ESCUELA POLITÉCNICA SUPERIOR
UNIVERSIDAD CARLOS III DE MADRID

GRADO EN INGENIERÍA EN TECNOLOGÍAS DE TELECOMUNICACIÓN
–
DEGREE IN TELECOMMUNICATIONS TECHNOLOGY ENGINEERING

BACHELOR THESIS – TRABAJO DE FIN DE GRADO

SMART PILL DISPENSER FOR DEPENDENT PEOPLE

Author: Guillermo de Juan Grau
Tutor: Daniel Díaz Sánchez
September 2015

ACKNOWLEDGEMENTS

In this section I would like to publicly thank everyone who has contributed in making possible for me to reach this point in my life.

To my parents, without I could not be the person I am today, for making me a better person since the day I was born.

To my sisters, for being always there when I need them.

To Ana, for every one of those moments in which I needed all the support and love in the world and she gave me much more than that.

To Gonzalo for those endless nights of studying, as it has finally been worth it.

To Rafa, for all he taught me the moments we spent together.

To Sara, Irene, Alicia and Dani for being the main drives of my life.

And to everyone who has helped in one or another to make this come true.

Thank you, from the bottom of my heart.

INDEX

CHART INDEX

FIGURE INDEX

ABSTRACT

Many medical errors are due to the fact that people in charge of patient or elder's medication have to deal with sorting huge amounts of pills each day. This paper consists on the conception, design and creation of a pillbox prototype intended to solve this deficiency in the medical area as it has the ability of sorting out the pills by itself as well as many other advanced features, with this device being intended to be used by hospitals or retirement homes. For the design of this device open-source based technologies will be used when possible including 3D design software and software IDE, and the prototype will be created using 3D printed plastic parts.

1 INTRODUCTION

1.1 INTRODUCTION

Medicine hasn't always been the precise science which we know nowadays, in fact, throughout history the vast majority of maladies and afflictions would have resulted in probable death. Fortunately medicine has experienced a great development in the last century, resulting in an increased life expectancy and the possibility to cure a great quantity of diseases and health disorders.

Pills are one of the pillars of medicine, being used as direct treatments of some sort of illness as to assuring some medical condition remains stable (people with organ transplants or patients suffering from a chronic disease such as VIH).

1.2 PROBLEM STATEMENT

As pills have taken such an important role in everyday life there has been the past years an increase in the number of medical neglect cases related to incorrect medication given to patients, such as the case of the nurse who gave a patient a paralytic instead of an antacid that was prescribed by the doctor, causing the patient's death [1]. After seeing so many of these cases it is evidently crucial that the correct pill is taken by the correct person at the correct time, otherwise taking an incorrect one or not taking one at all may expose the patient to several dangerous situations, ranging from mild health issues up to death.

Other cases of wrong pills being ingested by patients are caused by patients themselves, especially at an old age. As people grow old the human body tends to malfunction and the number of pills the average person has to take when certain age is reached greatly increases, were, according to a 2008 study published in the Journal of the American Medical Association, more than 40 percent of Americans age 65 and older take five medications a day [2]. Usually they are a wide range of different pills an elderly has to take at different times. Keeping track of taking the right pill at the right moment each day can become a challenging experience for the elderly, as it is not as easy as it could be for a younger person. This fact is easily explained when we understand that many of the abilities such as sight, memory or logical capabilities tend to decrease in a proportional way to age once human beings have entered old age, making it difficult for them to remember which pill to take at the correct time, remembering to take them or confusing one pill with another as the person may not be able to distinguish one from another thanks to their decreased sight as well as the similarity in the pills forms and colours. This problem will most surely be a cause for concern for the people surrounding the pill-taker, as not taking a pill at the correct time can cause severe problems (such as organ rejection in a patient with organ transplant or heart attack in patients suffering from grave heart conditions).

On the other hand several problems related to the high amount of pills nowadays are prescribed to patients are found in hospitals or in retirement homes. In these places on of the main jobs is to give out to its patient the correct pills. Managing, sorting and giving out the pills to each one of the patients can sometimes have a high chance of error, with a patient or resident receiving one or more incorrect pills.

Finally there are situation where taking an incorrect amount of pills is a matter of the patient's inexperience and/or ignorance. No matter the cause, it has been proven that there is a significant risk of people ending up swallowing the incorrect medication or dose.

1.3 OBJECTIVES

The main objective of this project will be to solve the aforementioned problems by designing and synthesize a tool which will enable the owner to track every pillf to ingest in an easy and simple way requiring no training or complex learning from their side in order to operate the device. This device will be an intelligent pill dispenser.

The pill dispenser will be designed to prevent errors in hospitals and retirement homes where many pills have to be given daily to each one of the patients, each patient owning a device will not only drastically reduce the chances of errors occurring but also well optimize and speed up work for the caretakers/nurses by allowing the device to take care of pill management for them and freeing the time slot usually dedicated to that.

This device is intended to log the pill name, number of pills and hours at which each pill is actually taken versus the time it should have been taken. Nowadays there has been an increasing awareness as the number of pills prescribed to elder people, stating that so many pills may have negative effects on the patient's health. [3] [4] The pillbox's logs will help gather data concerning this matter.

There is a need to ensure the device is wirelessly connected so device management (defining the hours when a particular pill must be taken, number of pills in each compartment, etc…) as well as possibility of emitting warnings to the owner's relatives or nurses if needed (such as the patient not taking pills).

In essence, the device will have to be a wireless electronic apparatus, having special attention to make it very precise as an error could prove fatal.

1.4 REGULATORY FRAMEWORK

The pillbox is a device which helps handling patient's medication.For this reason the process in which to implement the pillbox in the selected organizations (mainly hospitals and retirement homes as already mentioned) will be a process similar to the one medications have to follow when being implemented in the general market.

Chart 1 shown below shows the different level through which the product will have to traverse after being implemented in public institutions. [5]

Level 1: Medication agencies EMEA (European Medicines Evaluation Agency) execute the authorization registry report in which they authorize the product commercialization based on demographic risk versus benefit study.
Level 2: The Spanish Dirección General de Farmacia y Productos Sanitarios del Ministerio de Sanidad fix the product's Price as well as the government's funding conditions if any.
Level 3: The Health Counselors (Consejerias de Sanidad) in each of Spain's autonomous communities will be the ones funding the use of the product.
Level 4: Each one of the sanitary institutions will be the ones to value if the product if worth the implementation depending on the product's value and necessity inside the institution.

Chart 1: Regulatory framework leves for a new farmaceutical drug or related

Before the product traverses through any of the levels shown in the table it will be necessary to study if the invention patented in the Spanish territory in the OEPM (Oficina Española de Patentes y Marcas) to stop any plagiarism from happening.

1.5 DOCUMENT STRUCTURE

In this section the structure this document follows will be detailed along with a brief explanation of each one of the different sections:

Introduction

In this section the general aims of this document is provided alongside the proposed solution and the regulatory framework which concerns the product.

State of the art

In this section different technologies similar to the one being designed are analysed as well as the different existing technologies which will help implement the pillbox.

Product design specification

This section details the different requirements and features the pillbox must possess in order to achieve successfully the aim intended in this document.

Conceptual design

In the conceptual design section an early draft of the pillbox's behaviour is outlined as well as different draft drawings of how the pillbox could look like. These draft drawings will be later used to design the final pillbox.

Pillbox design

In this section the pillbox is designed. The pillbox functionality is divided into modules and a simplified schematic is produced. The different modules are detailed from an operational point of view and changes to the original design are also outlined. The 3D parts are designed and detailed alongside with the pillbox's electronic circuit schematics and the code needed for the pillbox to successfully function.

Synthesizing the pillbox

This section deals on how the prototype is converted from digital to a real model. The chosen 3D technology which will be used to print the pillbox is explained along with the procedure to follow an encountered problems while achieving that.

Finally the pillbox's creation and assembly is also covered.

Testing the pillbox

This section shows the different tests applied to the pillbox to ensure complete functionality and the changes the pillbox has to undergo to successfully pass the failed tests.

Planning and budget

This section details the planification followed for the completion of this project and the hours dedicated to each tasks as well as the necessary budget to cover the project.

Conclusions

This section covers the future improvements that could be done to the pillbox in order to optimize it and increment the number of features it presents as well as to correct errors the actual prototype possesses.

The project as a whole is reflected upon once it has been finished and the several conclusions drawn out of it are exposed.

References

The sources used and cited throughout this project are listed in this section.

Annex

Additional documentation such as high detail schematics of the printed parts are included in this section.

2 STATE OF THE ART

2.1 INTRODUCTION

Throughout this section a detailed a thorough investigation of products similar to the one developed in this project will be carried out so that a learning process is achieved in order to avoid committing common errors and flaws as well as ensuring that this project offers and outsmarts every other product in the market nowadays.

In second place the different available technologies needed for the creation of this product will be examined in order to pick the optimal ones which will be finally used to make it a reality.

2.2 SMART PILL BOXES

Despite the short time pill boxes have been in the market many different models can already be found with a wide range of different functionalities and intended for disparate uses and contexts. Most prominent ones will be cited.

2.2.1 AdhereTech's smart pill bottle

AdhereTech smart pill bottle [6] consists on a plastic bottle in which to store the pills and some circuitry which allows for the bottle how many times the pillbox is opened. It uses sensors for a precise measurement of remaining medication and can send messages, calls or notifications to patient if pill is not taken. Its main weak points are the impossibility of storing different pill types in one bottle as well as having a lack on information on how the pillbox can be managed, which may led to believe it will only be implemented at hospital-level. On figure 1 the pillbox is shown.

-

Figure 1: AdhereTech's smart pill bottle

2.2.2 iMediPac

iMediPac [7] is an intelligent pillbox in which pills are sorted by each day in a disposable envelope which will be later inserted in the pillbox. Each day the pillbox lights up the compartment which has to be taken that day and the user breaks and retrieves the pills from inside the pillbox. If pills are not taken the pillbox alerts relatives with notifications. On the down side, the pillbox is very bulky of a huge size making it difficult for a person to carry it all day, while pills will probably have to be taken throughout the day. It also shows in the promotional videos a complex method of configuring personal data, which may lead to elderly people not being able to program it correctly. Finally it doesn't solve the problem of pills complexity as nurses, pharmacists, etc… have to spend huge amounts of time preparing the dose for each day in the refill sheet as well as refills having to be constantly bought and refilled. Figure 2 shows the iMediPac pillbox with a refill loaded with pills.

[7]

Figure 2: iMediPac pillbox.

2.2.3 MemoBox

MemoBox [8] is a small pillbox with just one compartment in which many different pills can be stored. It possess a portable size and can track when the pillbox is forgotten at home. Whilst its simplicity can be very attractive it offers no more functionality than the one which could be achieved with a normal pillbox and a pill-reminder app installed in the phone. Figure 3 shows the MemoBox pillbox with different pills stored inside.

Figure 3: MemoBox pillbox

2.2.4 uBox

uBox [9] is a really small pillbox which can be easily carried by its user. It consists on a cylindrical structure with different compartments inside. It has a childproof lock and can alert user and relatives if pills are not taken. Different pills for each dose are stored in the same compartments, which may be its only inconvenient. Figure 4 is an actual photograph of the pillbox while Figure 5 shows an exploded view of the pillbox.

Figure 4: uBox pillbox

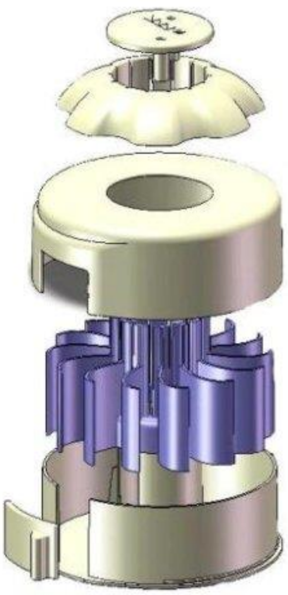


Figure 5: Exploded view of the uBox pillbox

2.2.5 GlowCap & GlowPacks

GlowCap [10] is a smart pillbottle lid which can attach to the normal pillbottle and comes with a wall plug device (GlowPack) which lights up each time the user has to take a pill. The pillbox lid has an interior button which can be pressed to directly order pills to the drugstore. This smart pillbox has many inconvenients:

- One cap per bottle of pills.
- Not portable as needs a wall plug-in.
- Elderly need to have computer knowledge in order to set it up.

- Pressing twice the button when pill box is near emptying places a call from your phone to the pharmacy, same thing could be achieved with speed-dial directly from the phone.
- As they are caps, only bottles of pills are allowed, not pill-boxes.

Figure 6 shows a photograph of both the GlowCap and the GlowPack.

Figure 6: GlowCap and Glowpack pillbox

2.2.6 MedMinder

MedMinder [11] is a huge pillbox without the possibility of being portable due to its size as well as having to be plugged into a wall outlet for it to work. Different pills go in the same compartment for each one of the pill doses, therefore not improving pill errors and for the pillbox to work it is required to pay a monthly fee. Figure 7 shows the MedMinder pillbox.

Figure 7: MedMinder pillbox

2.2.7 ePill's Monitored MedSmart PLUS

ePill's Monitored MedSmart PLUS [12] offers a range of automatic pill dispenser with none strong points as most of the require a monthly fee to work and are complex to operate. As well they are not portable due to their size and have an outdated style. Their elevated price also may dissuade the users to buy this pill dispenser. Figure 8 shows some of these pill dispensers.

Figure 8: ePill's Monitored MedSmart PLUS pillbox

2.2.8 Philips Medications Dispensing Service

Philips Medications Dispensing Service [13] is a non-portable pill dispenser oriented towards elder users which stay at home. It is operated in a very simple way with the use of one button and has voice prompts. It is equipped for emergencies with a large pill storage and a battery backup system so it can supply pills to the user in case a power outage. On its downside it is worth mentioning its non-portability and the use of individual cups with each day pills, not reducing probability of confusing pills for the caregivers. Figure 9 shows the Philips Medications Dispensing Service pillbox.

Figure 9: Philips Medications Dispensing Service

2.2.9 Conclusions

After studying the models in great detail the key strong aspects it will be necessary to focus will be:

Medication adherence

Defined as "whether patients take their medications as prescribed (eg, twice daily), as well as whether they continue to take a prescribed medication"[14] ,we are able to see that medical adherence is the key aspect the smart pillboxes focus on, as it will be the main issue, as if this is not achieved major health issues could derive.

Portability

Many of the reviewed pill boxes have to be plugged in to a power outlet, which may be a great inconvenience as it may stop medication adherence, that is due to the elderly people not constantly staying always at home, but pill remainders are constantly needed throughout the day. This will be the main reason that the project will focus on a portable device.

Simple use

Every one of the reviewed product was oriented towards elderly people, but, in contraposition, some of them expected some advanced knowledge on new technologies from the elderly people in order to configure and use the product. This is not what will be achieved in this project as it is a firm belief that systems have to be simple in order to be effective. Philips Medication Dispenser Service has implemented a one-button-only way for the seniors to interact and use their device, this project will focus on that aspect to create a simple to use interface in the device.

Reduction of error probability when filling the pill box

Every single one of the products in the market nowadays have different compartments at best were the pills to be taken each day are introduced, even though they may be different pill types. This does not differ in any way from the way nurses each day fill each of the patients cups with the day pills, and therefore, it is a system prone to suffering the same flaws than the ones manual systems suffer today, having equal chances of mistaking the pills and causing a medical neglect which can result even in a patient's death.

Tracking pill intakes and safety notification systems

Tracking each time the user takes a pill and notify the patient itself and relatives will be necessary for safety reasons as well as to maintain the competitive standards if it reaches the market. This aspect will also be vital as it will provide scientific studies with decisive information and will help doctors detect medical adherence problems as well as reassuring the patient's kin that its relative is taking its medication.

Low cost

Pillbox will need to be accessible to the general public as well as be adquired as a wholesale by hospitals and retirement homes in order for the product to succeed, therefore it will be a mandatory requisite to have a low cost price.

2.3 Electronic technologies

All the electronical circuitry which will act as a brain of the pillbox will need to possess the following capacities:

- Sufficient processing capacity to carry out the code precise and efficiently.
- Memory enough to store code and logs generated.
- Interaction with diverse communication modules in an easy and precise way.
- Small size.
- Low price.

It has been decided that the most optimal way to implement all the electronic modules will be by using an open-source electronics platform, the best-suited candidates being:

2.3.1 Arduino

This platform has been a few years going round as the preferred prototyping board for small projects defined as "a microcontroller board based on the ATmega328. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, an on-board resonator, a reset button, and holes for mounting pin headers. "[15]

While a cheaper substitute would come by directly using a microchip the Arduino comes in many models making up for a preassembled expandable miniature board. Such as the Arduino Mini which can be seen in figure 10.

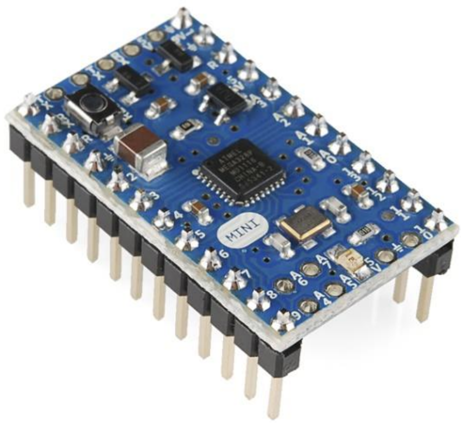


Figure 10: Arduino Mini

2.3.2 Raspberry Pi

This prototyping platform [16] is much bigger than the Arduino but comes with great processing power, as it can run a full operative system such as linux and many ports including usb and hdmi are also included. It may not be suitable for the pillbox as it has a high power consumption as well as being a big prototyping board which may not fit into the desired design inside the pillbox. Figure 11 shows the latest model available, the Raspberry Pi 2 Model B.

Figure 11: Raspberry Pi 2 Model B

2.3.3 The WiPy

Development board focused on developing IoT (Internet of Things) projects such as the one described by this document. Includes WiFi antenna, low power consumption all coming at a low cost [17]. This board has the most desirable features to be included in the project as it is more powerful than the Arduino board without the high power consumption the Raspberry Pi board presents as well as being very reduced in size. Figure 12 shows the WiPy board with some of its characteristic features.

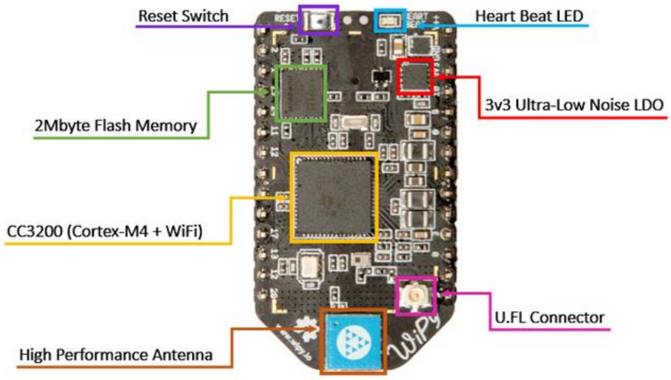


Figure 12: WiPy board

2.3.4 Conclusions

Considering the type of specifications the pillbox will have concerning portable size and wireless communications the WiPy is the best suited board for this project. Unluckily this board still is in development as it is a state-of-the-art board and access to it would mean a significant delay in the project completion.

