Universidade de Aveiro
Ano 2022

Vitor Emanuel Reis da Rocha

Conceção de sistema de aquisição e processamento de dados para análise da onda de pulso com um sensor em fibra ótica

Conception of an acquisition and data processing system for pulse wave analysis with a fiber optic sensor

Universidade de Aveiro
Ano 2022

Vitor Emanuel Reis da Rocha

Conceção de sistema de aquisição e processamento de dados baseado em sensor de fibra ótica para análise da pressão sanguínea

Conception of an acquisition and data processing system based on optical fiber sensors for blood pressure analysis

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Eletrónica e Telecomunicações, realizada sob a orientação científica do Doutor Mário José Neves de Lima, Professor Auxiliar do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro e do Doutor Paulo Fernando da Costa Antunes, Professor Auxiliar do Departamento de Física da Universidade de Aveiro.

o júri / the jury

presidente / president	Prof.ª Doutora Susana de Jesus Mota Professora Auxiliar, Universidade de Aveiro
vogais / examiners committee	Prof. Doutor Paulo Sérgio de Brito André Professor Catedrático do Instituto Superior Técnico da Universidade de Lisboa
	Prof. Doutor Mário José Neves de Lima Professor Auxiliar, Universidade de Aveiro

agradecimentos

Começo por agradecer aos Professores Mário José Neves de Lima e Paulo Fernando da Costa Antunes por toda a ajuda e disponibilidade ao longo de todo este processo quer direta ou indiretamente.

Em segundo lugar quero agradecer a Doutora Cátia Sofia Jorge Leitão pelo acompanhamento e informação fornecida que se revelaram ser fundamentais para a conclusão desta dissertação.

Um agradecimento á empresa Brightstuff e em especial ao Eng. Jorge Cunha por todos os recursos e pela hospitalidade oferecida desde o início deste percurso.

Também agradeço ao Eng. João Prata, ao Dr. Simão Carvalho e ao Dr. Mesquita Bastos por toda a ajuda em diversas partes técnico-científicas da dissertação. Agradeço também o suporte do IT (Instituto de Telecomunicações), do i3N (Institute for Nanostructures, Nanomodelling and Nanofabrication) do Hospital Infante D. Pedro do Centro Hospitalar do Baixo Vouga

Por último, quero agradecer aos meus pais e familiares mais próximos, assim como aos meus amigos e colegas de curso que fizeram com que estes 5 anos se tonassem uma experiência inesquecível.

Obrigado a todos.

palavras-chave Fibra ótica, sensor, onda de pulso arterial, saúde, processamento de sinal

resumo

O trabalho desenvolvido teve como foco a criação de um sistema de aquisição e processamento de dados capaz de extrair informação relacionada com a onda de pulso sanguínea de um individuo, utilizando sistemas óticos aliados a um suporte eletrónico. Para isso foi concebido um sensor baseado em fibra ótica que utiliza as variações na intensidade da luz refletida que vai ser modulada pelo movimento da pele como resultado das alterações no volume do sangue no interior das artérias. O sinal do sensor ótico é adquirido e processado por um sistema eletrónico também desenvolvido no âmbito desta dissertação. Foi, posteriormente, desenvolvida uma aplicação em MATLAB® que sará responsável pela maior parte do processamento do sinal obtido, incluindo: aquisição de ondas de pulso, extração e análise de parâmetros relacionados com a pressão sanguínea, amostragem de dados em tempo real, estabelecimento de comunicação por porta serie ou Bluetooth e também uma interface simples e interativa capaz de se adaptar a vários tipos de sensor.

Numa primeira fase, foi realizada uma revisão da literatura sobre vários métodos de aquisição de formas de onda adequadas, para a obtenção parâmetros chave relacionados com a pressão sanguínea utilizando sensores baseados em intensidade. Após a revisão da literatura, foi implementado o sistema principal, que é constituído por um sensor ótico baseado em modulações de intensidade ótica e por um sistema de aquisição, processamento e transmissão de dados. Posteriormente, é desenvolvida a aplicação onde serão implementados os algoritmos estudados, a comunicação sensor-computador via porta serie e Bluetooth e a interface para interação com o utilizador. O sistema de aquisição e sensor desenvolvidos foram pré-validados em ambiente hospitalar em comparação com um sistema comercial e usualmente utilizado.

O projeto descrito ao longo da dissertação foi realizado em parceria com a empresa Brightstuff.

keywords Optical fiber, sensor, arterial pulse wave, healthcare, signal processing.

abstract

The developed work was focused on the creation of a data acquisition and processing system capable of extracting information related to an individual's blood pulse wave, using optical systems allied to an electronic support. For that, a fibre optic-based sensor was designed, which uses the variations in the intensity of reflected light, that will be modulated by the movement of the skin as a result of changes in the volume of blood inside the arteries. The signal from the optical sensor is acquired and processed by an electronic system also developed within the scope of this dissertation. It was subsequently developed, a MATLAB® application, which will be responsible for most of the signal processing, including: pulse wave acquisition, extraction and analysis of blood pressure parameters, real time data sampling, serial port or Bluetooth communication and a simple and interactive interface capable of adapting to various types of sensors.

In a first phase, a literature review was carried out on various methods of waveform acquisition for obtaining key parameters related to blood pressure using intensity-based sensors. After the literature review, the main system was implemented, which consists of an optical sensor based on optical intensity modulations and a data acquisition, processing, and transmission system. Afterwards, the application is developed where the studied algorithms, the sensor-computer communication via serial port and Bluetooth and the interface for user interaction are implemented. The acquisition system and sensor developed were pre-validated in a hospital environment in comparison with a commercial and commonly used system. The project described throughout the dissertation was carried out in partnership with the company Brightstuff.

Index

ACRONYMS AND SYMBOLS

ABPI	Ankle-Brachial Pressure Index
ADC	Analog-to-Digital Converter
AIX	Augmentation Index
ASP	Ankle Systolic Pressure
ATT	Attribute Protocol
BLE	Bluetooth Low Energy
BP	Blood Pressure
CS/SS	Chip Select/Slave Select
CVD	Cardiovascular Diseases
DBP	Diastolic Blood Pressure
DN	Dicrotic Notch
DP	Diastolic Peak
DPTI	Diastolic Pressure-Time Integral
DSSS	Direct-Sequence Spread Spectrum
DW	Dicrotic Wave
ED	Ejection Duration/Ejection Fraction
EM	Electromagnetic Interference
ESH	European Society of Hypertension
FOS	Fiber Optic Sensors
FW	Forward Wave
GATT	Generic Attribute Profile
GFSK	Gaussian Frequency-Shift Keying
HBPM	Home Blood Pressure Monitoring
HR	Heart Rate
HTN	Hypertension
IP	Inflexion Point
LED	Light Emitting Diode
LSB	Least Significant Bit
MAP	Mean Arterial Pressure
MISO	Master-Input Slave Output
MOSI	Master-Output Slave-Input
MP	Mean Point
MSB	Most Significant Bit
NA	Numerical Aperture
OFDM	Orthogonal Frequency-Division Multiplexing
OPBM	Office Blood Pressure Monitoring
PPG	Photoplethysmography
PSC	Central Systolic Pressure
PTT	Pulse Transition Time
PWD	Pulse Wave Delineator
RF	Radio Frequency
RRK	Riva-Rocci/Korotkoff
RTC	Real Time Clock
RW	Reflected Wave

SAP	Systolic Ankle Pressure
SBP	Systolic Blood Pressure
SCLK	Serial Clock
SEVR	Subendocardial Viability Ratio
SPI	Serial Peripheral Interface
SP	Systolic Peak
SPTI	Systolic Pressure-Time Integral
UART	Universal Asynchronous Receiver-Transmitter
USB	Universal Serial Bus
UUID	Universally Unique Identifier

LIST OF FIGURES

LIST OF TABLES

Chapter I

1. Introduction

A stable blood pressure is vital to have a healthy life. Without the pressure that forces our blood to flow around the circulatory system, no oxygen or nutrients would be delivered through our arteries to the tissues or organs. The blood cells and antibodies delivered are essential to preserve the immune system.

Research shows that more than 30% of the population suffers from hypertension but less than 50% of those people monitor their BP regularly. High BP values can lead to cardiovascular diseases (CVD), and because of that, regular BP measurements are vital to the prevention and treatment of such diseases. Usually there aren't meaningful physical manifestations of early hypersensitive states. Hearth attacks, strokes or organ failures are the first symptoms of untreated high BP and that's why everyone (especially people with hypertension stages) should keep track of their BP daily. The dissertation is based on this principle, to develop a small device that is simple, cheap, and as accurate as possible to be used at home or at medical clinics.

This first chapter discusses the historical context and motivation behind this project, the proposed objectives, the main contributions, and the structure of the document.

1.1. Context and motivation

The following subsection focuses on the history and past development of the techniques used throughout this dissertation.

1.1.1. Blood pressure

Although there are indications that early Egyptian civilizations had already experimented with pulsation, it was not until the middle of the 18th century that the pressure generated by circulation in various parts of the body began to be measured. It was Stephen Hales (Janus 1913, Burget 1925) who was responsible for the first measurements of this phenomenon, and he can be credited with the discovery of blood pressure. Johannes Muller, a renowned physiologist of the 19th century quoted 'the discovery of the blood pressure was more important than the discovery of the blood' [1] which further proves the prestige of this achievement.

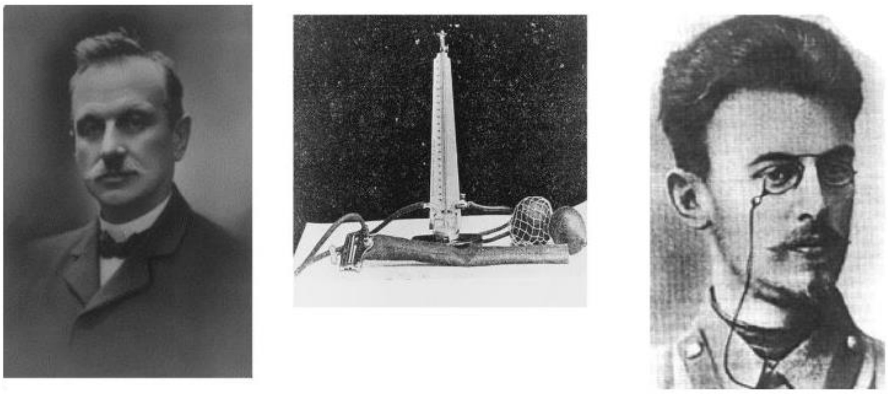


Figure 1.1 – Scipione Riva-Rocci (left), mercury manometer and inflatable cuff (middle) and Nicolai Korotkoff (right) [1]

Hales attended Cambridge University for twelve years, until 1710, and was particularly interested in natural history and astronomy and, while still a student, he constructed a brass machine which illustrated the movement of the planets. It wasn't until 1733 that his first experiments with blood pressure were described, in a publication called 'Volume II of the Statical Essays [2]. Hales introduced a brass pipe connected to a 9 feet glass tube into a horse's leg artery and observed the rise of the blood column to "8 feet and 3 inches above the level of the left ventricle of the heart" and when it was at its full height, "it would rise and fall at and after each pulse 2,3 or 4 inches…" [3]. His work was awarded and held in high regard in Europe, but only a century later the accurate study of blood pressure began when Jean Léonard Maria Poiseuille described the first mercury manometer for the measurement of arterial pressure in his doctoral dissertation. Carl Ludwig improved Poiseuille's work, an added the ability to trace real time arterial pressure waves which looked much like those obtained from standard arterial lines in modern medicine. Only in 1855, Vierodt developed the first non-intrusive method to quantify arterial BP by measuring the pressure required to obliterate an artery [4]. The Russian surgeon Nicolai Korotkoff reported in 1905 that the small changes in BP amplitude can easily be heard with a stethoscope to determine both systolic and diastolic BP, which defined the clinical assessment of BP to modern times.

Until the present day, these methods were refined, and smaller wearable devices started to appear capable of estimating the ambulatory BP. These devices contributed to improve the detection and management of hypertension (HTN) drastically, because the constant real time readings surpass by far the predictive power of isolated in-clinic measurements.

An ideal BP monitor device would have the following characteristics:

- Continuous BP measurements;
- High portability;
- High degree of accuracy and precision;
- Reimbursable;

1.1.2. Fibre Optic Sensors (FOS)

The detection of environmental changes from natural parameters has grown rapidly in the last decades. There is a high interest not only from the scientific community, but also from companies in the development of new fibre optic-based sensors, FOS. For those sensors to have a place in today's market, they must be robust, accurate and cost-effective.

In response to the need of creating non-intrusive techniques for monitoring adverse environments (avoiding their partial destruction) or areas of difficult access, the use of sensors based on optical fibre appears to be one of the most promising hypotheses [5]. There are several advantages when using FOS: their multiplexing ability, small size, lightweight, fast response, no electric signals at the measuring point, easy to sterilize, biocompatible and high sensitivity to external disturbances. Due to all these characteristics, optical sensors are very attractive for applications in the medical, aerospace, and wind energy industries [6],[7].

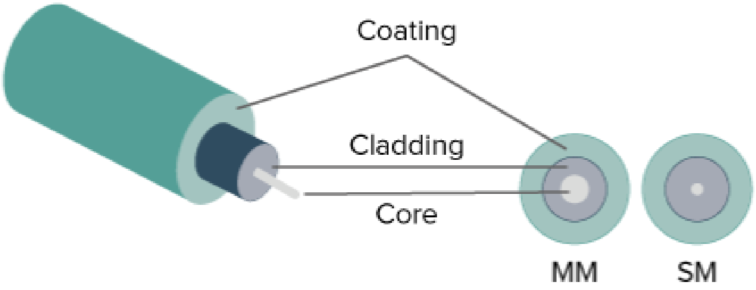


Figure 1.2 - Optical fibre layers

The average optical fibre cable is very similar to the one of figure 1.2. They are usually composed by a core made of pure silica (glass) that provides total internal reflexion, cladding and a plastic buffer coating that protects it from the environment and allows easy handling for splicing or termination [8]. The light is successively reflected inside the core that is protected by the cladding, preventing it from escaping the fibre. To achieve total reflection, the angle of incidence must be greater than the critical angle, and the latter can be manipulated by choosing core and cladding materials with a desired refraction index for the application. The following equation relates both the core and cladding refraction indexes, n_1 and n_2 respectively, with the critical angle [9]:

$$\theta_c = arcsen\left(\frac{n2}{n1}\right) \qquad \text{Eq. 1.1}$$

There is an angle that limits the total internal reflexion for any given fibre, and above that limit, the light will be refracted causing the reflected part of the light to not reach the core. The angle of total internal reflection defines the "numerical aperture" (NA) of the fibre. Using the Snell's Law, it is possible to calculate this limit (Figure 1.3):

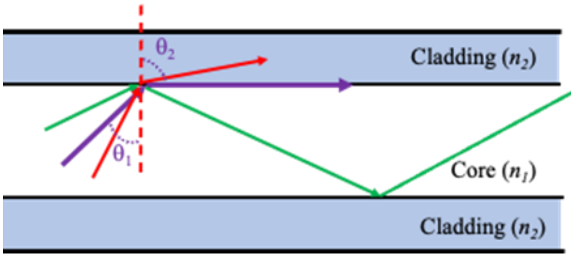


Figure 1.3 – Internal view of an optical fibre

$$n_2 \times sen\,\theta_2 = n_1 \times sen\,\theta_1$$

$$\Leftrightarrow \theta_{max} = sen^{-1}\sqrt{{n_1}^2 - {n_2}^2} \qquad \text{Eq. 1.2}$$

The working principle of FOS is that light from a light source such as an LED, laser, or a broadband light source, is sent through an optical fibre, that when suffering external perturbations it incites some kind of change in the propagated light properties, such as intensity, wavelength, polarization, or other, that can be measured by the optical detector. They can be intrinsic, where the fibre itself acts as the sensor; and extrinsic, where it is used only as a mode of transportation from source to the actual sensor.

FOS have several advantages when compared to other types of sensors: insulating properties allow operation in high-voltage environments, immunity to electromagnetic interference (EM) and corrosion, wide operating temperature range and multiplexing capabilities, where a single light source can be connected to multiple sensors.

These sensors can be classified based on working principles, sensor location and application. Based on operating principles, FOS can be classified into four types:

- **Intensity based FOS**

These types of sensors usually require more light from the source and the majority uses multi-mode fibres with a large cores. When there are physical perturbations in the fibre, there will be a variation of the light passing through the fibre due to the geometry of the fibre being changed, imposing different internal reflection conditions. The transmitted amount of light will be proportional to the external perturbance and can be measured by the receiver. They are subject to strain losses, losses in the fibre-emitter or fibre-receiver connections and losses due to micro & macro bending. Some of their advantages include low cost and easy implementation.

- **Wavelength based FOS**

Wavelength-modulated sensors measure the wavelength change in the fibre [10]. Fibre Bragg Grating sensors are the most common sensors of this type. These intrinsic sensors are fabricated by inscribing a periodic refractive index change in the core of the fibre, so that only the Bragg wavelength is reflected back, and the rest of the spectrum is transmitted unaffected. The reflected Bragg wavelength varies proportionately to external strains induced in the fibre. This makes it possible to measure strain values and its derivatives dynamically. Fibre Bragg Grating sensors are used in many different applications due to its multi-point sensing capabilities.

- **Polarization based FOS**

Polarization can be affected by a number of factors, and therefore this type of sensors can be used to measure various parameters. They are generally used in communication and signal processing and require special fibres with specific polarization features. Figure 1.4 provides an example setup for these types of sensors:

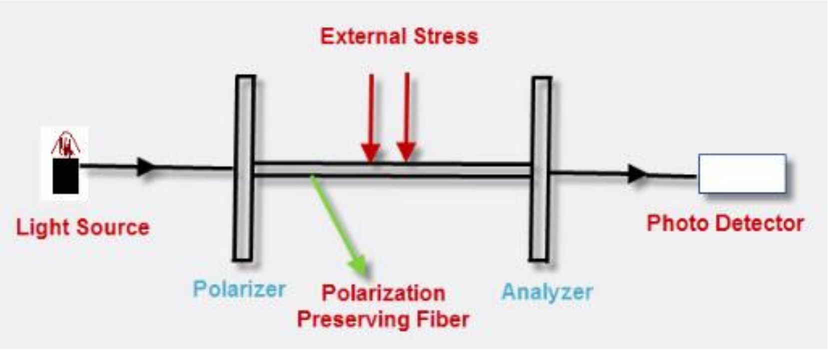


Figure 1.4 - Polarization based FOS [11]

Only part of the fibre is polarized, and it is this section that will act as the sensing element. When the fibre is subjected to external disturbances, such as stress or pressure, there is a change in the output polarization, and by examining the difference between the polarization states it is possible to detect external disturbances.

- **Phase-based FOS**

Two types of interferometers are commonly used in the phase-based FOS configuration: the Michelson and the Mach Zehnder interferometers (Figure 1.5). When the laser beam passes through an Interferometer, it splits in two: one that will be exposed to the sensing environment and another that will be isolated from it, serving as a reference. Both beams will then be recombined, and their mismatch is used to detect external events.

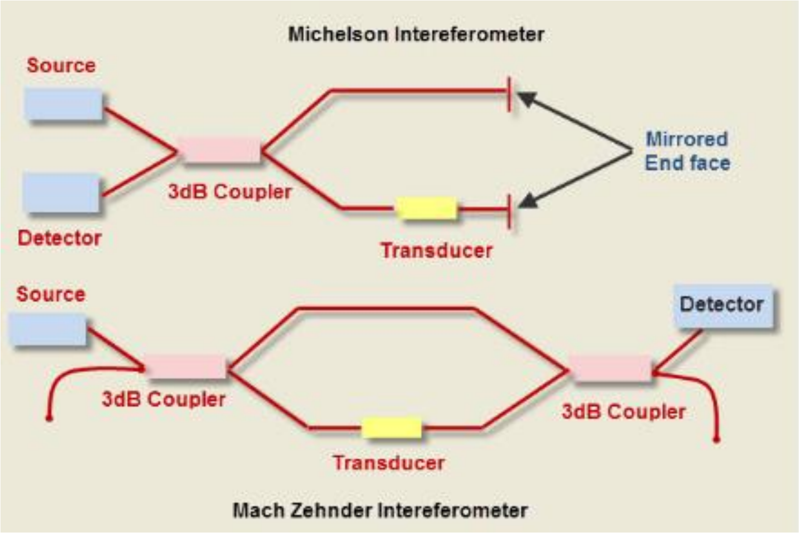


Figure 1.5 - Phase based FOS [11]

In the figure above, it is possible to see the two different implementations: in Michelson's configuration a single coupler is used, and the light passes through the same fibre twice, and consequently the original phase shift per unit is doubled; in Mach Zehnder's configuration two couplers are used and since the emitter and receiver are opposed to each other, the light only passes through the fibre once. The particularities of the Michelson configuration give it a higher sensitivity.

One can also classify them in types based on their application, such as (amongst others):

Chemical application – Sensor used to transform chemical properties into a physical signal capable of being measured. This type of sensor is composed of a receiver and a transducer: the receiver gathers the information, modulated in the transducer by the chemical signature, into an electrical signal. Its functions vary between signal processing, sampling, and data processing.

Physical application – Produced according to the physical effect to be detected. A few examples are photoelectric sensors, extensometers, or accelerometers.

Bio medical application – Mostly used in health care, these sensors are focused on translating biomedical parameters related to human pathologies into electrical quantities for future processing.

Amongst the optical sensors are also photoelectric sensors. These can detect presence, texture changes or movement using optical signal changes, such as signal intensity variation or interruption. A photoelectric sensor consists primarily of an emitter (light source) and a receiver (photodiode, phototransistor,). The light emitted is normally interrupted, reflected or refracted, which changes the amount of light detected by the receiver, and so, photoelectric sensors can be classified based on this behaviour:

- **Through-beam sensors**

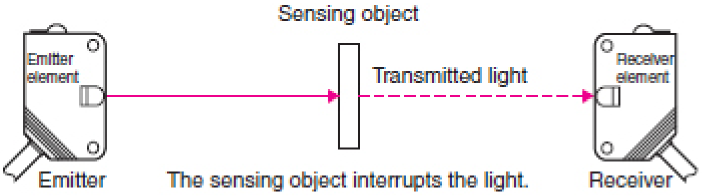


Figure 1.6 – Through-beam sensor [12]

These types of sensors are placed as in the Figure 1.6, so that the light beam from the emitter focuses directly on the receptor. When an object or material passes through and interrupts this beam, the receiver detects this object and preforms its conversion function.

