot-Savart Law. A more elegant way to calculate the magnetic field is using the Amp´ere Law, which states that a the integral of a closed loop around a current I has a fixed value:

$$\int_C \vec{B} \cdot \vec{ds} = \mu_0 I \tag{2.4}$$

If there are several currents or a continuous media traversing the closed loop, it results in:

$$\int_C \vec{B} \cdot \vec{ds} = \mu_0 \cdot \sum_{i=1}^{n} I_i \qquad \int_C \vec{B} \cdot \vec{ds} = \mu_0 \cdot \int_S \vec{j} \cdot \vec{dS} \tag{2.5}$$

2.2 Magnetic Fields in Matter

Experiment and theory have shown that all substances become magnetized when exposed to an external magnetic field. The effect of magnetization of matter is the change of the magnetic field, in the interior of the substance and in the space around it. ([4] , [5])

The total magnetic field $\vec{B}$ is given by the superposition of the internal magnetic field $\vec{B_m}$ of the media and of the external magnetic field $\vec{B_0}$:

$$\vec{B} = \vec{B_0} + \vec{B_m} \qquad (N/(A \cdot m)) \tag{2.6}$$

Specifically, the internal magnetic field $\vec{B_m}$ depends on the magnetic behaviour and properties of the material.

Magnetization Intensity

In the case of matter, the source of magnetic fields is the movement of charges in conducting matter and of the movement of electrons in atoms and molecules in isolating or dielectric materials, as well as the spin properties intrinsic to the quantum mechanic properties of the orbital inhabited by the electrons. The atomic or molecular magnetic moment $\vec{m}$ is given by a combination of the orbital and spin magnetic moments:

$$\vec{m} = -\gamma_L \vec{L} - \gamma_S \vec{S} \qquad (A \cdot m^2) \tag{2.7}$$

- $\gamma_L = \frac{e}{2m_e}$ - Gyromagnetic Factor $\vec{L}$ - Orbital Angular Momentum $\vec{S}$ - Spin Angular Momentum

By summing the magnetic moments of each one of the magnetic moments $\vec{m_i} = I_i \vec{S_i}$ the orbitals, we can obtain the total magnetic moment of the atom or molecule:

$$\vec{M_i} = \sum_{i=1}^{Z} \vec{m_i} = \sum_{i=1}^{Z} I_i \vec{S_i} \qquad (A/m) \tag{2.8}$$

By summing the magnetic moments of all atoms or molecules within the substance, we can obtain the total magnetic moment:

$$\vec{M_t} = \sum_{i=1}^{N} \vec{m_i} = \sum_{i=1}^{N} I_i \vec{S_i} \qquad (A \cdot m^2) \tag{2.9}$$

By dividing by the volume of the substance we can obtain the Volume Magnetic Moment or Magnetization Vector of the material:

$$\vec{M} = \frac{\vec{M_t}}{V} = \frac{\sum_{i=1}^{N} \vec{m_i}}{V} \qquad (A/m) \tag{2.10}$$

Induction field H

When a sample is exposed to an external bias field $\vec{H}$, the resulting magnetic field $\vec{B}$ is a superposition of the external field and the internal magnetization field $\vec{M}$ of the matter:

$$\vec{B} = \mu_0(\vec{H} + \vec{M}) \qquad (N/(A \cdot m)) \tag{2.11}$$

Thus the induction field $\vec{H}$ can then be defined as:

$$\vec{H} = \frac{\vec{B}}{\mu_0} - \vec{M} \qquad (A/m) \tag{2.12}$$

This allows the simplification of Ampere's Law:

$$\int_C \vec{H} \cdot \vec{ds} = I_{free} \qquad (Ampere) \tag{2.13}$$

- $I_t(A)$ - Total free currents within the closed loop C

Thus it can be stated that the sources of the induction field $\vec{H}$ are only the free currents, while the sources of the magnetization field $\vec{M}$ are only the internal magnetization currents within the matter. In the case of the magnetic field $\vec{B}$, it is a superposition of both, so it results from both the free currents and the internal magnetization currents of matter. These three field quantities are used to characterized general magnetic phenomenons and interactions.

Magnetic Susceptibility

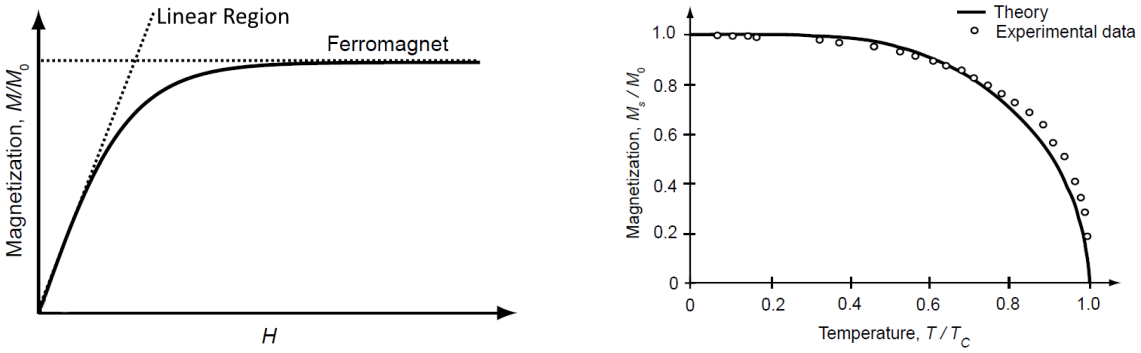


Figure 2.1: Typical Magnetization Curve for Ferromagnetic Materials - Left, Curie-Weiss Law Diagram (showing a typical fit between the theory and experiment) - Right

When an external bias field $\vec{H}$ is applied to a substance, the internal magnetization reacts aligning itself parallel or anti-parallel to the field by a factor χ_m, in a linear way, until it reaches the saturation magnetization: ([6] , [7])

$$\vec{M} = \chi_m \vec{H} \tag{2.14}$$

The total resulting field is then:

$$\vec{B} = \mu_0(\vec{M} + \vec{H}) = \mu_0(1 + \chi_m)\vec{H} = \mu_0\mu_r\vec{H} = \mu\vec{H} \tag{2.15}$$

Where total magnetic permeability μ is defined as a product of the relative magnetic permeability μ_r

and magnetic permeability in vacuum μ_0:

$$\mu_r = (1 + \chi_m) \qquad \mu = \mu_0 \mu_r \tag{2.16}$$

As the temperature increases, the thermal agitation counteracts the parallel alignment of internal field $\vec{M}$ of paramagnetic substances, by a factor proportional to the inverse of the temperature T multiplied by the Curie Constant C:

$$\chi_m = \frac{C}{T} \tag{2.17}$$

2.2.1 Diamagnetic Substances

In the case of diamagnetic substances, the total atomic or molecular magnetic moment in the absence of external magnetization is zero:

$$\vec{M}_i = \sum_{i=1}^{Z} \vec{m}_i = 0 \tag{2.18}$$

If an external bias field $\vec{H}$ is applied, the internal magnetization field $\vec{M}$ ceases to be zero and aligns itself anti-parallel to this field, producing a field $\vec{B}$ weaker than in the absence of the material.

2.2.2 Paramagnetic Substances

Similar to diamagnetic substances, in the case of paramagnetic substances, the total atomic or molecular magnetic moment in the absence of magnetization is zero.

If an external bias field $\vec{H}$ is applied, the internal magnetization field $\vec{M}$ aligns itself parallel to this field, producing a field $\vec{B}$ stronger than in the absence of the material.

2.2.3 Ferromagnetic Substances

Ferromagnetic substances are a special type of paramagnetic substances in which the magnetization $\vec{M} \neq 0$ arises from the formation of local microscopic magnetic domains within the material known as Weiss domains. The different domains are separated by the domain walls known as the Bloch Walls. These magnetic domains appear spontaneously in the material, and when exposed to an external magnetic field the size of the domains change and the Bloch Walls move so that the global magnetization of the material aligns itself with the external bias field. ([6] , [7])

This phenomenon of rearranging the micro-magnetic domains under the influence of an external field results in what is known as the ferromagnetic hysteresis loop shown in figure 2.2.

The magnetization M_s is known as the saturation magnetization which represents the maximum possible magnetization of the sample under influence of the external bias field, while M_r is the remnant magnetization of the material after the external field is not applied anymore. H_c is known as the coercive field, which is the magnitude of the required external bias field in order to cancel the internal magnetization M of the material.

If a ferromagnetic material has a large hysteresis loop, it takes a lot of energy to switch the magnetization of the material, so it is classified as being hard magnetic. If the hysteresis loop is small, the energy required for magnetic switching is low, so the material is considered soft magnetic.

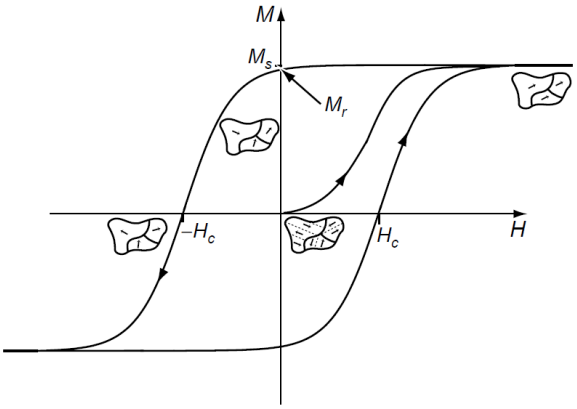


Figure 2.2: Typical Ferromagnetic Hysteresis Loop, showing the characteristic points of the Coercive Field H_c, Remnant Magnetization M_r and Saturation Magnetization M_s

2.3 Long-Range Magnetic Dipole Model

At a large enough distance from a magnetic dipole (an order of magnitude above the magnitude of the dipole), it can be considered punctual, which the magnetic potential being given by:

$$\psi(\vec{r}) = \frac{\vec{m} \times \vec{r}}{4\pi r^3} \tag{2.19}$$

The resulting magnetic field created by each pixel at a position vector $\vec{r}$ relative to it is given by: ([8] , [9])

$$\vec{H} = -\Delta\psi(\vec{r}) = \frac{1}{4\pi}\left[\frac{3\vec{r}(\vec{r}\cdot\vec{m})}{r^5} - \frac{\vec{m}}{r^3}\right] \qquad (A/m) \tag{2.20}$$

- $\vec{m}(A \cdot m^2)$ - Finite Element Magnetic Moment r - Finite Element Distance (m)

This expression was used in the simulations shown in chapter 3.

2.3.1 Validity of the Model

This model is valid under the assumption that the magnetic dipoles of the magnetic material can be considered approximately punctual at the measuring distance, so it has to be verified if this is the case.

The dipole size is related to the exchange length of the material, given by:

$$L_{exc} = \sqrt{\frac{A}{\mu_0 M_s}} \approx 3nm$$

- A - Exchange Stiffness of the material
- M_s - Magnitude of the magnetization

Given that the typical measuring distances for a sample is of the order of $\approx 500\mu m$, which is 5 orders of magnitude above the typical exchange length of $\approx nm$, the model is valid under these assumptions.

2.4 Magnetoresistive Sensors

Magnetoresistive sensors are sensors that detect an external magnetic field trough a change in their magnetoresistance (MR). There are three main types of magnetoresistive sensors. Anisotropic Magnetoresistive Sensors (AMR) use phenomenons based on the anisotropy of the material in order to detect the magnetic field. Giant Magnetoresistive Sensors (GMR) employ alternating layers of ferromagnetic and non-magnetic materials. Tunneling Magnetoresistive Sensors (TMR) employ the quantum mechanical tunneling effect trough a thin oxide layer between two ferromagnetic layers.

Of these three categories, the TMR sensors are the most sensitive and have the highest percentage of magnetoresistance of the order of 175%, as shown in [10] , [11]. Hence these will be the type of sensors used in this thesis, since the magnetic fields produced by magnetic ink are very weak, and the highest possible magnetoresistance is required for detection.

2.4.1 Magnetic Tunnel Junctions

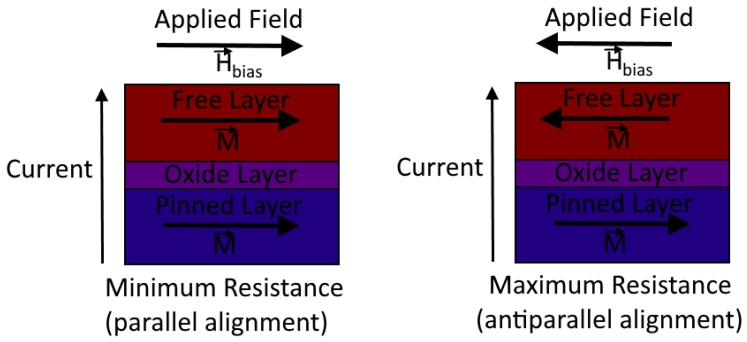


Figure 2.3: General Functionality of a typical TMR Junction, showing the minimum and maximum resistance configurations, respectively

Magnetoresistive Sensors are generally constituted by two ferromagnetic layers separated by a thin oxide layer, as shown above. One of the layers is a hard magnet and maintains a fixed magnetization. The other layer is a soft magnet that aligns itself with the external bias field. The relative orientation of the magnetization between these two layers determines the resistance of the junction. When the alignment is parallel, the resistance reaches a minimum. When the alignment is anti-parallel, the resistance reaches a maximum.

When two conductors are separated by a thin oxide layer of the order of 5-10nm, tunneling occurs when these two conductors are exposed to a voltage differential that results in a shift of the Fermi Levels. This quantum mechanical effect can be described using the WKB (Wentzel–Kramers–Brillouin) Approximation. The following expression for the current density as a function of the applied voltage, oxide thickness can be obtained:

$$J(A/m^2) = J_L[(\phi - \frac{V}{2})e^{-b\sqrt{\phi - \frac{V}{2}}} - (\phi + \frac{V}{2})e^{-b\sqrt{\phi + \frac{V}{2}}}] \tag{2.21}$$

- $J_L = \frac{e}{2\pi h s^2}$ $\quad b = \frac{4\pi s}{h}\sqrt{2m}$ $\quad s$ - barrier thickness $\quad \phi$ - Band Gap E_g (oxide)

When the electrons perform tunneling from one ferromagnetic layer to another ferromagnetic layer with a different overall spin orientation, it is suddenly exposed to a phenomenon known as spin-transfer torque, resulting in an increase in resistivity.

Ideal Transfer Curve and Linearization Methods

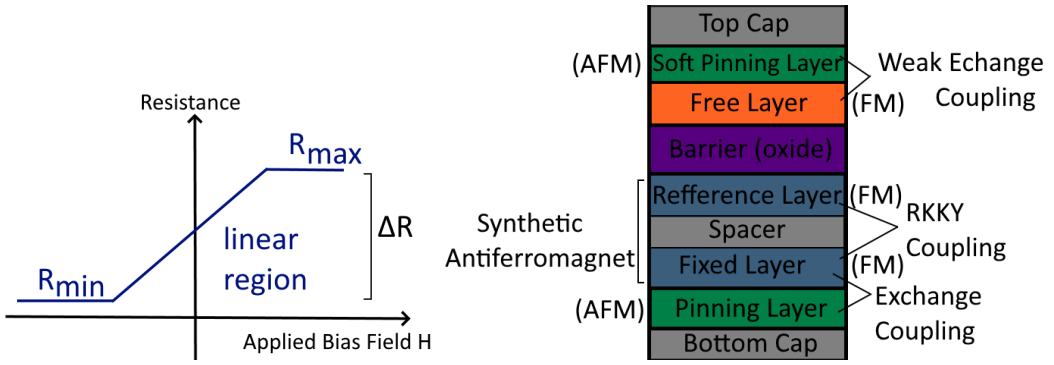


Figure 2.4: Ideal linear transfer curve (Left) and Multilayer interactions that guarantee linearization (Right)

An ideal transfer curve is shown in the figure 2.4, with the resistance varying linearly between the saturation values R_{min} and R_{max}. The slope of the curve in the linear region is given by:

$$MR(\%) = \frac{\Delta R}{R_{min}} \times 100 \qquad \Delta R(Ohm) = R_{max} - R_{min} \tag{2.22}$$

Due to several phenomenons, such as the presence of demagnetizing fields within the materials, this is not easy to obtain. The energy of the system is given by the expression, in the case of parallel orientation between the ferromagnetic layers:

$$\frac{E}{V_{FL}} = \mu_0 M_s \left[\frac{\sin^2\theta}{2}(H_k - N_h M_s) + \frac{N_h M_s}{2}\cos\theta(H_{ext} - H^d_{red} - H_N) \right] \tag{2.23}$$

- H_{ref} - magnetic field of the reference layer (Oe)
- H_K - anisotropic field
- N_H - demagnetizing factor
- H_{ext} - external applied bias field (Oe)
- M_S - internal magnetization field (emu/cm^3)
- θ - relative angle between free and pinned layer

This model is obtained from the Stoner-Wohlfarth Model. The system aligns itself in such a way as to minimize this energy. This is influenced by the several contributions shown above. In the case of perpendicular orientation, the energy is given by:

$$\frac{E}{V_{FL}} = \mu_0 M_s \left[\frac{\cos^2\theta}{2}(H_k + N_h M_s) - \cos\theta(H_{ext} - H^d_{red} + H_N) \right] \tag{2.24}$$

In order to achieve a linear behaviour as close to this figure as possible, several auxiliary layers needs to be positioned beneath or above the ferromagnetic (FM) layers.

In the case of the free layer, an anti-ferromagnetic layer pins this layer softly towards a certain direction trough a weak exchange coupling. In the case of the fixed layer, it forms a Synthetic Antiferromagnet with the auxiliary layer trough the RKKY (Ruderman–Kittel–Kasuya–Yosida) Coupling. Its magnetization is pinned towards a certain direction via the exchange coupling with an antiferromagnetic layer, as shown in the references [10] , [11] .

Important Parameters

Besides the Magnetoresistance MR defined in the previous section, two other parameters in the characterization of the TMR Sensors are the Sensitivity S and the Detectivity D. [10] , [11]

The Sensitivity S is given by the ratio between the variation of the magnetoresistance and the variation of the magnetic field in the linear region. It is determined and improved by employing appropriate linearization methods via a proper choosing of the magnetoresistive stack and by the optimization of the annealing process. In the type of sensors fabricated within this thesis, the expected values for the sensitivity are of the order of 6(Ohm/Oe). It is given by:

$$S(\%/T) = \frac{1}{R_{min}} \frac{\Delta R}{\Delta H} = \frac{MR}{\Delta H} \tag{2.25}$$

- ΔR - Resistance Variation (Ohm) ΔH - Magnetic Field Variation (Oe)

The Detectivity D described that smallest magnetic field that can be detected, and is related to the measuring frequency. It is described by:

$$D(T/\sqrt{Hz}) = \frac{1}{S}\sqrt{\frac{\alpha_H}{A \times f}} = \frac{\Delta H}{TMR}\sqrt{\frac{\alpha_H}{A \times f}} \tag{2.26}$$

- α_H - phenomenological Hooge's parameter (μm^2)
- f - Operating Frequency (Hz)
- A - Sensor Area (μm^2)

In general GMR sensors have a better detectivity than the TMR ones. In the case of TMR Sensors, the detectivity for practical sensors can go down to the μT range and pT range in ideal conditions. Since the magnetic field values measured are of the order mT, the detectivity of the TMR should not be a limiting factor during measurements.

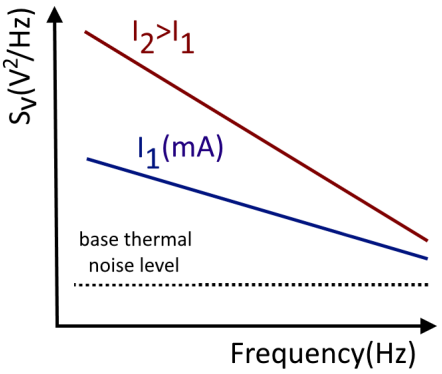


Figure 2.5: Typical Noise Spectrum Density of an MgO MTJ as function of the applied bias current

Another important aspect to consider is the noise spectrum of the sensors. The noise level depends on the frequency of the current applied to the sensor [10]. It decays in a linear fashion as shown in figure 2.5. There are three main contributions: the constant Base Thermal Noise level, the Shot Noise and the Electronic Noise.

The Shot Noise appears in MTJ junctions since there is a discontinuity in the conductive media due to the presence of the oxide barrier. As the applied bias current increases, so does the contribution of the shot noise. The Thermal Noise results for the random thermal agitation of the electrons and of the atoms within the conducting materials. This term is proportional to the temperature. The contributions contributions from these two phenomenons are independent of the applied frequency are given by 2.27.

$$S_V^{Thermal}(V^2/Hz) = 4k_BT \cdot R \qquad S_V^{Shot}(V^2/Hz) = 2eI \cdot R^2 \tag{2.27}$$

- S_V - Noise Spectrum Density (V^2/Hz)
- k_B - Boltzmann Constant ($m^2kg \cdot s^{-2}K^{-1}$)
- T - Temperature(K)
- R - Resistance(Ohm)
- I - Current (A)
- e - electron charge (C)

The quasi-linear decay as a function of the frequency is caused by the electronic noise, given by the equation 2.28. This term is in reality proportional to the inverse of the applied frequency $1/f$, but can be approximated to a linear decay over certain frequency ranges.

$$S_V^{1/fMTJ}(V^2/Hz) = \frac{\alpha_H I^2 R^2}{A \times f} \sim \frac{1}{f} \tag{2.28}$$

- I-Applied Current(A)
- R - Junction Resistance (Ohm)
- f - Operating Frequency(Hz)
- A - Sensor Area (μm^2)
- α_H - phenomenological Hooge's parameter (μm^2)

TMR Sensors are better than GMR ones for noisy applications since they have a better signal to noise ratio. For the applications considered in this thesis this is one of the most important parameter, since the work is done in the low frequency/DC regime, where the noise is larger than for setups operating at higher frequency. Hence the utilization of TMR Sensors is more justified than of GMR ones. The expression for the signal-to-noise ratio SNR of MTJs in the regime where thermal noise is dominant is given by equation 2.29, with the last equivalence valid at low bias voltages.

$$SNR_{MTJ} = TMR_0 \frac{2V_{1/2} - R \cdot I_b}{2V_{1/2}} \sqrt{\frac{R \cdot I_b^2}{4k_B T \Delta f + 2eI_b R \Delta f}} \sim \sqrt{I_b} \tag{2.29}$$

- $V_{1/2}$ - Bias Voltage (V)
- I_B - Bias Current (A)
- T - Temperature(K)
- R - Resistance (Ohm)
- Δf - Applied Frequency Range (Hz)
- e - electron charge (C)
- TMR_0 - Base Magnetoresistance

The Offset Field H_f from the center-line and the Coercive Field H_c, which is the bias field required to cancel the internal magnetization are also important parameters [11], given by the relations 2.30.

$$H_c(Oe) = H_k - N_s M_s^{sen} \qquad H_f(Oe) = H_d^{ref} - H_N \tag{2.30}$$

- H_k - Anisotropic Field (Oe)
- N_s - Neel Coeficient
- M_s - Internal Magnetization Field (emu/cm^3)
- H_N - Neel Coupling Field (Oe)
- H_d^{ref} - Demagnetizing Field of the Reference (Oe)

Another important parameter that is a constant for a given MTJ stack is the resistance $\times$ area product RA, with values of the order of $\sim kOhm \cdot \mu m^2$. given by equation 2.31.

$$RA(Ohm \cdot \mu m^2) = R \times A \tag{2.31}$$

- R - Junction Resistance(Ohm)
- A - Junction Area(μm^2)

2.5 QR Codes

In the Following Sections the theoretical background behind the structure and encoding of QR Codes is described in detail.

2.5.1 QR Code General Layout

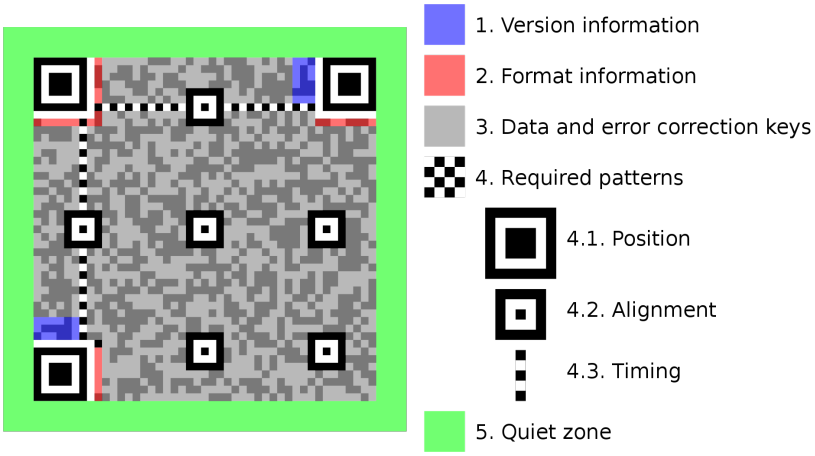


Figure 2.6: Standard QR Code General Layout, showing the various regions and their functionalities

The standard structure of QR Codes is shown in the figure 2.6. The reading is normally done by a 2D digital image sensor and digitalized by a programmed processor, that uses the three large positional squares at the edges and a smaller one in the free corner to calibrate the positioning, size and orientation of the scanned image. The other small squares contain information and are converted to binary code and run trough error-correcting algorithms.

There are 40 versions of QR codes, with the larger versions containing a larger amount of information. The amount of information stored also depends on the type of data used and level of error correction. For example, for the version 40 using a low error correction level (40-L), we can have up to 7089 numerical characters, with 3,3 bits per character. If besides numbers we also want to use letters and symbols, we can store up to 4296 alphanumerical characters, with 5,5 bits needed per character. If we want to store binary information, we can only store up to 2953 characters since 8 bits are required per character. In the case of kanji (Japanese symbols), only 1817 characters can be stored since 13 bits are required per characters. This is the largest amount of information that can be stored in a QR code using the current ISO standard. ([12] , [13])

Some examples of different versions of QR Codes are shown in figure 2.7.

