

MAGNETIC LOCALIZATION SYSTEM BASED ON ZYNQ-7010 FPGA HARDWARE PLATFORM

Submitted by

Tang Yuchao

Xidian University

School of Electronic Engineering

The final year project work was carried out under the 3+1+1 Educational Framework at National University of Singapore (Suzhou) Research Institute

May 2018

Abstract

We Are What We Eat: The Recipe of Success

by

CHICKEN Rice

Doctor of Philosophy in Cooking Science

National University of Singapore

In the detection of gastrointestinal tract, the traditional cable endoscope will bring great pain to patients. The development of wireless capsule endoscopy not only alleviates the sufferings of patients, but also achieves better examination of gastrointestinal tract. When using the wireless capsule endoscopy, the location parameters and orientation parameters of the capsule in the body need to be accurately obtained, in order to obtain the location parameters and orientation parameters of the focus tissue by the capsule endoscopy. And these parameters facilitate the doctor to diagnose accurately. Thus, the accurate and fast positioning of capsules is a critical issue. In this paper, a magnetic localization algorithm is proposed, which eliminates the noise from vertical geomagnetic field, proposes a hierarchical localization algorithm with weighting. In this paper, the magnetic field intensity model is established based on the magnetic dipole model. For the locating optimization, interior point method is used, and for the calibration, LM algorithm is used. The hardware of the magnetic localization system contains LSM303D magnetic sensor array, ZYBO-7010 FPGA embedded Board and PCB design of the magnetic sensor.

This paper proposes a novel hierarchical localization algorithm with weighting. By predicting the layer the target locate roughly and weighting, the positioning error is effectively reduced, even if the number of magnetic sensors is little. In this paper, both the sixteen sensors system and thirty-two sensors system perform the accurate and efficient localization results. Besides, in order to eliminate the error caused by

the geomagnetic field, the vertical geomagnetic field intensity is ignored because in most human activities, the axis of the upper part of body is point to the center of the earth.

Consequently, the average position error with hierarchical localization algorithm is $6.73mm$ and the orientation error is 6.01° under the thirty-two sensors system. Therefore, the hierarchical localization algorithm with weighting can greatly improved the accuracy of the localization results.

Keywords: Wireless capsule endoscope; Magnetic localization system; LM algorithm; Interior point method; ZYBO-7010 FPGA board; PCB design

Acknowledgments

At the point of finishing this paper, I'd like to express my sincere thanks to all those who have lent me hands in the course of my writing this paper. First of all, I'd like to take this opportunity to show my sincere gratitude to my supervisor, Prof. Guo Yongxin, who has given me so much useful advices on my FYP and writing. Secondly, I'd like to express my gratitude to my PhD senior, Shao Guoliang, who offered me references and information on time and helped me deal with my problems during the investigation. Last but not least, I'd like to thank my teammates, Tang Lin and Dai Qiqi, for their strong cooperation, encouragement and support. Without their help, it would be much harder for me to finish the FYP and this paper.

Contents

Abstract	ii
Acknowledgments	iv
List of Figures	viii
List of Tables	x
1 Introduction	1
1.1 Research Background and Significance	1
1.1.1 Development of Wireless Capsule Endoscopy	1
1.1.2 Development of Magnetic Localization Technology	2
1.2 Literature Review	3
1.2.1 Mathematical Model and Localization Algorithms	3
1.2.2 Model Design	6
2 Building the Hardware Platform	9
2.1 The Selection of Localization Target	9
2.2 LSM303D Sensor	10
2.3 I2C Serial Bus	14
2.3.1 Overview	14
2.3.2 Protocol	15
2.4 ZYBO-7010 SoC Board	15

2.5	The IDEs of Xilinx®	17
2.5.1	Xilinx Vivado	17
2.5.2	Xilinx SDK	18
2.6	Implementation of Hardware Platform Building	20
2.6.1	Reading Sensors Data	20
2.6.2	Building The Experiment Platform	23
2.6.3	Construction of Sixteen-Sensor Localization System	25
2.6.4	PCB Design	27
3	Implementation of Localization Algorithms	30
3.1	Magnetic Localization Algorithms	32
3.1.1	Localization Algorithm	32
3.1.2	Optimization Algorithm	33
3.2	Calibrations	36
3.2.1	Calibration of B_T	36
3.2.2	Calibration of The Position and Orientation of Sensor	37
3.2.3	Sensor Position Calibration	37
3.2.4	Sensor Orientation Calibration	38
3.2.5	Correction of The Geomagnetic Field	38
3.3	Simulation of The Localization System	39
3.4	Results In Sixteen-Sensor Localization System	41
4	The Improvement of Algorithm and Model	45
4.1	A Model of Thirty-two Sensors	45
4.2	Results of The Improved Model	45
5	Summary	48
5.1	Conclusion	48
5.2	Future Work	48

Bibliography	50
A. The Pseudo-code for Reading Four Sensors Data	54

List of Figures

1.1	BT	4
1.2	grid1	7
1.3	sensor	8
1.4	software	8
2.1	Schematic	10
2.2	magnet	11
2.3	LSM303D Sensor	12
2.4	I2C Sequence	15
2.5	zybo	16
2.6	vivado	18
2.7	SDK	20
2.8	One Sensor Block	21
2.9	(a) Connect the sensor to FPGA board;	22
2.10	(b) Send the x, y, and z axes magnetic field intensity data to PC.	22
2.11	Four Sensors Block	22
2.12	(a) Connect four sensors to FPGA board;	23
2.13	(b)The magnetic field intensity data of four sensors.	23
2.14	Sixteen Sensors Block	24
2.15	(a) The experiment platform made by acrylic board.	24
2.16	(b)The components of the platform.	24
2.17	platform	25

2.18	sixteen system	26
2.19	circuit	28
2.20	(a) The PCB design diagram of LSM303D.	29
2.21	(b)The PCB circuit board of LSM303D.	29
3.1	Activities	39
3.2	Spiral Path	40
3.3	Rectangular Path	40
3.4	Position error of 16 sensors after calibration	41
3.5	Orientation error of 16 sensors after calibration	42
3.6	Position error of 16 sensors by hierarchical localization algorithm	43
3.7	Orientation error of 16 sensors by hierarchical localization algorithm	44
4.1	Thirty-two-sensor System	46
4.2	Position error of 32 sensors	47
4.3	Orientation error of 32 sensors	47

List of Tables

2.1	Pin Description	13
3.1	Comparison Between Before and After Using Hierarchical Algorithm . .	43

Chapter 1

Introduction

1.1 Research Background and Significance

1.1.1 Development of Wireless Capsule Endoscopy

To exam a patient's internal stomach and intestines for an accurate diagnosis, endoscopy is used. Whereas, the traditional endoscopy, using a flexible tube with a lens entering your body from throat or anus, is an invasive procedure, because of the painful and uncomfortable feeling. However, a new type of endoscopy comes, in order to change the way of endoscopic inspection. It is the capsule endoscopy.

Compared to the traditional endoscopy, capsule endoscopy is a safer method to determine the unknown cause of the problems of the stomach and intestines [1]. Mentally, capsule endoscopy can release patients' fear of endoscopy. In conventional way, the endoscope is inserted directly into the organ, and during this process, a patient is suffering the pain and according to the mechanism of the body, the rejection occurs automatically. It is a kind of negative experiences for a person. On the contrary, the capsule endoscopy is minimally invasive. According to Given Imaging Ltd., the inventor of a capsule endoscopy, the capsule weighs less than four grams and is about the size of a vitamin [2]. Therefore, patients are easy to swallow it as a pill without any pain or rejection. Physically, the capsule can prevent patients from

CHAPTER 1. INTRODUCTION

getting injured or being infected during the inspection. Owing to the insertion of a tube into organ, the patient's stomach is in the risk of bleeding and they will easily get infected if the endoscope is not disinfected completely [3]. And there exist other risks, such as puncture of esophagus, stomach lining, colon or intestines, dehydration and adverse reaction to sedation, which can be avoided by capsule endoscopy.

In photography, the capsule endoscopy has a wider view which can provides more detailed images and tissue coverage. A capsule endoscope is capable of implementing stereoscopic photography, with which protruding states are easily recognized, for a nearby subject. By contrast, the traditional endoscope can only implement planar imaging, consisting of a tube with a lens at the top. According to the stereoscopic photography, a doctor is more likely to determine the location and the source of the problems inside the patient's body and then make a correct diagnosis after analyzing the images. Besides, the capsule allows doctors to see the entire small intestine, which cannot be done with traditional endoscopy[4].

Undeniably, there are still some disadvantages of using capsule endoscopy. In accordance with Erica Sanderson, an editor of HealthCentral.Com, when the capsule stays in the digestive tract for more than two weeks, the patient will get the capsule retention. And if a patient has skin irritation after swallowing the capsule, the patient's skin will get mild redness. However, the increasing use of capsule endoscopy is still the irresistible trend, with the development of advanced technology and the high demands for pleasing treatments.

Finally, in order to release the patients from the pain of experiencing the traditional endoscopy, introducing capsule endoscopy is a good way to figure out the problem.

1.1.2 Development of Magnetic Localization Technology

Magnetic tracking technique uses one or more magnets as the excitation source, which generates a magnetic field that can be detected and measured by magnetic sensors. With these sensing signals, position and orientation can be estimated. In

CHAPTER 1. INTRODUCTION

contrast to other state-of-the-art tracking technologies such as mechanical optical tracking or ultrasonic tracking, magnetic tracking is emerging to provide an occlusion-free tracking scheme for the estimation of full pose (position and orientation) of various instruments. Compared with optical tracking techniques, this occlusion-free feature brings substantial benefits for intracorporeal applications, which are typically lacking of direct line-of-sights between the base frames to the tracked targets.

Efficient magnetic tracking techniques are significant for further development of computer-assisted diagnosis, interventions, and surgeries, especially in tracking flexible wireless endoscopic devices.

1.2 Literature Review

1.2.1 Mathematical Model and Localization Algorithms

For the localization algorithms, I analyze the paper *Investigation of the Relationship Between Tracking Accuracy and Tracking Distance of a Novel Magnetic Tracking System (IEEE SENSORS JOURNAL, VOL. 17, NO. 15, AUGUST 1, 2017)* [5]. In this paper, the authors tend to find the relationship between the localization accuracy and the distance between the magnet and the sensor array by both simulations and experiments. Before implement the simulations and experiments, we need to know the mathematical model and the basic localization algorithms.

1.2.1.1 A. Mathematical Model

As the paper says, the magnetic field intensity of the cylindrical magnet can be represented by a dipole model because the magnet's pose parameters based on dipole model are reliable compared with the quadrupole extension model[6].

As shown in Figure 1.1, (a, b, c) is the center of the magnet and (x_i, y_i, z_i) is the position of the i -th sensor. \mathbf{H} is a unit vector which represents the magnet's orientation from south pole to north pole. F_i is the vector from (a, b, c) to $(x_i, y_i,$

CHAPTER 1. INTRODUCTION

z_i). The magnetic field intensity B_i at point (x_i, y_i, z_i) is listed as below.

$$B_i = B_{i_x}i + B_{i_y}j + B_{i_z}k = B_T \left(\frac{3(\mathbf{H} \cdot \mathbf{F}_i)\mathbf{F}_i}{R_i^5} - \frac{\mathbf{H}}{R_i^3} \right) (i = 1, 2, \dots, N). \quad (1.1)$$

Where N represents the total number of the sensors been used; B_T is a constant related to the magnet[7] and defined as:

$$B_T = \frac{\mu_r \mu_0 \pi \delta^2 L M_0}{4\pi} \quad (1.2)$$

Where μ_r represent the relative permeability of the medium ($\mu_r \approx 1$ in the air); μ_0 is the air magnetic permeability ($\mu_0 = 4\pi \cdot 10^{-7}(T \cdot m/A)$); δ and L denote the radius and length of the permanent magnet respectively; $M_0(A/m)$ is the magnetization on the magnet surface.

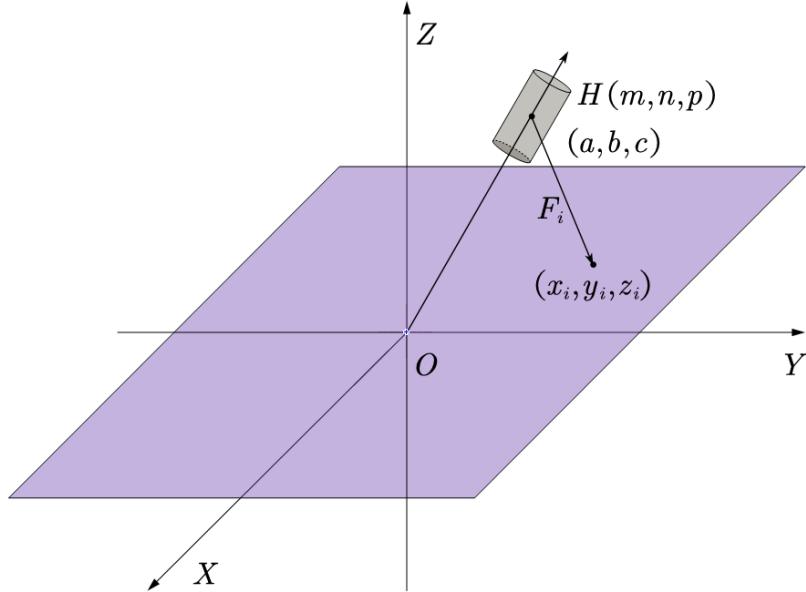


Figure 1.1: Magnetic dipole model for localization system. The cylindrical magnet is regarded as a magnetic dipole with the position parameters (a, b, c) and orientation parameters (x_i, y_i, z_i) . The B_i represents the magnetic field intensity at the i -th sensor and the sensor position is (x_i, y_i, z_i) , $1 < i < N$.

1.2.1.2 B. Localization Algorithms

In order to track the position and orientation of a permanent magnet, the Optimization algorithms are needed to use to minimize the error. As the sensors measuring the magnetic field intensity around the cylindrical magnet, we can calculate the position and orientation of the magnet based on the mathematical model above. Extending (1.1), we have three equations which are listed below.

$$\begin{cases} B_{i_x} = B_T \left\{ \frac{3[m(x_i - a) + n(y_i - b) + p(z_i - c)] \cdot (x_i - a)}{R_i^5} - \frac{m}{R_i^3} \right\}; \\ B_{i_y} = B_T \left\{ \frac{3[m(x_i - a) + n(y_i - b) + p(z_i - c)] \cdot (x_i - b)}{R_i^5} - \frac{n}{R_i^3} \right\}; \\ B_{i_z} = B_T \left\{ \frac{3[m(x_i - a) + n(y_i - b) + p(z_i - c)] \cdot (x_i - c)}{R_i^5} - \frac{p}{R_i^3} \right\}. \end{cases} \quad (1.3)$$

Where B_{i_x} , B_{i_y} and B_{i_z} are the magnetic field intensity data of the three-axis respectively.