- **Retro-reflective sensors**

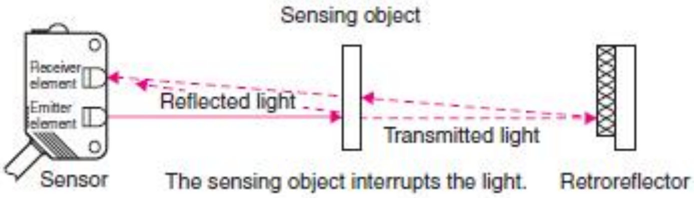


Figure 1.7 - Retro-reflective sensor [12]

In this case, the emitter and the receptor are in the same housing, facing the same direction, but with an added prismatic mirror. The principle is the same as before but in this case the mirror is responsible for the reflection. When and object enters the sensor range, it causes the light beam to diffuse, and the sensor detects the change. Retro-reflective sensors are less susceptible to conditions like colour and angle of the object being sensed. They can be used in applications such as detecting transparent objects like glass. However, retro-reflective sensors have a dead-zone at extremely close distances.

- **Diffuse-reflective Sensors**

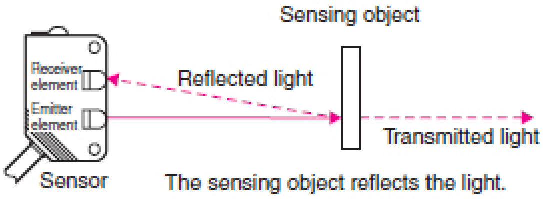


Figure 1.8 - Diffuse-reflective sensor [12]

In the same way as the retro-reflective sensor, here the emitter and receiver are in the same housing, but this time the reflective material is absent. Here the received light beam comes from the amount of light reflected by the object surface. This increase in light intensity is used to detect the object. They are used in situations when the distance of the sensing object changes, making changes in the position where the light is received on the receiver element. This change is converted into a displacement reading. The diffuse light method gives stable measurements and can be used even when the reflection angle variation is large. If the surface is shiny, the accuracy becomes low, and it may not be able to make measurements.

These sensors are quite flexible in terms of characteristics for their applications. For example, a sensor acquired to detect a weak light manifestation cannot have a low sensitivity, because if it does it will not function in the correct way. There are 3 factors that influence the sensor's accuracy: velocity, sensibility, and spectral response. Velocity refers to the sensor response time to light variations; Sensibility measures the proportion between the incident light and the output; and the spectral response which gives information about the limits of light energy that can be detected by the receptor.

1.1.3. Photoplethysmography (PPG)

Photoplethysmography is defined as a non-invasive technique that uses light for measuring the volumetric changes in blood associated with the cardiac cycle in the vascular tissue beds [13]. It was first introduced in 1938 by Hertzman, based on the principle that the empirical observation that the light transmitted through (or reflected from) the living tissue obtains a modulation in time at the heartbeat frequency (Hertzman and Spealman (1937)). After the year 2000 the PPG literature and studies grew rapidly, and it resulted in many different devices and applications through the years, but all the advancements made, the mechanism behind light modulation phenomena of caused by volumetric blood changes is still a matter of debate [14]. PPG is often used to detect blood oxygen saturation, heart rate, blood pressure, arterial aging, …, through the observation of its various waveforms with the added advantage of being able to get continuous results using inexpensive, small devices.

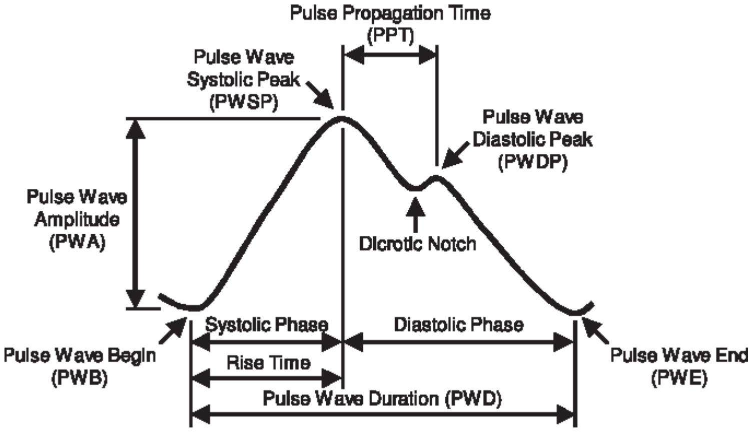


Figure 1.9 – PPG waveform [15]

The waveform seen in Figure 1.9 is obtained, as in all PPG experiments, by measuring the power of the light transmitted or reflected from biological tissue whit the help of a photodetector, when it is being illuminated by some form of light source, like an LED. It consists of two main components: the AC component, which is alternating through time and portrays the changes in blood volume; and the DC component, related to respiration, nervous activity, and thermoregulation. From these pulse waves it is possible to apply different methods to monitor the heart rate and cardiac cycles, respiration, hypo- and hypervolemia [16]. and most importantly for this report, blood pressure.

1.2. Objectives

The main objective of this dissertation is to build a prototype device capable of acquiring and transmit an optical signal from an intensity based optical fibre sensor. Also, a software application for data processing of the pulse wave from the blood pressure readings is intended to be developed. The following milestones were set to accomplish it:

- Develop and test hardware based on a LED and Phototransistor capable of detecting the light intensity variations in the scope of the applications covered in this study.
- Develop software related to microprocessor readings (ADC, sampling frequency, baudrate…).
- Implement series communication to transfer the acquired data to a Tablet/PC;
- Develop a MATLAB application with the following features:
    - Read real time data from the sensor
    - Filtering
    - BP parameters calculation
    - Data display (graphs, tables, displays)
- Add BLE communication.

1.3. Document Structure

In this first chapter the motivation, objectives, structure, and contributions were presented.

In the second chapter, with the theme "state of the art", some of the work developed and reported in literature on the dissertation topics, are presented.

The third chapter is focused on the process of signal acquisition. It will cover all the different phases of testing until reaching the final circuit, in which its output will produce a low noise signal and with all the necessary components for calculating blood pressure.

In chapter 4, the software development associated with the microcontroller and the app design is described. This phase includes a brief description of the logic behind the algorithms and equations used, as well as a description of the application's main functions.

Chapter 5 is where the experimental results are evaluated. Here the sensor is tested in different scenarios and the results are compared against reference studies.

Finally, in chapter 6 an overview is made, conclusions about the objectives achieved are drawn, as well as potential future developments.

1.4. Main Contributions

The work developed during this master's dissertation, allowed to achieve the proposed objectives, and with that, to contribute to the scientific technical knowledge of the arterial pulse wave monitoring systems, especially with:

- The successful conception and implementation of hardware and software for pulse wave acquisition and processing, enabling cardiovascular assessment and analyses.
- The development of a low-cost, non-invasive solution based on light intensity modulation that can rival its commercial counterparts.
- An application capable of processing and displaying data from different types of waveforms and estimate important parameter related with cardiovascular risk assessment.

Chapter II

2. State of Art

In this chapter, many recent applications and concepts related to the topic of this dissertation will be addressed, with the objective of understanding the path chosen for its development. There are many approaches that differ from each other while sharing the same final objective, but the focus will be on the articles that explore similar methodologies and solutions as proposed in this dissertation.

2.1. The gold standard for BP measurements

A method whose development started at the end of the 19th century introduced the first version of the sphygmomanometer (Figure 2.1), which measured brachial blood pressure. A sphygmomanometer is composed of an inflatable cuff responsible for collapsing and releasing the artery involved by the cuff in a controlled way, and a mercury or aneroid manometer used to measure the pressure. It was Scipione Riva-Rocci [17] the first to introduce these type of devices, but it was Nicolai Sergeivich Korotkoff who added the auscultatory technique which allowed both systolic and diastolic BP measurements (SBP and DBP). It became known as the Riva-Rocci/Korotkoff (RRK) method and it is the golden standard against which various BP-measuring devices are tested for the last century.

Figure 2.1 – Sphygmomanometer [18]

While these two researchers shared the same goal of achieving the highest precision BP measurements, both techniques differ from each other in some ways. In December 2008 the article called *"Reintroduction of Riva-Rocci measurements to determine systolic blood pressure?"* [19] pitted the two methods against each other. The authors compared simultaneous samples of the two methods, preformed successively on the same arm of different elderly subjects with vascular diseases and/or hypertension and verified an absolute BP difference of 0 to 5 mmHg between both methods. Their experiments have proven that while more emphasis is placed on SBP treatments, it did not mean that DBP should be ignored. In fact, both techniques did not differ significantly from each other, the only difference was a slight underestimation on the Riva-Rocci's measurements caused by delays on the audible Korotkoff sounds generation in response to the first passage of the pulse wave. Despite the RRK method being a standard, there are many clinicians that avoid using it since a 5 mmHg error in SBP would correspond to an incorrect classification of hypertension status in 84 million individuals worldwide.

2.2. Self-Blood Pressure measurement quality

Accurate hypertension diagnosis requires reliable blood pressure measurements. The average blood pressure values obtained at home are typically lower, but HBPM has been shown to provide more reproducible data than OBPM making them strongly recommended for hypertension monitoring [20].

A study made in 2021 called *"The quality of patients' self-blood pressure measurements: a cross-sectional study"* [21] aimed to get more information about user made errors that may jeopardize the veracity of blood pressure measurement readings. The subjects were chosen considering several criteria, such as: (1) must be at least 18 years old; (2) have declared hypertension in accordance with ESH guidelines; (3) declared regular BP monitoring at home. One hundred people were inquired regarding their knowledge of HBPM techniques and even recreated a BP measurement in the same way they would at home, following the ESH guidelines. Results have shown that only 3% made no errors while the simulation took place, and the most common errors were related to cuff placing and posture. Figure 2.2 shows the number and percentage for every type of error detected in this study.

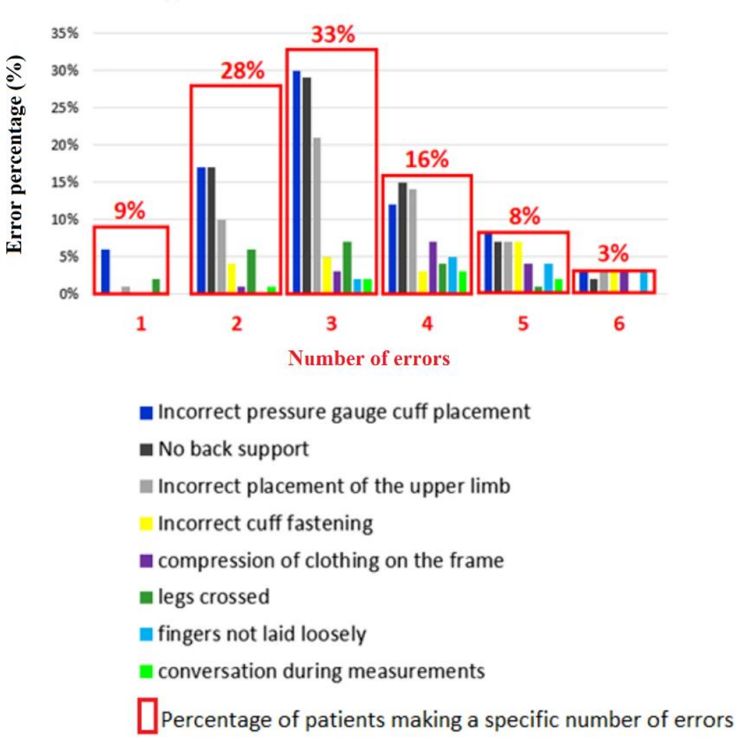


Figure 2.2 - Common self-monitoring error percentage by ocurrence

The credibility of this study may be questioned, due to the fact that these subjects are under pressure in a controlled environment, which may increase their attention span and result in an experience that may not mirror reality. Despite this fact, the results did not deviate from similar studies which further proves that the errors made, especially by recurring HTN patients, could negate the potential benefits of HBPM and that family doctors should act and educate their patients on how to do HBPM correctly.

2.3. Recent PPG applications

Non-invasive techniques have become the focus of BPM development in the last decade, mostly because they bring fewer risks to the patient. That said there are many recent PPG applications and studies that proved to be important in the way this dissertation was conducted.

In June 2019, it was introduced a new wearable device based on PPG to estimate an individual BP. In the paper *"A New Wearable Device for Blood Pressure Estimation Using Photoplethysmogram"* [22], two PPG signals were used as basis for BP estimates. One of these signals is used to estimate the heart rate (HR) and the Pulse Transit Time (PTT) is the time delay between them, after processing. Both HR and PTT are the inputs of a linear model used to calculate an estimation of the SBP and DBP. The final product consisted in a smartwatch with the respective algorithm embedded in it, that utilizes the time delay technique to calculate the BP of its users. The measurements were compared to commercial sphygmomanometer readings in two different database scenarios and the results have shown that the mean error was lower than the standard 5mmHg and that the standard deviation error only has acceptable readings for the DBP, while for the SBP they are a little higher than the standard. But even so, their device shows the same performance for SBP as the commercial devices, and an even better one for DBP measurements, which validates it to be a reliable alternative, non-intrusive method for measuring BP.

An alternative method that is also commonly used for BP estimation relies on the Mean Arterial Pressure (MAP) parameter [23]. This method utilizes the amplitude variations of the oscillations to calculate SBP and DBP. In Figure 2.3 we can see the main points of two different pulse waves, one of which corresponds to a more elastic type of artery, normally associated with younger individuals (blue), and the other corresponds to a more robust type of artery, associated with an older population (red):

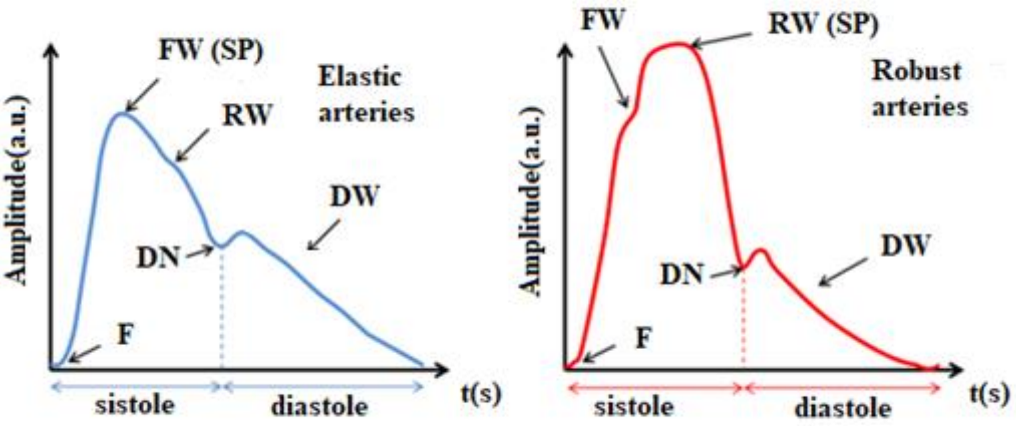


Figure 2.3 – Two types of BP waveform, one with higher elasticity (blue) and the other more robust (red)

In the figure, F corresponds to the wave's foot, which represents the instant where the waveform begins and the last one ends; FW and RW correspond to the pressure components exerted by the heart and its reflexions across the artery respectively. Both these terms can represent the systolic peak (SP), depending on the artery elasticity; the dicrotic notch (DN) represents the nadir point that occurs immediately after the closure of the aortic valves and marks the beginning of the dicrotic wave (DW); this last one results from a small pressure exerted by the heart after the DN.

To extract the best possible pulse, the body part where the light will be focused must provide the best representation of the blood volume changes in the arteries. That said, superficial arteries or places where the skin is thinner like the carotid artery and the earlobe are preferable. The carotid, due to being a superficial artery located close to the aorta, can effectively portray the properties of the latter when subjected to changes in blood volume.

Both waveforms observed in Figure 2.3 can be used to calculate several key indexes for assessing cardiovascular function. While analysing the pulse wave, there are 3 indexes that can be produced: ejection duration index (ED%), augmentation index adjusted for heart rate (AIX@75), and subendocardial viability ratio (SEVR%).

AIX@75 is a method based on pulse-wave reflection and is commonly used as a measure for the augmentation of the central aortic pressure by a reflected pulse wave. If this index assumes positive values, it means that the central pressure increases as the reflected waves increase too. In the case that the index is negative, the reflected waves have no influence on the value of the central pressure. It is calculated using the following equation:

$$AI_x\% = \frac{RW - FW}{SP - DP} \times 100 \qquad \text{Eq. 2.1}$$

RW, FW and SP can be obtained when analysing waveforms like the ones in the figure, while DP corresponds to the maximum point of the diastole.

SEVR is defined as the diastolic to systolic pressure-time integral ratio, and it basically estimates the balance between the blood flow supply and demand. SEVR, or Buckberg index is calculated by dividing the diastolic pressure-time integral (DPTI), which indicates sub endocardial blood supply, by the systolic pressure-time integral (SPTI), which indicates the myocardial oxygen demand.

$$SEVR\% = \frac{DPTI}{SPTI} \times 100 \qquad \text{Eq. 2.2}$$

By replacing both derivatives by each component's (systole and diastole) area in a pulse wave analysis, we get the following equation:

$$SEVR\% = \frac{A_{diastole}}{A_{sistole}} \times 100 \qquad \text{Eq. 2.3}$$

The last index mentioned is the ejection duration or ejection fraction and it is a very important parameter to be considered when preventing heart diseases as lower values usually mean higher probability of cardiovascular failure. It can be calculated by dividing the diastole duration by the systole duration:

$$ED\% = \frac{t_{diastole}}{t_{sistole}} \times 100 \qquad \text{Eq. 2.4}$$

Many experts believe that the central blood pressure in the aorta is a more accurate measurement for heart diseases than the peripheral blood pressure as it can give the doctor a more precise value for arterial robustness. The use of anti-hypertensive drugs can also exert different effects on brachial versus central pressure [24].

To get a patient's central pressure, first it is calculated the mean arterial pressure (MAP). The brachial artery pressure is measured to obtain the values for brachial systolic pressure (SBP) and brachial diastolic pressure (DBP) for them to be used in the following equation:

$$MAP = DBP + \frac{1}{3} \times (SBP - DBP) \qquad \text{Eq. 2.5}$$

The next step is to use the previous MAP value to calculate the central systolic pressure (PSC). By extracting the mean point (MP) form the normalized pulse and the previous PDB we get the following relation:

$$PSC = \frac{MAP - DBP}{MP} \qquad \text{Eq. 2.6}$$

One can also take advantage of the SBP value to calculate another parameter called the Ankle-Brachial Pressure Index (ABPI). An ABPI test is recommended for all patients who present with signs and symptoms suggestive of peripheral artery disease, as it serves as an additional non-intrusive prognostic tool for cardiovascular risks. To calculate this index, it is necessary that the systolic ankle pressure (SAP) is measured, in advance, with the help of a sphygmomanometer, along with the SBP and DBP. Once we have the SBP and SAP, the ABPI will correspond to the ratio between them:

$$ABPI = \frac{ASP}{SBP} \qquad \text{Eq. 2.7}$$

In the fourth chapter of this dissertation, it will be addressed the software approach in which these indexes are calculated, the methods used to reach the final values when analysing pulse waves taken from the carotid artery, and a final diagnostic based on all the indexes.

Chapter III

3. System structure and hardware description

In this chapter the characteristics of each of the project components will be described and its implementation justified. It will also be presented some of the test results regarding filtering and LED biasing that were proved to be decisive to produce the final system. Lastly, a complete system overview will be made including a full schematic and an image of the breadboard circuit used in the remaining chapters. Figure 3.1 illustrates how the final system should look like:

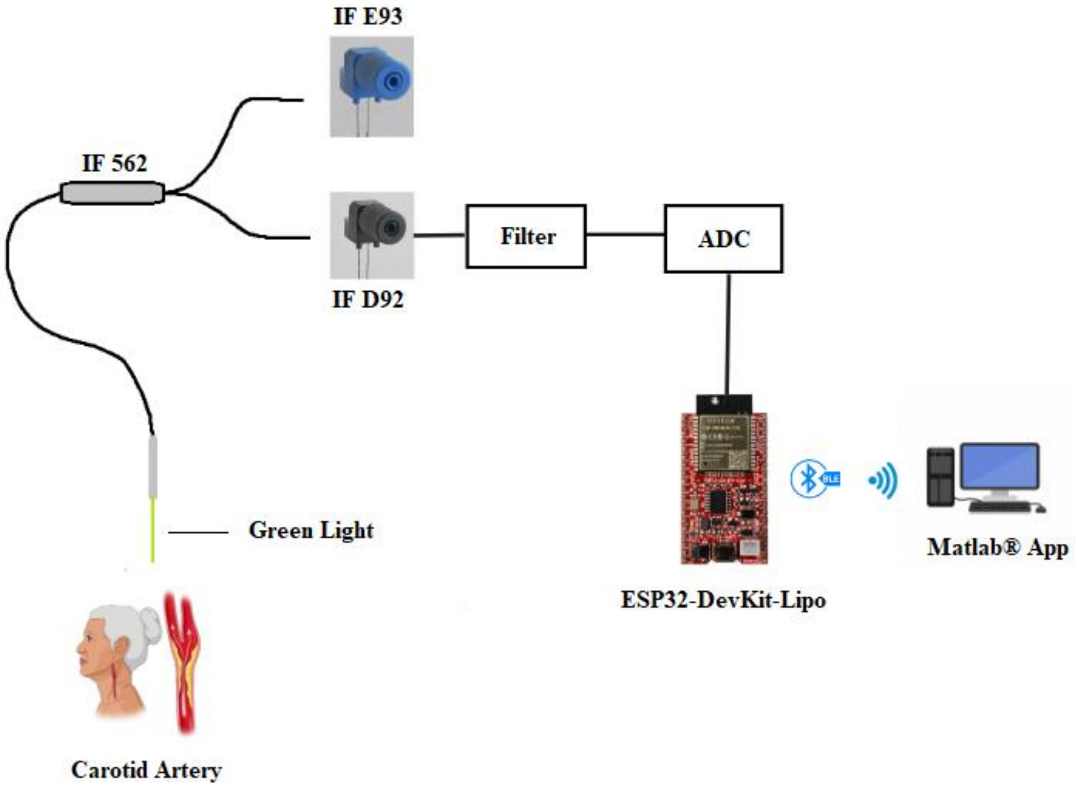


Figure 3.1 - Full system overview

The idea is to build a device that operates as a data acquisition module and that can provide real-time data used to calculate the key points of a blood pressure pulse with the aid of a MATLAB® application. The sensor part will be based on light intensity detection, and it includes a green LED (IF E93[25]), an optical fibre (IF 562 [26]) with a coupler and a phototransistor (IF D92[27]). The light intensity is controlled by varying the current injected in the LED, using a digital potentiometer. The light enters a fibre optic cable with the output being aimed to specific parts of the body where blood pressure measurements are more accurate (ex. carotid artery). Some of the incident light will be reflected back to the fibre and the rest is absorbed or refracted by the skin, tissue and that percentage of reflected light is then fed to the phototransistor which will produce a variable voltage. The change of blood volume in the arteries is what translates in the different voltage values at the output, so high blood pressure values will reflect more light than the cases in which the blood pressure is lower.