Raspberry Pi board results in a board too big for the design and at a higher power consumption due to its excessive processing power in comparison to the one required for this project therefore the Arduino will be the selected board in order to become the pillbox's CPU. One of its greatest advantages refers to the board being open-source, therefore many other companies can create their own Arduino boards, making it possible to acquire one with lower costs than an original Arduino board.

2.4 DESIGN TECHNOLOGIES

2.4.1 Designing the models

Modelling the physical parts which will become the main body of the pill box will need to be thoroughly designed by the help of a computerized software.

After an extensive search there was a clear need to make the production of the smart pillbox within a reasonable price, therefore a free open-source software extensively used and awarded by professionals was decided to be used. Software will be FreeCad, an Open Source parametric 3D CAD Modeler, the decisive advantage for it to be chosen being a parametric modeller, "Parametric modeling allows you to easily modify your design by going back into your model history and changing its parameters. FreeCAD is open source (LGPL license) and completely modular, allowing for very advanced extension and customization." [18]

2.4.2 Synthesizing the models: 3D printing

2.4.2.1 Introduction

3D printing is known as the technology to create a three-dimensional object mechanically based on the model of the object stored in a digital file. This process may also be referred as additive manufacturing. This denomination comes from the basic principle 3D printing uses to fabricate the object. Even though there are many different techniques to print a 3D object (different types of 3D printing will be examined thoroughly in the next section), the basic principle remains the same for all of them. This principle is based on slicing the digital 3D model into horizontal layers of very small height. The printers will be then adding layer after layer till the object is finished, with the different types of 3D printers varying only in the way layers are built.

3D printers are a state-of-the-art technology which has been exponentially reinforcing its presence within the market for the past couple of years. The popularity of this technology as well as the uses for it are also growing each day and with it the impact that it is causing in the lives of the people, as nowadays it is possible to print even a prosthetic hand (figure 13) and there are currently projects developing 3D-printed cars, houses or even organs with special printers which use biological tissue. [28]

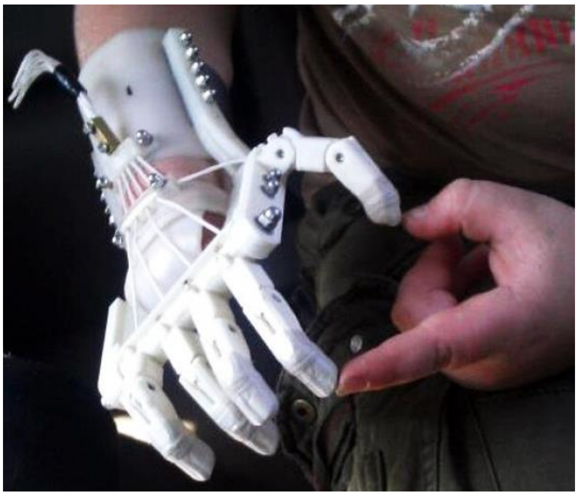


Figure 13: Prosthetic printed hand

2.4.2.2 Types of 3D printing

There are many types of 3D printing techniques which may print in an extensive range of materials:

SLA

SLA refers to Stereolithography, one of the oldest 3D printing processes, patented in 1986, which starts with a vat full of photo-reactive resin in which a platform is submerged and an ultraviolet laser. The laser shines where the first layer of the model is and the resin solidifies. The platform then lowers another millimetre and the process is repeated. Parts created using this method may require the use of supporting structures which will be removed once the object is finished printing. [29]

SLS/SLM/EBM

SLS (Selective Laser Sintering) and SLM (Selective Laser Melting) are both processes similar to SLA with the difference of powder instead of liquid resin being in the vat. In SLS the laser melts the powder enough so that it fuses together but the finished part can be porous or at least its porosity may be controlled. On the other hand, SLM fully melts the powder so that it is not fused but instead it is melted into a homogeneous part. [30] [31]

EBM follows the same procedure as SLM but instead of a laser being the one which melts the powder, it is an electron beam which fires electrons and the whole process usually is conducted under high temperatures of up to 1000ºC. This process is usually used to create titanium pieces and, as it is an expensive process it usually is used for bone replacements in the medical industry or in the aerospace industry.

LOM

LOM (Laminated Object Manufacturing) is a process in which adhesive-laminated paper sheets are used as the main material used to build the three-dimensional object. A sheet is rolled onto the platform and fixed by a heated roller which melts the plastic onto the platform or the layer below. A knife/laser cuts the layer of the object onto the sheet of plastic and the excess plastic layer is rolled while the platform lowers so that the next layer can be built. [32]

FDM

FDM (Fused Deposition Modeling), also called as FFF (Fused Filament Fabrication) are the same process with different names due to trademark issues (FDM is a Stratasys trademark).[33] This process extrudes melted plastic filament onto a platform layer by layer, with the plastic cooling enough to solidify once it has been fixed into the object being built. Parts created using this method may require the use of supporting structures which will be removed once the object is finished printing.

2.4.2.3 Conclusion

As for this project it was necessary to maintain the total production price of the pillbox as low as it could be. When reviewed it was conclusive to say that every one of the 3D printers which could be bought was well above a thousand euros. In spite of those prices, after investigation it was found out that there were people and organizations designing totally open-source 3D printers which could be built by each person in a DIY (Do-It-Yourself) fashion for a much lower price than any of the commercial 3D printers. Each one of the parts of the printer could be either bought locally in a generic shop (metal rods, rubber bands, etc…) or in an electronic shop (Arduino board, stepper motors, etc…). All the plans for building it as well as the necessary software to make it work is freely available in Internet as the whole of the 3D printer is open-source, therefore it was decided that it was mandatory to build one of these 3D printers as to finally create the pillbox's prototype.

Throughout the following sections the history of these particular open-source 3D printers as well as the printing basics the follow and the assembly of the particular printer used to create the pillbox will be detailed.

2.4.2.4 3D printing history

This section will detail the history behind the open-source RepRap project.

RepRap project (short for **Rep**licating **Rap**id-prototyper) started in the year 2005 founded by Adrian Bowyer with the idea on mind to design and create a 3D printer with an open-source philosophy of self-replication arguing that it would not be possible for the industry to create a self-replicating machine as it would never be profitable enough.

The project's first printer was finally completed by 2008 with the name Darwin. The project gained a lot of followers and finally the RepRap Foundation was created from where 3D printed parts could be bought. Each one of the plastic parts conforming a RepRap printer can be built up by another RepRap printer. The most iconic printer was designed and finished by 2009, this printer was called Mendel and it gave birth to a new type of printer called Prusa Mendel, engineered by Josef Prusa in 2010. The Prusa printer has been the best printer which, nowadays in 2015, the best considered printers are revisions of the Prusa Mendel. This project will be using a Prusa Steel 3, which is the 3rd revision of the Prusa with a steel frame to confer more stability while printing. In spite of Prusa being the most used there is a wide range of different RepRap printers to choose from.

Zack Smith, one of the first participants in the RepRap project created the site Thingiverse, a platform in which anyone can freely upload the designs they create so that everyone can download them free of cost and print them if they have access to a 3D printer.

The final aim of the RepRap project is to be able to position 3D printers in households, factories and schools and universities for a very low price. The freedom of making a real tangible object from a digital model than anyone can design is believed by many in and out of the RepRap project that can cause a revolution and make a huge technological progress. [34]

2.4.2.5 3D printing basics

Filament deposition

FDM 3D printing technique feeds plastic material from a roll into an extruder which melts down the plastic and extrudes it forming each time one of the printed object's layer. It is compulsory to calibrate the extruder's position with respect to the printing platform as a poorly calibrated extruder can have adverse effects on the print quality. This is due to the fact that the width of the deposed plastic filament is very small and precise and the extruder needs to have enough space for the filament to exit the extruder while being pressed at the same time onto the platform (or the last printed layer) by the extruder. If not enough space is left for the filament to exit the extruder, the plastic will be forced to exit through the sides leaving an unlevelled base for the next layer whilst if there is too much space between the extruder and the platform the plastic will likely stick into the extruder and not to the platform. In section “Printing in 3D” the extruder’s calibration procedure in order to get optimal results will be detailed. Figure 14 shows the correct and incorrect positions and the likely outcome in each one of the plastic filament.

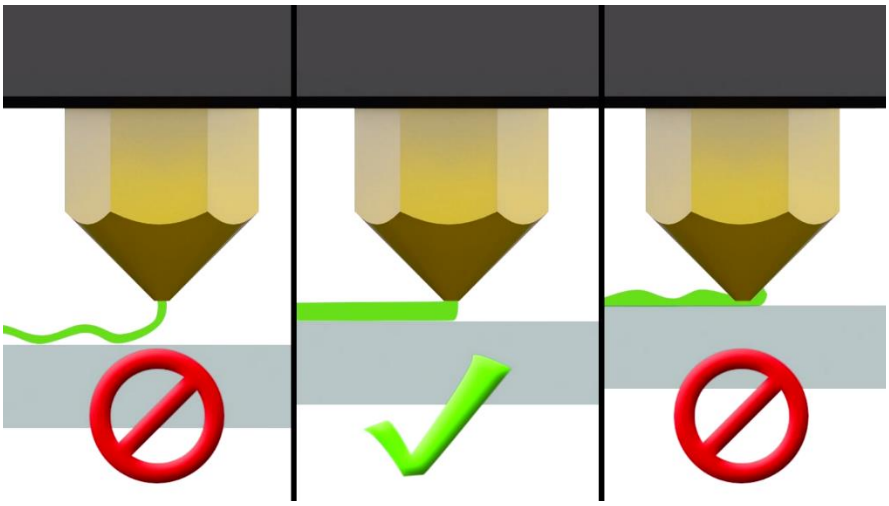


Figure 14: Extruder correct and incorrect positions with respect to the printing board

Types of plastic

There are different types of plastic to choose from when deciding to print using an FDM 3D printer in the market, being ABS and PLA the two most popular choices which can be bought. [34]

ABS (Acrylonitrile Butadiene Styrene) is a thermoplastic whose melting point is around 225ºC, printed parts with this material possess a great strength and flexibility. PLA (Polylactic Acid) on the other hand has a lower melting point than ABS, around 180ºC and may have some drawbacks compared with ABS, as it may be cracked more easily and glueing it together is found to be more difficult. ABS was considered at first to be the preferred option at which to print the pillbox parts but the composition of the plastic filaments was finally found to be crucial in deciding which plastic to use. ABS was found out to be petroleum-based while PLA is derived from “renewable resources, such as cornstarch, sugar cane, tapioca roots or even potato starch”. This fact not only makes PLA an environmentally friendly material to work with but, as the pillbox is intended to have medical applications, with the plastic directly in contact with the pills which will be swallowed by patients and residents, it is highly preferable to use a plant-based plastic than a petroleum-based one which could turn out to be hazardous to the human health. [35]

3 PRODUCT DESIGN SPECIFICATION

Product Design Specification section will define the aims the product is intended to achieve as well as the mínimum mandatory characteristics it will need to possess in order to fulfill the forementioned aims.

3.1 SCOPE OF THE PRODUCT

The product's target audience are elderly people with many different pills to take throughout the week. While initially the product was designed with thinking the target audience as the main buyer of the product, it was later changed to make hospitals and retirement homes the main clients of the product, as it could increase the center's productivity by simplifying pill management and delivery as well as giving more control and supervisión over pill intake by the center's residents.

3.2 PRODUCT DESIGN

The main product consisting in a pillbox will be designed around the following principles:

- **Portability**, as it is much easier for the elderly to carry with them the pillbox instead of having to move each time a pill needs to be taken as well as making it easier to notice when the pillbox notifies a new dose has to be taken.
- **Reasonable pill storage size**, as it will not be efficient for the caregivers to refill the pillbox very frequently.
- **Very clear acoustic and luminous notifications**, as many elderly possess greatly diminished senses and they need to be clearly aware when the pillbox is claiming their attention for a new pill dose.
- **Sturdiness** so that the product's lifespan is reasonably long while being subject to falls.
- **Wireless connectivity** so that caregivers may manage the device remotely
- **One compartment for each one of the pill types**, as this is a key aspect for highly reducing medication errors. By applying this characteristic the device is the one that manages which pills are given each day and simplifies the caregiver's work, making more time for them to use in more pressing matters than pill-sorting.
- **Ergonomic design** so that both caregivers and seniors may use the device in a comfortable and safe way.
- Made with **open source materials and components** so that the fabrication price of each unit is reduced.
- **Cheap fabrication price and low technical machinery needed** so that the pillbox may be easily produced without a great investment and can be bought in great numbers by hospitals, retirement centers, etc…

3.3 INITIAL CONSTRAINTS OF THE PROJECT

This project will start off with some initial constraints that may need to be taken into account.

Due to the extent of this project and the multiple pill sizes available today in the market this project will assume that all pills are of one equal size due to simplicity of the project, even though the issue with different pill sizes and their inclusion in the pillbox will be accounted for in the future improvements section further on in this document. The pillbox will need a software to manage the device remotely, from an Android/iPhone app to a PC application. The development of this software is considered out of the scope of this project due to the time constraint this project has.

3.4 PERFORMANCE DESIGN AND ISSUES

Product size and weight

As portability is a main issue to prioritize in this product the key idea will consist on a pillbox large enough to hold a reasonable amount of pills and the necessary electronics and equipment such as batteries which allow it to function properly while being at the same time small enough for it to be handheld and making it possible for an elder person to carry it with them all day long without being a drag for the user. This last point implies it has to be lightweight accounting for the full pillbox, making it essential to be constructed fully with lightweight materials. Final size and weight will be estimated during the conceptual model section and measured in the schematical model section.

Aesthetics and ergonomic requirement

The pillbox will be designed with portability and ease of use as key aspects in mind, therefore an ergonomic design which could adapt to a senior's hand is desired. Also, as seniors may have diminished sight one or two big coloured buttons from which the seniors can operate all the functions they need to without the use of a screen will be a feature to take into account when designing the final model.

Round or cylindrical form of the pillbox has been concluded to be optimal as it may be easier to design both the technological and mechanical inner workings of the pillbox in a cylindrical compact form as a rectangular form may result in a bigger container when trying to distribute the circuits and pill containers. In spite of that rectangular models will also be taken into account when the conceptual model is designed.

User/operator features

As the target audience for this product will be people in old age the product is required to have a very simple operating mechanism as most seniors have difficulties interacting with new technologies, therefore it is imperative that the operating mechanism is as simple as it can be, as a complicated mechanism would result in an ineffective product.

A foreseen problem the product is prone to have is the training required by the caregivers to manage and refill the pillbox. The design phase will try to work around this problem by simplifying as much as it can the pillbox management systems, but, as it is an important aspect of the pill, the product will come with a simple management and refilling program which will interactively help the caregivers through the process of refilling and managing the pillbox without no or little initial training.

Wireless connectivity is essential in the pillbox as wireless management is a key feature to simplify pillbox everyday management by the caregivers.

Performance requirements

Performance for this product will be optimized by tuning the different characteristics the pillbox ought to have regarding the environment and usage through which it will be working. The pillbox is intended to have a continuous usage, even when the pillbox is recharging or being refilled by a caregiver. Pillbox's battery will be intended to last as much as possible taking into account size and the different internal circuits it will have to power as constraints. Battery will be fitted within the pillbox and the product will come with a charging base/cable.

Product's shelf life will only be specified by the battery's shelf life only in the case that comes pre-packaged with the pillbox, therefore it will be recommended to store batteries and pillboxes separately. As mentioned before, the electronic equipment and the plastic which make the pillbox can last for years in a dry safe storage.

Product service life will greatly depend on the attention and care paid by the residents when using it. To expand this service life the product will be designed so that it may prevent shock loadings such as falls and bumps using by plastic as main material which may help to absorb some shocks while being relatively cheap. Product service life will depend on the battery life also, but allowing the centre/owner to replace its battery in an easy way once the battery is becomes diminished. Also a pillbox may be reused. Pillbox in hospitals may be given to patients for the duration of their stay in the hospital, once they leave the hospital that same pillbox may be given to a new patient entering the hospital, thus allowing the hospital to work in an optimal way.

Reliability and quality requirements

Quality of the pillbox has to be assured as the service the product offers is a medical service, therefore even the smallest error could result in fatal consequences. For that reason each one of the materials of the product as well as thorough testing of each one of its working modes and safe usage will have to be conducted in order to prevent any error from happening.

Safety requirements

Product will not be completed unless all the pertaining safety tests of the product failing or bugs happening in the product's code are passed.

In spite of the many tests, the product may still be subject to human errors (caregiver refilling the pillbox with the erroneous medication). Misuse/abuse of the product has also been taken into account as far as the scope of this project goes (residents may not access the pill compartment or let the pillbox dispense more medication than the one pre-programmed by the caregiver, but this does not prevent the caregiver programming erroneously the medication or confusing it as stated above or the resident throwing the pillbox till it breaks or using it in any other harmful way for himself or the people surrounding him).

Expected product service environment.

The pillbox will be designed to be used mainly by hospitals and retirement homes, therefore it can be concluded that no extraordinary or special conditions in temperature or pressure will be taken into account.

Plastic used by the pillbox can withstand reasonable amounts of heat, cold, pressure or humidity. The circuit components inside the pillbox can sustain a range of temperature wide enough to allow the pillbox to work normally in every condition possible inside the hospital/retirement home (Arduino microcontroller, main circuit board inside the pillbox can sustain temperatures from -30ºC up to 85ºC). The equivalent happens for humidity and pressure, no extreme conditions which alter the normal course of the pillbox will be encountered and all the circuitry will be encased tightly within the pillbox so that no dirt or similar may enter and stop the board from working.

Product maintenance and replacement requirements

The product will be designed around the idea of simplicity and reducing price not by reducing the quality of the components but by using parts which can be easily made or bought (open-source circuits and 3d printed parts) and avoiding the use of any special machinery to assemble the product. This will result in a very cheap price when a pillbox becomes broken, as replacement of the parts by the staff of the hospital/retirement homes will be impossible due to the size and distribution of the parts contained in the model.

As it is crucial that no errors occur during the product life cycle, the program intended to help the caregivers refill and program the pill dose times in the pillbox will also include a function to test each pillbox while being refilled.

Material requirements

Materials for this product will consist in plastic used by 3d printers as it will cut the production costs greatly and combined with a cylindrical shape it increases the product's resistance against shocks, falls and bumps.