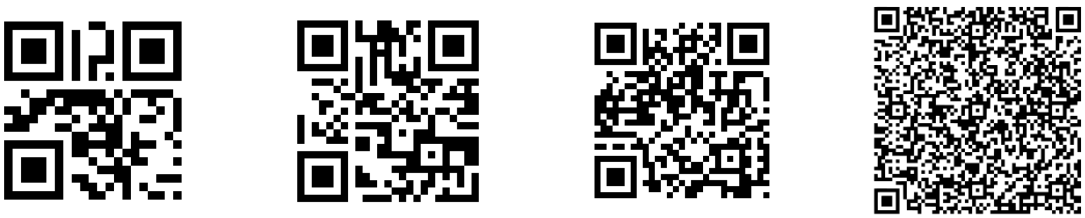


Figure 2.7: From left to right: Versions 1(21×21), 2(25×25), 3(29×29), 10(57×57) respectively

2.5.2 QR Code Structure

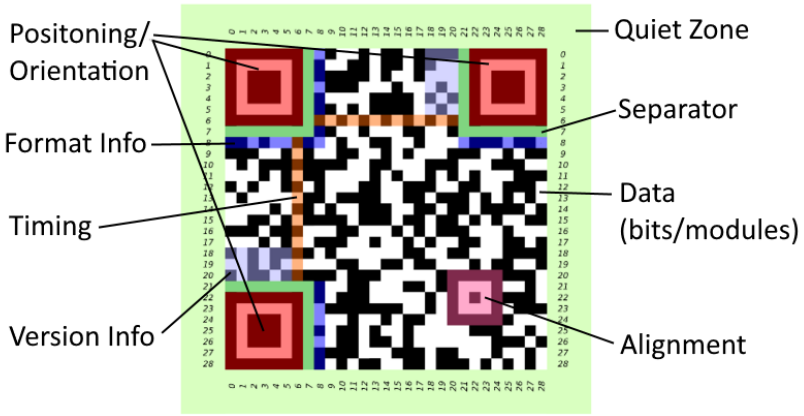


Figure 2.8: Detailed view of the different zones of a $Version3$ QR Code

The number of pixels along width or length is given by $N = 40 \cdot V + 17$ (V-version), with the total number of pixels N^2 over the area. Each pixel corresponds to a bit or $module$, with the light ones corresponding to 0's, and dark corresponding to 1's, arising from its reflectivity.

The quiet zone is an empty region surrounding the QR Code, with at least 4 modules width, as shown in figure 2.8. The three large positioning square normally have an area of 7x7 pixels and are surrounding by a separator of 8 pixels in length from the data bits in the center. The alignment square on the bottom right corner has an area of 5x5 pixels, and the number of alignment squares depends on the version, with larger version having a higher number of alignment squares (0 for version 1, and up to 46 for the version 40). There are two lines containing timing information that connect the three larger positioning squares up to the separator, and have a length (N-16). Surrounding the separator region there are 31 pixels containing information about the format, and 36 pixels containing version information (only for versions 7 and above), as shown in the figure above. The remaining pixels in the center contain the stored data. For example, in the case of Version3 shown above we have 567 pixels containing data. [2]

2.5.3 Format Information

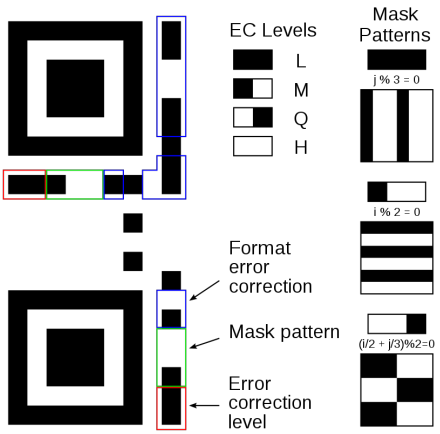


Figure 2.9: Format Information - Regions around the positioning markers

The format information region of 15 bits contains 2 bits of information about the error correction

level (L-low, M-$medium$, Q-$quartile$, H-$high$) marked with red, and 3 bits about the mask patterning of symbols (marked with green), which helps the scanner read the code, as shown in figure 2.9. These are patterns that are repeated throughout the Data region. The remaining 10 bits are used for error correction using Bose–Chaudhuri–Hocquenghem (BCH) error correcting code. [2]

2.5.4 Data Streaming and Decoding

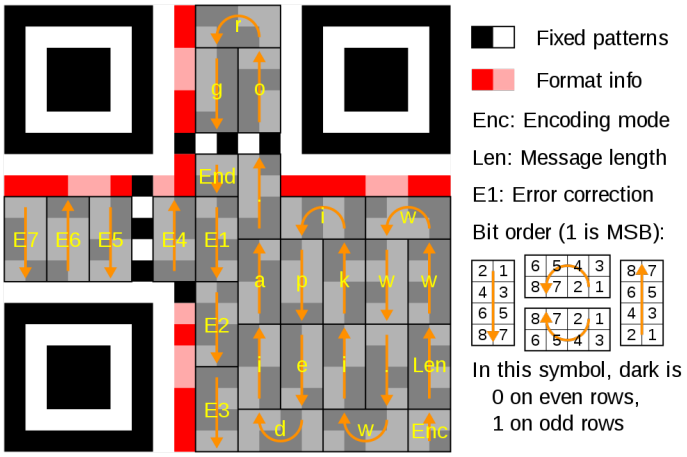


Figure 2.10: Data Streaming - $Version1$ QR Code - Reading is done from the right to the left, moving upwards and downwards cyclically

Reading order is done from right to left, starting in the right-side bottom corner and going upwards and downwards, as shown in the figure 2.10. First there is information about the encoding mode (numeric,alphanumeric,byte or kanji) and the message length, followed by the bulk of the information contained within the QR Code. Finally there is information for error correction at the end of the data stream. Information is divided into blocks of 8 bits. [2] , [14]

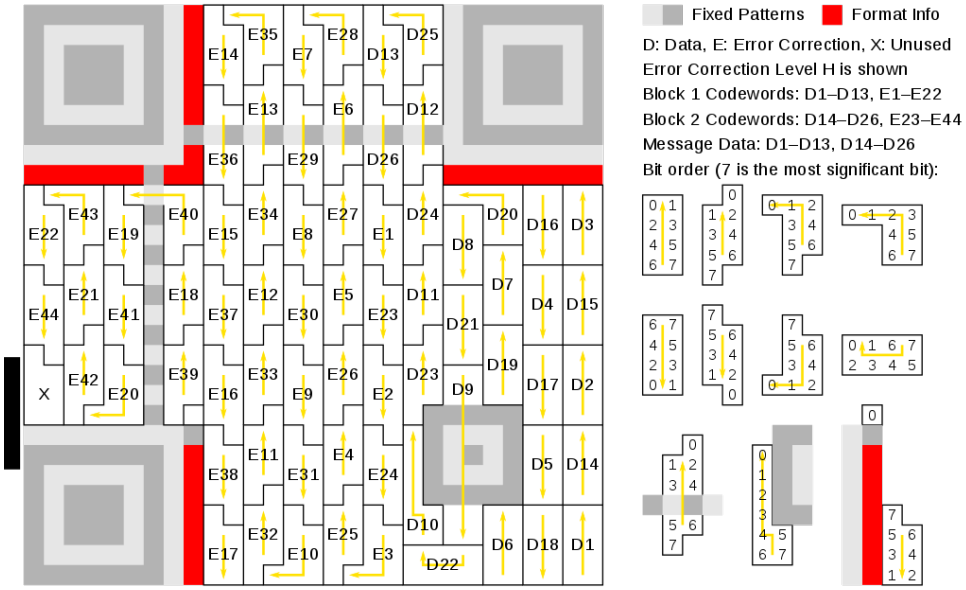


Figure 2.11: Data Streaming - $Version3$ QR Code - The code is divided into blocks of 8 bits

In larger versions such as the one shown in figure 2.11, there are alignment patterns along the data stream, and the message is broken up into several Reed–Solomon code blocks, with the bulk information (D) is alternating with error correction information (E). [14]

2.5.5 Error Correction

The Reed–Solomon method for error correction method is normally used when reading QR Codes, with information for error correction contained within the code itself. There are two main versions of the algorithm: the original one developed in 1960 by Solomon and Irvine [15], and the more modern and efficient BCH version [16].

In the original version, the message is divided into a set of symbols $x = (x_1, ..., x_k)$ which are assigned as coefficients to a polynomial p_x of degree $\leq k-1$:

$$p_x(a) = \sum_{i=1}^{k} x_i a^{i-1} \tag{2.32}$$

The sequence of values for a_i produces a set of code-words C, defined as a set of n polynomials:

$$C = \{p(a_1), p(a_2), ..., p(a_n)\} \tag{2.33}$$

Since within this code-word, two distinct polynomials of degree $\leq k-1$ (for example p(a1) and p(a2)) agree at most in $k-1$ points, they also disagree at most in distance $d = n-k+1$ points. The relative distance is then defined as the relative distance $\delta = d/n \sim 1-R$, with rate $R = k/n$. The relative distance δ and rate R satisfy the following inequality: $\delta + R \leq 1 + 1/n$.

In the BCH version of the algorithm, the message polynomial $p(x)$ of degree $k-1$ is multiplied by a generator algorithm $g(x)$ of degree $n-k$ shown in eq.2.34 (with roots α^i), producing a code-word $s(x)$.

$$g(x) = (x-\alpha^i)(x-\alpha^{i+1})...(x-\alpha^{i+n-k-1}) = g_0 + g_1 x + ... + g_{n-k-1}x^{n-k-1} + x^{n-k} \tag{2.34}$$

The resulting code-words are then given by the set:

$$C(x) = \{(s_1, s_2, ..., s_n) | s(a) = \sum_{i=1}^{n} s_i a^i \text{ polynomial with at least the roots } \alpha^1, \alpha^2, ..., \alpha^{n-k}\} \tag{2.35}$$

The transmitted messages c_i are then coefficients of the polynomial $s(x)$:

$$s(x) = \sum_{i=0}^{n-1} c_i x^i \tag{2.36}$$

The encoded message $s(x)$ is then divisible by the generator polynomial $g(x)$:

$$g(x) = \prod_{j=1}^{n-k} (x - \alpha^j) \tag{2.37}$$

Therefore is also possesses the roots of $g(x)$:

$$s(\alpha^j) = 0, \text{ j = 0,1,2,...,n-k} \tag{2.38}$$

If error is contained within the message, the resulting polynomial $r(x)$ is composed by the transmitted

polynomial $s(x)$ and error polynomial $e(x)$:

$$r(x) = s(x) + e(x) \qquad e(x) = \sum_{i=0}^{n-1} e_i x^i \tag{2.39}$$

The coefficient e_i is null if there is no error at that power of x and nonzero if there is an error. If there are n_e errors at powers i_k of x, the error polynomial is given by:

$$e(x) = \sum_{k=1}^{n_e} e_{i_k} x^{i_k} \tag{2.40}$$

The algorithm has the function of finding the number of errors (n_e), the positions of the errors (i_k), and error values at (e_{i_k}). Afterwards $e(x)$ is calculated and subtracted from $r(x)$, isolating the original encoded message $s(x)$.

At higher the error correction level there is less storage capacity since error correction occupies larger area within the code. [17] , [18], [19]. Correction capacity per level is given by:

- Level L (Low) - 7% of data bytes can be restored.
- Level M ($Medium$) - 15% of data bytes can be restored.
- Level Q ($Quartile$) - 25% of data bytes can be restored.
- Level H ($High$) - 30% of data bytes can be restored.

Chapter 3

Simulations

In this chapter, simulations of the magnetic field mapping produced by QR Codes are explored. Firstly, the proposed mathematical model is shown. Afterwards, the three main detection modes (parallel, perpendicular-in-plane, perpendicular) are shown and compared.

The effect of physical parameters such as measuring distance and number of sensors per pixel is then explored. The mapping contrast and mapping information as a function of these parameters is presented, providing an objective way for quantifying the mapping information.

Finally, a decoding algorithm is shown for the perpendicular scanning method, and applied to experimental results from a scanned QR Code. These results are then used to confirm the validity of the proposed simulation method and mathematical model, and of the proposed decoding algorithm.

3.1 VSM Measurements

Using the long-range dipole model described in the previous section, simulations of the 2D magnetic field mapping produced by QR Codes printed in magnetic ink can be obtained. In order to do this, the magnetization density per unit area of the magnetic ink is required. This can be obtained by measuring the magnetic moment of a printed sample and dividing by the area.

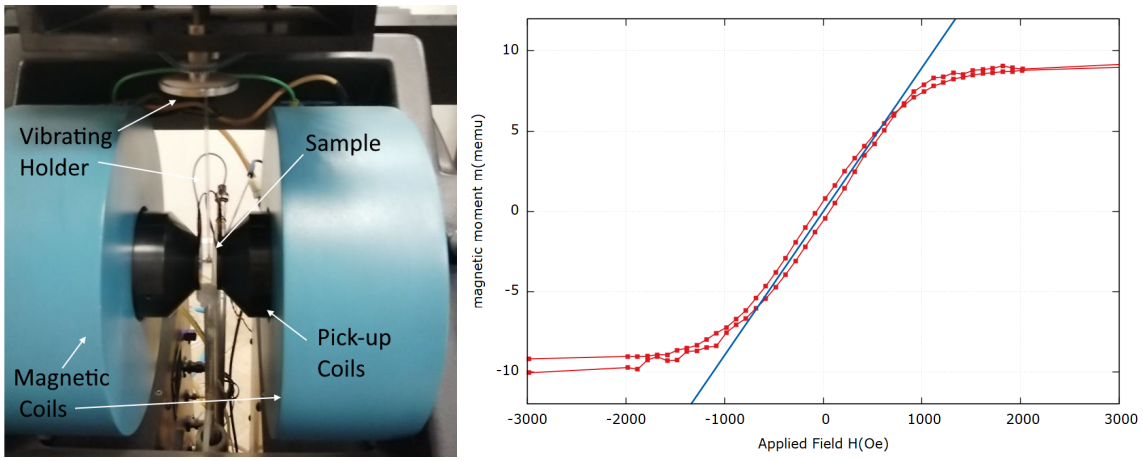


Figure 3.1: VSM Measuring Setup (Left) and Experimental Results, showing the linearity at the centerline (Right)

These measurements were done using the Vibrating Sample Magnetometer (VSM) device. A square with printed with magnetic ink and with well defined area of $25mm^2$ was placed on a holder that vibrates

between two magnetic coils. Magnetic pick-ups detect the magnetic response of the sample as a function of the applied field, thus giving the magnetic moment.

The magnetic ink considered is laser-jet ink that demonstrates magnetic properties due to the fact that it contains ferromagnetic particles. The measuring setup and the plot of the measurements are shown in figure 3.1. It can be observed that the sample shows almost no hysteresis and remnant magnetization (soft magnetic), so it needs to be magnetized by external magnets during the reading process. It is considered that the sample is completely saturated by permanent magnets attached to the scanning head, therefore the only parameter of importance to be extracted from the data is the saturation magnetization M_S (3.1), used in the following sections as the magnetization density of the sample.

Saturation Magnetic Moment	Saturation Magnetization/ u.a	Error
8.95 $memu$	0.36 $memu/mm^2$	4.8%

Table 3.1: VSM Measurements

3.2 General Method

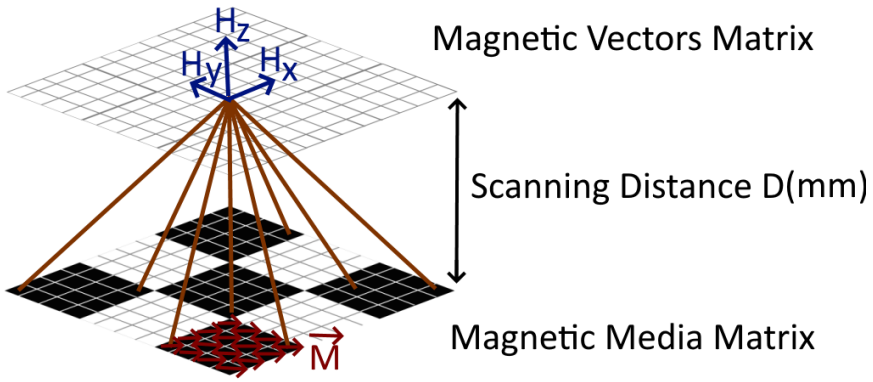


Figure 3.2: Finite Elements Method used for the calculation of the Magnetic Field Mapping of Magnetic QR Patterns

The finite elements method used for the simulations is shown in the figure 3.2. A matrix containing the magnetic pattern of the QR Code is created, with each pixel consisting of a certain number of smaller magnetic dipole elements with a certain magnetization vector $\vec{M}$.

Another matrix is created at a certain simulated measuring distance $D(mm)$. Each point in this new matrix adds the contribution of all the finite elements of the matrix beneath, using the long-range dipole approximation equation:

The field created by each pixel at a position vector $\vec{r}$ relative to it is given by: ([8],[9])

$$\vec{H} = \frac{1}{4\pi}\left[\frac{3\vec{r}(\vec{r}\cdot\vec{m})}{r^5} - \frac{\vec{m}}{r^3}\right] \qquad (A/m) \tag{3.1}$$

- $\vec{m}(A\cdot m^2)$ - Finite Element Magnetic Moment r - Finite Element Distance

With the magnetic moment of each finite element given by:

$$\vec{m} = \vec{M}\cdot dV \qquad (A\cdot m^2) \tag{3.2}$$

- $\vec{M}(A/m)$ - Volume Magnetization (material) $dV(m^3)$ - Finite Element Volume (infinitesimal)

This method thus sums all the contributions of all the finite elements contained in each pixels of the QR Code, so it accurately reproduces the interference between different pixels. This interference can be constructive or destructive. It also does not treat each pixel as an entire dipole, but as a quasi-continuous media that contains smaller magnetic domains (finite elements), thus reproducing the pixel surface.

The magnetization density from the previous section is used. It is considered that the external magnets that are used during reading magnetize the sample to the saturation value of 0.36 $memu/mm^2$. It is also considered that this magnetization is uniform throughout the simulated sample and the surface presents no roughness or printing irregularities.

3.3 Mapping Characterization

The accuracy of the mapping can be characterized by two main properties:

- Mapping Contrast
- Mapping Information

The contrast of the mapping represents the average amplitude between the maximum and the minimum peaks of the magnetic mapping. The information is related to the amount of distinguishable features present in the mapping.

The mapping contrast will be used to characterize the quality of the mapping as a function of physical parameters such as the measuring distance in the following chapters. Since this thesis is of experimental character, the information parameter was not explored since it would involve the development of more complex algorithms.

3.4 Mapping as a function of Magnetic Component

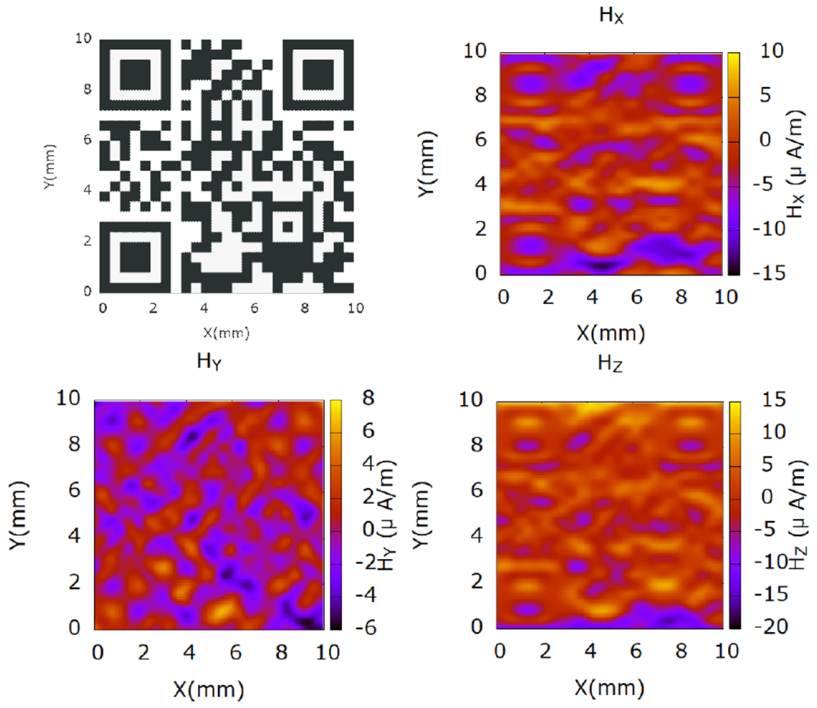


Figure 3.3: Magnetic Mapping using different components of the Magnetic Field ($D = 0.5mm$)

The mapping in this section is done using 4x4 finite elements per pixel of QR code Version 2 (25x25 pixels) with dimensions of $10mm \times 10mm$ - resulting in a simulation resolution of $R = 0.1mm$. The magnetic media magnetization is along the X-axis with the pixel thickness(along the Z-axis) of $T = 0.0001mm$ (estimated ink thickness). The simulated measuring distance is $D = 0.5mm$.

The results are shown in figure 3.3. It can be observed that the magnetic field component H_X in the direction of the media magnetization X allows the mapping of the bulk of the pixels, while H_Y maps the corners and HZ maps the left and right edges of the pixels.

3.4.1 Parallel In-plane reading (Bulk Mapping)

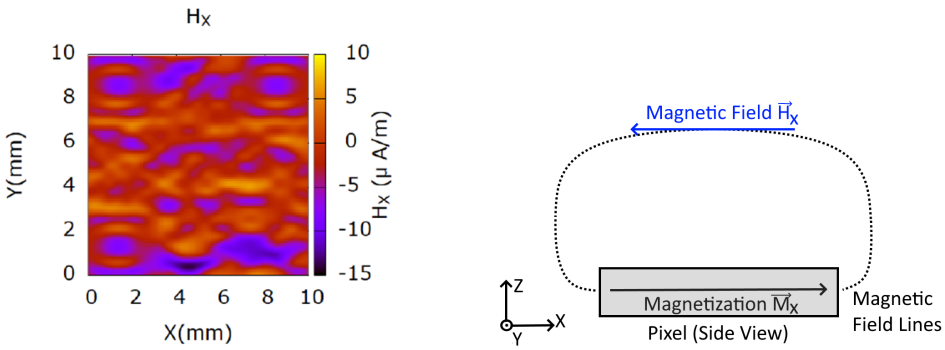


Figure 3.4: Bulk Mapping - Left, Magnetic Field Schematic - Right

In the case in which the reading the magnetic component of the field in the same direction as the magnetization is performed, a mapping of the bulk of the pixels is obtained. This is because the field forms a closed loop starting at the point of the magnetization vector and closes within itself above the body of the pixel, as shown in the figure 3.4.

3.4.2 Perpendicular In-plane reading (Corner Mapping)

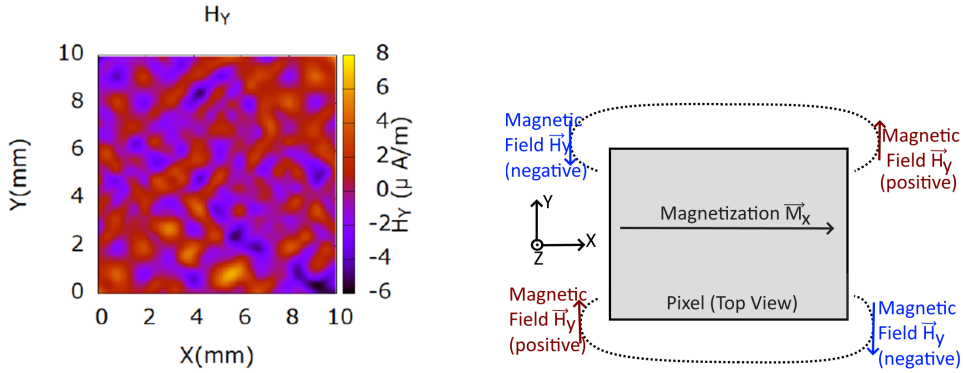


Figure 3.5: Corner Mapping - Left, Magnetic Field Schematic - Right

In the case in which the reading the magnetic component in the same plane as the pixel is performed, but perpendicular to the magnetization, a mapping of the corners of the pixels is obtained, as shown in figure 3.5.

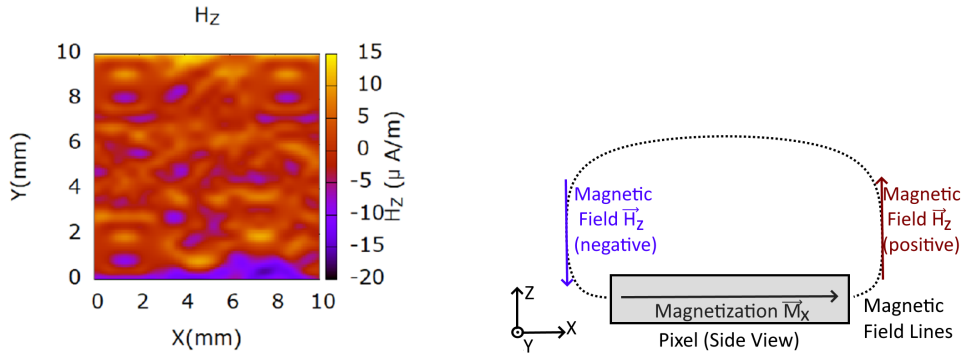


Figure 3.6: Edge Mapping - Left, Magnetic Field Schematic - Right

3.4.3 Perpendicular Reading (Edge Mapping)

When reading the component of the magnetic field H_Z perpendicular to the pixels and to the magnetization in the X direction, a mapping of the edges of the pixels is obtained, with upper edges corresponding to a positive field and lower edges corresponding to a negative field value.