The output of phototransistor is filtered by a Sallen-key type, 2nd order low-pass filter that has a cut-off frequency of 30 Hz [28], with the aim of removing unwanted high frequency noise while keeping all the blood pressure pulse components untouched. This filter is present in the final system due to the hardware being assembled on breadboard but it can be later replaced with software filtering when the hardware is converted to an integrated circuit.

An external ADC connected to the ESP32-DevKit-LiPo [29] microcontroller by SPI communication will acquire the data and then the µC will be responsible to transmit that data to a PC or Tablet housing the MATLAB® application. Lastly, the developed app can display the real time data, perform further filtering if necessary and calculate the key points necessary to get all the Blood Pressure measurements requested by the project. The µC, as well as the extra components like the operational amplifier used in the filter and the external ADC can be powered by a rechargeable battery (for portability), but here we are powering the circuit through a benchtop power supply.

3.1. Control unit

Here, the reasoning behind the choice of the control element of the circuit is presented, as well as some of its main characteristics.

3.1.1. ESP32-DevKit-LiPo

One of the main components of the whole system is the microcontroller. Its principal function is to acquire the filtered data from the phototransistor and forward it to the PC or Tablet hosting the application. In addition, it is also responsible for establishing connection to the external ADC, via SPI, and to act as a server/slave in a BLE network. The board features:

- ESP32-WROOM-32 WiFi/BLE module or ESP32-WROVER-B module (BLE capabilities)
- 2.2V ~ 3.3V supply voltage
- 34 GPIOS (3.3V and 12mA each)
- 12-bit ADC
- User button
- Micro USB connector

- Built-in USB-Serial programmer
- Built-in LiPo charger
- LiPo battery connector
- PCB dimensions: (1.9 x 1.1)" ~ (4.8 x 2.8) cm
- Operating temperature -40+85C

The board can be powered by USB or using a battery and has an average current consumption of 80mA when in normal mode and of 150µA during ULP (Ultra Low Power) mode. The 3,3V output from this board will be used to power up some of the components used throughout this dissertation such as the operation amplifier used in the filter and the analogue supply of the external ADC. The ideal option would be to use a lithium polymer battery capable of supplying the 3.3V, in conjunction with a regulator that enables voltage to change from 3.3V to the 5V required for the digital supply of the external ADC. Because the prototype is not yet completed, it was not possible to have an accurate estimation of the total system consumption, thus neither the battery nor the regulator was part of the final circuit of this dissertation.

3.2. LED biasing circuit

The light source should have variable intensity, so that it is adjustable for darker environments or for skin tones that absorb more light. Keeping that in mind, a simple circuit involving an LED and a potentiometer, as the one in Figure 3.2 was implemented:

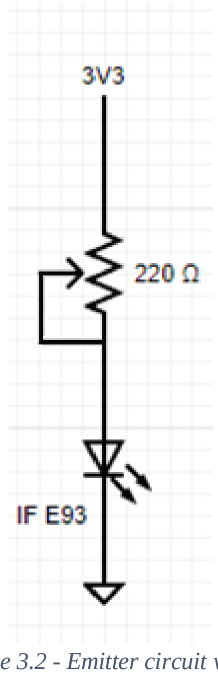


Figure 3.2 - Emitter circuit w/LED

The IF-E93 is a high-output, high-speed, green LED housed in a “connector-less” style plastic fibre optic package. High output and fast transition times are suitable for low-cost digital data links, achieving an attenuation value of less than 0.1 dB/m when coupled to the optical fibre. It has a peak wavelength of 522 nm, which is lower than the commonly used red LEDs, meaning that it has a lower skin penetration capability. Despite the higher penetration, red LEDs are also exposed to more interference from external light, which makes the heart rate signal harder to isolate. So, the green LED was chosen due to the fact that the slightly lower wavelength, makes it possible to sense the larger blood vessels without too much interference from deeper tissues [30].

Regarding the LED biasing circuit, a single resistor in series with the LED would be a good option for simplicity, but, since we want to control the current, a potentiometer was implemented, which value is mentioned ahead (a digital potentiometer can be implemented instead, where its value is set by the application). The resistor values for minimal and maximum current are obtained using the following expression:

$$R = \frac{Vcc - Vf}{If} \qquad \text{Eq. 3.1}$$

Figure 3.3 represents the LED’s forward current as a function of the forward voltage.

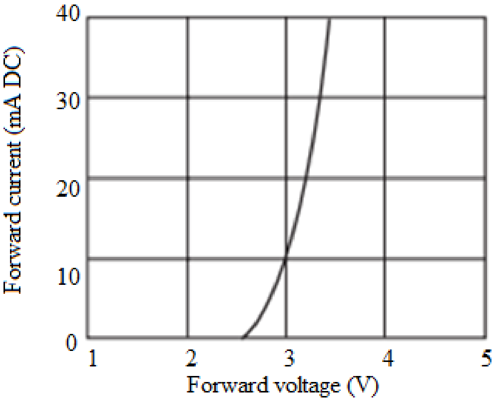


Figure 3.3 - Forward current vs forward voltage [25]

It is possible to predict that for a Vf = 2.6V, current starts flowing through the LED. Considering a Vcc value of 3.3V, R_{max} = (3.3-2.6) / 0.001 = 700 Ω. Based on that value, a potentiometer with a value of 220 Ω was used in primary testing, up until the development of the first prototype, because it allows us to get the maximum and minimum currents flowing across the LED terminals, thus achieving the maximum and minimum light intensity output. The following graph shows the current variation for different resistor values, related to the 220 Ω potentiometer:

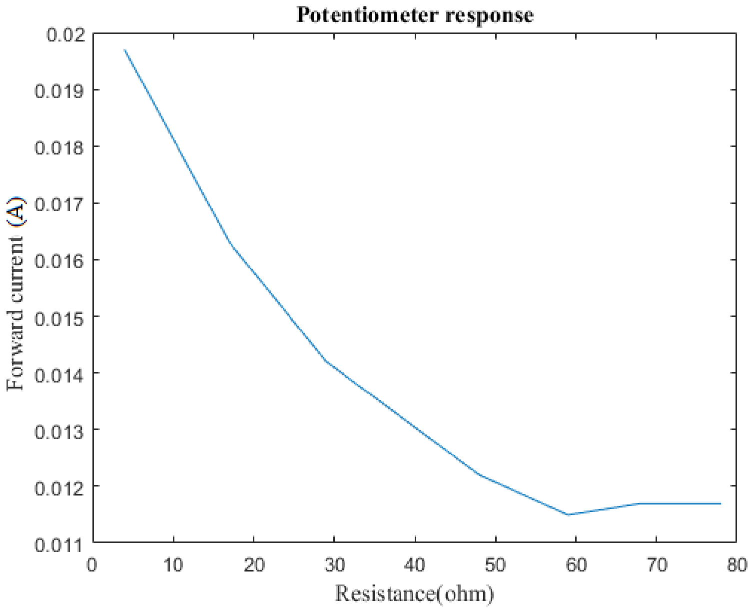


Figure 3.4 - Output current vs resistance

3.3. Reception and filtering

Here, we will address all the issues concerning the reception part of the light emitted by the LED. It includes the phototransistor biasing circuit and the selection of the filter.

3.3.1. Phototransistor biasing

On the receiver side of the circuit, the IF D92 phototransistor was the element used to detect light intensity variations resulting from blood pressure variations and consequently generate an output voltage dependent on the amount of light that is reflected. The choice of a phototransistor over a simple photodiode, despite the lower operational speed and response, has to do with its higher sensitivity and current output due to transistor current gain, and immunity to noise interference. The high efficiency of phototransistors makes them very popular for light detection applications. The circuit was designed to achieve that high sensibility. By using the configuration in Figure 3.5, the output voltage of the phototransistor will change proportionally to the light intensity detected at the base of the phototransistor.

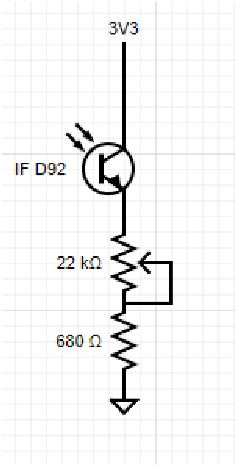


Figure 3.5 – Receiver biasing circuit

This circuit can be looked at as a voltage divider (although this isn't one, as the phototransistor does not have a constant resistance value) in the way that the transistor acts like a low value resistor when illuminated and like a high value one (in the order of megaohms) when in the dark. That said, the circuit output can be estimated based on the voltage divider formula:

$$Vout = \frac{Rtrans}{Rc + Rtrans} \times Vcc \qquad \text{Eq. 3.2}$$

Rc is obtained using the following expression:

$$Rc = \frac{Vcc - Vsat}{Isat} = 12\ 600\ \Omega \qquad \text{Eq. 3.2}$$

The values for Vsat = 0.15V and Isat = 250µA were obtained from the IF D92 datasheet.

For a LOW output (light is detected), the output is given by Vout = (Rtrans*3.3) / (Rtrans+12600). As it was mentioned above, Rtrans assumes a lower value and thus Vout will be lower too. On the other hand, for HIGH output (no light detected), the transistor resistance is so high that the output voltage will be almost equal to Vcc.

A 22k Ω potentiometer was used in the breadboard testing phase mostly due to component limitations at the time. Still, it is more than capable of achieving the maximum output of the IF D92. It was also added a lower value resistor of 680 Ω in series with the potentiometer to serve as protection in the case that the potentiometer value reaches minimum resistance.

3.3.2. Filtering and amplification

The filter choice is an important part of the project, as we are building the sensor “by hand”, which will result in inevitable systematic errors and will introduce additional noise to the ADC input. The breadboard circuit and the component value deviation are directly responsible of introducing the noise that can be seen in Figure 3.6:

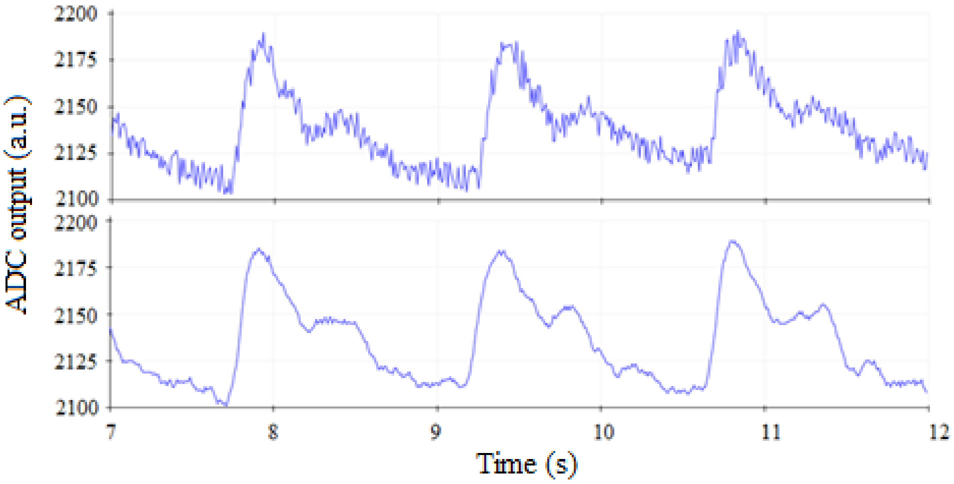


Figure 3.6 – Raw ADC output (above) vs after the moving average method, window size 10 (below)

For that reason, signal filtering is necessary in this primal state, to get the best results for further processing. Signal amplification was also considered but was later replaced by software-based amplification to promote simplicity and avoid additional noise caused by operational amplifiers, if needed. While filtering the phototransistor output, we aim to reduce the high frequency noise while maintaining the BP related components. Based on this fact, some possible implementations where tested. The first one being software filtering, using the moving average method to smooth the hardware output (Figure 3.6). Since this method proved to be quite advantageous, it was used in conjunction with the other implementations to see if it had some influence on the final output.

The tested filter configurations are a) active passband filter; b) 2nd order low-pass filter; c) Sallen-key filter.

a) Passband

The first test was made using a passband filter aiming to eliminate both high frequency noise and the DC component from the signal.

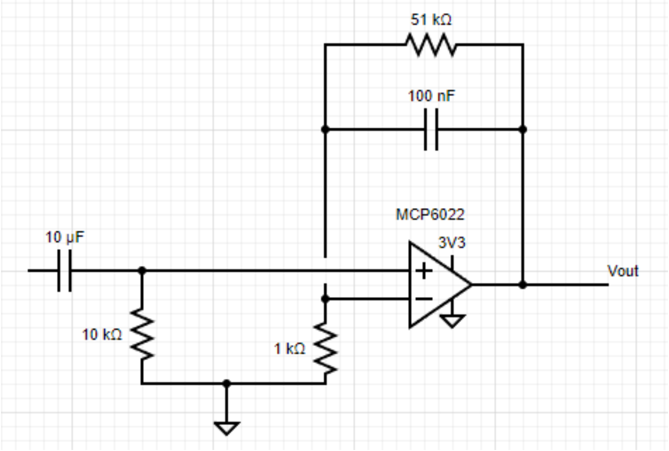


Figure 3.7 - Passband filter, BW = 1 to 31Hz

This passband filter allows us to remove significant noise at frequencies higher than 31Hz while eliminating the signal DC component. Although the pulse wave main signal is usually 0.4Hz to 2.5Hz, the higher cut-off frequency was set to 31 Hz, enabling to preserve all the signal components and subcomponents important to later allow the cardiovascular parameters estimation. The active part of the filter gives even more flexibility in situations when the reflected light is not enough to acquire precise samples. The following figure shows the filter output over time for a sampling frequency of 200 Hz:

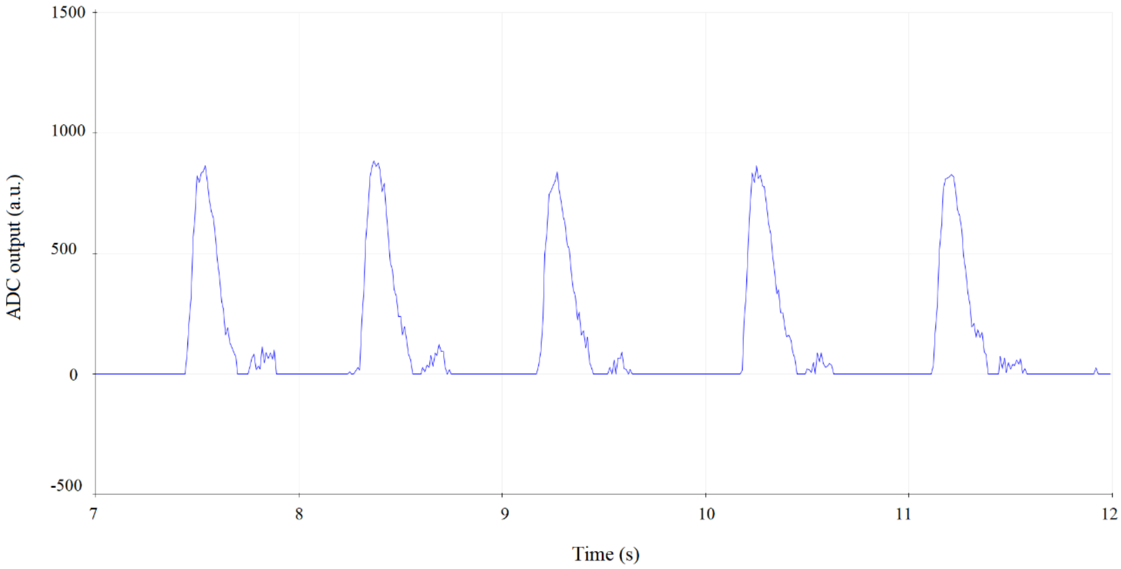


Figure 3.8 - Passband filter output

This first configuration eliminates the noise from the phototransistor, but it cuts off some of the original signal components due to its faster transition times. Furthermore, it was decided that the DC component of the signal could be useful to know if there is enough light reflected to produce variations in the phototransistor output, or to check if the circuit is saturating. So, although the filtering presents satisfactory noise cancelling results, this isn't the optimum configuration to achieve the proposed objectives.

b) 2nd order low-pass filter

Next, we tested an approach using a 2nd order RC filter (Figure 3.9) to promote simplicity and reduce the number of components used on the filter. Here we have set the cut-off frequency to approximately 36 Hz, to check whether the signal components become more prominent. Raising the cut-off frequency to 36 Hz should have no repercussions as the high frequency noise only appears at frequencies above. Figure 3.10 illustrates the comparison between this filter's output and the raw ADC output, for a sampling frequency of 200Hz.

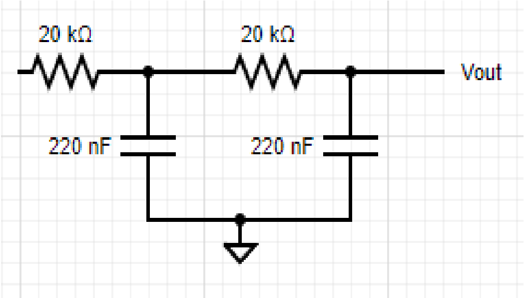


Figure 3.9 - Simple RC second order filter, cut-off freq. = 36Hz

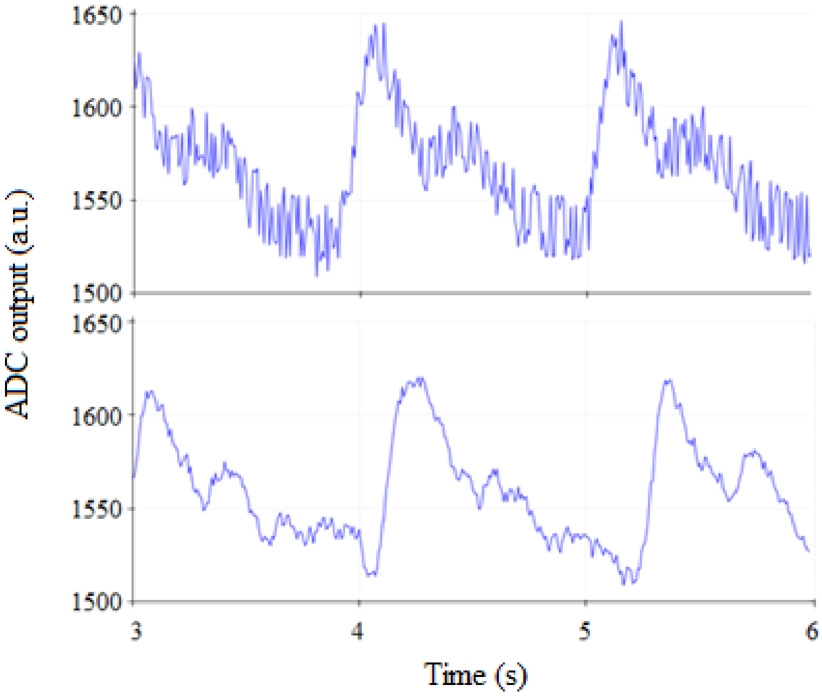


Figure 3.10 - Raw output (above) vs Second order RC filter output (below)

As expected, the simplicity of this filter will result in a lower attenuation at higher frequencies thus producing results noisier than the passband configuration (keeping the signal's DC component will also contribute to some of the added noise). In this case, we can observe that the waveform is much more similar to the expected output, with distinct systole and diastole phases, but nevertheless, the results are very similar to the ones shown when using the moving average method on the raw data. This means that this configuration is not valid as the results do not justify the extra hardware utilized.

c) Sallen-key filter

The low-pass RC filter is limited to its low roll-off factor (two real poles only). In contrast. the Sallen-Key topology can produce a 2nd order lowpass response with much better roll-off and various possible approximations (Butterworth, Chebyshev, Thomson-Bessel…). Figure 3.11 shows a unity gain Sallen-Key filter [31], where the positive feedback path comes from the capacitor C2 being connected to the output and no longer being grounded:

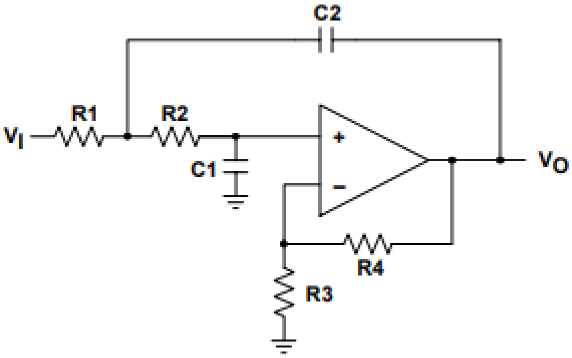


Figure 3.11 – Generalized Sallen-Key low-pass configuration

The circuit works as follows:

- At low frequencies, the capacitors act as an open circuit which means that the signal is simply buffed to the output.
- At high frequencies, the capacitors act as short circuits, and the signal is shunted to the ground at the input. This causes the amplified signal to be near 0V.
- At frequencies near cut-off the impedance of both capacitors matches the order of the resistors.