Recycling and safe disposal requirements

The designed product will be encased inside the electronic waste or e-waste. The recycling of this products will be done by the hospitals and retirement homes. Due to their nature, those organizations produce high quantities of residues of many different kinds (ranging from organic matter up to radioactive matter produced by chemotherapies as well as old electronic equipments), therefore they have special contracts with recycling plants, therefore solving the problem of recycling the pillbox as electronic waste. [19]

Manufacturing process requirements and limitations

No special measures have to be taken when manufacturing the product apart from any special limitations imposed by the different countries if necessary.

Product packaging requirements.

Packaging will consist initially in boxes made out of recycled cardboard to be as environmentally friendly as possible. The product is designed so that the hospital/retirement homes require an initial deployment by part of the producing company, therefore the packaging should be as optimal in functionality as possible, as the boxes are going to be mainly managed and operated by the producing company's employees.

Inside the packaging the product will come in three separate parts, the main pillbox without the battery, the battery in a proper package and the charger.

Patents

In the scope used in this project, which will mainly be national grounds (Spain) the following patents need to be checked once the initial conceptual design of the pillbox is designed:

- Patent ES-1044714_U [20]
- Patent ES-1068942_U [21]
- Patent ES-1064960_U [22]

3.5 MARKET ISSUES

Potential customer base

Hospitals and retirement homes will be target customers to buy pillboxes to optimize the time of the nurses and caregivers in those centres. Even though those centres will be the customers the end user will be the residents and patients in need of taking pills.

Depending on the results obtained after testing and researching what is the impact and implementation in the nursing centres it may be considered to tap into seniors private homes with the pillbox in a near future.

Market constraints on the product

Main expected market constrains focus mainly on the possible size of the market, as the product deals with large hospitals and retirement homes where any change to implement may advance slowly and with many detractors against it, therefore starting small and advancing towards more greater centres once the effectivity of the pillbox has been proven by third-parties will probably be the best strategy to take.

Expected product competition

Main expected competition will be the existing one examined throughout the state-of-the-art section, which are all different model of smart pillboxes. In spite of the different models there are, their penetration in hospitals and retirement homes is actually minimum and the product being designed in this document copes with a problem any of them solves, letting the pillbox organize the pills instead of the caregiver/nurse.

Target product price and volume

While product price is expected to be reasonably low thanks to the materials and procedures used to create the product the final price will be established in the business plan section of this document, although a estimation of it being around or less 50 euros may be a prior good estimation for each unit of the product.

Expected distribution environment

Once produced, the processor will be programmed with the code, the circuitry and the physical and mechanical parts will be assembled together. Packaging box will consist in pillbox, battery and charger hold inside the box in different compartments.

Boxes will be stored till the implementation into a hospital/retirement home is required, then deployment of the pillboxes will be done via the producing company's employees and a course for the caregivers/nurses in charge of managing the pillboxes will be offered as well as reference materials and online help, as it is vital the people in charge of refilling the pillboxes are correctly prepared.

4 CONCEPTUAL DESIGN

4.1 BEHAVIOURAL DESCRIPTION

4.1.1 Model description

Pillbox will work under three different working modes: user, charging or refill modes. The behaviour the pillbox will show in each of this modes will be detailed in the next subsection "Different working modes".

Pillbox will possess the physical characteristics below:

- A big, easy to see button which will be the only one the seniors will have to use during their time as users.
- One or various LEDs to signal different working modes or notifications.
- A small simple speaker which will be used as an acoustic notifier.
- A small hole from which the pills will be dispensed at the correct time.
- One or more power/data ports from which the pillbox will be charged and some other functions may be implemented.
- Small hidden buttons for caregivers/nurses to manage the device, used to load the pill program and refill the pillbox.
- A removable cap from where to access the pill compartments, allowing for a quick and efficient refilling.
- A charger specially made to charge the pillbox.
- A program which will allow management functions for the pillbox.

4.1.2 Different working modes

The pillbox will possess three main working modes:

4.1.2.1 Normal mode

Normal mode is intended to be the default working mode when the seniors are using it. During this mode the pillbox remains in standby till a pill/dose has to be taken by the user.

Once a pill is due the pillbox will emit acoustic and visual notifications via the internal speaker and the LEDs. The notifications will last for a short amount of time which will be defined later (around 10 seconds) and then emit a bleep each 10 seconds. The LEDs will remain lit up till the senior interacts with the device and takes the pill. User will have to press the button and then the pillbox will automatically dispense the pill, shutting off all the visual and acoustic notifications once the button is pressed and the pill dispensed.

If an amount of time (to be defined in later design stages) passes without the user pressing the button the pillbox will emit a wireless notification which will alert the nurse/caregiver in charge that there is a pill pending to be taken by the user of that particular pillbox.

4.1.2.2 Charging mode

Charging mode will initially be similar in every possible way to normal mode for the exception of a different colour led which will be on to indicate that charging is underway.

Charging mode will only happen when the pillbox is inserted into is charging base.

4.1.2.3 Management mode

Management mode will be engaged when the pillbox is being managed by the management program installed on a computer. Connection will be done wirelessly.

Management program may:

- Reprogram the pillbox with the new pill schedule.
- Activate the refilling sequence, for which each of the compartments of the pillbox will become accessible while the program will indicate which pill type corresponds in each compartment so that the caregiver/nurse replenishes it.
- Run the test function throughout which the nurse/caregiver will check the correct operation of the pillbox.

Once the device enters management mode the charging LED will blink twice at the start to indicate the pillbox has established correctly the connection with the managing program.

4.2 PHYSICAL DESCRIPTION

4.2.1 Concept art & sketches of the different proposed models

Sketches for various different models were made. These sketches were merely drafts with different shapes and accessories in which the sought characteristic were implemented in different ways. Each model was sketched after a brainstorm in which the sketch was initially designed and later was drafted in a sheet of paper. A total of 5 different models were finally created.

After discarding the majority of them for obvious flaws only two sketches were taken into account as a possible starting point to start designing the first prototype of the pillbox, model A and model B, of which only one will be the one used as a starting design to create the final prototype. These two models will be analysed in the next sections, with figure 15 and figure 16 being the concept sketches of these two models.

4.2.1.1 Model A

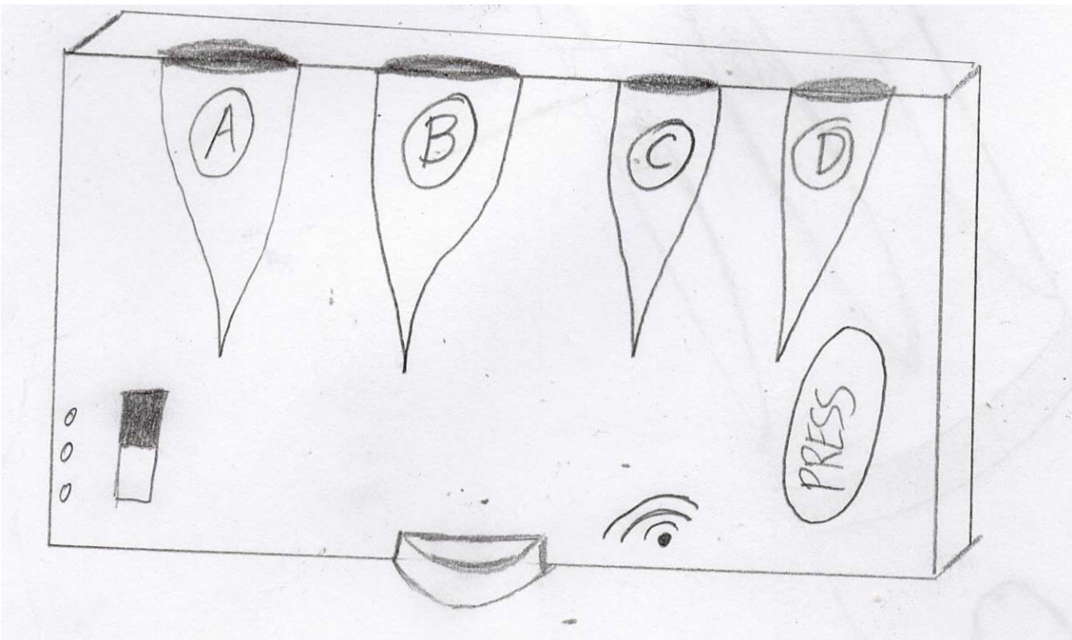


Figure 15**: Model A concept pillbox**

4.2.1.2 Model B

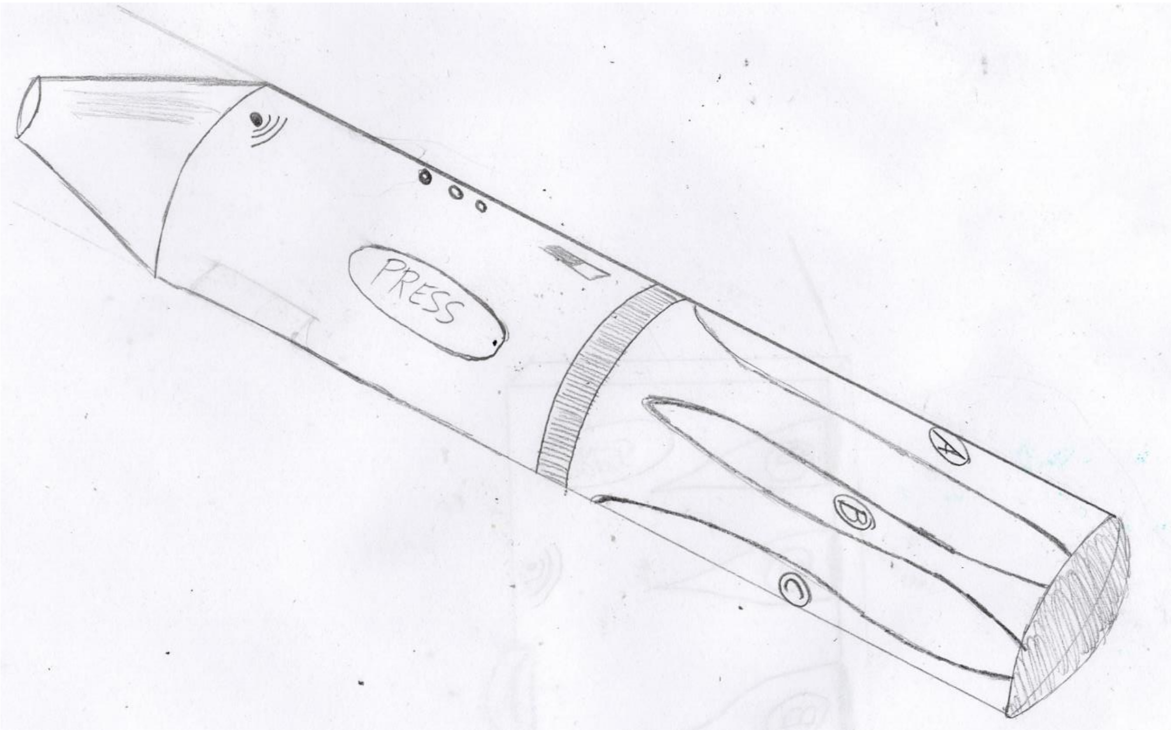


Figure 16: Model B concept pillbox

4.2.2 Description of each of the models

4.2.2.1 Model A

Model A consists on a rectangular box design in which there are four pill compartments (labelled A to D), a big visible button in the lower right-hand side of the pillbox and a smaller switch on the lower left-hand side. Leds which indicate different working modes and notifications are located at the lower left-hand side of the pillbox, right beside the small switch. Refilling will be made by opening each one of the pill compartments at a time and when the pillbox had to dispense a pill, the mentioned pill will fall into the tray found at the lower centre in the front side of the pillbox.

Its shape was conceived thinking the idea of locating all the circuits in the back of the pillbox (circuits usually come in rectangular form, as nearly all the Arduino boards) and by seeing in a clear way how many pills were left in each of the compartments. Additionally its shape resembles those of the everyday pillboxes, which is believed to make the elders include it in their lives quicker as it is a shape they are familiar with.

4.2.2.2 Model B

Model B consists on a cylindrical tube with a hole in the lowest part from where it is intended for the pills to be dispensed at the indicated time. On the lower section of the tube a big button to be used by the senior users will be located and beside it the different notification LEDs as well as a smaller switch to be used by the nurse/caregiver in charge for different advanced options such as management, refilling the pillbox, etc…

On the upper section of the pillbox the different pill compartments are located, visible from the outside and are positioned around the outer rim of the section at equally spaced distances from the centre of the pillbox.

On the uppermost part of the pillbox the lid which will be used to access the pill compartments for refilling the pillbox can be found.

It was conceived in a cylindrical form as that particular form resulted more ergonomical and easier to carry for the user rather than a rectangular pillbox as well as it would simplify the inner mechanics of the pillbox by positioning the compartments around the pillbox's outer rim in a circular way.

4.2.3 Conclusions

Finally became clear that the model which would continue onto the next production phase will be model B. This decision was made when an unexpected concept problem arose when studying each one of the models in further detail, which method was needed to make it possible for the pillbox to give only one pill and no more to the patients without using advanced technology which could make the both the production and the final price of the pillbox greatly rise. A purely mechanical mechanism using only plastic 3d printed parts was devised which could only be implemented in a cylindrical object and not in a rectangular (this mechanism will be examined in the ensuing sections of this document). This constraint as well as the advantages model B presented was the main motive to finally discard model A and continue the consequent phases using model B as a concept model.

4.3 Block diagram

A simple block diagram was created to use as a template when designing the different parts (both physical and software) of the pillbox, this diagram can be seen in figure 17.

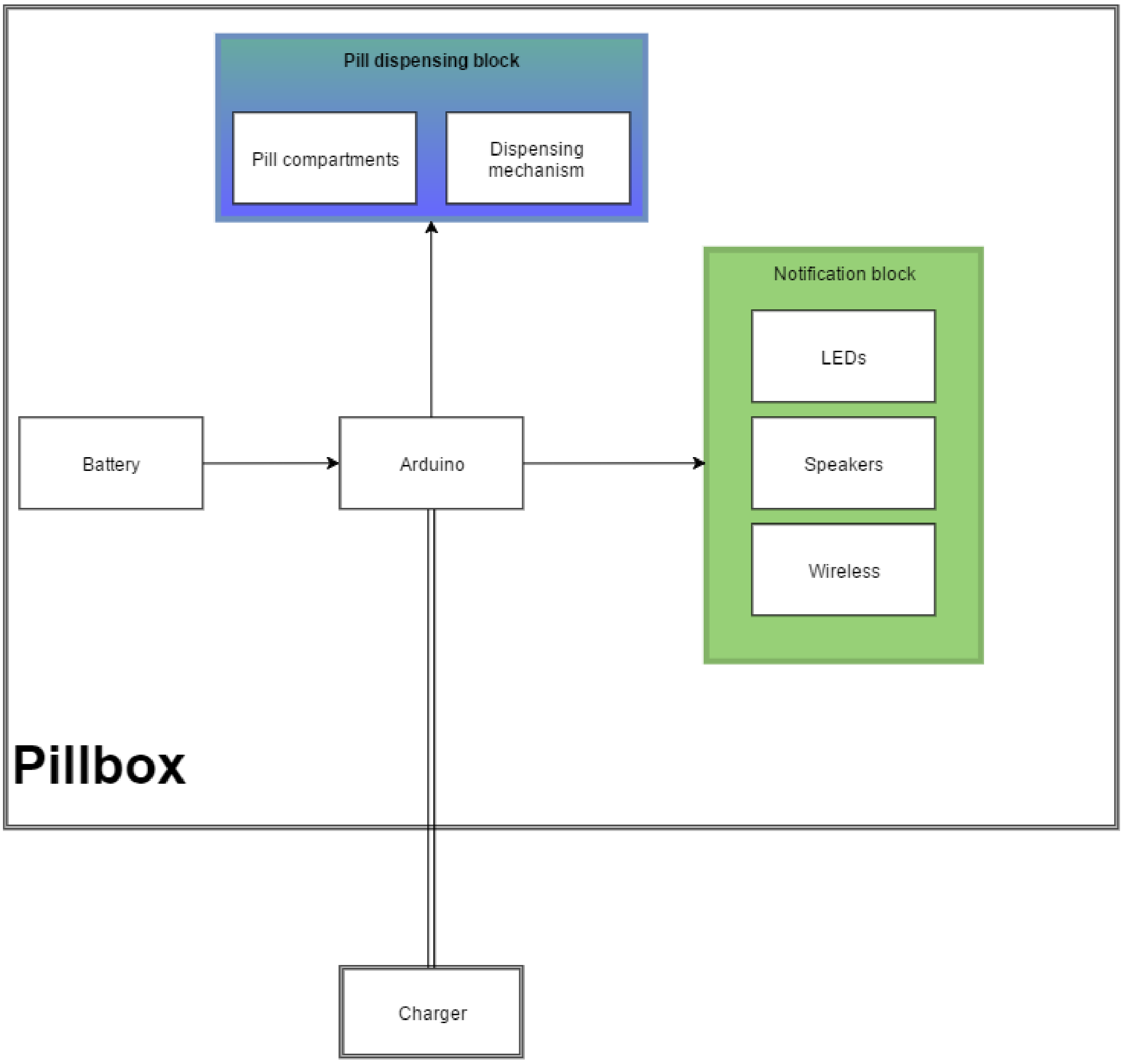


Figure 17: Pillbox's block diagram

Arduino board will be the "brain" of the pillbox, managing each one of the different areas and will be powered directly with the battery. The notification block will be in charge of emitting sound and light signals for the patient to know a pill has to be taken and wireless signals to the caregivers. In a separate block the pill compartments and the pill dispensing mechanism will be also managed apart by the Arduino. Finally an external charger will be in command of charge the battery directly through the Arduino.

4.4 Patents

After the creation of the conceptual design it becomes mandatory to check if there exists any patent which our conceptual design may interfere with. As stated in the Product Design Specification there are three main patents filed for a pillbox in Spain.

Patents ES-1044714_U [20] and ES-1068942_U [21] shown respectively in figure 18 and figure 19 are clear that do not interfere with the Smart pillbox designed in this document as they lack any sort of circuital device and are completely different in shape, size and functionality.

