

Túlio Gomes Pereira

**AN INDOOR POSITIONING SYSTEM BASED ON
ULTRA WIDEBAND MEASUREMENTS FOR PLANAR
CABLE-DRIVEN ROBOTS LOCALIZATION DURING
PAYLOAD ANALYSIS**

Trabalho submetido ao Departamento
de Engenharia Elétrica e Eletrônica
para a obtenção do Grau de Bacharel
em Engenharia Eletrônica.
Orientador
Instituto SENAI de Inovação em Sis-
temas Embarcados: Dr. Leonardo Kessler
Slongo
Coorientador
Fraunhofer ENAS: Dr. Jörg Martin

Florianópolis

2019

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Pereira, Túlio Gomes

An indoor positioning system based on Ultra WideBand measurements for planar cable-driven robots localization during payload analysis. / Túlio Gomes Pereira ; orientador, Leonardo Kessler Slongo, coorientador, Fernando Rangel de Souza, 2019.

101 p.

Trabalho de Conclusão de Curso (graduação) - Universidade Federal de Santa Catarina, Centro Tecnológico, Graduação em Engenharia Eletrônica, Florianópolis, 2019.

Inclui referências.

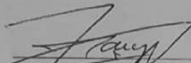
1. Engenharia Eletrônica. 2. Indoor Positioning. 3. UWB. 4. Localization. I. Kessler Slongo, Leonardo. II. Rangel de Souza, Fernando. III. Universidade Federal de Santa Catarina. Graduação em Engenharia Eletrônica. IV. Título.

Túlio Gomes Pereira

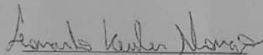
AN INDOOR POSITIONING SYSTEM BASED ON ULTRA WIDEBAND
MEASUREMENTS FOR PLANAR CABLE-DRIVEN ROBOTS LOCALIZATION
DURING PAYLOAD ANALYSIS

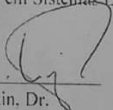
Este Trabalho Conclusão de Curso foi julgado adequado para obtenção do Título de "Bacharel em Engenharia Eletrônica" e aprovado em sua forma final pelo Curso de Engenharia Eletrônica.

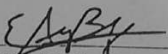
Florianópolis, 13 de Novembro de 2019.

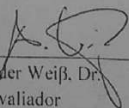

Prof. Fernando Rangel de Souza, Dr.
Coordenador do Curso

Banca Examinadora:


Leonardo Kessler Slongo, Dr.
Orientador
Instituto SENAI de Inovação em Sistemas Embarcados


p.p. Jörg Martin, Dr.
Coorientador
Fraunhofer ENAS


Prof. Eduardo Augusto Bezerra, Ph.D.
Avaliador
Universidade Federal de Santa Catarina


Alexander Weiß, Dr.
Avaliador
Fraunhofer ENAS

To my family.

AGRADECIMENTOS

Dedico este trabalho aos meus pais, Mônica de Sá Gomes Pereira e Pedro Paulo Pereira, e ao meu irmão Caio Gomes Pereira, pois sem o apoio deles, nada disso teria sido possível. Além disso, gostaria de agradecer a toda a minha família por sempre estar ao meu lado, e sem eles eu não teria chego até aqui. Em especial, deixo meu agracecimento ao meu tio, Tiago de Sá Gomes, que sempre ajudou a me guiar, e me aconselhou durante toda a minha vida.

Agradeço também a todos os membros do grupo de óptica e sistemas baseados em nanocompositos, ONKS, principalmente ao Dr. Jörg Martin e ao Dr. Alexander Weiß, pela receptividade, pelo suporte neste trabalho, pela oportunidade que me deram de realizar um grande sonho e pelos ensinamentos durante meu período na Alemanha.

Não posso deixar de agradecer ao meu orientador, Dr. Leonardo Kessler Slongo, que me guiou na condução deste trabalho e abriu as portas para o desenvolvimento do projeto FloripaSat, onde iniciei a base da minha carreira profissional. Além disso, ele me mostrou que a engenharia estava muito além do que apenas a sala de aula, e que com dedicação qualquer aprendizado é possível. Também agradeço ao professor Eduardo Augusto Bezerra, pela criação do projeto FloripaSat, e pela oportunidade de ter feito parte dele. Não posso deixar de agradecer também aos meus colegas Sara Vega Martinez e Bruno Eiterer, por todo o aprendizado que me deram, assim como todos os colegas da equipe FloripaSat.

Por fim, mas não menos importante, agradeço aos meus colegas de curso, que estiveram ao meu lado durante toda a graduação e tornaram a caminhada mais leve e divertida. Deixo meu agradecimento especial ao Felipe de Souza Tortato, que se tornou um grande amigo, me deu suporte e me apresentou pessoas incríveis. Além dele, agradeço também aos colegas e amigos: Fernando Lonzetti, José Arbugeri, Maicon Francisco, Ericson Meier, Pedro Fornari, e todos os que estiveram comigo durante a graduação.

RESUMO

Este trabalho aborda o desenvolvimento de um sistema completo de posicionamento indoor para análise de cargas úteis em robôs planares guiados por cabo. Inicialmente, um sistema de posicionamento é definido comparando-se tecnologias utilizadas nesta área. Assim, um sistema ultra wideband é proposto, no qual o kit de desenvolvimento da empresa Pozyx é utilizado, e modificado para ser aplicado na estrutura do robô. Portanto, uma análise dos dois algoritmos, TRACKING e UWB ONLY, que estão disponíveis pelo fabricante para o cálculo do posicionamento é feita. Em seguida uma combinação deles é proposta para ser utilizada nesta aplicação. Logo, a performance dos algoritmos é avaliada em testes realizados em um laboratório, de 4x4 metros de área, sendo que o algoritmo proposto foi escolhido para ser utilizado na versão final. Este algoritmo apresentou melhores resultados do que os demais, com a maior probabilidade de obter erros abaixo de 6 cm, além de atingir um erro médio menor do que 5 cm nos pontos medidos. Finalmente, uma placa de circuito impresso e uma estrutura mecânica são desenvolvidas para que o sistema esteja completo para ser aplicado no robô.

Palavras-chave: Posicionamento. Ultra wideband. Robô. Exatidão. Algoritmo.

ABSTRACT

The present work covers the development of a complete positioning system for planar cable-driven robot's payload analysis. First of all, an positioning system is defined after technologies used in this field have been compared. Thus, an ultra wideband system is proposed, where the development kit from Pozyx company is used, and modified to be applied in the robot's structure. Thus, an analysis of the two algorithms, TRACKING and UWB ONLY, available from the manufacturer for the position estimation is performed, and a combination of them is proposed to be used in this application. Then, the performance of the algorithms is evaluated in tests executed inside a laboratory, with an area of 4x4 meters, and the proposed has been chosen to be used in the final version. This algorithm has presented the best results among others, with the greatest probability of achieving errors smaller than 6 cm, as well as an average error smaller than 5 cm in the measured points. Finally, a printed circuit board and a mechanical structure are developed to let the system complete to be applied in the real robot.

Keywords: Positioning. Ultra wideband. Cable-driven robot. Accuracy. Algorithm.

LIST OF FIGURES

Figure 1	Example of a typical multipath effect: (a) pulse to be transmitted; (b) the same pulse traveling different distances; (c) a second copy of the pulse arriving at different time at the receiver; (d) wrong pulse received by the radio receiver.....	25
Figure 2	Example of a vertical planar cable-driven robot structure.	26
Figure 3	Basic concept of satellite positioning.	30
Figure 4	Trilateration simplified example.	39
Figure 5	Typical ToA example.	40
Figure 6	TWR explanation: (a) transmitter sends the data; (b) receiver process and sends the data back; (c) time-line of the system.	41
Figure 7	Pozyx prototyping kit.....	47
Figure 8	Raspberry Pi 3 Model B+.....	49
Figure 9	Arduino Model Uno Rev3.....	49
Figure 10	Block diagram of the proposed system.	50
Figure 11	Pozyx antenna radiation directions.....	51
Figure 12	Mechanical structure for anchors position.	52
Figure 13	Top view of the measurement area.	52
Figure 14	Test area ground truth.	53
Figure 15	TRACKING algorithm average error in x-axis measurements of 10 minutes in each point.	54
Figure 16	UWB ONLY algorithm average error in x-axis measurements of 10 minutes in each point.	54
Figure 17	UWB parameters evaluation.	56
Figure 18	Firmwares Flowchart. (a) Flowchart of the Raspberry's firmware; (b) Flowchart of the Arduino's firmware.	58
Figure 19	Prototype of the system used in the experiments.	59
Figure 20	Test plan. (a) First test plan; (b) Second test plan.	60
Figure 21	Histograms of the errors of the algorithms.....	61
Figure 22	Probability density functions of the errors of the algorithms.....	62
Figure 23	Schematic of the flyback converter used to power the embedded system in the robot with 24V.....	64
Figure 24	Assembly in 3D of the PCB embedded in the robot. ...	65

Figure 25 PCB under evaluation in the test bench. 66

Figure 26 Relation of the errors in the cartesian plane. 67

Figure 27 TRACKING algorithm results in x-axis at point 10. ... 68

Figure 28 TRACKING algorithm results in y-axis at point 10. ... 68

Figure 29 TRACKING algorithm average error in x-y plane. 70

Figure 30 TRACKING algorithm standard deviation in each point. 70

Figure 31 Rotation of the anchors in clockwise. 71

Figure 32 UWB ONLY algorithm results in x-axis at point 10. ... 72

Figure 33 UWB ONLY algorithm results in y-axis at point 10. ... 73

Figure 34 UWB ONLY algorithm average error in x-y plane. 73

Figure 35 UWB ONLY algorithm standard deviation in each point. 74

Figure 36 Proposed algorithm results in x-axis at point 10. 75

Figure 37 Proposed algorithm results in y-axis at point 10. 75

Figure 38 Proposed algorithm average error in x-y plane. 76

Figure 39 Proposed algorithm standard deviation in each point. ... 77

Figure 40 Example of the first measurement during the second test. 79

Figure 41 TRACKING algorithm results for the second test plan. 79

Figure 42 UWB ONLY algorithm results for the second test plan. 80

Figure 43 Proposed algorithm results for the second test plan. 80

Figure 44 Comparison between algorithms average measurement 5 times in each point. 81

Figure 45 Cumulative distribution function of the algorithms. 82

Figure 46 PCB efficiency analysis. 84

Figure 47 PCB voltage regulation with load current analysis. 84

Figure 48 Result of the embedded PCB. 85

Figure 49 Result of the embedded PCB. 86

Figure 50 System placement inside the robot's structure. 87

Figure 51 PCB schematic of the AS-Interface circuit sheet 1. 99

Figure 52 PCB schematic of the AS-Inteface circuit sheet 2. 100

Figure 53 PCB schematic of the flyback converter and connectors sheet. 101

LIST OF TABLES

Table 1	Comparison between technologies.....	37
Table 2	Comparison between related works.....	46
Table 3	PCB Power consumption estimation.....	63
Table 4	First experiment results.....	78
Table 5	Analysis of the CDF for erros in x-y plane.....	82
Table 6	Average error of all measurements in x-y plane.....	83

LIST OF ABBREVIATIONS AND ACRONYMS

AP	Access Point
AoA	Angle of Arrival
CDF	Cumulative Density Function
GPIO	General-purpose Input/Output
GPS	Global Positioning System
ISM	Industrial, Scientific and Medical
IMU	Inertial Measurement Unit
LOS	Line of Sight
MAC	Medium Access Control
NLOS	Non-line of Sight
PHY	Physical Layer
PCB	Printed Circuit Board
PDF	Probability Density Function
RF	Radio Frequency
RFID	Radio Frequency Identification Device
RSS	Received Signal Strength
RSSI	Received Signal Strength Indicator
3D	Three-Dimensions
ToA	Time of Arrival
2D	Two-Dimensions
UWB	Ultra Wideband
WLAN	Wireless Local Area Network

LIST OF SYMBOLS

x	Cartesian plane x-axis
y	Cartesian plane y-axis
z	Cartesian plane z-axis
ϱ^s	Satellite distance to the center of the earth
ϱ_R	Receiver distance to the center of the earth
ϱ	Distance between satellite and receiver
P_{Rx}	Received power
P_{Tx}	Transmitted power
G_{Rx}	Receiver's antenna gain
G_{Tx}	Transmitter's antenna gain
λ	Wavelength
D	Distance
v	Velocity
t	Time of flight
T_{total}	Total time
T_{proc}	Processing time
μ	Mean value
σ	Standard deviation
σ^2	Variance

CONTENTS

1	INTRODUCTION	23
1.1	MOTIVATION	24
1.2	GOALS	26
1.2.1	Main Goal	26
1.2.2	Specific Goals	26
1.3	MANUSCRIPT STRUCTURE	26
2	LOCALIZATION SYSTEMS ANALYSIS	29
2.1	GLOBAL POSITIONING SYSTEM	29
2.2	INERTIAL MEASUREMENT UNITS	30
2.3	RADIO FREQUENCY IDENTIFICATION DEVICE	31
2.4	ZIGBEE	32
2.5	BLUETOOTH	33
2.6	WIFI	33
2.7	ULTRA WIDEBAND	34
2.8	COMPARISON BETWEEN TECHNOLOGIES	35
2.9	TECHNIQUES AND ALGORITHMS	38
2.9.1	Trilateration	38
2.9.2	Time of Arrival	39
2.9.2.1	Two Way Ranging	40
3	RELATED WORK	43
4	METHODOLOGY	47
4.1	ULTRA WIDE BAND SYSTEM FROM POZYX	47
4.2	RASPBERRY PI	48
4.3	ARDUINO	49
4.4	PROPOSED SYSTEM	50
5	DEVELOPMENT	51
5.1	POZYX DATA BEHAVIOR	53
5.2	COMMUNICATION PARAMETERS SELECTION	55
5.3	FIRMWARE	57
5.4	DATA EVALUATION	61
5.5	POWER SUPPLY	63
6	RESULTS	67
6.1	FIRST EXPERIMENT	67
6.1.1	TRACKING Algorithm	67
6.1.2	UWB ONLY Algorithm	72
6.1.3	Proposed Algorithm	74
6.1.4	First Experiment Results Comparison	77

6.2	SECOND EXPERIMENT	78
6.3	POWER SUPPLY EVALUATION	83
6.4	SYSTEM PLACEMENT	85
7	CONCLUSION	89
7.1	FUTURE WORK	89
	REFERENCES	91

1 INTRODUCTION

Since the beginning of the human exploration of the world across the sea, human beings have the necessity to localize themselves in space. At that time, they used specific constellations as reference points for maritime navigation. Thus, they were able to achieve back their homeland after some time on the ocean. However, due to the stormy weather and changes in the wind condition, ships often got lost during the navigation. So passing the time, compass, chronometer and other tools were being developed to improve these localization skills (GALAT; LOWE, 2012).

With the improvement in the knowledge of electromagnetic waves behavior, localization systems achieved a higher level of complexity and accuracy. With the discovering of electromagnetic radiation in 1887 by the young German scientist, Heinrich Hertz, who proved physically the electromagnetic theory proposed by James Clerk Maxwell, the studies in wireless communication started to become more and more expressive (BUCHWALD, 1994).

Thanks to the advance of technology, many approaches and architectures have been proposed and developed in order to achieve systems requirements for the most diverse applications, such as medicine, humans or objects tracking, robots positioning and others (ZHANG et al., 2010; LEDERGERBER; HAMER; D'ANDREA, 2015; VAHDATPOUR; AMINI; SARRAFZADEH, 2011).

As localization can be defined as the action of providing position information of an object in space, one can distinguish in two main types of localization systems: outdoor and indoor positioning (RIBEIRO et al., 2018; ZAFARI; GKELIAS; LEUNG, 2017). These two categories of systems are chosen according to the target environment, for instance, in an outdoor context the use of global navigation satellite systems (GNSS), such as the global positioning system (GPS), has been widely chosen. These systems are used as a solution for applications that require long distances measurements, for example: to track objects in high speed; to guide people or for transport navigation. However, they are not suitable for an indoor or an underground environment, due to the fact that signals from GPS satellites must handle interference with objects in the line of sight, such as buildings or walls, making them ineffective for indoor localization (FARID; NORDIN; ISMAIL, 2013).

On the other hand, according to (ZHANG et al., 2010) indoor positioning can be defined as a system that is able to provide the position

of a person or a thing inside of a closed structure, such as hospitals, stadiums, laboratories, universities, etc. These systems must be able to handle interference caused by obstacles that are in the radios' line of sight (LOS), (like walls, furniture, human beings, etc.), and introduce many adversities for the electromagnetic waves propagation, e.g. reflections, obstructions, multipath-effect, and noise interference. Although many approaches have been developed in order to handle these difficulties (FARID; NORDIN; ISMAIL, 2013; ZAFARI; GKELIAS; LEUNG, 2017), each solution has advantages and disadvantages, and a lot of research still being required in this area to improve the weakness of each system. So, one must choose the solution that fits better to the desired application.

1.1 MOTIVATION

Due to the increasing interest in autonomous robots, the localization problem in indoor environments has received new attention in the past few years with the development of new techniques that improved the precision of the measurements (BRESSON et al., 2017).

In this field, a fundamental problem that makes the localization more difficult in indoor environments is the multipath effect. This effect is caused by obstacles or sources of reflection near the system network. Thus, all data sent during the transmission are subject to suffer from reflections, traveling different distances and arriving in the receiver at different times, causing the data to be overwritten and consequently lost (ALARIFI et al., 2016).

For instance, Figure 1 shows an example of a typical indoor scenario. First of all, a pulse carrying data is created and transmitted from the radio transmitter in an omnidirectional manner. Then, this pulse travels different distances due to the reflections caused by objects or walls, and the copies of this signal arrive in the receiver in slightly distinct times, causing communication errors (GHAVAMI; MICHAEL; KOHNO, 2007).

Another problem frequently observed in indoor environments is the obstruction of the line of sight between the transmitter and the receiver, which introduces fading in the communication link. This effect is also called non-line of sight (NLOS) situation (ALARIFI et al., 2016).