3.5 Mapping as a function of Measuring Distance (Bulk Mapping)

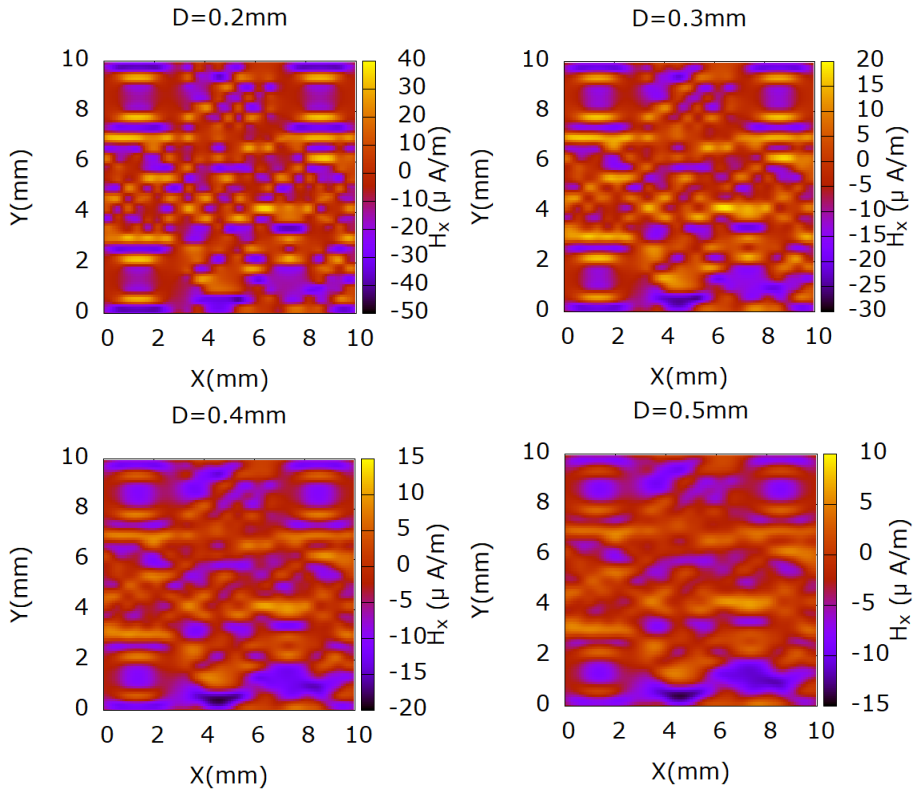


Figure 3.7: Magnetic Field H_X mapping at different measuring distances

In the figure 3.7 it can be observed that for the case of a $10 \times 10mm$ QR Code, only at measuring distances of $D = 0.2mm$ and bellow is it possible to distinguish the individual pixels properly. Hence, the recommended ratio between the measuring distance and QR Code width W should be no larger than $\frac{D}{W} = 0.02$ (2%) in order to scan and properly decode the information present within the QR Code.

For this reason, reducing the size of the QR code is limited by the practicality of reducing the measuring distance for actual measurements. Based on this it is possible to build a table with the maximum recommended measurement distances as a function of the QR Code dimensions, for the QR $Version2$ with 25×25 pixels:

QR Dimensions	Recommended Max. Measuring distance
$10cm \times 10cm$	2 mm
$1cm \times 1cm$	0.2 mm
$1mm \times 1mm$	0.02 mm

Table 3.2: Recommended maximum measuring distances

For example, for a realistic measuring distance of $D = 0.5mm$, a QR code (Version 2) with at least $2.5cm \times 2.5cm$ is needed in order to be possible to properly to distinguish individual pixels.

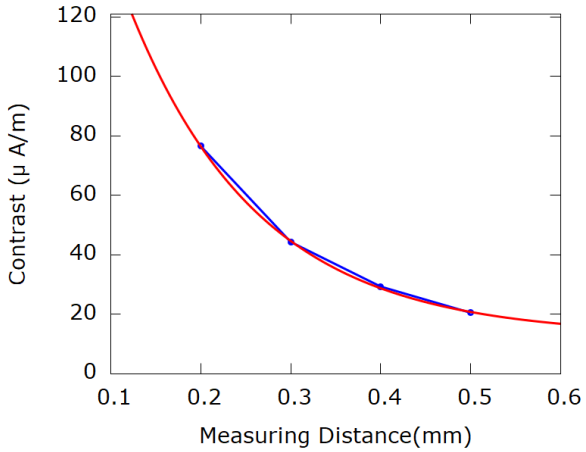


Figure 3.8: Contrast of the Mapping as a function of the detection distance: Negative Exponential Fit

A plot of the contrast of the mapping as a function of the measuring distance is shown in figure 3.8. The following negative exponential model was used:

$$f(x) = a \cdot exp(-b \cdot x) + c \tag{3.3}$$

The parameters for the negative exponential fitting are given in table 3.3.

Parameter	a($\mu A/m$)	b(mm^{-1})	c($\mu A/m$)
Value	254 ± 19	6.9 ± 0.5	12.8 ± 1.8

Table 3.3: Fitting parameters

It can be seen that the last parameter c does not converge down to the expected zero value, indicating that a larger ensemble of simulations is required to obtain a better fitting.

3.6 Mapping as a function of the number of Sensors/Pixel (Bulk)

The two geometrical methods for mapping (mapping at the center of the sensor and mapping over the sensor area) are described and compared in the following subsections.

3.6.1 Detection at Sensor Center

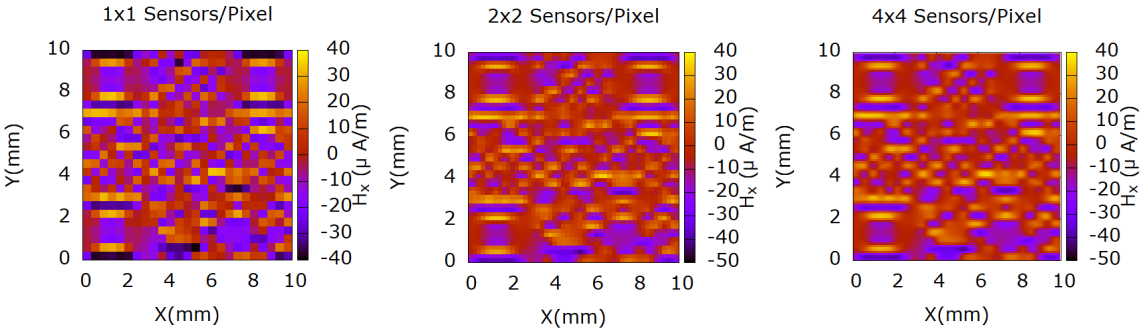


Figure 3.9: Magnetic Field H_X mapping for different number of sensors/pixel(detection at center)

In the simulations shown in figure 3.9, it is considered that detection is done at the center of the sensors. Therefore, only the elements present at the pixel center are considered as field entries.

3.6.2 Detection over Sensor Area

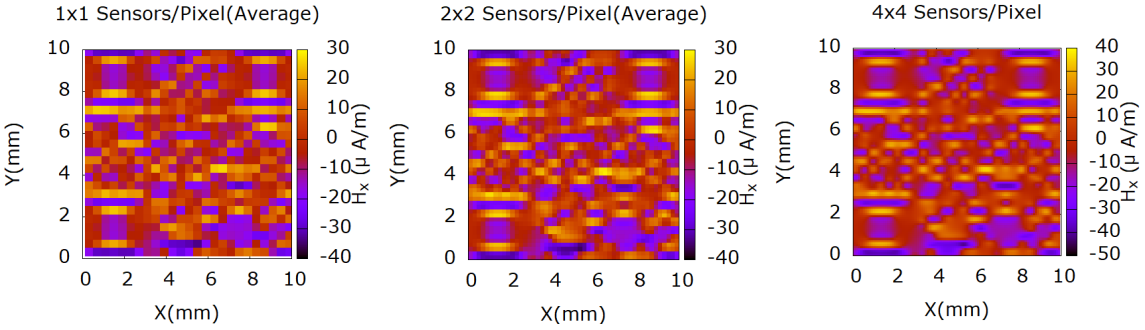


Figure 3.10: Magnetic Field H_X mapping for different number of sensors/pixel(detection over area)

The second method consist of considering that the detection is done over the sensor area. Therefore an average of simulated field values covered by the sensor area is considered, as shown in figure 3.10.

It can be observed that as the number of sensors per pixel increase, the quality of the mapping improves, since the signal quantity of information provided is proportional to $\sim \sqrt{N}$, with N being the number of sensors used.

The decision of which method is more accurate (detection at center or over area of the sensor) needs to be done by comparison with experimental results or taking into account how the sensors function physically during detection. Generally, it can be considered that the second method of averaging the magnetic fields over the sensor area seems to be the most realistic.

However, it was considered important to present these two possibilities since certain magnetoresistive sensors fabricated may have a stronger detectability at the center, while others may have a detectability more wide-spread over the area.

3.7 Perpendicular Reading - Decoding Algorithm

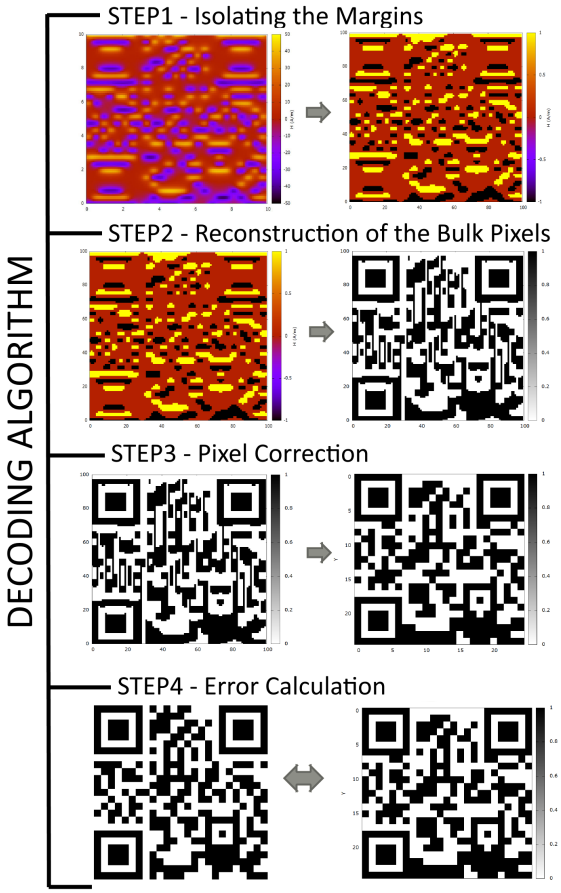


Figure 3.11: Schematic of the Pixel decoding algorithm, showing the four steps of the process

From an experimental point of view, reading the component of the magnetic field perpendicular to the magnetization direction in the sample is the only practical way to do the magnetic mapping of the sample. This is because if the reading and the magnetization were done in the same direction, the external permanent magnet would saturate the sensor, so it would not be able to detect the magnetic field of the sample.

When reading the magnetic field perpendicular to the magnetization direction, a mapping of the superior and inferior edges of the pixels is obtained instead of the bulk of the pixels themselves.

In order to obtain the mapping of the pixels, an algorithm that reconstructs the bulk mapping of the pixels from information of the edges is required. A schematic of the general method is shown in the figure 3.11. A detailed description of each one of the steps will be given in the following sections.

3.7.1 Step1 - Isolating the Edges of the Pixels

The first step is to isolate the regions representing the edges of the pixels. The negative values bellow a certain threshold are converted to $-1(LowerEdge)$ while the ones above a certain threshold are converted to $1(UpperEdge)$, the remaining being considered 0. The threshold used was of $Tr = 65\%$ of the minimum/maximum field value. The simulated mapping of the magnetic field component perpendicular to the plane of magnetization for a QR of $A = 1x1cm^2$ measured at distance $D = 0.2mm$ with a resolution $R = 4x4/pixel$ was used, as shown in figure 3.12.

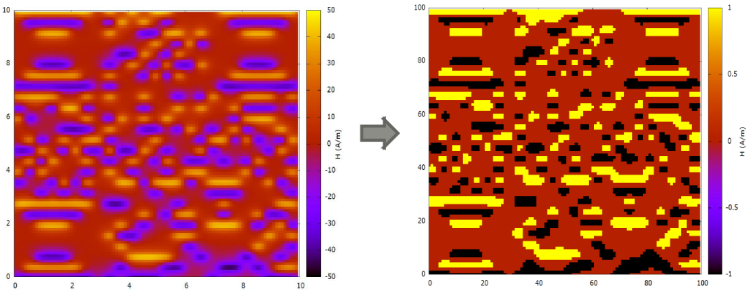


Figure 3.12: Algorithm for the isolation of the upper and lower edges of the pixels

3.7.2 Step2 - Reconstruction of the Bulk Pixels

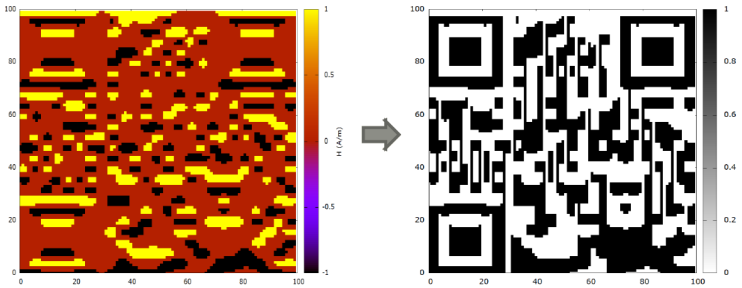


Figure 3.13: Reconstruction of the bulk of the pixels from edge mapping to pixel mapping

The next step is to reconstruct the bulk of the pixels by ”filling” the regions between then negative (lower edges) and positive (upper edges). Filling the vertical zone from the inferior edge($negative = -1$) to the superior edge ($positive = 1$) allows us to obtain the reconstructed “bulk” of the pixels. The output mapping is a binary matrix file: ($pixel = 1$) and ($space = 0$), shown on the right side of figure 3.13.

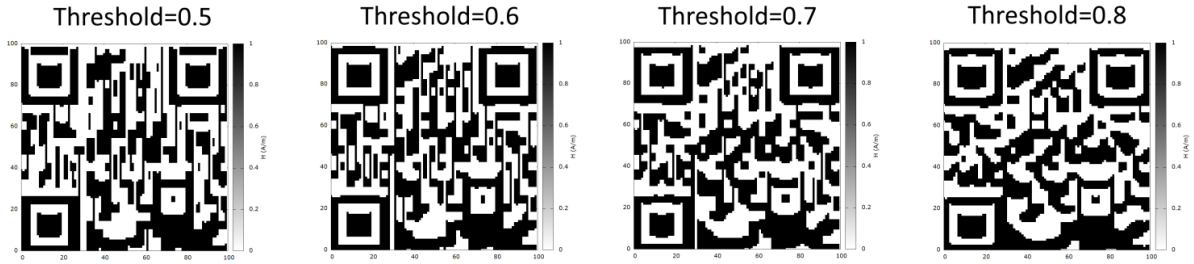


Figure 3.14: Pixel Reconstruction Mapping for different Threshold Values (Simulations)

A comparison of the effect that different threshold values have on the pixel reconstruction algorithm is shown in the figure 3.14. The optimal value is observed around the value of $Tr = 65\%$ for the simulations.

3.7.3 Step3 - Pixel Correction

Finally a correction algorithm needs to be applied to filter that bulk reconstruction of the pixels. This is done by comparing each pixel with the expected nominal size. Experimentally, this could be done by having a standard size for the scanned QR relative to the scanning magnetic camera.

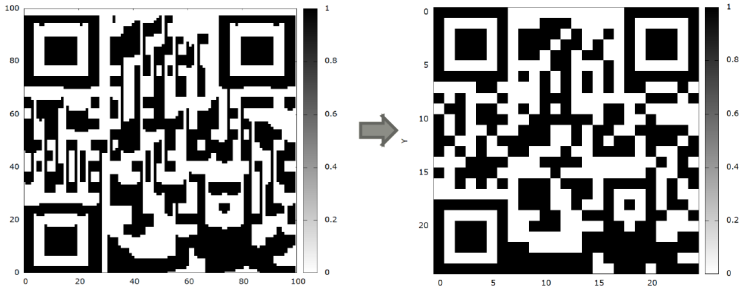


Figure 3.15: Pixel Correction Algorithm Applied, cleaning the pixel mapping

If the area filled is above a certain percentage (Threshold) of the nominal size, it is considered a ($pixel = 1$), otherwise it is considered ($space = 0$). The percentage threshold used in this case was of $Tr = 45\%$. The final output is a $25x25$ matrix ($pixel = 1$, $space = 0$).

3.7.4 Step4 - Error Calculation

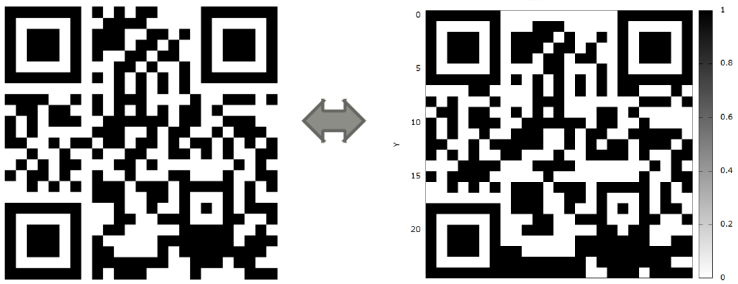


Figure 3.16: Original QR (25x25 Pixels) - Left, QR Reconstructed from Simulations - Right

Finally an algorithm that compares the obtained QR Code with the original one, pixel by pixel. In this case the error was of $E = 10\%$ (below $E = 30\%$, which is the maximum error reconstruction capacity of the Solomon Algorithm used in QR Code Readout), as shown in figure 3.16.

3.8 Experimental Confirmation

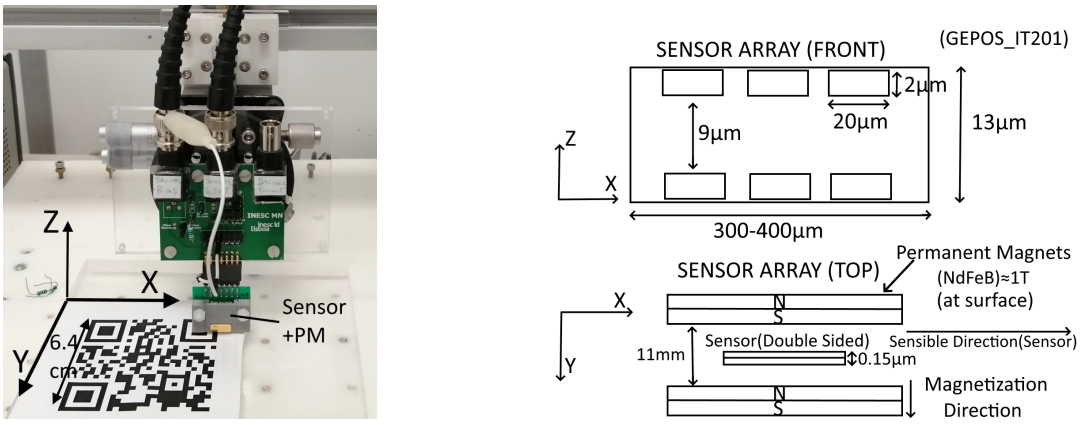


Figure 3.17: Scanner Setup - Left, GEPOS_IT201 Sensor Head Schematic - Right

The Scanner Setup shown in the figure 3.17 was used to obtain a 2D mapping of printed QR codes in order to confirm the simulations. The sensor used contains an array of 6 MTJ Sensors and the holder has permanent magnets attached to it that magnetize the sample printed in Laserjet HP ink (1 layer).

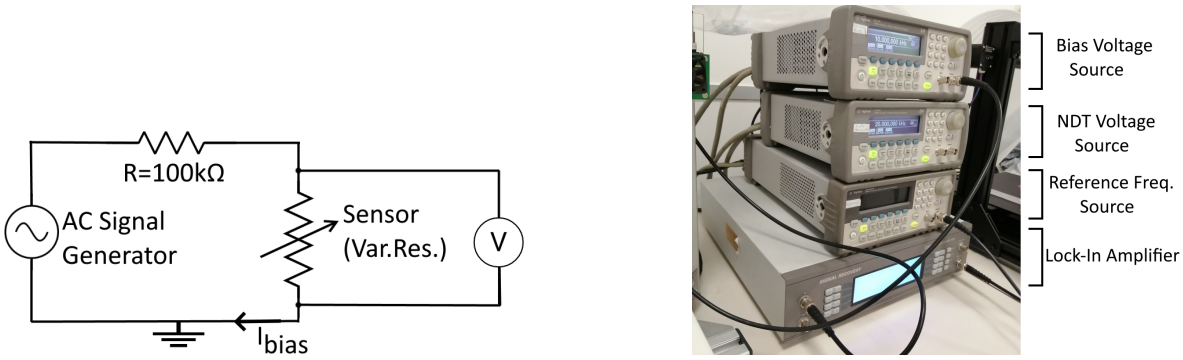
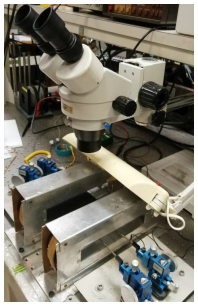


Figure 3.18: Scanner Setup: Electrical Schematic (Left), Supporting Electronics (Right)

The setup also contains supporting Electronics that power the sensor (using a modulating and demodulating $20kHz$ signal) and the mechanical setup that allows the scanner to move along the 3 dimensions during the reading process. The electrical schematic setup and schematic are shown in figure 3.18



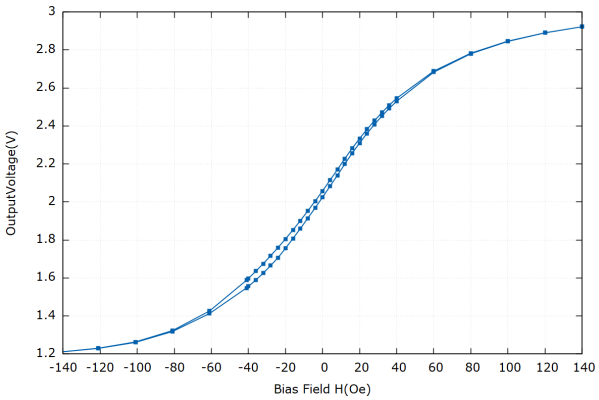


Figure 3.19: In-house Characterization Setup - Left, Sensor Transfer Curve - Right

In order to convert the voltage reading into magnetic units, the sensor was characterised using the in-house setup shown in the figure 3.19. The sensor is biased by the magnetic coils that sweep the range $[-140; 140]Oe$ while applying a bias current of $I_{bias} = 100\mu A$ (equal to the one used in the scanner to operate the sensor), thus producing a Voltage as a function of Bias Field plot.

TMR(%)	R_0	I_{bias}
240%	20kOhm	$100\mu A$

Voltage Base (V)	Conversion Ratio (Oe/Volt)
1.65 ± 0.2	76 ± 1.43

Table 3.4: Parameters Used for Characterization (Left), Voltage to Magnetic Field Conversion Parameters (Right)

The results obtained are shown in the table 3.4, and are used in the rest of the section for the voltage to magnetic units conversion. The conversion expression is given by the expression 3.4.

$$H(Oe) = ConversionRatio \cdot V(Volt) - VoltageBase \tag{3.4}$$

3.8.1 Step1 - Isolating the Edges of the Pixels

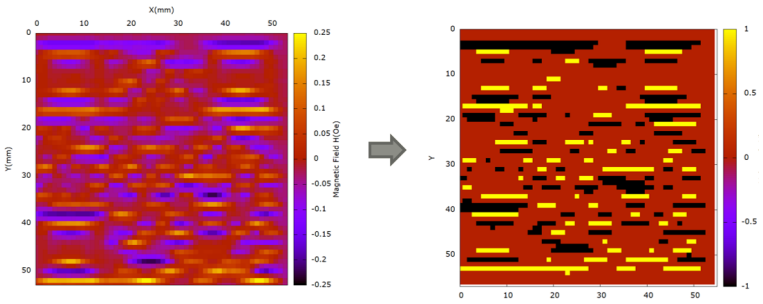


Figure 3.20: Algorithm for the isolation of the upper and lower edges of the pixels

The same procedure as in the previous section was used for decoding the 2D mapping of the edges, as shown in figure 3.20. The best experimental mapping from the ones scanned was used (QR Code Size $A = 5x5cm^2$, Scanning distance $D = 0.7mm$, Resolution $R = 2x2/pixel$).

The threshold used for the isolation of the edges was of $Tr = 90\%$ of the minimum/maximum magnetic field value, higher than the $Tr = 65\%$ threshold used in the simulations.

3.8.2 Step2 - Reconstruction of the Bulk Pixels

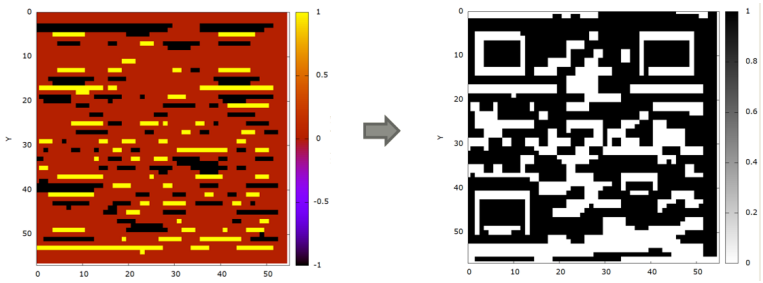


Figure 3.21: Reconstruction of the bulk of the pixels from edge mapping to pixel mapping

The same procedure from reconstructing the bulk mapping of the pixels was used as in the case for the simulations, by filling the space between the lower and upper edges, as shown in figure 3.21.

3.8.3 Step3 - Pixel Correction

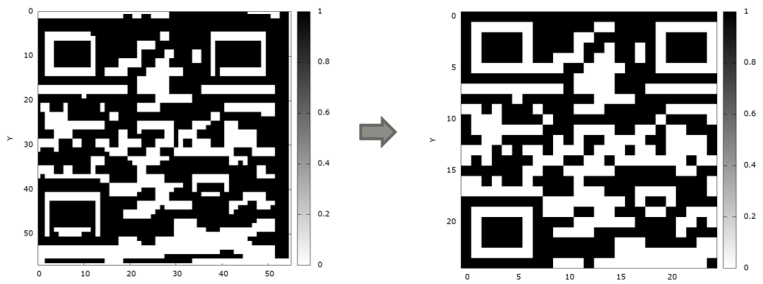


Figure 3.22: Pixel Correction Algorithm Applied, cleaning the pixel mapping

In the experimental case, the optimal threshold for the pixel correction was found to be $Tr = 80\%$, much higher than the value of $Tr = 45\%$ in the case for the simulations.