As we have got the measured values $(B'_{i_x}, B'_{i_y}, B'_{i_z})$ of the magnetic field intensity around our target magnet, we can subtract our measured values and theoretical values to define the error function, so our error function is listed below.

$$E = \|(B'_{i_x} - B_{i_x}, B'_{i_y} - B_{i_y}, B'_{i_z} - B_{i_z})\|_2^2 = \sum_{i=1}^N [(B'_{i_x} - B_{i_x})^2 + (B'_{i_y} - B_{i_y})^2 + (B'_{i_z} - B_{i_z})^2] \quad (1.4)$$

Where E is the total error; $B'_{i_x}, B'_{i_y}, B'_{i_z}$ are the measured magnetic field intensity values of three-axis respectively; $B_{i_x}, B_{i_y}, B_{i_z}$ are the theoretical values.

There are several algorithms to solve the error function above, such as Powell algorithm, Levenberg-Marquard(LM) algorithm and DIRECT algorithm[8, 9, 7, 6, 10, 11, 12, 13, 14]. According to the paper, we need to consider the real-time performance and accuracy, and in practical application the initial point is very difficult to guess and determine. Considering the issues mentioned before, LM algorithm is a great method to solve this question because even if the initial guess point is far away from the true point, the LM algorithm can be easy to converge to the local optimal solution.

CHAPTER 1. INTRODUCTION

In order to minimize the value of error mentioned above, we need to optimize the solution. In this paper[5], the particle swarm optimization (PSO) algorithm has been used. The advantage of PSO algorithm is that the initial guess point is not needed. However, in our investigate, we define the initial point, so the PSO algorithm has been abandoned. In our algorithms, there exists a constraint which is listed beneath.

$$m^2 + n^2 + p^2 = 1 \quad (1.5)$$

Therefore, we need to solve the constrained nonlinear optimization. We have considered many optimization algorithms, such as trust region algorithm, interior point method, sequential quadratic programming (SQP) and active set method. In this methods, we consequently choose the interior point method, because other three algorithms have some drawbacks. For instance, the trust region algorithm cannot solve the constraint problem, the active set method needs to calculate the verbose Jacobian matrix and the SQP method has the poorest performance both on speed and accuracy. Consider that this algorithm only performs on small scale, the active set method has the best performance. Therefore, the active set method is the best method we choose.

1.2.2 Model Design

Focusing on the model design, I analyze the paper *Design and Optimization Strategy of Sensor Array Layout for Magnetic Localization System (IEEE SENSORS JOURNAL, VOL. 17, NO. 6, MARCH 15, 2017)* [15]. In this paper, the authors use a mathematical way to calculate the optimal layout strategy, they suppose \mathbf{A} is an $N \times N$ sensor array. For example, as shown in Figure 1.2 , $N = 5$, $N_s = 4$, $A_{4,2} = 1$ and $A_{4,3} = 0$.

In the proposed layout patterns, the optimized sensor array layout is symmetrical, not only for the left and right parts, but also for the upper and lower parts.

The Figure 1.3 is the layout design in the paper which shows the magnetic sensor array layout, whose drawbacks are that this layout is only one dimension which will

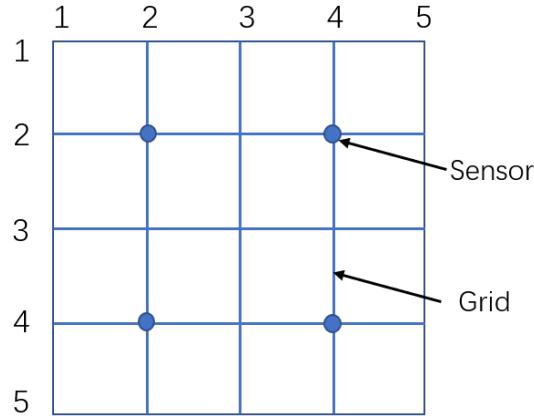


Figure 1.2: Grid based sensor layout strategy.

reduce the accuracy, and the height is determined which means that the value on Z-axis is fixed and the degree of freedom is decreased.

In Figure 1.4 (from [15]), we can find the pathway is in a horizontal plane, which means that the magnet translate on the plane without rotating. Therefore, there are two degree of freedom constrained, which will let the system be lack of robustness.

As considering the points mentioned above, we have taken the rotation of the magnet into account and add more degree of freedom into our system.

In the reference papers [5, 15], we find a common drawback in these two papers, that is the whole design of the experiment platform is very small. We can see the errors of localization in these two papers are both small which can be accepted, but the fact is that the small size of the platform makes contribution to the result. As we know, the specifications of the platform are close to $15cm \times 15cm \times 10cm$ in paper [5] and $20cm \times 20cm \times 10cm$ in paper [15]. Therefore, these two experiment platforms can not be used in application owing to the small size.

Taking the application and making breakthrough in the size into account, we have designed a $38cm \times 25cm \times 28cm$ platform to conduct the experiment.

CHAPTER 1. INTRODUCTION

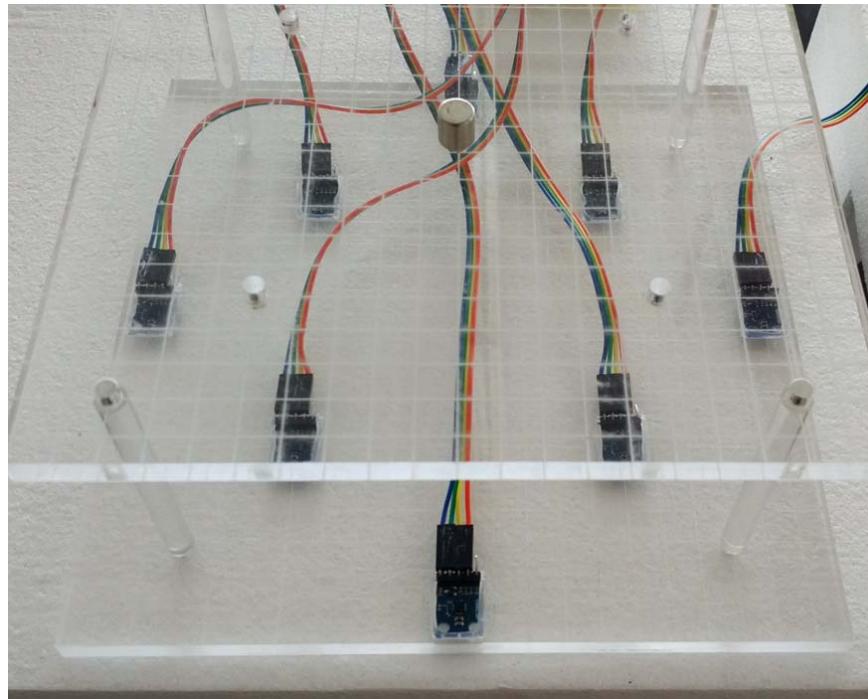


Figure 1.3: Magnetic sensor array layout of case 5 shown in Figure ??(f).

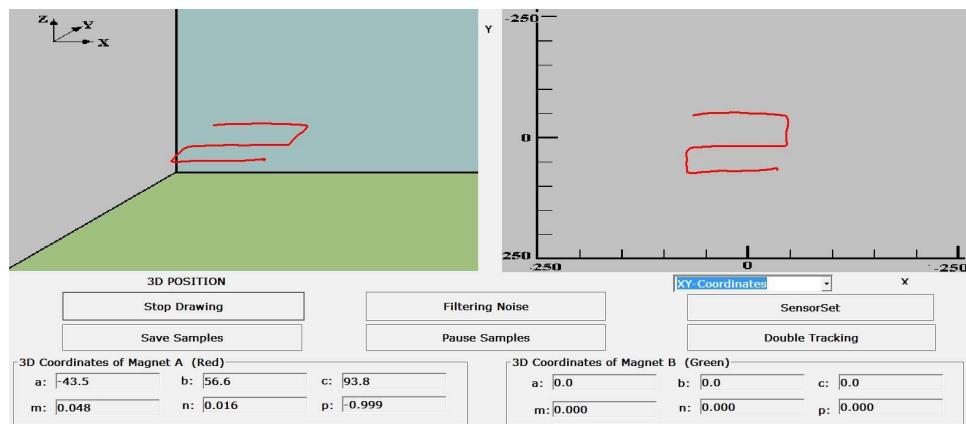


Figure 1.4: Software interface for visualize the localization results in 3-D and 2-D environments.

Chapter 2

Building the Hardware Platform

In this chapter, I will introduce the hardware platform building process in detail, especially the sensor we use, the data transform method, the PCB we design and the experiment platform building. The Figure 2.1 is the working principle diagram of our magnetic localization system. The magnetic field intensity which magnet produces is collected by the LSM303D magnetic sensor on magnetic sensor array and after being processed by the LSM303D sensor, the magnetic field intensity data (raw data) is transmitted to the ZYBO-7010 board using I2C serial bus protocol. When the ZYBO-7010 board gets the raw data, it will process the data. As we set the magnetic sensitivity of the LSM303D is $\pm 4\text{g}$, so in ZYBO-7010, the raw data should be multiplied by 0.16 to get the actual magnetic field intensity data, then the data is transmitted to the PC through UART. On PC, we use the data that ZYBO-7010 board sends to execute our localization algorithm and eventually the locating result and the error are shown.

2.1 The Selection of Localization Target

For implementing the magnetic localization, the located target need to be chose. The locating target should be magnetic, as we know there exist two kinds of magnetic substance, one is permanent magnet and the other one is electromagnet. In our investigation, the magnetic target (capsule) need to be put in the human body. From the point of view of the energy supply, it is unrealistic to use electromagnet

CHAPTER 2. BUILDING THE HARDWARE PLATFORM

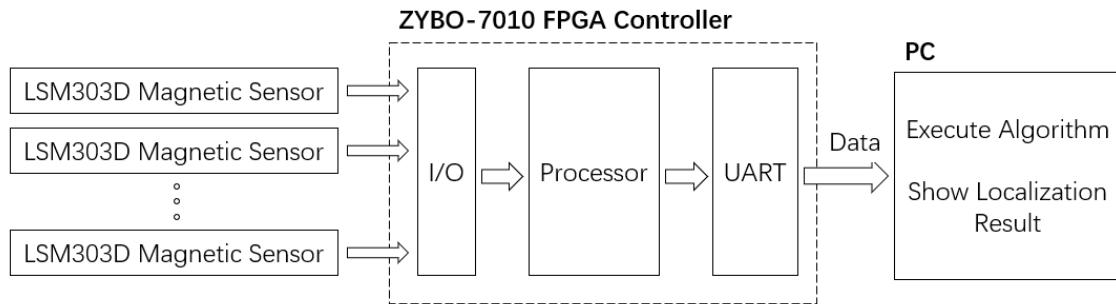


Figure 2.1: The block diagram of magnetic localization system

as localization target, because the electromagnet requires large electrical energy to achieve the same magnetic field intensity as permanent magnet. Therefore, the permanent magnet is the localization target that we choose.

Given that the permanent magnet is placed in the capsule, therefore the shape of the magnet is ellipsoidal. Figure 2.2 shows the N52 rare-earth magnet that we choose whose material quality is Nd-Fe-B rare earth, and the element of electroplated layer is nickel. The size of the magnet is $\phi 10mm * 15mm$. This magnetism of the N52 rare-earth magnet is very strong, the magnetic field intensity of the surface is more than 5000 gauss.

2.2 LSM303D Sensor

For locating the magnet mentioned above, we need to choose a kind of sensor to collect the magnetic field intensity of the magnet. There are many kinds of magnetic sensor, such as ferromagnetic induction coil sensors[16], HMC1043 three-axis magnetic sensor[17] and LSM303D 3D accelerometer and 3D magnetometer sensor[18]. In these sensors, we eventually chose the LSM303D sensor to collect the magnetic field intensity data. Because, for the ferromagnetic induction coil sensors, the size of the whole sensor is too huge to be embedded into the equipment that human wear. Second, for

CHAPTER 2. BUILDING THE HARDWARE PLATFORM

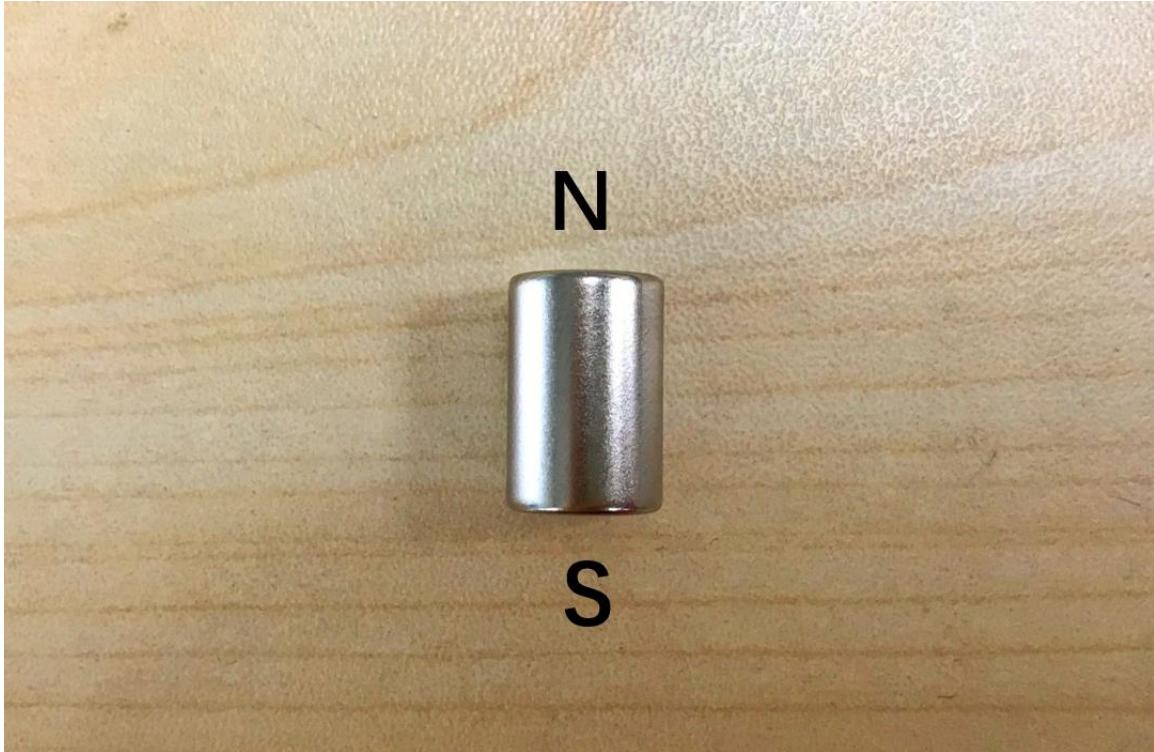


Figure 2.2: N52 Rare-earth Magnet

HMC1043 three-axis magnetic sensor, the output of this magnetic sensor is a kind of analog signal, which means that the output of HMC1043 sensor cannot be used directly, because the data that PC processes should be digital. If we want to use HMC1043 sensor, the output analog signal of HMC1043 sensor need to be amplified by the AD623 differential amplification circuit, and then the amplified voltage signal will be sent to the embedded module K60 sequentially by analog strobe chip DG406. After that, the A/D conversion module on the K60 system will sample and filter the amplified analog voltage signal and output the analog voltage signal and the output analog voltage signal will be sent to the upper PC through the TTL module at last[11]. However, if we use the LSM303D sensor, the output of the sensor is the magnetic field intensity data that we can use directly because the output is digital data, so the output data of the LSM303D sensor is sent directly to our ZYBO-7010

CHAPTER 2. BUILDING THE HARDWARE PLATFORM

FPGA board to be processed and then the data after processed is transmitted to the PC through the UART module, finally the data is shown on the PC. Taking into account the system simplification and facilitation, we eventually decided to use the LSM303D sensor.