This is a very popular configuration as it shows the least dependence of the filter preformance on the op amp preformance. For higher degree of simplification we have set R1 = R2 = R and C1 = C2 = C, which results in the following equations for the cut-off frequency and Q:

$$\mathrm{fc} = \frac{1}{2\pi * \mathrm{R} * \mathrm{C}} \qquad \text{Eq. 3.3}$$

$$\mathrm{Q} = \frac{1}{3 - \mathrm{K}} \quad , \quad K = \frac{R3 + R4}{R3} \qquad \text{Eq. 3.4}$$

Using the components that were available at the lab, R = 20k Ω and C = 220nF, the fc will be aproximatly 36 Hz. Figure 3.12 shows the tested configuration:

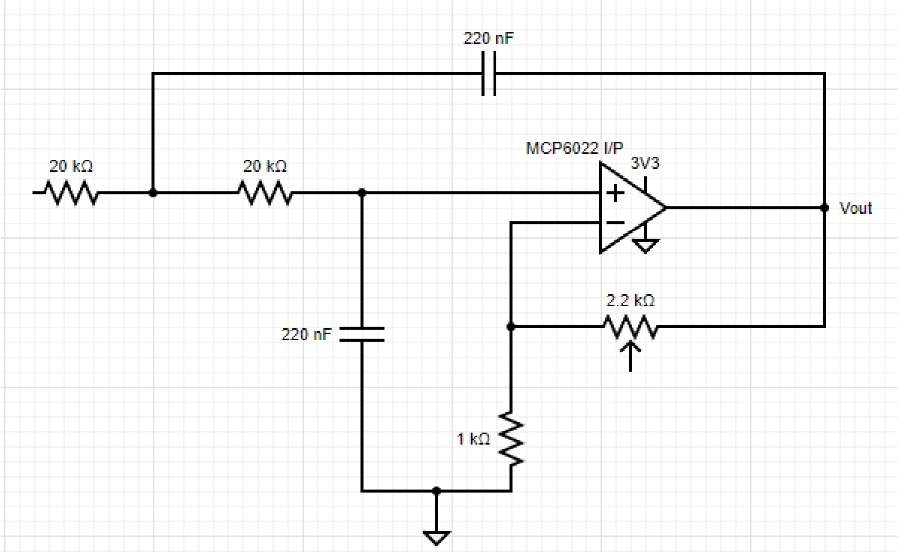


Figure 3.12 - Implemented Sallen-Key configuration, cut-off freq. = 36Hz

The circuit in Figure 3.12 allows a variable gain within]0, 2.2], but testing has sown that the circuit oscillates for a gain of only 1.3. For that reason, amplification will be done exclusively by software and both the potentiometer and the 1k Ω resistor will be removed in the final circuit, making the gain unitary. The output for a sampling frequency of 200Hz can be seen in Figure 3.13:

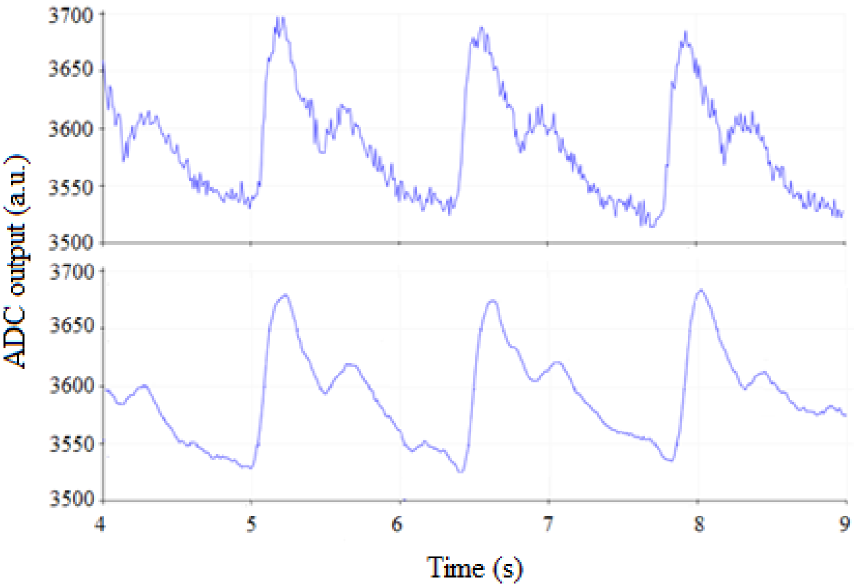


Figure 3.13 - Raw output (above) vs Sallen-Key (below)

The last method proved to get the most satisfactory results in both cases. Here, the filter outperformed the previous configurations, and the result is a much “cleaner” waveform where all the key points are highlighted for further processing. This is the chosen configuration to be used in the testing phase.

By introducing the coupled optical fibre to both LED and Phototransistor and connecting the output of the latter to the Sallen-Key filter, we get the full circuit shown in Figure 3.14:

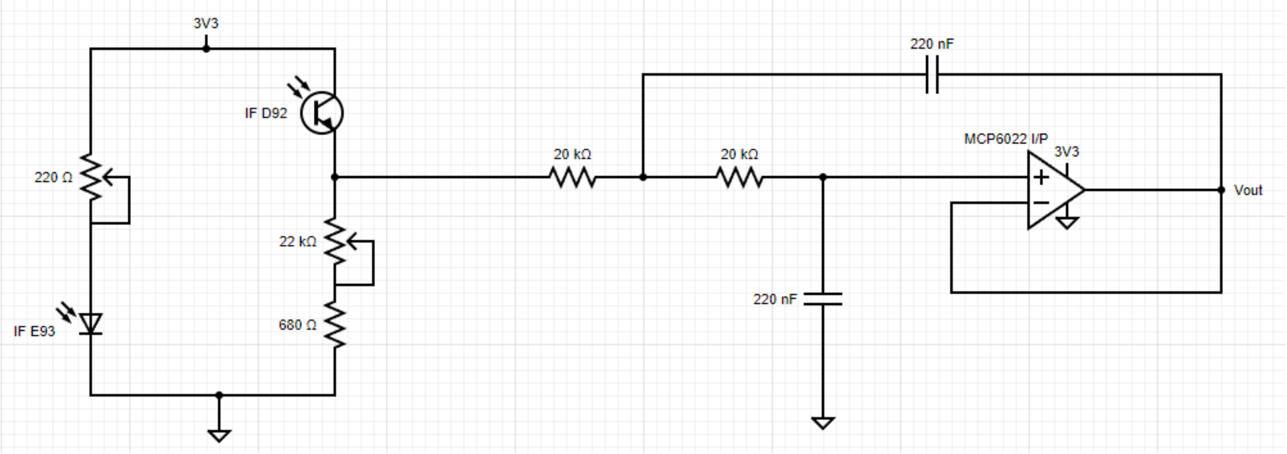


Figure 3.14 - Full breadboard circuit overview

3.4. ADC

The first versions of hardware and software where all tested using the ESP32-DevKit-LiPo internal ADC. The differences between this ADC and the project requirements are shown in Table 3.1:

	ESP32-DevKit-LiPo	Project requirements
Total number of bits	12 bits	At least 16 bits
Sampling Rate (máx)	200 ksps for RTC controller	20 ksps
	2 Msps for DIG controller	
Number of channels	18 channels (analog-enabled pins)	2 channels
Architecture	SAR	Not specified

Table 3.1 - Comparison between the ESP32 internal ADC and the project requirements

While most of the requirements are met, the lower number of bits makes it necessary to use an external ADC that has the right characteristics for the project. The ADC selection process was based, not only on meeting all the requirements, but also based on price and market limitations.

3.4.1. SPI protocol

SPI is a synchronous communication protocol able to transfer continuous information in a Master-Slave relationship [32]. The Master is the one who initiates communication with the Slave/Slaves and can either send or receive data simultaneously. There can only be one Master and, theoretically, an infinite number of slaves, limited by the system load capacity. The four control lines in an SPI system are: MOSI (Master output/Slave input), MISO (Master input/Slave output), SCLK/CLK (clock) and SS/CS (Slave select/Chip select). The **CLK** synchronizes the Master's output to the sampling Slave's sampling bits and each bit requires one clock cycle to be transferred; The **SS/CS** line can be set to LOW whenever the master wants to start communication with a slave, but for more than one salve, there must be multiple CS/SS pins available, corresponding to the total number of slaves in the system; The **MOSI and MISO** lines allow both master and slave to exchange information bit by bit. The ESP32 has 4 SPI different peripherals: SPI0, SPI1, SPI2, and SPI3. The first two are not available as they are used to communicate with the built-in flash memory. SPI2 and SPI3 correspond to HSPI and VSPI respectively, and they are open to use. As there is no difference between the two, it will be used the ESP32 default SPI GPIOs, which correspond to VSPI and can be consulted in Table 3.2:

	MOSI	MISO	SCLK	CS
GPIO	23	19	18	5

Table 3.2 - ESP32 SPI pins

This protocol has several advantages and some negative points when compared to other widely used protocols like UART and I2C:

+Continuous data transfer (no START or STOP bits)
+Simple addressing
+Higher transfer rates (up to 10Mbps)
+MISO and MOSI are independent (simultaneous communication)

-Four links (wires) instead of only two
-No acknowledgements for the received data
-No error detection or correction protocols
-Only one master

It is also important to mention that SPI has four different modes of operation, related to the phase and polarity of the clock. The CPOL and CPHA bits are used to set the clock's polarity and phase respectively. Depending on the CPHA bit, sampling/shifting data can be done on the rising edge of the clock or on the falling edge of the clock. The Master is responsible of choosing the operation mode, depending on its application. All modes and bit combinations are shown in Table 3.3 [32]:

SPI mode	CPOL	CPHA	Clock polarity	Sampling/Shifting Data
0	0	0	Logic Low	Sampling on the rising edge / shifting on the falling edge
1	0	1	Logic Low	Sampling on the falling edge / shifting on the rising edge
2	1	0	Logic High	Sampling on the rising edge / shifting on the falling edge
3	1	1	Logic High	Sampling on the falling edge / shifting on the rising edge

Table 3.3 - SPI mode description

Chapter IV further explains the software involved in the ESP32 SPI configuration to justify all the links and settings chosen to work together with the ADC.

3.4.2. ADS8361 EVM

Due to time limitations, lack of stock and supplier delays, further testing was made with the aid of the ADS8361 EVM demoboard (Figure 3.15) which is the evaluation module for the ADS8361 ADC. This new component checks all the previous requirements with the addition of SPI compatibility. Some of the characteristics are listed in Table 3.4 (the full details can be checked in the product datasheet [33] [34]):

Figure 3.15 - ADS8361

	ADS8361
Resolution	16 bits
Number of channels	4 (2+2)
Sample rate (Max) (ksps)	500
Interface type	SPI
Input range (Max) (V)	5.25
Input range (Min) (V)	0

Table 3.4 - ADS8361 main parameters

The board is powered with 3.3V for the analogue supply, from the ESP32, and 5V for the digital supply, from a computer USB port or benchtop power supply. There is no reference input voltage given, as the EVM provides a 2.5V internal reference. The ADS8361 also has four operation modes depending on the number of channels converted and the number of outputs. The M0 and M1 pins are used to choose between each of the four modes and A0 is used to select between input channels 0 and 1 and is ignored in modes III and IV (four-channel modes). Each of the different pin combinations is summarized in Table 3.5:

M0	M1	A0	Number of channels	Output	Converted channels	Mode
0	0	0	Two-channel	A and B	A0 and B0	I
0	0	1	Two-channel	A and B	A1 and B1	I
0	1	0	Two-channel	A	A0 and B0	II
0	1	1	Two-channel	A	A1 and B1	II
1	0	x	Four-channel	A and B	Sequential	III
1	1	x	Four-channel	A	Sequential	IV

Table 3.5 - ADS8361 EVM operating mode description

Figure 3.16 illustrates all the necessary links/wiring to enable communication between the ADS8361 EVM and the ESP32-DevKit-LiPo, using SPI.

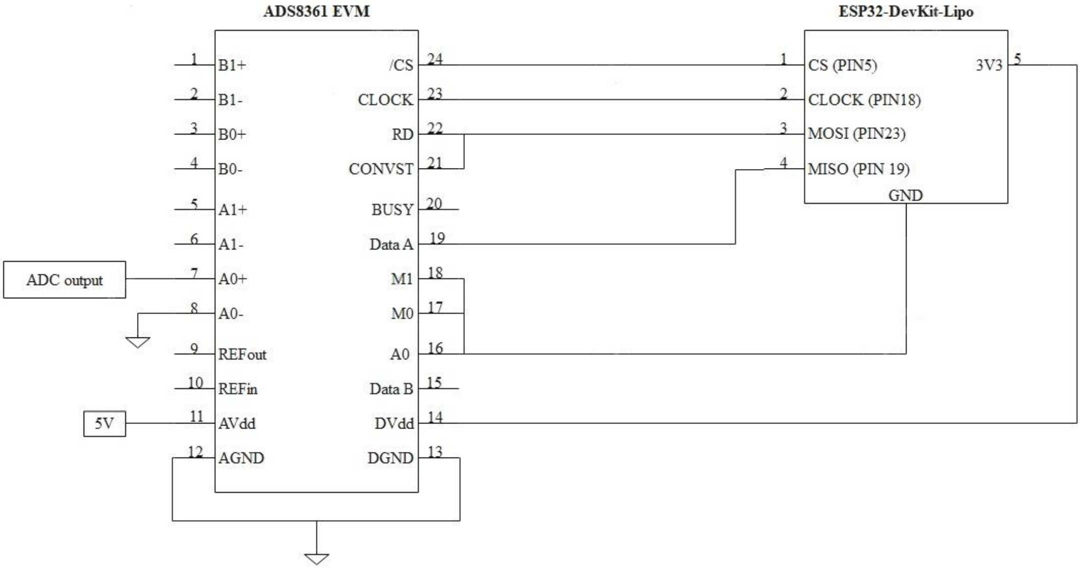


Figure 3.16 - ESP32 and ADS8361 SPI links

- The ADC input corresponds to the output of the Sallen-Key filter implemented previously. In this case, the analogue input is differential, and its amplitude will match the difference between the pins A0+ and A0- which will correspond to the filter output and ground, respectively.

- The master CLK and /CS lines are connected to their slave counterparts, and it is the first who is responsible for setting the clock frequency and for bringing the /CS to a logical ‘0’ to enable communication.

- M0, M1 and A0 are all grounded. This corresponds to the SPI Mode I.

- The master MOSI is directly connected to the CONVST and RD (EVM Pin 7) to control the start of a conversion.

- The output of the EVM (Data A) is connected to the master’s MISO, to receive the converted data.

Chapter IV will provide further information about timing, conversion control and data processing.

3.5. Bluetooth module

As it was mentioned in the project description, the ESP32-DevKit-LiPo is compliant with Bluetooth v4.2 BR/EDR end BLE specifications. Focusing on its BLE capabilities, the results will be further processed and displayed in a computer or tablet, and thus, communication between the device and server is handled based on wireless, with the aim of providing high flexibility and mobility, low consume and relatively high throughput. To reduce energy consumption while keeping a high enough throughput, communication is implemented trough BLE.

Bluetooth has been available in most smartphones and computers and most of us use it to connect to a headset and make calls using our cell phones. Over the last decade, its popularity grew insanely fast, and the emergence of new use cases made the limitations of Bluetooth apparent, mostly related to high power consumption and setup time. Bluetooth Low Energy, or BLE is fairly new, being only introduced in 2010. It shares a lot of similarities with Bluetooth classic, but it offers simplicity, lower cost, and a decrease in battery consumption. The simplicity in terms of software contributes to a high flexibility, allowing any developer with an idea and a need to exchange information to do so, as opposed to the rigid structure of Classic Bluetooth.

	BLE	Wi-Fi	Zigbee
Frequency Band	2.4 GHz	2.4GHz/5GHz	2.4GHz
Modulation	GFSK 1Mbps,2Mbps, Coded	OFDM, DSSS	DSSS
Range	< 100m to 1km+	< 300m	< 100m Point to Point, more with meshing
Topology	Point to Point, Star, Mesh	Star	Star, Mesh
Data Rate	1Mbps, 2Mbps (Bluetooth 5.0)	11Mbps, 54Mbps, 150Mbps+	250 kbps
Peak current consumption	<15mA, ~5.5mA in latest devices	60mA RX, 200mA Tx	19mA Rx, 35mA TX
Standby current	< 2µA	< 100µA	5µA

Table 3.6 - Comparison between BLE, Wi-Fi and Zigbee characteristics

The shorter range of BLE radio (<100m) has evolved in the Bluetooth 5.0 specification where a max of 20dBm output power was achieved, allowing ranges of up to 1km (LE Long Range). Long-range mode is not only useful for extending the range of a Bluetooth connection or discovery of advertisements, but it also helps in achieving more robust communication in noisy RF environments and in areas with many obstacles. In the project, the communication will be Point to Point only, which means that the device must be in range of the Tablet/computer to be reached.

The design of the BLE protocol intentionally shifts the power requirements to the Central device. Central devices are the ones that initiate communication while Peripherals are listening for information and can accept requests. There are 4 states in which a device in the BLE network can be, those are [35]:

- **Standby** - A power conservation state. The device isn't transmitting or receiving data.
- **Advertising** - A Peripheral is sending packets on the advertising channels or listening to any responses from the Central, triggered due to advertising packets. This state can be reached from the "standby state".
- **Scanning** - Used to scan for other advertising devices. This state can be reached from the "standby state".
- **Initiating** – A Central device enter this state before a connection is established. It will listen for advertisements and responds to these packets to initiate connections. This state can be reached from the "standby state".

The connection state is the final state in which the Slave (Peripheral) and Master (Central) can exchange data. When a device is connected to a Central, the GATT profile comes into play, as it is the way that two BLE devices transfer data back and forth using the Attribute Protocol (ATT) to organize data in sections called Services and Characteristics (note that a BLE peripheral can only be connected to one central at a time). GATT has two main roles: the client, who is responsible for initiating communication and sends data requests; and the server, who receives these requests and forwards data to the client. Each attribute has a unique identifier (handle), type, permissions (read, write, read, or write), encryption and value which is where the attribute holds the actual data content.

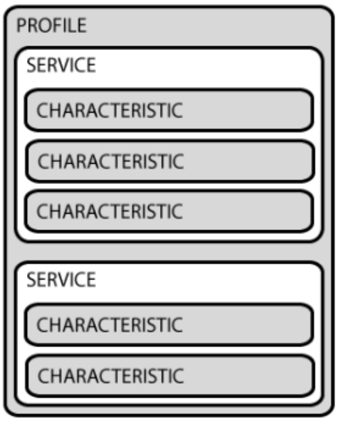


Figure 3.17 - BLE profile

Services contain chunks of data called characteristics. A characteristic can encapsulate a single data point (it may contain an array of related data). Both services and characteristics are distinguished by their UUIDs which can be either 16-bit for officially adopted BLE services or 128-bit for costume made services. A characteristic is the main point of interaction between server and client, in this case, a costume service was created with a single characteristic. Its value is set according to the ADC readings, at a desired sampling frequency for it to be read by the application. The signal processing and the BP parameters will be calculated based on this information.

3.6. Full circuit overview

With all the components set, the breadboard assembly is shown in Figure 3.18:

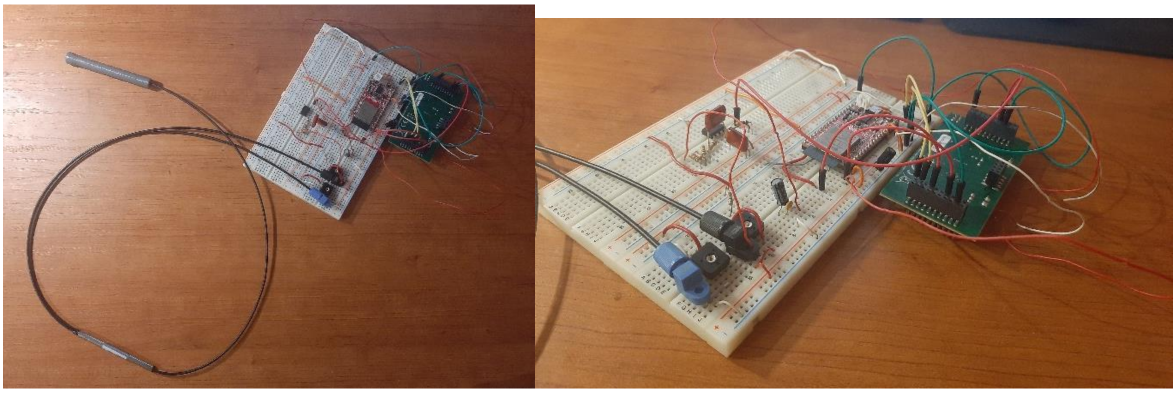


Figure 3.18 - Overview of the breadboad layout with the external ADC connections

In the figure, we can see the coupled fibre connections to the LED and phototransistor. The latter output goes through the Sallen-Key filter and the filter’s output is then fed into the external ADC. The coloured jumpers represent the SPI control links (yellow), ground links (green) and data links (orange and red). The long red and white copper cables provide the supply voltage and ground links to the ADC, respectively. The converted data is then fed into the ESP32 which is responsible for the first part of the processing and for delivering the data to the app.

Chapter IV

4. Software

The data processing will be firstly done by the microcontroller, with the aid of the Arduino IDE software. It is responsible for establishing the connection with the external ADC, via SPI, and with the PC application, via Bluetooth or USB and for defining the sampling frequency of the system. The second part of the processing consists in developing a MATLAB® application capable of processing and displaying received data to obtain the various indices essential for calculating BP.

This chapter aims to describe how the information coming from the sensor is handled by both the Arduino IDE and the MATLAB® application. On the Arduino IDE side, the focus is on the SPI and BLE protocol configuration, and on the application side, the methods used to obtain the blood pressure indexes and each one of the user interface functionalities will be described.

4.1. ESP32 configuration (Arduino IDE)

This section focuses on the code developed on the Arduino IDE open-source software, compatible with the ESP32-DevKit-LiPo microcontroller as it provides the necessary functions to implement both BLE and SPI communication protocols. The software primarily uses the C++ programming language, with the addition of unique functions such as those mentioned above.