3.8.4 Step4 - Error Calculation

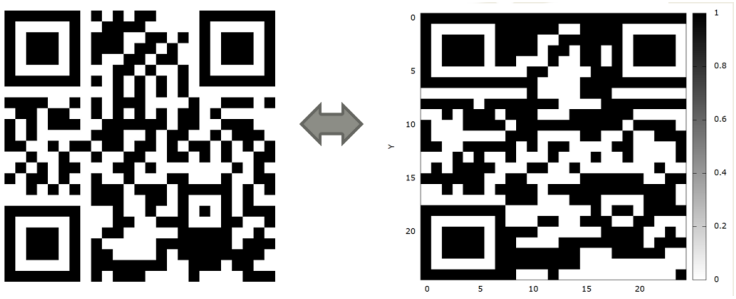


Figure 3.23: Original QR (25x25 Pixels) - Left, QR Reconstructed from Simulations - Right

The error in this case was of $E = 24.96\%$, higher than in the case of the simulations, as expected. The main source for errors in the experimental case is the electronic noise present in the measurements, the non-uniformity of the pixels (due to the roughness of paper) and the fact that the printed QR Code is not perfectly flat, nor is the scanning device perfectly aligned.

3.9 Comparison

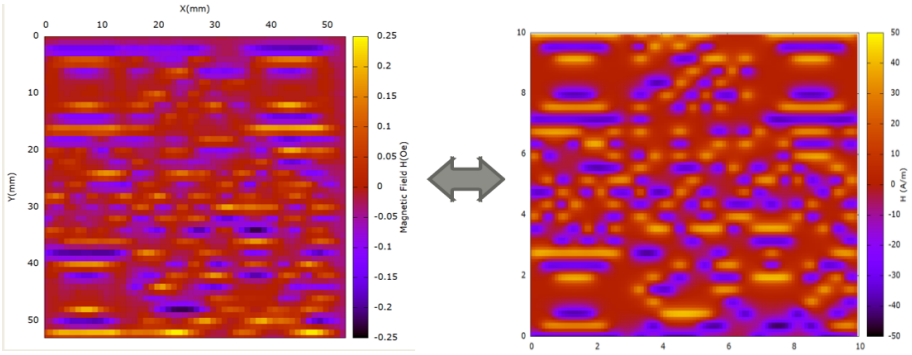


Figure 3.24: Magnetic Field Mapping from the scanned QR - Left, Magnetic Field Mapping from the simulated QR - Right

The comparison between the experimental and simulated field mapping is shown in the figure 3.24. The conversion in the experimental was done using the conversion method shown in section 3.8.

It can be observed that there is a good correlation between them, showing that the algorithm used can produce reliable simulations despite the approximations used.

The comparison of the experimental error is shown in the table 3.5. It can be observed that the error is bellow the $E = 30\%$ which is the maximum error reconstruction capacity of the Solomon Algorithm in both cases.

Simulated Error	Experimental Error	Solomon Algorithm Threshold Error
10%	24.96%	30%

Table 3.5: Error Comparison between experimental measurements and simulations

In both sections 3.8.1 and 3.8.3, the Threshold value used was much higher for the experimental data than for the simulated one. This might be due to the fact the the SNR (signal to noise ratio) increases with the strength of the magnetic field detected, so the regions at the center of the pixels have a better

Figure 3.25: Comparison between the Simulated (Left), Scanned (Middle), and Original QR Code (Right)

SNR, therefore a higher threshold is needed to isolate them since the transition is more discrete, while in the case of the simulations the transition from empty space to pixel is smoother.

Chapter 4

System Setup

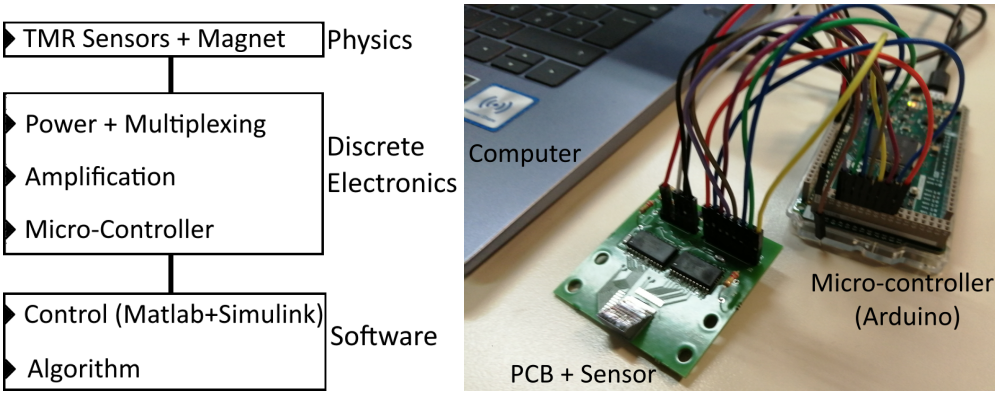


Figure 4.1: General System Setup Schematic (Left) and Photo of the Final System Setup (Right)

The General Setup of the proposed system is shown in figure 4.1. A sample printed on paper using magnetic ink is to be read by a matrix of Tunneling Magnetoresistive (TMR) Sensors. The sample is to be magnetized by the permanent magnets installed on the sensor.

A PCB (Printed Circuit Board) Powers the sensors, chooses the sensor to be read by multiplexing, amplifies the signal and sends it to the micro-controller (Arduino Due). The Micro-controller controls the the multiplexers using digital logic, allowing the user to sweep the sensor matrix and send the data to the computer.

At the Software Level, the reading is done using Matlab and Simulink. The signal is to be filtered digitally by averaging a certain amount of inputs. The data is then passed trough an algorithm and then finally converted to a a binary matrix text file containing the pixel mapping of the sample.

In the following sections, a detailed description of each element of the System Setup is presented.

4.1 Sensor Setup

The proposed Sensor Configuration is shown in the figure 4.2. The main idea is to create a $10x10$ matrix of 100 Tunneling Magnetoresistive Sensors in order to read 2d magnetic patterns printed in magnetic ink. Since the magnetic ink produces a very weak magnetic field, the sensors are to be positioned as close as possible to the sample in order to increase the detected signal. In order to achieve that, a stair-shaped geometry was proposed, with 10 sensors as close as possible to one edge of each one of the 10 levels, with the contact pads at the opposite edge.

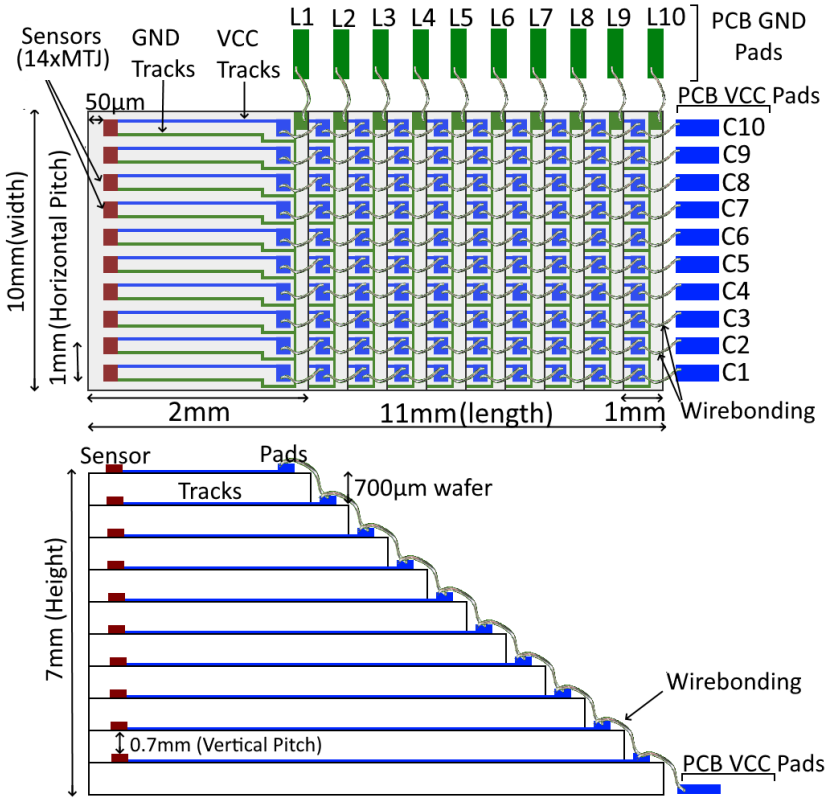


Figure 4.2: Sensor Setup (Top and Side View) showing the proposed Stair-Shaped Sensor Setup

Each consecutive level is taller so as to expose the pads. Wire-bonding is to be done linking the same column of each level downwards until the PCB level pads (VCC). This allows the user to select the column to be read. The GND pad of each level is connected directly to the PCB pads sideways, allowing the user to choose the line to be read. By choosing the line and the column,each sensor is read individually.

4.2 Electrical Setup

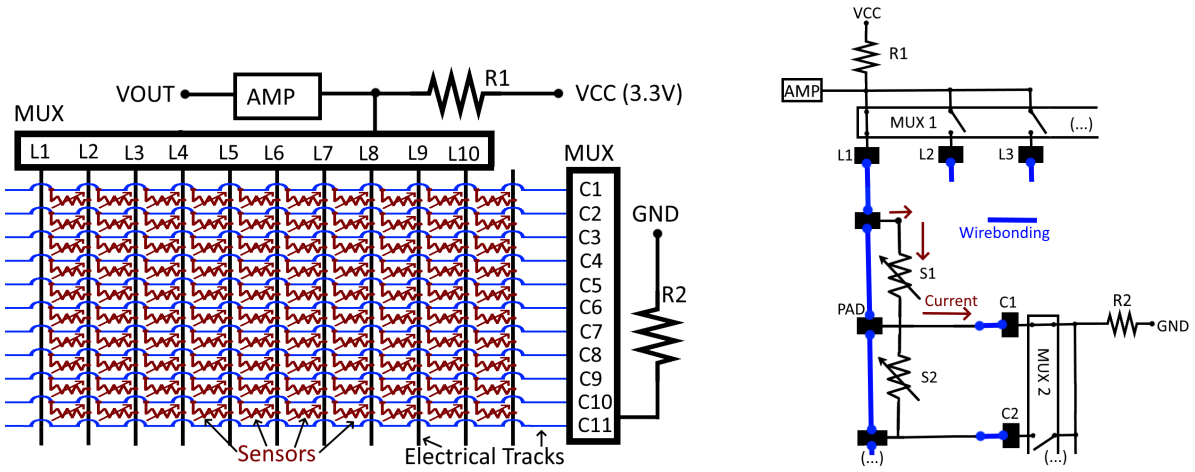


Figure 4.3: General Electrical Setup Schematic(Left), Detailed Electrical Setup Schematic (Right)

The proposed electrical setup is shown in the figure 4.3. The Tunneling Magnetoresistive Sensors

are represented as variable resistances (red) and are connected in parallel to an electrical grid. The negative side is connected to the ground via a multiplexer that chooses the column to be read. The positive side is connected to VCC($3.3V$) via a multiplexer that chooses the line to be read.

Two resistance on the positive and negative terminal are used as voltage dividers in order to limit the current passing trough the sensor so as to avoid breakdown of the oxide layer within the sensor that occurs above certain value of current. Two equal resistances are used instead of just one so as to center the signal in the middle range between GND and VCC, making it easier to process electronically during amplification (AMP) before being sent as output (VOUT) to the micro-controller and then to the computer.

The drift of the signal is to be filtered using digital filtering by averaging several dozens (10-100) readings per measurement (depending on the limitations imposed by the desired total measuring time for the whole sample). The origin of the drift of the signal is the thermal, shot and electronic noise as described in section 2.4.1 and more explicitly shown in figure 2.5.

4.2.1 Electrical Setup PCB Design

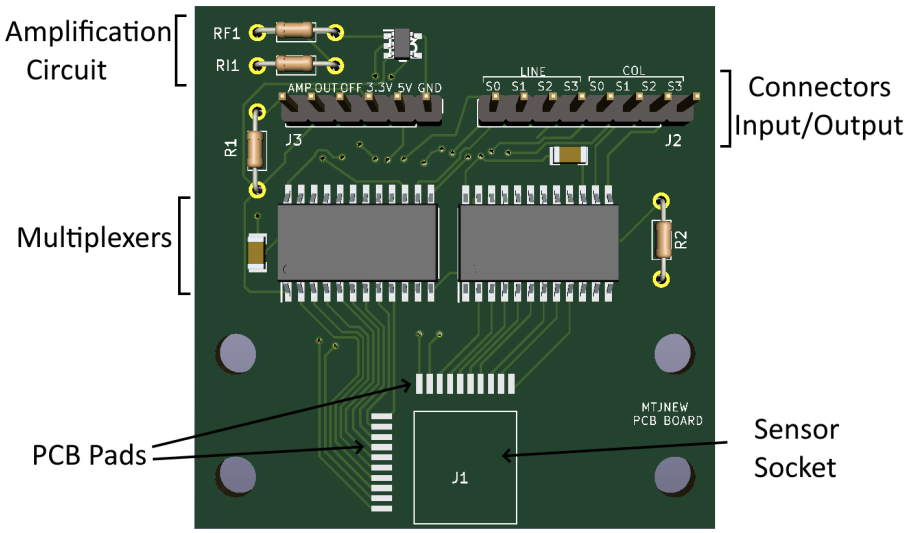


Figure 4.4: Snapshot of the proposed PCB Board, showing the sensor, multiplexing and amplification regions

A PCB Board was designed using Kicad (CERN Open Source Software) and ordered online from Aisler (Custom PCB Board Supplier), along with the electrical components. As shown in figure 4.4, it contains three main regions: the region containing the sensor socket, one for the multiplexers and one for the amplification circuit.

On the left the region for mounting the sensor is marked around J1. The pads for reading the lines L and the pads for reading the columns C are around it, and connection to the sensors is to be done via wire-bonding. In the middle there are two multiplexers that allow the user to choose the column and line to be read via digital logic input.

At the top left-side there is the amplifying circuit, which plays a very important role since the magnetic fields to be read are very weak. Amplification by a factor of $A = 1000$ to $A = 2000$ are required in order to detect patterns printed on paper using magnetic ink. Finally there are two connectors: one for power input and analog output, and another one for the digital inputs that control the multiplexers.

4.2.2 Signal Amplification and Filtering

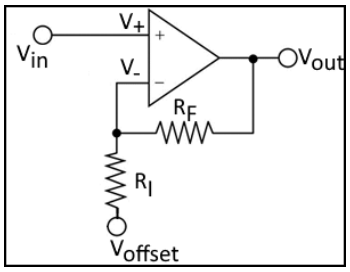


Figure 4.5: Electrical Schematic of the Amplification Circuit Used

The schematic of the amplification circuit used is shown in figure 4.5. It consists of an operational amplifier and two resistances, whose ratio A controls the level of gain/amplification from the point of V_{offset}, given by:

$$A = \frac{R_F}{R_I} \tag{4.1}$$

In order not to saturate the amplifier during amplification, an external bias voltage V_{offset} needs to be used, which is done using the DAC(Digital to Analog converter) output present in the Micro-controller Board. This results in an output voltage given by:

$$V_{out} = \left(1 + \frac{R_F}{R_I}\right) V_{in} - \frac{R_F}{R_I} V_{offset} \tag{4.2}$$

If the offset voltage is equal to the background voltage of the sensor, this expression simplifies to:

$$V_{out} \approx \bar{V}_{in} + A \cdot \Delta\tilde{v}_{in} \tag{4.3}$$

- $\bar{V}_{in}$ - background idle voltage of the sensor
- $\Delta\tilde{v}_{in}$ - variation of the voltage of the sensor

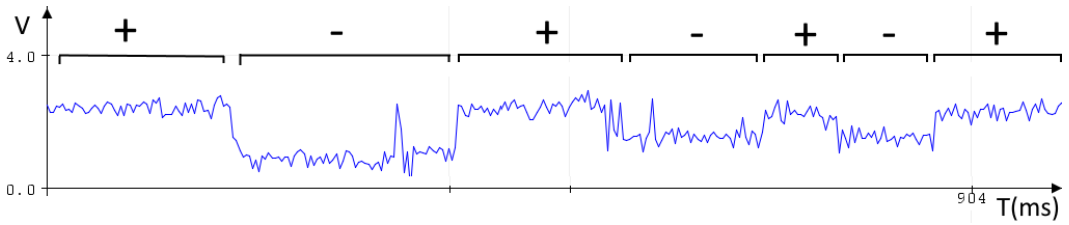


Figure 4.6: Signal test of the amplification circuit used (A=1000) on magnetic scales

It is the amplified signal of the variation of the voltage $\Delta\tilde{v}_{in}$ that provides detection of magnetic patterns. This amplification circuit was tested on magnetic scales using an already existing sensor setup employing MTJ sensors and permanent magnets.

The results are shown in the figure 4.6. The amplification level required was $A = 1000$, beneath this level no detection was observed experimentally. This might be due to the fact that beneath certain a certain amplification threshold, the noise level is larger than the detected signal ($SNR < 1$), therefore no actual noticeable signal is detected.

The strength of the signal of $\pm 1V$ observed correspond to a sensor signal of $\pm 1mV$, and the noise or drift level of the signal is of the order of $\pm 0.2V$, corresponding to a value of $\pm 0.2mV$, at sensor level.

Digital filtering is to be used in the final setup by taking the average of a large number of measurements as a reading (10-100 measurements, depending on the desired total measurement time), as previously mentioned in 4.2.

Since there is a dispersion on the base values of resistance of the different sensors within the same stack (as shown in Section 5.4.1) of the order of $400Oe$, the V_{offset} value would need to be readjusted when switching trough different sensors in order to maintain the consistency of reading. This can be done by measuring the non-amplified voltage V_{in} of each sensor and adjusting the V_{offset} to the same value using the DAC converter, before reading the amplified signal and applying the digital filtering.

4.2.3 Electrical Setup Fabrication

Step Number	Step Description	Equipment Used
1	Contact Leads Deposition	Nordiko 7000
2	Leads Definition DWL	Heidelberg DWL 2.0
3	Lead Definition Etching and Ashing	SPTS
4	Contact Protection DWL	Heidelberg DWL 2.0
5	Isulating Layer Deposition	UHV2
6	Oxide Lift-Off	Wet Bench
7	Wafer Dicing	Disco DAD Dicing Saw
8	Sample Assembly	Bio-lab Oven

Table 4.1: Fabrication Steps employed in the fabrication of the Electrical Prototype Setup

A simplified version of the proposed sensor setup was developed containing only the electrical pads and leads (without sensors) for testing purposes, before fabricating the final sensor setup. The areas corresponding to the sensor are replaced by alternatively connected and disconnected leads. A summary of the fabrication steps and of the deposited structure is shown in fig. 4.1. The following structure was deposited: [Si / SiO_2 2000Å / $AlSiCu$ 3000Å / Al_2O_3 3500Å].

Leads Definition (Steps 1 to 3)

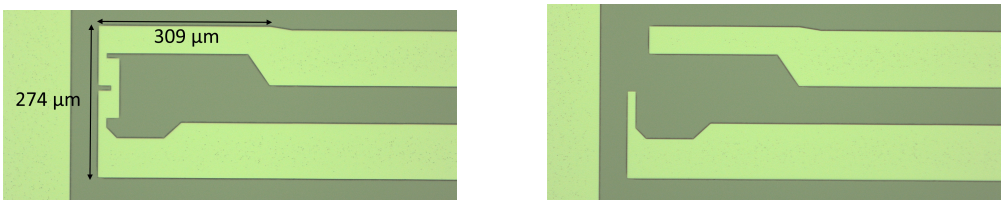


Figure 4.7: Lead Definition under Microscope - Connected (Left) and Disconnected (Right)

A layer of 3000 Å Stack of AlSiCu was deposited on top of a 6 inch wafer using the Nordiko 7000 Machine. A layer of 1.5 μm of positive photoresist was then deposited on top after 30 minutes of vapour priming. The pattern of the electrical paths or leads was then defined using lithography in the Heidelberg DWL 2.0 machine and the photoresist was then exposed. The sample was then Etched during 60s in the SPTS Machine, followed by 180s of Ashing to removed the remaining photoresist.

After optical inspection under the microscope it was concluded that the sample was over-etched, with the 3000Å $AlSiCu$ layer etched and also the 2000Å SiO_2 insulating layer that was present on the wafer in the beginning being etched as well. This was observed by the change of colour of the exposed surface (the surface of the wafer with oxide is Violet, but changed to Grey after etching). This is then confirmed using the profilometer.

The region at the end of the leads where the MTJ Junctions would be positioned in the final setup were chosen as being alternatively connected and disconnected in order to test the functioning of the multiplexers when switching channels in the prototype version. This is shown in the figure 4.7.

Pads Definition (Steps 4 to 6)

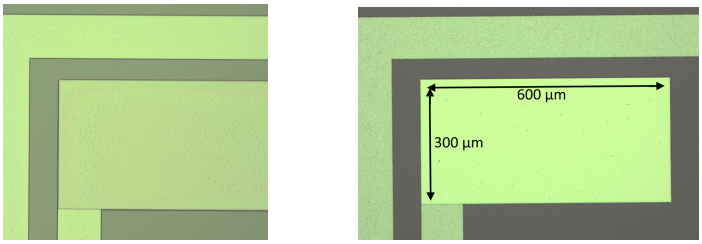


Figure 4.8: Pad Definition under Microscope - Before Liftoff (Left) and After Liftoff (Right)

Once again a layer of 1.5 μm positive photoresist was deposited and the pads were defined using lithography using the Heidelberg DWL 2.0. An Insulating layer of 3500 Å of Al2O3 was then deposited in the UHV2 Machine. Liftoff was then performed in the wet bench. The results is shown in the figure 4.8.

In the table 4.2 a summary of the profilometry results obtained are given. This proves that the sample was over-etched, and also that the thickness of the deposited AlO_3 oxide was much larger than expected, so the deposition time was miscalculated.

Deposited Material	Expected Value(Å)	Measured Value(Å)
AlSiCu	3000	3056
AlO_3	3500	4896
Al Etching	3500	5056

Table 4.2: Profilometer Measurements - Electrical Setup

Wafer Dicing (Step 7)

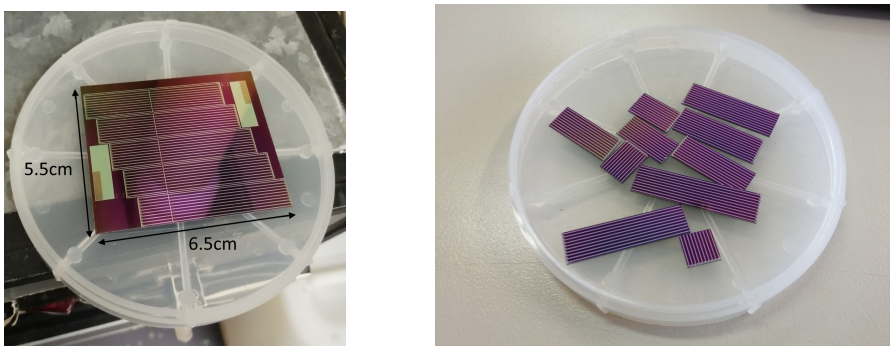


Figure 4.9: Sample Before Dicing (Left) and after Cutting (Right)

The cutting of the wafer was done in the Disco DAD Dicing Saw Setup using a saw with a thickness of $150\mu m$. First the square around the setup was cut from the wafer, then the middle line along the sensor area was cut, and then perpendicularly along the height of the levels. Finally the individual pieces were reassembled in the cutting machine and the top part of each level was cut. This procedure is shown in the figure 4.9.

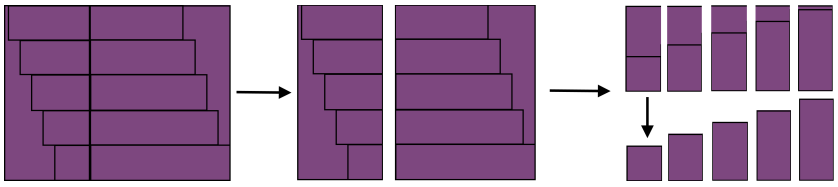


Figure 4.10: Schematic of the Dicing Procedure used for the Electrical Prototype Setup

Gluing and Assembling Levels (Step 8)

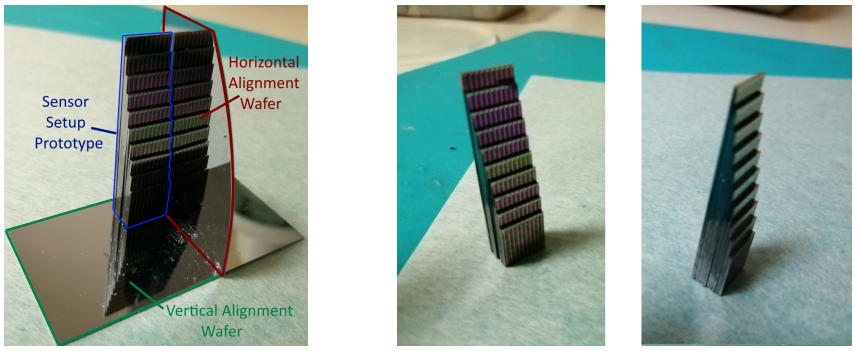


Figure 4.11: Manual Assembly Method (Left) and Result (Right) of the Electrical Prototype Setup

An attempt to glue the setup manually was tried by using a wafer for horizontal alignment and another one for vertical alignment. A small drop of Super Glue 3 was put between each successive layer and spread along the sample by pressure. This was done in the bio-lab under typical atmospheric conditions.

As was afterwards observed, this method produced quite large misalignment between the different levels. This proves that for the final setup a mechanical device for alignment has to be developed and the gluing of the setup has to be done in the Class 100 clean-room after cutting and washing the sample to remove impurities. The method and results are shown in figure 4.11.

4.2.4 Electrical Setup Resistance Test

The resistance of the vias was measured using the 4-point in-house setup. A plot of these results is shown in the figure 4.12.The uncertainty of the resistance is not shown due to being negligible. It can observed that while advancing along the column the distance from the ground track increases, so the resistance increases slightly. While advancing along the levels, the tracks become longer so the resistance also increases.

The results are summed in the table 4.3. The slope of the fitting allows the obtaining of the resistivity of the material, which is compared to the tabulated value, while the value at the origin gives the value of the resistance of the electric tracks that in the region that will be connected to the sensor. It can be observed that the resistivity of the compound $Al_{98}Si_{1.5}Cu_{0.5}$ used is larger than of pure aluminium.