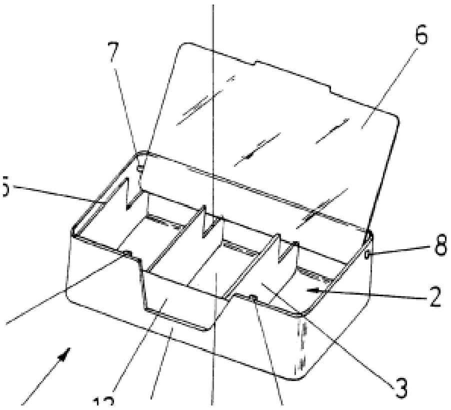


Figure 18: ES-1044714_U patent

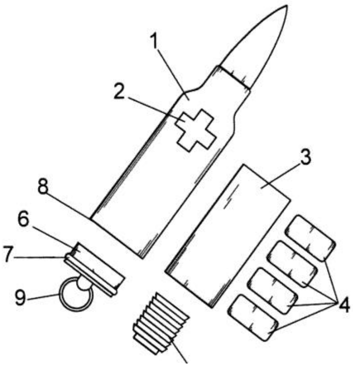


Figure 19: ES-1068924-U patent

Patent ES-1064960_U [22] shown in figure 20 does have some resemblance as it is a circular pillbox with different compartments for pills positioned in a similar way to the pillbox being designed and has some technology which allows to notify the user when to take the pills. Although it has some points in common the pillbox being designed, it has the ability to dispense for itself the necessary pills at each moment as well as to send wireless notifications to the people nursing the pillbox's owner. Moreover the compartments in the patent are designed to hold the different pills for a day/time of day, whilst the pillbox being designed holds in each compartment the pills of the same type and lets the pillbox sort them out at the precise time of the pill dosage.

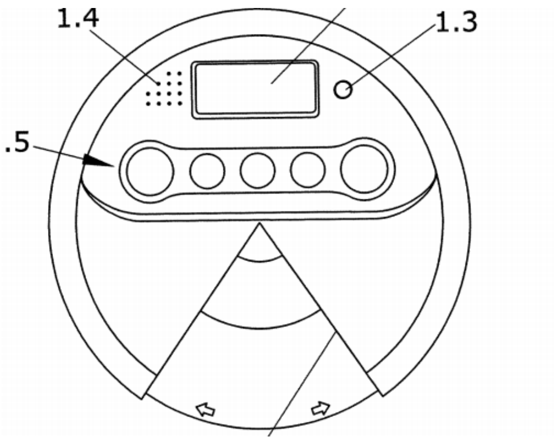


Figure 20: ES-1064960-U patent

5 PILLBOX DESIGN

5.1 PILLBOX MODULE DISTRIBUTION SCHEME

Different modules have to be defined as stated in section "Block diagram" in the conceptual design. In order to start the final design of the pillbox it is mandatory to locate each one of the modules in a specific area. These modules will be the pill refilling mechanism, the pill dispensing mechanism and the electronic module, where most of the circuitry will be located such as the Arduino, battery, switches, etc… Also the location of the pill tray where each one of the pills will exit the pillbox must be noted in the diagram.

It has been conceived for simplicity that the outer body of the pillbox will be constructed in three different parts which will later be assembled together. Pills refilling module will be done through the lid of the pillbox, therefore its mechanism will be located in the uppermost part, pill dispensing module comprises both the mechanism needed for dispensing the pills as well as the pill storage, located in the centre part. On the lower part both the electronic module and the pill tray/hole through which the pills will be dispensed to the user will be located on the furthermost lower part, with the pill passing through the dispensing mechanism through a conduct to the pill tray.

Figure 21 represents a minimalist cross-section of the pillbox locating the modules inside the pillbox's body.

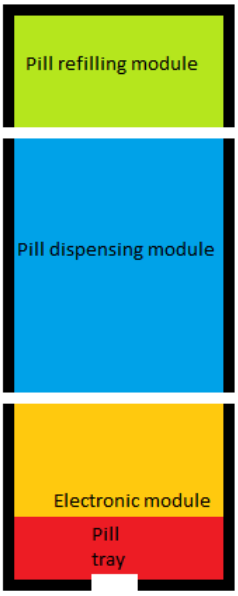


Figure 21: Pillbox's module location scheme

5.2 PILL REFILLING MODULE DESIGN

5.2.1 Definition

This module will be in charge of allowing the person in charge of refilling the mechanism to refill adequately the pillbox with a minimal chance for errors in the process. Pillbox will only have one entry through which the pills will be introduced inside the pillbox but many different pill compartments (one for each pill type), the main function of the pill refilling module will therefore be to successfully pipe the different pills introduced in the pillbox to the correct compartment in the pill storage.

5.2.2 Operation

The opening will be located in the top surface of the pillbox. Refilling will be done with help of the pillbox's management software. The person in charge will be prompted to insert a specific pill type in the pillbox and press a key in the computer when that action has been completed and the compartment for that pill has been fully replenished. Once the user hits the key in the computer the program will ask to refill another pill type and press a key and the action will be repeated until all of the pill compartments have been refilled.

5.2.3 Mechanism design

Pill storage will consist on a cylindrical tube where all the different compartments will be contained in the form of hollow pillars, which will be the ones holding the pills in them. In order for the pills to access each pill compartment a servo installed below the lid which will be holding the cylindrical tube will align via a command sent by the computer each one of the compartments with the entry hole, making the pills fall directly to the selected compartment. Once the compartment in use is filled the servo will rotate the tube as to align the next compartment with the entry hole.

5.3 PILL DISPENSING MECHANISM DESIGN

5.3.1 Definition

This module will take action each time a pill needs to be dispensed as it will be the module in charge of making the pill fall in the pill tray at the moment the Arduino sends the signal.

5.3.2 Operation

Module will need to be capable of selecting the correct compartment from the different compartments of the pill storage and it is vital it only gets one pill from the compartment, as getting more than one pill could prove fatal.

5.3.3 Mechanism design

The mechanism for the pill dispensing module will use two Arduino-controlled servos, one of them located under the lid and attached to the pill storage cylinder (same servo as the one used in the pill refilling mechanism) and another one which will control an intermediate piece.

Mechanism will consist in a total of four inner 3D plastic parts including the pill storage, two of them will be rotated in different positions by the servos. Diagrams will be now used to explain the different position these parts will show the mechanism extracts a pill from a selected compartment and throws it through the exit tray or chute.

Mechanism consists in three different stages: default stage, pill load stage and pill release stage. After the pill release stage the system will reset to the default stage. The system goes through the three stages each time a pill has to be dispensed to the user.

5.3.3.1 Parts involved in the dispensing mechanism

Parts have been labelled 1 to 4 in the following diagrams for an easier comprehension:

- Part 1 – Pill storage.
- Part 2 – Pill hatch.

- Part 3 – Pill chamber.
- Part 4 – Pill pipe.

A brief overview of these parts including simplified schematics will now be presented covering the basics needed for the reader to understand the pill dispensing mechanism. This overview will barely explain the shape of the parts as the functionality behind that shape will be covered in the next section where the three main stages for the pill dispensing mechanism will be explained. For the complete scaled schematics of these parts please refer to “Parts overview” section of this document.

5.3.3.1.1 Part 1: Pill storage

This part consists on a cylinder with hollow tunnels which cover the total length of the tube distributed radially along the tube. On the centre of the part there is a circular hole where different cables for circuits will pass when the pillbox is assembled. This compartments will be the ones storing the pills until they need to be dispensed. Each compartment will hold only pills of the same type, consequently there can be as many pill types as usable compartments minus one. This is because a compartment will always be empty and not used by the pillbox, this is further detailed in the stages overview.

A view of the part is presented in figure 22.

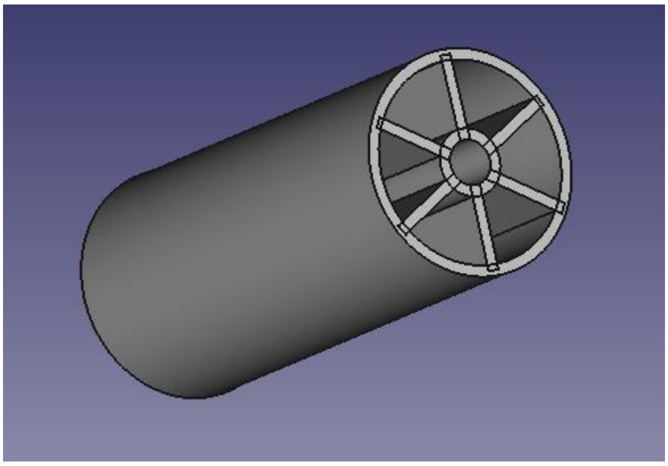


Figure 22: Pill storage

5.3.3.1.2 Part 2: Pill hatch

Pill hatch will be a fixed part which allows to communicate the pill storage part through a small hole with the pill chamber. It consists on a very thin circular piece with an opening big enough for a pill to pass through it and a small ramp connecting the opening to the upper level of the cylinder.

A small circular hole is located at the centre of the part designed for different cables to pass when the pillbox is assembled.

A view of the part is presented in figure 23.

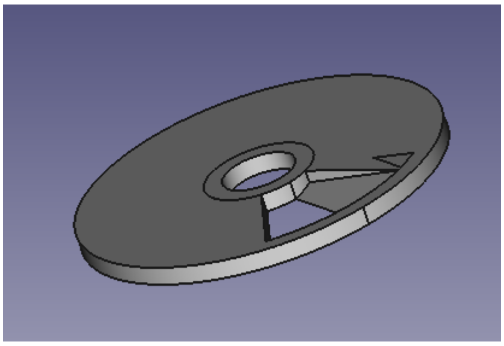


Figure 23: Pill hatch

5.3.3.1.3 Part 3: Pill chamber

The pill chamber consists on a cylinder with a circular hole at the centre and another passage beside the hole. This passage is wide enough to hold a pill in an upright vertical position. The height also is the same as the model pill which will be used in this pillbox, therefore this part can only hold one pill standing in an upright position in its storage space (the storage space will be the side passage of the part, which will be also addressed as the pill chamber from now on). Part may be viewed in figure 24.

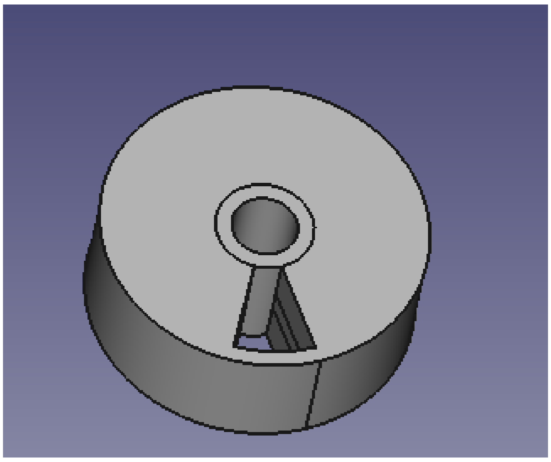


Figure 24: Pill chamber

5.3.3.1.4 Part 4: Pill pipe

The pill pipe part consists in a semi-circular piece with a large pipe of a considerable greater height than the circular part protruding from its side, connecting with a hole the upper part of the circular part with the end of the pipe. A semi-circular opening is also found at the centre of the circular part helping cables pass through the pillbox's centre when it's finally assembled. Part is shown in figure 25.

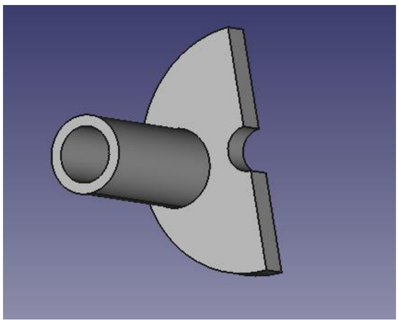


Figure 25: Pill pipe

5.3.3.2 Stage overview

During this overview the different stages will be presented with the help of simplified schematics which will aid in helping the reader understand the operational functioning of the pill dispensing system. The four parts will be presented in a minimalist way as seen in figure 26. During this explanation the location of the different passages will be marked in red in each of the parts (these passages will be the ones the pill has to traverse to get to the exit), as can be seen in figure 27.

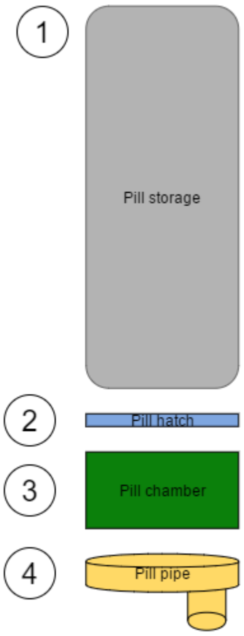


Figure 26: Pillbox refilling mechanism schematic

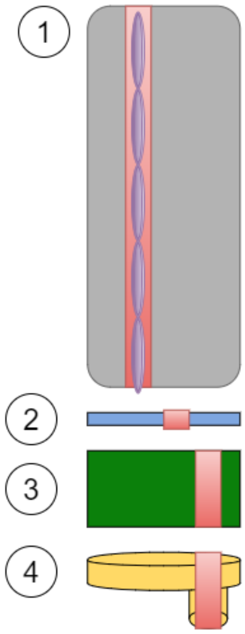


Figure 27: Pillbox refilling schematic showing pills in the cross-section

5.3.3.2.1 Stage 1: default stage

This stage is the stage where the pillbox is by default when no pill needs dispensing, the starting position from where the pill dispensing mechanism starts.

The pill compartments in the pill storage (part 1) are positioned so that they are **not** aligned with the opening of the pill hatch (part 2), therefore preventing any pills to exit the pill storage.

Pill chamber (part 3) has the chamber aligned with the pill pipe (part 4), meaning it is not aligned with the opening in the pill hatch (part 2). Parts position in this stage is shown in figure 28.

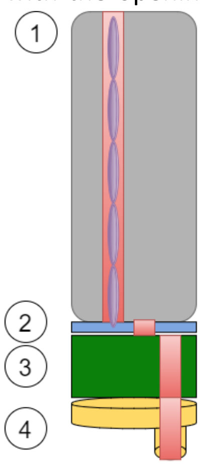


Figure 28: Pillbox position of inner parts at default stage

5.3.3.2.2 Stage 2: pill load stage

In the pill loading stage both the pill storage (part 1) and the pill chamber (part 3) align with the pill hatch (part 2). It has to be noted that as the pill chamber (part 3) is aligned with the hatch then it is not aligned with the pipe in the pill pipe part (part 4). This means the chamber is closed in its lower part (preventing any pills from falling below) and open in its upper part, which connects directly to the pill storage through the pill hatch. This causes a pill to fall from the pill compartment to the pill chamber, but, as there is only space for a pill in the pill chamber, the pill itself will prevent any more pills to fall into the pill chamber. The position of the different parts is shown in figure 29.

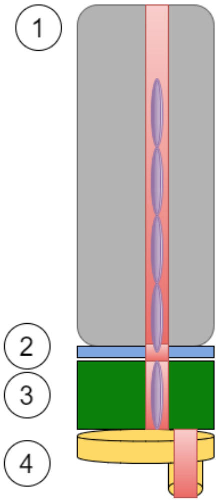


Figure 29: Pillbox position of inner parts at pill loading stage

5.3.3.2.3 Stage 3: pill release stage

When the pillbox reaches this stage the pill chamber (part 3) will move aligning itself with the pill pipe and therefore letting the pill stored in the chamber fall through the pipe, this position is shown in figure 30. Pill storage (part 1) then rotates, aligning the empty compartment of the pill storage with the hatch opening, preventing any other compartments to be aligned over the hatch. Final position is then shown in figure 31.

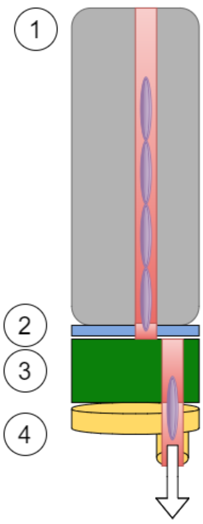


Figure 30: Pillbox position of inner parts at the first step of thepill release stage

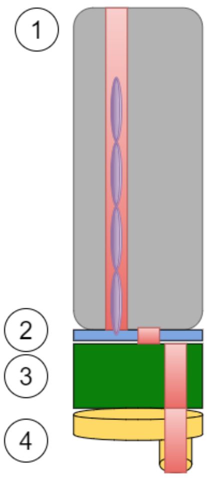


Figure 31: Pillbox position of inner parts at the last step of thepill release stage

5.4 CHANGES IN THE ORIGINAL DESIGN

After the start of the design certain issues have raised forcing the original design to be subject to some minor changes. These changes will be the location of the battery and the pill tray.

The battery was intended to be included besides the Arduino board with the main electronics but after a thorough search it has been decided to use a bigger battery which will allow for more autonomy, therefore having to be placed outside the main body.

The pill tray has been relocated to the side of the pillbox so that the pill has to traverse less distance from the pill storage to the pill tray, therefore preventing errors such as the pill getting stuck.

The pillbox's new distribution is shown in figure 32.

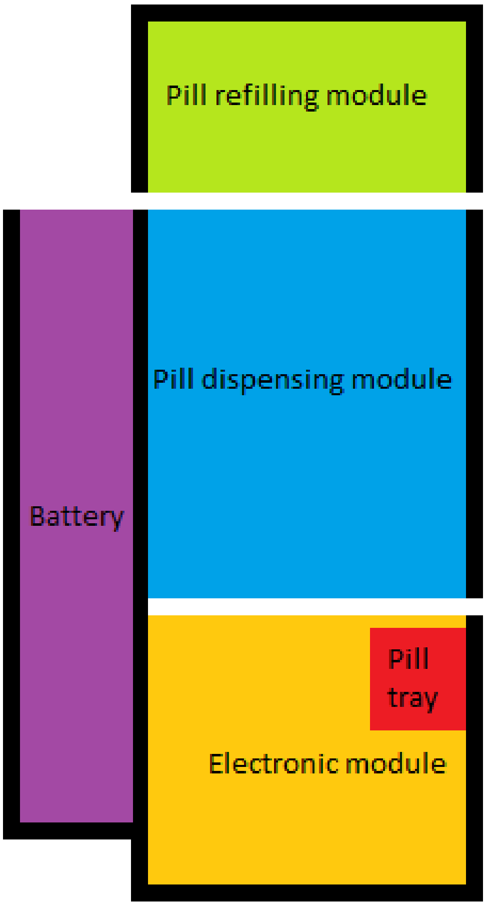


Figure 32: Pillbox module location after revision of the original idea

5.5 SIMPLIFIED SCHEME

A first draft of the different parts composing the pillbox as well as some involved components such as the servos are presented in figure 33 in a simplified view of their supposed location once the pillbox is finished. This first draft alongside the different mechanisms will be used as a roadmap when designing the final models of the 3D parts.