Considering these issues, the main motivation of this work is to develop a positioning system for the "HoFaM" project. This project consists of an autonomous planar cable-driven robot, which has been

developed in partnership between Fraunhofer ENAS and Technische Universität Chemnitz, in Germany. The structure of the robot, allows a payload to be moved in a two dimensional (2D) way around a whole wall, where motors are used to change the tension in four cables. An example of a similar structure is shown in Figure 2 (BRUCKMANN et al., 2008).

However, the problem in controlling cable-driven robots is related to the bending fatigue in the rope, where the system loses accuracy with passing time. In addition, when placed near a wall, many sources of electromagnetic waves reflection emerge, which increases the multipath problem. Also, many metals are present in the structure, which introduces challenges to the radios communication. Therefore, aiming to have a precise position data for payload analysis, an accurate localization system must be developed to be used in this scenario.

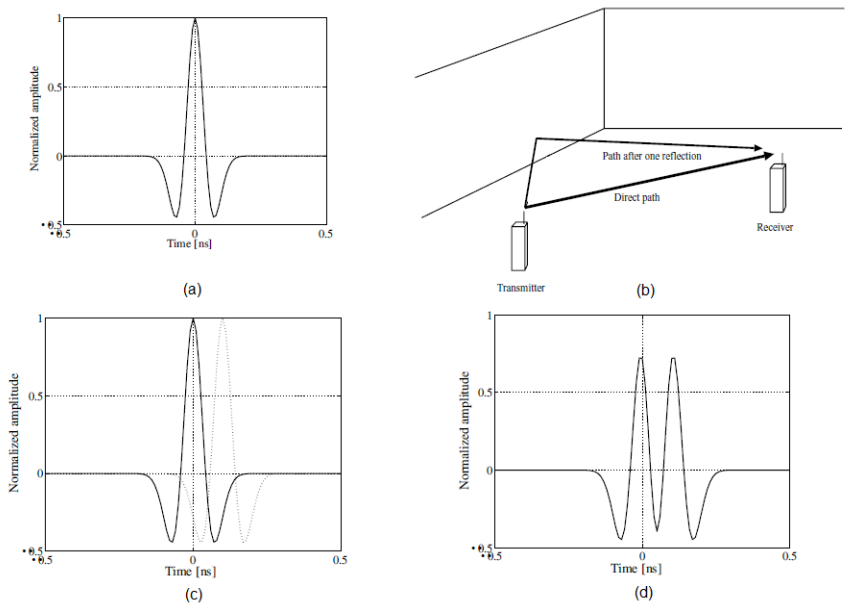


Figure 1 – Example of a typical multipath effect: (a) pulse to be transmitted; (b) the same pulse traveling different distances; (c) a second copy of the pulse arriving at different time at the receiver; (d) wrong pulse received by the radio receiver.

Source: Adapted from (GHAVAMI; MICHAEL; KOHNO, 2007).



Figure 2 – Example of a vertical planar cable-driven robot structure.
Source: Adapted from (BRUCKMANN et al., 2008).

1.2 GOALS

1.2.1 Main Goal

The main goal of this bachelor thesis is to define, apply and evaluate an indoor localization system for autonomous cable-driven robots, as a feedback positioning system, during payload analysis.

1.2.2 Specific Goals

Considering the main goal, this work will present different technologies, used for indoor positioning measurements, based in RF signals and inertial measurements. After a detailed description of the advantages and disadvantages in each technology, the most suitable one will be adopted. Then, this system will be developed and analyzed to meet the projects requirements. At the end of this work, it is expected to have a fully working positioning system, capable of self localizing a tag embedded in the robot, with an accuracy better than 10 cm, working in a 2D area of 4 x 4 meters, and providing the position data in two seconds after a request is received.

1.3 MANUSCRIPT STRUCTURE

The present work is organized in seven main chapters. First of all, an introduction is presented, where the main differences between indoor and outdoor positioning systems are stated. Also, the motivation and the goals for the development of this work are described.

Chapter 2: In the second chapter, the most common localization system technologies based on radio frequency signals are described, and in the end, one of the technologies is chosen to be used in this project, after all of them are compared in detail. Then, the technique used for calculating the position is shown as well as the geometric properties which allow this calculation.

Chapter 3: In this chapter, a review of the localization systems based on the technology chosen is shown. Then, a table comparing the related works is also provided.

Chapter 4: In this chapter, the project development methodology is presented. Then, the components and sub-systems are described, and a complete system is proposed to connect them.

Chapter 5: This is the chapter where the development of the system is explained. Here, the behavior of the sub-system is analyzed, and the firmware to connect all the components is described. In the end, the test plan is defined in order to analyze the algorithms available to be used, and a printed circuit board to power the system is developed.

Chapter 6: In this chapter, the results are reported. First, two algorithms available from the sub-system manufacturer are tested, from the analysis of the behavior of these algorithms, a combination mixing both is proposed. So the three algorithms are tested under the same test plan, and an analysis comparing their performance is presented aiming to choose the one that fits better the application in this work. Besides, other components of the system, such as mechanical structures, and power supply are presented, as well as their placement in the robot structure, aiming to let the system completely able to be deployed.

Chapter 7: This is the chapter where the conclusions and the future work are presented.

2 LOCALIZATION SYSTEMS ANALYSIS

2.1 GLOBAL POSITIONING SYSTEM

The Global Positioning System (GPS) is the world's largest positioning system used nowadays. It was developed by the necessity of improving the Navy Navigation Satellite System (NNSS), so-called TRANSIT, which was created by the U.S. military, to determine the coordinates of vessels and aircraft. The TRANSIT was composed of six satellites orbiting in a nearly circular polar orbit in altitudes of about 1 100 km. However, the relatively low accuracy and the large time gaps in coverage, were the main problems that resulted in the development of the GPS system (HOFMANN-WELLENHOF; LICHTENEGGER; COLLINS, 2012).

To overcome these barriers, GPS uses 21 satellites placed in 12-hour circular orbits, inclined 55° to the equatorial plane. Thus, this arrangement has proved to provide at least four satellites in good geometric position 24 hours per day, anywhere on the earth. Hence, all satellites in this constellation broadcast a package with their own position and precise clock, such as the atomic clock, which can be used for any receiver in the Earth to self determine its position (XU; XU, 2016).

For instance, considering a system where the satellites are frozen in space in a given instant, the satellite coordinates ϱ^s relative to the center of the Earth are known, and a receiver, defined by its distance to the center of the Earth ϱ_R , is in the Earth's surface, as shown in Figure 3.

Thus, the real distance between the satellite and the receiver, ϱ , can be calculated by the time required to a signal, sent from the satellite, to reach the receiver, considering the receiver with a clock set precisely to the system time and the signal containing the exact time that it was sent. Through this approach, in theory, only three different satellite signals are needed for the receiver to fully determine its position, in terms of longitude, latitude, and height.

However, due to the high cost of having a precise clock in each receiver on the earth, in practice, GPS systems use four satellite signals and an inexpensive crystal clock, which is set almost for the GPS time. The clock introduces an offset in the measurements, which is fixed by the signal received from the fourth satellite (KAPLAN; HEGARTY, 2017).

Although GPS systems are largely used for outdoor positioning, due to the worldwide coverage area, it is shown by (FARID; NORDIN; IS-

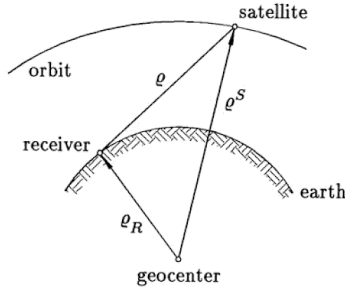


Figure 3 – Basic concept of satellite positioning.

Source: Adapted from (HOFMANN-WELLENHOF; LICHTENEGGER; COLLINS, 2012).

MAIL, 2013) that the accuracy of this system is between 6 to 10 meters, which is not suitable for precise localization under sub-meter accuracy. Also, according to the authors, the electromagnetic waves sent from the satellite are spread and attenuated by the obstacles in the line of sight of the receiver, such as buildings, and outdoor obstacles, what makes the signal too weak to come across walls, resulting in poor indoor measurements.

2.2 INERTIAL MEASUREMENT UNITS

An inertial measurement unit (IMU) is usually a micro-electro-mechanical system (MEMS), composed by a three-axis accelerometer, a three-axis gyroscope, a three-axis magnetometer, a temperature sensor, an A/D converter and a microcontroller or a digital signal processor. This integrated circuit is present in most of the mobile equipment nowadays, what makes this a cheap solution for orientation measurements. Furthermore, the data provided by these units can be used for determining the position of the equipment in which it is connected (PERTTULA et al., 2014).

However, regarding positioning measurements, systems based in IMUs must handle some obstacles to achieve an acceptable accuracy. In (GLANZER et al., 2009) a method for indoor positioning is presented, where the values obtained from the three-axis accelerometer are double integrated over time to calculate the distances (in "x", "y" and "z", axis) traveled by the device from a known starting point.

Even though, this approach has a huge coverage area, as in theory the system must only know the last starting point. In practice, a small error in measurement will induce large errors in positioning, due to the integration, that will be propagating in each position calculation. To overcome this difficulties, the authors show a technique using the Kalman filter and previous data of the building, to be able to achieve 5% of position error in a giving building. Unfortunately, the solution presented can not be used for all applications, which makes the system specific for each building, increasing the costs of development.

2.3 RADIO FREQUENCY IDENTIFICATION DEVICE

Typically, a system based on radio frequency identification devices (RFID) is composed by a tag, a reader and a host computer running the software or infrastructure. These systems can be classified in passive and active. In passive systems, the reader sends a signal which powers the RFID passive tag and waits for the response, that can be a signal sent from the tag or just a reflection of the wave sent from the reader. Then, the reader receives an ID and the data from the tag, and send this information to be processed in a host computer, which processes the data, and performs an action or sends some information back to the reader to be written in the tag. Usually, passive RFIDs are used in short range communications, due to the power needed to be transmitted from the reader to power the tags (HUANG et al., 2015).

On the other hand, an active RFID system is composed of the same structure, against the tag, which has a built-in battery to perform communication with the reader, allowing it to work in longer ranges (HUANG et al., 2015).

Regarding positioning, RFID is commonly used to identify objects in limited communication range, between one to two meters in the passive mode, due to the low cost of implementation of tags, that do not need a battery to communicate. As one can infer, this short range is not suitable for an indoor positioning system, however, this approach can be integrated with other technologies to give a more precise position of the target (SCHERHÄUFL; PICHLER; STELZER, 2015).

Differently from the passive mode, active tags can communicate in long ranges, up to 300 m, as they can use a battery to send the data. But, they still have problems in the localization application, mainly indoor, as the communication must be applied through a pre-defined protocol, the response time is high, and the presence of obstacles

introduces the multipath effect in the communication, which makes the system inaccurate for large areas measurements (BAI et al., 2012).

2.4 ZIGBEE

ZigBee is a standard wireless protocol developed by the ZigBee Alliance, which is an association of companies working together to develop more reliable, low power, cost effective wireless network standards (ALLIANCE, 2005). The protocol is built upon the IEEE 802.15.4 standard, which defines the MAC and the physical layer for a more efficient low rate wireless network, while the ZigBee standard defines the higher levels of the stack and the application.

In (SUGANO et al., 2006) a localization system based in a ZigBee network is presented, the authors used the received signal strength indicator (RSSI) of the receiver, as the starting data for the self-localization calculation. As stated by the authors, the algorithm based in the received signal strength (RSS) measurements must handle problems caused by the effects of fading and shadowing, what results in large variations in the RSSI, and can be overcome by collecting more data to achieve higher accuracy. However, this solution increases the traffic and the energy consumption of the sensors in the network.

In contrast to the problems presented in RSSI measurements, this technique is one of the simplest and most used for localization calculation. From (KUMAR; REDDY; VARMA, 2009) a method to calculate the distance through the RSSI value is shown, where the authors define a relation between the received RSSI and the distance using a simple path-loss propagation model. So, starting from the known Friss Equation:

$$P_{Rx} = P_{Tx} * G_{Tx} * G_{Rx} * \left(\frac{\lambda}{4 * \pi * D}\right)^2 \quad (2.1)$$

Where, " P_{Rx} " is the power in the receiver, " P_{Tx} " is the power in the transmitter, " G_{Rx} " is the receiver's antenna gain, " G_{Tx} " is the transmitter's antenna gain, " λ " is the signal wavelength, and "D" is the distance between the transmitter and the receiver. The following relation is achieved:

$$RSSI = -10n * \log(D) + A \quad (2.2)$$

where "n" is the path loss exponent, which can vary from 2 in free space to 4 in indoor environments, and "A" is the RSSI value at

a reference distance from the receiver as stated in (KUMAR; REDDY; VARMA, 2009). Thus, at least three transmitters placed in known reference points are needed for the receiver to be able to localize itself using the trilateration method which is presented in section 2.9.1. Unfortunately the system can be interfered by other signals in the same communication band.

2.5 BLUETOOTH

Nowadays, the use of mobile devices with Bluetooth capabilities is widespread, mainly in the mobile phones field, where most part of the devices are able to communicate over this protocol. Bluetooth is a protocol based in the IEEE 802.15.1 standard, that specifies the physical and MAC layers for the connection of devices (ALARIFI et al., 2016; IGLESIAS; BARRAL; ESCUDERO, 2012).

Similarly to ZigBee, localization systems based in Bluetooth use RSS measurements to localize themselves, as this systems are less complex. One technique frequently used in this filed is called Fingerprinting, where values of RSSI are collected in an offset state, before the deployment of the system, and a mapping of the environment is done. Then, this known values are compared with the measurements in real-time, and a probabilistic analysis is realized to determine the position of the target (IGLESIAS; BARRAL; ESCUDERO, 2012).

2.6 WIFI

WiFi is a protocol defined by the IEEE 802.11 standard, which specifies the Medium Access Control (MAC) and the Physical Layer (PHY) for wireless local area networks (WLAN). Initially it was created to provide network capabilities for the internet access in local areas (IEEE Std 802.11-2016, 2016).

With passing the time, the system became even more popular, and it is present in most of the mobile devices in the current market, what makes WiFi networks a good alternative for indoor localization, due to the low cost of implementation, and the presence of access points in a large number of buildings (ZAFARI; GKELIAS; LEUNG, 2017).

However, the use of the Industrial, Scientific, and Medical (ISM) band, and the carrier frequency, usually 2.4 Ghz, can be an obstacle for the accuracy in localization using these systems. As it is shared

with other technologies, such as Bluetooth and ZigBee, the interference in this band can be uncontrollable. Also, the existing networks were developed for communication purposes, which means that they are optimized for maximizing the coverage area and the data throughput, but not for localization. So in order to achieve higher accuracy, the systems must have an increase in hardware or complex algorithms (FARID; NORDIN; ISMAIL, 2013).

In (KOTARU et al., 2015) an interesting approach is described, where the authors developed an algorithm, called "SpotFi", which combines three different strategies to estimate the target localization with an accuracy of 40 cm. The basic idea of the algorithm consists in the utilization of the data already available in any WiFi system with at least two access points (AP). First of all, the algorithm measures the time of flight (ToF), which is presented in section 2.9.2, and the angle of arrival (AoA) from the target to each AP.

The AoA technique requires the use of at least one array of three antennas in each AP, to be able to estimate the incident angle of the signal coming from the target. Thus, knowing the distance between antennas and the difference of phase that the same signal introduces to achieve each antenna, it is possible to estimate the incident angle of arrival (YU et al., 2016).

Finally, the algorithm identifies the first package to arrive in the AP, which is the direct path that did not undergo any reflection, then it gets the RSSI measurements and uses this values combined with the AoA calculation to estimate the target's position (KOTARU et al., 2015).

Unfortunately, this approach introduces a higher complexity in the system, and increase the cost, as well as it is also exposed to interference. Furthermore, as mentioned by (FARID; NORDIN; ISMAIL, 2013), localization systems based in WiFi are always under the influence of interference, changes in signal strength and multi-path. Even though the SpotFi could achieve very good accuracy results, it is not the most accurate solution considering its complexity.

2.7 ULTRA WIDEBAND

Ultra wideband (UWB) are radio systems with relative bandwidth larger than 20% or absolute bandwidth greater than 500 Mhz. Firstly, it was developed for military applications with the aim to obtain accurate radars and low probability of interception communication. However, the first UWB system started to be commercialized only in

late 1990s (GHAVAMI; MICHAEL; KOHNO, 2007). After the creation of the IEEE 802.15.3a standardization, which allowed the use of UWB systems in short range personal areas networks, an increase in the number of emerging applications was noticeable (GEZICI et al., 2005).

This increase in interest can be explained by the fact that UWB is more suitable for indoor localization than other technologies. Due to its wide bandwidth, which allows the signal to easily come across obstacles without any interference, this systems can also overpass the multipath effect, and achieve better results in accuracy. In addition, the same reason allows the system to work with a high data rate, low power consumption, and avoid interference from other communication devices (ALARIFI et al., 2016).

The reason for UWB to overpass the multipath effect is the short pulse width, with which the system works. This can be explained from equation 2.6. Considering the velocity of the electromagnetic wave as approximately $3 * 10^8$ m/s, which is the light velocity in free space, and an UWB pulse with 0.1 ns of duration. The distance for the first reflection arriving in the receiver, to do not overlap the data, will be 3 cm, what makes the system able to filter paths that are not in the LOS of the receiver (GHAVAMI; MICHAEL; KOHNO, 2007).

Also, due to the large bandwidth, these pulses are suitable to come across obstacles because of its low frequencies components, what makes the system able to identify the first arriving path, which is the direct path in the communication link and allows more precise estimation of the distance between transmitter and receiver (SHI; MING, 2016).

2.8 COMPARISON BETWEEN TECHNOLOGIES

Intending to have a solution which fits better the desired application, a comparison between the technologies is provided in Table 1. The fundamental reason for the use of UWB system in this work, is its capability to overcome the multipath effect and the high accuracy that the system achieves. Even though some disadvantages are present, like the moderate power consumption, in this application a power supply is available in the robot's structure, which eliminates the necessity of batteries.