4.1.1. SPI

First the SPI communication must be set, to make sure that the correct data is advertised by BLE. Before starting the transmission, all four SPI pins must be initiated, and CS must be defined as an output:

```
//Set default SPI GPIO’s
#define VSPI_MISO   MISO
#define VSPI_MOSI   MOSI
#define VSPI_SCLK   SCK
#define VSPI_SS     SS
…
vspi->begin(VSPI_SCLK, VSPI_MISO, VSPI_MOSI, VSPI_SS);
pinMode(VSPI_SS, OUTPUT);       //CS as output
```

To start communication, we must choose the desired SPI settings. The software provides three configurable parameters: the clock frequency, order of transmitted data (MSB first or LSB first), and the SPI mode. Also, the CS pin must be set to digital LOW to enable sensor readings:

```
static const int spiClk = 1000000; // 1 MHz
…
vspi->beginTransaction(SPISettings(spiClk, MSBFIRST, SPI_MODE0));
digitalWrite(VSPI_SS, LOW);
```

Here the clock frequency is set to 1MHz instead of the maximum 10MHz because we are using cables with considerable size for the power supplies, which for very high frequencies cause synchronisation problems. The MSB is transmitted first, and the SPI is configured to mode 0. To trigger the start of a conversion, the CONVST pin must be brought to HIGH for a minimum of 15ns. As the CONVST is connected to the MOSI line, we must send a buffer to bring CONVST to a logical ‘1’ while simultaneously giving the 20 clock cycles required for each 16-bit conversion [36]:

```
byte buffer [3];
…
buffer [0] = 0xC0;
buffer [1] = 0x00;
buffer [2] = 0x00;
vspi->transfer(buffer,3);
```

With the help of the Logic 1.2.40 software from Saleae and Az-Delivery analyser, we were able to validate the correct operation of each of the SPI control signals, illustrated in Figure 4.1:

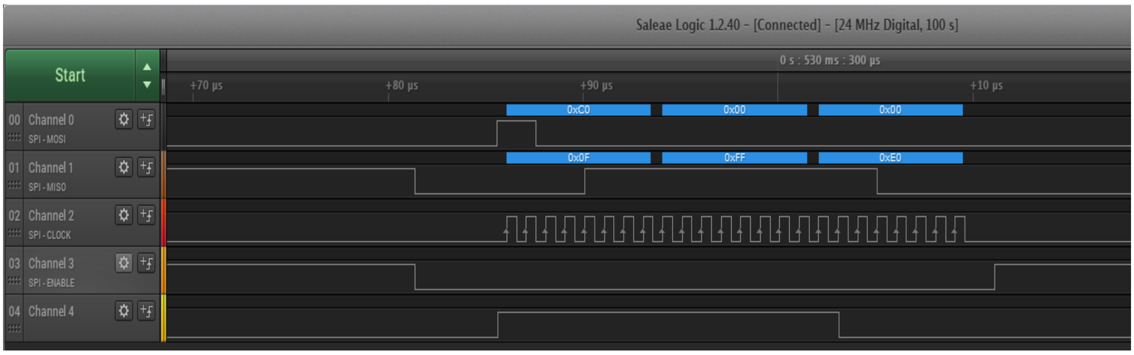


Figure 4.1 - Signalling in a single SPI 16-bit transfer

In this example, we are feeding the EVM input with a voltage of 3.27V to get an output very close to the maximum ADC resolution (65535 steps). The MOSI line represents the buffer mentioned previously, and in the analyser, we can see that the 20 clock cycles are present, as well as the CONVST transition from LOW to HIGH, starting the conversion. It is also possible to check that the ENABLE (CS) remains at LOW state during the whole conversion, as intended. Channel 4 represents the BUSY control PIN; it goes from LOW to HIGH the instant the conversion starts and goes back to LOW when it ends, and data becomes available.

In reality, we are sending 24 clock cycles instead of the 20 required. This means that the four extra bits should be discarded in the receiving side, as well as the four MSB (control bits). The logical operations responsible for assembling the final 16-bit word are the following:

```
short a = buffer[0] & 0x0f;     // 4 LSB

short b = buffer[1];            // all 8 bits

short c = buffer[2] & 0xf0;     // 4 MSB

unsigned short adcvalue = (a << 12) + (b << 4) + (c >> 4);  // final 16-bit word
```

Example from Figure 4.1:

Buffer[0] = 0000 1111   Buffer[1] = 1111 1111   Buffer[2] = 1110 0000

```
a = 0000 0000 0000 1111    <<12   a = 1111 0000 0000 0000
b = 0000 0000 1111 1111    <<4    b = 0000 1111 1111 0000
c = 0000 0000 1110 0000    >>4    c = 0000 0000 0000 1110

              1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0
              0 0 0 0 1 1 1 1 1 1 1 1 0 0 0 0
          +   0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0
      ___________________________________________

adcvalue   =    1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0
           =    65534 (decimal)
```

As the maximum would be 65535 LSB, an output of 65534 LSB for 3.27V is an accurate result. Finally, the *adcvalue* variable is transmitted to the app, via BLE.

4.1.2. Bluetooth Low Energy

The second step was to implement BLE communication between a server and a client. In this case, the ESP32 will function as the server, as it contains the data that the client wants to read. The client is the device hosting the MATLAB® App, and it is capable of scanning for nearby advertising devices, establish communication and receive the advertised data.

The BLE server was created based on the ESP32 BLE_notify example from the Arduino library. After importing the necessary libraries for the BLE capabilities, we need to define the UUID of the service and the respective characteristic, and with the aid of the https://www.uuidgenerator.net website, it was possible to assign two random UUIDs for each parameter:

```
#define SERVICE_UUID  089ce877-ad78-46f3-b749-66923e137de7"

#define CHARACTERISTIC_UUID  04f3856a-5533-4946-beed-4a1480ddaf20"
```

After that, the name “ESP32” was assigned to the device. Note that this name is the identifier that will allow the device to differentiate itself from other Bluetooth devices when we perform a scan. The following line of code sets the name of the device:

```
BLEDevice::init("ESP32");
```

Then, following the BLE_notify example, the server, service, and characteristic are created and after that both the service and the advertising can be started so that other BLE devices can scan and find this device. Because the characteristic has the Notify property, the following line, which assigns the descriptor to the characteristic, must be added to the code:

```
// Create a BLE Characteristic
        pCharacteristic = pService->createCharacteristic(
                CHARACTERISTIC_UUID,
                BLECharacteristic::PROPERTY_NOTIFY
              );


  // Create a BLE Descriptor
        pCharacteristic->addDescriptor(new BLE2902());
```

With the setup done, the last thing to do is to set the characteristic value to the previously determined ADC output. This line of code must be written inside a loop with a predefined frequency, so that it is updated at the system sampling frequency:

```
pCharacteristic->setValue(value);
pCharacteristic->notify();
```

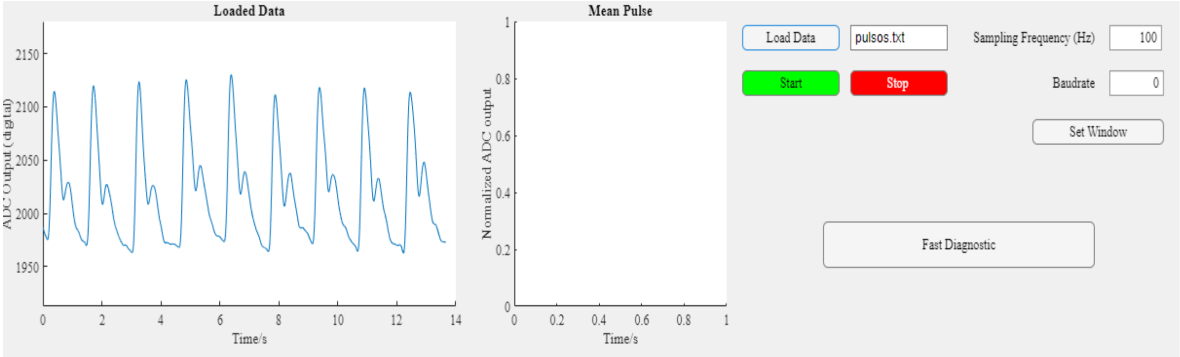
With all set and done, the system is now ready to move forward with data processing and BP parameter calculations. The full code can be found in the following link (Final_v4.ino).

4.2. Matlab App

This is one of the most important components of this project, as it is the part where all the previous work converges, and because it is responsible for displaying the final result. Besides, it will be the only point of interaction with the user, apart from the sensor itself. There are three different ways that the app can acquire data: read from a .txt file, live plotting from USB connection and live data from BLE.

- **Read from .txt file**

To avoid relying solely on our sensor readings, the application is prepared to analyse information from other types of sensors capable of generating a waveform similar to the one studied in this dissertation and that was saved in a text file in a single column format (Figure 4.2):



pulsos.txt

Sampling Frequency (Hz) 100

Baudrate 0

Figure 4.2 - Load Data button pushed

The image shown in the Figure 4.2 ilustrates a .txt file reading. This type of reading requires the exact file name + extension and the respective sampling frequency for the data to be plotted. Before being plotted, the data is filtered using a second order Savitzky-Golay

FIR filter. This type of filter, compared to the moving average, has shown better results for noisier samples as it preserves data features such as peak height and width, which are ususally atteuated by the moving average method. The following graphs show the comparison between both filters:

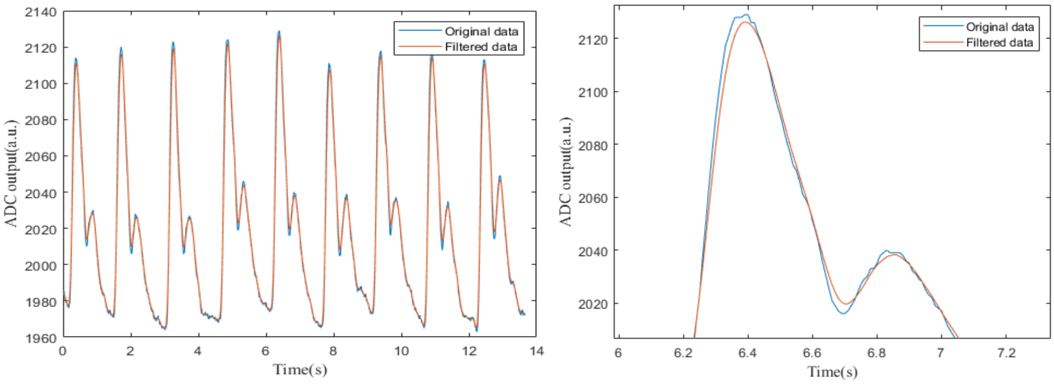


Figure 4.3 - Moving average method

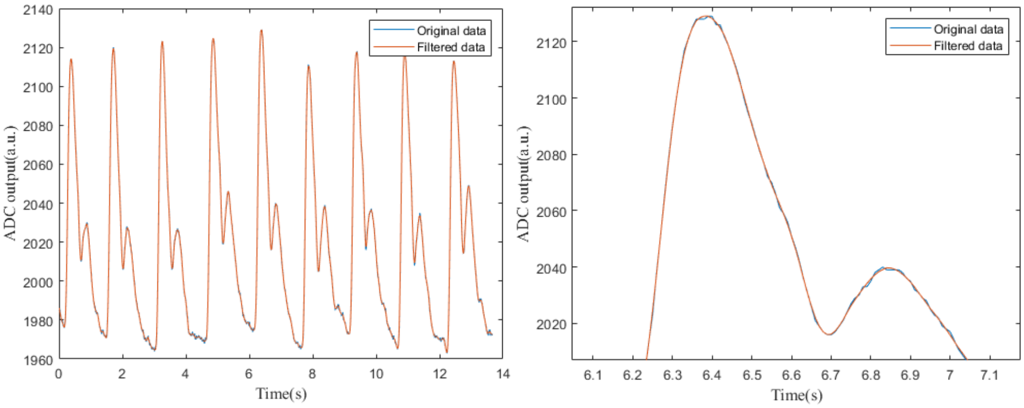


Figure 4.4 - Savitzky-Golay filter

After typing the filename and defining the sampling frequency, the **Load Data** button can be pushed to generate the plotted data.

- **Serial Port live plotting**

When the data comes from an USB connection, the application can display a live plot of the real time sensor readings (the application is capable of distinguish whether the sensor is connected by USB or Bluetooth, prioritizing the first). To enable this type of reading, the sampling frequency and baudrate must be defined, the same way as the previous method. MATLAB® uses the baudrate and the port value (ex. ‘COM3’) to create the serial client for communication with the serial port. Then, the **Start** button can be pushed to trigger a pop-up graph that illustrates a live feed of the data being drawn on it. For this we create an *animatedline* that enables the plot and then use the addpoints() function to add the real time sensor readings to it. This process proved to be most suitable for the app, as it provides the live feed preciously mentioned and for being less resource intensive. When the user feels like there are enough samples in the graph, the **Stop** can be used to save that data in vector, filter it (using the Savitzky-Golay FIR filter), and plot the filtered data in a graph. Figure 4.5 illustrates an example where the plot is generated by USB connection:

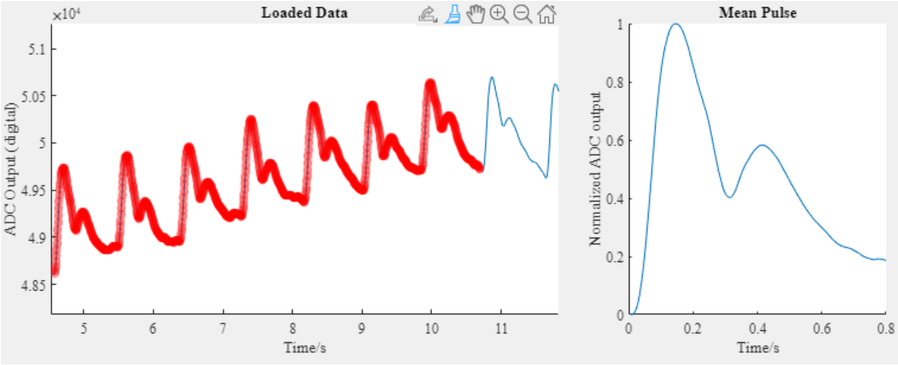


Figure 4.5 - Serial port live plot

- **BLE live readings**

For BLE, two objects are created, one corresponding to the device and the other to the characteristic containing the ADC output value. The most efficient way to read the advertised data with the desired sampling frequency was to create a call-back function named **displayCharacteristicData(obj)** where the input argument is the characteristic object created earlier. This function will read the BLE device data, convert it to the correct ADC output value and store the reading in a vector. In this case, the *animatedline* function is not supported. The user can only verify the sensor reading after the Stop button is pushed, and the data becomes available in the interface plot. An alternative method like the one where we use the *animatedline* was tested but the application could not keep up with the desired sample frequency, and therefore we prefer to maintain the integrity of the measurements by sacrificing the live plot functionality. The same Start and Stop buttons can be used to initiate and end the measurements, respectively. Figure 4.6 illustrates an example where the plotted data is acquired by BLE:

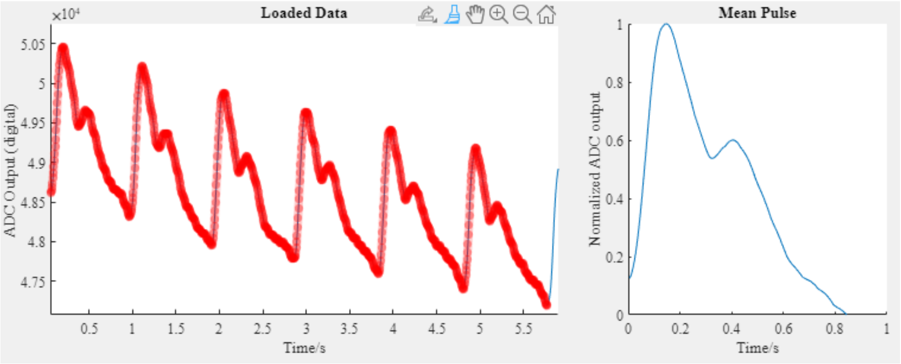


Figure 4.6 - BLE live plot

After obtaining the desired graph, it will be necessary to choose an arbitrary set of consecutive pulses to calculate its average. It is based on this average pulse that all the indexes will be calculated, making the determination of this average pulse a decisive factor for the result. The MATLAB® Brushing Tool is used to select data points from the graph interface inside the app and automatically stores that data in a separate variable. This project utilizes this tool to highlight the pulses from which the average will be calculated, as shown in Figure 4.7:

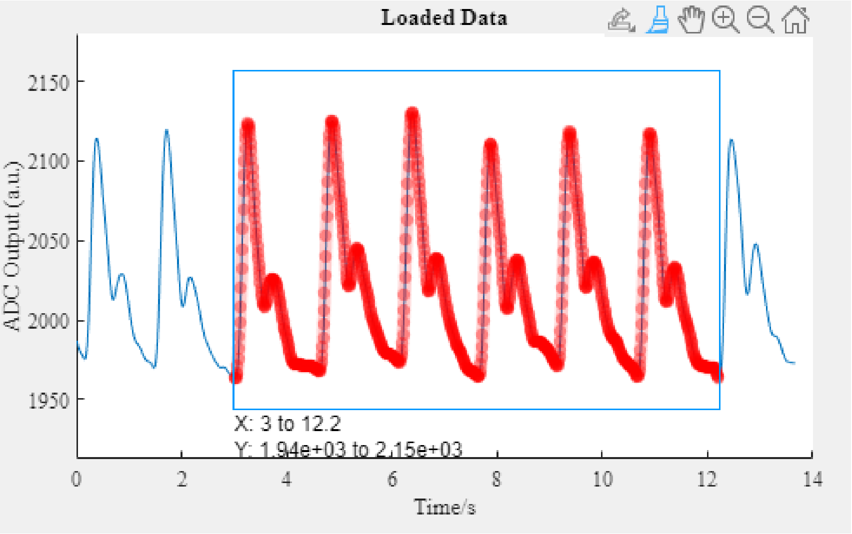
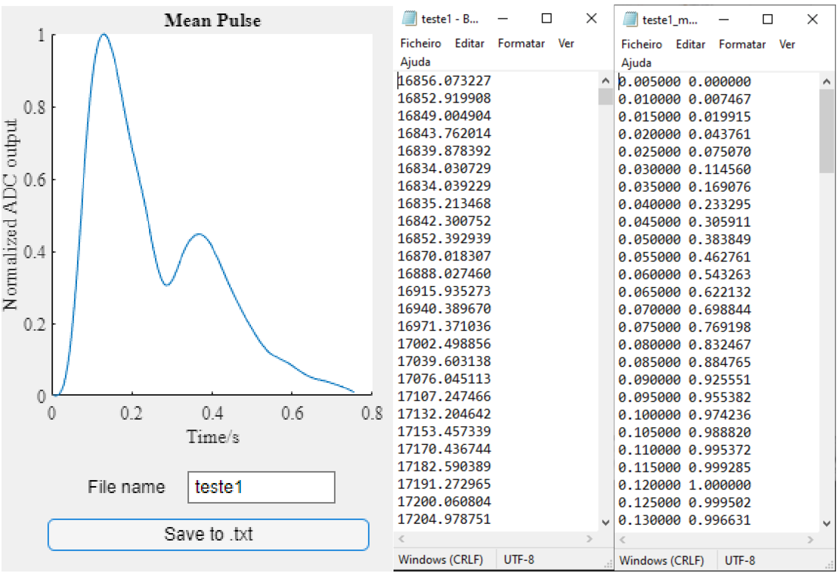


Figure 4.7 - Data selection using the Brush Tool

With the data selected, the average pulse can be produced by pushing the **Set Window** button. This also triggers a smaller graph containing the normalized waveform that represents the mean pulse. The app has also the option to save the highlighted data to a .txt file named according to a user input. After pushing the Save to .txt button the app creates two text files: one containing the raw brushed data and the other containing two columns, containing the x and y coordinates of the mean pulse as shown in Figure 4.8:



```
16856.073227
16852.919908
16849.004904
16843.762014
16839.878392
16834.030729
16834.039229
16835.213468
16842.300752
16852.392939
16870.018307
16888.027460
16915.935273
16940.389670
16971.371036
17002.498856
17039.603138
17076.045113
17107.247466
17132.204642
17153.457339
17170.436744
17182.590389
17191.272965
17200.060804
17204.978751
```

```
0.005000 0.000000
0.010000 0.007467
0.015000 0.019915
0.020000 0.043761
0.025000 0.075070
0.030000 0.114560
0.035000 0.169076
0.040000 0.233295
0.045000 0.305911
0.050000 0.383849
0.055000 0.462761
0.060000 0.543263
0.065000 0.622132
0.070000 0.698844
0.075000 0.769198
0.080000 0.832467
0.085000 0.884765
0.090000 0.925551
0.095000 0.955382
0.100000 0.974236
0.105000 0.988820
0.110000 0.995372
0.115000 0.999285
0.120000 1.000000
0.125000 0.999502
0.130000 0.996631
```

Figure 4.8 – Data saved from Figure 4.7

- **Average pulse calculation**

The first step when it comes to make an average from a set of pulses is segmentation. By isolating each individual pulse, it is possible to overlap them to calculate a mean pulse. This process looks simple, but for the segmentation to be precise, each pulse must be isolated with a very low error margin. There are many different procedures that use wave root estimation based on the signal first and second derivatives, but they proved to be inaccurate when applied to noisier signals. Therefore, with the aid of a MATLAB® add-on named "Pulse Waveform Delineator" (PWD), it's possible to get the SP, DN and the wave's foot points(F) for each one of the pulses. The PWD searches for the pairs of inflexion points and zero-crossing points and uses the combinatorial amplitude and interval criteria to select the onset and SP. Once a new beat is settled, the delineator seeks the derivative backward to locate the dicrotic notch in the preceding beat. In a nutshell, the delineator is based on the combinatorial analysis of arterial blood pressure waveforms and their derivatives (Figure 4.9) [37].