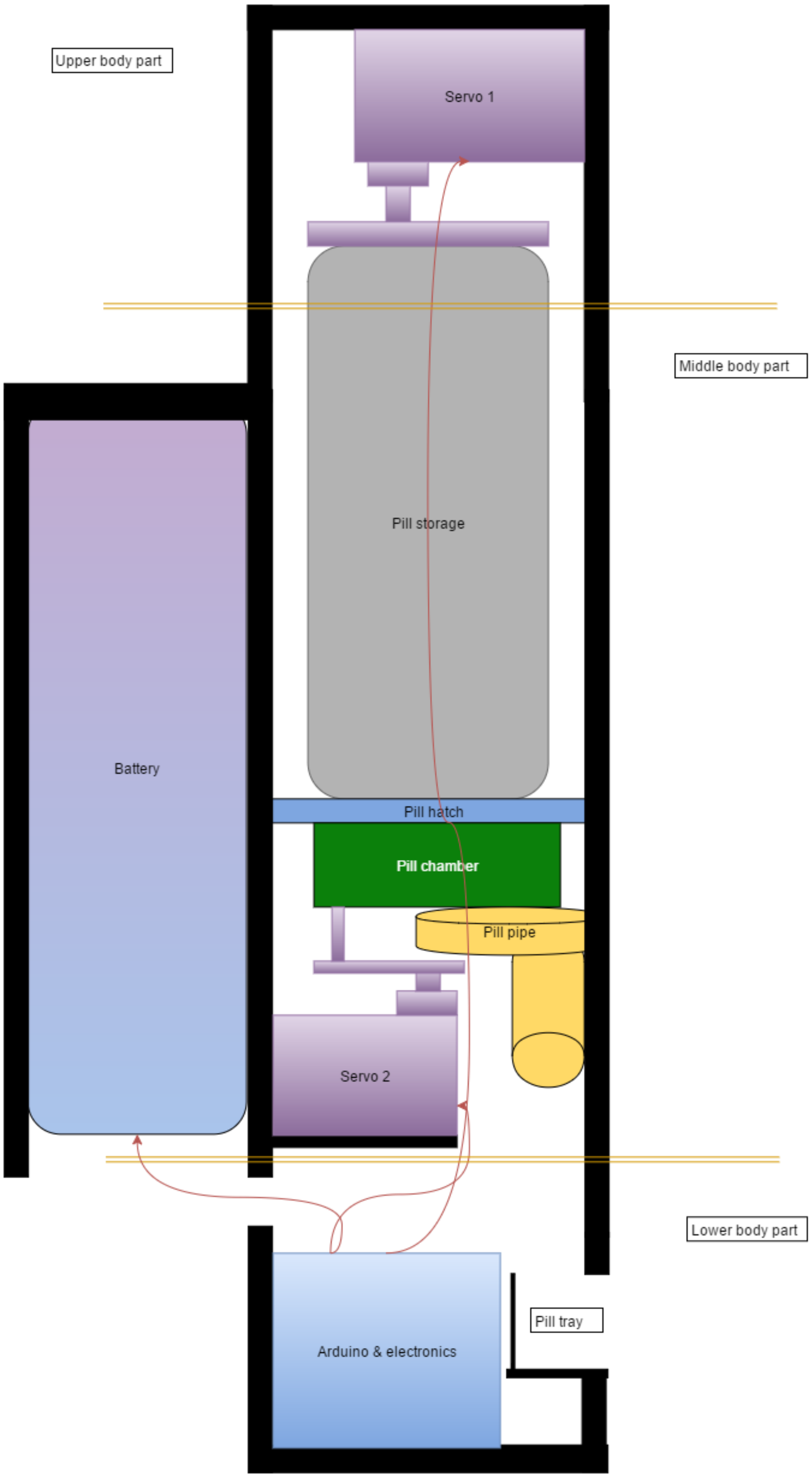


Figure 33: Simplified scheme of the pillbox inner part location

5.6 DESIGN SOFTWARE: FREECAD

The different parts will be design using an open-source software with no cost assigned to it, as it can offer a very similar functionality than other expensive programs and therefore the same results while not rising up the cost of designing and producing the pillbox. This software is the program FreeCAD [18], which is defined as a parametric 3D modeller, a program which helps to design and modify different versions of the model the user is currently working on with great ease.

Process used to design the 3D parts was to use simple shapes and make operations between them such as addition or subtraction to create the pillbox's parts. Below a simple example is used as how the process was committed:

1. Firstly two simple cylinders are created with the desired measurements (figure 34):

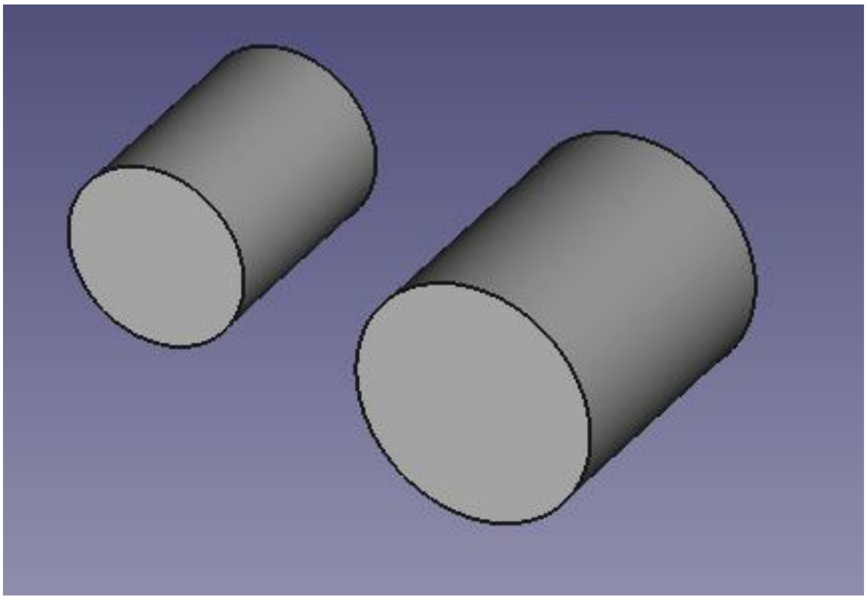


Figure 34: Two simple cylinder in FreeCad software

2. The smaller shorter cylinder is positioned inside the cylinder (figure 35):

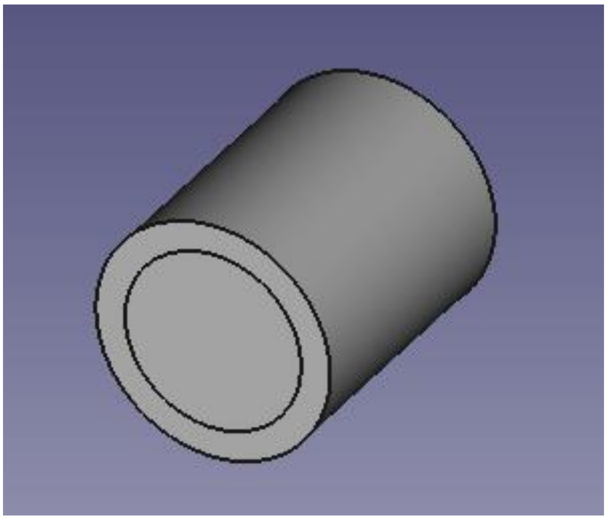


Figure 35: Cylinders have been positioned one inside the other

3. The smaller cylinder is subtracted from the bigger one forming a primitive version of the pillbox’s upper part (figure 36):

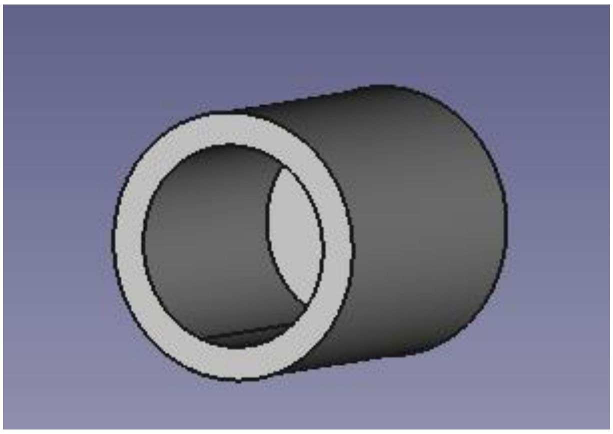


Figure 36: The smaller cylinder has been substracted from the wider one

4. Necessary details such as the hole from where the pills will entry the pillbox can be done using the same procedure by subtracting a cube to the created part (figure 37 and figure 38):

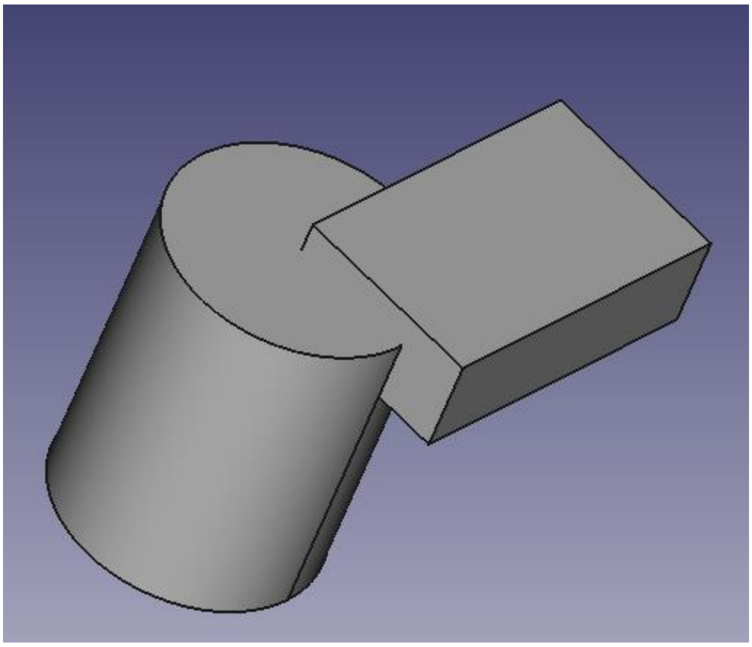


Figure 37: A cube is overlapped with the created part to make an entry hole in the cylinder

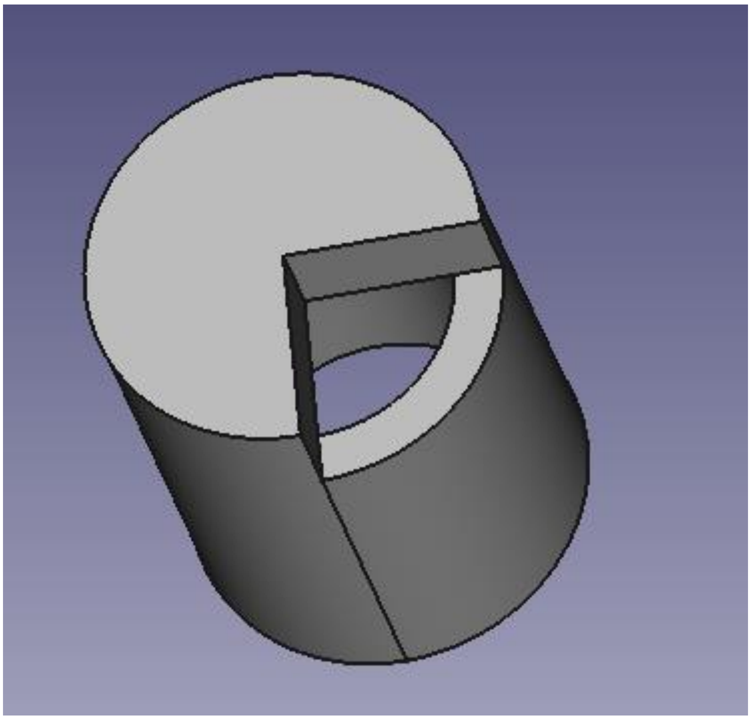


Figure 38: After the cube has been substracted from the cylinder the entry hole is created

Note: This example was done specifically for this section solely with explanatory purposes, as the finished parts are far more complex than the one shown in this section. Also different operations as fusing or intersecting shapes were also used for the pillbox's design.

5.7 PARTS OVERVIEW

The pillbox consists on a total of 9 plastic 3D-printed parts which conform both the outer body as the inner mechanisms of the pillbox. The plans for the different parts will be shown here in a reduced scale. The actual full-size plans with measurements of each one of the parts will be appended in the annex.