For the sake of completeness, it is important to note that some localization systems not based in RF or inertial movements are not covered in this work. Systems such as ultrasound, infrared and image

based technologies suffer from issues in noisy environments. The presence of sound, sunlight or fluorescent light noise sources, as in the case of industrial environment, have a high impact in the accuracy of these systems. Also, the coverage area is a key point in this work, which is limited in some of these technologies.

Table 1 – Comparison between technologies.

Technology	Advantages	Disadvantages
GPS	Worldwide coverage.	Accuracy greater than 5 meters; not suitable for indoor measurements.
IMU	Huge coverage area based on one known point.	The system must be recalibrated after some time due to the increasing error in the integration of each measurement.
RFID	System can be made with low power; wide coverage range.	The system is highly affected by the multipath effect and the accuracy is low.
ZigBee	Low cost.	The use of shared frequency bands can introduce fails in the communication; the RSSI measurements are affected by obstruction in the radio's LOS.
Bluetooth	Low cost due to the availability in most of the mobile devices.	Systems based in Fingerprinting need a recalculation of the pre-defined map of RSS values in case of change in environment.
WiFi	Large number of available systems already placed.	These systems need changes in hardware or a complex software to be able to perform localization; loses accuracy with multipath effect, changes in the RSS and interference from other systems due to the use of 2.4 Ghz spectrum.
UWB	High accuracy; do not interfere with other RF systems; has the best performance against multipath effect and NLOS situations.	High cost; moderate power consumption; in some systems clock synchronization is needed.

2.9 TECHNIQUES AND ALGORITHMS

2.9.1 Trilateration

The trilateration based positioning algorithm can be used in almost all the positioning systems not based on angle estimation. This algorithm uses at least three fixed points, also called anchors, to estimate the distance to the moving target, and it is based on geometrical properties of spheres. The fixed points are the center of the spheres, and they can be considered as transmitter omnidirectional antennas, transmitting a signal in all directions to the target. The minimum distance from the target to the anchors will be the radius of the spheres. So knowing the distance from the target to the fixed points it is possible to fully determine its position in a 3D localization (LOY; VERBEECK; KNAPEN, 2018).

Thus, in a general case, the equation, for a sphere centered in (x_n, y_n, z_n) , will be:

$$(x - x_n)^2 + (y - y_n)^2 + (z - z_n)^2 = R_n^2 \quad (2.3)$$

Where " R_n " is the radius of the sphere, and "n" is the anchor number. However, for simplicity, considering the same method presented in (COTERA et al., 2016), where all the anchors are positioned in the same height, with " $z_n = 0$ ", one of them is placed in the origin of a x-y plane, and the another one along the x-axis. The spheres can be simplified for circumferences, as shown, adapted from the authors, in Figure 4.

In this case the equations can be written as:

$$\begin{aligned} R_1^2 &= x^2 + y^2 + z^2 \\ R_2^2 &= (x - x_2)^2 + y^2 + z^2 \\ R_3^2 &= (x - x_3)^2 + (y - y_3)^2 + z^2 \end{aligned} \quad (2.4)$$

Hence, the target position can be calculated by:

$$\begin{aligned} x &= \frac{R_1^2 - R_2^2 + x_2^2}{2x_2} \\ y &= \frac{R_1^2 - R_3^2 + x_3^2 + y_3^2 - (2x_3x)}{2y_3} \\ z &= \sqrt{R_1^2 - x^2 - y^2} \end{aligned} \quad (2.5)$$

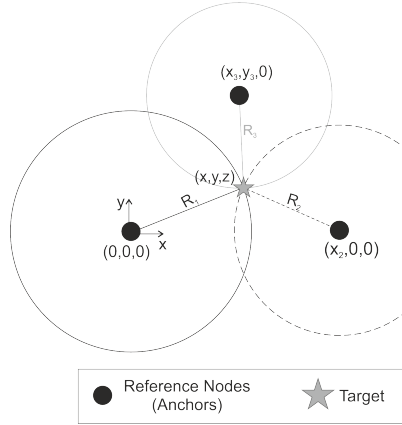


Figure 4 – Trilateration simplified example.

Even though this simplification can result in negative values in the root during the z calculation, it is still suitable for a 2D localization, when the height of the target is known and unchanged (COTERA et al., 2016; SHIT et al., 2018).

Due to the simplicity of this algorithm, and since the application in this project will move just in a 2D plane, it was chosen to be used in this work. Furthermore, this algorithm allows the use of a wide range of different techniques and let enormous flexibility in the choice of the technology to measure the distance from the anchors to the target.

2.9.2 Time of Arrival

One widely used technique to estimate the distance between radios in a localization system is called Time of Arrival (ToA), also known as Time of Flight (ToF). It uses the traveling time of one path from the transmitter to the receiver to calculate the distance, since distance can be obtained from equation 2.6.

$$D = v * t \quad (2.6)$$

Where "D" is the distance between the radios, "t" is the time that the electromagnetic wave travels, and "v" is the velocity of the wave, typically the light velocity for the free space. Then considering the velocity constant, the distance can be found just measuring

the time that the signal travels. In Figure 5, an example of a typical ToA is shown, where a path containing a reference time is sent from the transmitter, and the time of traveling is calculated by the receiver. The calculation is done subtracting the exact time that the signal arrived from the reference time, received inside the package of data (LOY; VERBEECK; KNAPEN, 2018).

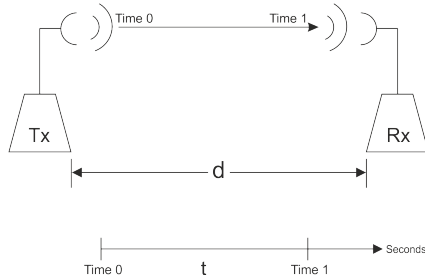


Figure 5 – Typical ToA example.

Even though, in theory, this technique is simple to be implemented, it requires a synchronized clock in both sides to be able to calculate precisely the time, what is difficult to achieve in practice and increases the cost of the system.

2.9.2.1 Two Way Ranging

To overcome the synchronization problem presented in section 2.9.2, the Two Way Ranging (TWR) approach makes the measurement in two steps. First, the signal is sent from the transmitter in a known time, then it is received and transmitted back from the receiver. Finally, the transmitter uses the time of arrival of the package and calculates the distance from the receiver, considering the double time that the package has traveled (KOLAKOWSKI; DJAJA-JOSKO, 2016). Figure 6 gives a better idea about the distance measurement, where "t" is the data traveling time, and " T_{proc} " is the known time which the receiver takes to process the data received and send it back.

So the total time " T_{total} " is calculated by:

$$T_{total} = 2 * t + T_{proc} \quad (2.7)$$

Hence, the time required for the signal to travel, will be:

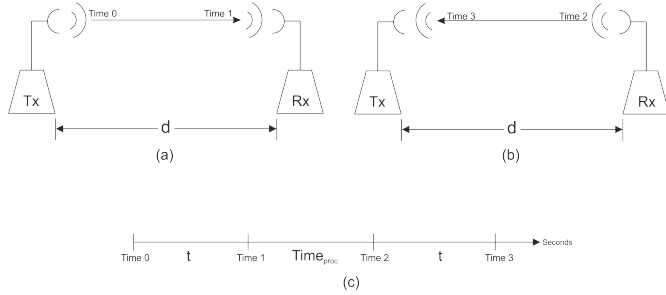


Figure 6 – TWR explanation: (a) transmitter sends the data; (b) receiver process and sends the data back; (c) time-line of the system.

$$t = \frac{T_{total} - T_{proc}}{2} \quad (2.8)$$

Thus, considering the total time to send and receive the package, the transmitter can calculate the distance from a fixed point, through equation 2.6, and using the trilateration algorithm, it can self-localize eliminating the necessity of clock synchronization (KOLAKOWSKI; DJAJA-JOSKO, 2016).

3 RELATED WORK

Cable-driven robots are nowadays widely used in the industrial field, due to its high speed, heavy payloads capability, simple mechanical structure, and huge workspace. However, for useful applicability, they must be precisely localized and controlled. One way to localize robots in industrial environments is based in UWB systems. Due to the high accuracy and multipath overpass potential, many studies and companies have been developing these systems to perform indoor positioning.

In (GUO et al., 2016) an UWB system based in TWR ToA is presented to localize autonomous robots. In this case, a quadricopter navigation method is described, where the authors integrated the autonomous quadricopter with the UWB development platform, P410 RCM, from Time Domain. Thus, they were able to achieve an accuracy better than 0.2 meters in an indoor laboratory with a total area of 7 x 7 meters. Even though this system can achieve high accuracy in real measurements, the price of the used platform still has been a disadvantage, since the basic kit costs about 9,995 USD.

The use of different techniques, for example time difference of arrival (TDOA), also allows better results in accuracy. In this technique, different transmitters send a signal at the same time, so the receiver calculates the delay between the signals arriving and uses this information to estimate the position. The authors in (OUMAR; SATTAR; TOKHI, 2018) described an implementation of the UWB system in an autonomous robot, where the platform P410 is again utilized. In this case, the coverage area is 15 x 8 meters, and the accuracy achieved is 5,5 cm through the TDOA algorithm. However, in this system, the position estimation is done from a computer, and time synchronization is needed, also the same problem of high cost shall be considered.

Regarding the cost efficiency, a different system is used in (MAI et al., 2018), an UWB system was used to track a blimp inside a stadium. In this case, the system used was developed by the Pozyx company and could achieve 20 mm of standard deviation in a clean LOS. On the other hand, during the implementation in the real structure of the blimp, the results have shown an accuracy of 80 cm for an area of 50 x 50 meters. Even though the results in accuracy were considered enough for the application, they may not be suitable for all situations. However, the fact of the small standard deviation, the low cost, among others, helped to choose the same Pozyx system for the development

of this work, as detailed in section 4.1.

Some systems for humans tracking in indoor environments are also based on UWB technology. One example of these systems is presented in (XU et al., 2017). In this work, the authors developed a 2D localization system based on a Decawave DW1000 integrated circuit to perform the positioning measurements. The authors then used and extended finite impulse response filter to perform the position estimation of a tag placed on a human hat. Thus, they were able to achieve an accuracy of 54 cm in an area of 18,2 x 3,2 meters in a complete LOS situation. The drawback of this system is that the position estimation is calculated by a computer, and the accuracy is not suitable for sub decimeter applications.

One interesting algorithm is proposed by (SHI et al., 2019) to perform localization on the internet of things (IoT) area. In this approach, the UWB measurements are combined with inertial measurements, and the TWR is made twice in order to self localize the anchors and the tag. The aim of the authors is to have a system able to be deployed in a large scale area without the necessity to have the precise position of the anchors. Unfortunately, the proposed algorithm was tested only in simulation, achieving a mean error of 12,73 cm in an area of 20 x 20 x 10 meters. It is expected that in a real situation the results will be slightly different, for example, due to the presence of reflections and poor placement of the anchors.

Another work which combines data from different sources is presented by (YOON et al., 2017), where a UWB system, from Ubisense company, is fused with inertial measurements and a biomechanical model of a human body to achieve better results in accuracy. The system was tested in an area of 1,9 x 2,3 meters at an indoor laboratory, and it used a combination of TDOA and AoA measurements to estimate the position. Thus the proposed algorithm could achieve mean errors of 9,9 cm, while only the UWB measurements presented mean error of 24,9 cm. On the other hand, the position estimation was not performed locally as the algorithm was running on a computer, the small test area and the use of a biomechanical model, are some disadvantages of the system. The authors also verified many outliers in the measurements due to the presence of the human body near the communication channel.

In unmanned vehicles, approaches fusing sensors measurements are also developed. As shown by (STROHMEIER et al., 2018), a system which combines IMU data, barometric pressure data, and UWB estimation data, could achieve an accuracy of 0.19 meters in an area of ap-

proximately 3 x 4,2 meters. The UWB transceiver used in this project is the Decawave DWM1000 model. Even though the system could achieve great results in accuracy, unfortunately, there is a necessity to have a calibration of the system in the beginning of the measurements.

In relation to the presented works, this project will use a different approach to estimate the position. In this work, the positioning data will be computed during 1.8 seconds, and in the end, an average will be performed in order to have a final result in 2 seconds. Thus, it is expected that better results in accuracy can be achieved after the average computation. This approach will be used due to, in this application, the payload needs two seconds to perform the task, which allows the positioning system more time to estimate the payload's position. In addition, the use of the TWR technique, and the fusion of UWB measurements with inertial measurements will be applied aiming to avoid time synchronization and achieve better results. So table 2 can be used to position this work against the main points of the related works.

Authors	Avoid time synchronization using TWR	Tag estimates self position	Fusion of sensors	Sub decimeter accuracy
(GUO et al., 2016)	X	X	-	-
(OUNMAR; SATTAR; TOKHI, 2018)	-	-	-	X
(MAI et al., 2018)	X	X	X	-
(XU et al., 2017)	X	-	-	-
(SHI et al., 2019)	X	X	X	-
(YOON et al., 2017)	-	-	X	X
(STROHMELER et al., 2018)	X	X	X	-
Present work	X	X	X	X

Table 2 – Comparison between related works.

4 METHODOLOGY

4.1 ULTRA WIDE BAND SYSTEM FROM POZYX

For this work, the hardware chosen was the prototyping development kit from Pozyx (POZYX, 2019). The fundamental reason for the use of this system is the low cost, compared with others solutions available in the market, the large number of libraries available from the manufacturer, and the flexibility in the configuration of all parameters, which makes the system easy and fast to be implemented.

This system uses four anchors, as fixed points, and two tags to perform the localization. One of the tags is used as the "master" of the communication network, so it is responsible for controlling and configuring all the parameters of the structure. The second tag is used as the mobile tag which will be located. The manufacturer ensures that the system uses the TWR technique, the radios can communicate over 30 meters, and the operation frequency can be defined in 7 channels located between 3,5 to 6,5 Ghz, with an update rate up to 60 Hz, and an accuracy of 10 cm (POZYX, 2019). Figure 7 shows the prototyping kit used in this work, the kit is composed by four anchors, two tags, one ATMEGA328 based PCB, two USB cables, four power supply cables, and one 1200 mAh powerbank.



Figure 7 – Pozyx prototyping kit.

The manufacturer offers two libraries to program the system.

One library is based in the Python programming language, and the second one is designed to be used in interaction with an Arduino, presented in Section 4.3.

The UWB communication is controlled over an Decawave DW1000 transceiver, which allows the configuration of the UWB parameters such as: the preamble length, the bitrate, and the frequency channel. Also, this transceiver allows the wireless communication of data, between two radios, where a 100 bytes buffer is available to be accessed over the libraries. According to Pozyx, the hardware is equipped with a 9-axis inertial measurement unit, and a STM32F4 microcontroller, which can communicate over I2C or USB with a host processor (POZYX, 2019).

In addition, two algorithms to calculate the positioning are available, they are called UWB ONLY and TRACKING. These algorithms let the x and y axis data available in a 32 bits register after the calculation is performed. The only information available from the manufacturer about the difference between both, is that the TRACKING algorithm combines inertial data with UWB data to estimate the position. So in this work, both algorithms will be tested and analysis will be done aiming to choose the best one for this application.

4.2 RASPBERRY PI

Raspberry Pi is a fully featured computer which runs the Linux operating system, commonly the Raspbian distribution. The hardware, shown in Figure 8, is based in the Broadcom BCM2837B0 Cortex-A53 ARMv8 quad-core processor, which can run up to 1.4GHz, and it is equipped with a 1GB LPDDR2 SDRAM memory, four USB ports, one HDMI connector, one connector for a CSI camera, forty GPIO pins, and it is powered by an 5V and 2.5A DC power input (RASPBERRY PI FOUNDATION, 2019).

There are a considerable number of programming languages that have been adapted for Raspberry Pi, the most common are: Python, C, C++, and Java (RASPBERRY PI FOUNDATION, 2019). Python programming language is recommended by Raspberry Pi foundation, and it was chosen to be used in this project due to the integration with the Pozyx library, which has already defined all the registers and memory access addresses for the communication over the I^2C protocol.

In addition, this system was chosen to be used as the payload main processor of the robot, due to the capacity to run parallel codes,

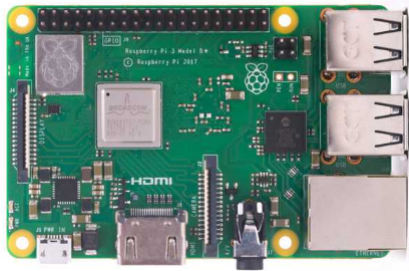


Figure 8 – Raspberry Pi 3 Model B+.
Source: (RASPERRY PI FOUNDATION, 2019).

what makes it possible to run the payload application while it calculates the self position, achieving faster results.

4.3 ARDUINO

Arduino is an open source hardware and software platform. The hardware is based on an ATmega328P microcontroller, with speed of 16 Mhz and flash memory of 32 KB. Also, 14 GPIOs are available for interaction and communication with other devices, and it operates with 5V power supply (ARDUINO, 2019).

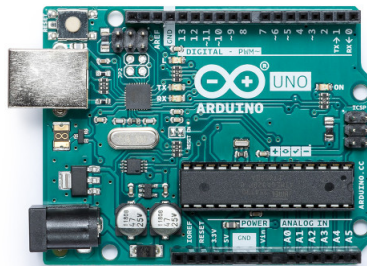


Figure 9 – Arduino Model Uno Rev3.
Source: (ARDUINO, 2019).

The software is compiled by the Arduino IDE, and it is based in the C++ programming language, with some functions and libraries

developed specifically for these boards (ARDUINO, 2019). Even though, there are many models of arduinos available on the market, this work used the model Uno Rev3, presented in Figure 9, as one board of this model comes inside the Pozyx development kit.

4.4 PROPOSED SYSTEM

The proposed system for this work uses the Pozyx kit, one Arduino and one Raspberry Pi board. The UWB anchors are placed around the robot structure, in a rectangular format. The moving part of the robot, is embedded with one Pozyx tag communicating over I^2C with the Raspberry board, thus the Raspberry can calculate the self position, using the TWR technique. Also, it can run the payload software and control the time when sending the data to the control station.