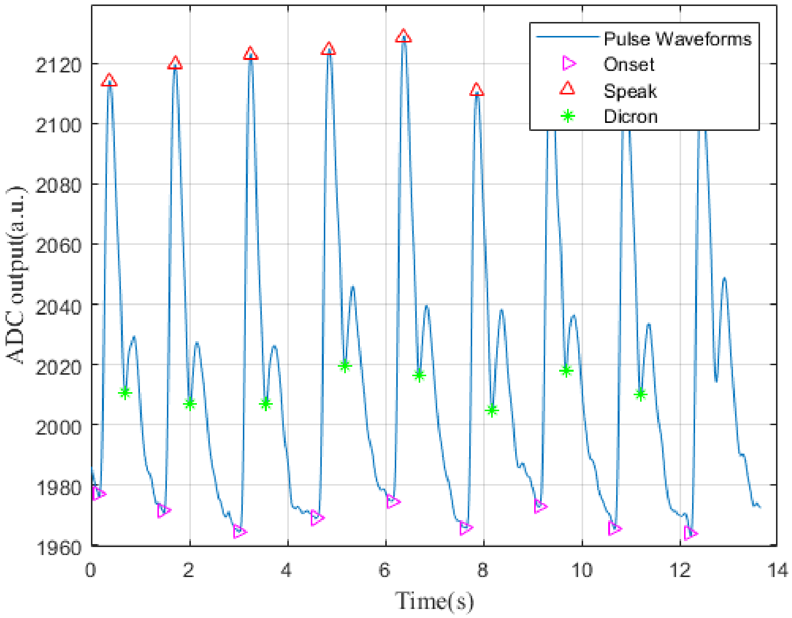


Figure 4.9 - Main points acquired from the PWD add-on delineator() function

Each pulse is isolated based on the vector containing every Onset point, and the pulse lengths are matched by discarding some of the samples. Then proceeds the overlapping phase and finally the average pulse is calculated. The normalised result can be observed in the smaller graph in the right of the application on Figure 4.10:

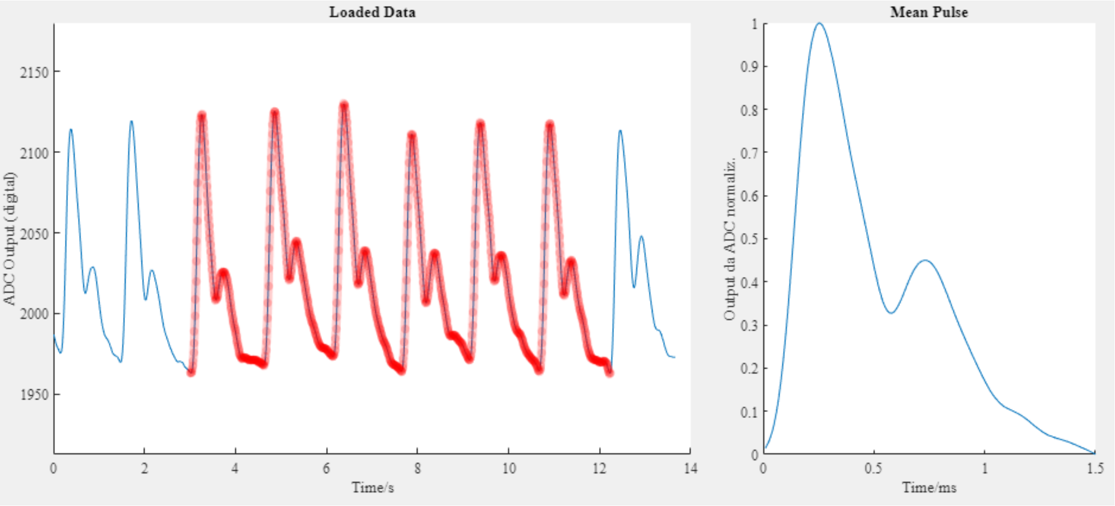
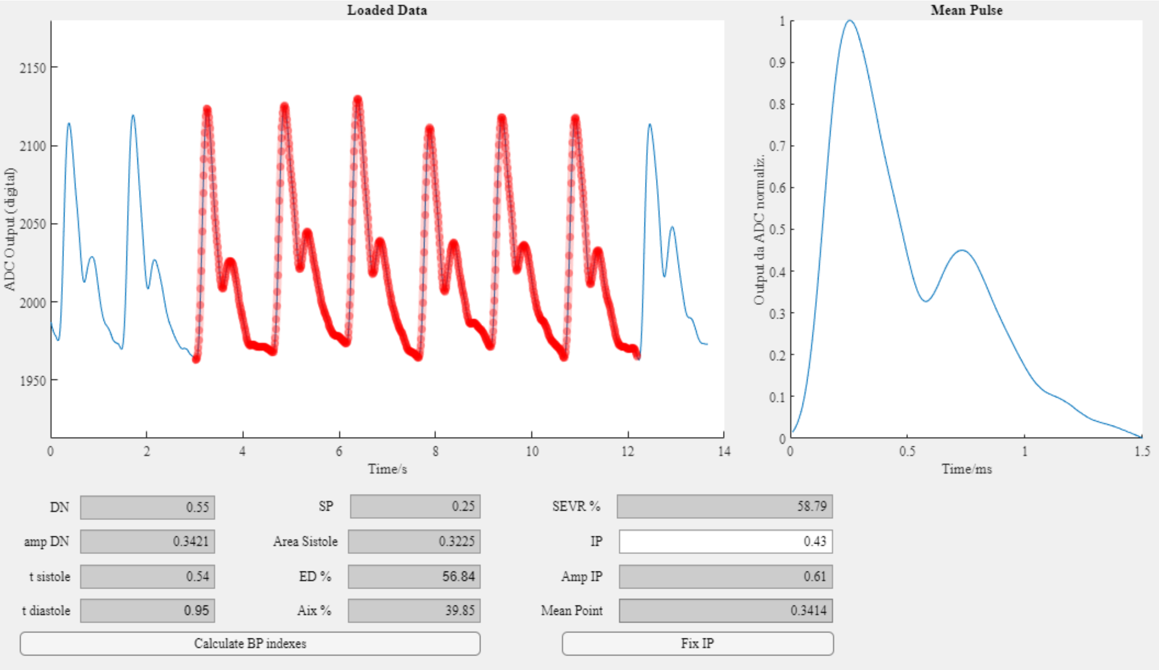


Figure 4.10 - Mean pulse calculation

- **Index and BP parameter calculation**

Having the normalized pulse, the system is now ready to calculate every index that is included in the project, related to cardiovascular prognostic. The **Calculate BP indexes** button triggers the algorithm responsible for the calculation of every index (Figure 4.11).



DN 0.55
amp DN 0.3421
t sistole 0.54
t diastole 0.95
SP 0.25
Area Sistole 0.3225
ED % 56.84
Aix % 39.85
SEVR % 58.79
IP 0.43
Amp IP 0.61
Mean Point 0.3414

Figure 4.11 - BP parameter calculation

The DN can be obtained by finding the point located between 20% to 50% of the total pulse, where the second derivative of said pulse assumes its maximum value. This point will correspond to the start of an upwards concavity in that portion of the pulse, hence, the dicrotic notch; both the diastole and systole periods depend on the DN, as it dictates where the systolic wave ends and the diastolic wave begins; SP coordinates correspond to the waveform's maximum point; MP matches the average of the pulse wave amplitude; SEVR, ED and Aix calculations have already been addressed in Chapter II.

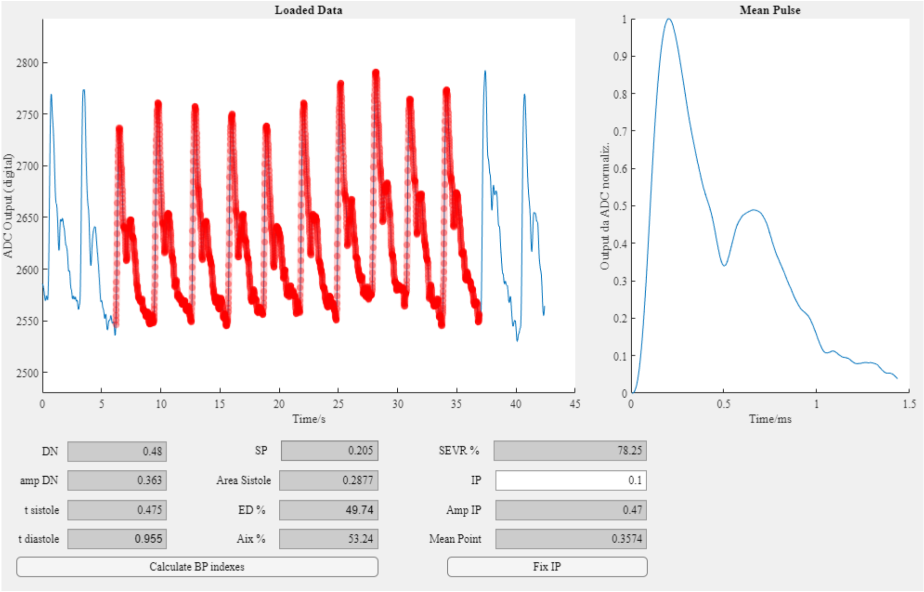
The last one of this indexes is the IP, but its calculation proved to be more complex than expected. Simply put, the IP corresponds to the point where the curve goes from concave up to concave down (or vice versa). As mentioned in before, the IP can represent both the FW or RW, depending on the moment it happens: if the IP precedes the SP, it corresponds to the FW and if it appears after the SP it corresponds to the RW. As both these points appear early in the pulse wave, only 20% of the pulse is analysed, starting from F. The second derivative of the pulse data can give information about the concavity of the function representing that data: if it is a positive value the concavity faces upwards; if it is negative, the concavity faces downwards; if the second derivative equals zero, it is a possible IP. Based on this information, the IP are calculated following three steps:

Step 1: Find the second derivative of the pulse wave.

Step 2: Find the points where the second derivative is zero.

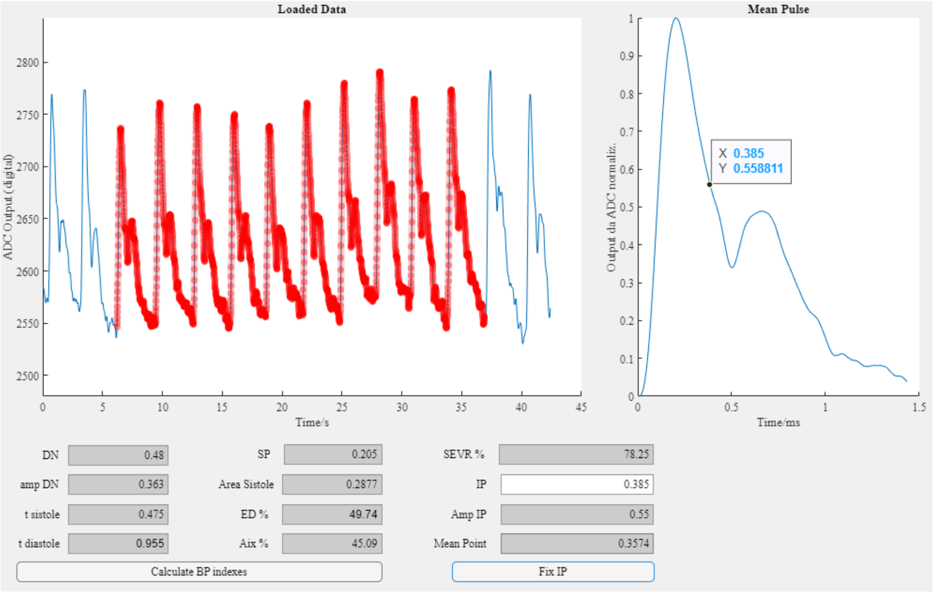
Step 3: Check whether adjacent points to the ones found in Step 2 have opposite signs.

If Step 3 checks out, then the point corresponds to the IP, otherwise, Step 3 is repeated for the remaining points, in order of occurrence, until the IP is found. This method proved to have the best results, but is still not 100% accurate, and for that reason, the application has an alternative button to set the IP. By changing the value inside the IP box to the correct one and pressing the **Fix IP** button, all the indexes that depend on the IP value will be updated (Figure 4.12).



DN 0.48
SP 0.205
SEVR % 78.25
amp DN 0.363
Area Sistole 0.2877
IP 0.1
t sistole 0.475
ED % 49.74
Amp IP 0.47
t diastole 0.955
Aix % 53.24
Mean Point 0.3574

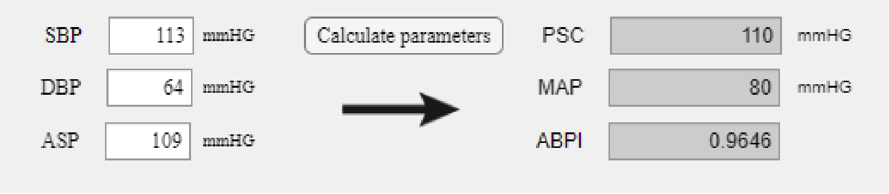
Figure 4.12a - Wrong IP calculation



DN 0.48
SP 0.205
SEVR % 78.25
amp DN 0.363
Area Sistole 0.2877
IP 0.385
t sistole 0.475
ED % 49.74
Amp IP 0.55
t diastole 0.955
Aix % 45.09
Mean Point 0.3574

Figure 4.12b - Fixed IP

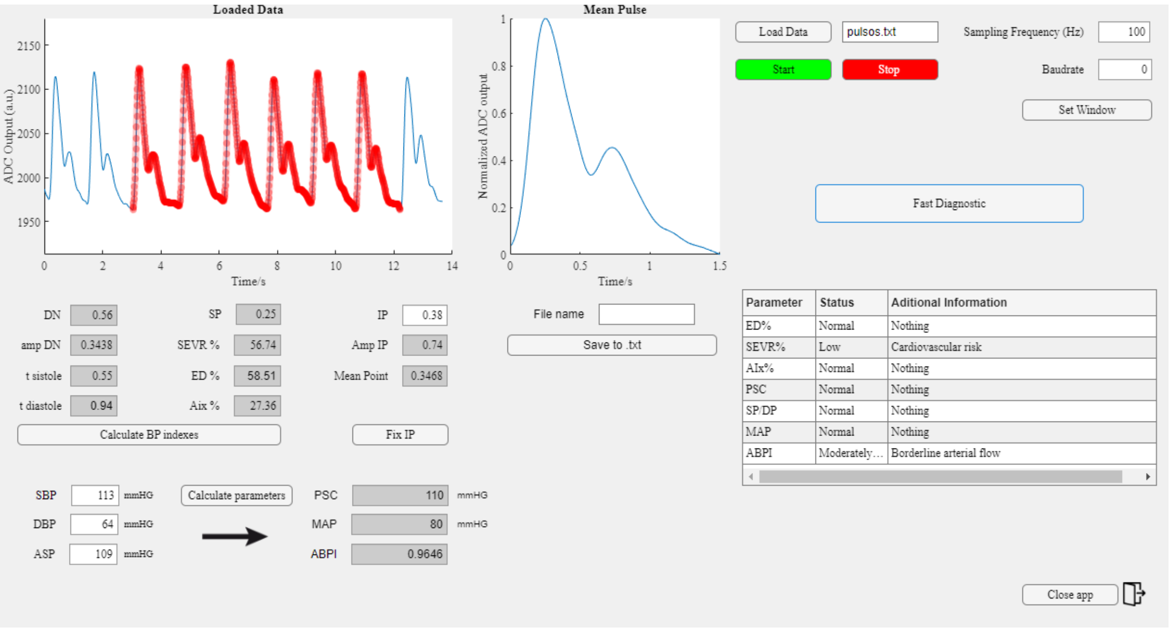
Lastly, the app also has a dedicated interface for calculating the PSC, MAP and ABPI (Figure 4.13). To do this, the algorithm makes use of the previously calculated MP and the SBP, DBP and ASP values that must be manually input by the user, as our system doesn't have the necessary components for the acquisition of the last three.



SBP 113 mmHG
DBP 64 mmHG
ASP 109 mmHG
PSC 110 mmHG
MAP 80 mmHG
ABPI 0.9646

Figure 4.13 - Extra BP parameter calculation

Once all the indices and additional parameters have been calculated, the app is prepared to make a quick diagnostic, able to inform the user about his or her cardiovascular health. These results are obtained in accordance with various scientific articles and research papers [38] [39] [40] [41] [42] [43]. The **Fast Diagnostic** button triggers a table pop-up that has information about the status of each parameter, as well as some additional information about the resulting values, as shown in Figure 4.14:



pulsos.txt
Sampling Frequency (Hz) 100
Baudrate 0

DN 0.56
amp DN 0.3438
t sistole 0.55
t diastole 0.94
SP 0.25
SEVR % 56.74
ED % 58.51
Aix % 27.36
IP 0.38
Amp IP 0.74
Mean Point 0.3468

File name

SBP 113 mmHG
DBP 64 mmHG
ASP 109 mmHG
PSC 110 mmHG
MAP 80 mmHG
ABPI 0.9646

Parameter	Status	Aditional Information
ED%	Normal	Nothing
SEVR%	Low	Cardiovascular risk
AIx%	Normal	Nothing
PSC	Normal	Nothing
SP/DP	Normal	Nothing
MAP	Normal	Nothing
ABPI	Moderately…	Borderline arterial flow

Figure 4.14 - Fast diagnostic example

The link for the application installer can be found here. It also includes the folder containing all additional files and prior versions of the application.

Chapter V

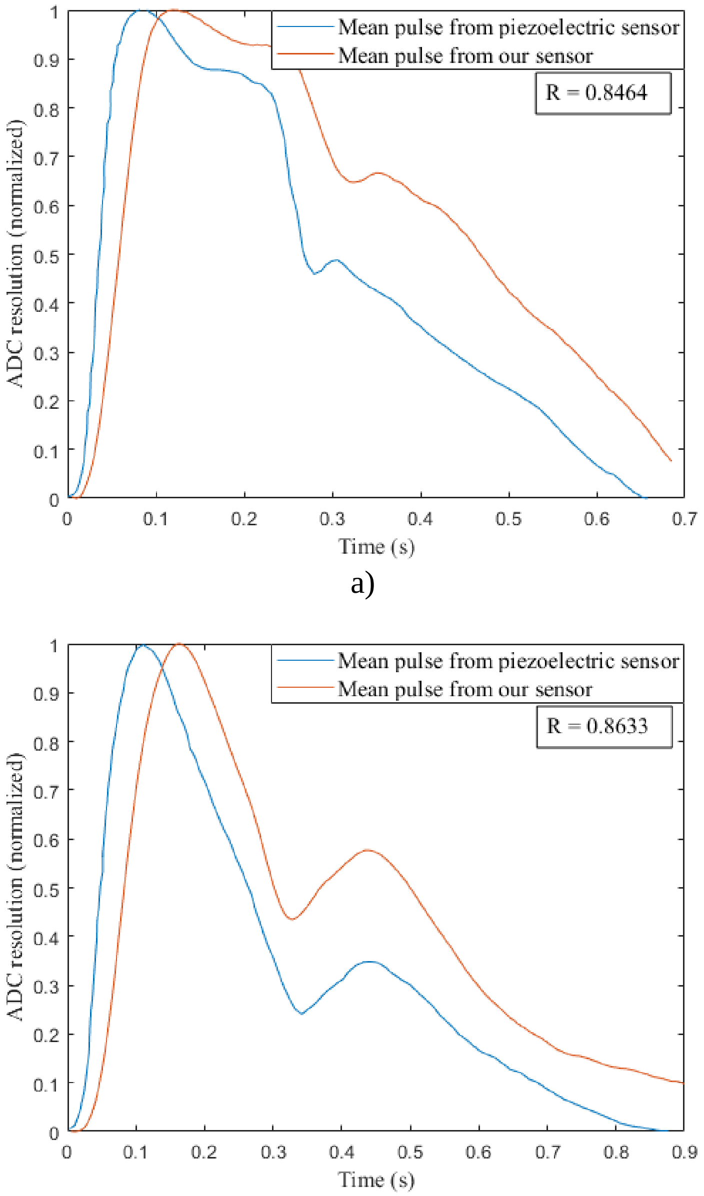
5. Experimental results and analysis

The fifth chapter aims to validate the waveform analysis, the parameter calculation and the data acquired. Here it will be demonstrated that the system works for different types of waves, its performance will be tested against other similar methods and a comparison between BLE, and Serial acquisition will be presented. It includes Pearson correlation, mean difference error and Bland-Altman plots to analyse the statistical differences between methods.

5.1. Performance evaluation

In this first section, we put our system face to face with the Complior Analyse® device (Alam Medical) available at the premises of Hospital de Aveiro, to evaluate the veracity of some of its results. Taken from the Complior website (http://www.complior.com/products#introcomplior) the Complior Analyse® measures carotid pressure and central pressure during pulse wave velocity assessments and it uses piezoelectric sensors in a coin-like shape probe, with an acquisition frequency of 1000 Hz [44].

The tests were performed on two subjects: each one was lying down, with the body relaxed and the SBP and DBP were measured following the Complior measurement protocol; then, the tests started with the Complior device followed by our system. In both measurements, the subjects remained immobile, and the second measurement starts as soon as the first one ends. Both methods preform data acquisition based on an average calculation of successive pulses. The resulting pulse, obtained for each one of the subjects, as well as the correlation analysis can be verified in Figure 5.1 below (The full analysis performed by the Complior Analyse® are in this document attachments, Figures 1 and 2):



a)

b)

Figure 5.1 – a) Subject 1 and b) Subject 2 average pulse comparison and correlation coefficient

The piezoelectric sensor pulses seen on Figure 5.1 was obtained by digitizing the pulse waves of Figures 1 and 2 of the annexes through the MATLAB®, GRABIT add-on. As this method is very user dependent, inevitable errors are introduced in the resulting waveform, which can justify some of the differences in the pulse duration and amplitude.

In both cases, the waveforms are very resembling and from the point-to-point correlation between the two waveforms for subjects 1 and 2 we've got a correlation coefficient of 0.8464 and 0.8633 respectively. As for the calculated parameters, Table 5.1 shows the difference between both methods:

Parameters	Subject 1		Difference (%)	Subject 2		Difference (%)
	Comp.	Main		Comp.	Main	
MAP (mmHg)	96	96	**0.00**	86	86	**0.00**
PSC (mmHg)	105	103	**1.92**	107	118	**9.78**
IP (s)	0.177	0.130	**30.62**	0.250	0.215	**15.05**
SP (s)	0.091	0.120	**27.49**	0.131	0.165	**22.97**
T systole (s)	0.272	0.295	**8.11**	0.337	0.325	**3.63**
T diastole (s)	0.403	0.385	**4.57**	0.614	0.638	**3.83**

Table 5.1 – Key parameter, MAP and PSC results for both methods

Regarding the instants when the key points were detected, all differences are between 12 ms (3.63%) and 47 ms (30.62%). The MAP calculations were identical in both systems, but for the PSC a difference of 2 mmHg was obtained for subject 1 and 11 mmHg for subject 2.

5.2. Algorithm validation

Here, the same dataset is used as the object of study in two different analysis systems: the one developed in this project, and a reference system developed by a research group from the University of Aveiro, successfully tested in the medical field.