LID

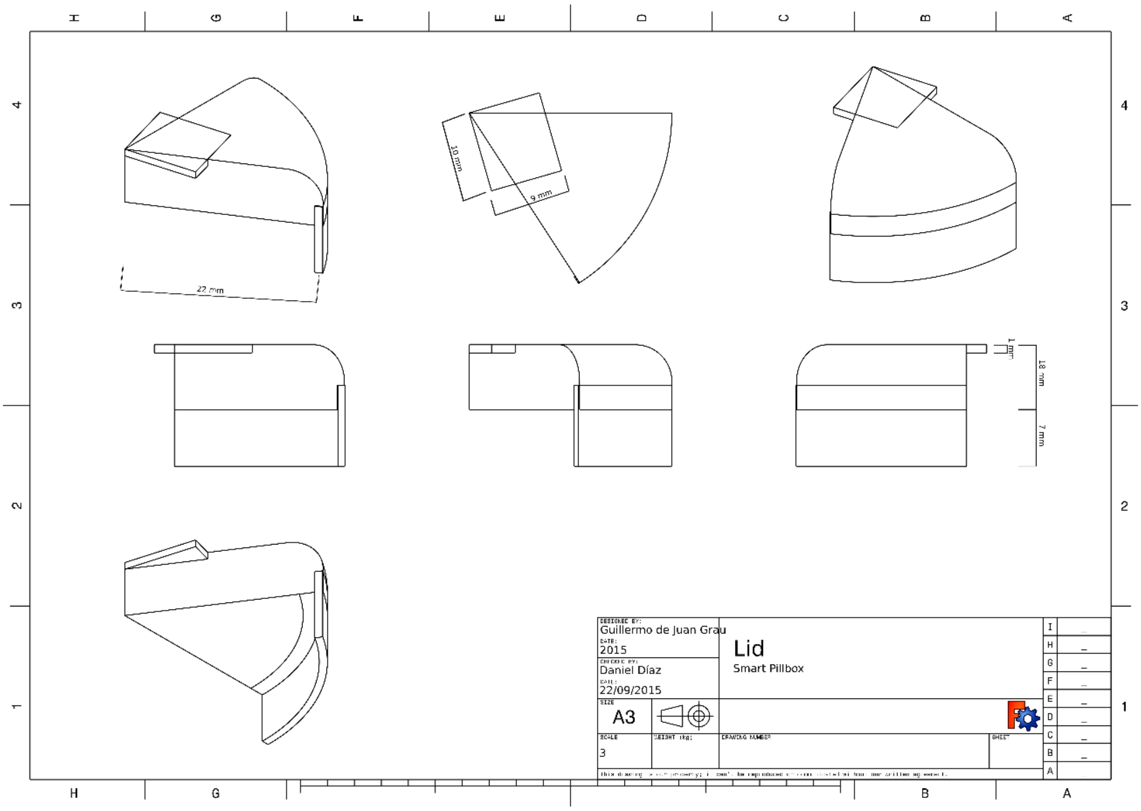


Figure 39: Scaled plan for the lid part

Lower body

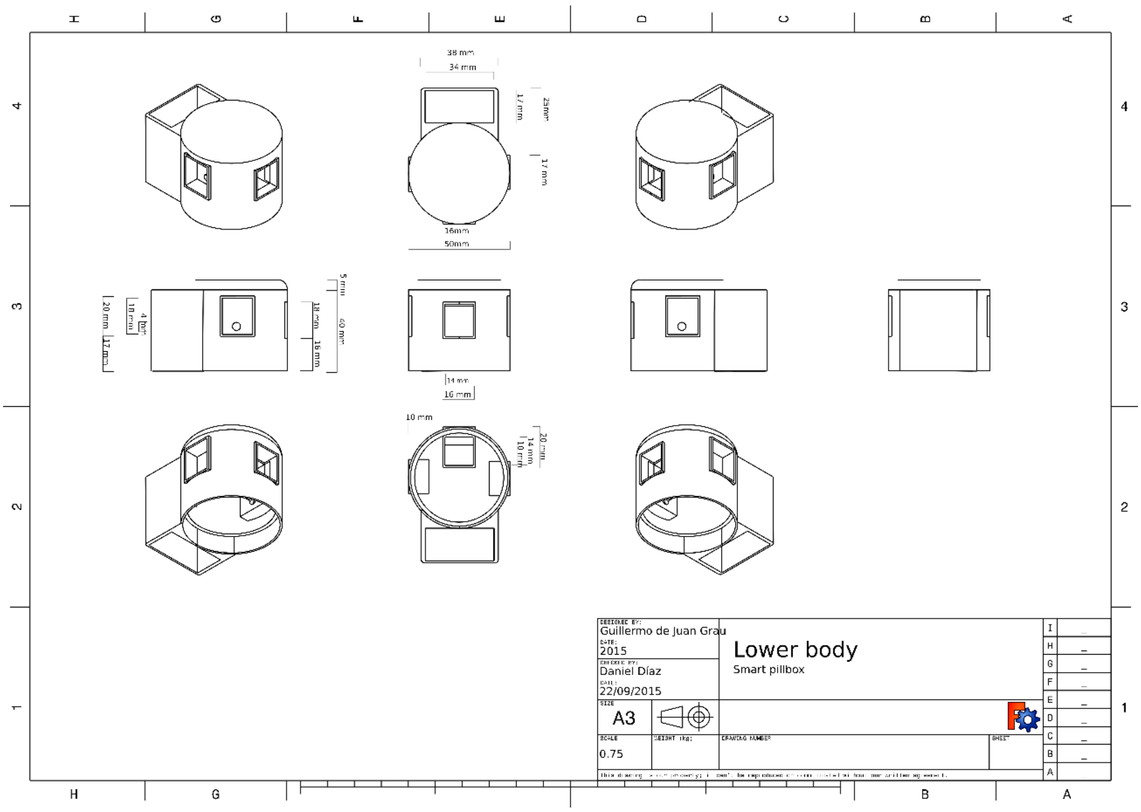


Figure 40: Scaled plan for the lower body part

Mid-body

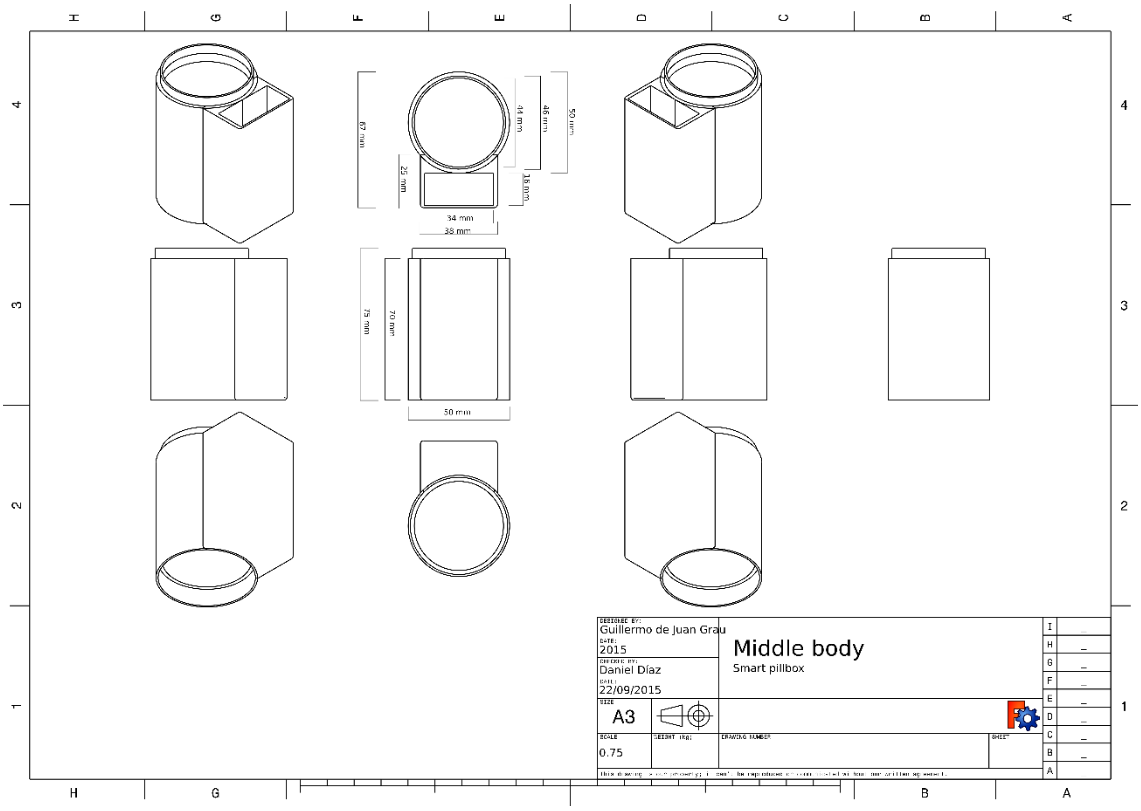


Figure 41: Scaled plan for the middle body partpper mid-body

Pill chamber

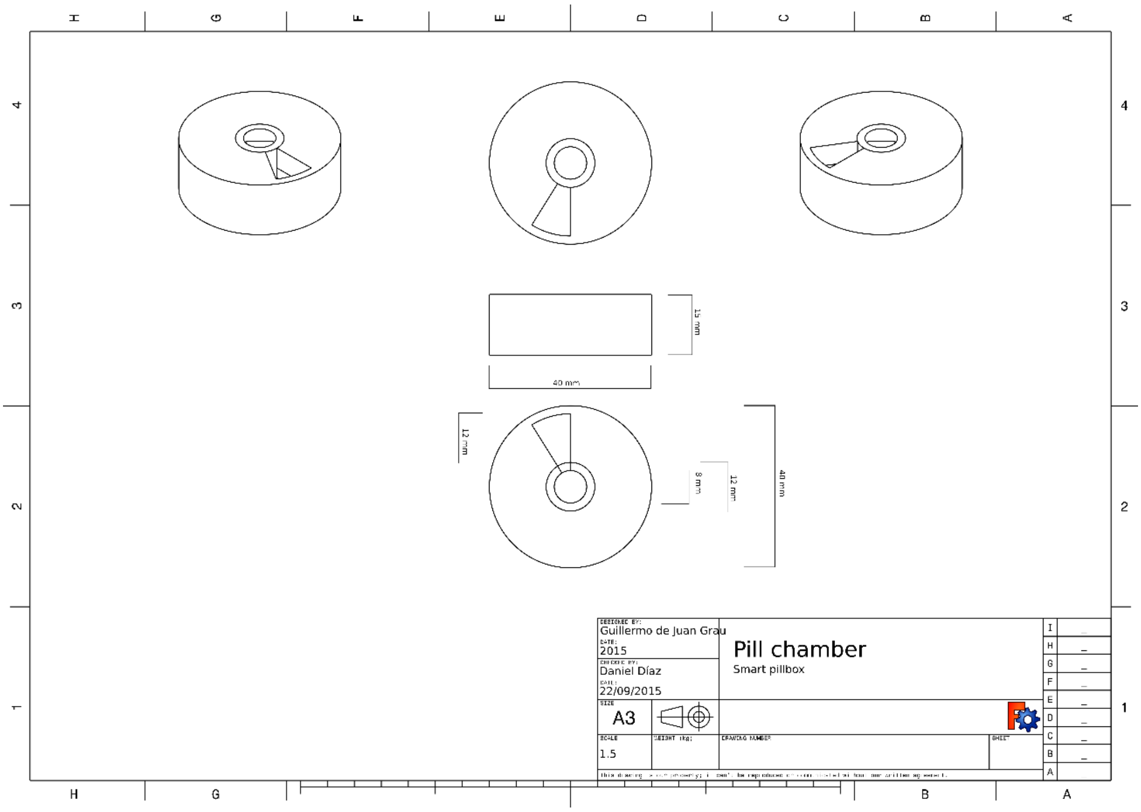


Figure 42: Scaled plan for the pill chamber part

Pill hatch

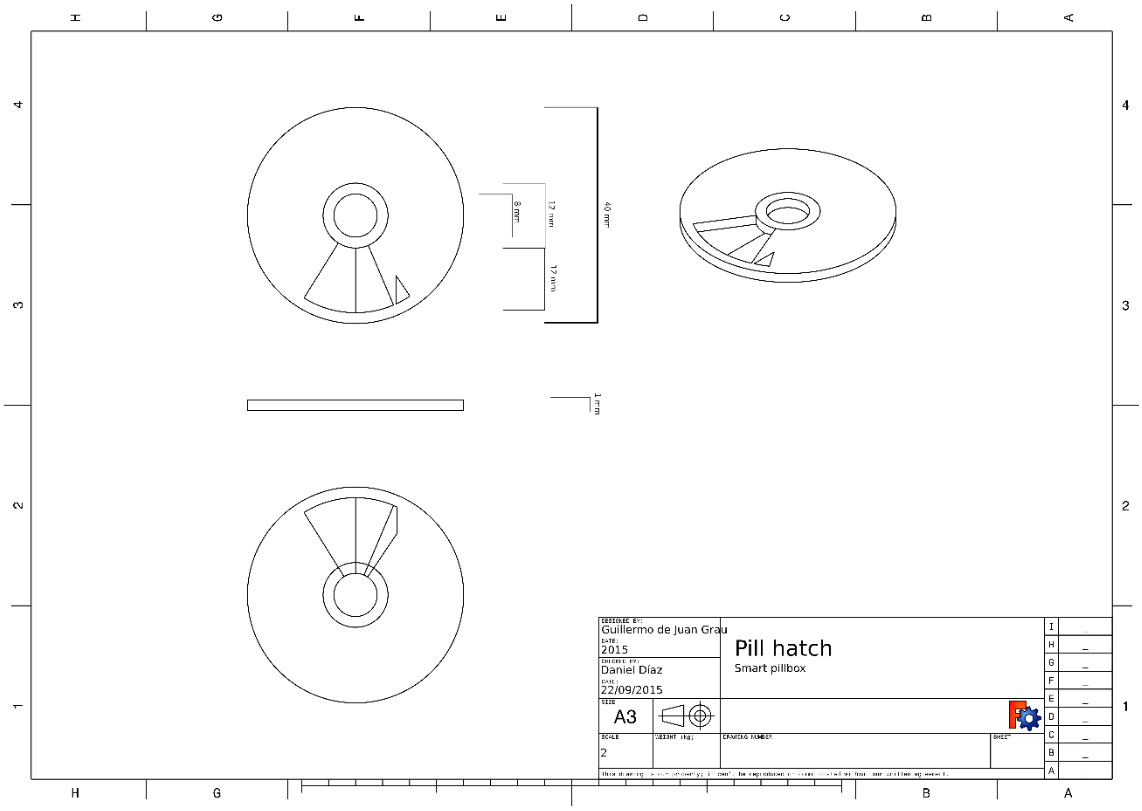


Figure 43: Scaled plan for the pill hatch part

Pill pipe

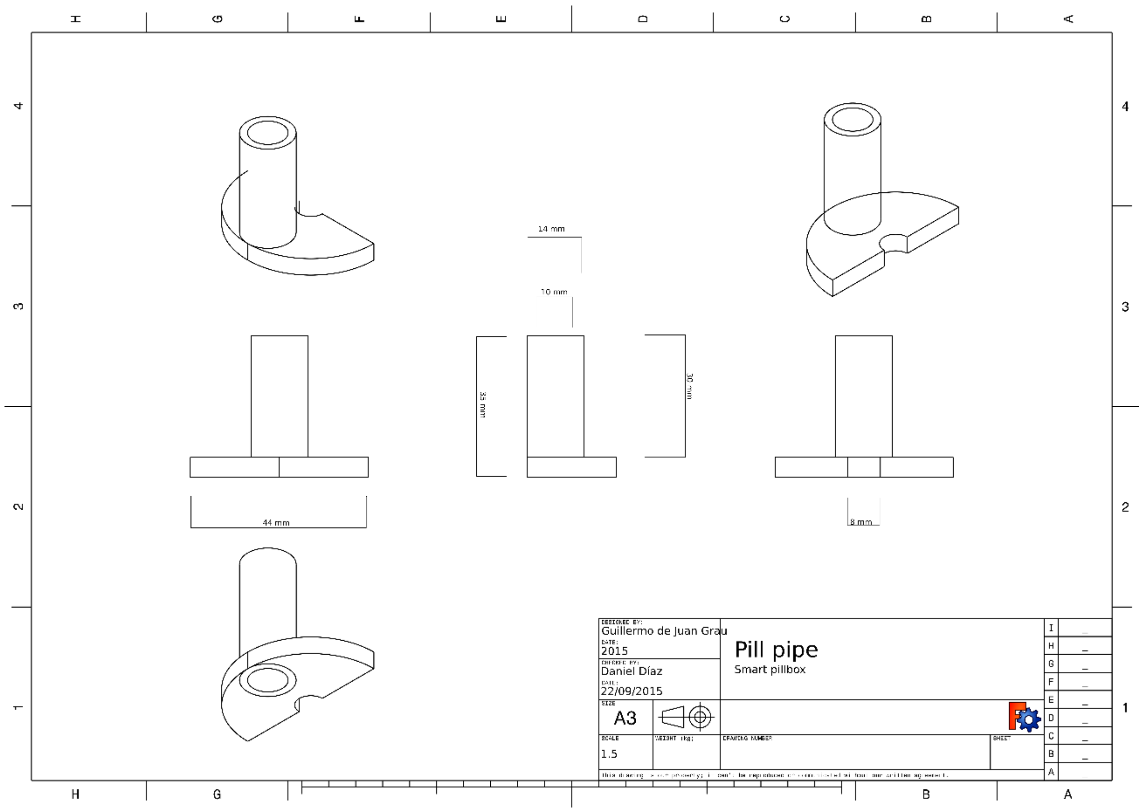


Figure 44: Scaled plan for the pill pipe part

Pill storage

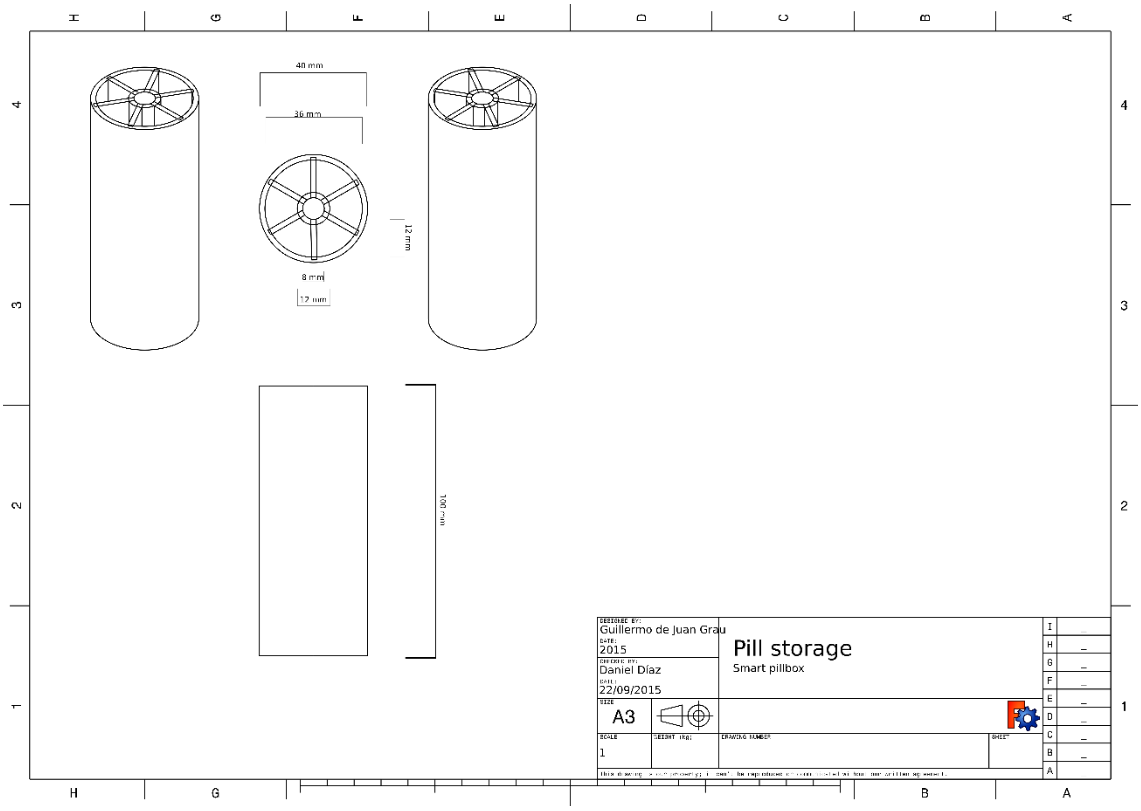


Figure 45: Scaled plan for the pill storage part

Upper section

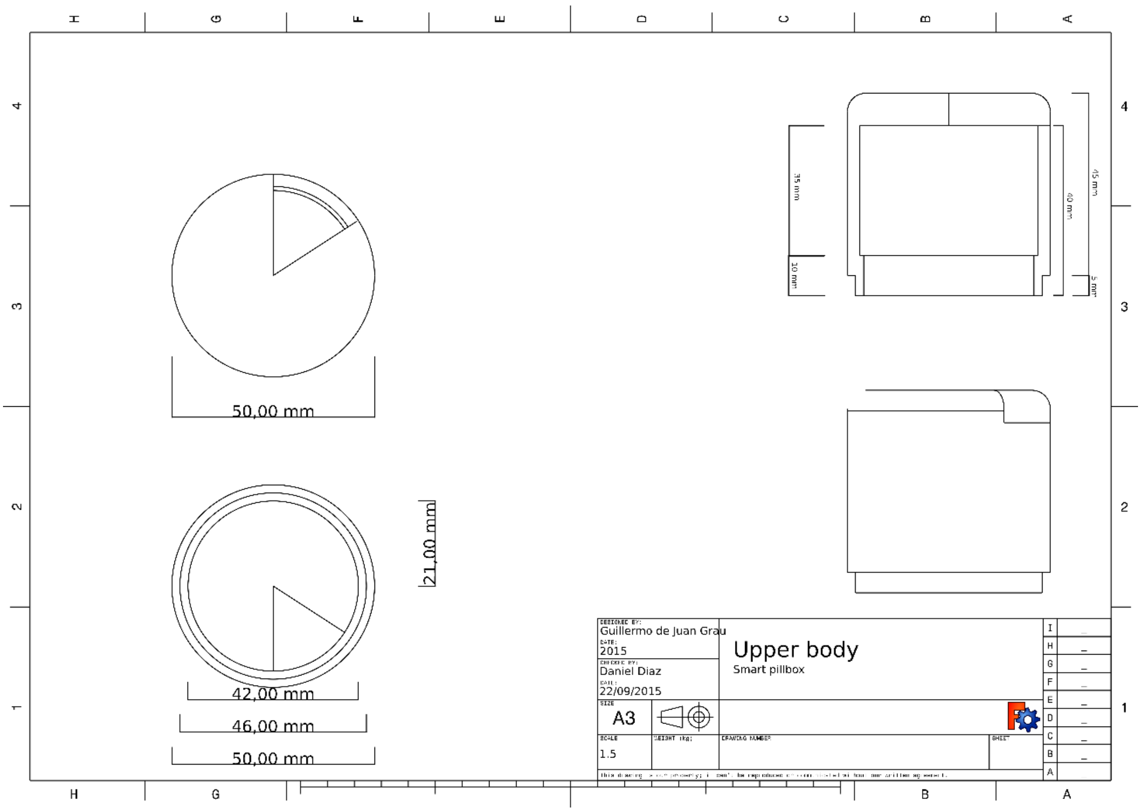


Figure 46: Scaled plan for the upper body part

Battery lid

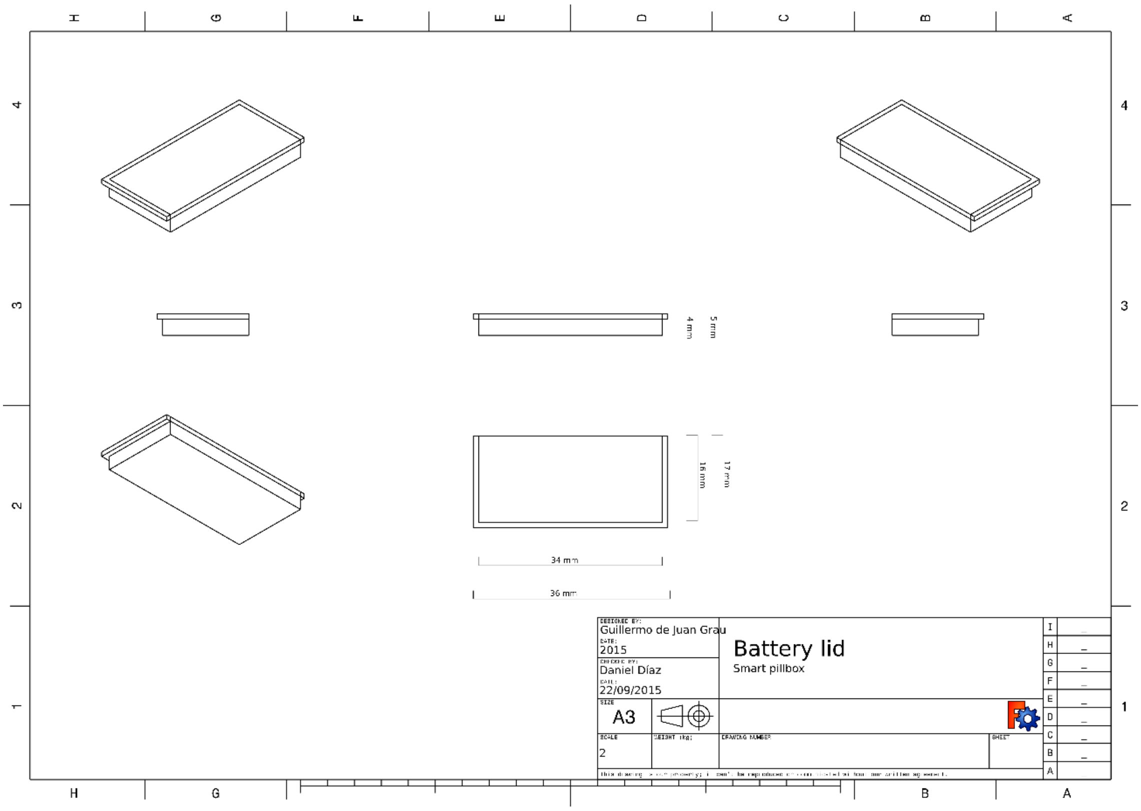


Figure 47: Scaled plan for the battery lid part

5.8 SOFTWARE AND ELECTRONICS DESIGN

5.8.1 Introduction and pillbox operation

A series of electronic devices will be installed inside the pillbox's body to achieve the desired functionality of the pillbox. An overview of the selected electronic equipment will be thoroughly examined in the "Electronics overview" section while the different design steps of the software will be shown below such as the code flowchart, the pseudocode and the final code.