The control station is composed of one UWB master radio, which is connected to an Arduino board over one I^2C bus. The Arduino receives the data from the embedded part using the UWB wireless communication, and send the data received to be processed in a later state by a host computer over serial communication. To start the position calculation, a "start" command is sent to the embedded part over the UWB communication from the control station. The block diagram of the whole system can be seen in Figure 10.

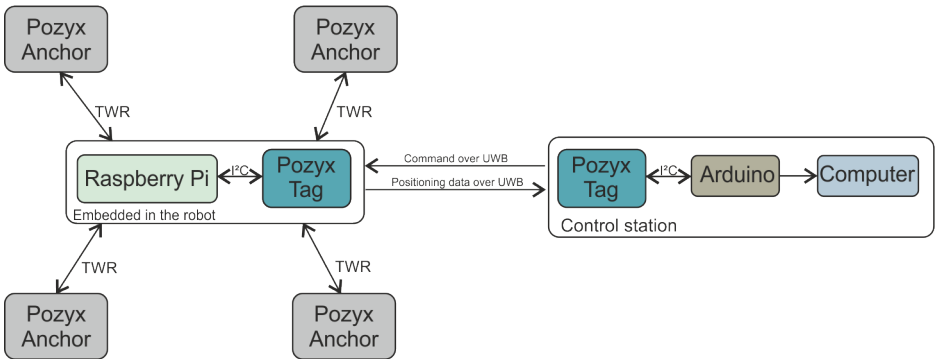


Figure 10 – Block diagram of the proposed system.

5 DEVELOPMENT

During the beginning of the project, the system was deployed and the Pozyx's tutorials, available in (POZYX, 2019) where tested in order to validate the hardware. As soon as the system was running, measurements in the position were performed. The results achieved for the accuracy where in contrast with the ones guaranteed from the manufacturer, with errors up to 40 cm. The source of this errors was observed to be the poor placement of the anchors. Thus, the pattern of radiation of the antenna had to be considered, as it radiates almost omni-directionally in the z-x plane, but does not perform so well in the y-axis showed in Figure 11 (POZYX, 2019). An antenna radiation analysis would be needed to better define the correlation of this error with the anchors positions. However, this characterization is beyond the scope of this work.

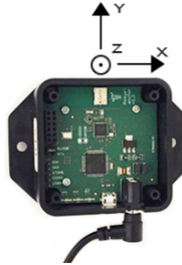


Figure 11 – Pozyx antenna radiation directions.
Source: Adapted from (POZYX, 2019).

In order to make the system flexible to be installed in different scenarios, a mechanical structure was developed to place the anchors in the best angle to have the signal covering the whole measuring area. To achieve this goal, considering the axis in Figure 11, four pieces were designed using the SolidWorks software and fabricated in a 3D printer. This allows the anchor to be moved in y-axis and x-axis, and be rotated in steps of 45° around the z-axis, as shown in Figure 12. Thus, more flexibility in changing the anchors placement can be achieved during the deployment of the system in the robot's structure.

In addition, according to the manufacturer, the anchors must be placed at least 20 cm away from any metal structure, to avoid changes

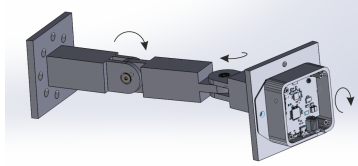


Figure 12 – Mechanical structure for anchors position.

in antenna's parameters. So, considering that the robot basis is made of aluminum profile, with this structure, the anchors can be placed in a distance of 26 cm away from the metals.

After the validation of the hardware, a test area was defined to be used for the evaluation of the system. The aim of the tests was to have a better understanding about the system functionality, comparing and defining the algorithm that would be used in the final version.

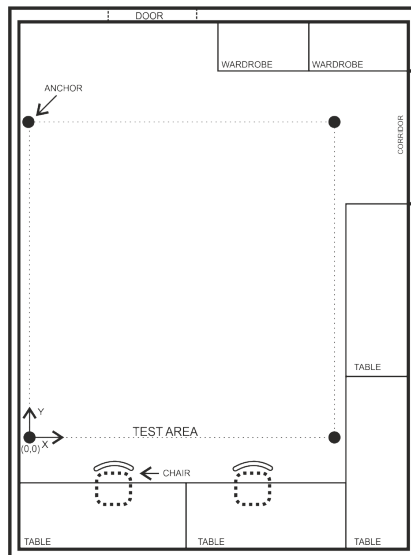


Figure 13 – Top view of the measurement area.

Therefore, a planar test area of 4 x 4 meters was adopted. The anchors were placed in a height of 9 cm from the floor, and the LOS was guaranteed in this area. However, the presence of objects and walls in the room were expected to impact the final positioning measurements.

The top view of the room is shown in Figure 13.

Then 10 fixed points were placed to be used as ground truth for the measurements. The points were millimeter accurate marked on the floor using a metric tape for the distances in relation to the defined reference point $(0,0)$. The reference points can be seen with a grid in Figure 14, where all dimensions are in millimeters.

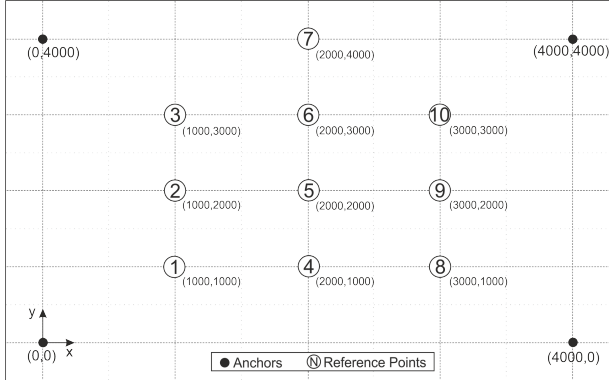


Figure 14 – Test area ground truth.

5.1 POZYX DATA BEHAVIOR

Since the area was defined, a test plan was made to evaluate the Pozyx system performance in stand alone mode. For this, the tag was placed in all points for 10 minutes each, and each algorithm was applied following the tutorials available in (POZYX, 2019). The system was turned off when the tag was moved from one point to the other. The algorithm was performing the measurements in an update rate of 25 Hz, so in 10 minutes 1500 samples were acquired in each point, and in total 15000 samples were received by each algorithm.

This experiment was realized to verify the behavior of each algorithm. It was noticed that, the TRACKING algorithm resulted in exponential decay behavior in the measurement errors during the early state of the test. Then, after some time, the measurements started to converge to the expected target.

This behavior has shown that the algorithm needed past measurements in order to estimate the new position. It was confirmed when the system have been changed from one point to the other, and

the measurements were turned on again. The initial data received from the algorithm, was the position of the past point where the system was placed, and the next measurements smoothly started to converge to the new expected value.

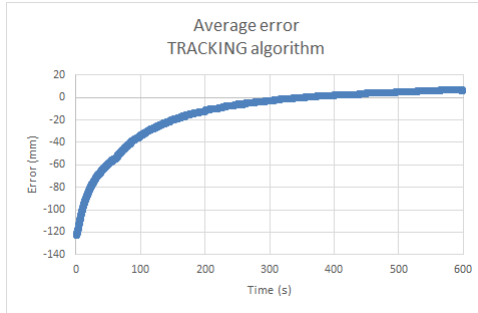


Figure 15 – TRACKING algorithm average error in x-axis measurements of 10 minutes in each point.

In Figure 15 the results of 10 minutes measurements in each point of Figure 14, using the TRACKING algorithm are shown. The exponential behavior can be observed in the beginning of the measurements, with errors up to 12 cm. However, after approximately 200 seconds, the errors started to become as low as 2 cm.

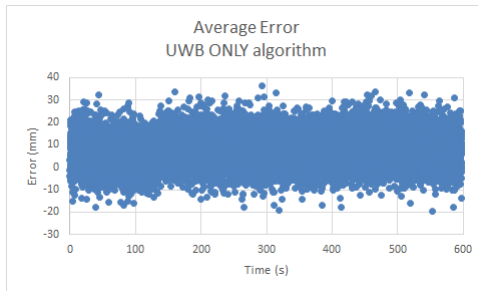


Figure 16 – UWB ONLY algorithm average error in x-axis measurements of 10 minutes in each point.

The same experiment was performed using the UWB ONLY algorithm, the result is shown in Figure 16. From the graph, one can infer that calculating the average value from the measurements in a

defined period of time, would result in smaller error than getting just one sample in this period.

Considering these results, and keeping in mind that the robot has 2 seconds to calculate the positioning, the approach used in this work will get the samples during 1.8 seconds, and calculate the average value to achieve less variation in the results.

5.2 COMMUNICATION PARAMETERS SELECTION

According to the manufacturer, the update rate of the measurements depends on the UWB communication parameters, mainly the preamble length and the bitrate. The preamble length can be considered as the header of the communication path, and the bitrate is the velocity of the communication in the network (POZYX, 2019). A trade-off is observed in the configuration of these parameters, as a higher preamble length allows the communication over longer distances, it results in a lower update rate. The same occurs with the bitrate, since the probability of error in the communication is greater with higher bitrates, the update rate is also higher.

So aiming to define the communication parameters used to test the system, an analysis was performed varying both values to identify the best configuration for the measurements area. For this, the system was placed in a fixed known point and the algorithm used was the UWB ONLY. Then, the preamble length and the bitrate were changed and, for each configuration, the measurements were performed during 2 seconds. The results of average error are presented in bars, the update rate are presented as points in the lines, and the preamble length are presented in different colors on Figure 17.

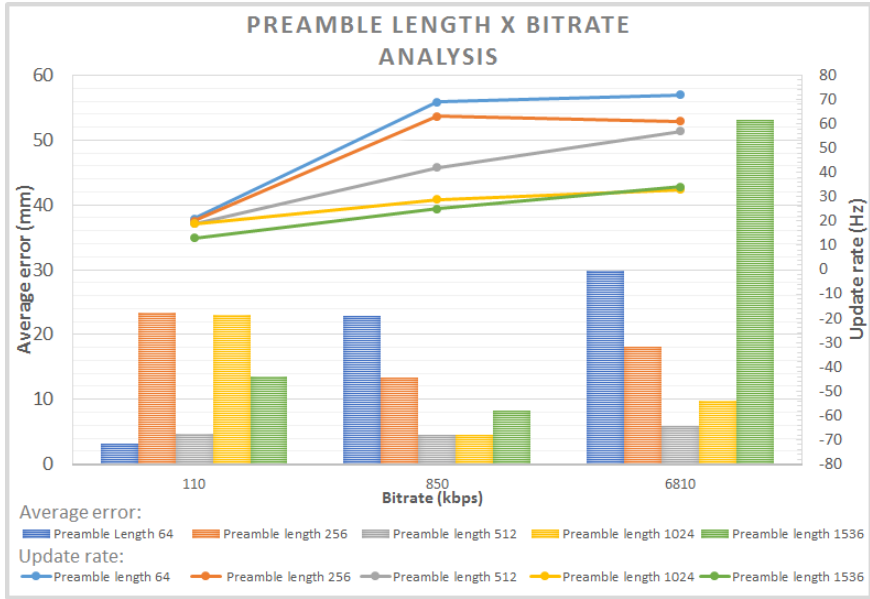


Figure 17 – UWB parameters evaluation.

For the 4 x 4 meters area, a reliable communication was obtained using a preamble length of 512, where it is possible to observe in the graph that the average error was approximately the same for all bitrates. Furthermore, from the results, it can be seen that for this preamble length, the smallest mean error was obtained with a bitrate of 850 kbps, what has resulted in an update rate of 42 Hz and the second global smallest average error of 4,48 mm. The overall smallest average error of 3,14 mm was obtained with a preamble of 64 and a bitrate of 110 kbps, however, this configuration allowed an update rate of only 21 Hz.

For this reason, it was chosen to use the preamble length of 512 and the bitrate of 850 kbps in this work, because it has shown to provide a good relation between update rate and average error. These results were presented aiming to define a reliable configuration for the evaluation of the positioning algorithms in the test area, allowing the repeatability of the tests in the same configuration. However, it is important to note that this results will change according to the area of measurement, and it is possible that better results can be found according to the placement area and the configurations of the network.

5.3 FIRMWARE

After defining the UWB parameters, two firmwares were developed for the tests. The first was developed in the C++ based language adapted for Arduino boards, and runs in the Arduino in the control station. The second one was developed in Python and runs in the Raspberry which is the embedded part in the robot. The flowchart of both firmwares can be seen in Figure 18.

Basically, the system is first configured by the Arduino algorithm, where the network parameters are set and the anchors' position are defined. Then all the settings are spread for the members of the communication channel. After this, the algorithm waits for the push button to be pressed. Then, a start command is sent, over UWB, to the embedded tag to start the position calculation. As soon as the command is sent, the control algorithm waits for an interruption to be generated from the Pozyx's master tag, indicating that the new data is available to be read over I^2C .

On the other hand, the embedded code begins by configuring the I^2C communication with the Pozyx's tag. Then it waits for an interruption to be generated indicating that the start command has arrived. So the algorithm starts the measurement calling the "DO POSITIONING" function, which is available from the manufacturer as a memory address of the tag. After the positioning calculation is done, a GPIO pin of the Raspberry is set down from the Pozyx tag, indicating that new data are available. Thus, the data are read from Raspberry as x and y-axis. Each axis is available in a 32 bits register in the tag's memory. This process is repeated during 1.8 seconds. When the timer is over, the Raspberry computes the average of all measurements and send the data over UWB to the master tag, which generates an interruption in an Arduino's GPIO pin.

These firmwares were used to test both algorithms, the UWB ONLY and the TRACKING. The only difference during the tests is that the Raspberry code, in the final version, sends only the average value to the master tag. However, during the evaluation of the algorithms all the position data available has been sent to the master tag in order to plot the graphs and analyze the behavior of each algorithm.

Furthermore, a third algorithm was developed and tested. It combines the measurements of UWB ONLY and TRACKING aiming to achieve a more accurate value when calculating the average. From now, this algorithm is called "Proposed" and it was developed having in mind that the TRACKING algorithm uses previous values to estimate

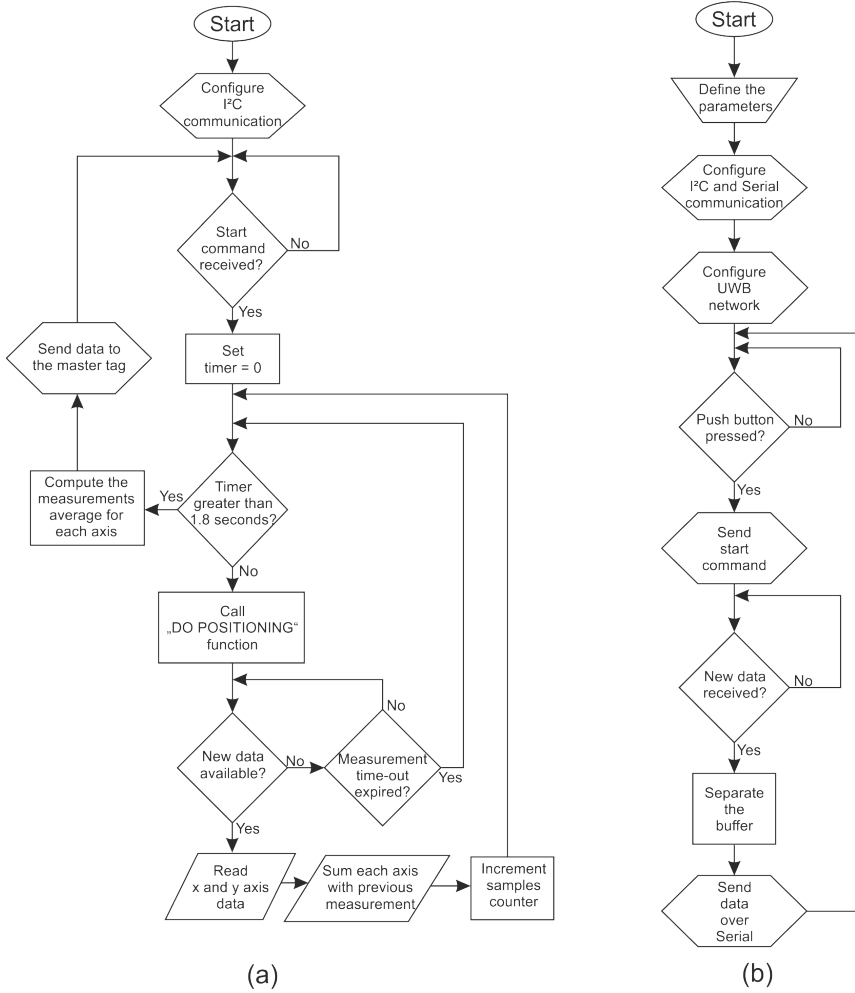
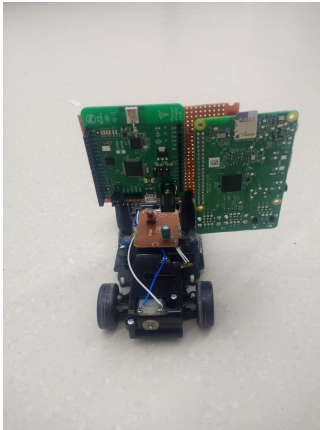


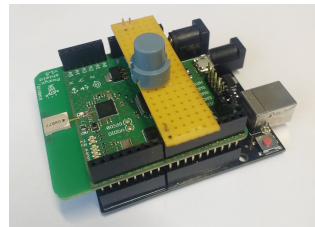
Figure 18 – Firmwares Flowchart. (a) Flowchart of the Raspberry’s firmware; (b) Flowchart of the Arduino’s firmware.

the position. So, intending to let the previous values closer to the expected value, the proposed algorithm starts the measurements using the UWB ONLY and changes to the TRACKING algorithm. Hence, it is expected to eliminate the exponential response shown in Figure 15 , or at least make it smoother.