The LSM303D is a system-in-package featuring a 3D digital linear acceleration sensor and a 3D digital magnetic sensor.

The LSM303D has linear acceleration full scales of $\pm 2g$ / $\pm 4g$ / $\pm 6g$ / $\pm 8g$ / $\pm 16g$ and a magnetic field full scale of ± 2 / ± 4 / ± 8 / ± 12 gauss. It includes an I²C serial bus interface that supports standard and fast mode (100 kHz and 400 kHz) and SPI serial standard interface[18].

The system can be configured to generate an interrupt signal for free-fall, motion detection and magnetic field detection. Thresholds and timing of interrupt generators are programmable by the end user.

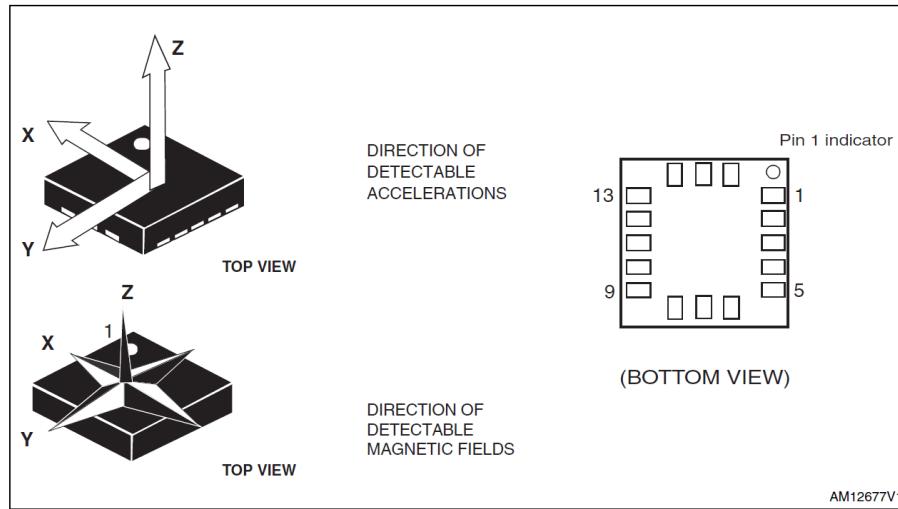


Figure 2.3: LSM303D Sensor

And magnetic and accelerometer blocks can be enabled or put into power-down mode separately. The LSM303D is available in a plastic land grid array package (LGA) and is guaranteed to operate over an extended temperature range from -40 °C

CHAPTER 2. BUILDING THE HARDWARE PLATFORM

to +85 °C . Figure 2.3 is the pin description diagram of the LSM303D.

The table 2.1 is the description of each pin of LSM303D, which describes the detail function of each pin. In this table, we can see that the LSM303D sensor supports I2C serial bus interface and SPI serial standard interface. In our research, we choose the I2C serial bus to transmit the data, because I2C serial bus is a simple way, which is only composed of one serial clock (SCL) and one serial data (SDA).

Table 2.1: Pin Description

Pin	Name	Function
1	Vdd_IO	Power supply for I/O pins
2	SETC	S/R capacitor connection (C2)
3	SETP	S/R capacitor connection (C2)
4	SCL SPC	I2C serial clock (SCL) SPI serial port clock (SPC)
5	GND	0 V supply
6	SDA SDI SDO	I2C serial data (SDA) SPI serial data input (SDI) 3-wire interface serial data output (SDO)
7	SDO SA0	SPI serial data output (SDO) I2C less significant bit of the device address (SA0))
8	CS	SPI enable I2C/SPI mode selection (1: SPI idle mode / I2C communication enabled; 0: SPI communication mode / I2C disabled)
9	INT 2	Interrupt 2
10	Reserved	Connect to GND
11	INT 1	Interrupt 1
12	GND	0 V supply
13	GND	0 V supply
14	Vdd	Power supply
15	C1	Capacitor connection (C1)
16	GND	0 V supply

2.3 I2C Serial Bus

2.3.1 Overview

The I2C serial bus was designed by Philips in the early '80s to allow easy communication between components which reside on the same circuit board. Philips Semiconductors migrated to NXP in 2006.

The name I2C translates into "Inter IC". Sometimes the bus is called IIC or I²C bus.

The original communication speed was defined as a maximum of 100 kbit per second and many applications don't require faster transmissions. For those that do there is a 400 kbit fast mode and - since 1998 - a high speed 3.4 Mbit option available. Recently, fast mode plus a transfer rate between this has been specified. Beyond this, there is the ultra-fast mode UFM, but it frankly is no real I2C bus.

I2C is not only used on single boards but also to connect components which are linked via cable. Simplicity and flexibility are key characteristics that make this bus attractive to many applications.

And the most significant features include:

- Only two bus lines are required
- No strict baud rate requirements like for instance with RS232, the master generates a bus clock
- Simple master/slave relationships exist between all components
- Each device connected to the bus is software-addressable by a unique address
- I2C is a true multi-master bus providing arbitration and collision detection

CHAPTER 2. BUILDING THE HARDWARE PLATFORM

2.3.2 Protocol

Communication via I2C is more complex than with a UART or SPI solution. The signaling must adhere to a certain protocol for the devices on the bus to recognize it as valid I2C communications. Fortunately, most devices take care of all the fiddly details for you, allowing you to concentrate on the data you wish to exchange.

There exist two ways to implement I2C communication: 7-bit addressing system and 10-bit addressing system. Because we only use the 7-bit addressing mode in our project, so I have learned the 7-bit addressing system.

Messages are broken up into two types of frame: an address frame, where the master indicates the slave to which the message is being sent, and one or more data frames, which are 8-bit data messages passed from master to slave or vice versa. Data is placed on the SDA line after SCL goes low, and is sampled after the SCL line goes high. The time between clock edge and data read/write is defined by the devices on the bus and will vary from chip to chip. And the Figure 2.4 is the basic transmitting sequence chart based on 7-bit addressing system.

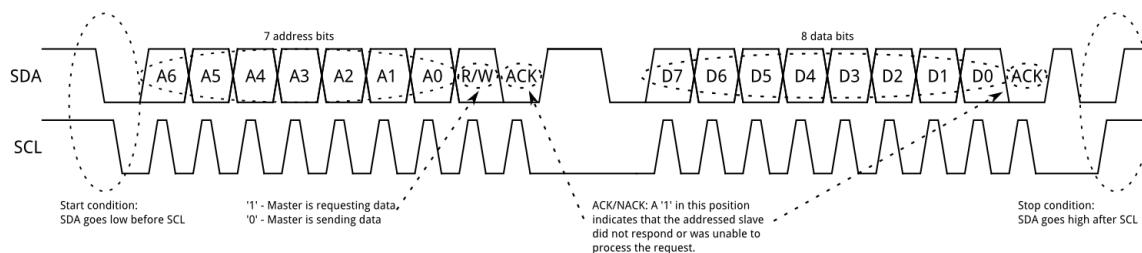


Figure 2.4: Basic I2C Sequence Chart

2.4 ZYBO-7010 SoC Board

This ZYBO-7010 SoC board is the hardware platform that I use to implement our project.

CHAPTER 2. BUILDING THE HARDWARE PLATFORM

The Z-7010 is a member of Xilinx Zynq-7000 family, which integrates a dual-core ARM Cortex-A9 processor with Xilinx 7-series Field Programmable Gate Array (FPGA) logic. The ZYBO-7010 is shown as Figure 2.5.

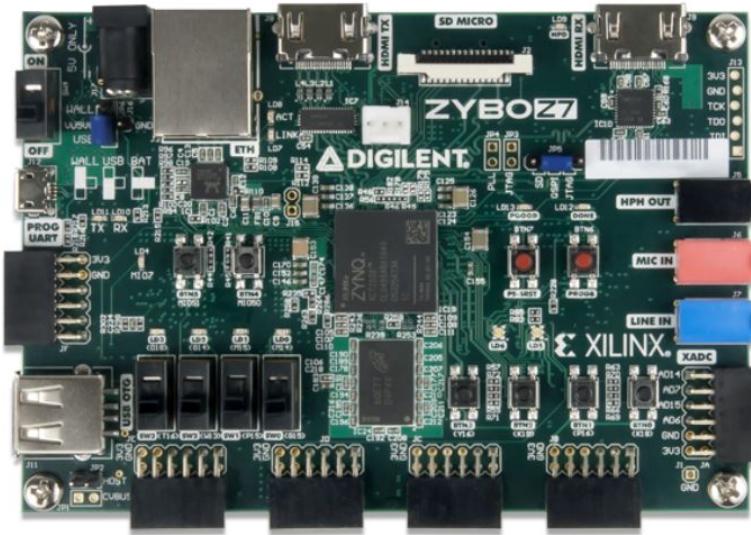


Figure 2.5: ZYBO-7010 SoC Board

Here are some features[19] about this SOC board:

- 650MHz dual-core Cortex-A9 processor
- DDR3 memory controller with 8 DMA channels High-bandwidth peripheral controllers: 1G Ethernet, USB 2.0, SDIO
- Low-bandwidth peripheral controller: SPI, UART, CAN, I2C
- Reprogrammable logic equivalent to Artix-7 FPGA
- 512MB x32 DDR3 w/ 1050Mbps bandwidth
- 16-bits per pixel VGA source port
- Trimode (1Gbit/100Mbit/10Mbit) Ethernet PHY

CHAPTER 2. BUILDING THE HARDWARE PLATFORM

- OTG USB 2.0 PHY (supports host and device)
- 128Mb Serial Flash w/ QSPI interface
- External EEPROM (programmed with 48-bit globally unique EUI-48/64-compatible identifier)
- On-board JTAG programming and UART to USB converter

The reason I choose the ZYBO-7010 FPGA board but not the MCU board such as STM32[20], STC89C51[21] and Arduino board[22] is that compared with MCUs, the FPGA board has more powerful computing capability, it can process large amounts of data when we increase the number of sensors we use, the FPGA board can also have a good performance in real time and for programming, the IDE of Xilinx® is excellent and convenient because most of code have been integrated in package, therefore the code we program will be robust.

2.5 The IDEs of Xilinx®

2.5.1 Xilinx Vivado

Xilinx Vivado is the software that I use to design the programmable logic system for reading the data of sensor LSM303D.

Vivado Design Suite, as the Figure 2.6 showing, is a software suite produced by Xilinx for synthesis and analysis of HDL designs, superseding Xilinx ISE with additional features for system on a chip development and high-level synthesis.

Unlike ISE which relied on ModelSim for simulation, the Vivado System Edition includes an in-built logic simulator. Vivado also introduces high-level synthesis, with a toolchain that converts C code into programmable logic.

Vivado enables developers to synthesize (compile) their designs, perform timing analysis, examine RTL diagrams, simulate a design's reaction to different stimuli, and

CHAPTER 2. BUILDING THE HARDWARE PLATFORM

configure the target device with the programmer. Vivado is a design environment for FPGA products from Xilinx, and is tightly-coupled to the architecture of such chips, and cannot be used with FPGA products from other vendors.

Vivado was introduced in April 2012, and is an integrated design environment (IDE) with system-to-IC level tools built on a shared scalable data model and a common debug environment. Vivado includes electronic system level (ESL) design tools for synthesizing and verifying C-based algorithmic IP; standards based packaging of both algorithmic and RTL IP for reuse; standards based IP stitching and systems integration of all types of system building blocks; and the verification of blocks and systems[23]. A free version WebPACK Edition of Vivado provides designers with a limited version of the design environment[24].

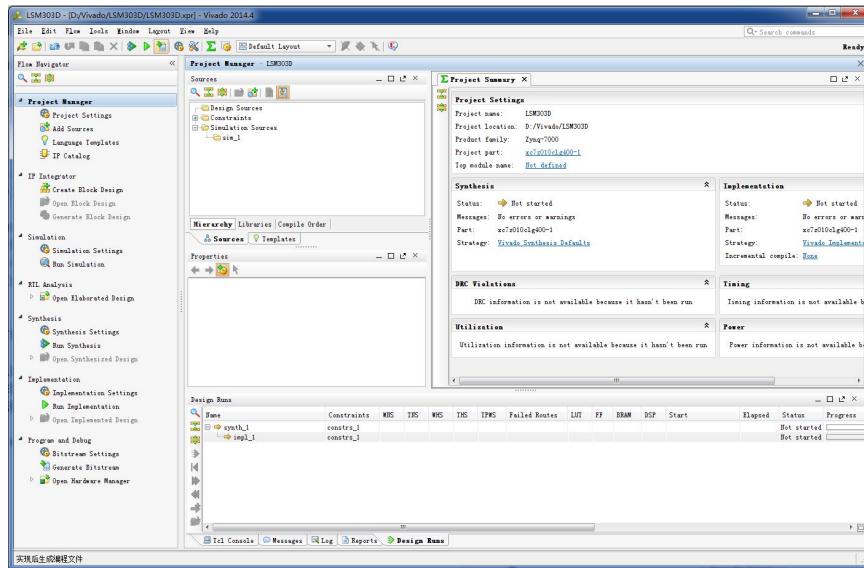


Figure 2.6: Xilinx Vivado Design Suite 2014.4 with project summary (center) and project navigation tree (left)

2.5.2 Xilinx SDK

Xilinx Software Development Kit (XSDK) is the coding environment that I have learned to design the processing system for driving the sensor LSM303D to collect

CHAPTER 2. BUILDING THE HARDWARE PLATFORM

the magnetic field data.