Aluminium Resistivity ($\Omega \cdot m$)	Experimental ($\Omega \cdot m$)	Error	Junction Resistance (Ω)	Error
$2.65 \cdot 10^8$	$3.24 \cdot 10^8$	1.4%	1.37	1.27%

Table 4.3: Resistivity Measurements of the Electrical Setup

The tracks have a width of $100\mu m$ and a thickness of $0.3\mu m$, giving a profile area of $A = 30\mu m^2$.

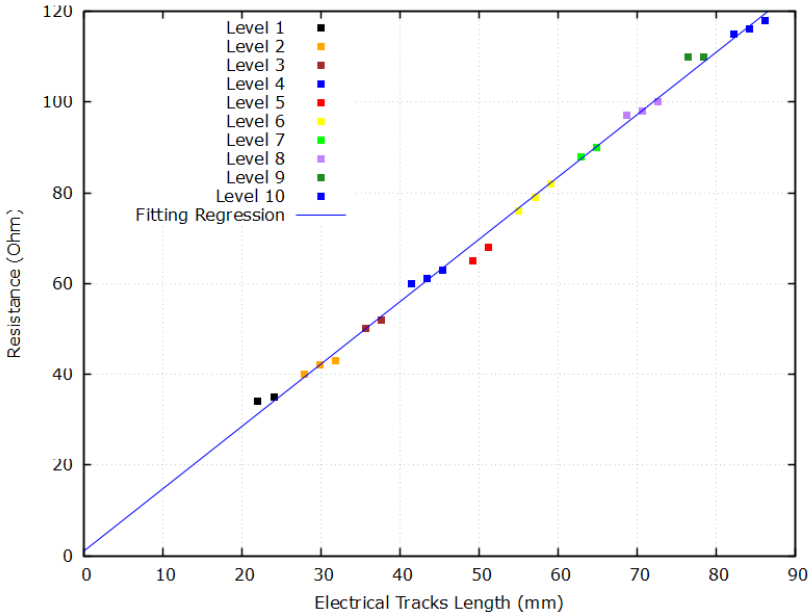


Figure 4.12: Plot of the resistances as a function of the length of the electrical tracks

The resistivity was calculated from the resistance using the following expression:

$$R = \frac{\rho}{A} \cdot l \qquad \rightleftharpoons \qquad \rho = \frac{A \cdot R}{l}$$

- ρ - electrical resistivity ($\Omega \cdot m$) A - profile area (m^2)

However, it can be observed from figure 4.12 that the resistance values of the electrical tracks are of the order of $100Ohm$ in the longest tracks, which is negligible compared to the values of the resistance of the sensors that are of the order of $kOhm$.

Chapter 5

Sensor Fabrication

In this chapter, the methods employed in the fabrication and characterization of the magnetoresistive sensors is described. First, the general sensor structure and design of the sensors is shown, followed by the detailed description of the nano-fabrication process. Finally the characterization methods and the obtained parameters are shown.

5.1 Sensor Structure and Design

The structure and design of the fabricated sensors is described in the following sections.

5.1.1 MTJ Stack

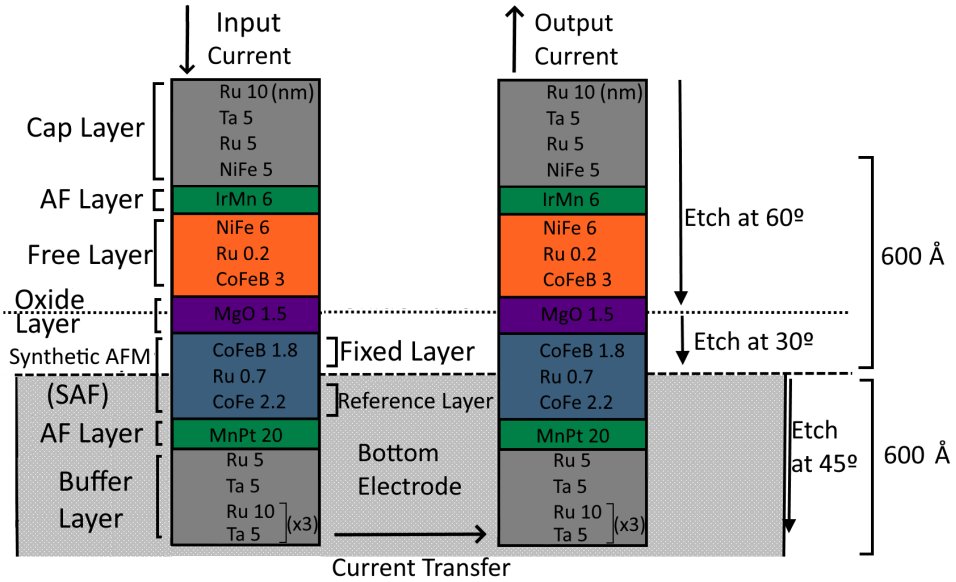


Figure 5.1: MTJ Junctions Overview, showing the different magnetic layers within the junctions

The MTJ junctions fabricated are constituted by various levels, as shown in the figure 5.1. In order to define the pillars, etching at $60º$ is to be done reaching the level of the oxide layer of the stack, and then etching at $30º$ is to be done further, up to 600Å. The bottom is to be defined using etching at $45º$ up to a depth of 600Å, such as to reach the isolating oxide layer present on the wafer beneath the stack.

Thus a system is formed, by which current enters trough one of the pillars, is transferred to another pillar after traversing the oxide layer beneath the free and fixed layers, and exits in the same way. The relative orientation between the free layers and the fixed layers in each pillar induced by external magnetic fields affects the output current, thus resulting in detection.

The MTJ Stack used was the following (units in nm):

[(Ta5/Ru10)x3/Ta 5/Ru 5/MnPt 20/Co80Fe20 2.2/Ru 0.7/ CoFeB 1.8/ MgO 1.5/ CoFeB 3/ Ru 0.2/ NiFe 6/ IrMn 6 / NiFe 5/ Ru 5/ Ta5 / Ru 10]

The expected results for CIPT(Current In-Plane Testing) after annealing for the given MTJ stack were: $RA \sim 2.42kOhm$ e $TMR \sim 175\%$.

5.1.2 Sensor Design

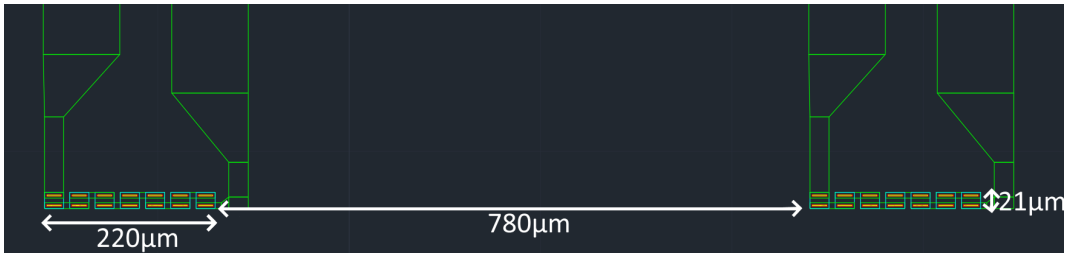


Figure 5.2: Sensor Design Overview showing the geometry used in the nanofabrication masks

The sensor is constituted by 14 MTJ Junctions with dimensions $20 \times 2\mu m^2$, resulting in a total sensing area of $A = 220 \times 21\mu m^2$, with a pole-pitch of $PP = 780\mu m$ between the sensors, as shown in the figure above. The constant resistivity$\times$area of the MTJ stack used is of $RA = 2.38k\Omega \times \mu m^2$, resulting in an expected total resistance of $R = 4.25k\Omega$ per sensor. The design is shown in figure 5.2.

5.2 Processes and Equipment Used

The processes and the equipment used in the fabrication of the sensors are described in the following sections.

5.2.1 Photoresist Coating and Lithography

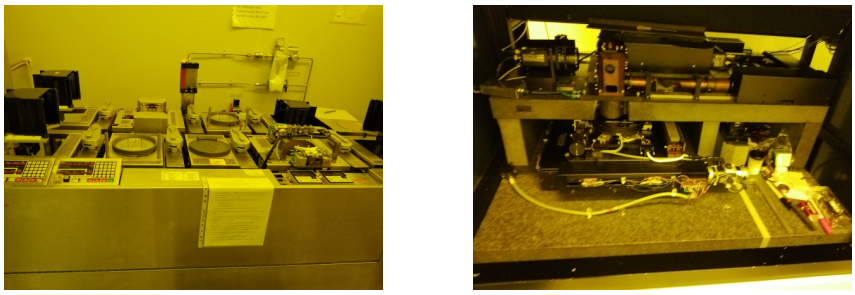


Figure 5.3: Silicon Valley Group coating track (Left) and Heidelberg 2.0 DWL (Right)

Most steps of the process began by defining the design of the mask using lithography. First the sample is mounted on a support wafer and exposed to vapor priming for 30 min to make its surface more receptive to the photoresist, or baked on the oven of the coating track at 100ºC for 120s. It is then

coated with photoresist while being spun at 2500 rpm for 40s and then baked at 100ºC for 60s, resulting in a uniform layer of $1500\mu m$ positive photoresist that is reactive to certain wavelength of lights, thus allowing patterns to be drawn on it.

The pattern of the mask is then inserted in the Heidelberg 2.0 Direct Write Laser, that uses a $405\mu m$ laser beam to draw the pattern of the mask on the photoresist layer. The process is done in lics of $200\mu m$ width, and the setup has a maximum resolution of $0.8\mu m$ and maximum alignment precision of $0.1\mu m$.

After laser exposure, the sample is developed during 100s at the SVG Track, after 60s of heating at 110ºC. During this process, the positive photoresist is removed from the region that was exposed to the laser, while in the remaining areas it remains intact. The sample is the examined under microscope before proceeding to the etching or liftoff steps.

5.2.2 Etching

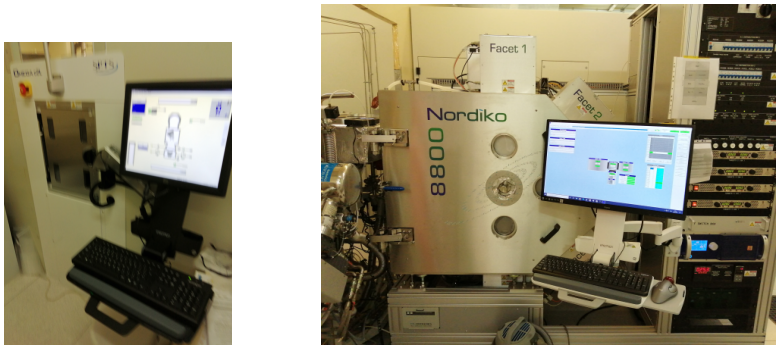


Figure 5.4: SPTS Machine (Left) and Nordiko 8800 (Right) Etching Machines

After material deposition and lithography and revealing, the sample is normally exposed to etching, which does the actual physical definition of the desired features by removing the material around or within the structures of the mask. This is done in machines such as Nordiko 3600, which used an accelerated ion beam in vacuum to remove material in the regions not protected by photoresist.

Another machine used mainly to open the vias of the sensors was the SPTS Machine, that has a selective reactive ion etching of metals and oxides, that allowing to remove one without affecting the other.

5.2.3 Metal Deposition and Passivation

Figure 5.5: Oxford Instruments Machine (Left) and Nordiko 7000 (Right) used for deposition

The deposition of metallic layers was done using Nordiko 7000, using magnetron sputtering deposition. The material removed is propelled towards the sample, ending up being deposited on its surface. For oxide deposition, the Oxford Instruments Machine was used.

5.2.4 Resist Strip, Optical Inspection and Profilometry

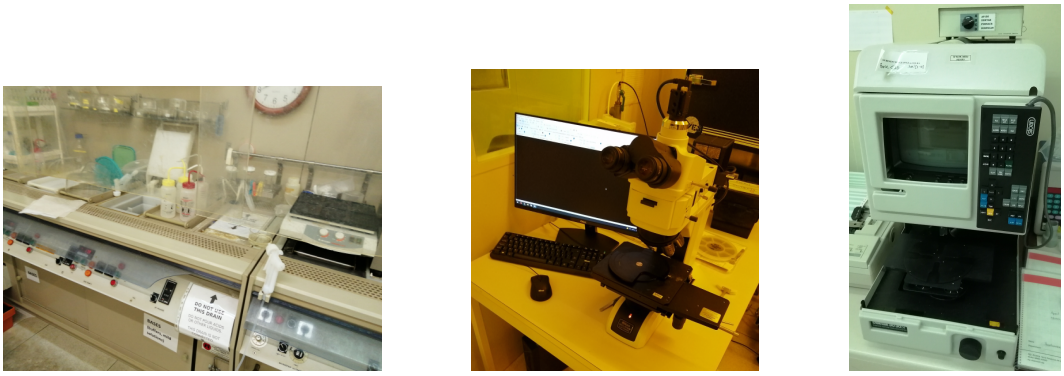


Figure 5.6: Wet Bench (Left), Optical Microscope (Right) and Profilometer Setup (Right)

After the etching process, the remaining photoresist is removed using the Fujitsu Microstrip 3001, while being exposed to ultrasounds at temperatures of $T = 60 - 70$ºC for 2-3 hours. Afterwards the sample is cleaned with acetone, isopropanol, water and dried using compressed air. This is done at the wet bench, where the liftoff process is also done, which consist of doing the lithography before deposition a material layer on top. In this case, the areas with photoresist are removed together with the material above it, a process opposite of etching, where photoresist is used to protect.

After each step of resist strip the sample is normally inspected under microscope to check if the features were correctly etched, and under profilometer to check if the etch was done with the correct depth.

5.2.5 Electrical and Magnetic Characterization

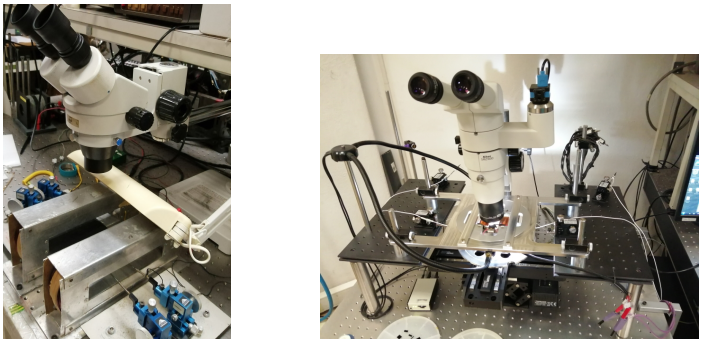


Figure 5.7: In-house four-point setup (Left) and Auto-Prober (Right)

The electrical and magnetic measurements around the center-line of the transference curve of the sensors is done using the ”in-house four-points 140 Oe measurement setup”. Two contact probes are used to insert current into the pads of the sample, while the other two are used to measure the voltage, producing an I-V of the sample. Another possible measurement is of the transference curve of the sensor, which is the Voltage as a function of the Applied Bias Field. The positioning of the probes is

done using a microscope, and the magnetic field is produced by two magnetic coils capable of producing fields up to $H_{bias} = 140Oe$. In order to measure the saturation and total magnetoresistance, the auto-prober is used since it has more powerful coils that reach values of up to $H_{bias} = 300Oe$.

5.2.6 Wafer Dicing

Figure 5.8: DAD Wafer Dicing Setup (Left) and Sample Positioning (Right) during the Dicing Process

At the end of the fabrication process, the dicing of the sample is done using the DAD Disco Setup, employing a dicing saw with a thickness of $t = 150\mu m$ or $t = 60\mu m$ for more sensible samples. During the cutting process, the dicing saw is cooled and cleaned by the water jet of the setup. Before inserting in the machine, the sample has to be glued and centered on a support structure.

5.2.7 Thermal annealing (Steps 18)

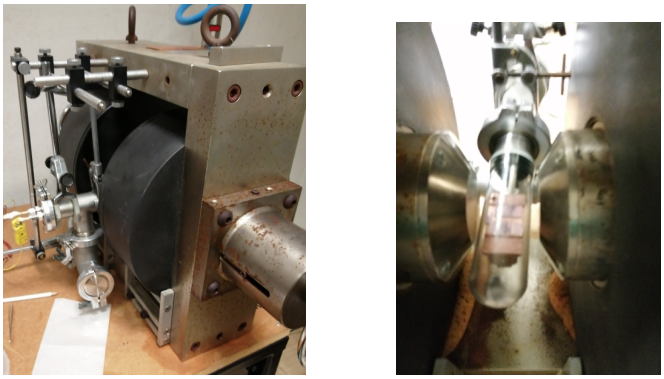


Figure 5.9: Electromagnet Annealing Setup (Left) and Sample during annealing (Right)

Thermal Annealing is the process by which the magnetic layers within the sample are heated to a high temperature of around $T = 300$ºC and to a strong magnetic field of the order of $B = 0.5 - 1T$, thus aligning the microscopic magnetic domains within the sample towards a certain direction. The sample is immersed in vacuum, while a halogen lamp produces the heating. The field is produced by powerful water cooled electromagnets.

5.3 Fabrication Sequence

An overview of the fabrication steps is shown in the table 5.1. The deposited structure was: [TMR Stack 1200Å / SiO_2 2700Å / AlSiCu 3000A / SiO_2 4000Å].

Step Number	Step Description	Equipment Used
1	TMR Stack Deposition	Nordiko 3600/8800
2	1st Lithography – Pillars	Heidelberg DWL 2.0
3	MTJ Definition by Ion Milling	Nordiko 3600
4	Resist Strip	Wet bench
5	2nd Lithography – Bottom electrode Definition	Heidelberg DWL 2.0
6	Bottom electrode Definition by Ion Milling	Nordiko 3600
7	Resist Strip	Wet bench
8	Passivation – SiO2 Deposition	Oxford
9	3rd Lithography – Vias Opening	Heidelberg DWL 2.0
10	Vias opening by Reactive Ion Etching + Ashing	SPTS
11	4th Lithography – Top electrode	Heidelberg DWL 2.0
12	Metallization	Nordiko
13	Liftoff	Wet Bench
14	Passivation – SiO2 Deposition	Oxford
15	5th Lithography – Pads opening	Heidelberg DWL 2.0
16	Pads opening by Reactive Ion Etching + Ashing	SPTS
17	Sample Dicing	Disco DAD Dicing Saw
18	First Thermal annealing	Turbomagnet Annealing
19	Second Thermal annealing	Turbomagnet Annealing

Table 5.1: Nanofabrication steps used for the final sensor setup

These fabrication steps will be explained in detail in the following sections.

5.3.1 TMR Stack Deposition (Step 1)

The MTJ Stack was already deposited on a wafer and a square sample was cut from it and used to fabricate the sensor setup.

5.3.2 MTJ Pillar Definition (Steps 2 to 4)

Figure 5.10: Individual Junction (Left) and Sensor Overview (Right) after the Pillar Definition Process

After the first lithography, the sample was inspected under the microscope to verify if the dimension of the sensor structures were correctly defined. The results are shown in the figure 5.10 and summarized in the table 5.2. It can be observed that the lithography definition of the width of the pillars is slightly lower than the expected value, while the height is correctly defined.

Average Height(μm)	Expected Value(μm)	Average Width(μm)	Expected Value(μm)
18.9$\pm$0.23	20	1.9$\pm$0.1	2

Table 5.2: Dimensional Measurements - MTJ Pillars

Figure 5.11: Individual Junction (Left) and Sensor Overview (Right) after the Bottom Definition Process

Average Height(μm)	Expected Value(μm)	Average Width(μm)	Expected Value(μm)
23.89$\pm$0.04	25	20.21$\pm$0.002	21

Table 5.3: Dimensional Measurements - Bottom Electrode

5.3.3 Bottom electrode Definition and Passivation (Steps 5 to 8)

The bottom electrode was then defined during the second etching at 45° in the Nordiko 3600 Machine. The results are shown in figure 5.11. Afterwards a layer of SiO_2 oxide with thickness of $T = 2700$Å was then deposited in the Oxford Machine. Profilometer measurements were performed to verify if the etching was sufficient in order to define the pillars and the bottom. The results are shown in table 5.4. It can be seen that the pillars were slightly over-etched, which is a positive sign, and the total etch thickness of $T = 1456$Å surpassed the stack thickness of $T = 1200$Å, meaning the bottom was well defined and the etching process reached the oxide layer of the wafer.

Pillar Etch(Å)	Expected Value(Å)	Bottom Etch(μm)	Expected Value(Å)
782$\pm$61	600	674$\pm$26	600

Table 5.4: Profilometer Measurements - MTJ Pillars and Bottom

5.3.4 Vias Opening and Metallization (Steps 9 to 11)

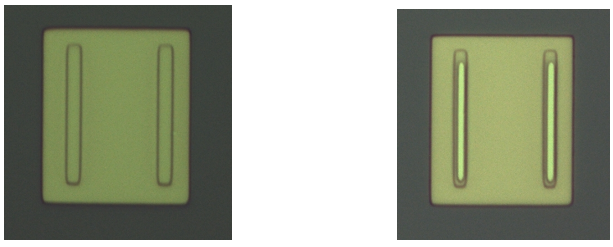


Figure 5.12: Closed Junction (Left) and Opened Junction (Right) after the Vias Opening Process

After the pasivation process, vias were opened trough the oxide layer in order to provide contact between the top electrode and the MTJ pillars. In this step of the process, alignment is crucial and needs to be done the smallest uncertainty possible. Unfortunately there were issues with the focusing of the laser in the period when the lithography was done, and as a consequence only half of the vias were opened. The evidence towards that come from the fact that the vias were either opened or closed along a certain lic of the laser. The dimensional results are shown in the table 5.5.

Average Height(μm)	Expected Value(μm)	Average Width(μm)	Expected Value(μm)
16.72±0.09	18	1.02±0.06	1

Table 5.5: Dimensional Measurements - Vias Openning

5.3.5 Top electrode and Passivation (Steps 12 to 14)

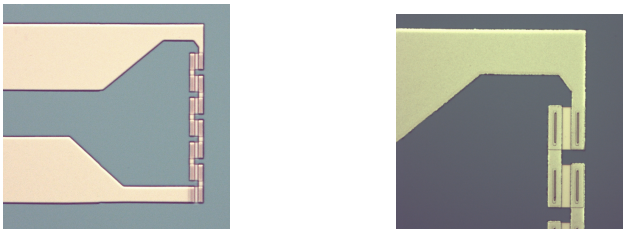


Figure 5.13: Sensor Overview (Left) and Junctions Overview (Right) after the Top Electrode Definition Process

After the vias opening, a layer of 3000Å of AlSiCu with a TiWn cap to protect from oxidation was deposited after photoresist coating and lithography. Liftoff process at the Wet Bench then used to define the top electrode. Afterwards a 4000Å layer of oxide is deposited using the Oxford Machine in order to protect the electrical vias. The dimensions of the features obtained are shown in the table 5.6.

Average Height(μm)	Expected Value(μm)	Average Width(μm)	Expected Value(μm)
56.8±0.11	58	8.14±0.05	7.5

Table 5.6: Dimensional Measurements - Top Electrode

5.3.6 Pads opening (Steps 15 to 16)

Figure 5.14: Ground Pad (Left) and Positive Pad (Right) after the Pad Opening Process

Finally the access to the electrical pads is done by performing lithography and etching trough the oxide layer using the SPTS machine. The pad dimensions are shown in table 5.7.

Average Height(μm)	Expected Value(μm)	Average Width(μm)	Expected Value(μm)
575.7±0.6	600	289.2±0.5	300

Table 5.7: Dimensional Measurements - Electrical Pads

5.3.7 Sample Dicing (Step 17)

After the fabrication process was completed, the sample was diced using the DAD Dicing Setup. The saw with $t = 60\mu m$ thickness was used in order to avoid damaging the sensors and provide a clean cut.

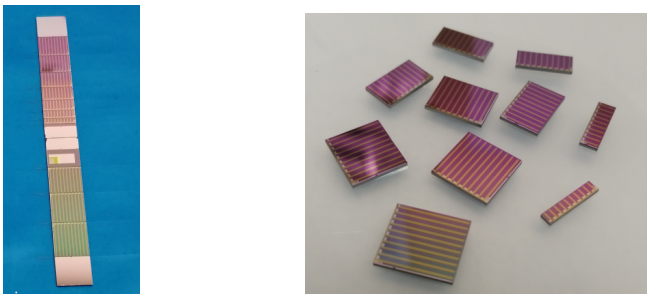


Figure 5.15: Sample during dicing (Left), sample after dicing (Right)

5.3.8 Thermal annealing (Steps 18 and 19)

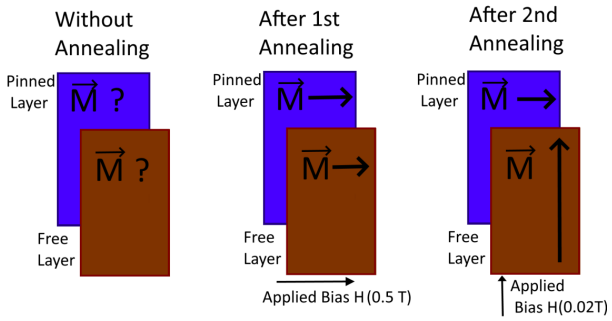


Figure 5.16: Schematic of The Effects of the 1st and 2nd Annealing on the Pinned and Free Layers

After dicing, annealing needs to be performed in order to align the free and pinned layers. The first annealing is performed for 2h at $T = 330$ºC under a field of $B = 0.5T$ in the sensible direction of the sensors in order to align the pinned layer (hard magnet). A second annealing is then performed for 2h at $T = 270$ºC using a field of $B = 0.02T$ in order to align the free layer (soft magnet). This second annealing is performed perpendicular to the first one. The process is illustrated in figure 5.16.

5.4 Sensor Characterization

The sensor characterization process for the fabrication sensors is described in the following subsections. First the sensor sensitivity around the center-line is measured and the linearity and coercivity are characterized. Afterwards the total magnetoresistance is measured.

Due to nano-fabrication mistakes, only 4 of the 10 fabricated sensor layers were working. This may be due to the fact that the Heidelberg DWL 2.0 laser setup used during lithography had some focusing issues during the week that the process was done, hence only some of the vias of the sensors were opened, allowing the connection between the junctions and the top electrode, as explained in section . Hence only these levels were characterized.