After the firmwares were developed, a prototype of the embedded part was assembled to be used during the tests. The intention was to connect the Raspberry with the Pozyx tag and move the boards around the tests area. Thus, knowing that the TRACKING algorithm uses inertial measurements to perform the positioning calculation, the system should be moved in a way that it does not interfere in the height of the tag. As the system was measuring in a two-dimensional area, interference in the results due to height changes should be avoided. Therefore, the system was placed on a remote-controlled car, which allowed the movement around the complete area, maintaining always the same height in the tag, as shown in Figure 19a. The prototype of the control station with the push button to send the measurement command is shown in Figure 19b.



(a) Embedded system prototype.



(b) Control station prototype.

Figure 19 – Prototype of the system used in the experiments.

Both boards were connected over I^2C . Also, one GPIO pin was connected to inform the Raspberry when data is available to be read from the tag. A 5 V and 20000 mAh battery pack was used to power the embedded systems during the tests to allow the movements around the area. However, in the real operation, inside the robot structure, two

cables connected to a power supply of 24 V and 240W are available in the movable part to power the embedded board. So a power converter of 24 V to 5 V must be developed to allow the embedded system to be powered.

After the prototype was developed, a test plan had to be defined to evaluate the behavior of the system. Then, two main tests were designed to characterize the algorithms moving the robot around the test area. During the first test, the robot was moved in a square shape, stopping in each vertex along two seconds in order to perform measurements. In this experiment, instead of taking the measurements average, the Raspberry was sending all the measurement data to the control station, aiming to have all the samples for data analysis, in an off state, in the computer. This pattern was repeated 10 times, with each positioning algorithm, to verify the repeatability of the measurements in the same point after a movement has been performed. The measured points for the first test are shown in Figure 20, and the results are shown in Section 6.

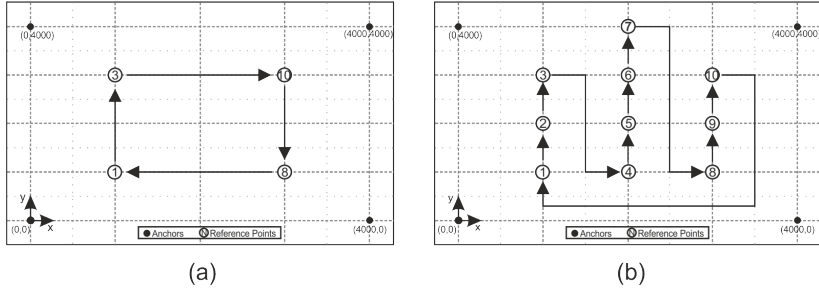
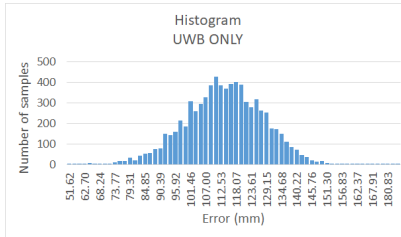


Figure 20 – Test plan. (a) First test plan; (b) Second test plan.

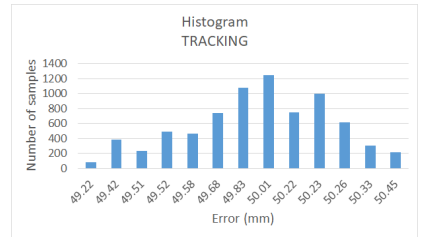
In the second experiment, the system was moved around all the ground truth points. In each point it was stopped, the calculation was performed and only the average value was sent to the control station. The tests were repeated 5 times with each algorithm. The goal of this experiment was to verify the behavior of the final version of the firmware, and define which algorithm would be used in the final version, considering the performance in the defined points.

5.4 DATA EVALUATION

For the evaluation of the data presented in this work, a basic statistical analysis was performed. The evaluation began by verifying the behavior of the data acquired with both algorithms. For this, the system was placed in the position 2000 mm in x and y-axis, which represents the middle of the test area. The data were acquired during 5 minutes, after letting the system running for 5 minutes to avoid the TRACKING exponential decay behavior. So the histogram of the errors in the distance to the ground true point were plotted. The results with the UWB ONLY algorithm are shown in Figure 21a and the TRACKING are shown in Figure 21b.



(a) UWB ONLY algorithm histogram.



(b) TRACKING algorithm histogram.

Figure 21 – Histograms of the errors of the algorithms.

From these graphs, it can be observed that the data seems to follow a normal distribution. For this assumption, the most basic D’Agostino test of normality was performed to verify the symmetry of the data around the mean value. In this test, the symmetry is compared with a standard normal distribution, and if the result is less than 2, it is a good indicative that the data follows this pattern (DAS; IMON, 2016). For these data, the UWB ONLY resulted in a coefficient of 0.03 and the TRACKING resulted in 0.27, so they were considered a normal distribution. The formal prove that these data follows a normal distribution is beyond the scope of this work.

The normal distribution behavior shows that the samples acquired from the algorithms are random variables, "X", and basic probabilistic analysis for normal distribution can be used in these data. So the results are evaluated according the mean " μ ", also called average in this context, and the standard deviation " σ " of the errors, which

can be calculated by equations 5.1 and 5.2, respectively (SMITH et al., 1997).

$$\mu = \frac{\sum x}{n} \quad (5.1)$$

$$\sigma = \sqrt{\frac{\sum (x - \mu)^2}{n - 1}} \quad (5.2)$$

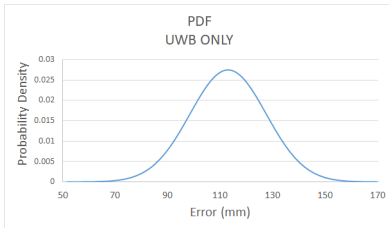
In these equations, "x" is the value of the random variable, and "n" is the number of samples. However, only the mean value and the standard deviation are not enough to achieve a conclusion about the performance of the algorithms, so a probabilistic analysis is performed through the probability density function (PDF) and the cumulative distribution function (CDF).

The PDF describes the behavior of the probability of the samples in a data-set. For a normal distribution, the PDF is given by equation 5.3.

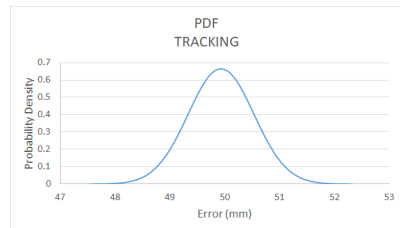
$$f(x) = \frac{1}{\sqrt{2 * \pi * \sigma^2}} * e^{-\frac{(x - \mu)^2}{2 * \sigma^2}} \quad (5.3)$$

Graphically, the area under the PDF represents the probability of the sample to be inside a given interval. For instance, the probability of a given random variable to be between an interval of "a" and "b" can be calculated by equation 5.4 (SMITH et al., 1997). For the data presented in Figure 21, the PDF graph is presented in Figure 22.

$$P\{a \leq X \leq b\} = \int_a^b f(x)dx \quad (5.4)$$



(a) UWB ONLY algorithm PDF.



(b) TRACKING algorithm PDF.

Figure 22 – Probability density functions of the errors of the algorithms.

The CDF represents the probability of the random variable "X" to achieve a value equal or less than "a". It is calculated by the area of the PDF graph from minus infinity until the desired value "a", and can be defined by equation 5.5.

$$F_X(a) = P\{X \in (-\infty, a]\} = \int_{-\infty}^a f(x)dx \quad (5.5)$$

The aim of this project is not to classify statistically the behavior of the algorithms, but choose one to be used in the desired application. For this, the evaluation of the CDF was defined to be used, as it shows the probability of the errors of each algorithm to be below a certain value, which in this context means a greater accuracy for the positioning estimation.

5.5 POWER SUPPLY

Finally, a power supply should be developed to power the embedded system. Inside the robot, 24 V and 240 W power cables are available to be used. However, the Raspberry and the Pozyx tag work with a voltage level of 5 V. So a power converter should be used in this project to allow the system to be powered directly inside the robot. It is important to note that the Poxyz anchors will be fixed in the robot structure and will use an off-the-shelf 5 V power supply, which came inside the Pozyx's kit, connected directly with the power outlet. The same is true for the master tag, which will be connected over a USB cable with the computer for serial communication, powering the system.

Table 3 – PCB Power consumption estimation.

Component	Voltage	Maximum Current Consumption
Raspberry Pi	5 V	1 A
Pozyx	5 V	180 mA
Payload	5 V	250 mA

In order to define the parameters for the converter, a power consumption estimation was performed to give more reliable prevision about the power needed for the embedded PCB. In Table 3, the current consumption of each component powered by the PCB is presented. In total, a power source of 5 V and 1,43 A is needed to power the system.

However, as the table presents only an estimation of the consumption and considering that the payload is still in development, a power supply of 5 V and 2 A was designed to convert the 24 V and power the components, allowing more flexibility to the payload.

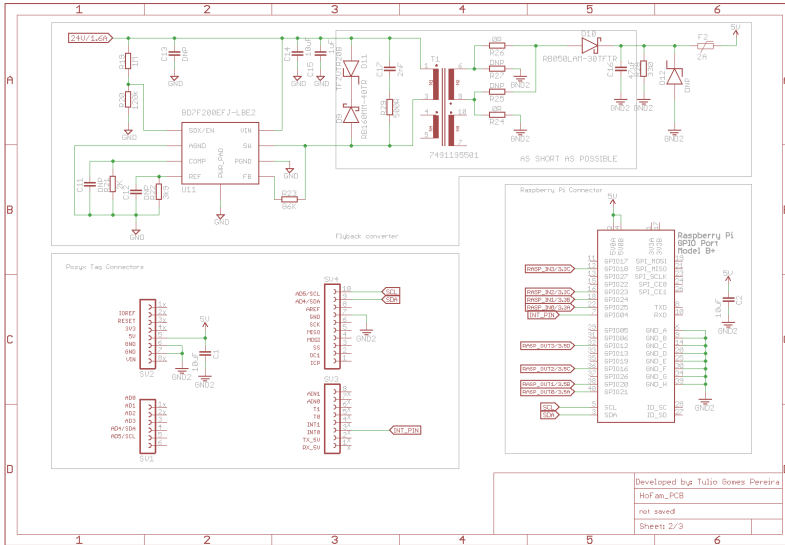


Figure 23 – Schematic of the flyback converter used to power the embedded system in the robot with 24V.

Thus, the flyback converter topology was adopted. This topology allows the conversion of the voltages isolating completely the input from the output, which makes it safe and reliable for the connection of the external power supply with the embedded components. The integrated circuit chosen was the model BD7F200EFJ-LBE2 from ROHM semiconductors. This component can be supplied with voltages between 8 V to 40 V, and can be configured to allow a stable 5 V and 2 A output after a transformer, which isolates completely the system. Based in the example available in the component’s datasheet in (ROHM SEMICONDUCTOR, 2017), a PCB was designed to be used inside the robot, connecting all the modules. The schematic for the flyback and the connectors for the sub-modules is shown in Figure 23.

The PCB was designed to power the Raspberry and the Pozyx tag using the cables available inside the robot structure, these cables allow the system to be powered and also communicate over an industrial

AS-Interface protocol, and should be connected in a screwed connector. A circuit to allow the communication over this protocol was also designed to let the system more flexible, however, this communication will be implemented in the future and it will not be detailed in this work. Also, the Raspberry pins were mounted in a way that they could be accessed after the PCB was assembled, to allow a connection with the payload. The 3D model of the assembled embedded PCB without the payload is shown in Figure 24.

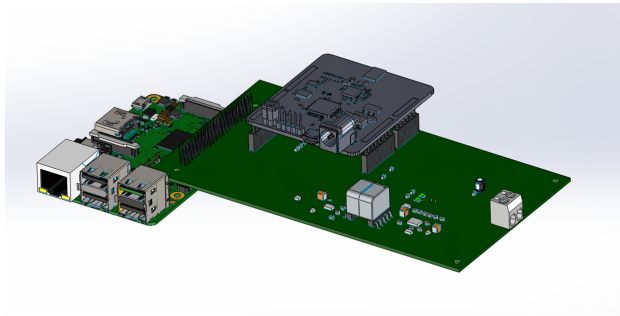


Figure 24 – Assembly in 3D of the PCB embedded in the robot.

The PCB was produced and the behavior was analyzed using a programmable power supply HMP4040, an electronic load model LD300, and an oscilloscope DPO 4054B, before the connection with the Raspberry and the Pozyx's tag to guarantee a safe connection between the boards. The electronic load was used to simulate the board's current consumption to test the converter. The system under test is shown in Figure 25, and the results will be presented in chapter 6.

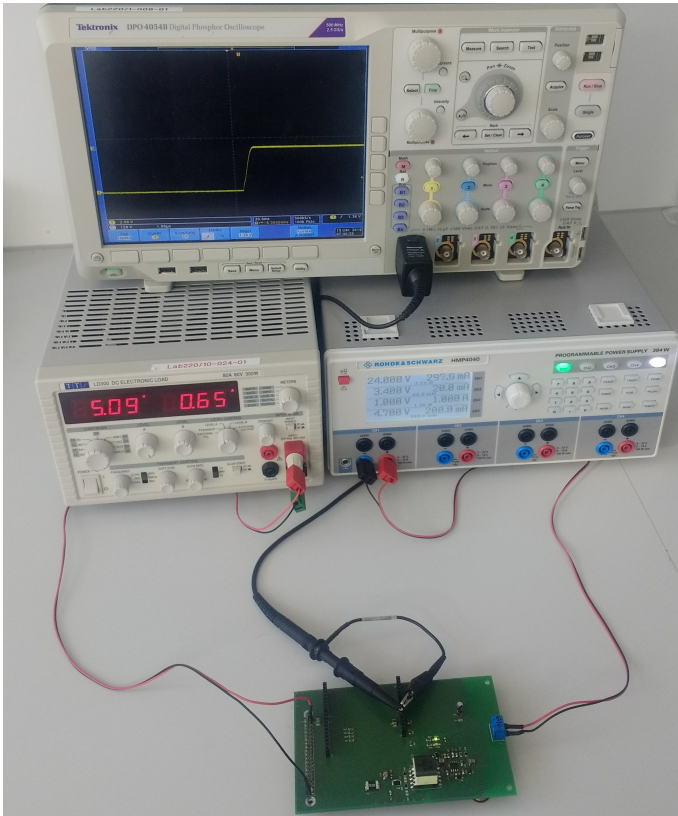


Figure 25 – PCB under evaluation in the test bench.

6 RESULTS

The metric to evaluate the measurement system under analysis in this work is the x-y plane error, as described in Figure 26. Even though they have been acquired in separated x and y values, the aim is to simplify the analysis, so the relation of errors will be given by the geometrical distance between the predicted point with the ground truth point.

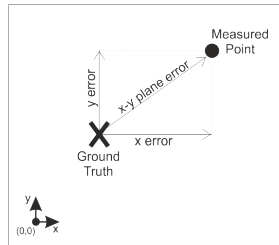


Figure 26 – Relation of the errors in the cartesian plane.

6.1 FIRST EXPERIMENT

The first experiment was conducted according to the test plan presented in Figure 20. The UWB parameters were configured to use the preamble length of 512 and the bitrate of 850 kbps. The experiment was repeated 10 times. Hence, data during 1.8 seconds from each point were received 10 times by the computer.

6.1.1 TRACKING Algorithm

For the TRACKING algorithm, to evaluate the system without the interference of the exponential response, the measurements started 3 minutes after the system was placed and turned on in the first point. Thus, the test was performed with the system running all the time, even though the system was moving from one point to the other. The only difference was that during the movement the algorithm was still running in the background and the data was only acquired after the command was received. The update rate of this algorithm during the

test was 46 Hz, so 82 samples were expected from each measurement, during 1.8 seconds.

Then, the results for the x and y-axis were calculated. They were evaluated for 10 measurements at one point. In this case, the ground truth point number 10, positioned in (3000,3000), was chosen randomly to be analyzed. The results in x-axis can be seen in Figure 27, for the y-axis they are shown in Figure 28. In these graphs, the colors represent the different measurements, and the points represent the samples acquired for each measurement in 1.8 seconds.

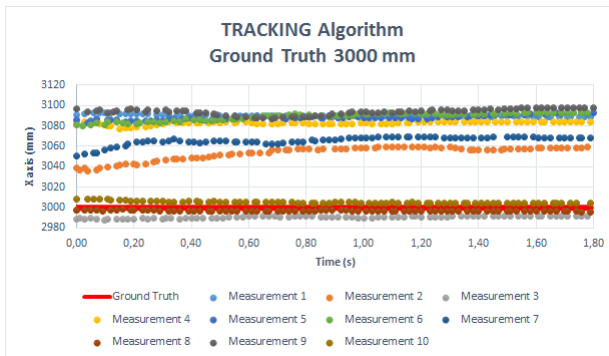


Figure 27 – TRACKING algorithm results in x-axis at point 10.
Source: From the author.

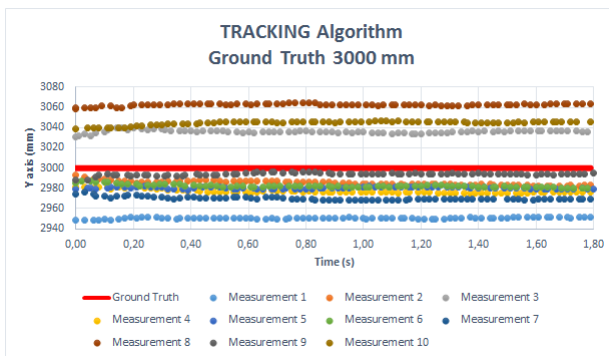


Figure 28 – TRACKING algorithm results in y-axis at point 10.

From these results, one can observe a shift from the expected value in each measurement. This shift could not be removed as it is in-

herent to the measurements. It is expected that the source of this shift can be related to errors in inertial measurements, or even in errors during UWB measurements. To avoid interference in IMU measurements the movements from one point to the other were performed following the arrows presented in the test plan.