The Xilinx Software Development Kit (XSDK) is the integrated design environment for creating embedded applications on any of Xilinx's award winning microprocessors: Zynq[®] UltraScale+ MPSoC, Zynq-7000 All Programmable SoCs, and the industry-leading MicroBlazeTM soft-core microprocessor. The SDK is the first application IDE to deliver true homogenous and heterogeneous multi-processor design, debug, and performance analysis. Benefits include:

- Zynq UltraScale+ MPSoC, Zynq-7000 AP SoCs, and MicroBlaze support
- Included with the Vivado Design Suite or available as a separate free download for embedded software developers
- Based on Eclipse 4.5.0 and CDT 8.8.0 (as of the 2016.3 release)
- Complete Integrated Design Environment (IDE) that directly interfaces to the Vivado embedded hardware design environment
- Complete software design and debug flows supported, including multi-processor and hardware/software co-debug capabilities
- Editor, compilers, build tools, flash memory management, and JTAG debug integration
- Full suite of libraries and device drivers
- FreeRTOS integrated as RTOS available for all platforms
- Xilinx Software Command Line Tool (XSCT) available for scripting

In addition to the above native Eclipse provided features, SDK also provides the following tools for use in Xilinx embedded software development. And Figure 2.7 shows the main interface.

CHAPTER 2. BUILDING THE HARDWARE PLATFORM

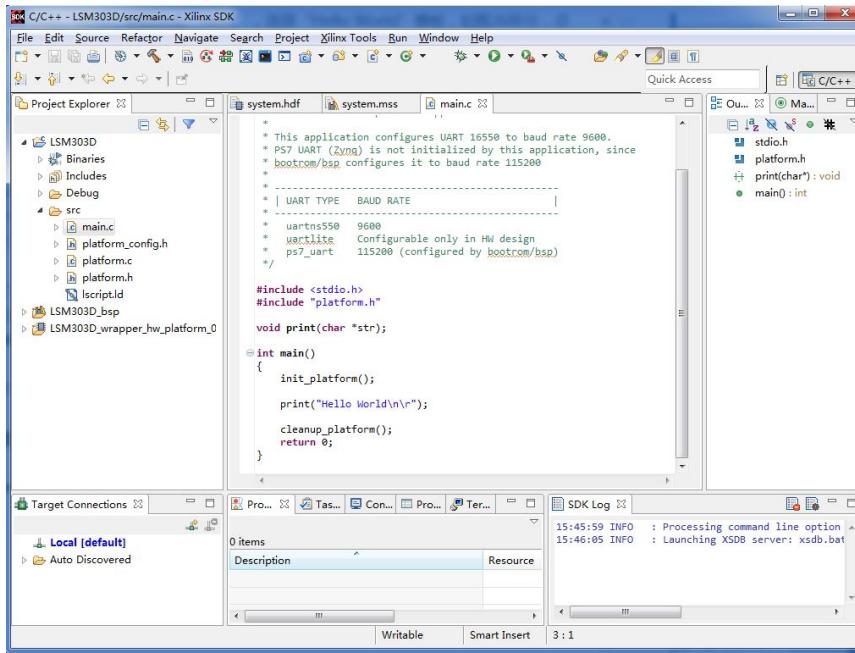


Figure 2.7: Xilinx Software Development Kit 2014.4 with main.c (center) and project explorer tree (left)

2.6 Implementation of Hardware Platform Building

2.6.1 Reading Sensors Data

When I started to implement reading data from sensor, I tried almost two months to find a correct way to design the IP system and code exactly to match the I2C time sequence. Finally, I figured it out and the whole system worked correctly. Here, I will introduce the three steps that I have done to collect data from more and more sensors.

2.6.1.1 A. Reading One Sensor Data

In order to read more sensors data, the basic thing is to read the data from one sensor successfully.

CHAPTER 2. BUILDING THE HARDWARE PLATFORM

Firstly, I finished the block design for reading one sensor on the Vivado. Figure 2.8 shows the block design with a ZYNQ7 processing system, a processor system reset, an AXI interconnect and an AXI I2C IP core.

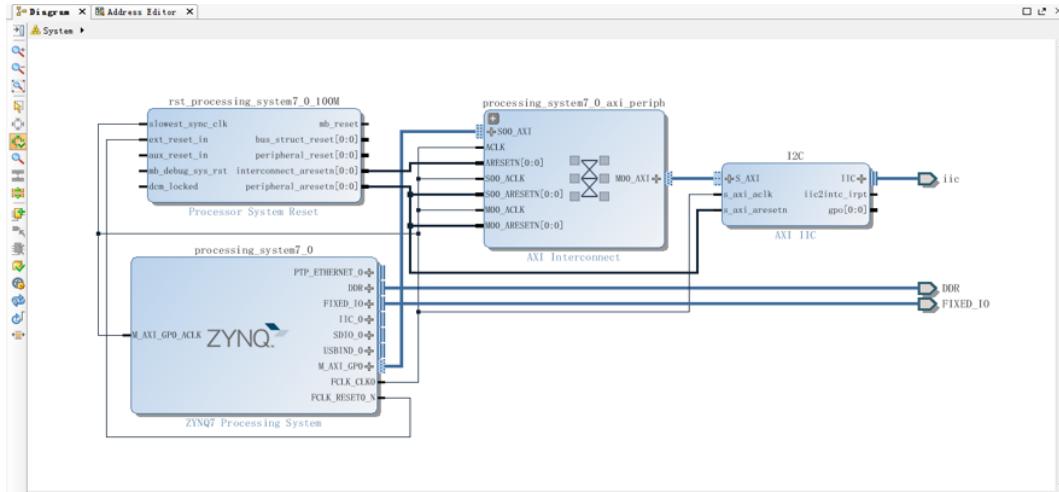


Figure 2.8: Block design for reading one sensor

Secondly, I coded the C language on the Xilinx SDK for enabling the ARM of ZYBO-7010 to read the data from sensor and transmit data to UART in order to realize communication between the board and PC. Figure 2.6.1.1 (a) shows the connection between the LSM303D and ZYBO-7010 board, and Figure 2.6.1.1 (b) shows the magnetic field intensity data of x, y, and z axes, and the unit of the data is mg(milligauss).

2.6.1.2 B. Reading Four Sensors Data

As finishing reading data from one sensor, I started to read data from four LSM303D sensors. because the LSM303D has two different address by polling up or done the pin SDO/SA0 of the sensor. Therefore, I can use only two I2C bus to control four LSM303D sensors. I finished the block design for reading four sensors on the Vivado. Figure 2.11 shows the block design with a ZYNQ7 processing system, a processor system reset, an AXI interconnect and two AXI I2C IP cores.

CHAPTER 2. BUILDING THE HARDWARE PLATFORM

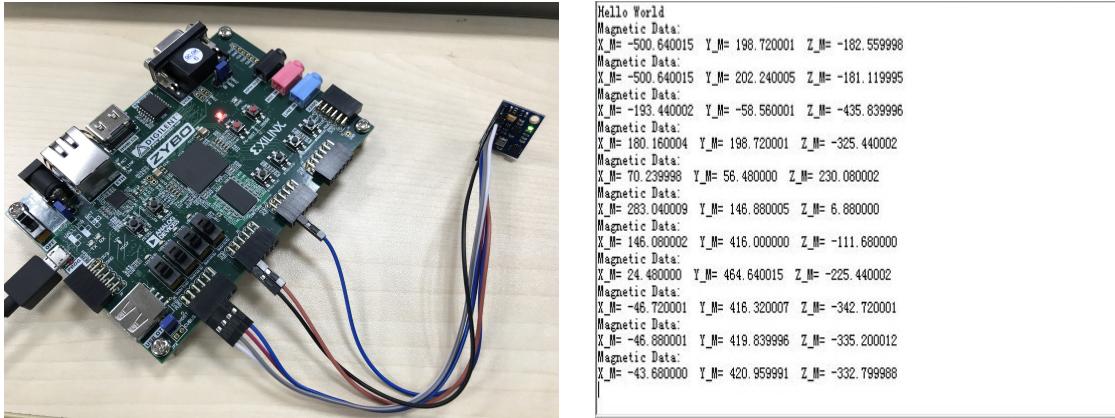


Figure 2.9: (a) Connect the sensor to Figure 2.10: (b) Send the x, y, and z axes magnetic field intensity data to PC.

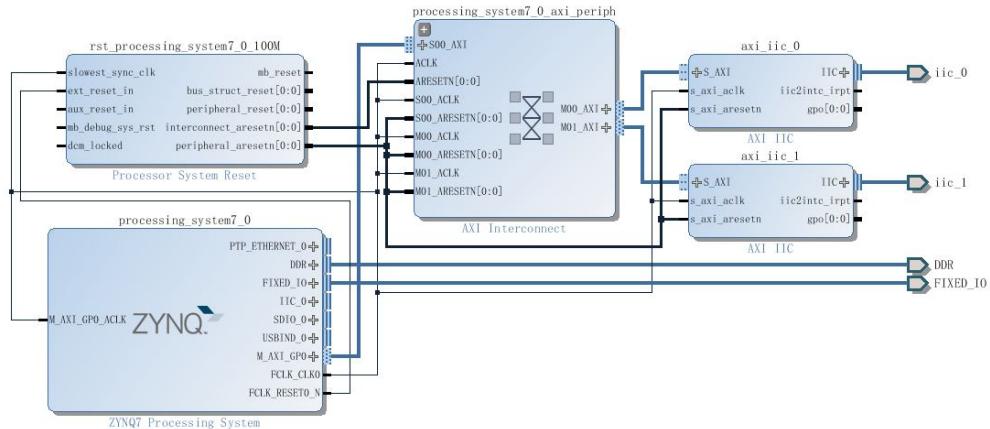


Figure 2.11: Block design for reading one sensor

Figure 2.6.1.2 (a) shows the connection between the four sensors and ZYBO-7010 board, and Figure 2.6.1.2 (b) shows the magnetic field intensity data of x, y, and z axes while reading four sensors' data, and the unit of the data is *mg*(milligauss).

CHAPTER 2. BUILDING THE HARDWARE PLATFORM

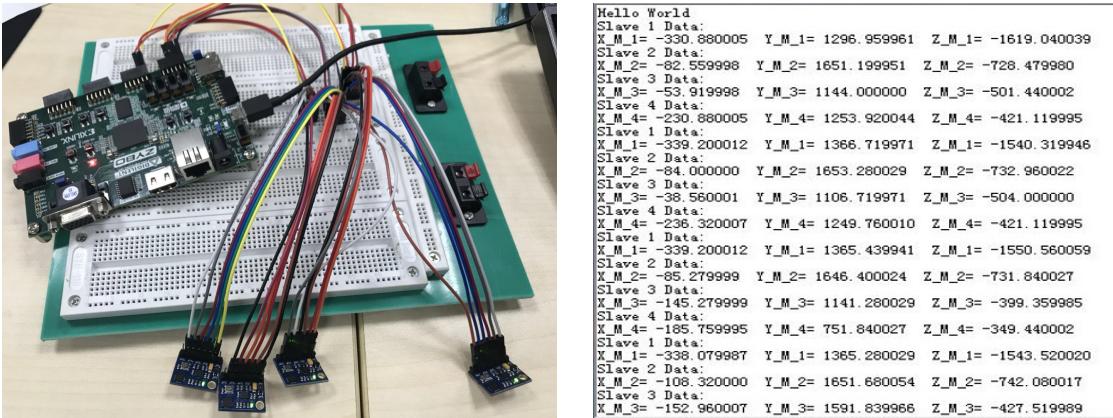


Figure 2.12: (a) Connect four sensors to Figure 2.13: (b)The magnetic field intensity data of four sensors.

2.6.1.3 C. Reading Sixteen Sensors Data

As we can read data from four sensors, it is easy for us to read the magnetic data from sixteen sensors. In this process, I use eight I2C buses, each I2C bus controls two LSM303D sensors. In order to control reading data manually, the reading data program is designed to start when the button is triggered. Figure 2.14 describes the block design with a ZYNQ7 processing system, a processor system reset, an AXI interconnect and eight AXI I2C IP cores.

2.6.2 Building The Experiment Platform

In order to realize our system and improve the robustness of our system, we design an experiment platform made by acrylic board to process our experiments. Figure 2.6.2 shows the experiment platform and components of the platform.

As we can see, in Figure 2.6.2 (a), the layers represents the scale of Z axis, the plate with plenty of holes is detachable, which we can plug it in any layer we want, and the components in Figure 2.6.2 (b) are used to plug into the hole of the plate mentioned above to determined the location of the magnet when we insert a magnet into the component, also the different hole of a component represents a different angle

CHAPTER 2. BUILDING THE HARDWARE PLATFORM

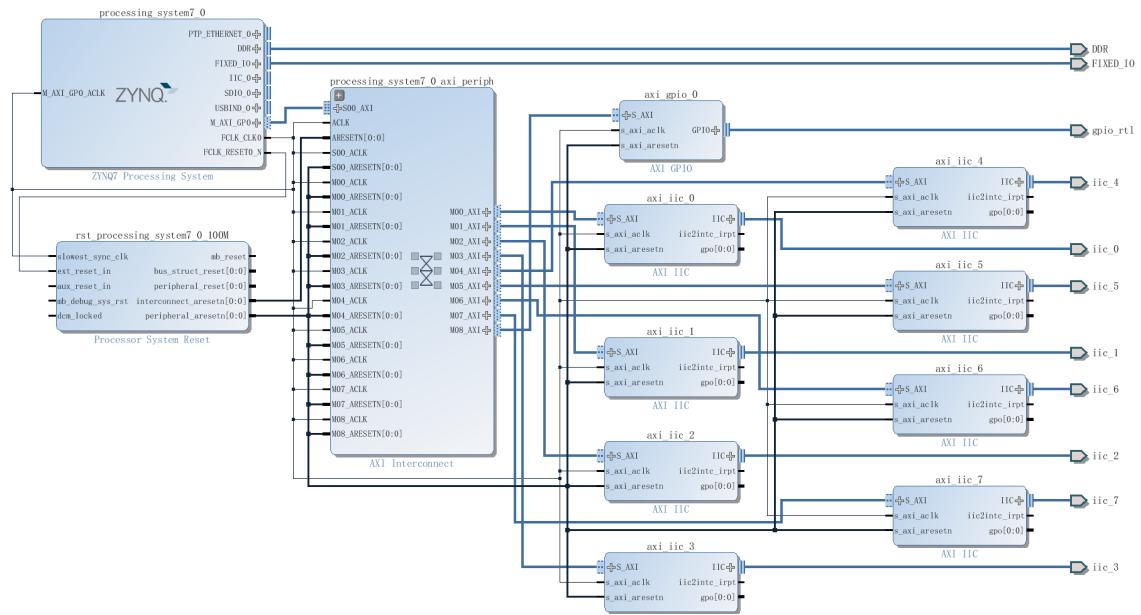


Figure 2.14: Block design for reading one sensor

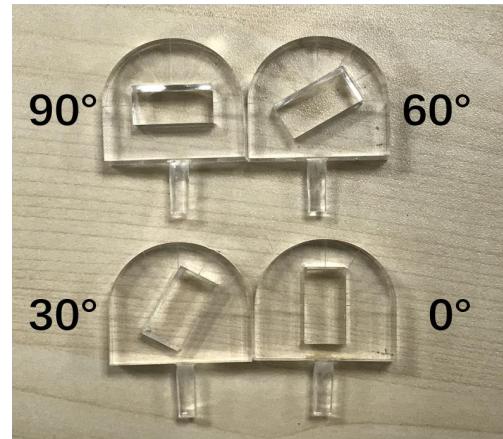


Figure 2.15: (a) The experiment platform made by acrylic board.