5.4.1 Sensor Sensitivity and Linearity

The magnetic fields produced by magnetic ink are very weak, of the order of $H = \pm 10Oe$ to $H = \pm 20Oe$, as shown in the simulations and experimental measurements using the scanner. Hence, it is of interest to characterizing the sensors in the linear region around the center-line, far away from the saturation region.

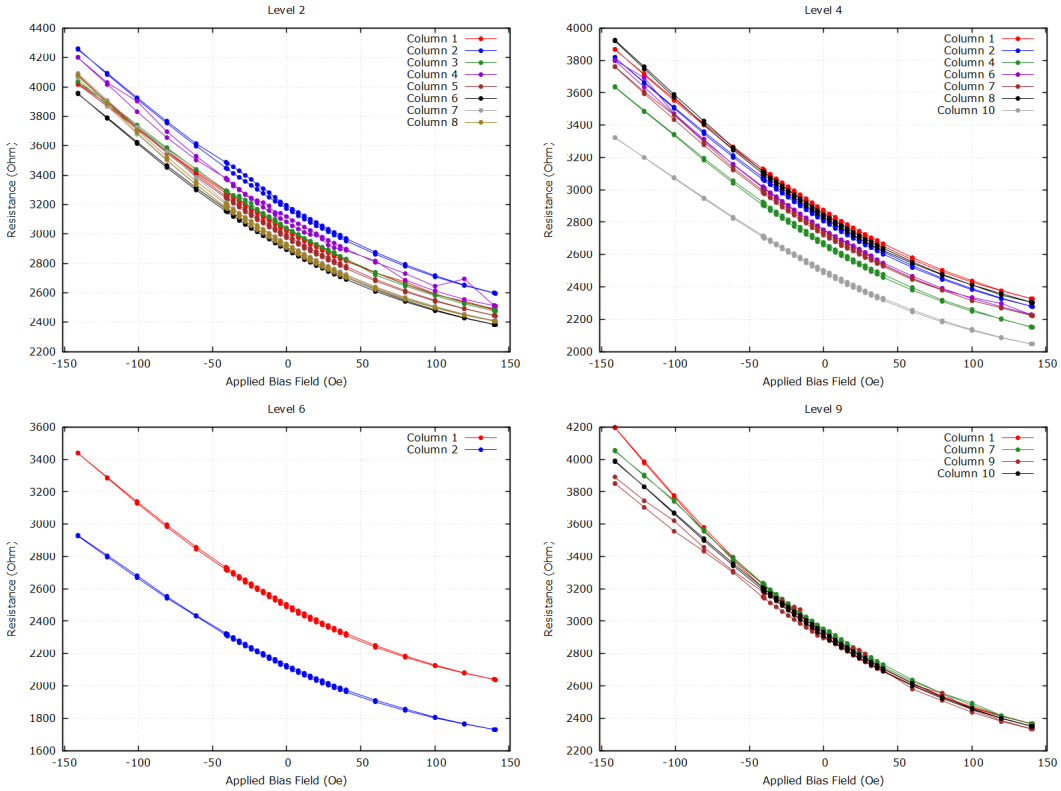


Figure 5.17: Magnetoresistance as a Function of the Applied Field for the 4 functioning levels

The 4 functioning levels were characterized using the ”140 Oe in-house setup” using a current of $I = 1\mu A$. The plot of the results is shown in figure 5.17. It can be observed that the sensors are very linear along the center-line and display similar behaviour. They also present very low coercivity. In the case of the $Level6$, some sensors were damaged during the handling process, hence less data is available. In the other levels, around 2 or 3 sensors present present half the resistivity of the other sensors, so they were not included in the plot. The results are summarized in the table 5.8.

The base resistance is defined as the resistance in the absence of external bias field. The uncertainty of the sensibility and base resistance is obtained from the dispersion of the different values for the sensors of different columns within the same level.

Level	Base Resistance(Ohm)	Sensibility(Ohm/Oe)	Coercivity(Oe)
2	3 028	5.73 ± 0.12	0.96
4	2 868	5.72 ± 0.15	1.00
6	2 498	5.01 ± 0.14	1.09
9	2 920	6.36 ± 0.20	1.06
Average	2 829	5.71 ± 0.31	1.03

Table 5.8: Sensibility, Coercivity and Resistance Measurements - MTJ Sensors

5.4.2 Magnetoresistance

In order to obtain the magnetoresistance of the sensors, the saturation region needs to be reached. For that purpose, the 4 functioning levels were characterized using the ”300 Oe Auto-Prober Setup” (which has a field range of [-300;300]Oe) using a current of $I = 1\mu A$.

Even in this case, the saturation region was not completely reached. One of the possible causes is

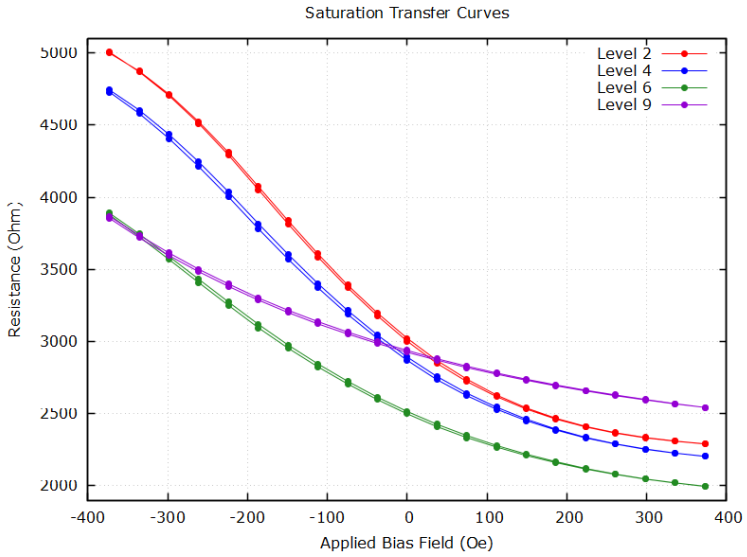


Figure 5.18: Magnetoresistance as a Function of the Applied Field for the 4 levels using the auto-prober

the fact that the annealing process was repeated twice (due to an error in the relative annealing angle between the first and second annealing step). Hence, this may have caused some loss of sensitivity in the sensors, meaning that the slope of the transference curve was lower. Hence a larger range of field values is required to saturate the sensors.

However, the measurements produce a good estimate of the maximum magnetoresistance of the sensors. Unfortunately the $2000Oe$ measuring setup which has a larger range of [-2000;2000]Oe was unavailable at the time of the measurements.

The results are shown in the figure 5.18 and summarized in the table 5.9. It can be observed that the sensors reach large values of magnetoresistance, but not very close to the expected value of $TMR = 175\%$. Only one sensor per level was measured, since that auto-prober can only read pads close to each other.

Level	$Rmin$(Ohm)	$Rmax$(Ohm)	ΔR(Ohm)	MR(%)
2	2 290	5 005	2 715	118.6
4	2 203	4 743	2 540	115.3
6	1 995	3 889	1 894	94.9
9	2 543	3 866	1 324	52.1
Average	2 257	4 380	2 118	95.2

Table 5.9: Magnetoresistance Measurements - MTJ Sensors

Chapter 6

System Assembly

In this chapter, the methods for assembling the Sensor Setup, the PCB and System Setup are shown, together with the misalignment measurements and the final system setup.

6.1 Mechanical Assembly

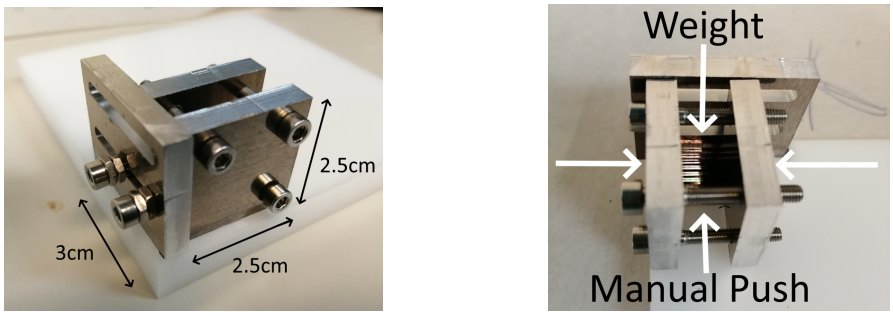


Figure 6.1: Overview of the Mechanical Setup Used to align the sensors (Left), Procedure (Right)

In order to align the sensor levels during the gluing process, a mechanical setup was design as shown in figure 6.1. The alignment process is shown on the right. The levels were glued one by one on top of each other using a small drop of $Loctite403$ glue between them.

Afterwards, the final sample aligned using the mechanical setup and the electrical test setup that was aligned manually were inspected under microscope. A comparison between the alignment using the manual and mechanical methods are shown in figure 6.2. The miss-alignment presented is of each level relative to the level above it, with level 10 being the bottom level.

Surprisingly, the results don't seem much better when using the mechanical setup for alignment. This could be from the fact that the measurements were done in a different way for the prototype, since the sample was much larger and it was hard to measure the miss-alignment under the microscope because of that. Hence the direct comparison is not very reliable.

6.2 PCB Soldering

The soldering of the PCB components was done at the soldering station of INESC-ID shown in figure 6.3. This is done before the sensor is installed since it is the most sensible component in order to avoid overheating it.

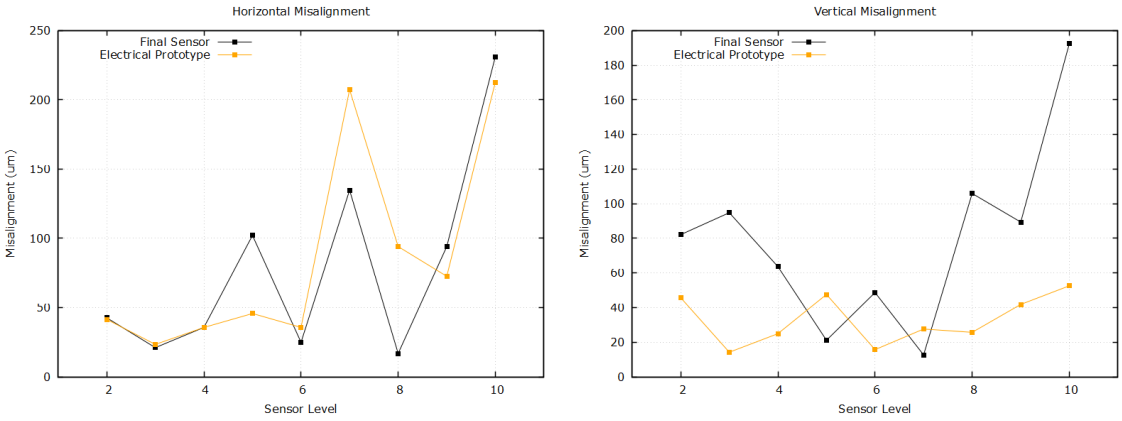


Figure 6.2: Horizontal Miss-alignment (Left) and Vertical Miss-alignment (Right)

Figure 6.3: Soldering Station (Left) and the Soldering of the Sample (Right)

Components such as resistance and condenser are soldered first, before soldering the components containing Integrated Circuits (ICs) such as the Multiplexers and Amplifiers.

6.3 Wirebonding

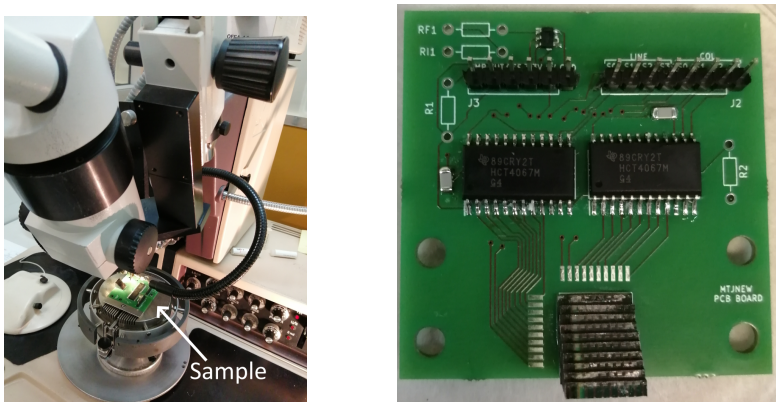


Figure 6.4: Wirebonding Station (Left) and the Final Result (Right)

After the electronic components are soldered, the sensor is finally glued to the PCB and Wirebonding was done at the Wirebonding Station at INESC-MN between the different levels and with the PCB pads. The process is shown in figure 6.4.

6.4 System Integration

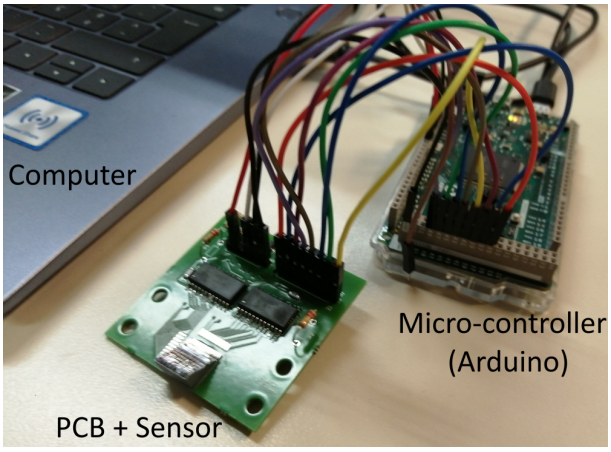


Figure 6.5: Final System Setup, showing the PCB with the sensors and the jumper connections to the the micro-controller

Finally the resistances are installed on the PCB, and jumpers are used to connect it to the Arduino Due, which is connected to a computer. Using the Matlab, the multiplexers are controlled using digital logic in order to choose the sensors to read. The value for the voltage for the DAC (Digital to Analog Converter) used in amplification is controlled via the Simulink Package, and the amplified signal from the sensors is imported trough the analog inputs of the Arduino. The result is shown in figure 6.5.

In this case, since less than half of the sensors were functioning, the permanent magnets were not installed and the setup was not tested on a sample. The simulations for the expected results were presented instead.

Chapter 7

Expected Results

Since it was not possible to obtain experimental measurements with a finished setup, simulations of the expected results shall be shown in this Chapter. First the case of 3×3 Pixel Matrices is explored, afterwards all the possible combinations of 2×2 Pixel Matrices are studied.

7.1 3x3 Pixel Matrix

Since the matrix fabricated was a 10×10 matrix, it would be ideal for the mapping of 3x3 pixel matrices, since we would have a resolutions of approximately 3×3 sensors per pixel. Therefore the simulations of the expected magnetic mapping for this case will be presented.

7.1.1 Detection Direction

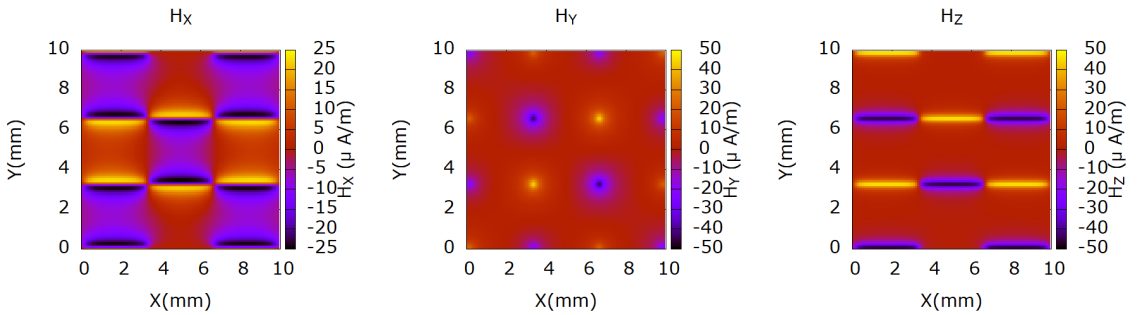


Figure 7.1: Mapping as a function of the Detection Direction (Magnetization X, $1cm^2$ sample, $D = 0.2mm$)

The simulations of the magnetic mapping as a function of the direction in which the magnetic field is being detected is shown in the figure 7.1. The magnetization is in the X direction. Hence, the H_X mapping gives a mapping of the bulk of the pixels, the H_Y component gives a mapping of the corners of the pixels, while the H_Z component gives a mapping of the edges of the pixels. The simulated sample of $10 \times 10mm$ is magnetized in the X direction and the detection is done at a distance of $0.2mm$.

As stated before, the only practical reading mode from an experimental point of view is the mapping of the magnetic field perpendicular to the magnetization. Hence, only the mapping of the H_Z component shall be studied in the following sections.

7.1.2 Mapping Contrast as a function of the Detection Distance

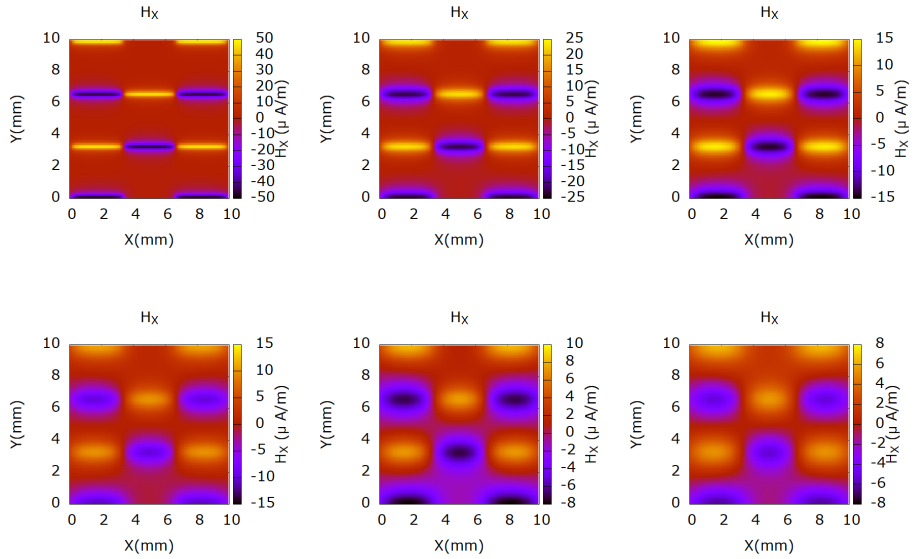


Figure 7.2: Perpendicular Mapping of the Edges as a function of the measuring distance

The simulated mapping of the perpendicular magnetic field H_Z as a function of the measuring distance is shown in figure 7.2. Since the magnetization is in the X direction, this mode of scanning produces a mapping of the lower and upper horizontal edges of the pixels. If the sample was to be rotated by 90º during the measurement process and then scanned again, a mapping of the vertical edges would be obtained, this producing a much more complete measurement with a much lower error value.

It can be observed that as the distance increases, the image becomes more blurred and the mapping of the edges becomes less defined.

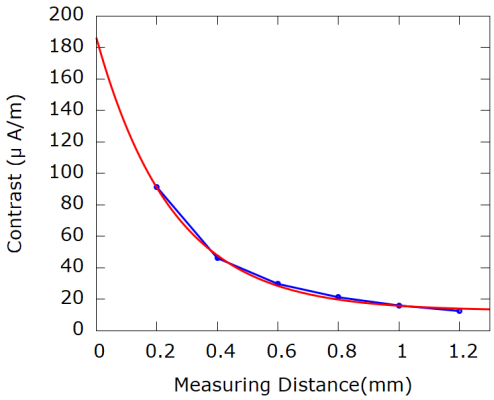


Figure 7.3: Mapping Contrast with Negative Exponential Fit

The algorithm that calculates the contrast is applied to the simulations to these simulations. The resulting plot is shown in figure 7.3. The equation used for fitting is shown in the expression 7.1 and the fitting parameters are given in table 7.1. It can be observed that this model produces a good fit to the data. However, the value for the c parameter does not converge to zero, meaning a larger ensemble of

simulations would be needed.

$$f(x) = a \cdot exp(-b \cdot x) + c \tag{7.1}$$

Parameter	a($\mu A/m$)	b(mm^{-1})	c($\mu A/m$)
Value	174 ± 10	4.0 ± 0.3	12.6 ± 1.6

Table 7.1: Fitting Parameters - Contrast Simulations 3x3 pixels

7.2 2x2 Pixel Matrix

In order to study in more detail the interaction between neighbouring pixels, the case of a 2×2 pixel matrix ($10mm \times 10mm$) needs to be studied. For the fabricated matrix, this would correspond to a large resolution of 5×5 sensors per pixel, so the data obtained would be very reliable.

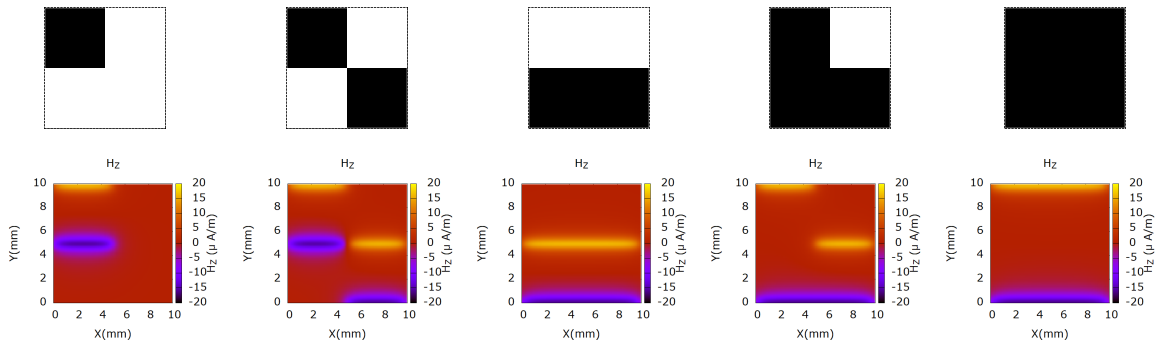


Figure 7.4: Possible Pixel Combinations (2x2 Matrix), and perpendicular field mapping at $D = 0.6mm$

The possible combinations of pixels are explored. For the case of 2 pixels, the case with diagonal and side-by-side positioning was studied. For the 3 and 4 pixel cases, only one meaningful arrangement is considered, since the re-combinations are redundant. These combinations are shown in figure 7.4.

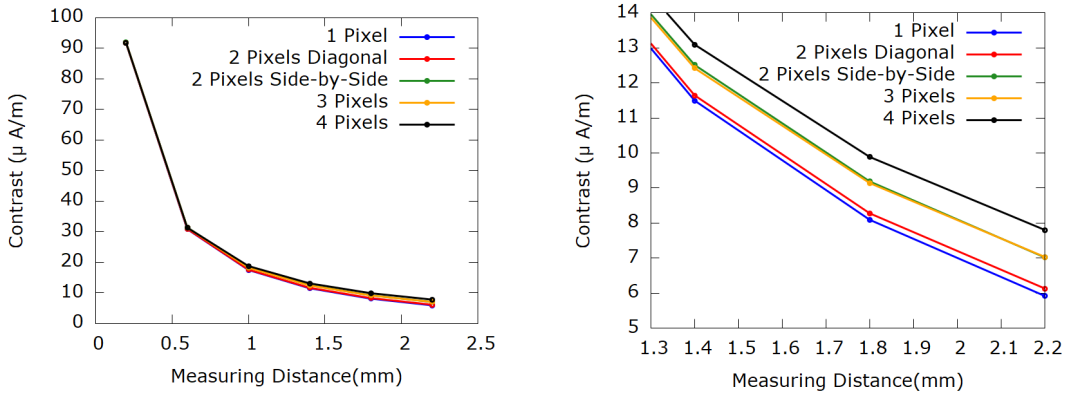


Figure 7.5: Contrast Plot for the different combination of pixels (Left), detailed view (Right)

In the figure 7.5 it can be observed that the contrast pattern is similar in the several combinations of pixels, with a divergence only occurring at larger measuring distance, where it is observed that the more pixels we have, the stronger the contrast.

Chapter 8

Conclusion

The subject of Magnetic QR Codes is a fascinating one, since it is still very unexplored. Devices based on this technology may prove very useful for security and counterfeiting applications in the near future. However, this is a new and emerging field, therefore is not much previous research to start on. This provides a great opportunity for researchers to produce and develop technologies within this field as an answer to the needs and requests of the industry for such.

The main objective of this thesis was to prove that it was possible to read 2D magnetic patterns printed with magnetic ink and to study how parameters such as the scanning distance, number of sensors per pixel, and concentration of pixels influenced the magnetic interference between them. The first objective was clearly was clearly achieved experimentally, while the second one was done using simulations.

First the subject of Barcodes and QR Codes was discussed in some detail, exploring the historical background, the applications and the utility of this thesis to this subject. The state of the art of the magnetic barcodes was also presented.

The theoretical background was then explored. The subject of magnetism was developed from the topic of magnetic fields in free space to the interaction between magnetic fields and matter. The subject of magnetoresistive sensors was then explored, with a specific focus on the tunneling magnetoresistive effect used in the devices fabricated during the thesis. The linearization procedures and the most important characterization parameters were presented, together with their typical ranges of values.

A simple yet effective mathematical model was then employed in order to simulate the magnetic field produced by magnetic QR codes. This model was confirmed experimentally and showed how the different directions of magnetization affected the reading. This model was then used to characterize the magnetic mapping as a function of the reading distance and other physical parameters such as the number of sensors per pixel. An algorithm for perpendicular reading was also developed for practical purposes.

The simulations of the field mapping produced by magnetic QR Codes and the results obtained from them, together with the decoding algorithm employed for perpendicular mapping represent one of the main contributions of this thesis, since the subject is very much unexplored, and no starting point was available the the beginning of the dissertation.

The proposed system setup was then presented. A sensor matrix consisting of layers of sensors stacked on top of each other was proposed, together with the electronic setup to read and amplify the signals from the sensors. The required multiplexing and amplifying circuits were described. A prototype consisting only of the electrical tracks was first fabricated in order to study the practicality of stacking several sensor levels on top of each other, and measure the resistance of the electrical tracks.

The final layered sensor matrix was then fabricated. The nanofabrication process only succeeded in

reaching its end on the third run, and only some of the sensor levels were functional. However, most of the sensors on those levels presented good transference curves and high percentage of magnetoresistance. An attempt to begin the nanofabrication again was not possible due to the lack of time.