Also from the results, it can be observed that in some measurements the number of samples is less than in others. The fundamental reason for this behavior is that the firmware in Raspberry, has a timeout during the measurements, which means that, if a sample was not acquired before the timeout, it is ignored and measured again. The occurrences can be observed as the time gap between the samples in the graph. This was done in order to avoid holding the code in case of a fault in communication, or errors due to the multipath effect. Even though the number of samples is not the same, at least more than one sample will be acquired in this period, what will be used to calculate the average.

From a separate analysis of each axis, it was verified to be difficult to compare the real error of the measurements with the ground truth. For this reason, a geometric computation was performed to have the real distance from the ground truth to the estimated point, in the x-y plane, as shown in Figure 26. Thus, this geometric distance was used as the real error in the measurements to compare the algorithms.

The error was calculated for each sample in all the measurements before the analysis. In Figure 29, the error in x-y plane is shown for all samples separated by the respective measurement, which is represented by different colors. Calculating the average of these errors it is possible to achieve an overall average error of 76,97 mm in this point. This calculation allowed a performance comparison between the algorithms.

Besides computing the geometric error, the standard deviation of 22,05 mm was also calculated, and it has been caused due to the shift observed in each measurement. However, for this algorithm, it can be seen that, if the standard deviation of the samples in each measurement are analyzed, the result is smaller, in average it is 2,42 mm.

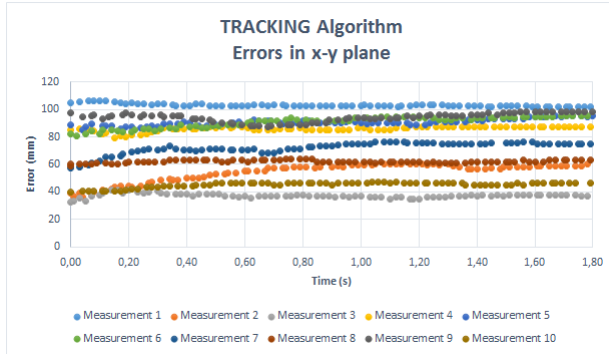


Figure 29 – TRACKING algorithm average error in x-y plane.

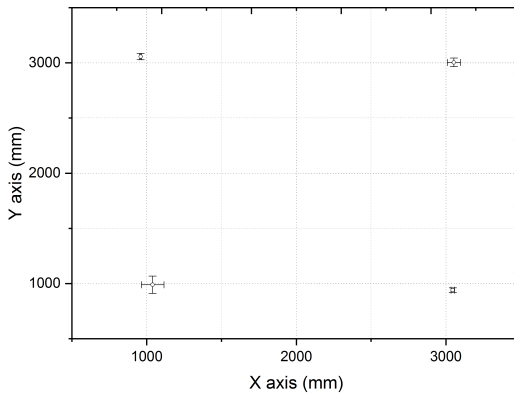


Figure 30 – TRACKING algorithm standard deviation in each point.

Finally, the mean error and the standard deviation in x and y of all samples were plotted to all the four measured points. The graph can be verified in Figure 30. The aim of this analysis was to certify the behavior of the system in all the points measured. It could be observed that in all the points there is a shift in the mean value, related to the ground truth point, which is the same behavior observed before, and can be an indication that it is inherent to the algorithm. Also, in point 1 the standard deviation was greater than in the others. One possible reason for this behavior, is the closeness of this point with the chairs and

walls in the room presented in Figure 13, unfortunately, the reflections presented in the laboratory were not possible to be measured.

The anchors were changed from the original position in clockwise, but the origin point was maintained the same, in order to verify the dependence of the results with anchors' positions. Therefore, it was expected that the point (1000,1000) would present the greatest error again. The anchors were called "A", "B", "C" and "D", so that their position in the measurement area in each experiment are presented in Figure 31, as well as the results of the experiments.

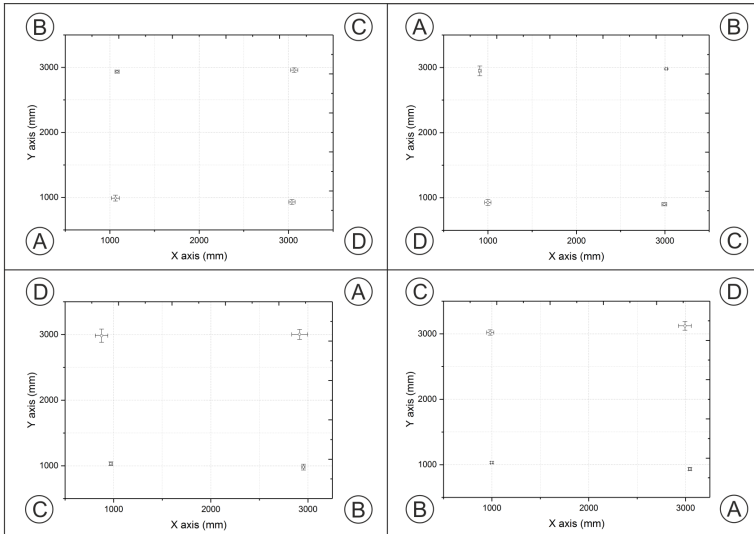


Figure 31 – Rotation of the anchors in clockwise.

The results of this experiment have shown that there is a relation between the error and the anchors' position. It could be verified that placing the anchors in different positions, results in different standard deviation for each measured point. Thus, this behavior has shown that not only reflections are acting in this test area, but also problems such as antennas' gain direction, and differences in the processing time of the anchors can be degrading the position estimation. Even though these problems have been observed, extensive tests should be performed to define the exact source of the errors in this experiment area. However, the aim of this project is not to characterize the errors in this system, but define an algorithm to be used in the application presented. So the

anchors were placed back to the original position, and the other two algorithms were tested under the same conditions as the first one.

6.1.2 UWB ONLY Algorithm

For the UWB ONLY algorithm the tests could be started as soon as the system was turned on, because no pattern in the behavior over the time was observed. Therefore, the experiment was conducted in the same way used in the past algorithm, and the result for the x-axis is shown in Figure 32. For the y-axis, the result is presented in Figure 33.

For this algorithm, using the configuration explained before, the update rate achieved was the same as the TRACKING algorithm.

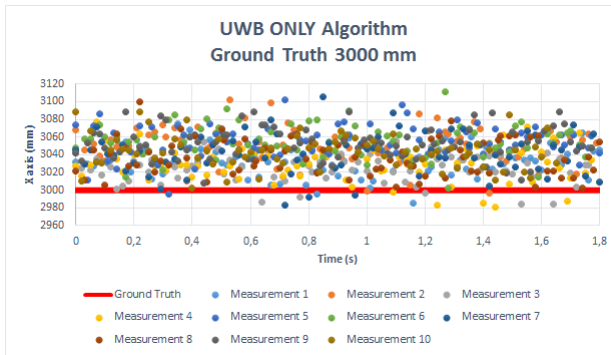


Figure 32 – UWB ONLY algorithm results in x-axis at point 10.

From the results it can be observed that the variation of the samples around the center value in each measurement, is greater for this algorithm than for the last one.

To compare the results with other algorithms, the error in the x-y plane was computed, in the same manner as described in section 6.1.1, and it is presented in Figure 34.

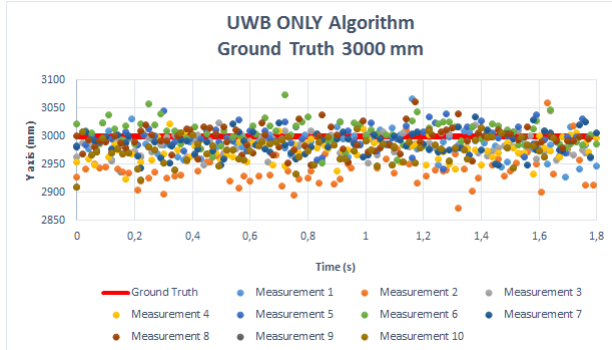


Figure 33 – UWB ONLY algorithm results in y-axis at point 10.

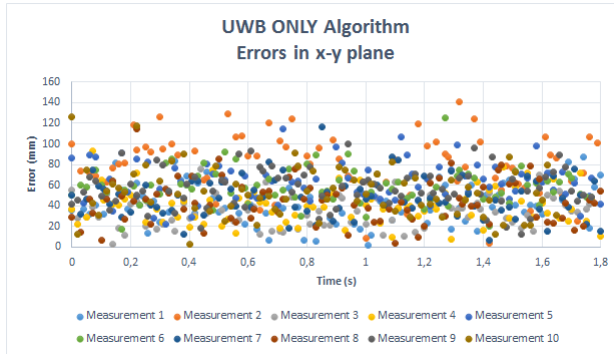


Figure 34 – UWB ONLY algorithm average error in x-y plane.

It can be observed that the samples errors are spreader than in the TRACKING. On the other hand, the shift in the mean value of the errors is closer to the target than using the previous algorithm. Calculating the average of the errors presented in Figure 34, it is possible to obtain an overall mean error of 51,91 mm and a standard deviation for all samples of 23,42 mm. Verifying the standard deviation of each measurement separately and later calculating the average, the result found was 19,65 mm.

Comparing both algorithms in one point, it can be concluded that the UWB ONLY has presented better results in accuracy with an overall error closer to the expected value. However, the variation of the errors was higher. On the other hand, the TRACKING algorithm has

presented a smaller standard deviation, but with greater mean error. Thus, these results have reinforced the idea that a combination of both algorithms could produce a good relation between standard deviation and average error.

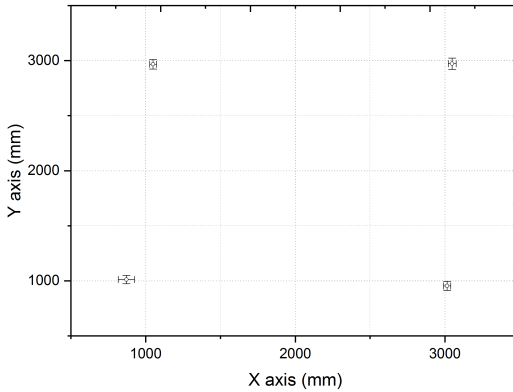


Figure 35 – UWB ONLY algorithm standard deviation in each point.

In Figure 35, the mean value and the standard deviation of all the four measured points are shown. The standard deviation presented in this graph, is the calculation of x-axis and y-axis separately, and not the combined value presented before. As in the past algorithm, the greatest mean error, was found in the point 1, and it can be caused due to the reflections presented inside the room, or even small difference in the hardware of the fixed points closer to this point. Even though the source of the errors where not identified, all of the tests in all algorithms were performed under the same conditions, and the same environment.

6.1.3 Proposed Algorithm

The Proposed algorithm was developed firstly to verify the performance of an approach combining both solutions presented before. The idea behind this methodology is to start the measurements in the point using the UWB ONLY and then, after a defined number of samples, the algorithm is changed to TRACKING. Thus, a significant reduction in the exponential decay behavior presented in Figure 15 is

expected. Also, the intention was to reduce the standard deviation and increase the accuracy when calculating the average of the measurements during 1,8 seconds.

Hence, the firmware was modified to change from one algorithm to the other when the number of samples acquired was equal to 40, and it was submitted to the same experiment. This number of samples have been chosen to be used, as it represents approximately half of the number of samples acquired in both algorithms during this defined period. Thus, half of the samples are acquired with one approach and half with the other. The results in x-axis and y-axis are shown in Figures 36 and 37 respectively.

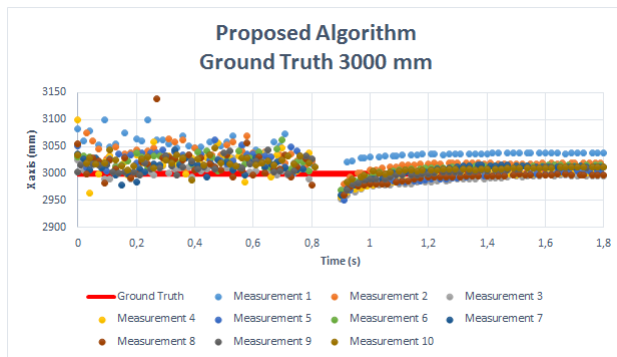


Figure 36 – Proposed algorithm results in x-axis at point 10

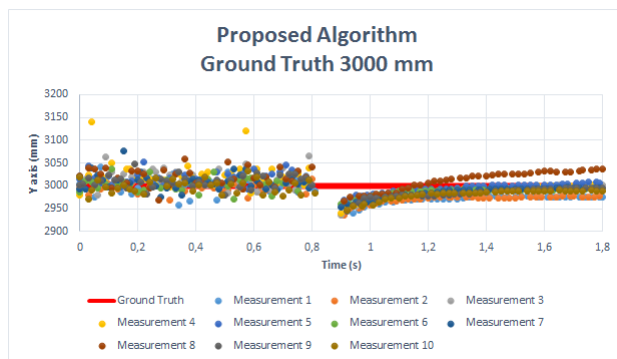


Figure 37 – Proposed algorithm results in y-axis at point 10.

From the results, the expected behavior can be observed in ap-

proximately 0,9 seconds, when the system is turned to the TRACKING algorithm, and the exponential decay behavior still present but with a smoother response. Also, the shift in the measurements was removed. This can be explained by the fact that the system is started with previous values closer to the target, which introduces fewer errors to the position estimation than letting it running all the time. In addition, it can be observed that there is a time gap between the change of the algorithms, which is also expected considering the time to process this change. However, this time gap reduces the number of samples acquired in the period defined.

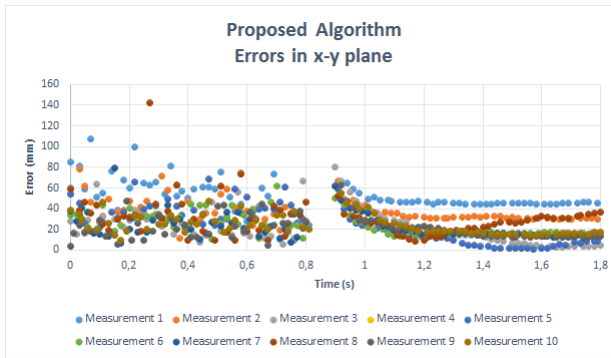


Figure 38 – Proposed algorithm average error in x-y plane.

Thus, to compare the performance of the algorithm after a measurement in a fixed point, the same errors were calculated as before. They are shown in Figure 38. When calculating the overall average error of the samples in the graph, during 1.8 seconds, the result is 28,29 mm. A possible reason is that, when the algorithm is changed, the errors of the UWB ONLY are corrected by the exponential response of the TRACKING algorithm. So during the average calculation, the result is closer to the expected point. In addition, for this algorithm, the standard deviation achieved for all the samples was 18,10 mm, and the average of the standard deviation of each measurement was 15,56 mm.

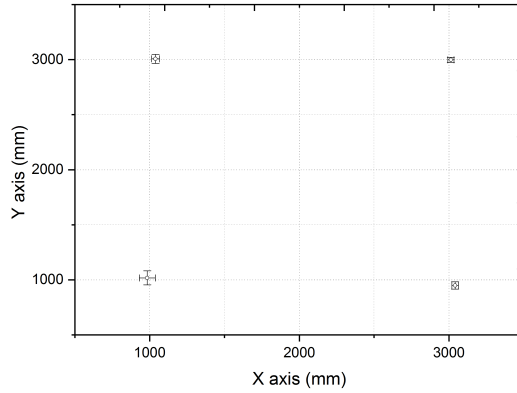


Figure 39 – Proposed algorithm standard deviation in each point.

Finally, the mean value and the standard deviation is presented in Figure 39. From this graph it can be verified that this algorithm has a good relation between mean error and standard deviation, due to the closeness of the estimated position with all the four ground truth measured points. Also, the same problem of higher errors was observed in the point number 1 as stated in all the past results.

6.1.4 First Experiment Results Comparison

From the results presented, Table 4 can be used to compare the behavior of each algorithm.

Comparing the results, one could infer that the Proposed algorithm is the best choice for this application, as it has shown the minimum average error, and the smaller standard deviation in all the samples. It shows that the response of this algorithm is very accurate and precise. Also the objective to decrease the standard deviation of the UWB ONLY in each measurement when combining the algorithms was achieved, and can be verified in the third column of the table, which shows that in each measurement, in average, it is between the UWB ONLY and the TRACKING algorithms.

However, it is important to observe that the analysis presented in this experiment was obtained with the results for just one point. Even though the experiment has been performed in four different points, the

data collected was considered not enough to have a statistical comparison of the algorithms in the complete area.

Furthermore, the results have shown that the errors are different in each point, even though the test have been performed equal. So, the performance of the algorithms must be analyzed in more points to allow a statistical comparison of the algorithms.

Table 4 – First experiment results.

Algorithm	Average Error in X-Y Plane	Overall Standard Deviation in X-Y Plane	Average of the Standard Deviation of each Measurement
TRACKING	76.97 mm	22.05 mm	2.42 mm
UWB ONLY	51.91 mm	23.42 mm	19.65 mm
Proposed	28.29 mm	18.10 mm	15.56 mm

6.2 SECOND EXPERIMENT

Aiming to have more data to achieve a statistical analysis about the behavior of the algorithms in a greater coverage area, a second experiment was performed, moving the robot around 10 fixed points. The pattern of movement is presented in Figure 20, and it was repeated five times to have a reliable average in the errors calculation.

During this experiment, all the algorithms calculated the positioning during 1.8 seconds, but only the average of the data was sent to the control station. This was performed in order to test the firmware working in the final version.

The example of the first data acquired from the control station, for each point, using the TRACKING algorithm is shown in Figure 40.

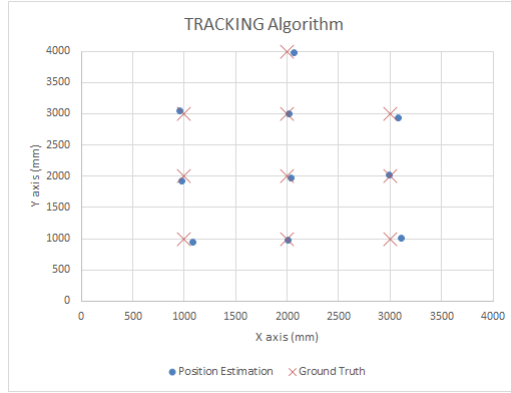


Figure 40 – Example of the first measurement during the second test.