CHAPTER 2. BUILDING THE HARDWARE PLATFORM

(there exist 0° , 30° , 60° , 90°).

2.6.3 Construction of Sixteen-Sensor Localization System

When we designed the experiment platform, we had already set the position of each sensor and the distance between two sensors. On the outside surface around the experiment platform, we have made marks for the sensors' position. Then we start to construct the sixteen-sensor localization system, we stick four sensors on each side around the platform and the diagram of experiment platform is shown below as Figure 2.17.

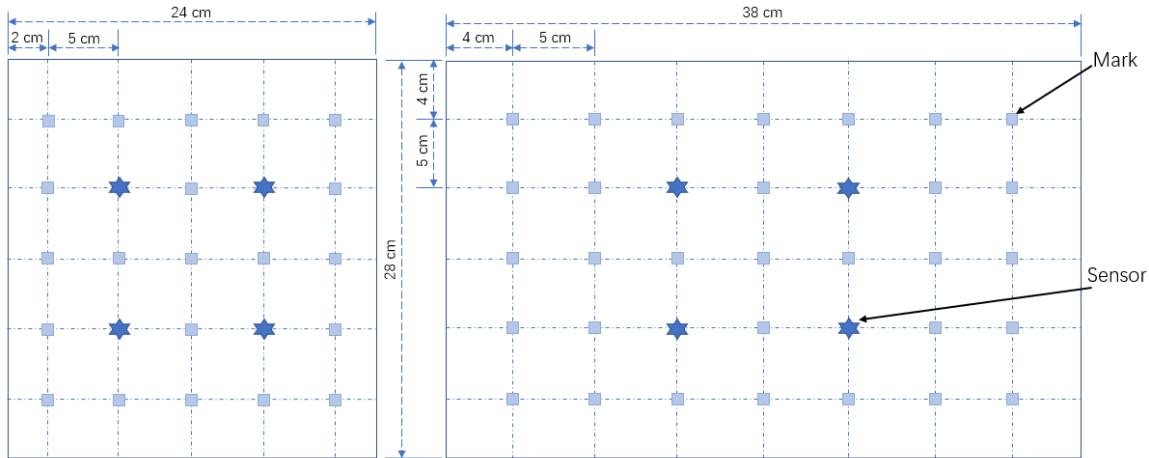


Figure 2.17: The design diagram of the experiment platform

After sticking sixteen sensors to the surface around experiment platform according to the design diagram of the experiment platform, we finished the construction of sixteen-sensor localization system, and the illustration of the system is shown below as Figure 2.18. In this figure, the coordinate system is the absolute coordinate system that we define. As we can see, the coordinate system of every four sensors on each side of the platform is different, so the coordinate system of sensors need to be converted to the absolute coordinate system mentioned above. Here, we number each side of

CHAPTER 2. BUILDING THE HARDWARE PLATFORM

the platform in this figure, and the coordinate transformation equations are listed below as 2.1 and 2.2.

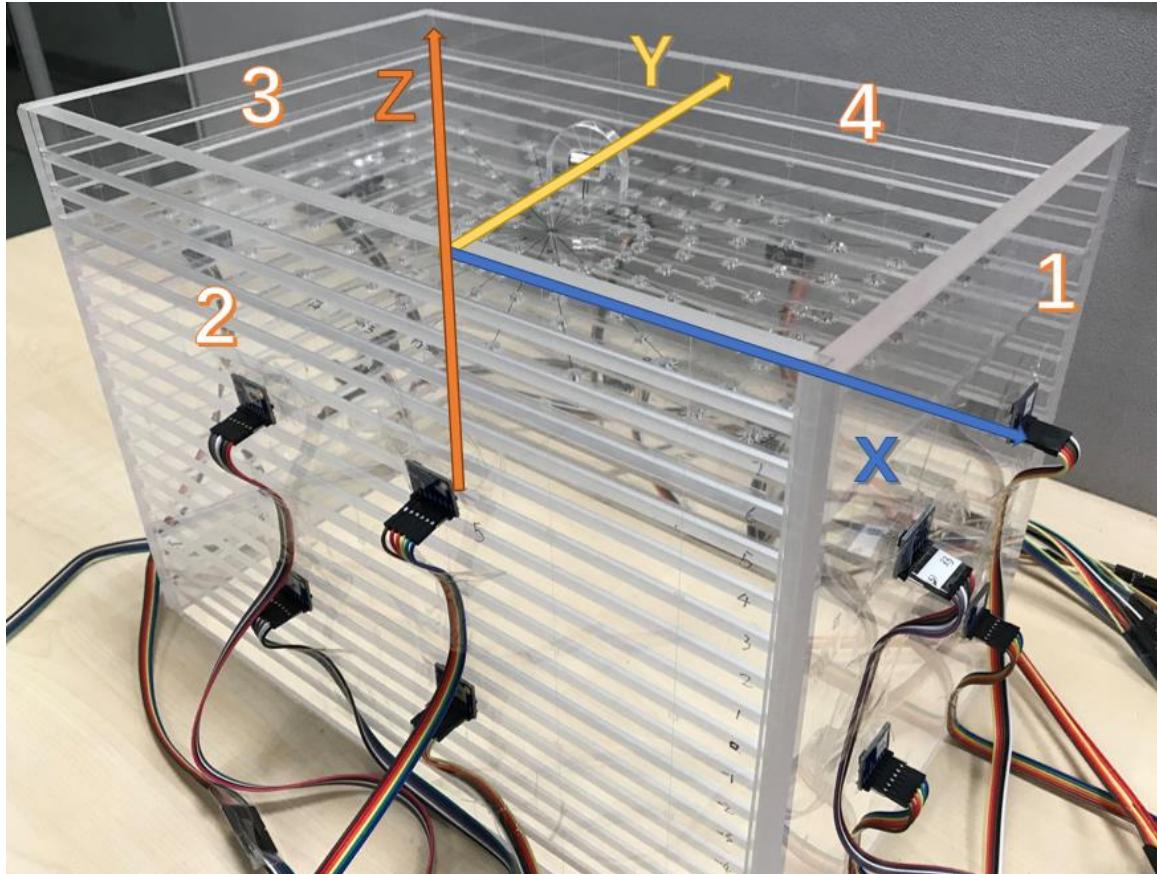


Figure 2.18: The localization system with sixteen sensors

For the four sides, the coordinate transformation equations are:

$$(X_1, Y_1, Z_1) = (x_1, y_1, z_1) \begin{pmatrix} 0 & -1 & 0 \\ 0 & 0 & 1 \\ -1 & 0 & 0 \end{pmatrix}, (X_2, Y_2, Z_2) = (x_2, y_2, z_2) \begin{pmatrix} -1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \quad (2.1)$$

CHAPTER 2. BUILDING THE HARDWARE PLATFORM

$$(X_3, Y_3, Z_3) = (x_3, y_3, z_3) \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}, (X_4, Y_4, Z_4) = (x_4, y_4, z_4) \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{pmatrix} \quad (2.2)$$

where the (X_1, Y_1, Z_1) , (X_2, Y_2, Z_2) , (X_3, Y_3, Z_3) and (X_4, Y_4, Z_4) are the coordinates of the first, second, third and fourth surface individually after coordinate transformation, the (x_1, y_1, z_1) , (x_2, y_2, z_2) , (x_3, y_3, z_3) and (x_4, y_4, z_4) are the coordinates of the sensor who is on the first, second, third and fourth surface individually.

2.6.4 PCB Design

As mentioned above, the magnetic sensor we use is LSM303D, but the sensor we bought is a module whose model is GY-89, with 3 sensor chips (LSM303D, L3GD20 and BMP180). Besides, it will cost a lot when the need for magnetic sensor is huge. However, the unit price of the LSM303D chip is cheap, therefore we decide to design the PCB with the LSM303D chip for taking into account lowering the cost of investment and application.

We used the Altium Designer software to finish the whole process of designing the LSM303D PCB, included the circuit schematic designing, component packaging and PCB designing. Based on the diagram of the LSM303D electrical connections in the LSM303D Datasheet[18], we first designed the LSM303D circuit principle diagram which is shown below as Figure 2.19. Secondly, the connection of components was completed in the PCB design file, which is shown as Figure 2.6.4 (a). The EIA size of the surface mount components in figure 2.6.4 (a) is 0603 and this PCB board is a two-layer board, whose ink thickness is 10-15 μm , the thickness of tin spray is 15 μm on average and the thickness on immersion gold is 0.8-1.2 μin . Figure 2.6.4 (b) is the PCB circuit board of LSM303D that we solder. The pin 3V3 is the high potential which provide the power for the components. The pin GND connects to the ground and the top and bottom layers of the PCB are clad copper which connects to the

CHAPTER 2. BUILDING THE HARDWARE PLATFORM

ground. The pin SCL is the I²C serial clock which control the clock frequency of the LSM303D sensor. The pin SDA is used to transmit the I²C serial data. And the pin CS is the enable port which will let the LSM303D sensor work when the electrical level of CS is high and it will disable the PCB circuit when the electrical level is low.

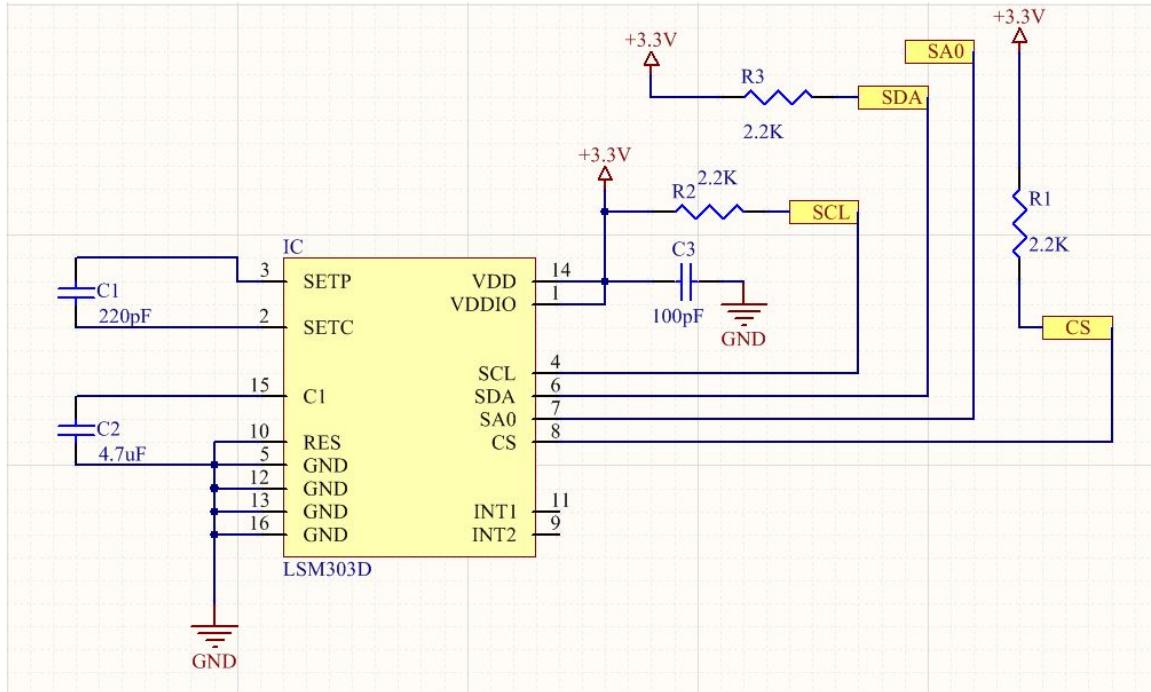


Figure 2.19: The LSM303D circuit schematic

CHAPTER 2. BUILDING THE HARDWARE PLATFORM

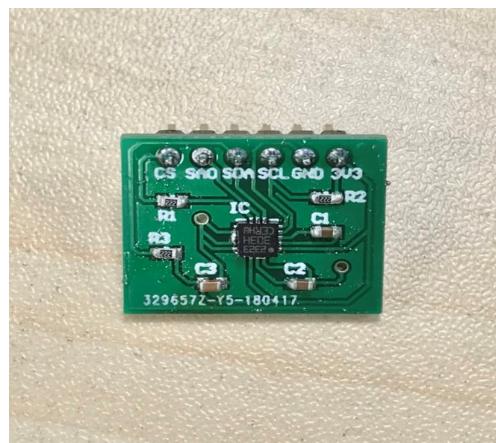
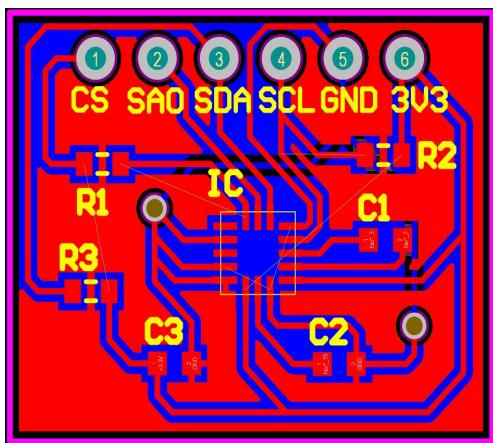


Figure 2.20: (a) The PCB design diagram of LSM303D.
Figure 2.21: (b)The PCB circuit board of LSM303D.

Chapter 3

Implementation of Localization Algorithms

Given a permanent magnet, the permanent magnet will produce a magnetic field in space. The position and orientation of the magnetic source is determined by this given permanent magnet. According to the position and orientation of the permanent magnet, we can use the equations 1.3 of the mathematical model of magnet to calculate the magnetic field intensity of the permanent magnet in every position in space. Conversely, if we know the magnetic field intensity in some specific locations in space, we can calculate the position and orientation of the magnetic source, that is, the position and orientation of the permanent magnet. In our magnetic localization system, we use the magnetic sensor array designed in chapter 2, which is shown as Figure 2.18, to measure the magnetic field intensity of the 16 specific points in space. The 16 LSM303D magnetic sensors can detect the magnetic field intensity in three orthogonal directions produced by the permanent magnet in real time when the target magnet move in the magnetic sensor array. According to the magnetic field intensity in three orthogonal directions measured by the 16 magnetic sensors, we can calculate the position and orientation of the permanent magnet by the accurate and efficient algorithm.