5.8.2 Electronics overview

5.8.2.1 Introduction

Pillbox's internal electronics will need to perform a range of different functions, therefore a wide range of electronic modules have to be included.

In chart 2 the different functions required are shown alongside the different electronic devices which will be used to perform those functions in figure 48.

Needed functionality	Electronic device performing the functionality
Controlling the device and different electronic modules	Arduino board
Rotating inner parts of the pillbox	Servos
Light notifications	LEDs
Acoustic notifications	Speaker
Keep track of time	RTC (Real-Time Clock) module
Wireless communication	Bluetooth module
Keep logs of different pill intakes	Arduino’s EEPROM memory
Portability	External battery

Chart 2: Functionality apported by electronic device in the pillbox chart

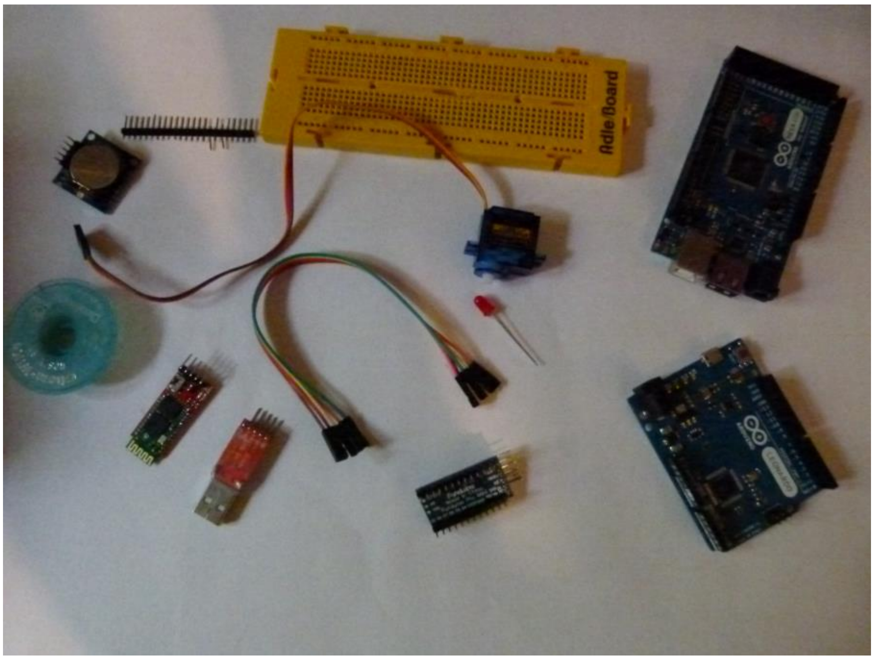


Figure 48: Some of the electronic components which have been used to test and create the electronic circuitry

5.8.2.2 Arduino

Definition

Arduino is an “open-source prototyping platform”, a microcontroller-based board with various analogue and digital input/output pins and different serial communication interfaces as well as the capability to expand its functionalities by connecting external devices or circuits to the board.

The board comes with an IDE (Integrated Development Environment) which allow the user to write a code and upload it to the board.

Implementation

There is a wide range of Arduino boards in the market as well as several other companies producing copies of Arduino board (as it is totally an open-source product) which also work with Arduino IDE and cost much less than the original boards while offering the same capabilities as the original ones.

The one selected for this project will be a Funduino Pro Mini, a version of the Arduino Pro Mini which comes at a lower price offering the same functionality as the Arduino Pro Mini board. This board needs a USB to TTL serial converter to upload the code to the microcontroller as this board “comes with the minimum of components (no on-board USB or pin headers) to keep the cost down” [15], thus the CP2102 USB to TTL serial converter was adquired.

Soldering was required as the board comes without the head pins (needed for uploading the code and powering up the device) as well as pins A4 and A5 (needed for the communication with the Real Time Clock module), therefore a soldering iron and soldering wire along with some spare pins were acquired and the pins subsequently soldered in the right places. The soldered pins can be seen standing upright from the board in figure 49 and the materials used for this soldering process are shown in figure 50.

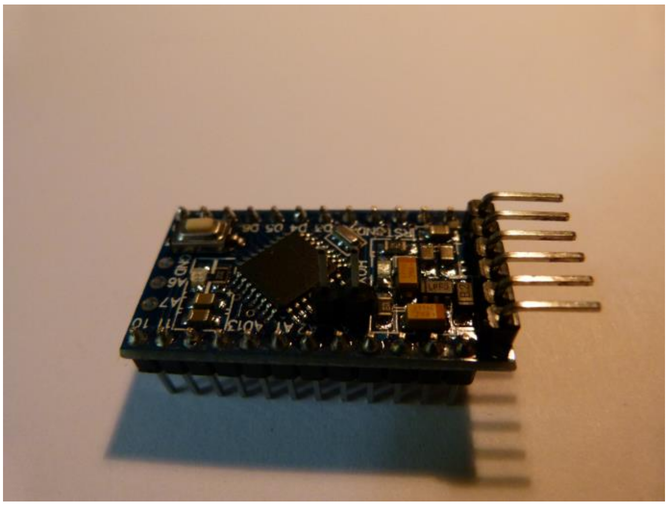


Figure 49: Funduino Pro Mini after soldering the necessary pins

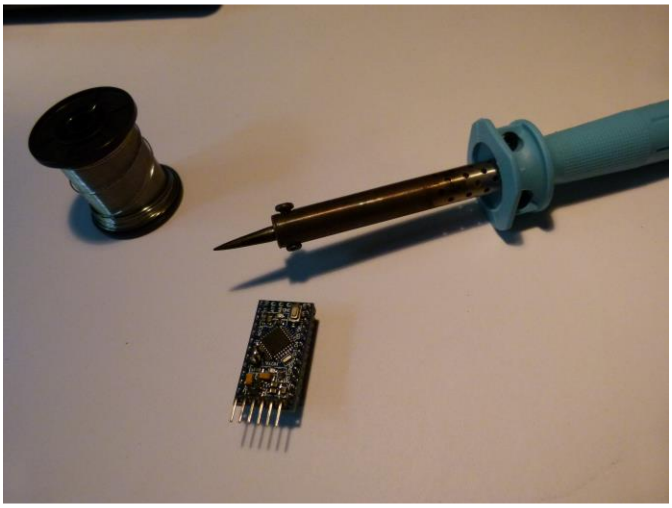


Figure 50: Funduino along with the tools used to solder the necessary pins onto the board

At the moment of uploading the code to the board a problem was encountered as the board did not synchronize with the computer at the time of uploading the code. After some research [15] the cause of the problem was found to be the reset pin of the board. The program which uploads the code sends a reset signal to the board when uploading the code, but the Arduino Pro Mini lacks the pin to receive the reset signal, therefore the button needs to be manually pressed in order to successfully upload the code.

A typical Arduino code consists on a file with three main blocks:

- At the top of the file all the external libraries which will be used are included with the following syntax: *#include <library.h>* where library.h is the library which will be included.
- setup() is a function which runs at the start of the code once and is mainly used to set up all the different variables, the pins and any other preparations.
- loop() is the main function which will be looping continuously, the main code for the program will be written inside this function.

The schematic for the Arduino connection with the PC via the USB to TTL serial converter is shown in figure 51.

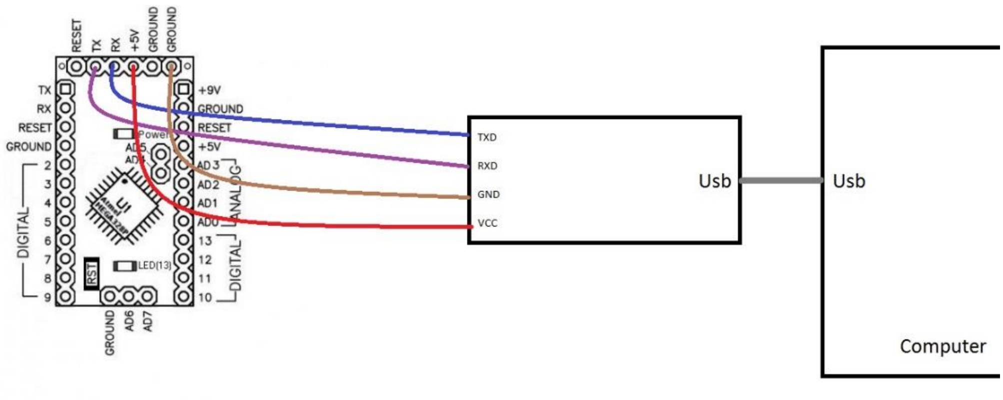


Figure 51: Funduino connection schematic to the USB to TTL serial converter

5.8.2.3 Servos

Definition

Servo motors are electronic devices consisting on gears and a shaft which can be controlled via the microcontroller. Servo motors can adjust the position of the shaft between 0 and 180 degrees, which is done by the microcontroller sending a PWM (Pulse-Width Modulation) signal, with the width of the pulse being the factor which makes the motor turn. Minimum pulse width or duration will be around 1ms (which will make the servo turn the shaft to 0 degrees) and 2ms (which will make the servo turn the shaft to 180 degrees) on most of the small servos to be used with Arduino boards.

Servos will be needed to control the rotation of the pill storage and pill chamber parts so that pill refilling and pill dispensing mechanism can work correctly.

Implementation

Two servos SG90 were adquired for this project. These servos are quite small in size (22.2 x 11.8 x 31 mm) and weight (9 grams) which suits them perfectly for the pillbox as keeping both size and weight to a minimal value was a priority.

SG90 servos come with three cables, two are for power supply (Vcc and ground), which will be connected directly to the Arduino 5V and Gnd pins respectively and a third cable which will be connected to a PWM ouput pin of the Arduino. There are some digital pins which are capable to send PWM signals in the Arduino, in the case of the Arduino Pro Mini these pins are D3, D5, D6, D9, D10 and D11, a servo can also be attached to any of the analogue pins as it can also work with them. The connection are shown in figure 52.

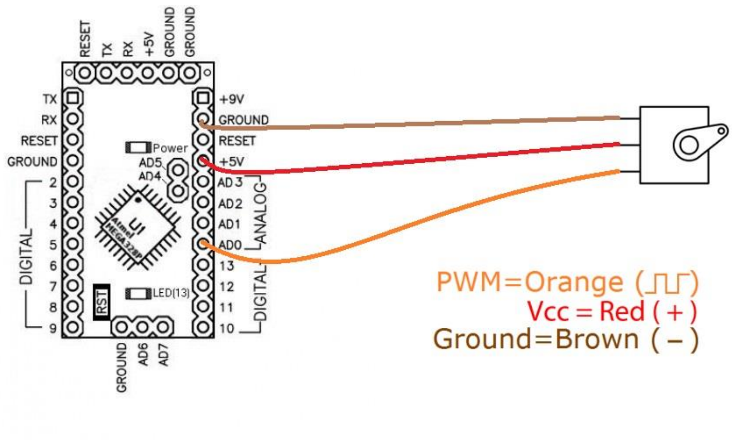


Figure 52: Servo connection schematic to the Funduino board

For the use of servos in Arduino the following steps are to be followed:

1) Include library Servo.h at the top of the code*: #include<Servo.h>*
2) Create an instance of Servo: *Servo myservo;*
3) In the setup() function it is necessary to attach the servo to a pin: *myservo.attach(PINNUMBER);* where PINNUMBER is the number of the pin in the Arduino board and myservo is the servo object we created in step 2.
4) In loop() function the servo can be controlled using *myservo.write(pos)* where pos is a number between 0 and 180 which will be the degree number that the servo must rotate its shaft.

5.8.2.4 Bluetooth module

Definition

Bluetooth is a "telecommunications industry specification that describes how mobile phones, computers, and personal digital assistants (PDAs) can be easily interconnected using a short-range wireless connection."[23] Therefore Bluetooth is a wireless technology which enables two devices using this technology to communicate in a fast and secure way. "Bluetooth technology operates in the unlicensed industrial, scientific and medical (ISM) band at 2.4 to 2.485 GHz, using a spread spectrum, frequency hopping, full-duplex signal at a nominal rate of 1600 hops/sec. [24]

There are 3 different types of Bluetooth, Class 1, Class 2 and Class 3. Class 1 has a connection range of about 1m, Class 2 can handle connections up to 10m and Class 3, mainly used for industrial purposes, can handle connections up to 100m. Class 2 Bluetooth is the most commonly used Bluetooth and the one which will be implemented in this project. This type of Bluetooth has a very low power consumption of about 2.5mW.

Implementation

There are many Arduino-compatible Bluetooth device in the market due to its popularity in a wide spectrum of prices. The first option that was considered was to acquire the official Arduino Bluetooth shield.

This module had three main drawbacks, it was overprized (around 25 euros) and its size, as it was bigger than the Arduino board itself and lastly it took up many of the pins of the Arduino board only for Bluetooth. For those reasons the Arduino Bluetooth Shield was discarded and a small unofficial Bluetooth module was selected. This module works by using the serial communications of the Arduino board, costs around 10 euros and can be seen in figure 53.

Bluetooth module has 4 pins, 2 for powering the device (which will be connected to Vcc and Gnd pins of the Arduino board), one for receiving data from the Arduino board and sending it via Bluetooth and another one for transmitting the data received from Bluetooth to the Arduino board. Connections are shown in figure 54.

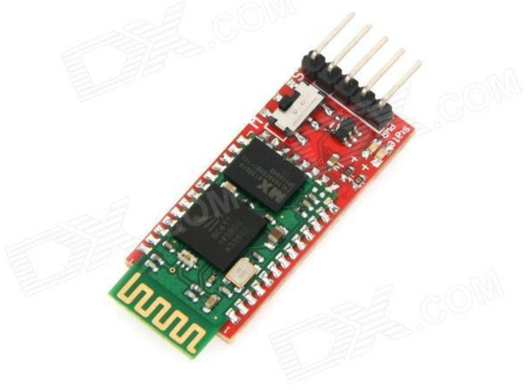


Figure 53: Bluetooth module acquired for this project

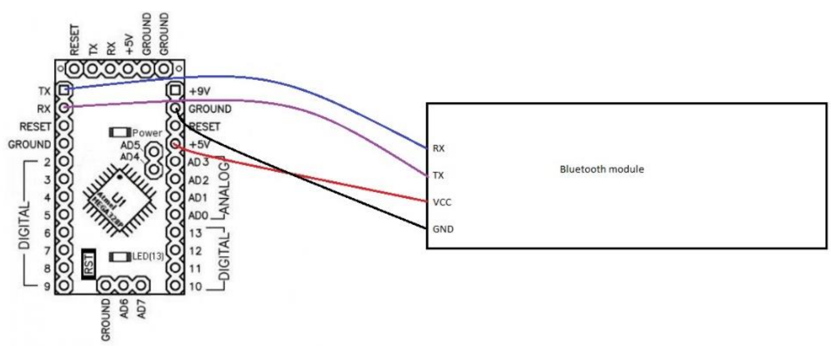


Figure 54: Bluetooth module connection schematic to the Funduino board

Code for managing the Bluetooth communications are done via the serial port of the Arduino:

1) First, the serial communications must be started at a already defined baud rate which can be handled by the Bluetooth module,

this is done with the following code: *Serial.begin(BAUD_RATE);* where BAUD_RATE will likely be 9600 bps.

2) To read data received it is necessary to check if there is data available to read with the code: *Serial.available()* if there is any data to read it can be recovered with received_data = *Serial.read();*
3) Sending data via Bluetooth can be done with code *Serial.println(STRING_TO_SEND);*

In order for communication to happen the device communicating with the pillbox must have paired with it previously. Pairing is a process in which two Bluetooth devices establish communication with a preassigned password, typically 0000 which is the case of the Bluetooth module used in the pillbox.

5.8.2.5 RTC (Real-Time Clock) Module

Definition

Real-Time Clock [25] are usually referred to tiny circuits which form part of a bigger circuit but have their own power source (commonly a coin cell,a CMOS battery). This circuit is in charge of tracking the time even when the bigger circuit to which they are attached is powered off. This type of circuits are found in computers, laptops, etc… so that the operative system knows what time is it when the user powers on the computer. Figure 55 shows the RTC module acquired for this project which is described in the implementation section below.

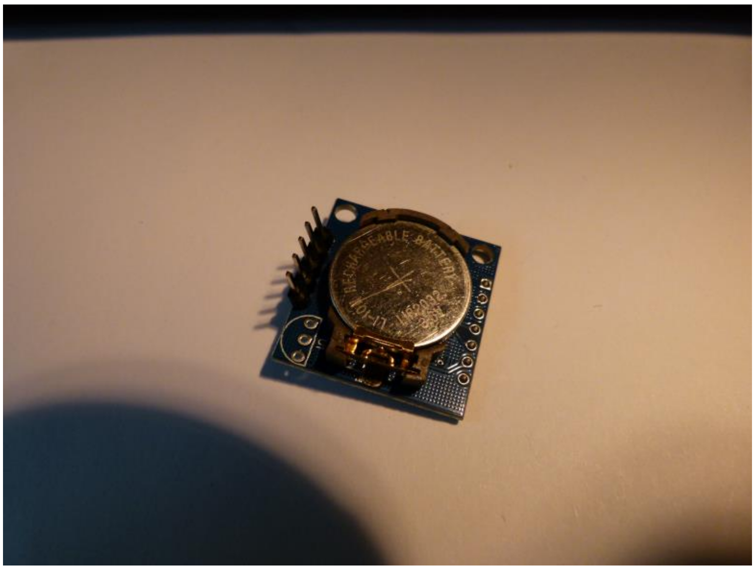


Figure 55: RTC module acquired for this project

Implementation

The pillbox may not always be switched on and while the Arduino has an internal clock, it only takes track of the time from the exact moment de Arduino is powered on, thus lacking the track of what hour and day it is in the week, which becomes a mandatory requisite for tracking when the pills have to be dispensed to patients.