The test was repeated five times, and the results are shown in Figure 41.

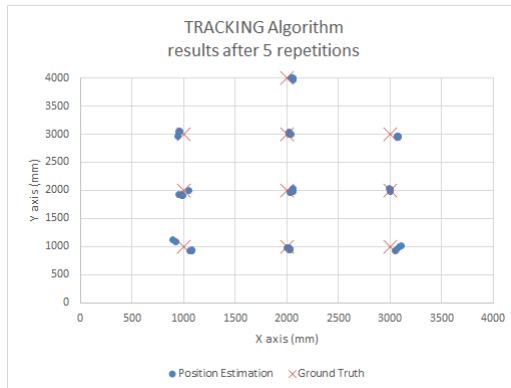


Figure 41 – TRACKING algorithm results for the second test plan.

The same experiment was performed with the UWB ONLY and the Proposed algorithms, and the results are presented in Figure 42 and 43, respectively.

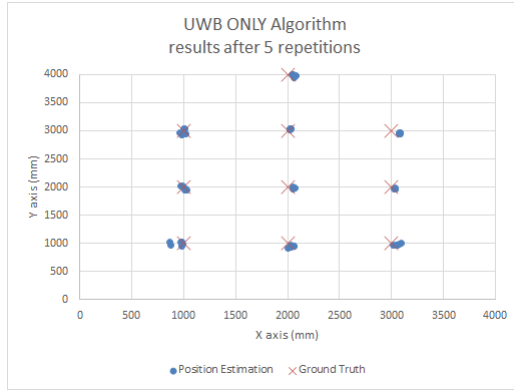


Figure 42 – UWB ONLY algorithm results for the second test plan.

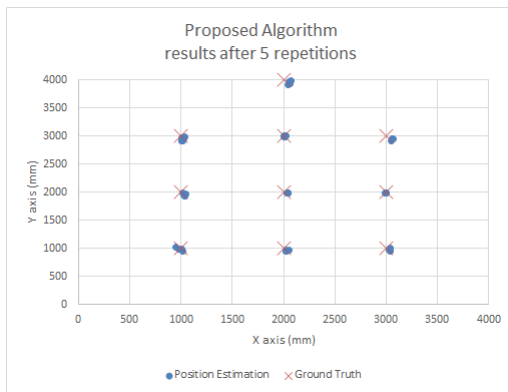


Figure 43 – Proposed algorithm results for the second test plan.

Thus, all the algorithms were compared and the results are shown in Figure 44.

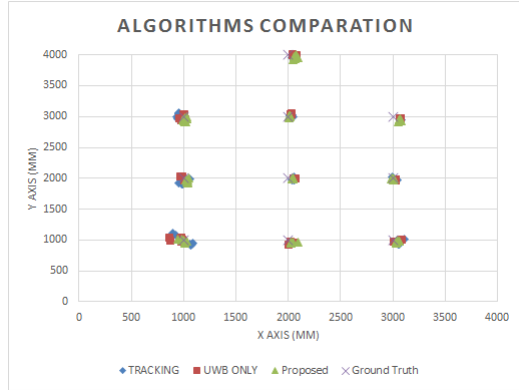


Figure 44 – Comparison between algorithms average measurement 5 times in each point.

From these results, it can be observed that all the algorithms performed similar, with all the measurements close to the ground truth points. However, it can also be stated that the points positioned in the x-axis value of 1000 mm had greater errors than the others. Also, for the ground truth point number 1, this behavior can be seen.

A possible reason for this behavior can be related to the fact that the anchors closer to these points, in x-axis position equal zero, were placed near the walls of the laboratory, what can have caused reflections in the signal, adding errors to the measurements. In addition, as presented in section 6.1.1, the errors in this area can be related to many different sources which, unfortunately, could not be measured during these experiments. On the other hand, the experiments were performed aiming to analyze the behavior of the algorithms under the same environment, and in the same conditions during all the tests.

It is known that in the real robot, new sources of reflections can be present, and many other tests could be executed to evaluate the sources of the errors in this area. However, the approach used in this work was to evaluate the behavior of the algorithms to choose one to be used in this system, considering that they would perform similar, independent of the measurement area. Hence, the system would be flexible to be used in different scenarios.

Aiming to achieve numerical results from the comparison presented in Figure 44, the cumulative distribution function of the data from each algorithm was realized. This graph represents the probability of the error to be equal or less than the value in the abscissa axis.

The graph is shown in Figure 45. It is a result of 50 measurements from each algorithm, after the test pattern have been repeated 5 times in the 10 defined points.

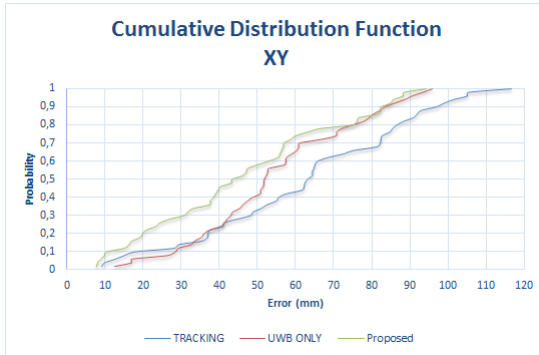


Figure 45 – Cumulative distribution function of the algorithms.

The analysis of this graph must be given in relation to the abscissas value, for instance, the probability of a measurement result in an error equal or less than 50 mm is 34% for the TRACKING, 44% for the UWB ONLY, and 58% for the Proposed algorithms.

So from this result, it can be observed that the TRACKING algorithm had the worst performance in the measured points, with errors up to 116 mm, and the greatest probability of high errors. However, for the last two algorithms, the results in this analysis were similar with the maximum error achieved of 95,96 mm for the UWB ONLY, and 94,30 mm for the Proposed algorithm. On the other hand, the difference of probability between the algorithms start to get more relevant when the errors are smaller.

Table 5 – Analysis of the CDF for errors in x-y plane.

Algorithm	30%	50%	70%	90%
TRACKING	48.17 mm	63.16 mm	82.29 mm	97.01 mm
UWB ONLY	43.14 mm	51.86 mm	61.06 mm	83.49 mm
Proposed	30.68 mm	43.29 mm	56.94 mm	82.42 mm

For this reason, aiming to compare the algorithms, the table 5 was created and it presents the numerical results for the cumulative

distribution function. Table 6 presents the average of the errors in all the measurements and all the points.

Table 6 – Average error of all measurements in x-y plane.

Algorithm	Average Error
TRACKING	62.58 mm
UWB ONLY	54.95 mm
Proposed	46.63 mm

From these analysis, it can be observed that in general the probability of smaller errors occurs with the Proposed algorithm. Even though the Proposed algorithm and the UWB ONLY have performed similar for greatest errors, the high probability of small errors, the lower mean error, and the best performance in the results in test 1 were considered enough to define the Proposed algorithm to be used in this application.

6.3 POWER SUPPLY EVALUATION

Since the embedded PCB in the robot must be powered by a 24 V power supply, a flyback converter was used to isolate the sub-systems to the main power source. So the converter was evaluated to guarantee a safe connection with the sub-systems.

Figure 46 shows the efficiency of the converter according to the output current. In normal operation, the Raspberry needs 0,5 to 1 ampere to operate, so at least 60 % of efficiency are guaranteed in the standard mode. Also, the load regulation was evaluated, and Figure 47 shows the relation between the output voltage with the load current.

During the experiments, no peaks in voltage exceeded 5.6 V, even when turning the system on or off, neither during current load transitions. Also, the protection circuit present in the converter's controller was acting to protected against currents over 2 A, so the PCB was considered safe to be connected with the other sub-systems. The PCB working connected with the sub-modules is shown in Figure 48 as well as the model designed during the project.

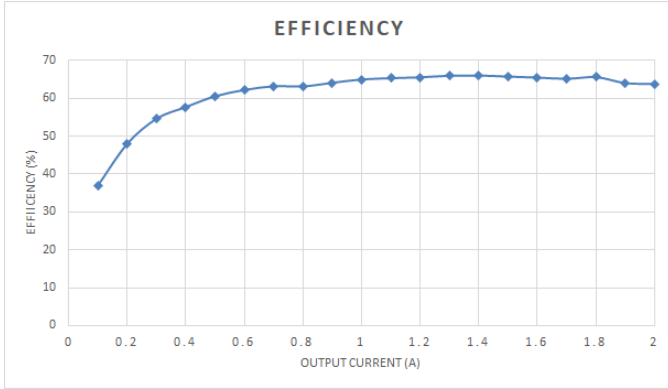


Figure 46 – PCB efficiency analysis.

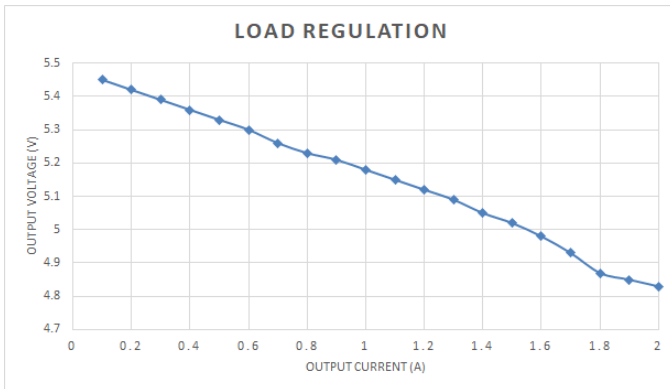
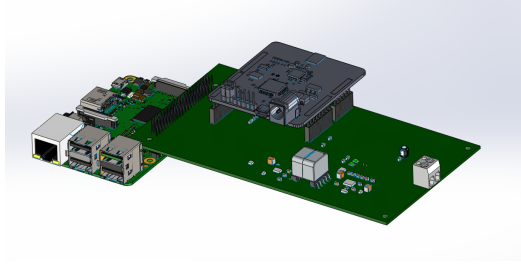
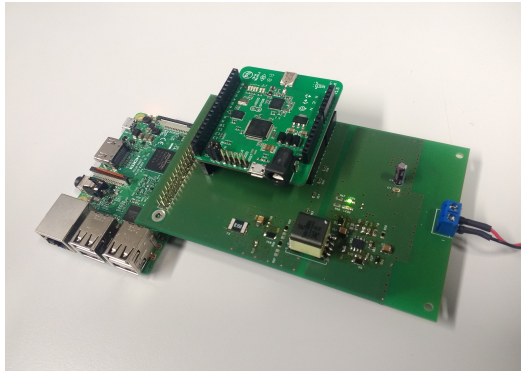


Figure 47 – PCB voltage regulation with load current analysis.



(a) Project in 3D of the embedded PCB.

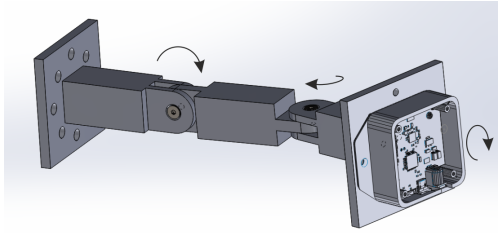


(b) Prototype of the embedded PCB.

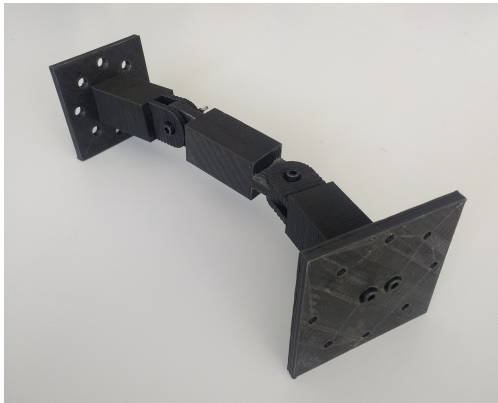
Figure 48 – Result of the embedded PCB.

6.4 SYSTEM PLACEMENT

For the system placement, a prototype of the mechanical structure to place the anchors was produced. The project presented in Figure 12 was printed in a 3D printer, and the parts were assembled using flat-head M5 screws and wing nuts. The prototype is compared with its 3D model and shown in Figure 49.



(a) Project in 3D of the anchors' holder.



(b) Prototype of the mechanical structure to place the anchors.

Figure 49 – Result of the embedded PCB.

In the base of the prototype, holes for M8 screws were made to allow the placement in the aluminum profile present in the robot's structure. Thus, four of this mechanical structure were printed, to be used in each one of the anchors. The placement of the final system in the robot is shown in Figure 50, where the blue marks present the position of the anchors and the red mark presents the embedded part without the connection with the payload.

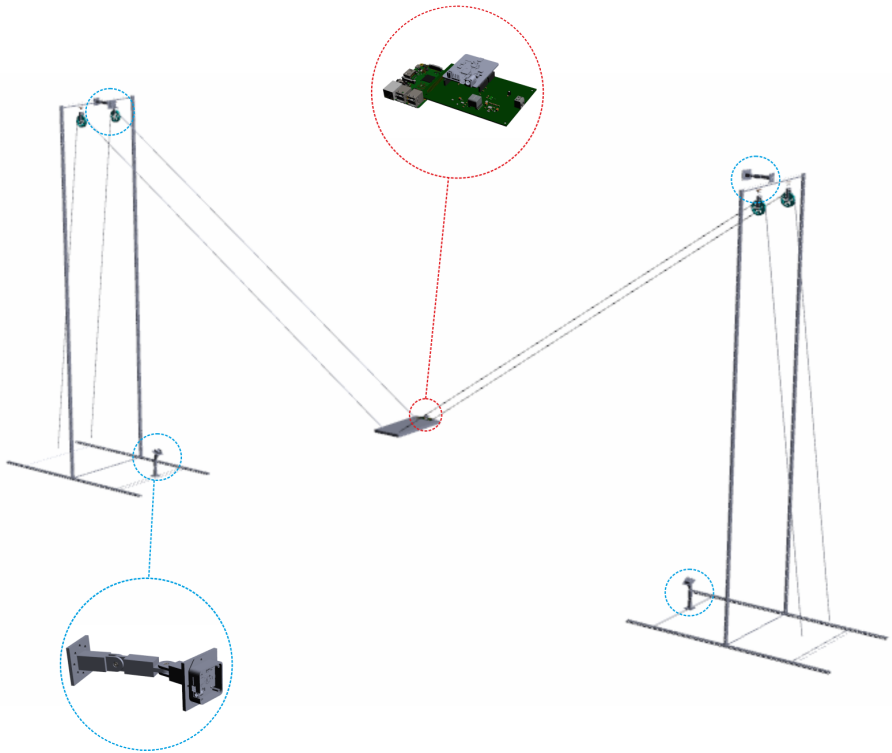


Figure 50 – System placement inside the robot's structure.

7 CONCLUSION

This work has presented the development of a full positioning system for planar cable-driven robots. At the beginning of the work, an overview of positioning systems technologies is presented and the ultra wideband is chosen to be used in this application. In this project, this technology was chosen due to its performance against the multipath effect, and the high accuracy that this system is able to achieve in indoor scenarios.

Then, a whole system was proposed and tested inside a laboratory to evaluate the behavior of the algorithms available in the Pozyx's development kit. During the experiments, patterns were observed in the errors and one new algorithm was proposed to be evaluated. So, basic statistical analysis was performed aiming to choose the best configuration, which would result in a flexible system able to be used in any planar robot structure. Thus, the results have shown that the proposed algorithm has performed better in the points measured, with a mean error smaller than 5 cm. Also it can be used immediately after the system is turned on, which differs from the TRACKING algorithm.

In addition, a PCB was developed for the embedded part of the robot. The tests have shown that the project is safe to power the embedded part of the system with industrial power supplies and the efficiency achieved was greater than 60% for the load in the range evaluated. Finally, a mechanical structure was developed to place the anchors, and a suggestion of the system placement inside the robot structure is presented for complete application.

7.1 FUTURE WORK

The next steps for this work start with the test of the system applied in the real robot, where it is possible that the placement in the movable part can introduce reflections in the positioning system. Therefore, tests shall be performed to evaluate the system inside the structure, and validate its accuracy. Also, the performance of the system must be evaluated when connected with the payload, which in this project will be a camera, and it can again introduce reflections to the RF signals.

For the communication, the cables available inside the robot for powering the PCB can transfer data over an AS-Interface, which is an

industrial standard open protocol. Even though it was not covered in this work, the PCB was designed to allow the communication over this protocol. Therefore, the firmware in the Raspberry must be changed to receive the start command and also answer the data over it. Furthermore, the control station must be modified to send the command over serial, and not through a push-button. Thus, the system could communicate directly with the robot's main computer, and it would be flexible to be installed in any robot structure, with or without communication cables in the movable part.

Finally, enclosures for the embedded part and the control station must be developed, to operate the PCBs safely.

REFERENCES

ALARIFI, A. et al. Ultra wideband indoor positioning technologies: Analysis and recent advances. **Sensors**, Multidisciplinary Digital Publishing Institute, v. 16, n. 5, p. 707, 2016.

ALLIANCE, Z. **Zigbee specification version 1.0**. [S.l.]: April, 2005.

ARDUINO. **Arduino Uno Rev3 Technical Especification**. 2019. Available in: <<https://www.arduino.cc>>.

BAI, Y. B. et al. Overview of rfid-based indoor positioning technology. In: CITESEER. **GSR**. [S.l.], 2012.

BRESSON, G. et al. Simultaneous localization and mapping: A survey of current trends in autonomous driving. **IEEE Transactions on Intelligent Vehicles**, v. 2, n. 3, p. 194–220, Sep. 2017. ISSN 2379-8904.

BRUCKMANN, T. et al. Wire robots part i: Kinematics, analysis & design. 2008.

BUCHWALD, J. **The Creation of Scientific Effects: Heinrich Hertz and Electric Waves**. [S.l.]: University of Chicago Press, 1994. ISBN 9780226078885.