The reference study aimed to analyse the different wave types using an algorithm developed by the authors, which is able to determine several of the parameters discussed in this dissertation, using a FOS based on intensity, similar to the one developed in this project. They've gathered data, including SBP and DBP measurements for MAP and PSC calculation, of 12 subjects, from 26 to 32 years old, all presenting some stage of hypertension. Table 5.1 shows the results of both methods for each of the main BP indexes:

Parameters:		**ED%**	**Aix%**	**SEVR %**	**SP (s)**	**DN (A/s)**	**MAP**	**PSC**
Subject 1	Main	58.80	52.29	78.24	0.220	0.72/0.33	90	114
	Ref.	60.95	75.40	60.26	0.218	0.68/0.34	90	121
Subject 2	Main	79.59	0.84	58.61	0.152	0.56/0.31	82	98
	Ref.	68.27	-3.22	54.96	0.120	0.58/0.28	82	99
Subject 3	Main	68.77	19.50	61.06	0.252	0.65/0.37	99	124

	Ref.	60.75	16.19	78.13	0.253	0.69/0.35	99	122
Subject 4	Main	61.47	19.8	80.02	0.180	0.68/0.29	105	130
	Ref.	57.38	-1.19	70.12	0.160	0.66/0.27	105	133
Subject 5	Main	63.75	35.78	77.52	0.243	0.76/0.32	115	139
	Ref.	60.73	6.72	70.19	0.212	0.74/0.29	115	140
Subject 6	Main	48.34	15.38	93.30	0.115	0.51/0.32	83	110
	Ref.	42.86	-24.99	84.78	0.086	0.49/0.29	83	114
Subject 7	Main	56.01	4.70	72.46	0.210	0.66/0.33	98	117
	Ref.	42.51	1.49	78.42	0.190	0.66/0.28	98	119
Subject 8	Main	44.88	6.39	129.3	0.240	0.81/0.3	111	127
	Ref.	42.99	8.44	111.01	0.210	0.77/0.28	111	129
Subject 9	Main	55.67	-4.04	73.96	0.140	0.63/0.31	83	108
	Ref.	59.14	2.93	90.34	0.160	0.67/0.33	83	103
Subject 10	Main	43.37	31.07	111.66	0.200	0.73/0.28	101	128
	Ref.	51.21	50.08	131.60	0.230	0.82/0.3	101	123
Subject 11	Main	44.08	39.78	112.18	0.230	0.79/0.29	130	157
	Ref.	59.33	40.45	108.1	0.250	0.81/0.32	130	150
Subject 12	Main	51.90	20.36	78.85	0.210	0.69/0.29	116	143
	Ref.	63.81	28.75	84.86	0.250	0.69/0.34	116	139

Table 5.2 – Comparison of PPG and BP parameters between the reference algorithm (Ref.) and the algorithm developed in this project (Main)

The correlation and Bland-Altman plots displayed in Figure 5.1 are used to evaluate the accuracy of results:

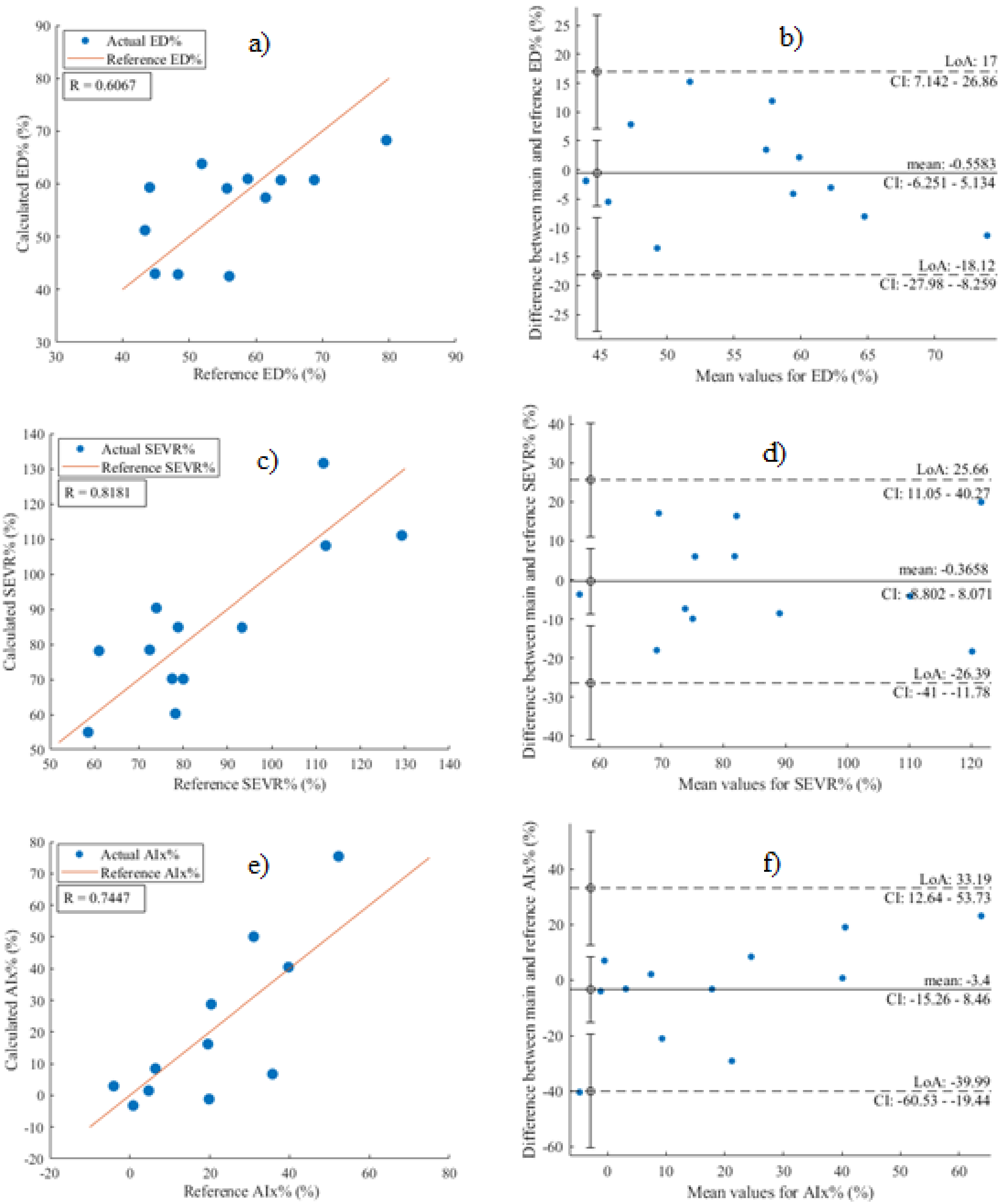


Figure 5.2 – Experimental results. Correlation plots of (a)ED%, (c)SEVR%, (e)Aix% and Bland-Altman plots of (b)ED%, (d)SEVR%, (f)Aix%

Focusing on the correlation plots for the three parameters, we were able to achieve a correlation coefficient of 0.6067, 0.8181, and 0.7447 for ED%, SEVR% and Aix% respectively. The absolute value of this coefficient gives information about the magnitude of the correlation, and the greater it is, the stronger the correlation (its values range from -1 to 1). The correlation analysis results tell us that the SEVR% and Aix% have a very strong association with the reference study, which is optimal, and the ED%, despite having a lower coefficient (r = 0.6067), also shows a strong association with the reference.

However, the strong correlation between both methods doesn't give information about an agreement between them, as a perfect agreement would require that all the paired data points coincided with the diagonal line. Since this is not case, an additional analysis with the help of Bland-Altman plots was conducted. Graphs b), d), and f) represent the Bland-Altman plot for ED%, SEVR% and Aix% respectively. The x-axis represents the average between both methods (for each parameter); the y-axis represents the difference between the measured values (for each parameter); the horizontal dotted lines are the limits of agreement, or the 95% confidence interval and the central horizontal line represents the mean of differences. In graphs b) and d) we've got a mean of differences of -0.5583 and -0.3658 for ED and SEVR respectively, and all the data point fall between the limits of agreement, as intended. For Aix the mean of differences is a little higher (-3.4) ,and 11 of the 12 points of data fall between the limits of agreement, with the last one being very close to the lower limit, indicating satisfying results.

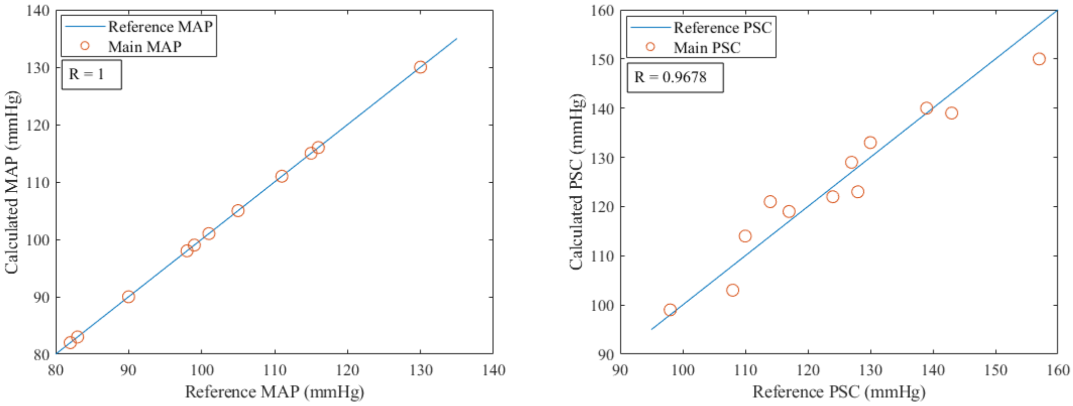


Figure 5.3 – Results for MAP (left) and PSC (right) calculations

A simpler analysis can be made for MAP calculations, as it only requires a user input, so all the results correspond to the reference. But for PSC, the normalized mean pulse amplitude directly influences its value, and taking this into account, a correlation factor of 0.9678 was obtained.

Subject	DN coordinates						SP coordinates		
	Normalized amplitude (a.u.)			Time (s)			Time (s)		
	Main	Ref.	Diff.	Main	Ref.	Diff.	Main	Ref.	Diff.
1	0.72	0.68	**0.04**	0.33	0.34	**-0.01**	0.220	0.218	**0.002**
2	0.56	0.58	**-0.02**	0.31	0.28	**0.03**	0.152	0.120	**0.032**
3	0.65	0.69	**-0.04**	0.37	0.35	**0.02**	0.252	0.253	**-0.001**
4	0.68	0.66	**0.02**	0.29	0.27	**0.02**	0.180	0.160	**0.020**
5	0.76	0.74	**0.02**	0.32	0.29	**0.03**	0.243	0.212	**0.031**
6	0.51	0.49	**0.02**	0.32	0.29	**0.03**	0.115	0.086	**0.029**
7	0.66	0.66	**0**	0.33	0.28	**0.05**	0.210	0.190	**0.020**
8	0.81	0.77	**0.04**	0.3	0.28	**0.02**	0.240	0.210	**0.030**

9	0.63	0.67	**-0.04**	0.31	0.33	**-0.02**	0.140	0.160	**-0.020**
10	0.73	0.82	**-0.09**	0.28	0.3	**-0.02**	0.200	0.230	**-0.030**
11	0.79	0.81	**-0.02**	0.29	0.32	**-0.03**	0.230	0.250	**-0.020**
12	0.69	0.69	**0**	0.29	0.34	**-0.05**	0.210	0.250	**-0.040**

Table 5.3 - Difference of DN and SP measurements based on the two methods

Lastly, analysing the coordinates for the DN calculated by our method, in comparison to the reference, presented in Table 5.2, we've have achieved a value of -0.0058±0.0382 a.u. for the mean difference error for amplitude (A) and 0.0058±0.0311 s for the detection instant (s). The SP detection had a lower mean difference error of 0.0044±0.0263 s.

5.3. Serial Port vs BLE

In this last section, we are comparing the measurements taken by the two data acquisition methods developed in this dissertation, Serial live plot and BLE. For that, we have made 10 successive analyses from a single person, 5 from the serial port and 5 from BLE, and we calculated the mean pulse and the BP parameters for each one of the analyses. The subject was lying down, in a relaxed posture and each new analysis started 2 seconds after the previous one finished. There was a longer pause where the system was setup to change the communication protocol to BLE, that took approximately 10 seconds. After that, the system resumed the previous behaviour. The acquired data and the mean pulse can be checked in the attachments of this dissertation, Figures 3 and 4, and the resulting BP parameters are shown in Table 5.3:

		Sample 1	**Sample 2**	**Sample 3**	**Sample 4**	**Sample 5**	**Mean**
T systole (s)	Serial	0.315	0.280	0.300	0.325	0.300	**0.304±0.017**
	BLE	0.290	0.310	0.300	0.305	0.310	**0.303±0.008**
T diastole (s)	Serial	0.705	0.540	0.605	0.585	0.585	**0.604±0.061**
	BLE	0.605	0.570	0.655	0.575	0.580	**0.597±0.035**
SP (s)	Serial	0.155	0.145	0.165	0.165	0.150	**0.156±0.009**
	BLE	0.160	0.160	0.160	0.160	0.125	**0.153±0.016**
ED (%)	Serial	44.68	51.85	49.59	55.56	51.28	**50.59±3.96**
	BLE	47.93	54.39	45.80	53.04	53.45	**50.92±3.81**
Aix (%)	Serial	8.184	9.886	14.400	12.420	9.217	**10.82±2.538**

	BLE	6.613	10.070	2.240	7.350	7.249	**6.70±2.828**
SEVR (%)	Serial	121.50	98.30	98.64	97.64	100.30	**103.28±10.23**
	BLE	98.15	103.6	118.2	92.47	94.61	**101.40±10.30**
IP (s/A)	Serial	0.195/ 0.92	0.185/ 0.90	0.215/ 0.86	0.225/ 0.88	0.195/ 0.91	**0.203/0.89 ±0.016/0.02**
	BLE	0.190/ 0.92	0.210/ 0.90	0.175/ 0.98	0.195/ 0.93	0.185/ 0.93	**0.194/0.93 ±0.013/0.03**
DN (s/A)	Serial	0.32/ 0.564	0.29/ 0.494	0.30/ 0.527	0.33/ 0.596	0.30/ 0.472	**0.31/0.530 ±0.02/0.050**
	BLE	0.29/ 0.605	0.32/ 0.574	0.30/ 0.577	0.31/ 0.532	0.32/ 0.538	**0.31/0.565 ±0.01/0.030**
Mean point (A)	Serial	0.4559	0.4403	0.4233	0.4628	0.4428	**0.4450 ±0.0153**
	BLE	0.4112	0.4755	0.4565	0.4317	0.4663	**0.4482 ±0.0264**

Table 5.4 - Comparison of the BP parameters from the two data acquisition methods (Serial vs BLE)

The results proved to be quite satisfactory, as the overall difference between the means of the two methods for each parameter was generally very low: the systole and diastole periods have a difference of 1 ms and 7 ms respectively, which are low enough values considering the number of samples; the instant where the SP was detected has a difference of 3 ms and the DN has a normalized amplitude difference of 0.035 a.u; the IP coordinates are also very close to one another, with differences of 9 ms for time and 0.04 for amplitude, which is very relieving as the BLE waveforms are not as clear as the ones taken from Serial connection (the IP coordinates were determined solely by the developed algorithm, without using the Fix IP functionality); the low differences in T systole and T diastole, cause the ED% calculations to also have a very low difference, of only 0.33%; the SEVR% had a difference of 1.88% and the Aix% had a difference of 4.12%. These two parameters are highly dependent on the waveform clarity and the IP calculations, and for that reason the difference between the two methods has a greater influence. Regardless, the difference in means proved to be satisfactory considering the number of samples for each method.

Chapter VI

6. Conclusion

This last chapter aims to make a final overview of the work carried out, taking into account the objectives proposed initially. The analysis is focused in 3 main parts: the data acquisition part, the communications part, and finally, the data processing part which involves the MATLAB® application and the result analysis. Finally, possible future work is indicated, considering some of the unfinished tasks mentioned throughout the dissertation.

The main objective was to build a device from scratch, composed by a LED, photodiode, and an optical fibre, capable of detecting intensity variations of the light reflected by the skin due to blood flow fluctuations inside the vessels. After analysing the transistor output, it was decided to apply a filter to eliminate the additional noise from the signal. After testing several different filter configurations, we decided to implement a Sallen-Key filter, capable of attenuating the noise at high frequencies with great efficiency while maintaining the DC component of the signal. To achieve high sampling rates, the ADS8361 (ADS 8361 EVM) ADC was added to the project, as the ESP32-DevKit-LiPo internal ADC does not meet the project requirements. The communication with the external ADC via SPI was carried out correctly and we were able to debit the converted 16-bit data at sample rates in the order of hundreds of ksps, but due to MATLAB® limitations, we can only read that data at an approximate maximum of 277 sps (considering the acquisition and display methods used). So, we settled on a sampling frequency of 200Hz for the final system.

Subchapter 4.1 highlights the main software used to achieve SPI communication with the external ADC, and BLE communication between the ESP32-DevKit-LiPo and the PC or Tablet running the application. Both forms of communication were tested separately, and each output was examined before merging it all into the final code. The results of both forms of communication had some differences: while the SPI data received matched the expected values, the values advertised by BLE showed an inconsistency as compared to the real value, defined by software. To correct this incoherence, it was introduced in the microcontroller the ability to filter the signal sent, through the moving average method with a window of 10. The new results proved to be much closer to the real values, but we still do not know what is responsible for this unexpected outcome. The method used to acquire data by BLE also turned out to be different from what was intended. Both BLE and serial communication should be able to provide a live feed of the data being plotted to further analysis, but we couldn’t replicate the method used in the serial live plot nor find an alternative solution, as each new idea either compromised the performance of the system or changed its the acquisition.

In section 4.2, the focus is on the MATLAB® application interface and functionalities. The main functionalities such as the different data acquisition methods, the mean pulse calculation, the BP parameter calculations and the auxiliar diagnostic are all complemented by a figure illustrating their respective results. The waveforms for all the acquisition methods (.txt file reading, serial live plot and BLE) are all clear as intended and the mean pulse calculation proved to accurate in every case tested. The algorithms responsible for determination of every parameter displayed in the application interface are all getting good results, apart from the IP calculation. Due to some waveforms not having an explicit IP, this algorithm is not always accurate, but for better approximation, it is given the option for the user to introduce the instant where the IP is detected manually, partially eliminating the error. The final functionality of the application, the Fast Diagnostic, should be looked at as a warning, as most results are based on studies performed on specific populations, which do not illustrate the general cardiac behaviour.

In Chapter 5 we have tested our system in three different cases: comparison with a commercial device, adaptability to different waveforms and comparison with reference algorithm, and a comparison between the two forms of communication implemented. First, our system was placed against a commercial product named Complior Analyse®, that is accepted by the medical community, and typically used in hospitals. We have compared the results from the mean pulse wave of 5 tests preformed on 2 different subjects for each method and reached the conclusion that our system can produce pulse waves that are very similar in both cases. The point-to-point correlation analysis resulted in a correlation factor of 0.8464 and 0.8633 for subjects 1 and 2 respectively, which represents a strong correlation between both methods. Furthermore, the Complior Analyse® also provides an estimation of some of the BP parameters addressed in this dissertation. The systole and diastole periods and the instants where the IP and SP are detected by our method have shown very little difference from the Complior device, with a maximum difference of 47 ms and a minimum of 12 ms. The MAP calculations are very accurate, as expected, and the PSC values had a significant difference in one of the subjects. As the latter utilizes the mean amplitude from the normalized mean pulse, the amplitude differences in both waveforms could have had influence in the result. Nevertheless, it is proved that our system can provide accurate results when compared to its commercial counterparts. The second analysis aimed to use our system to process different samples, taken from another similar sensor, and compare the parameter calculations from both. A correlation and Bland-Altman plot analysis were made for some of the parameters. We have achieved correlation coefficient of 0.6067, 0.8181 and 0.7447 for ED%, SEVR% and AIx% respectively which implies a strong correlation between both methods. In the Bland-Altman plots almost every value is within the limits of agreement, showing a good accuracy profile, but only a medic or clinician, who uses the test results in a clinical setting can decide whether the mean bias and LoA are acceptable or not. Again, the MAP calculations proved to be very accurate, and the PSC values had similar results to the previous analysis, having a mean difference of 0.25±4.29 mmHg. Lastly, the mean difference error for both DN and SP instants of detection of 0.058±0.0306 s and 0.0044±0.0263 s respectively, show very good results considering the limited number of samples. The last section of Chapter 5 is a comparison between the two forms of communications between device and application, BLE and USB. In this point, 5 samples were taken by each method, successively, to test the repeatability and accuracy of both methods when put against each other. The mean values from de 5 samples are very similar for most of the parameters, in both acquisition methods. This proves that the choice of sacrificing the live plot function in favour of a more accurate data acquisition process was a wise decision, since both methods show very similar results, despite obtaining a cleaner waveform using USB connection.

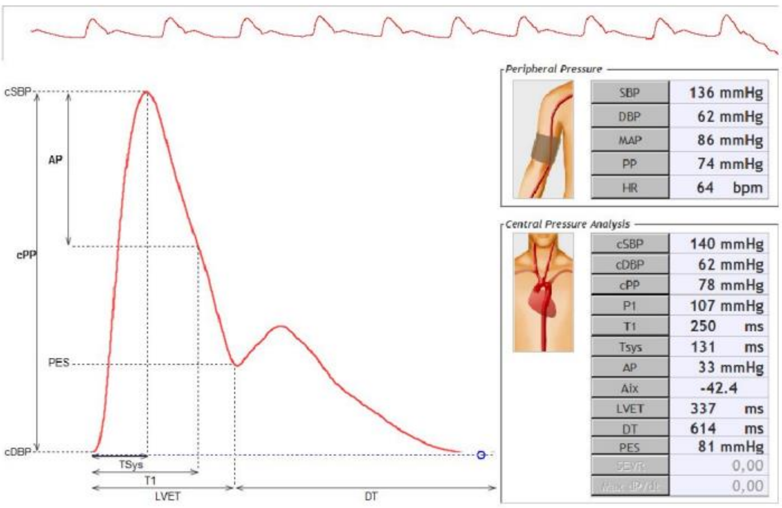
In short, the general results were as expected. Further developments and improvements, software wise, may be related to BLE communication and diagnostic validation.