Afterwards the sensor layers were alignment and glued on top of each other using a mechanical device. The PCB soldering and assembly was then done, followed by the wire-bonding of the sensor. The system was not tested due to the lack of time, as it was not expected to produce valuable useful mapping since some levels were not working.

Finally simulations of the expected results were shown. The contrast of the mapping was studied and the results obtained showed an interesting pattern, following a negative exponential decay. Future work could include reproducing the nano-fabrication process correctly and testing the entire setup. The deadline of the thesis did not allow a repetition of the nanofabrication process.

Bibliography

[1] Roger C. Palmer. *The Bar Code Book*.

[2] Denso ADC. Qr code essentials. 2011.

[3] Sofia Alexandra Cruz Abrunhosa. Magnetoresistive sensors for industrial positioning applications. Master's thesis, Instituto Superior Técnico, 2018.

[4] J. M. Coey. *Magnetism and magnetic materials - Section 2.2*. Cambridge University Press, 2010.

[5] Kannan M. Krishnan. *Fundamentals and Applications of Magnetic Materials - Chapter 1*. Oxford University Press, 2016.

[6] J. M. Coey. *Magnetism and magnetic materials - section 1.2*. Cambridge University Press, 2010.

[7] Kannan M. Krishnan. *Fundamentals and Applications of Magnetic Materials - Chapter 3*. Oxford University Press, 2016.

[8] J. M. Coey. *Magnetism and magnetic materials - section 2.1.1 and 2.4*. Cambridge University Press, 2010.

[9] Kannan M. Krishnan. *Fundamentals and Applications of Magnetic Materials - section 8.7.1.4*. Oxford University Press, 2016.

[10] S. Cardoso P. Freitas, R. Ferreira and F. Cardoso. Magnetoresistive sensors. *Journal of Physics: Condensed Matter*, (19(16):165221), 2007.

[11] J. Valadeiro J. Amaral P. P. Freitas A. V. Silva, D. C. Leitao and S. Cardoso. Linearization strategies for high sensitivity magnetoresistive sensors. *The European Physical Journal Applied Physics*, (72(1):10601), 2015.

[12] Denso-Wave. Qr code—about 2d code. 2016.

[13] Denso-Wave. Information capacity and versions of qr code. 2016.

[14] ISO/IEC 18004:2006(E). Data encoding; table 3 – number of bits in character count indicator for qr code. 205.

[15] Gustave Reed, Irving S.; Solomon. Polynomial codes over certain finite fields. *Journal of the Society for Industrial and Applied Mathematics*, 1960.

[16] W. Wesley Peterson. *Error Correcting Codes3*. MIT Press, 1961.

[17] Orli Sharaby. Form meets function: Extreme makeover qr code edition. 2012.

[18] Russ Cox. Qart codes: How to make pictures with qr codes, part ii. 2015.

[19] Manjurul Ahsan Gupta, Kishor Datta and Stefan Andrei. Extending the storage capacity and noise reduction of a faster qr-code. *BRAIN. Broad Research in Artificial Intelligence and Neuroscience 9.1*, pages 59–71, 2018.

[20] R. Ferreira P. P. Freitas and S. Cardoso. Spintronic sensors. *Proceedings of the IEEE*, (104(10): 1894–1918), 2016.

[21] M. Julliere. Tunneling between ferromagnetic films. *Physics letters A*, (54(3):225–226), 1975.

[22] P. Freitas A. Guedes, M. Mendes and J. Martins. Study of synthetic ferrimagnet-synthetic antiferromagnet structures for magnetic sensor application. *Journal of applied physics*, (99(8):08B703), 2006.

Appendix A

Simulations C++ Code

A.1 Magnetic Field Simulations

```
#include <iostream>
#include <fstream>
using namespace std;

struct vector
{
public:
        float x = 0.0f, y = 0.0f, z = 0.0f;

        //default constructor
        vector() : x(0), y(0), z(0) {}
        //constructor
        vector(float _x, float _y, float _z) : x(_x), y(_y), z(_z) {}
        //copy constructor
        vector(const vector& v) : x(v.x), y(v.y), z(v.z) {}

        void set(float _x, float _y, float _z)
        {
                this->x = _x;
                this->y = _y;
                this->z = _z;
        }
        void reset()
        {
                this->x = 0;
                this->y = 0;
                this->z = 0;
        }
        void print()
        {
                cout << "(" << x << ";" << y << ";" << z << ") ";
```

```
}
float mag()
{
        return sqrt(x * x + y * y + z * z);
}
vector norm()
{
        float r = 1 / mag();
        return vector(x * r, y * r, z * r);
}
vector operator + (const vector& rhs)
{
        return vector(this->x + rhs.x, this->y + rhs.y, this->z + rhs.z);
}
vector operator - (const vector& rhs)
{
        return vector(this->x - rhs.x, this->y - rhs.y, this->z - rhs.z);
}
vector operator * (const float& rhs)
{
        return vector(this->x * rhs, this->y * rhs, this->z * rhs);
}
vector operator / (const float& rhs)
{
        return vector(this->x / rhs, this->y / rhs, this->z / rhs);
}
void operator += (const vector& rhs)
{
        this->x += rhs.x;
        this->y += rhs.y;
        this->z += rhs.z;
}
void operator -= (const vector& rhs)
{
        this->x -= rhs.x;
        this->y -= rhs.y;
        this->z -= rhs.z;
}
float dot(const vector& rhs)
{
        return this->x * rhs.x + this->y * rhs.y + this->z * rhs.z;
}
vector cross(const vector& rhs)
{
        return vector(this->y * rhs.z - this->z * rhs.y, this->z * rhs.x - this-
}
```

```
};

int main()
{
        //QR Codes
        ifstream qr_file;
        qr_file.open("pattern4.txt");
        ofstream qr_hx, qr_hy, qr_hz; //4x4 per Pixel
        qr_hx.open("pattern4x_1.0mm.txt");
        qr_hy.open("pattern4y_1.0mm.txt");
        qr_hz.open("pattern4z_1.0mm.txt");

        ofstream qr_hx_2x2, qr_hx_1x1; //2x2, 1x1 per Pixel
        qr_hx_2x2.open("qr_magnx_hx_2x2_w10mm_d0.2mm.txt");
        qr_hx_1x1.open("qr_magnx_hx_1x1_w10mm_d0.2mm.txt");

        ofstream qr_hx_2x2_avg, qr_hx_1x1_avg; //2x2, 1x1 per Pixel
        qr_hx_2x2_avg.open("qr_magnx_hx_2x2avg_w10mm_d0.2mm.txt");
        qr_hx_1x1_avg.open("qr_magnx_hx_1x1avg_w10mm_d0.2mm.txt");

        // 2D PART (res: 1unit = 0.1mm) (10E-4m)  (QR 100x0.1mm = 10mm width)
H(micro A/m)
        float d = 10; //plane distance
        float PI = 3.1416;
        float magn_perm = (4 * PI) * 10E-7; // Magnetic Permeability
        float c = 1 / (4 * PI);//constant in front of equation
        float Mag_0 = 605000; //Volum Magnetization M = 605(kA/m) (CoPt perm magnet)
        float Mag_ink = 33000; //Volum Magnetization M = 33(kA/m) (magnetic ink)
        float t_struct = 0.1E-6; //CoPt struct thickness T = 0.1 m
        float t_ink = 8.9E-6; //magn ink thickness T = 8.9 m
        float dv = 10E-15; // unit volume (m^3)
        float factor = 10E3;
        float mag = Mag_0 * dv * 10E6 * factor; //magnetization/unit volume (micro Am^2)
        vector r1r2;
        //magnetized plane OK

        int qr[100][100]; //10x10mm qr pattern
        vector r1[100][100];
        vector m[100][100];

        //loading qr pattern
        for (int i = 99; i >= 0; i--)
        {
                for (int j = 0; j < 100; j++)
                        qr_file >> qr[i][j];
        }
```

```
        //setting position matrix OK

        for (int i = 0; i < 100; i++)
        {
                for (int j = 0; j < 100; j++)
                        r1[i][j].set(i, j, 0);
        }


        //setting magnetization vectors X - QR
        for (int i = 0; i < 100; i++)
        {
                for (int j = 0; j < 100; j++)
                        if(qr[i][j]==1)
                                m[i][j].set(mag, 0, 0);
        }


        //detection plane OK
        vector r2[100][100];
        vector h[100][100];

        //setting position matrix

        for (int i = 0; i < 100; i++)
        {
                for (int j = 0; j < 100; j++)
                        r2[i][j].set(i, j, d);
        }


        //calculating h matrix - QR codes
for (int i = 0; i < 100; i++)
{
        for (int j = 0; j < 100; j++)
        {
                for (int k = 0; k < 100; k++)
                {
                        for (int n = 0; n < 100; n++)
                        {
                                r1r2 = r2[i][j] - r1[n][k];
                                h[i][j] += r1r2 * c * 3 * r1r2.dot(m[n][k]) / pow(r1r2.m
                        }
                }
        } cout << i << "%" << endl;
}
```

```
//printing matrix file - QR codes 4x4
for (int i = 0; i < 100; i++)
{
        for (int j = 0; j < 100; j++)
        {
                qr_hx << r2[i][j].x / 10 << "␣" << r2[i][j].y / 10 << "␣" << h[i][j].x <
                qr_hy << r2[i][j].x / 10 << "␣" << r2[i][j].y / 10 << "␣" << h[i][j].y <
                qr_hz << r2[i][j].x / 10 << "␣" << r2[i][j].y / 10 << "␣" << h[i][j].z <
        }
        cout << endl;
}

//printing matrix file - QR codes 2x2
for (int i = 1; i < 100; i+=2)
{
        for (int j = 1; j < 100; j+=2)
        {
                qr_hx_2x2 << r2[i][j].x / 10 << "␣" << r2[i][j].y / 10 << "␣" << h[i][j]
        }
        cout << endl;
}

//printing matrix file - QR codes 1x1
for (int i = 2; i < 100; i += 4)
{
        for (int j = 2; j < 100; j += 4)
        {
                qr_hx_1x1 << r2[i][j].x / 10 << "␣" << r2[i][j].y / 10 << "␣" << h[i][j]
        }
        cout << endl;
}

//printing matrix file - QR codes 2x2 AVERAGE
for (int i = 1; i < 100; i += 2)
{
        for (int j = 1; j < 100; j += 2)
        {
                qr_hx_2x2_avg << r2[i][j].x / 10 << "␣" << r2[i][j].y / 10 << "␣" << (h[
        }
        cout << endl;
}

//printing matrix file - QR codes 1x1 AVERAGE
for (int i = 3; i < 100; i += 4)
{
```

```
        for (int j = 3; j < 100; j += 4)
        {
                qr_hx_1x1_avg << r2[i][j].x / 10 << " " << r2[i][j].y / 10 << " " << (h[
        }
        cout << endl;
}

qr_hx.close();
qr_hy.close();
qr_hz.close();
qr_hx_2x2.close();
qr_hx_1x1.close();
qr_hx_2x2_avg.close();
qr_hx_1x1_avg.close();
qr_file.close();

        return 0;
}
```

A.2 QR Image File to Text File for Simulations Conversion

```
#include<opencv2/opencv.hpp>
#include<iostream>
#include<fstream>
using namespace std;
using namespace cv;
int main()
{
        Mat img = imread("QR_CODE.png", IMREAD_GRAYSCALE);
        namedWindow("image", WINDOW_NORMAL);
        imshow("image", img);
        waitKey(0);

        ofstream file("QR_CODE_for_simulations.txt");
        ofstream file2("QR_CODE_for_error_comparisson.txt");
        uchar darray[1000][1000]; //  create 2D array

        for (int i = 0; i < img.cols; i++)
        {
                for (int j = 0; j < img.rows; j++)
                {
                        darray[i][j] = img.at<uchar>(i, j);
                }
        }
        //display matrix screen
        for (int i = 0; i < 100; i++)
```

```
        {
                for (int j = 0; j < 100; j++)
                {
                        if ((int)darray[i][j] != 0)
                                cout << 0;
                        else cout << 1;
                }
                cout << endl;
        }

        //print matrix file1
        for (int i = 0; i < 300; i+=3)
        {
                for (int j = 0; j < 300; j+=3)
                {
                        if ((int)darray[i][j] != 0)
                                file << 0 << ' ';
                        else file << 1 << ' ';
                }
                file << endl;
        }

        //print matrix file2
        for (int i = 0; i < 300; i += 12)
        {
                for (int j = 0; j < 300; j += 12)
                {
                        if ((int)darray[i][j] != 0)
                                file2 << 0 << ' ';
                        else file2 << 1 << ' ';
                }
                file2 << endl;
        }


        return 0;
}
```

A.3 Pixel Reconstruction Algorithm from the Edge Mapping

```
#include<iostream>
#include <fstream>

using namespace std;

int main()
```

```
{
    int bulk = 0;
        float min=0, max=0, base=0;
    float thresh_pos = 0, thresh_neg = 0;
        float qr_hz[100][100];
    float qr_edge[100][100];
    float qr_bulk[100][100];
    float QR_corrected[25][25];
    float percentage = 0.65;
    float correction_threshold = 0.45;
    float x_offset = 1, y_offset = 1;

    ifstream input_simulation;
    input_simulation.open("qr_magnx_hz_Am2_w10mm_d0.2mm.txt");
    ofstream edge_simulation;
    edge_simulation.open("qr_magnx_hz_Am2_w10mm_d0.2mm_edge.txt");
    ofstream bulk_simulation;
    bulk_simulation.open("qr_magnx_hz_Am2_w10mm_d0.2mm_bulk_tr0.65.txt");
    ofstream QR_corrected_output;
    QR_corrected_output.open("qr_magnx_hz_Am2_w10mm_d0.2mm_bulk_tr0.65_corrected_tr0.45.

    //Simulation File
    float column1, column2;
    float x = 100, y=100;
if (input_simulation.is_open())
{
    //Loading Input File
    for (int j = 0; j < 100; j++)
    {
        for (int i = 0; i < 100; i++)
        {
            input_simulation >> column1 >> column2 >> qr_hz[i][j];
        }
    }

    //Calculate min, max values
    base = 0;
    min = qr_hz[0][0];
    max = qr_hz[0][0];

    for (int i = 0; i < y; i++)
    {
        for (int j = 0; j < x; j++)
        {
            if (qr_hz[i][j] < min)
                min = qr_hz[i][j];
            else if (qr_hz[i][j] > max)
```

```
                max = qr_hz[i][j];
            }
        }

        //Calculate Threshold
        thresh_neg = min + percentage * (base - min);
        thresh_pos = max - percentage * (max - base);


        //Edge Mapping
        for (int i = 0; i < y; i++) {
            for (int j = 0; j < x; j++) {
                if (qr_hz[i][j] > thresh_pos)
                    qr_edge[i][j] = 1;
                else if (qr_hz[i][j] < thresh_neg)
                    qr_edge[i][j] = -1;
                else
                    qr_edge[i][j] = 0;
            }
        }

        //Bulk Mapping
        for (int i = 0; i < x; i++)
        {
            for (int j = 0; j < y-1; j++)
            {
                if (qr_edge[i][j] == -1)
                    bulk = 1;
                if (qr_edge[i][j] == 1)
                    bulk = 0;
                qr_bulk[i][j] = bulk;
            }
        }
    }

    else
        cout << "Cannot open Input File" << endl;

    cout << "Base: " << base << " Min: " << min << " Max: " << max << endl;
    cout << "Thresh_pos: " << thresh_pos << " Thresh_neg: " << thresh_neg << endl;

    //Printing Edge File
    for (int i = 0; i < y; i++) {
        for (int j = 0; j < x; j++) {
            edge_simulation << qr_edge[i][j] << " ";
        }
        edge_simulation << endl;
```

```
}

//Printing Bulk File
for (int i = 0; i < y; i++) {
    for (int j = 0; j < x; j++) {
        bulk_simulation << qr_bulk[i][j] << " ";
    }
    bulk_simulation << endl;
}

//Correction Algorithm
int n, m;

    //Calculate #Pixels Filled
    for (int n = 0; n < 25; n++)
    {
        for (int m = 0; m < 25; m++)
        {
            QR_corrected[n][m] = 0;
            for (int i = n * 4; i < (n + x_offset) * 4; i++)
            {
                for (int j = m * 4; j < (m + y_offset) * 4; j++)
                {
                    if (qr_bulk[i][j] == 1)
                        QR_corrected[n][m] += qr_bulk[i][j];
                }
            }
        }
    }

    //Normalize QR Corrected
    for (int n = 0; n < 25; n++)
    {
        for (int m = 0; m < 25; m++)
        {
            QR_corrected[n][m] /= 16;
        }
    }

    //Correction Algorithm
    for (int n = 0; n < 25; n++)
    {
        for (int m = 0; m < 25; m++)
        {
            if (QR_corrected[n][m] > correction_threshold)
                QR_corrected[n][m] = 1;
            else
```

```
                QR_corrected[n][m] = 0;
            }
        }

        //Printing Corrected QR File
        for (int j = 24; j >=0; j--) {
            for (int i = 0; i < 25; i++) {
                QR_corrected_output << QR_corrected[i][j] << " ";
            }
            QR_corrected_output << endl;
        }

//Closing Files Simulation
input_simulation.close();
edge_simulation.close();
bulk_simulation.close();


        //Experimental Data File
ifstream input;
input.open("QR1_5x5cm_d0.7mm_res1mm_fullscan57x55_Oe.txt");
ofstream edge_file;
edge_file.open("QR1_5x5cm_d0.7mm_res1mm_fullscan57x55_Oe_edge_tr0.9.txt");
ofstream bulk_file;
bulk_file.open("QR1_5x5cm_d0.7mm_res1mm_fullscan57x55_Oe_bulk_tr0.9.txt");
ofstream QR_experimental_corrected;
QR_experimental_corrected.open("QR1_5x5cm_d0.7mm_res1mm_fullscan57x55_Oe_bulk_tr0.9_corr
percentage = 0.9;
correction_threshold = 0.8;
x_offset = 0, y_offset = 0;
x = 55, y = 57;
        if (input.is_open())
        {
            //Loading Input File
            for (int i = 0; i < y; i++)
            {
                for (int j = 0; j < x; j++)
                {
                    input >> qr_hz[i][j];
                }
            }

            //Calculate min, max values
            base = qr_hz[0][0];
            min = qr_hz[0][0];
            max = qr_hz[0][0];
```

```
    for (int i = 0; i < y; i++)
    {
        for (int j = 0; j < x; j++)
        {
            if (qr_hz[i][j] < min)
                min = qr_hz[i][j];
            else if (qr_hz[i][j] > max)
                max = qr_hz[i][j];
        }
    }

    //Calculate Threshold
    thresh_neg = min + percentage * (base - min);
    thresh_pos = max - percentage * (max - base);


    //Edge Mapping
    for (int i = 0; i < y; i++) {
        for (int j = 0; j < x; j++) {
            if (qr_hz[i][j] > thresh_pos)
                qr_edge[i][j] = 1;
            else if (qr_hz[i][j] < thresh_neg)
                qr_edge[i][j] = -1;
            else
                qr_edge[i][j] = 0;
        }
    }

    //Bulk Mapping

    for (int j = 0; j < x; j++)
    {
        for (int i = 0; i < y-1; i++)
        {
            if (qr_edge[i][j] == -1)
                bulk = 1;
            if (qr_edge[i][j] == 1)
                bulk = 0;
            qr_bulk[i][j] = bulk;
        }
    }



}
else
    cout << "Cannot open Input File" << endl;
```

```
cout << "Base: " << base << " Min: " << min << " Max: " << max << endl;
cout << "Thresh_pos: " << thresh_pos << " Thresh_neg: " << thresh_neg << endl;

//Printing Edge File
for (int i = 0; i < y; i++) {
    for (int j = 0; j < x; j++) {
        edge_file << qr_edge[i][j] << " ";
    }
    edge_file << endl;
}

//Printing Bulk File
for (int i = 0; i < y; i++) {
    for (int j = 0; j < x; j++) {
        bulk_file << qr_bulk[i][j] << " ";
    }
    bulk_file << endl;
}


//Correction Algorithm
x_offset = 0;
y_offset = 3;
//Calculate #Pixels Filled
for (int n = 0; n < 25; n++)
{
    for (int m = 0; m < 25; m++)
    {
        QR_corrected[n][m] = 0;
        for (int i = n * 2+y_offset; i < (n + 1) * 2+y_offset; i++)
        {
            for (int j = m * 2; j < (m + 1) * 2; j++)
            {
                if (qr_bulk[i][j] == 1)
                    QR_corrected[n][m] += qr_bulk[i][j];
            }
        }
    }
}

//Normalize QR Corrected
for (int n = 0; n < 25; n++)
{
    for (int m = 0; m < 25; m++)
    {
        QR_corrected[n][m] /= 4;
```

```
        }
    }

    //Correction Algorithm
    for (int n = 0; n < 25; n++)
    {
        for (int m = 0; m < 25; m++)
        {
            if (QR_corrected[n][m] > correction_threshold)
                QR_corrected[n][m] = 1;
            else
                QR_corrected[n][m] = 0;
        }
    }

    //Printing Corrected QR File
    for (int i = 0; i<25; i++)
    {
        for (int j = 0; j < 25; j++) {
            QR_experimental_corrected << QR_corrected[i][j] << " ";
        }
        QR_experimental_corrected << endl;
    }

    //Closing Files
    input.close();
    edge_file.close();
    bulk_file.close();


        return 0;
}
```

A.4 Error Calculation Algorithm

```
#include<iostream>
#include <fstream>

using namespace std;

int main()
{
    float QR_original[25][25];
    float QR_simulated[25][25];

    ifstream input_simulation, input_measurements;
```

```
input_simulation.open("qr_magnx_hz_Am2_w10mm_d0.2mm_bulk_tr0.65_corrected_tr0.45.txt
input_measurements.open("QR1_5x5cm_d0.7mm_res1mm_fullscan57x55_Oe_bulk_tr0.9_correct
ifstream input_original;
input_original.open("QR_Original_25x25.txt");
ofstream output_simulated, output_measurements;
output_simulated.open("qr_magnx_hz_Am2_w10mm_d0.2mm_bulk_tr0.65_corrected_tr0.45_err
output_measurements.open("QR1_5x5cm_d0.7mm_res1mm_fullscan57x55_Oe_bulk_tr0.9_correc

//Simulation File
float errors = 0;
float x = 25, y = 25;
if (input_simulation.is_open() && input_original.is_open())
{
    //Loading Original Input File
    for (int i = 0; i < x; i++)
    {
        for (int j = 0; j < y; j++)
        {
            input_original >> QR_original[i][j];
        }
    }

    //Loading Simulation Input File
    for (int i = 0; i < x; i++)
    {
        for (int j = 0; j < y; j++)
        {
            input_simulation >> QR_simulated[i][j];
        }
    }

    //Calculating Error
    for (int i = 0; i < x; i++)
    {
        for (int j = 0; j < y; j++)
        {
            if (QR_simulated[i][j] != QR_original[i][j])
                errors += 1;
        }
    }
    output_simulated << "Errors: " << errors << endl;
    cout << "Errors: " << errors << endl;

    //Normalizing Error
    errors = errors / (25 * 25);
    output_simulated << "Error: " << errors * 100 << "\%" << endl;
    cout << "Error: " << errors * 100 << "\%" << endl;
```

```
    }
    else
        cout << "Cannot open Simulation Input File" << endl;

    //Simulation File
    errors = 0;
    x = 25, y = 25;
    if (input_measurements.is_open() /*&& input_original.is_open()*/)
    {
        //Loading Measurements Input File
        for (int i = 0; i < x; i++)
        {
            for (int j = 0; j < y; j++)
            {
                input_measurements >> QR_simulated[i][j];
            }
        }

        //Calculating Error
        for (int i = 0; i < x; i++)
        {
            for (int j = 0; j < y; j++)
            {
                if (QR_simulated[i][j] != QR_original[i][j])
                    errors += 1;
            }
        }
        output_measurements << "Errors: " << errors << endl;
        cout << "Errors: " << errors << endl;

        //Normalizing Error
        errors = errors / (25 * 25);
        output_measurements << "Error: " << errors * 100 << "\%" << endl;
        cout << "Error: " << errors * 100 << "\%" << endl;
    }
    else
        cout << "Cannot open Measurement Input File" << endl;

    //Closing Files
    input_simulation.close();
    input_original.close();
    output_simulated.close();

    return 0;
}
```

A.5 Mapping Contrast Calculation Algorithm

```
#include<iostream>
#include <fstream>

using namespace std;

int main()
{
    float min = 0, max = 0, base = 0;
    float contrast[7];
    float input[100][100];


    ifstream input1, input2, input3, input4;
    input1.open("qr_magnx_hx_Am2_w10mm_d0.2mm.txt");
    input2.open("qr_magnx_hx_Am2_w10mm_d0.3mm.txt");
    input3.open("qr_magnx_hx_Am2_w10mm_d0.4mm.txt");
    input4.open("qr_magnx_hx_Am2_w10mm_d0.5mm.txt");
    ofstream contrast_out;
    contrast_out.open("contrast_qr.txt");

    //Matrix Dimensions
    float column1, column2;
    float x = 100, y = 100;

    //Input1
    if (input1.is_open())
        if (input1.is_open())
        {
            //Loading Input File
            for (int j = 0; j < 100; j++)
            {
                for (int i = 0; i < 100; i++)
                {
                    input1 >> column1 >> column2 >> input[i][j];
                }
            }

            //Calculate min, max values
            min = input[0][0];
            max = input[0][0];

            for (int i = 0; i < y; i++)
            {
                for (int j = 0; j < x; j++)
                {
```

```
                if (input[i][j] < min)
                    min = input[i][j];
                else if (input[i][j] > max)
                    max = input[i][j];
            }
        }
        //Calculate Contrast
        contrast[1] = max - min;
    }
//Input2
if (input2.is_open())
    if (input2.is_open())
    {
        //Loading Input File
        for (int j = 0; j < 100; j++)
        {
            for (int i = 0; i < 100; i++)
            {
                input2 >> column1 >> column2 >> input[i][j];
            }
        }

        //Calculate min, max values
        min = input[0][0];
        max = input[0][0];

        for (int i = 0; i < y; i++)
        {
            for (int j = 0; j < x; j++)
            {
                if (input[i][j] < min)
                    min = input[i][j];
                else if (input[i][j] > max)
                    max = input[i][j];
            }
        }
        //Calculate Contrast
        contrast[2] = max - min;
    }
//Input3
if (input3.is_open())
    if (input3.is_open())
    {
        //Loading Input File
        for (int j = 0; j < 100; j++)
        {
            for (int i = 0; i < 100; i++)
```

```
            {
                input3 >> column1 >> column2 >> input[i][j];
            }
        }

        //Calculate min, max values
        min = input[0][0];
        max = input[0][0];

        for (int i = 0; i < y; i++)
        {
            for (int j = 0; j < x; j++)
            {
                if (input[i][j] < min)
                    min = input[i][j];
                else if (input[i][j] > max)
                    max = input[i][j];
            }
        }
        //Calculate Contrast
        contrast[3] = max - min;
    }
//Input4
if (input4.is_open())
    if (input4.is_open())
    {
        //Loading Input File
        for (int j = 0; j < 100; j++)
        {
            for (int i = 0; i < 100; i++)
            {
                input4 >> column1 >> column2 >> input[i][j];
            }
        }

        //Calculate min, max values
        min = input[0][0];
        max = input[0][0];

        for (int i = 0; i < y; i++)
        {
            for (int j = 0; j < x; j++)
            {
                if (input[i][j] < min)
                    min = input[i][j];
                else if (input[i][j] > max)
                    max = input[i][j];
```

```
                }
            }
            //Calculate Contrast
            contrast[4] = max - min;
        }

    //Printing Contrast File
    contrast_out << 0.2 << " " << contrast[1] << endl;
    contrast_out << 0.3 << " " << contrast[2] << endl;
    contrast_out << 0.4 << " " << contrast[3] << endl;
    contrast_out << 0.5 << " " << contrast[4] << endl;

    //Closing Files
    input1.close();
    input2.close();
    input3.close();
    input4.close();
    contrast_out.close();

    return 0;
}
```