COTERA, P. et al. Indoor robot positioning using an enhanced trilateration algorithm. **International Journal of Advanced Robotic Systems**, v. 13, n. 3, p. 110, 2016. Available in: <<https://doi.org/10.5772/63246>>.

DAS, K. R.; IMON, A. A brief review of tests for normality. **American Journal of Theoretical and Applied Statistics**, v. 5, n. 1, p. 5–12, 2016.

FARID, Z.; NORDIN, R.; ISMAIL, M. Recent advances in wireless indoor localization techniques and system. **Journal of Computer Networks and Communications**, v. 2013, 09 2013.

GALAT, J.; LOWE, W. **Discovery of Longitude, The**. [S.l.]: Pelican Publishing Company, 2012. ISBN 9781455616374.

GEZICI, S. et al. Localization via ultra-wideband radios: a look at positioning aspects for future sensor networks. **IEEE signal processing magazine**, v. 22, n. 4, p. 70–84, 2005.

GHAVAMI, M.; MICHAEL, L.; KOHNO, R. **Ultra wideband signals and systems in communication engineering**. [S.l.]: John Wiley & Sons, 2007.

GLANZER, G. et al. Semi-autonomous indoor positioning using mems-based inertial measurement units and building information. In: **2009 6th Workshop on Positioning, Navigation and Communication**. [S.l.: s.n.], 2009. p. 135–139.

GUO, K. et al. Ultra-wideband-based localization for quadcopter navigation. **Unmanned Systems**, World Scientific, v. 4, n. 01, p. 23–34, 2016.

HOFMANN-WELLENHOF, B.; LICHTENEGGER, H.; COLLINS, J. **Global Positioning System: Theory and Practice**. [S.l.]: Springer Vienna, 2012. ISBN 9783709132975.

HUANG, C. et al. Real-time rfid indoor positioning system based on kalman-filter drift removal and heron-bilateration location estimation. **IEEE Transactions on Instrumentation and Measurement**, v. 64, n. 3, p. 728–739, March 2015. ISSN 0018-9456.

IEEE Std 802.11-2016. IEEE standard for information technology?telecommunications and information exchange between systems local and metropolitan area networks specific requirements - part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications. **IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012)**, p. 1–3534, Dec 2016.

IGLESIAS, H. J. P.; BARRAL, V.; ESCUDERO, C. J. Indoor person localization system through rssi bluetooth fingerprinting. In: **2012 19th International Conference on Systems, Signals and Image Processing (IWSSIP)**. [S.l.: s.n.], 2012. p. 40–43. ISSN 2157-8702.

KAPLAN, E.; HEGARTY, C. **Understanding GPS/GNSS: Principles and Applications, Third Edition**. [S.l.]: Artech House Publishers, 2017. (GNSS technology and applications series). ISBN 9781630814427.

KOLAKOWSKI, M.; DJAJA-JOSKO, V. Tdoa-twr based positioning algorithm for uwb localization system. In: **2016 21st International Conference on Microwave, Radar and Wireless Communications (MIKON)**. [S.l.: s.n.], 2016. p. 1–4.

KOTARU, M. et al. Spotfi: Decimeter level localization using wifi. In: ACM. **ACM SIGCOMM Computer Communication Review**. [S.l.], 2015. v. 45, n. 4, p. 269–282.

KUMAR, P.; REDDY, L.; VARMA, S. Distance measurement and error estimation scheme for rssi based localization in wireless sensor networks. In: **2009 Fifth International Conference on Wireless Communication and Sensor Networks (WCSN)**. [S.l.: s.n.], 2009. p. 1–4.

LEDERGERBER, A.; HAMER, M.; D'ANDREA, R. A robot self-localization system using one-way ultra-wideband communication. In: **2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)**. [S.l.: s.n.], 2015. p. 3131–3137.

LOY, N. V.; VERBEECK, G.; KNAPEN, E. High accuracy indoor positioning system to monitor spatial use in dwellings. In: **MEASURING BEHAVIOUR**. [S.l.], 2018.

MAI, V. et al. Local positioning system using uwb range measurements for an unmanned blimp. **IEEE Robotics and Automation Letters**, v. 3, n. 4, p. 2971–2978, Oct 2018. ISSN 2377-3766.

OUMAR, O.; SATTAR, T.; TOKHI, M. Indoor localization of mobile robots with wireless sensor network based on ultra wideband using experimental measurements of time difference of arrival. In: **congresoutp, vol. 1, n.º 1**. [S.l.: s.n.], 2018. p. 455–468.

PERTTULA, A. et al. Distributed indoor positioning system with inertial measurements and map matching. **IEEE Transactions on Instrumentation and Measurement**, v. 63, n. 11, p. 2682–2695, Nov 2014. ISSN 0018-9456.

POZYX. **Documentation**. 2019. Available in: <<https://www.pozyx.io/>>.

RASPBERRY PI FOUNDATION. **Raspberry Pi 3 Model B . The final revision of our third-generation single-board computer**. 2019. Available in: <<https://www.raspberrypi.org/>>.

RIBEIRO, G. G. L. et al. An outdoor localization system based on sigfox. In: **2018 IEEE 87th Vehicular Technology Conference (VTC Spring)**. [S.l.: s.n.], 2018. p. 1–5. ISSN 2577-2465.

ROHM SEMICONDUCTOR. **BD7F200EFJ Datasheet**. 2017. Available in: <<https://www.rohm.com/datasheet/BD7F200EFJ-LB/bd7f200hfn-lb-e>>.

SCHERHäUFL, M.; PICHLER, M.; STELZER, A. Uhf rfid localization based on phase evaluation of passive tag arrays. **IEEE Transactions on Instrumentation and Measurement**, v. 64, n. 4, p. 913–922, April 2015. ISSN 0018-9456.

SHI, G.; MING, Y. Survey of indoor positioning systems based on ultra-wideband (uwb) technology. In: ZENG, Q.-A. (Ed.). **Wireless Communications, Networking and Applications**. New Delhi: Springer India, 2016. p. 1269–1278. ISBN 978-81-322-2580-5.

SHI, Q. et al. Anchor self-localization algorithm based on uwb ranging and inertial measurements. **Tsinghua Science and Technology**, v. 24, n. 6, p. 728–737, Dec 2019. ISSN 1007-0214.

SHIT, R. C. et al. Location of things (lot): A review and taxonomy of sensors localization in iot infrastructure. **IEEE Communications Surveys Tutorials**, v. 20, n. 3, p. 2028–2061, thirdquarter 2018. ISSN 1553-877X.

SMITH, S. W. et al. **The scientist and engineer’s guide to digital signal processing**. [S.l.]: California Technical Pub. San Diego, 1997.

STROHMEIER, M. et al. Ultra-wideband based pose estimation for small unmanned aerial vehicles. **IEEE Access**, v. 6, p. 57526–57535, 2018. ISSN 2169-3536.

SUGANO, M. et al. Indoor localization system using rssi measurement of wireless sensor network based on zigbee standard. **Wireless and Optical Communications**, v. 538, p. 1–6, 2006.

VAHDATPOUR, A.; AMINI, N.; SARRAFZADEH, M. On-body device localization for health and medical monitoring applications. In: **2011 IEEE International Conference on Pervasive Computing and Communications (PerCom)**. [S.l.: s.n.], 2011. p. 37–44.

XU, G.; XU, Y. **GPS: Theory, Algorithms and Applications**. [S.l.]: Springer Berlin Heidelberg, 2016. ISBN 9783662503676.

XU, Y. et al. Uwb-based indoor human localization with time-delayed data using efir filtering. **IEEE Access**, v. 5, p. 16676–16683, 2017. ISSN 2169-3536.

YOON, P. K. et al. Robust biomechanical model-based 3-d indoor localization and tracking method using uwb and imu. **IEEE Sensors Journal**, v. 17, n. 4, p. 1084–1096, Feb 2017. ISSN 1530-437X.

YU, B. G. et al. A time-based angle-of-arrival sensor using cmos ir-uwb transceivers. **IEEE Sensors Journal**, v. 16, n. 14, p. 5563–5571, July 2016. ISSN 1530-437X.

ZAFARI, F.; GKELIAS, A.; LEUNG, K. A survey of indoor localization systems and technologies. **arXiv preprint arXiv:1709.01015**, 2017.

ZHANG, D. et al. Localization technologies for indoor human tracking. In: **2010 5th International Conference on Future Information Technology**. [S.l.: s.n.], 2010. p. 1–6. ISSN 2159-7006.

APPENDIX A - PCB Schematic

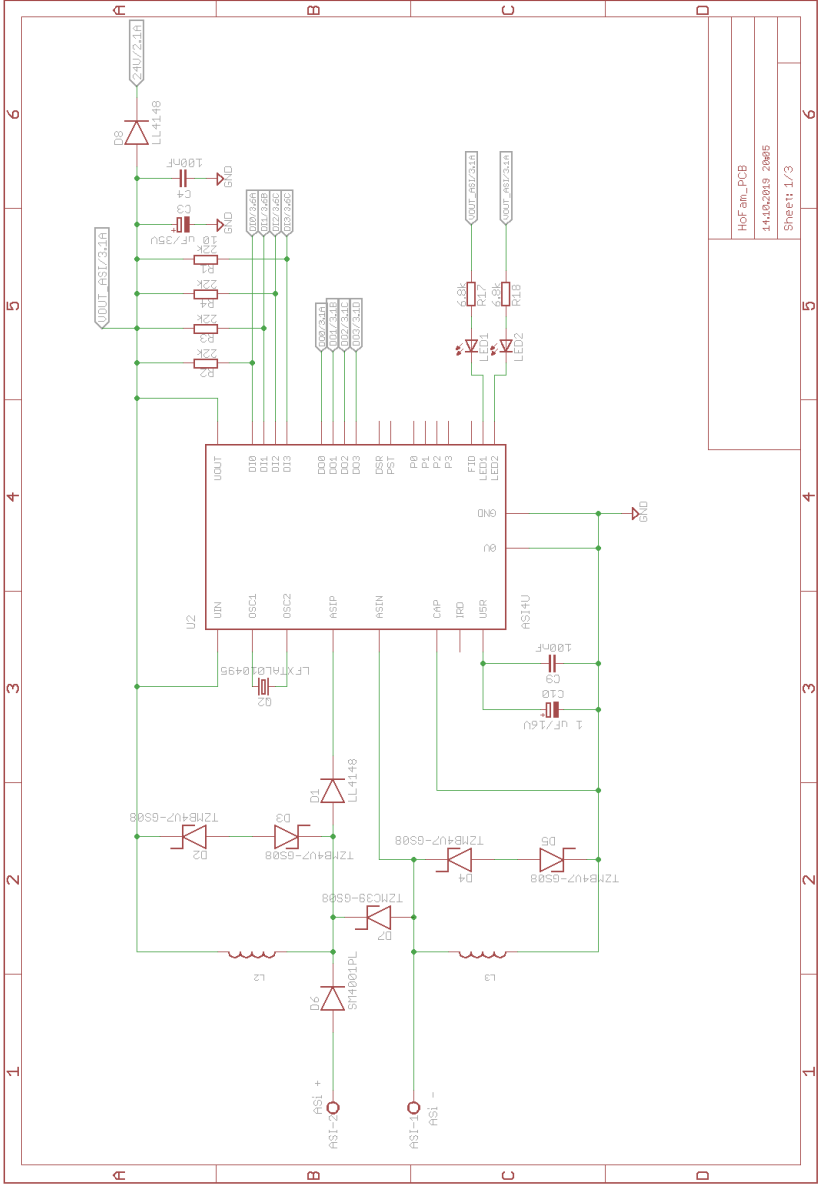


Figure 51 – PCB schematic of the AS-Interface circuit sheet 1.
Source: From the author.

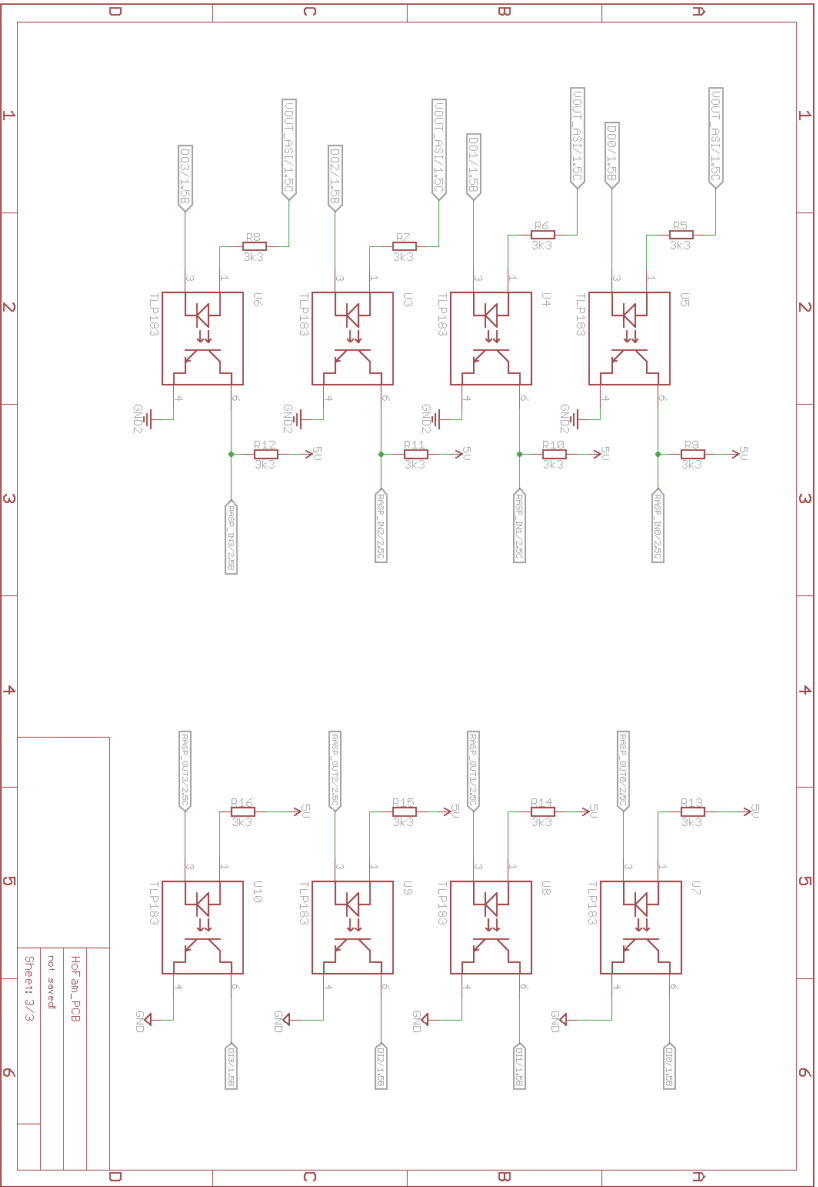


Figure 52 – PCB schematic of the AS-Interface circuit sheet 2.
Source: From the author.

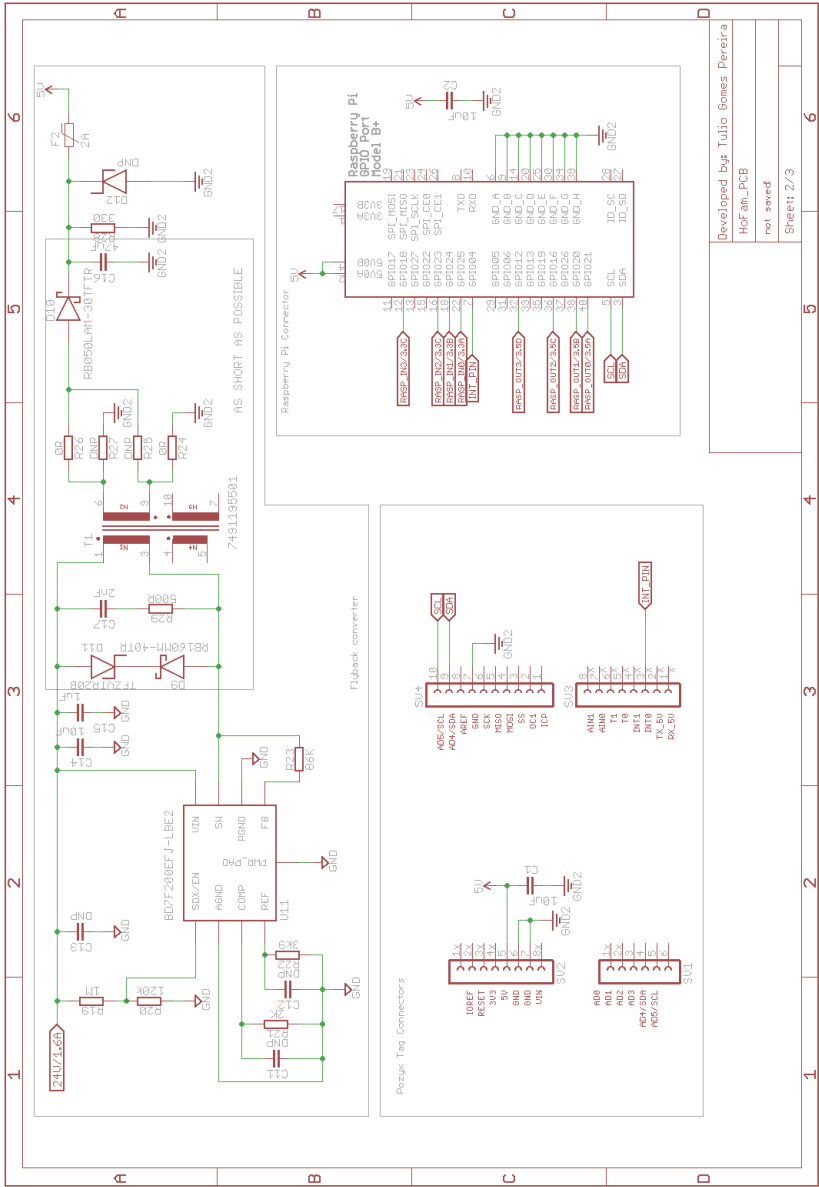


Figure 53 – PCB schematic of the flyback converter and connectors sheet.
Source: From the author.