In the process of solving the problem, if we use the magnetic dipole model to calculate the magnetic field intensity, we can obtain the five dimensional space information (three dimensional position coordinate information and two dimensional orientation

CHAPTER 3. IMPLEMENTATION OF LOCALIZATION ALGORITHMS

coordinate information) of the target magnet. In general, the six dimensional spatial information (three dimensional position coordinate information and three dimensional rotation coordinate information which contains two dimensional direction coordinate information and one dimensional spin angle information) of the permanent magnet can be solved when the self characteristics of permanent magnet satisfies the following two conditions:

- When the permanent magnet rotates around its axis, no matter how much angle it rotates, the magnetic field intensity produced by the permanent magnet varies in space, that is to say, the variation of the magnetic field intensity can be detected during the rotation of the permanent magnet.
- The magnetic field model used in calculating the magnetic field intensity needs accord with the magnetic field model produced by the permanent magnet.

In the calculation, if the above two conditions are met at the same time, we can obtain the angle of rotation information of the permanent magnet which can not be obtained when using the magnetic dipole model to calculate. In our localization system, the permanent magnet we use is cylinder whose outside magnetic field intensity around the side is the same, therefore, we can not gain the angle of rotation information.

In this chapter, the five dimensional localization algorithm based on the magnetic dipole model and the shape of the given cylinder permanent magnet are mainly introduced.

3.1 Magnetic Localization Algorithms

3.1.1 Localization Algorithm

As we mentioned above in first chapter, in order to locate the position and orientation of the target permanent magnet, the error function,

$$E = \|(B'_{ix} - B_{ix}, B'_{iy} - B_{iy}, B'_{iz} - B_{iz})\|_2^2 = \sum_{i=1}^N [(B'_{ix} - B_{ix})^2 + (B'_{iy} - B_{iy})^2 + (B'_{iz} - B_{iz})^2], \quad (3.1)$$

needs to be minimized. As this is a nonlinear convex optimization problem, therefore, we choose the interior point method [25] to optimize the error function.

There exists a penalty function in the interior point method, which is used to describe convex sets. Different with the simplex method, it searches the optimal solution by traversing the inner feasible region. Here is the description of the linear programming problem:

$$\begin{aligned} & \min f(x) \\ & s.t. c(x) \geq 0, x \in R^n, c(x) \in R^m \end{aligned} \quad (3.2)$$

The logarithmic penalty function corresponding to 3.2 is:

$$B(x, \mu) = f(x) - \mu \sum_{i=1}^m \ln(c_i(x)) \quad (3.3)$$

The μ in 3.3 is a positive parameter which is often referred to as *penalty factor*. When μ converges towards zero, $B(x, \mu)$ will close to the solution of 3.2. And the gradient of the penalty function is:

$$g_b = g - \mu \sum_{i=1}^m \frac{1}{c_i(x)} \nabla c_i(x), \quad (3.4)$$

where g is the gradient of the primitive function $f(x)$ and $\nabla c_i(x)$ is the gradient of $c_i(x)$.

CHAPTER 3. IMPLEMENTATION OF LOCALIZATION ALGORITHMS

In addition to the primitive variable x , we also introduce the Lagrange multiplier $\lambda \in R^m$ (Sometimes also known as *slack variable*):

$$\forall_{i=1}^m c_i(x) \lambda_i = \mu \quad (3.5)$$

The 3.5 is called the complementarity condition of the disturbance sometimes which is similar to the complementary relaxation in the Karush-Kuhn-Tucker conditions (KKT conditions) [26]. We try to find the (x_μ, λ_μ) s that make the penalty function gradient be equal to zero.

Comparing 3.4 with 3.5, we can easily obtain an equation about gradient that is listed below.

$$g - A^T \lambda = 0 \quad (3.6)$$

In 3.6, A is the Jacobian matrix of the constrain conditions $c(x)$. And this equation shows that the gradient of $f(x)$ should be located in the subspace stretched by the conditional gradient. For 3.5 and 3.6, we can obtain the equation 3.7 listed below when using the Newton method.

$$\begin{pmatrix} W & -A^T \\ BA & C \end{pmatrix} \begin{pmatrix} P_x \\ P_\lambda \end{pmatrix} = \begin{pmatrix} -g + A^T \lambda \\ \mu - C \lambda \end{pmatrix} \quad (3.7)$$

Where W is the Hessian Matrix [27] of $f(x)$; B is the diagonal matrix of λ .

Because of the equations 3.2 and 3.5, the condition $\lambda \geq 0$ must be satisfied in each iteration, therefore we can calculate the equation 3.8 below by choosing a proper α .

$$(x, \lambda) \rightarrow (x + \alpha p_x, \lambda + \alpha p_\lambda) \quad (3.8)$$

3.1.2 Optimization Algorithm

Besides using the localization algorithm mention above to minimize the error function 3.1 in order to realize locating the position and orientation of magnet, we also need to calibrate the position and three coordinate axis direction of each sensor

CHAPTER 3. IMPLEMENTATION OF LOCALIZATION ALGORITHMS

according to the differences between the position and direction of sensors that we set and the actual position and direction of them; and we need to calibrate the value of B_T in equation 1.2 because the sensitivity of our LSM303D sensor which is digital magnetic sensor has been calibrated before being delivered from the factory, and the output of the LSM303D magnetic sensor.

For the processing of calibration, we need to optimize the error of the position and orientation of each sensor and the error of B_T . Therefore, the optimization algorithm need to be introduced. In our investigation, Levenberg-Marquardt algorithm [28] is an ideal algorithm to optimize the error mentioned above.

The Levenberg-Marquardt algorithm(LM algorithm) is an iterative procedure that locates the minimum of a multivariate function that is expressed as the sum of square of non-linear real-valued functions [29, 30, 31]like other numeric minimization algorithms . For the least square problem, the optimization goal is listed below as 3.9

$$f(x) = \frac{1}{2} \sum_{j=1}^m e_j^2(x) = \frac{1}{2} \|e(x)\|_2^2 \quad (3.9)$$

Where the $e(x)$ is a matrix shown below as 3.10.

$$e(x) = [e_1(x), e_2(x), \dots, e_m(x)]^T \quad (3.10)$$

Now, we can obtain the Jacobian matrix of $e(x)$, which is listed below as 3.11

$$\mathbf{J}^T(x) = \begin{Bmatrix} \frac{\partial e_1}{\partial x_1} & \frac{\partial e_2}{\partial x_1} & \dots & \frac{\partial e_m}{\partial x_1} \\ \frac{\partial e_1}{\partial x_2} & \frac{\partial e_2}{\partial x_2} & \dots & \frac{\partial e_m}{\partial x_2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial e_1}{\partial x_n} & \frac{\partial e_2}{\partial x_n} & \dots & \frac{\partial e_m}{\partial x_n} \end{Bmatrix} = \left[\frac{\partial e_j}{\partial x_i} \right]_{j=1,2,\dots,m, i=1,2,\dots,n} \quad (3.11)$$

After obtaining the Jacobian matrix, it is easy for us to calculate the gradient functions:

$$\begin{aligned} \nabla f(x) &= \mathbf{J}(x)^T e(x) \\ \nabla^2 f(x) &= \mathbf{J}(x)^T \mathbf{J}(x) + \sum_{j=1}^m e_j(x) \nabla^2 e_j(x) \end{aligned} \quad (3.12)$$

CHAPTER 3. IMPLEMENTATION OF LOCALIZATION ALGORITHMS

As we gain the Jacobian matrix mentioned above, we can obtain a part of the Hessian matrix which is important for us to decrease the amount of calculation, and it is the point that the algorithm can solve the least square problem efficiently. For all the other algorithms based on gradient descent, we approximate the solutions by Taylor series. And the Taylor series expansion of the equation 3.9 is listed below as 3.13

$$f(x) \approx f(x_0) + \mathbf{J}(f(x))|_{x_0}(x - x_0) + \frac{1}{2}(x - x_0)^T \mathbf{H}(f(x))|_{x_0}(x - x_0) + \dots \quad (3.13)$$

$$\nabla^2 r_j(x) = 0$$

If $\|e(x)\|$ is linear function, the Jacobian matrix of it is a constant matrix for all the j . Here, use the Taylor series to replace the functions 3.12 and solve the derivation we can get:

$$\nabla f(x) = \mathbf{J}^T(\mathbf{J}x + e) = 0 \quad (3.14)$$

And,

$$\mathbf{J}^T \mathbf{J}x = -\mathbf{J}^T e(x) \quad (3.15)$$

This is a linear least square problem with solutions. Ideally, we can see that this is:

$$\mathbf{A}x = \mathbf{B}, \quad (3.16)$$

where the $\mathbf{A} = \mathbf{J}^T \mathbf{J}$ is a square matrix and $\mathbf{B} = -\mathbf{J}^T e(x)$, and we can solve it as:

$$x = -(\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T e \quad (3.17)$$

However, the computation is very large and the solution is numerically unstable, in order to solve this problem, the LM algorithm introduce parameter λ to adjust the function 3.17. And the modified function is listed below as 3.18

$$x = -(\mathbf{J}^T \mathbf{J} + \lambda I)^{-1} \mathbf{J}^T e \quad (3.18)$$

Where I is the identity matrix. The (non-negative) damping factor λ is adjusted at each iteration. If reduction of $e(x)$ is rapid, a smaller value can be used, bringing the algorithm closer to the Gauss-Newton algorithm, whereas if an iteration gives

CHAPTER 3. IMPLEMENTATION OF LOCALIZATION ALGORITHMS

insufficient reduction in the residual, λ can be increased, giving a step closer to the gradient-descent direction. The damping factor also handles situations where the Jacobian is rank deficient and $\mathbf{J}^T \mathbf{J}$ is therefore singular. In this way, LM can defensively navigate a region of the parameter space in which the model is highly nonlinear. If the damping is small, the step of LM algorithm approximates the exact quadratic step appropriate for a fully linear problem. LM algorithm is adaptive because it controls its own damping factor λ : it raises λ if a step fails to reduce the e otherwise it reduces the damping. In this way, the LM algorithm is capable to alternate between a slow descent approach when being far from the minimum and a fast convergence when being at the neighborhood of minimum.

3.2 Calibrations

3.2.1 Calibration of B_T

For the analog magnetic sensor, the output is not the magnetic field intensity data, which can not be used directly, and there exists a factor between the magnetic field intensity and the analog magnetic sensor, whose value can be determined in calibration [32]. However, for the LSM303D digital sensor that we use, the sensitivity has been calibrated before leaving factory, which is mentioned above that the output of the LSM303D digital sensor is the magnetic field intensity which can be used directly. Therefore, the parameter B_T of 1.1 need to be modified. Because the parameter B_T is associated with the magnetic properties of a permanent magnet, so the parameter B_T should be determined before the localization process.

Given that the $(B'_{ix}, B'_{iy}, B'_{iz})^T$ is the output of the LSM303D sensor when the permanent magnet is put at the i -th position, the magnetic field intensity $(\tilde{B}_{ix}, \tilde{B}_{iy}, \tilde{B}_{iz})$

CHAPTER 3. IMPLEMENTATION OF LOCALIZATION ALGORITHMS

of the permanent magnet is defined as the equations 3.19 below.

$$\begin{cases} \tilde{B}_{ix} = \frac{3[m(x_l - a_i) + n(y_l - b_i) + p(z_l - c_i)] \cdot (x_l - a_i)}{R_i^5} - \frac{m_i}{R_i^3}; \\ \tilde{B}_{iy} = \frac{3[m(x_l - a_i) + n(y_l - b_i) + p(z_l - c_i)] \cdot (x_l - b_i)}{R_i^5} - \frac{n_i}{R_i^3}; \\ \tilde{B}_{iz} = \frac{3[m(x_l - a_i) + n(y_l - b_i) + p(z_l - c_i)] \cdot (x_l - c_i)}{R_i^5} - \frac{p_i}{R_i^3}. \end{cases} \quad (3.19)$$

Therefore the error function of the B_T calibration is:

$$E_{BT} = \sum_{i=1}^D (B'_{ix} - B_T \cdot \tilde{B}_{ix})^2 + \sum_{i=1}^D (B'_{iy} - B_T \cdot \tilde{B}_{iy})^2 + \sum_{i=1}^D (B'_{iz} - B_T \cdot \tilde{B}_{iz})^2 \quad (3.20)$$

Where the D is the number of sampled data. The parameter B_T can be calculated by the least square method, the Levenberg-Marquardt algorithm, mentioned above. Because of the measurement errors and noise, each B_T from different sensors is different slightly. Therefore, the final value of B_T is the average value of the different B_T .

3.2.2 Calibration of The Position and Orientation of Sensor

3.2.3 Sensor Position Calibration

Although we have set the position of each sensor, there still exists some differences between the preset position and actual position because of the error of manual placement. Thus, we need to minimize the error between preset and actual position of each sensor. We using the $(B'_{ix}, B'_{iy}, B'_{iz})^T$ as the actual output of the LSM303D sensor when the permanent magnet is put at the i -th, and the theoretical output of the sensor is $(B_{ix}, B_{iy}, B_{iz})^T$. Therefore, the error objective function can be defined as below:

$$E_{SP} = \sum_{i=1}^D (B'_{ix} - B_{ix})^2 + \sum_{i=1}^D (B'_{iy} - B_{iy})^2 + \sum_{i=1}^D (B'_{iz} - B_{iz})^2 \quad (3.21)$$

Where the E_{SP} is the optimization target, and we need to minimize the value of E_{SP} by the LM algorithm to obtain the sensor position (x_l, y_l, z_l) in $(B_{ix}, B_{iy}, B_{iz})^T$.

3.2.4 Sensor Orientation Calibration

As we mentioned above, there exists error when we place the sensor at the position we set, not only the position error, but also the orientation error will affect the localization accuracy. Therefore, the orientation of the sensor also need to be calibrated.