6.1. Future work

Although the main objectives were reached, at the end there are always some suggestions and room for improvement. To complement the project development, some future works may be done, improving the system overall performance:

- Replicate the developed application to an Android or IOS environment;
- Implementing the possibility of saving the results on SD card if the control application is on a mobile device;
- Include tables and graphs illustrating a patient's measurement history over time;
- System miniaturisation (smaller size, lower consumption, …);
- Implement BLE live plotting capabilities;
- Transform the system into a small portable device powered by a battery capable of lasting at least one day's use;
- Add some form of SBP and DBP estimation capabilities to make the system independent;

Annexes



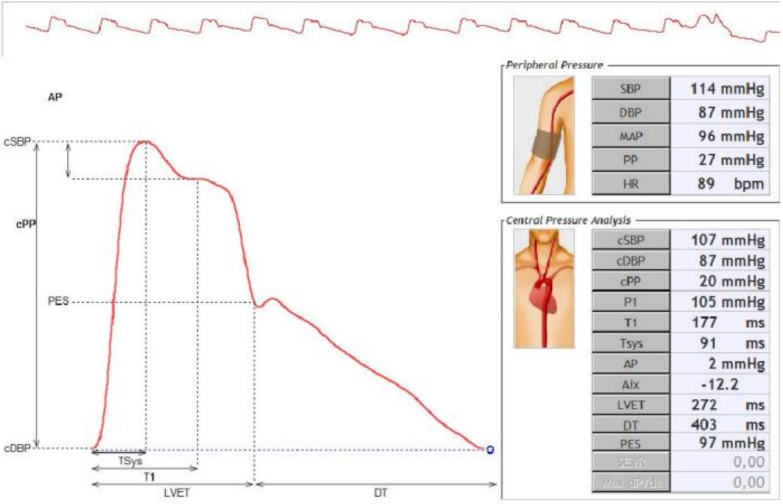
Peripheral Pressure

SBP	136 mmHg
DBP	62 mmHg
MAP	86 mmHg
PP	74 mmHg
HR	64 bpm

Central Pressure Analysis

cSBP	140 mmHg
cDBP	62 mmHg
cPP	78 mmHg
P1	107 mmHg
T1	250 ms
Tsys	131 ms
AP	33 mmHg
Aix	-42.4
LVET	337 ms
DT	614 ms
PES	81 mmHg
SEVR	0,00
Max dP/dt	0,00

Figure 1 – Full Complior Analyse® results for Subject 1



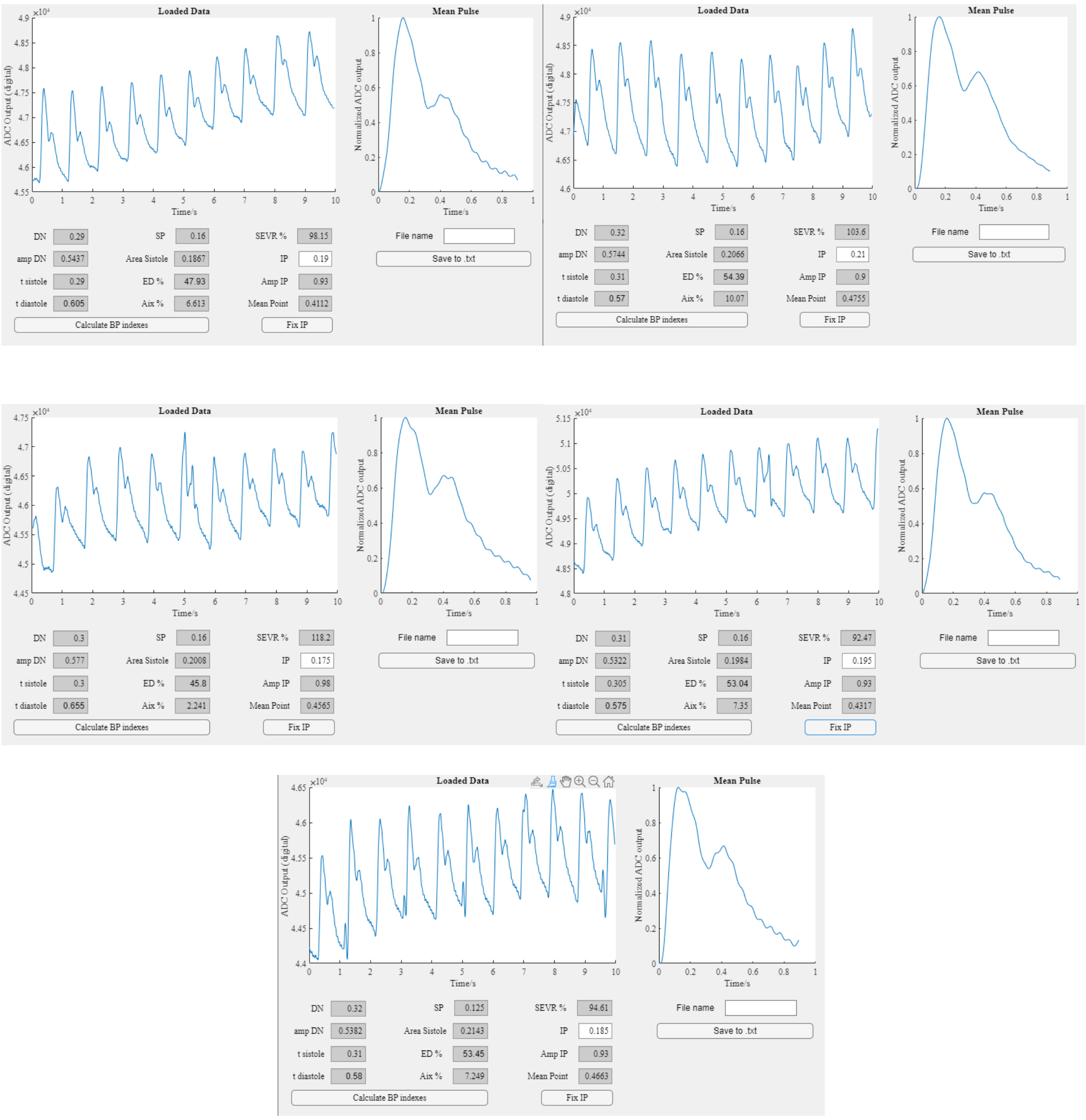
Peripheral Pressure

SBP	114 mmHg
DBP	87 mmHg
MAP	96 mmHg
PP	27 mmHg
HR	89 bpm

Central Pressure Analysis

cSBP	107 mmHg
cDBP	87 mmHg
cPP	20 mmHg
P1	105 mmHg
T1	177 ms
Tsys	91 ms
AP	2 mmHg
Aix	-12.2
LVET	272 ms
DT	403 ms
PES	97 mmHg
SEVR	0,00
Max dP/dt	0,00

Figure 2 - Full Complior Analyse® results for Subject 2



DN 0.29
amp DN 0.5437
t sistole 0.29
t diastole 0.605
SP 0.16
Area Sistole 0.1867
ED % 47.93
Aix % 6.613
SEVR % 98.15
IP 0.19
Amp IP 0.93
Mean Point 0.4112

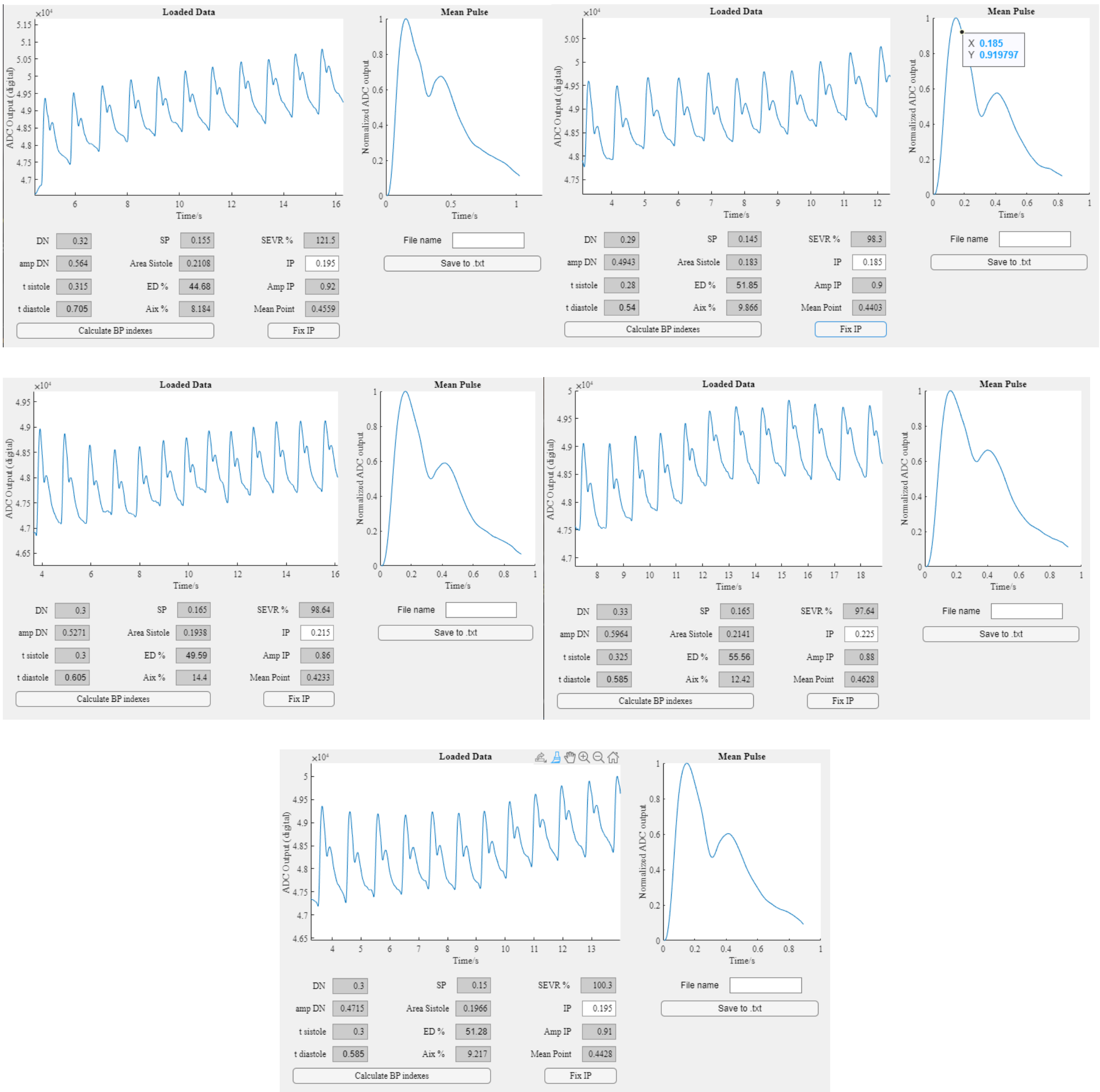
DN 0.32
amp DN 0.5744
t sistole 0.31
t diastole 0.57
SP 0.16
Area Sistole 0.2066
ED % 54.39
Aix % 10.07
SEVR % 103.6
IP 0.21
Amp IP 0.9
Mean Point 0.4755

DN 0.3
amp DN 0.577
t sistole 0.3
t diastole 0.655
SP 0.16
Area Sistole 0.2008
ED % 45.8
Aix % 2.241
SEVR % 118.2
IP 0.175
Amp IP 0.98
Mean Point 0.4565

DN 0.31
amp DN 0.5322
t sistole 0.305
t diastole 0.575
SP 0.16
Area Sistole 0.1984
ED % 53.04
Aix % 7.35
SEVR % 92.47
IP 0.195
Amp IP 0.93
Mean Point 0.4317

DN 0.32
amp DN 0.5382
t sistole 0.31
t diastole 0.58
SP 0.125
Area Sistole 0.2143
ED % 53.45
Aix % 7.249
SEVR % 94.61
IP 0.185
Amp IP 0.93
Mean Point 0.4663

Figure 3 – BP parameters and mean pulse calculations from BLE connection



DN 0.32
amp DN 0.564
t sistole 0.315
t diastole 0.705
SP 0.155
Area Sistole 0.2108
ED % 44.68
Aix % 8.184
SEVR % 121.5
IP 0.195
Amp IP 0.92
Mean Point 0.4559

DN 0.29
amp DN 0.4943
t sistole 0.28
t diastole 0.54
SP 0.145
Area Sistole 0.183
ED % 51.85
Aix % 9.866
SEVR % 98.3
IP 0.185
Amp IP 0.9
Mean Point 0.4403

DN 0.3
amp DN 0.5271
t sistole 0.3
t diastole 0.605
SP 0.165
Area Sistole 0.1938
ED % 49.59
Aix % 14.4
SEVR % 98.64
IP 0.215
Amp IP 0.86
Mean Point 0.4233

DN 0.33
amp DN 0.5964
t sistole 0.325
t diastole 0.585
SP 0.165
Area Sistole 0.2141
ED % 55.56
Aix % 12.42
SEVR % 97.64
IP 0.225
Amp IP 0.88
Mean Point 0.4628

DN 0.3
amp DN 0.4715
t sistole 0.3
t diastole 0.585
SP 0.15
Area Sistole 0.1966
ED % 51.28
Aix % 9.217
SEVR % 100.3
IP 0.195
Amp IP 0.91
Mean Point 0.4428

Figure 4 - BP parameters and mean pulse calculations from Serial (USB) connection

References

[1] Johnson RJ, Lanaspa MA, Gabriela Sánchez-Lozada L, Rodriguez-Iturbe B, (2015), “The discovery of hypertension: evolving views on the role of the kidneys, and current hot topics.” Am J Physiol Renal Physiol.

[2] Hales S, (1733), “Volume II of the Statical Essays”, Palala Press.

[3] Booth J, (1977), “A short history of blood pressure measurement.”, Proc R Soc Med.

[4] Roguin, A, (2006), "Scipione Riva-Rocci and the men behind the mercury sphygmomanometer.", *Int. J. Clin. Pract.* vol. 60, no. 1, pp. 73–9.

[5] Soman R, Wee J, Peters K, (2021), “Optical Fiber Sensors for Ultrasonic Structural Health Monitoring: A Review.”, *Sensors* 2021, no. *21*: 7345.

[6] Shah R Y, Agrawal Y K, (2011), “Introduction to fiber optics: Sensors for biomedical applications”, Indian journal of pharmaceutical sciences, vol. 73, no.1, pp. 17-22.

[7] G Quattrocchi, Berri P C, Vedova Dalla M D L, Maggiore P, (2021) “Optical fibers applied to aerospace systems prognostics: design and development of new FBGbased vibration sensors”, Department of Mechanics and Aerospace, Politecnico di Torino.

[8] Sabri N, Aljunid S A., Salim M S, Fouad S, (2015), “Fiber Optic Sensors: Short Review and Applications.”, Springer Series in Materials Science, vol. 204, pp. 299-231.

[9] Ghatak A, Thyagarajan K, (2012), “An Introduction to Fiber Optics”, Cambridge: Cambridge University Press.

[10] Pendão, Cristiano, and Ivo Silva, (2022), "Optical Fiber Sensors and Sensing Networks: Overview of the Main Principles and Applications", Sensors 22, no. 19: 7554.

[11] Elprocus, “Introduction to Fiber Optic Sensors and their Types with Applications”, URL: https://www.elprocus.com/diffrent-types-of-fiber-optic-sensors/ (accessed: 12.09.2022).

[12] OMRON, “Photoelectric Sensors”, URL: https://www.omron-ap.com/service_support/technical_guide/photoelectric_sensor/index.asp (accessed: 12.09.2022).

[13] Chatterjee S, Budidha K, Kyriacou P A, (2020), “Investigating the origin of photoplethysmography using a multiwavelength Monte Carlo model”, IOP Publishing.

[14] Kamshilin A, Margaryants N, (2017), “Origin of Photoplethysmographic Waveform at Green Light.”, Physics Procedia, vol. 86, pp. 72-80.

[15] Fischer C, Domer B, Wibmer T, Penzel T, (2017), “An Algorithm for Real-Time Pulse Waveform Segmentation and Artifact Detection in Photoplethysmograms.”, IEEE Journal of Biomedical and Health Informatics, vol. 21, pp. 372-381.

[16] Alian A A, Galante N J, Stachenfeld N S, Silverman D G, Shelley K H, (2011), “Impact of central hypovolemia on photoplethysmographic waveform parameters in healthy volunteers. Part 1: time domain analysis.”, Journal of clinical monitoring and computing, vol. 25, pp. 377-85.

[17] Roguin A, (2006), “Scipione Riva-Rocci and the men behind the mercury sphygmomanometer.”, International journal of clinical practice, vol. 60, pp. 9-73.

[18] Physiopedia, (2022), “Sphygmomanometer”, URL: https://commons.wikimedia.org/wiki/File:2020_Sfigmomanometr_elektroniczny.jpg (accessed: 2.09.2022).

[19] Verrij E, van Montfrans G, Bos WJ, (2008), “Reintroduction of Riva-Rocci measurements to determine systolic blood pressure?”, The Netherlands journal of medicine, vol. 66, pp. 2-480.

[20] Shimbo D, (2016), “Studies comparing ambulatory blood pressure and home blood pressure on cardiovascular disease and mortality outcomes: A systematic review.”, Journal of the American Society of Hypertension: JASH, vol. 10, pp. 224-234.

[21] Nessler K, (2021), “The quality of patients' self-blood pressure measurements: a cross-sectional study.”, BMC cardiovascular disorders, vol. 21, pp. 1-539.

[22] Lazazzera R, Belhaj Y, Carrault G, (2019), “A New Wearable Device for Blood Pressure Estimation Using Photoplethysmogram.”, Sensors (Basel), vol. 19, pp. 11-255.

[23] H. Miyashita, (2012), “Clinical Assessment of Central Blood Pressure”, Curr. Hypertens. Rev., vol. 8, no. 2, pp. 80–90.

[24] McEniery C M, Cockcroft J R, Roman M J, Franklin S S, Wilkinson I B, (2014), “Central blood pressure: current evidence and clinical importance”, European Heart Journal, vol. 35, issue 26, pp. 1719–1725.

[25] Industrial Fiber Optics, Inc., (2020), “Fiber Optic Green LED IF E93”, IF-E93 datasheet, URL: https://i-fiberoptics.com/pdf/ife93-11-20.pdf (accessed: 24.08.2022).

[26] Industrial Fiber Optics, Inc., (2020), “Fiber Optic Phototransistor IF D92”, IF-D92 datasheet, URL: https://i-fiberoptics.com/pdf/ifd92-1809.pdf (accessed: 24.08.2022).

[27] Industrial Fiber Optics, Inc., (2020), “Plastic plastic fiber couplers (Splitters)”, IF-562 datasheet, URL: https://www.i-fiberoptics.com/pdf/coupsplitr.pdf (accessed: 24.08.2022).

[28] Chowdhury M H, Shuzan M N I, Chowdhury M E H, Mahbub Z B, Uddin M M, Khandakar A, Reaz M B I, (2020) “Estimating Blood Pressure from the Photoplethysmogram Signal and Demographic Features Using Machine Learning Techniques”, Sensors (Basel, Switzerland), vol. 20, pp. 11-3127.

[29] Espressif Systems, (2021), “ESP32 Series”, ESP32 Series Datasheet v3.8.

[30] Collins T, Woolley S, (2019), “Some heart-rate monitors give less reliable readings for people of colour”, URL: https://theconversation.com/some-heart-rate-monitors-give-less-reliable-readings-for-people-of-colour-121007 (accessed: 26.09.2022).

[31] Karki J, (2002), “Analysis of the Sallen-Key Architecture”, Texas Instruments Inc.

[32] Dhaker P, (2018), “Introduction to SPI Interface”, Analog Dialogue, vol. 52.

[33] Texas Instruments Incorporated, (2007), “Dual, 500kSPS, 16-Bit, 2 + 2 Channel, Simultaneous Sampling ANALOG-TO-DIGITAL CONVERTER”, ADS8361 datasheet, URL: https://www.ti.com/lit/ds/sbas230e/sbas230e.pdf (accessed: 12.10.2022).

[34] Texas Instruments Incorporated, (2011), “User’s guide for ADS7861/8361/7863EVM”, URL: https://www.ti.com/lit/ug/slau094a/slau094a.pdf (accessed: 12.10.2022).

[35] Townsend K, Cufí C, Akiba, Davidson R, (2014), “Getting Started with Bluetooth Low Energy”, O’Reilly Media, Inc.

[36] Hendrick T, “Using the ADS8361 with the MSP430 USI port”, © 2006 Texas Instruments Incorporated, URL: https://www.ti.com/lit/an/slyt244/slyt244.pdf?ts=1667108468973 (accessed: 14.10.2022).

[37] Li B N, Dong M, Vai M, (2010), “On an automatic delineator for arterial blood pressure waveforms”, Biomedical Signal Processing and Control, vol. 5, pp. 76–81.

[38] Catherine T P, (2010), “Augmentation pressure and subendocardial viability ratio are associated with microalbuminuria and with poor renal function in type 1 diabetes.”, Diabetes & vascular disease research, vol. 7, pp. 24-216.

[39] Emre A, (2017), “Baseline subendocardial viability ratio influences left ventricular systolic improvement with cardiac rehabilitation.”, Anatolian journal of cardiology, vol. 17, pp. 37-43.

[40] Janner J H, Godtfredsen N, Ladelund S, Vestbo J, Prescott E, (2010), “Aortic Augmentation Index: Reference Values in a Large Unselected Population by Means of the SphygmoCor Device”, American Journal of Hypertension, vol. 23, issue 2, pp. 180–185.

[41] Chung J, (2010), “Reference values for the augmentation index and pulse pressure in apparently healthy korean subjects.”, Korean circulation journal, vol. 40, pp. 71-165.

[42] Bamira D, Picard M H, (2018) “Echocardiology—Assessment of Cardiac Structure and Function”, Heart, lung & circulation, vol. 11, pp. 167-73.

[43] Herbert A, Cruickshank J K, Laurent S, Boutouyrie P, (2014), “Establishing reference values for central blood pressure and its amplification in a general healthy population and according to cardiovascular risk factors”, European Heart Journal, vol. 35, issue 44, pp. 3122–3133.

[44] Leitão C, “Sensores de fibra ótica como tecnologia não-invasiva para avaliação da pressão arterial central”, Doctoral thesis, Aveiro University, 2017.