Appendix B

Sensor Fabrication Runsheet

Runsheet – MTJ Fabrication

Process Start: 1 / 10 / 2021 **Process Finish:** 26 / 10 / 2021

Responsible: Maria Carvalho, Rita Macedo, Susana Freitas Alberto Nicolicea

STEP 0: *TMR Stack Deposition* ____/____/_____

Sample ID: **Machine:** N3600/ N8800

Run #_____

Stack: (Ta5/Ru10)x3/T a5/Ru 5/MnPt 20/Co80Fe20 2.2/Ru 0.7/ CoFeB 1.8/ MgO 1.5/ CoFeB 3/ Ru 0.2/ NiFe 6/ MnIr 6 / NiFe 5/ Ru 5/ Ta5 / Ru 10

(RA ~ 2.42 kOhm e TMR ~ 175 %)

Total Thickness (without TiWN): 1200 nm

STEP 2: 1st *Lithography – Pillar (L1)* *1/10/2021*

Operator: Alberto/José Bernardo **Machine:** DWL

1.1 Vapor Prime 30 min (Recipe – 0)

1.2 Coat 1.5 µm Photoresist (Recipe 6/2)

1.3 Lithography

Map: AMSION_C

SIZE: 37200 x 20708 um2

DIESIZE: 37200 x 20708 um2

Inverted mask

Masks: MTJNEW_L1_PILLAR

X=168; Y=20054				
...				
X=168; Y=10054				
X=168; Y=5054				
X=168; Y=54	X=5168; Y=54	X=10168; Y=54	...	X=35168; Y=54

[X_0: _168___; Y_0: ___54__] µm (From bottom left corner)

Energy: 95.9 **Focus:** 0 **Power:** 84mW

Die dimensions: (X, Y) = 37200 x 20708 µm^2

Total Time: _____

1.4 Develop (Recipe 6/2)

Optical Inspection | Comments: OK

- Defined structures ☐
- Structures dimensions correct? ☐

STEP 3: *MTJ Definition by Ion Milling* *6/10/2021*

Operator: Alberto/Rita Macedo **Machine:** N3600

Total stack thickness: 1210 Å **Thickness to etch:** 600 Å **Etch rates (for ref) 60º: 1,1** Å/s **Etch rate (for ref) 30º: 1,2** Å/s

Sequence name: Etch Junction 2

Time steps: 2x170'' @ pan = 60°

Time steps: 2x150'' @ pan = 30°

Base Pressure: 10^-7 Torr **Run # 3871**

Conditions:

Assist Gun		Power (W)	V+ (V)	I+ (mA)	V- (V)	I- (mA)	Ar Flux (sccm)	W.P. (Torr)
Set Point		180	650	-	350	-	16	-
Read Values	60°	184	656	105	349	6	16	
	30°	183	657	107	349	6	16	

Neutralizer: 0 mA/ 3 sccm

Inspection | Comments OK

Runsheet – Short run Acceina Run #3

STEP 4: Resist Strip *7/10/2021*

Operator: Alberto **Machine:** Wet bench

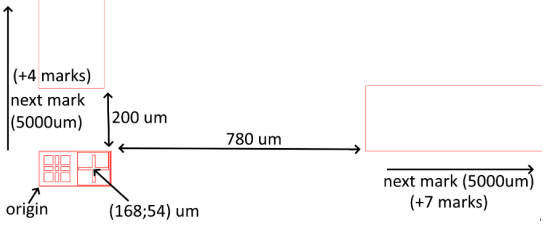
Conditions: Microstrip + Temperature: 65 °C + Ultrasounds

Cleaning: IPA rinse / DI water rinse / Dry with compressed air gun

Total Time: 2h30m

Optical Inspection | Profilometry | Comments OK

STEP 5: 2nd *Lithography – Bottom electrode Definition (L2)* *8/10/2021*



Operator: Alberto/Manuela **Machine:** DWL

1.1 Vapor Prime 30 min (Recipe – 0)

1.2 Coat 1.5 µm Photoresist (Recipe 6/2)

1.3 Lithography

Map: AMSION_C

SIZE: 37200 x 20708 um2

DIESIZE: 37200 x 20708 um2

Masks: MTJNEW_L2_BOTTOM **(/h1)** **Inverted mask**

X=168; Y=20054				
...				
X=168; Y=10054				
X=168; Y=5054				
X=168; Y=54	X=5168; Y=54	X=10168; Y=54	...	X=35168; Y=54

Energy: 95,9 **Focus:** -15 **Power:** 84mW

Total Time: 30min

Alignment cross marks: [X_0: 168; Y_0: 54] [Δx;Δy] = [5000; 5000] µm

1.4 Develop (Recipe 6/2)

Optical Inspection | Comments: OK

STEP 6: *Bottom electrode Definition by Ion Milling* *11/10/2021*

Operator: Alberto/Rita Macedo **Machine:** N3600

Etch rates (for reference) 45º: 1,4 Å/s

Sequence name: Etch_Stack_45Deg

Time steps: 3 x 170'' @ pan = 45°

Base Pressure: 10^-7 Torr **Run # 3896**

Conditions:

Neutralizer: 0 mA/ 3 sccm

Assist Gun	Power (W)	V+ (V)	I+ (mA)	V- (V)	I- (mA)	Ar Flux (sccm)	W.P. (Torr)
Set Point	180	650	-	350	-	16	-
Read Values							

Optical Inspection | Comments: OK

STEP 7: Resist Strip *11/10/2021*

Operator: Alberto **Machine:** Wet bench

Conditions: Microstrip + Temperature: 65 °C + Ultrasounds

Cleaning: IPA rinse / DI water rinse / Dry with compressed air gun

Total Time: 3h00

Optical Inspection | Profilometry | Comments: **OK**

STEP 8: *Passivation – SiO_2 Deposition* *12/10/2021*

Operator: Alberto/Fernando **Machine:** Oxford

Recipe: SiO_2 300ºC

Thickness to deposit: 2700Å **Dep rate:** 78.4 nm/min

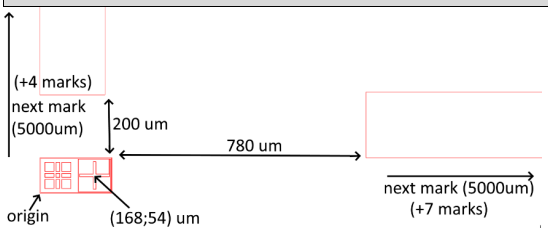
Rate: 13 Å/s **Time:** 3m27s

Conditions:

	Total time (min)	Gas flow (sccm)			Temperature (°C)	Source Power (W)
		N_2O	SiH_4	N_2		
Set Point		710	8.5	161.5	300	20
Read Values		709	8.4	161.2	299	20

Inspection | Comments **OK**

STEP 9: *3rd Lithography – Vias opening* *13/10/2021*



Operator: Alberto/José Bernardo **Machine:** DWL

1.1 Vapor Prime 30 min (Recipe – 0)

1.2 Coat 1.5 µm Photoresist (Recipe 6/2)

1.3 Lithography

Map: AMSION_C

Masks: MTJNEW_L3_VIAS **(/h1)** **Non-Inverted mask**

X=168; Y=20054				
...				
X=168; Y=10054				
X=168; Y=5054				
X=168; Y=54	X=5168; Y=54	X=10168; Y=54	...	X=35168; Y=54

Energy: 95.9 **Focus:** -5 **Power:** 84mW

Total Time: 30min

Alignment cross marks:

L1: [X;Y] = [168 ; 54] µm; [Δx;Δy] = [5000; 5000] µm

L2: [X;Y] = [390 ; 54] µm; [Δx;Δy] = [5000; 5000] µm

Runsheet – Short run Acceina Run #3

X=390; Y=20054				
...				
X=390; Y=10054				
X=390; Y=5054				
X=390; Y=54	X=5390; Y=54	X=10390; Y=54	...	X=35390; Y=54

1.4 Develop (Recipe 6/2)

Optical Inspection | Comments: **OK**

STEP 10: *Vias opening by Reactive Ion Etching* *13/10/2021*

Operator: Alberto/Manuela **Machine:** SPTS

Thickness to etch: 2700Å **Etch rate:** 161 nm/min **Overetch:** ____%

Recipe: SiO2_Vias_etch_30s **(4 x (30s etch + 5 min cooldown in LL))** **Conditions:**

	P_{source} (W)	VP-P_{source} (V)	P_{platen} (W)	VP-P_{platen} (V)	APC (%)	Gas Flux (sccm)		Pressure (mTorr)	He Press. (Torr)
						CF4	O_2		
Set point	750		100			60	60	12	5
Read Val.	750/1	2788	103/5	580	9.3	60	60	12	5

Optical Inspection | Profilometry | Comments: **OK**

STEP 11: Resist Strip *14/10/2021*

Operator: _____ **Machine:** wet bench

Conditions: Microstrip + Temperature: 65 °C + Ultrasounds

Cleaning: IPA rinse / DI water rinse / Dry with compressed air gun

Total Time: 3h40m

Optical Inspection | Profilometry | Comments **OK**

STEP 12: *4th Lithography – Top electrode* *14/10/2021*

Operator: Alberto/José Bernardo **Machine:** DWL

1.1 Vapor Prime 30 min (Recipe – 0)

1.2 Coat 1.5 µm Photoresist (Recipe 6/2) **()**

1.3 Lithography

Map: AMSION_C

SIZE: 37200 x 20708 um2 DIESIZE: 37200 x 20708 um2

Non-Inverted mask **Masks:** MTJNEW_L4_METAL(/h1)

X=168; Y=20054				
...				
X=168; Y=10054				
X=168; Y=5054				
X=168; Y=54	X=5168; Y=54	X=10168; Y=54	...	X=35168; Y=54

[X_0: _168___; Y_0: ___54__] µm (From bottom left corner)

Energy: 100.9 **Focus:** -5 **Power:** 84mW

Die dimensions: (X, Y) = 37200 x 20708 µm²

Total Time: 37min

1.4 Develop (Recipe 6/2)

Optical Inspection | Comments: **OK**

STEP 13: Metallization *14/10/2021*

Operator: Alberto/Fernando **Machine:** N7000

Deposition rates (ref): AlSiCu: _____ A/s TiWN:_____ A/s

Sequence name: Metallization 4500:

1. Soft etch (120s) 40 w/60 w: Mod 2, F.3
2. AlSiCu 400nm (_____s) 2 kW: Mod 4, F.1
3. TiW(N_2) 50nm (_____s): Mod 3, F.19

Base Pressure: _____ Torr **Run #22527** **Conditions:**

Module Function		Time (s)	Substrate Table		Plasma Source	Target DC			Ar Flux (sccm)	N_2 Flux (sccm)	Pressure (Torr)
			RF (W)	DC Bias (V)	RF (W)	Power (kW)	V (V)	I (A)			
1	Set Point	60	40	-	60	-	-	-	50	-	$3x10^{-3}$
	Read Values					-	-	-		-	
2	Set Point	72	-	-	-	2	-	-	50	-	$3x10^{-3}$
	Read Values		-	-	-					-	
3	Set Point	26	-	-	-	0.5	-	-	50	10	$3x10^{-3}$
	Read Values		-	-	-						

Optical Inspection | Comments: **OK**

STEP 14: Oxide Lift-Off *14/10/2021*

Operator: Alberto **Machine:** wet bench

Conditions: Microstrip + Temperature: 65 °C + Ultrasounds

Cleaning: IPA rinse / DI water rinse / Dry with compressed air gun

Total Time: 3h40m

Optical Inspection | Profilometry | Comments **OK**

STEP 15: *Passivation – SiO_2 Deposition* *20/10/2021*

Operator: Alberto/Fernando **Machine:** Oxford

Recipe: SiO_2 300º (5m07s, 13ª/s)

Thickness to deposit: 4000Å **Dep rate:** 78.4 nm/min **Conditions:**

	Total time (min)	Gas flow (sccm)			Temperature (°C)	Source Power (W)
		N_2O	SiH_4	N_2		
Set Point		710	8.5	161.5	300	20
Read Values						

Optical Inspection | Comments: OK

STEP 16: 5th *Lithography – Pads opening* *20/10/2021*

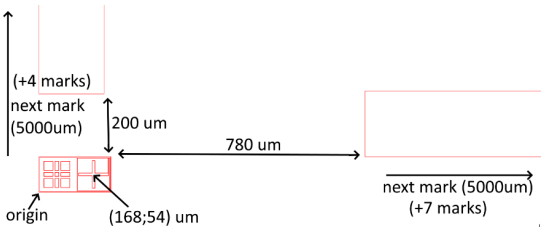
Operator: Alberto/Virginia Soares **Machine:** DWL

1.1 Vapor Prime 30 min (Recipe – 0)

1.2 Coat 1.5 µm Photoresist (Recipe 6/2)

1.3 Lithography

Map: AMSION_C



Runsheet – Short run Acceina Run #3

SIZE: 37200 x 20708 um2 DIESIZE: 37200 x 20708 um2

Non-Inverted mask **Masks:** MTJNEW_L5_PADS (/h1)

X=168; Y=20054				
...				
X=168; Y=10054				
X=168; Y=5054				
X=168; Y=54	X=5168; Y=54	X=10168; Y=54	...	X=35168; Y=54

[X_0: _168___; Y_0: ___54__] µm (From bottom left corner)

Energy: 95.9 **Focus:** 0 **Power:** 84mW

Die dimensions: (X, Y) = 37200 x 20708 $µm^2$

Total Time: 10min

1.4 Develop (Recipe 6/2)

Optical Inspection | Comments: **OK**

STEP 17: *Pads opening by Reactive Ion Etching* *21/10/2021*

Operator: Alberto/Manuela **Machine:** SPTS

Thickness to etch: 4000Å **Etch rate:** 161 nm/min **Overetch:** ____%

Recipe: SiO2_Vias_etch_112s **(6 x (30s etch + 5 min cooldown in LL)** **Conditions:**

	P_{source} (W)	VP-P_{source} (V)	P_{platen} (W)	VP-P_{platen} (V)	APC (%)	Gas Flux (sccm)		Pressure (mTorr)	He Press. (Torr)
						CF4	O_2		
Set point	750		100			60	60	12	5
Read Values									

Optical Inspection | Comments: **OK**

STEP 18: Resist Strip *21/10/2021*

Operator: Alberto **Machine:** wet bench

Conditions: Microstrip + Temperature: 65 °C + Ultrasounds

Cleaning: IPA rinse / DI water rinse / Dry with compressed air gun

Total Time: _____

Optical Inspection | Profilometry | Comments: **OK**

STEP 20: Dicing *22/10/2021*

Operator: Alberto/Virginia Soares **Machine:** DAD Dicing Saw

Saw used: 60 um

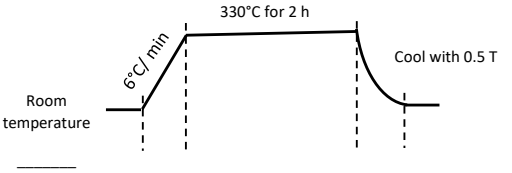
Die dimensions: (X, Y) = 37200 x 20708 $µm^2$

STEP 21: First Thermal annealing *26/10/2021*

Operator: Alberto/Pedro Araujo **Machine:** Turbomagnet Annealing

Magnetic Field Applied: 0.5 T

Direction: Parallel to detection direction

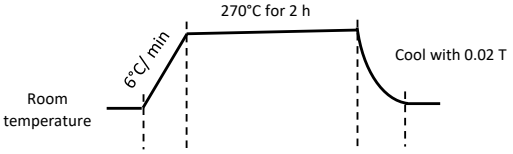


STEP 22: Second Thermal annealing *26/10/2021*

Operator: Alberto/Pedro Araujo **Machine:** Turbomagnet Annealing

Magnetic Field Applied: 0.5 T

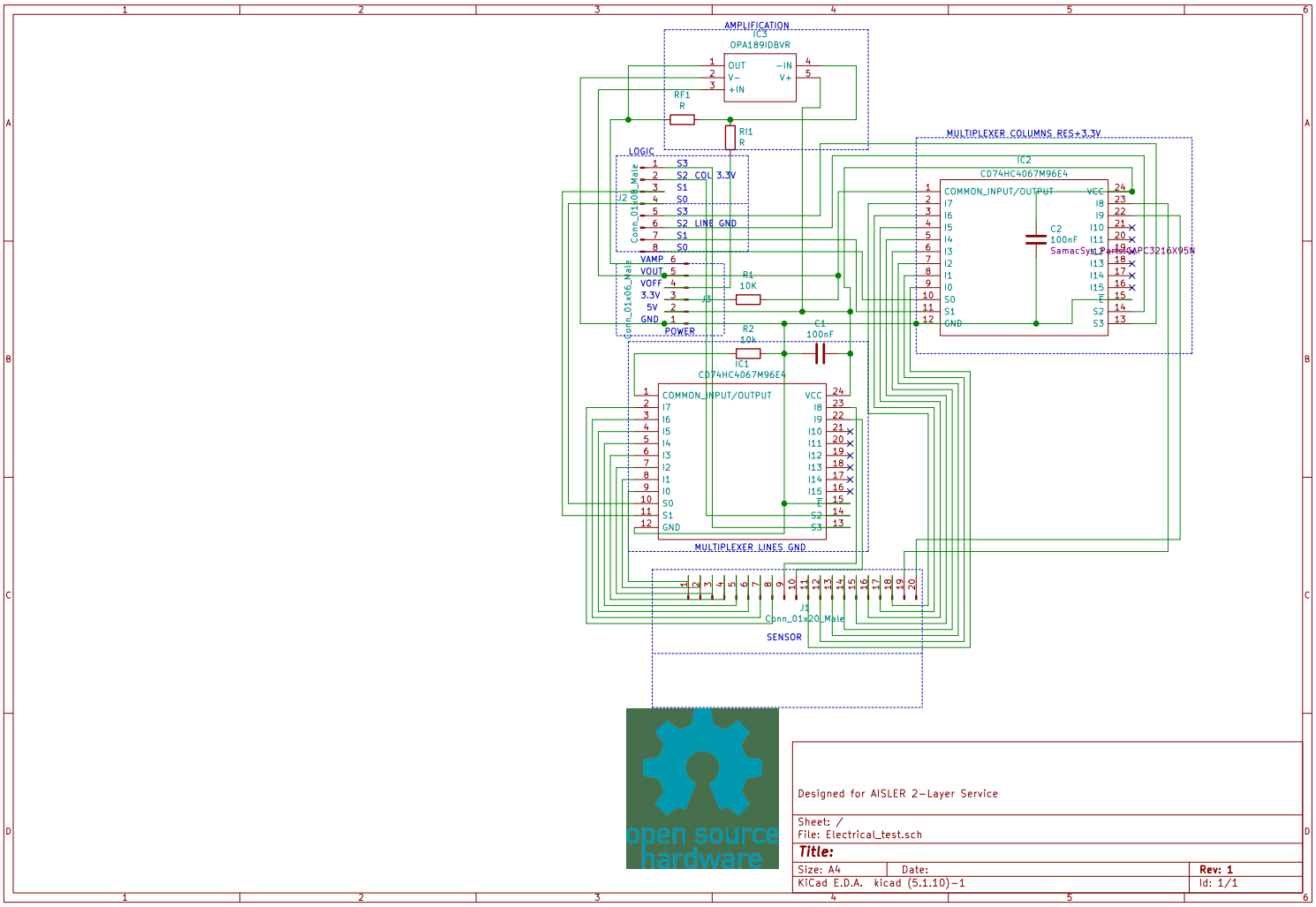
Direction: Perpendicular to detection direction



7

Appendix C

PCB Schematic and Layout


AMPLIFICATION
IC3
OPA189IDBVR
OUT
V-
+IN
-IN
V+
RF1
R
RI1
R
LOGIC
J2
Conn_01x08_Male
S3
S2 COL
3.3V
S1
S0
S3
S2 LINE
GND
S1
S0
POWER
J3
Conn_01x06_Male
VAMP
VOUT
VOFF
3.3V
5V
GND
R1
10K
R2
10k
C1
100nF
IC1
CD74HC4067M96E4
COMMON_INPUT/OUTPUT
I7
I6
I5
I4
I3
I2
I1
I0
S0
S1
GND
VCC
I8
I9
I10
I11
I12
I13
I14
I15
E
S2
S3
MULTIPLEXER LINES GND
MULTIPLEXER COLUMNS RES+3.3V
IC2
CD74HC4067M96E4
C2
100nF
SamacSys_Parts:SAPC3216X95N
J1
Conn_01x20_Male
SENSOR
Designed for AISLER 2-Layer Service
Sheet: /
File: Electrical_test.sch
Title:
Size: A4
Date:
Rev: 1
KiCad E.D.A. kicad (5.1.10)-1
Id: 1/1

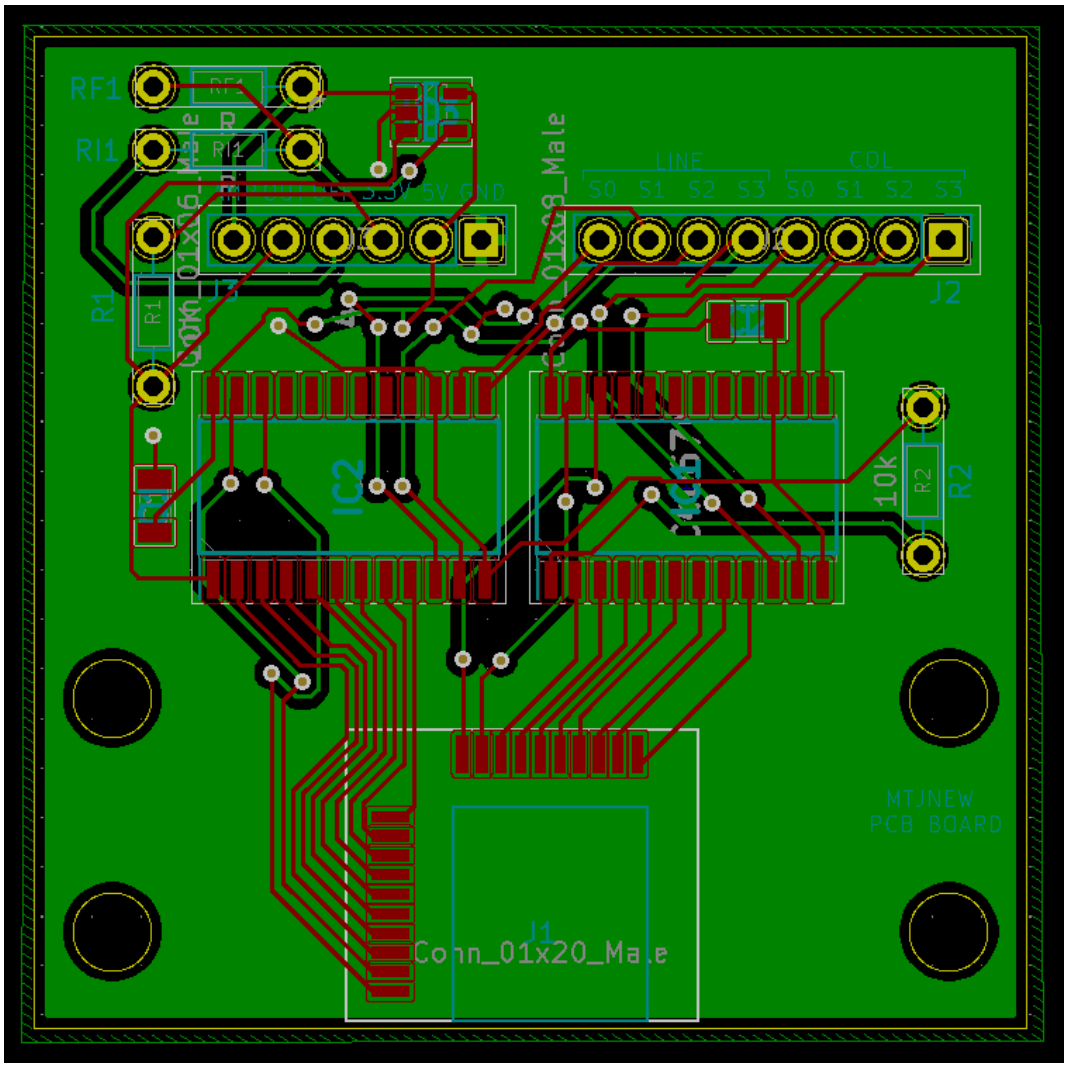


Figure C.1: PCB Design