The $(B'_{ix}, B'_{iy}, B'_{iz})^T$ is defined as the sensor output when the magnet is put at the i -th position, and $(\hat{B}_{ix}, \hat{B}_{iy}, \hat{B}_{iz})$ is defined as the adjusted magnetic field intensity. Referring to the orientation adjustment in paper [5], the absolute value of the adjusting matrix O is equal to one, it is hard to control in calculation. Hence, we modified the adjusting matrix O into a matrix about the three angles: α, β, γ . So, the modified orientation adjustment function is described as 3.22 below.

$$\begin{bmatrix} \hat{B}_{ix} \\ \hat{B}_{iy} \\ \hat{B}_{iz} \end{bmatrix} = \begin{bmatrix} B'_{ix} \\ B'_{iy} \\ B'_{iz} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha \\ 0 & -\sin\alpha & \cos\beta \end{bmatrix} \begin{bmatrix} \cos\alpha & 0 & -\sin\beta \\ 0 & 1 & 0 \\ \sin\beta & 0 & \cos\beta \end{bmatrix} \begin{bmatrix} \cos\gamma & \sin\gamma & 0 \\ -\sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.22)$$

And now we can define the error function of the orientation calibration, the function is listed below as 3.23.

$$E_{SO} = \sum_{i=1}^D (\hat{B}_{ix} - B_{ix})^2 + \sum_{i=1}^D (\hat{B}_{iy} - B_{iy})^2 + \sum_{i=1}^D (\hat{B}_{iz} - B_{iz})^2, \quad (3.23)$$

where $(B_{ix}, B_{iy}, B_{iz})^T$ is calculated by 1.3 according to the preset position and direction of the magnet. Now we can obtain the minimal value of E_{SO} by using the LM algorithm.

3.2.5 Correction of The Geomagnetic Field

In most human activities, it is universally acknowledged that our upper part of the body is vertical, that is to say the axis of the vertebra points to the center of the earth, as the Figure 3.1 shows. Hence, we can ignore the geomagnetic field intensity, which can remove the impact of geomagnetic field.



Figure 3.1: The most common human activities (Walking, running and sitting)

3.3 Simulation of The Localization System

Before implementing the localization experiment, we need to simulate the localization system. Therefore we design a system with thirty-two sensors. There are two paths that we have designed, the first one is a spiral path and another one is rectangular path combined with a spiral path. The Figure 3.2 is the simulation of the spiral path, in this figure, there exist 100 points on the magnet motion path, and we process the simulation of these points, the average position error is $4.81mm$ and the orientation is 8.46° on average.

The Figure 3.3 is the simulation of the rectangular path combined with a spiral path and in this diagram, we simulate 100 points of the magnet motion path, among the 100 point, the first 50 points form the rectangular and the second . Consequently, the average position error is $7.12mm$ and the orientation is 9.90° on average.

As we can see, the localization accuracy of spiral path is higher than the rectangular path combined with a spiral path, that is because the initial point we use to optimize is updated during the locating process which is determined by that if the distance between the new locating point and the last locating point is less than $5cm$, the initial point will be updated as the new locating point. For the spiral path, the route

CHAPTER 3. IMPLEMENTATION OF LOCALIZATION ALGORITHMS

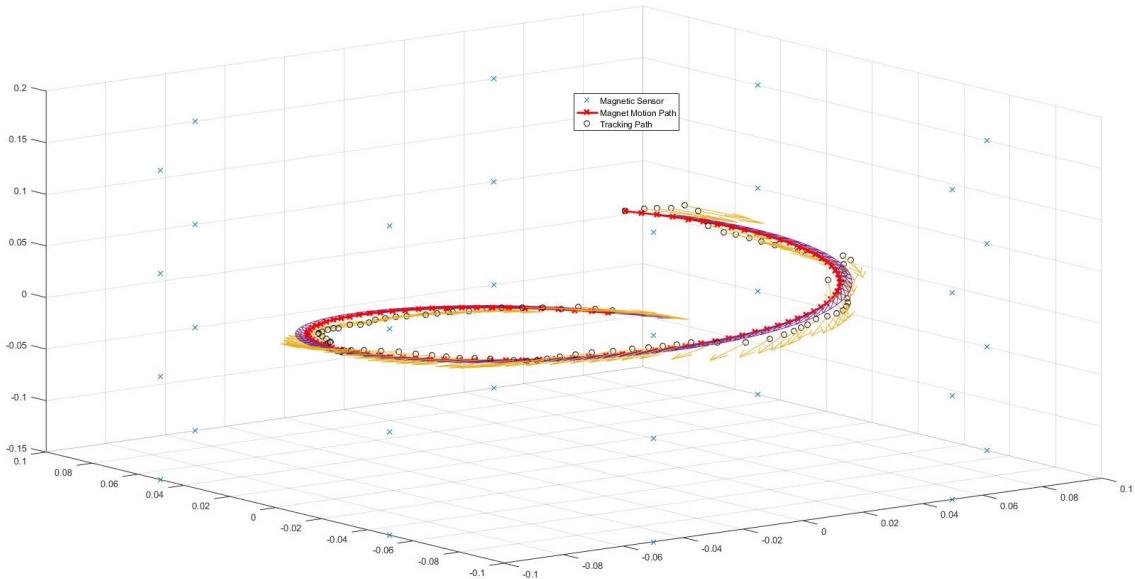


Figure 3.2: The simulation of the spiral path

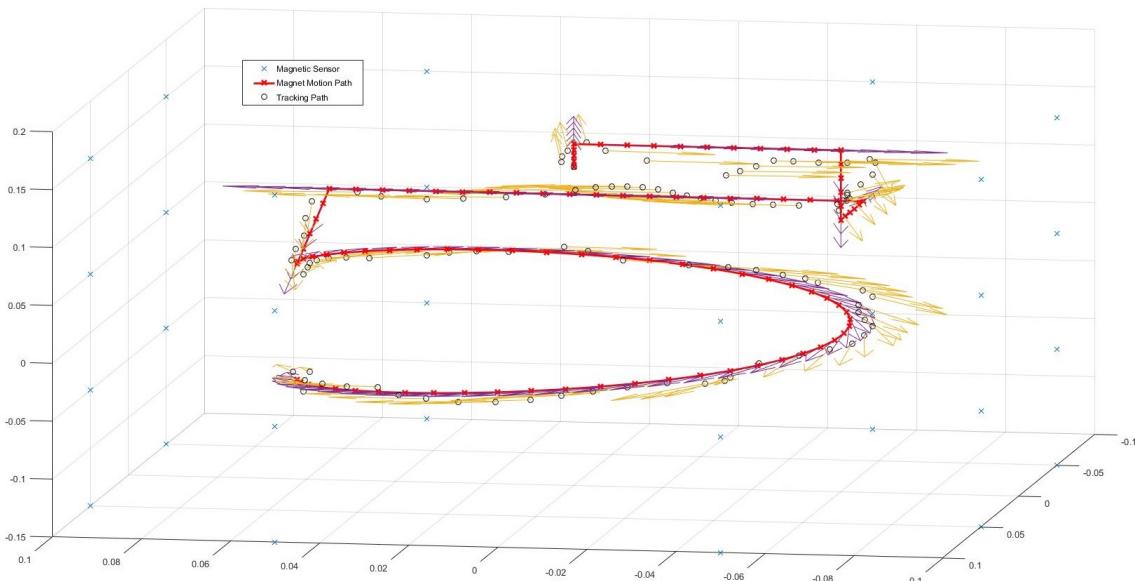


Figure 3.3: The simulation of the rectangular path combined with a spiral path

CHAPTER 3. IMPLEMENTATION OF LOCALIZATION ALGORITHMS

change is small in a tiny distance, thus the localization accuracy is high. However, for the rectangular path combined with a spiral path, most of the points with large error are from the rectangular path, that is because the change of the path is large at the corner of the path, the huge change of the path will affect the process of optimization. Therefore, the error of the second path is larger than the spiral path.

3.4 Results In Sixteen-Sensor Localization System

For implementing the localization algorithm mentioned above, we made a experiment on the sixteen-sensor platform. We measured 81 point in even plates with 3 different magnet postures. After processing the B_T calibration, sensor position and orientation calibration and the correction of geomagnetic field, we obtained acceptable results, the sensor three axes error of the position in even plates is shown below as Figure 3.4 and sensor three axes orientation error in even plates is shown as Figure 3.5.

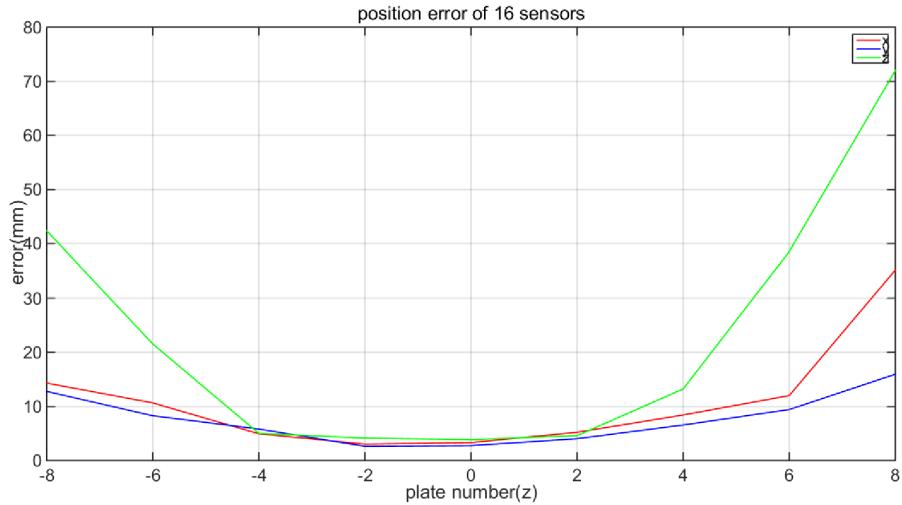


Figure 3.4: The three-axis position error of 16 sensors in even plates

Although the error of the position and orientation error above is acceptable, the accuracy is not high enough. Therefore, a novel hierarchical localization algorithm

CHAPTER 3. IMPLEMENTATION OF LOCALIZATION ALGORITHMS

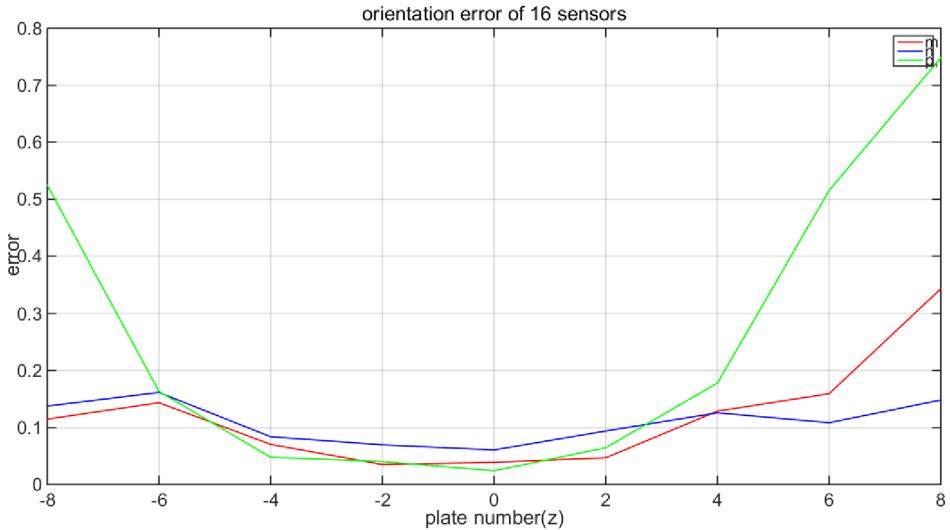


Figure 3.5: The three-axis orientation error of 16 sensors in even plates

with weighting is proposed. The hierarchical algorithm with weighting is an algorithm that the platform firstly is divided into three layers: the upper layer, middle layer and the bottom layer; and then we determine the layer that target is located according to the variance of the data; after that the sensor in each layer is weighted by the distance between the layer and the target, the larger the distance is, the lower the sensor's weight is. After implementing the hierarchical localization algorithm, we obtained better results.

Figure 3.6 depicts the sensor three axes error of the position in even plates and Figure 3.7 describes the sensor three axes orientation error in even plates. As we can see, the accuracy is improved a lot by the hierarchical localization algorithm with weighting. Table 3.1 shows the detail comparison of the two results (one is the result before processing the hierarchical localization algorithm and the other one is after adopting the hierarchical algorithm) mentioned above.

CHAPTER 3. IMPLEMENTATION OF LOCALIZATION ALGORITHMS

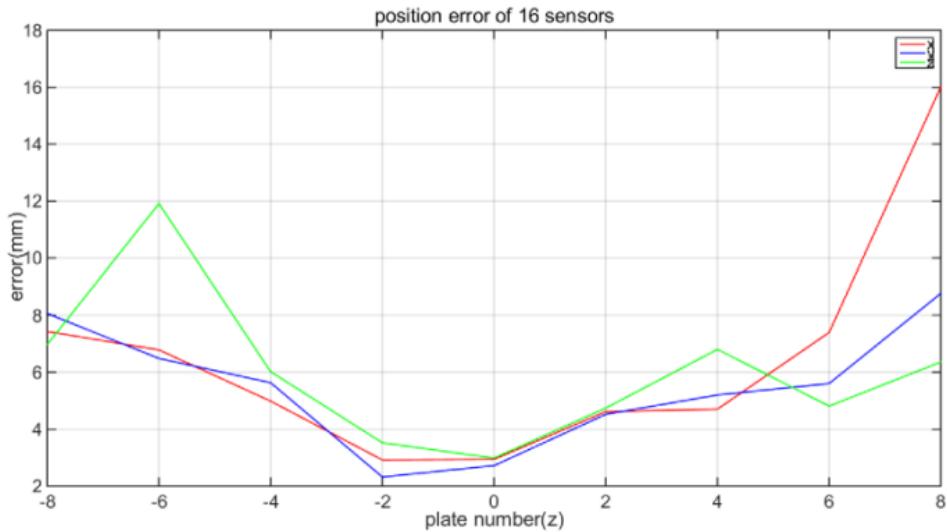


Figure 3.6: The three-axis position error of 16 sensors in even plates with the hierarchical localization algorithm with weighting

Table 3.1: Comparison Between Before and After Using Hierarchical Algorithm

Error	Without Hierarchical Algorithm	With Hierarchical Algorithm
X(mm)	11.56	6.42
Y(mm)	7.67	5.48
Z(mm)	22.96	6.01
Position(mm)	30.27	12.18
m	0.12	0.084
n	0.11	0.101
p	0.25	0.073
Orientation(°)	25.25	10.45

CHAPTER 3. IMPLEMENTATION OF LOCALIZATION ALGORITHMS

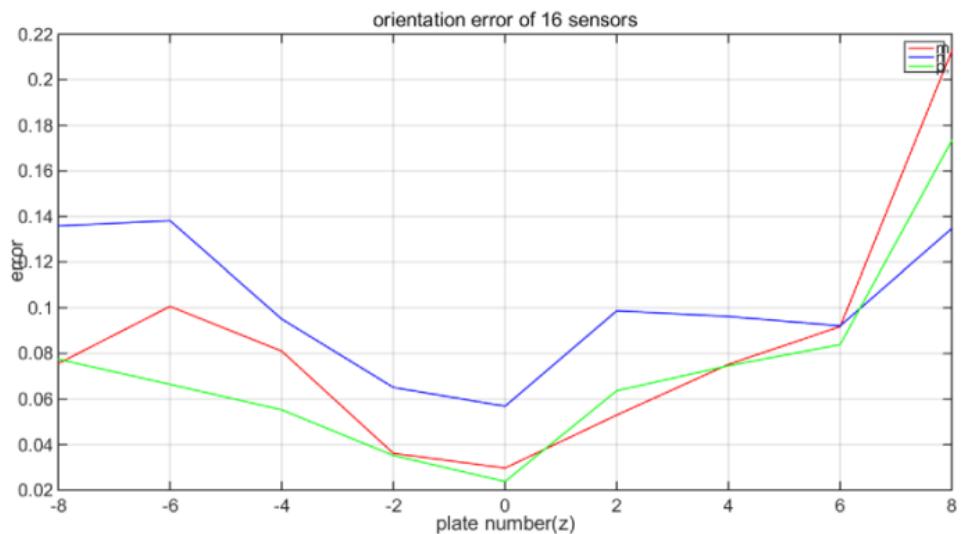


Figure 3.7: The three-axis orientation error of 16 sensors in even plates with the hierarchical localization algorithm with weighting

Chapter 4

The Improvement of Algorithm and Model

4.1 A Model of Thirty-two Sensors

In order to obtain more accurate localization results and minimize the error, we add sixteen more sensors to build a thirty-two-sensor system. If two sensors are close, the output of the two sensors are similar, which will lead a huge error, thus we put eight sensors at the top of the experiment platform and the other eight are added at the bottom of the system. The thirty-two-sensor system is shown as Figure 4.1 below.

4.2 Results of The Improved Model

We measured 81 point in even plates with 3 different magnet postures using the the localization algorithm with hierarchical algorithm based on the thirty-two sensors platform shown above and the accurate and efficient results was obtained. In Figure 4.2, the blue broken line represents the average position error of 32 sensors in even plates, and the red line is the average position error of 16 sensors; and in Figure 4.3, the blue broken line represents the average orientation error of 32 sensors in even plates, and the red line is the average orientation error of 16 sensors. As we can see

CHAPTER 4. THE IMPROVEMENT OF ALGORITHM AND MODEL

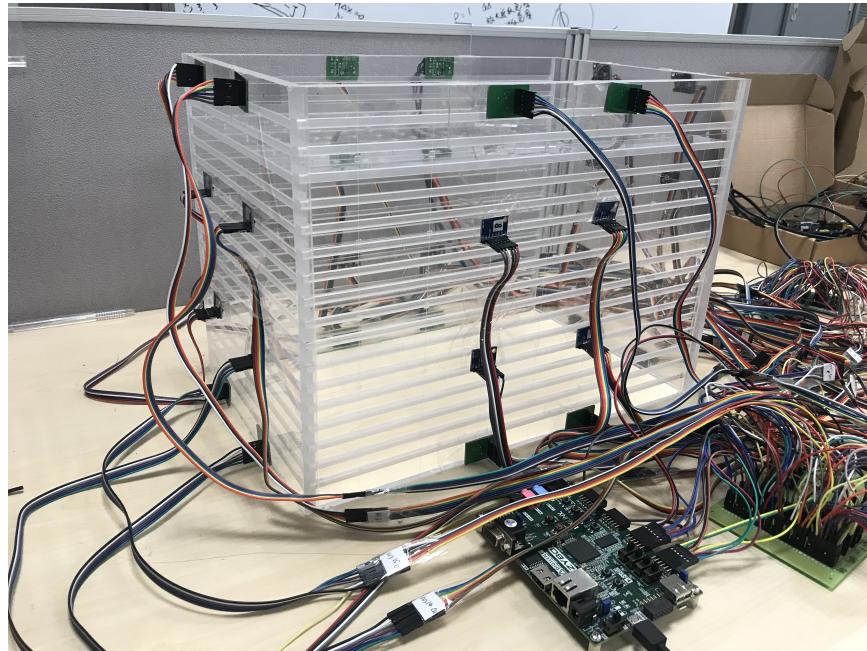


Figure 4.1: The thirty-two-sensor system

in these two pictures, the error of the thirty-two-sensor system is much more lower than sixteen-sensor system. For the thirty-two-sensor system, the average position error is 6.73mm and the average orientation error is 6.01° .

CHAPTER 4. THE IMPROVEMENT OF ALGORITHM AND MODEL

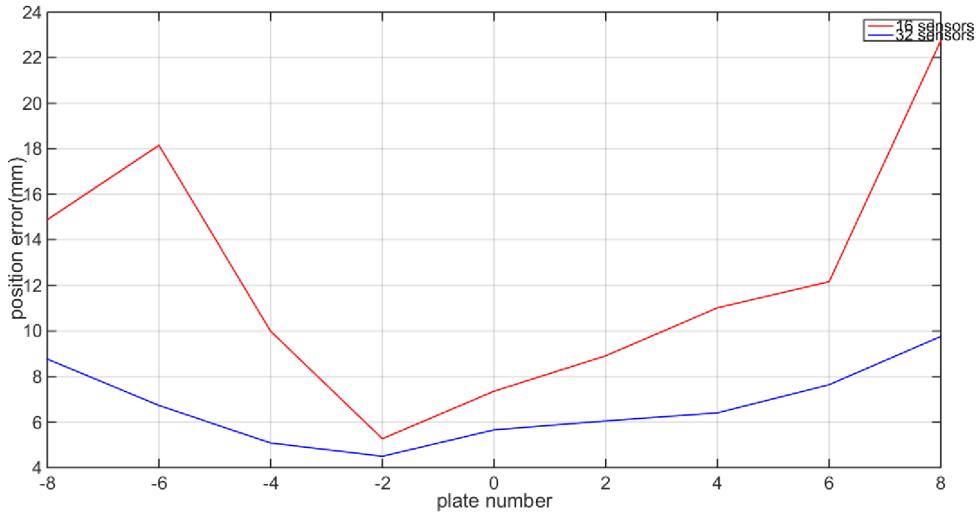


Figure 4.2: The average position error of 32 sensors in even plates

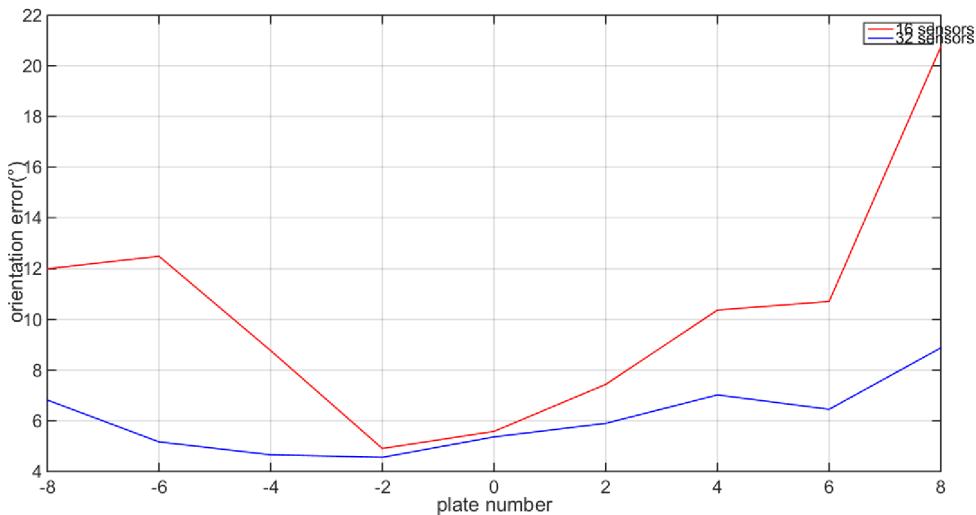


Figure 4.3: The average orientation error of 32 sensors in even plates

Chapter 5

Summary

5.1 Conclusion

From above analyses, we can see that using the novel hierarchical localization algorithm with weighting is a good method to minimize the error between the actual position and orientation of the permanent magnet target and the locating position and orientation. Combined with calibrations and hierarchical localization algorithm, in sixteen-sensor system, the error of position can be decreased from $30.27mm$ to $12.18mm$ and the orientation error can be reduced from 25.25° to 10.45° . With more sensors added, the accuracy of localization increases tremendously. In the thirty-two-sensor system, the error of sensor position can be reduced to only $6.73mm$ and the orientation error can be decreased to 6.01° , which can be considered that the localization results are accurate and efficient.

5.2 Future Work

This paper is about the investigation of magnetic localization, Due to the constraints of time, equipment and other objective conditions, the subject needs further improvement. The improvement can be mainly included the following aspects:

- As we can see that adding sensors can improve the accuracy, maybe add more sensors further can decrease the error between actual position and orientation

CHAPTER 5. SUMMARY

of the magnet target and the locating position and orientation.

- The research of sensor automatic diagnosis algorithm, because the system uses a large number of sensors, there will inevitably exist sensor abnormalities. The sensor diagnosis algorithm is designed to identify the invalid sensors and make the technical results more accurate.
- The research of the arrangement form of the sensor array, maybe the arrangement form of sensors can affect the localization results. The best way of arrangement form may be found in the future to increase the accuracy.
- Under mature conditions, we will conduct *in vivo* experiments to test the performance of the entire localization system and introduce the system to the market at an early date.

Bibliography

- [1] J. M. Remes-Troche and Y. Khan, “Longest duration of retention of video capsule: A case report and literature review,” vol. 5, pp. 352–5, 2013.
- [2] “Capsule endoscopy.” <https://en.wikipedia.org/wiki/Endoscopy>.
- [3] “Endoscopy.” <http://www.givenimaging.com/en-int/Innovative-Solutions/Capsule-Endoscopy/Pages/default.aspx>.
- [4] “Understanding capsule endoscopy.” <https://www.asge.org/list-pages/patient-informations/understanding-capsule-endoscopy>.
- [5] S. Shijian, Y. Wan, D. Houde, X. Xuke, L. Mingqiang, S. Bo, and H. Chao, “Investigation of the relationship between tracking accuracy and tracking distance of a novel magnetic tracking system,” vol. 17, pp. 1558–1748, August 2017.
- [6] Y. Barrell and H. W. L. Naus, “Detection and localisation of magnetic objects,” vol. 1, pp. 245–254, September 2007.
- [7] C. Hu, M. Li, S. Song, R. Yang, W. Zhang, and M. Q.-H. Meng, “A cubic 3-axis magnetic sensor array for wirelessly tracking magnet position and orientation,” vol. 10, pp. 903–913, May 2010.
- [8] C. D. Natali, M. Beccani, N. Simaan, and P. Valdastri, “Jacobian-based iterative method for magnetic localization in robotic capsule endoscopy,” vol. 32, pp. 327–328, April 2016.

BIBLIOGRAPHY

- [9] J. T. Sherman, J. K. Lubkert, R. S. Popovic, and M. R. DiSilvestro, “Characterization of a novel magnetic tracking system,” vol. 43, pp. 2725–2727, June 2007.
- [10] J. Wahlström, N. Callmer and F. Gustafsson, “Magnetometers for tracking metallic targets,” pp. 1–8, July 2010.
- [11] C. Hu, X. Ren, Y. You, and W. Yang, “Locating intra-body capsule object by three-magnet sensing system,” vol. 16, pp. 5167–5176, April 2016.
- [12] E. Paperno, I. Sasada, and E. Leonovich, “A new method for magnetic position and orientation tracking,” vol. 37, pp. 1938–1940, July 2001.
- [13] C. Hu, S. Song, X. Wang, and Q. H. Meng, “A novel positioning and orientation system based on three-axis magnetic coils,” vol. 48, pp. 2211–2219, July 2012.
- [14] C. Guo, X. Song and R. Yan, “Optimization of multilayer cylindrical coils in a wireless localization system to track a capsule-shaped microdevice,” vol. 46, pp. 117–124, January 2013.
- [15] S. Song, X. Qiu, J. Wang, and M. Q. H. Meng, “Design and optimization strategy of sensor array layout for magnetic localization system,” vol. 17, pp. 1849–1857, March 2017.
- [16] J. Hanji, Z. Wanhua, and F. Guangyou, “A review on ferromagnetic induction coil sensors,” vol. 25.
- [17] “Three-axis magnetic sensor,” HMC1043 Datasheet, Honeywell, August 2012.
- [18] “Ultra-compact high-performance ecompass module: 3d accelerometer and 3d magnetometer,” LSM303D Datasheet, ST Microelectronics, November 2013.
- [19] “Zybo™ FPGA Board Reference Manual,” DIGILENT®, February 2013.

BIBLIOGRAPHY

- [20] “STM32F101xx, STM32F102xx, STM32F103xx, STM32F105xx and STM32F107xx advanced Arm®-based 32-bit MCUs,” RM0008 Reference manual, ST Microelectronics, March 2018.
- [21] “STC89C51(52)RC/RD+series MCU Datasheet,” STC MCU Limited., July 2011.
- [22] W. Durfee, “Arduino Microcontroller Guide,” University of Minnesota, October 2011.
- [23] EDN, “The vivado design suite accelerates programmable systems integration and implementation by up to 4x,” Jun 15, 2012.
- [24] C. Maxfield, “Webpack edition of xilinx vivado design suite now available,” in *EE Times*, Dec 20, 2012.
- [25] “Interior point method.” https://en.wikipedia.org/wiki/Interior_point_method.
- [26] “Karush-kuhn-tucker conditions.” https://en.wikipedia.org/wiki/Karush-Kuhn-Tucker_conditions.
- [27] “Hessian matrix.” https://en.wikipedia.org/wiki/Hessian_matrix.
- [28] J. J. Moré, “The levenberg-marquardt algorithm: Implementation and theory,” 630: 105-116, 1978.
- [29] K. Levenberg, “A method for the solution of certain non-linear problems in least squares,” 2(2):164-168, Jul. 1944.
- [30] H. Madsen, K. Nielsen and O. Tingleff, “Methods for non-linear least squares problems. technical university of denmark,” 2014.
- [31] D. Marquardt, “An algorithm for the least-squares estimation of nonlinear parameters,” 1(2):431-441, Jun.1963.

BIBLIOGRAPHY

- [32] S. Shijian, Y. Wan'an, D. Houde, X. Xuke, L. Mingqiang, S. Bo, and H. Chao, “A new calibration method for magnetic sensor array for tracking capsule endoscope,” pp. 1561–1566, December 2009.