An RTC module was acquired, a Tiny RTC Module which communicates with the Arduino board via I2C communications. I2C communications (Inter-Integrated Circuit) communication protocol [26] which allow the RTC module to communicate with the Arduino. I2C will use to cables attached in each of the components to the SCL pin (through which a clock signal will be transferred to synchronize the data) and the SDA pin (Serial Data Line through which the data will actually be transferred). These connections are illustrated in figure 56.

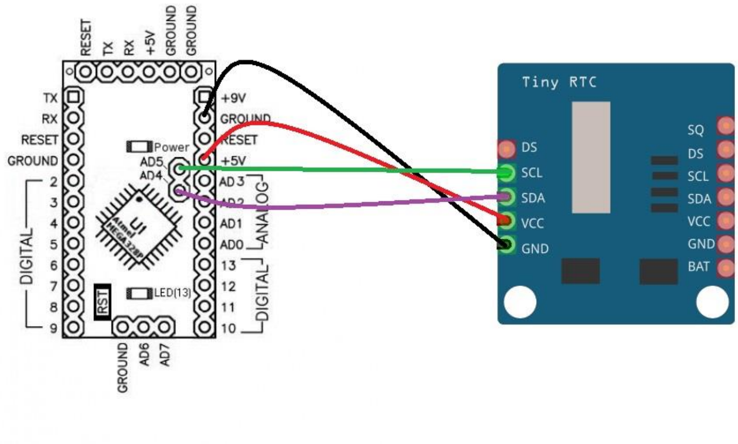


Figure 56: RTC connection schematic with the Funduino board

To control this software:

1) It is mandatory to include default libraries <Wire.h> and the specific library for the module <RTClib.h> which has been downloaded from [GitHub] [39] and define which module will be used with the line *RTC_DS1307 RTC;*
2) It is necessary to start both the Wire (the I2C communication) and the RTC module with *Wire.begin()* and *RTC.begin()*
3) The time can now be instantiated with *DateTime now = RTC.now();* and be accessed with several functions: *now.hour(); now.day(); now.month(); now.year(); now.seond(); now.dayOfWeek();*

The module can be programmed with the correct time using *RTC.adjust(DateTime(__DATE__, __TIME__));*

which sets the time of the module at the time the program was compiled. This line will not be inserted into the final code as it would cause the module to reset its internal clock each time the pillbox was switched on. The module will be pre-programmed with the correct time before being inserted in the pillbox's circuitry.

5.8.2.6 EEPROM MEMORY

Definition

EEPROM (Electrically Erasable Programmable Read-Only Memory) is a non-volatile memory which comes with each one of the Arduino boards.

Non-volatile memory stays when the device is powered off and this will be needed to store the pill schedule information, this is, each one of the times in a week the pillbox will dispense a pill and which pill type is the one dispensed.

It is mandatory to store the pill schedule in non-volatile memory as the device may be powered off in some occasions, thus making it essential for the pillbox to retain the pill schedule in a non-volatile memory so that it may resume functioning properly once turned on again.

Implementation

EEPROM memory in the Arduino model this project uses (Arduino Pro Mini) has 1kB of memory. A single entry in the pill schedule will take up a total of four bytes in memory as will be shown in chart 3.

Byte number	1	2	3	4
Information stored	Day of week	Hour	Minute	Pill type
Possible values	Monday/Tuesday/Wednesday/Thursday/Friday/Saturday/Sunday	0-23	0-59	A/B/C

Chart 3: Byte map for a single pill dose stored in memory

The day of the week in which the pill will have to be taken will be stored in the first byte, the hour of the day in the second, the minute in the third byte and the last byte will be reserved to specify which one of the pill compartments the pill will have to be taken out of.

A table will be done to map the different values of each of the fields into a number which will be written as a byte in the eeprom memory. As the information will be stored in separate bytes it was considered to write the byte number which directly corresponded to the value being written (e.g. if the pill dose had to be on Friday at 5:05 then the value 5 would be written in byte 1 corresponding to the fifth day of the week, value 5 would also be written in bytes 2 and 3 corresponding to the fifth hour and the fifth minute within the hour) but was decided to map each one of the values into different values to prevent errors from happening when reading from the memory.

Day of the week:

Chart 4 shows the number mapping each day of the week.

Day of the week	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Number	2	3	4	5	6	7	8

Chart 4: Day of the week value mapping to a byte value

Hour of the day:

The hour of day will be a number from 0 to 23 and will be mapped subsequently from numbers 9 to 33. The possible values for the hour will be 9≤X≤33, therefore X-9 = hour at which the pill will need to be dispensed.

Minute:

Minutes will be mapped in the range of numbers from 34 to 94, consequently

34≤Y≤94, where Y-34 will be the formula to find the minute at which the pill has to be dispensed.

Pill type:

There are three possible pill types, A, B and C which can be dispensed by the pillbox, the mappings are shown in chart 5.

Pill type	A	B	C
Number	95	96	97

Chart 5: Pill type mapping to a numerical byte value

The number 1 will be written as the last byte when there are no more pill doses to be read from the memory as an EOF character. Each time a value is read from the EEPROM memory a check to see if the value of the number is within the expected range of numbers for the byte that has been read.

As the EEPROM memory can hold 1024 bytes and each entry in the pill schedule takes 4 bytes then the memory could hold up to 256 entries in the weekly pill schedule, a number which will be enough for the common user of the pillbox.

To operate with the EEPROM memory the library EEPROM.h has to be included in the code with the line *#include <EEPROM.h>* then the memory can be accessed with different functions such as *EEPROM.write(a,b)* which writes byte b in address a of the memory or *EEPROM.read(a)* which reads the value at the address a.

5.8.2.7 Switch, button, LEDs and speaker

Definition

Switch

A switch will be used for accessing the managing options of the pillbox. If the switched is in the ON position a LED will be lit up and indicating the pillbox is in Bluetooth mode, with the nurse or caregiver in charge of managing the pillbox prepared to access the pillbox's management options.

Button

A push button will be used in the board to activate the dispensing procedure. When a pill is pending to be dispensed the button will be pressed by the user and the pillbox will be dispensed.

LEDs

Leds will be used to notify if the pillbox is in management state and to notify the user that a pill is pending to be dispensed.

Piezospeaker

A piezospeaker will be used to notify with an acoustic notification that a pill is pending to be dispensed till five minutes have passed or the user presses the dispensing button

Implementation

The piezospeaker will be connected directly to an output pin and can be controlled through the code with the tone function: *tone(pin_number, frequency_of_tone, time_to_play);*

The LEDs and the switches and buttons will be connected with 220 ohms resistors to output or input pins depending on the function they will be executing (refer to Pin Mapping section in this document).

The breadboard view in figure 57 and the schematic in figure 58 show how the connections with these components will be made.

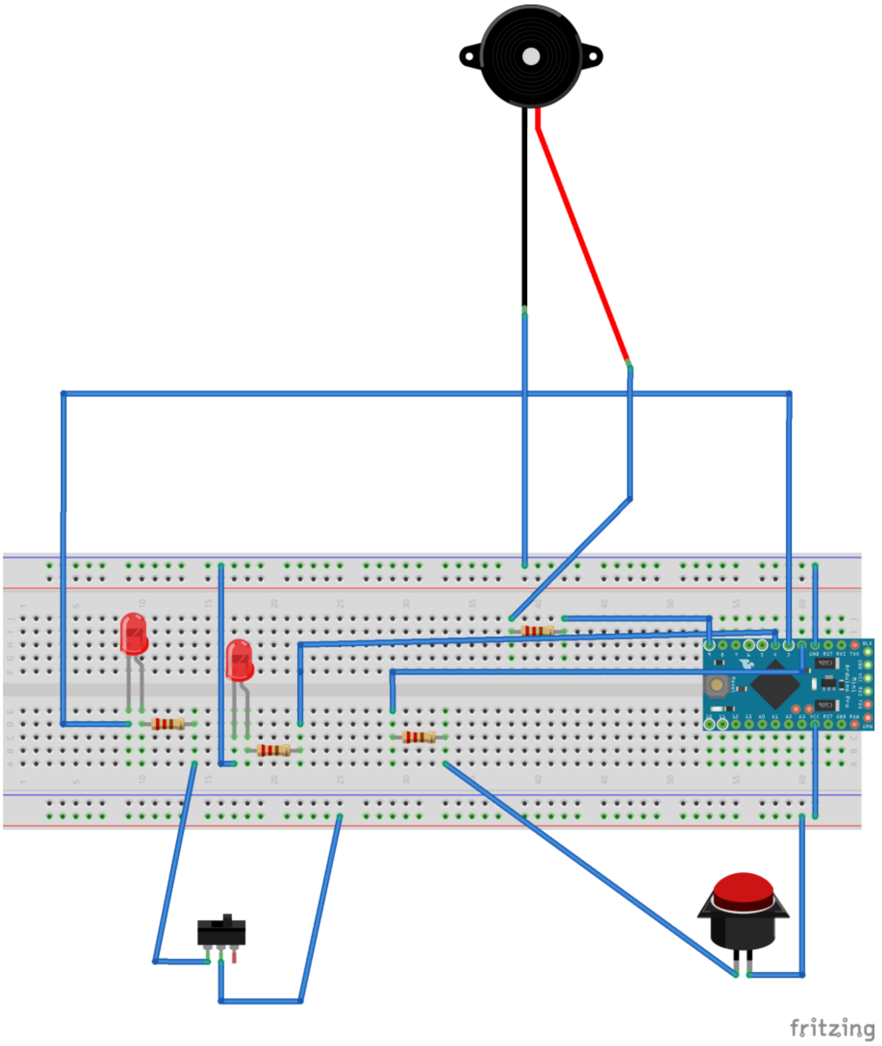


Figure 57: Connection breadboard schematic of the speaker, LEDs, switch and button components

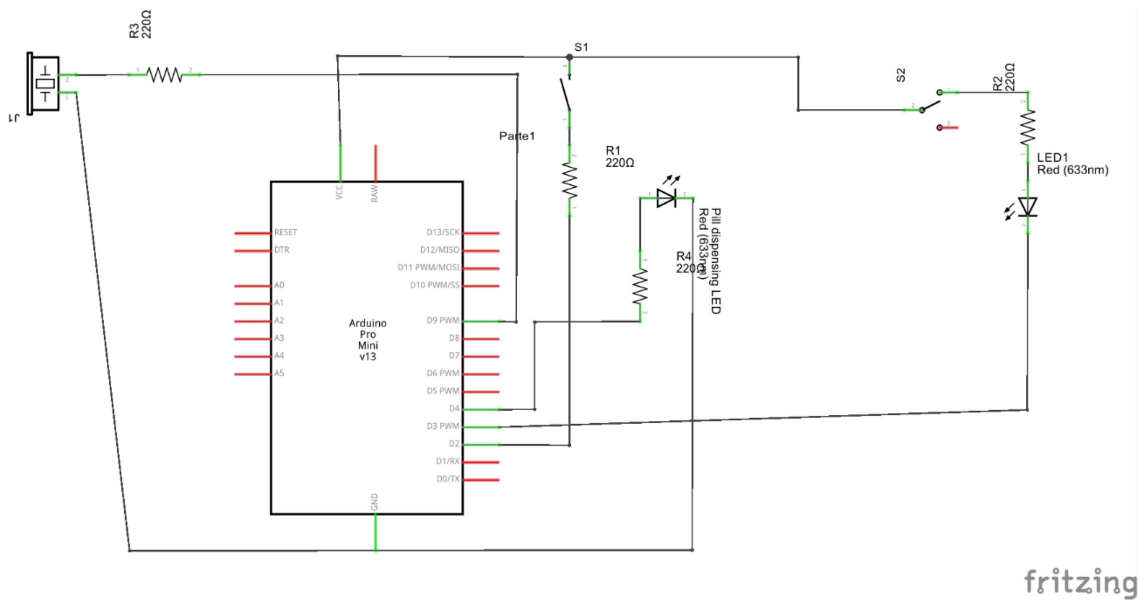


Figure 58: Connection schematic of the speaker, LEDs, switch and button components

5.8.2.8 Battery

Definition

The battery used will be a LiPo battery which will be connected to the RAW and the GND pins in the Arduino board powering it. The selected board to be used will be a LiPo battery which supplies 7.4V and has a capacity of 2200mAh. This battery will be able to power the pillbox for at least a day and a half approximately (depending on the use it is given to the pillbox). Figure 59 shows the mentioned battery.

Figure 59: Battery used for this project

Implementation

As the pillbox will be acquired by an organization (hospital or retirement home) the company will supply with the number of pillboxes a greater number of batteries for the pillbox as well as a number of LiPo chargers. These chargers (figure 61) will be able to charge the pillbox's spare batteries the retirement home/hospital has while the patients are using the pillboxes using other batteries. Once the batteries of one of the pillboxes start to run out it will be changed by one of the caregivers/nurses and the empty battery will be put to charge. As the pillbox hasn't been implemented with a battery measuring device the pillboxes will also be sent with lipo voltage check circuits (figure 60) so that the nurses/caregivers will be able to monitor the voltage level of the battery when it is running low.

Figure 60: Voltage checker component

Figure 61: Battery charger

5.8.3 Electronic schematics

5.8.3.1 Designing the schematics: Fritzing

Fritzing is defined in their web page as "an open-source hardware initiative that makes electronics accessible as a creative material for anyone".[27] This software allows its users to create easy schematics in various different designs, with a breadboard view were the circuits are implemented with a prototyping breadboard aide, the proper schematical view and an alternative PCB view were the circuit implemented in the schematic view is automatically calculated and produced as it would be if it was implemented in a PCB.

5.8.3.2 Pin mapping

Chart 6 shows the different digital pins to which the different components will be connected and if those pins will be in output (sending signals to the devices) or input (receiving signals from the devices).

Connected part	Arduino pin number	Input/output pin
Servos	D5 and D6 (PWM)	Output
Button	D2	Input
BT switch and LED	D3	Input
Dispensing LED	D4	Output
Speaker	D9	Output

Chart 6: Pin mapping chart

5.8.3.3 Breadboard and PCB schematic

The first step to créate the final schematic was to design a breadboard view in which the connections could be made with the help of a breadboard. Figure 62 shows this breadboard connection scheme. Afterwards the program used (Fritzing) was able to design a PCB version of the circuit which can be seen in figure 63.

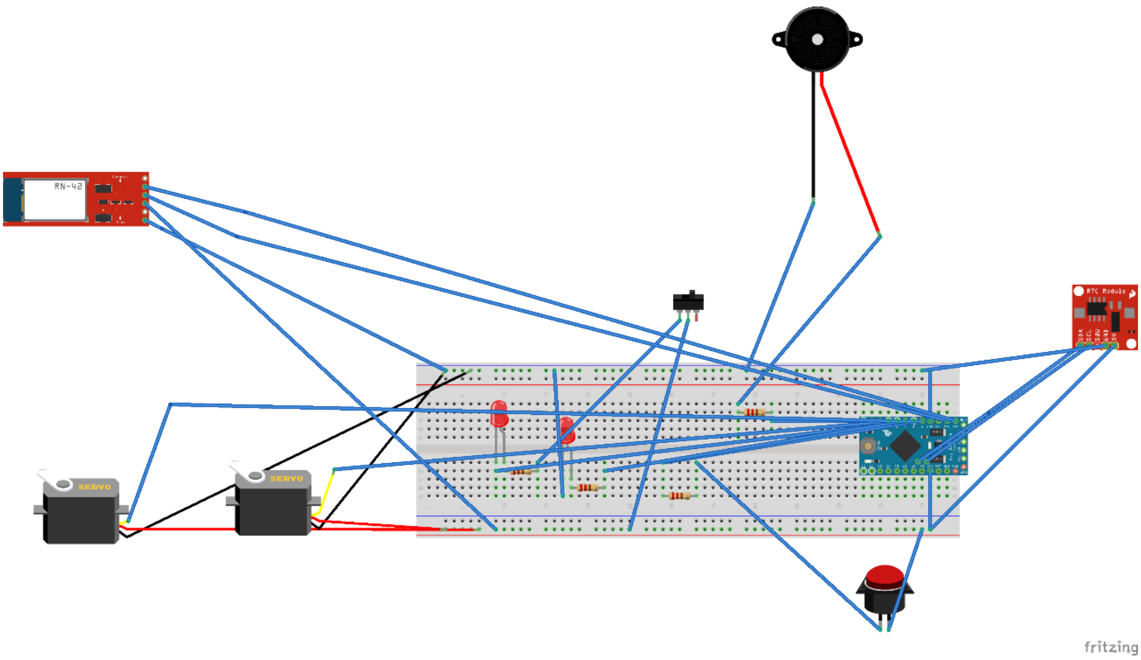


Figure 62: Breadboard connection schematic of the pillbox circuitry

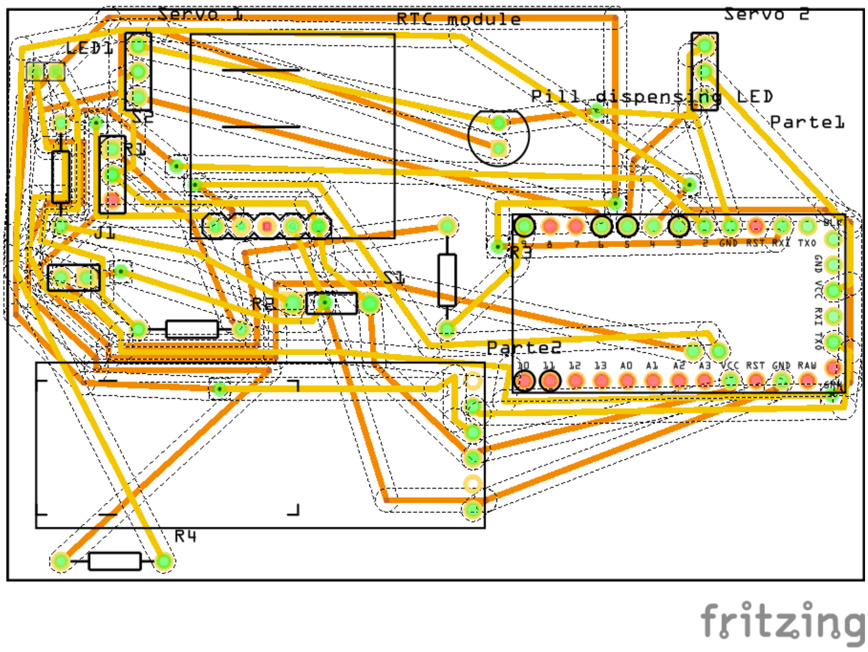


Figure 63: PCB schematic of the pillbox circuitry

5.8.3.4 Final schematic

While the PCB version shown in figure 63 was produced automatically by the program the final schematic of the connections had to be drawn manually. Figure 64 shows this final schematic version with all the connections are clearly shown and marked.

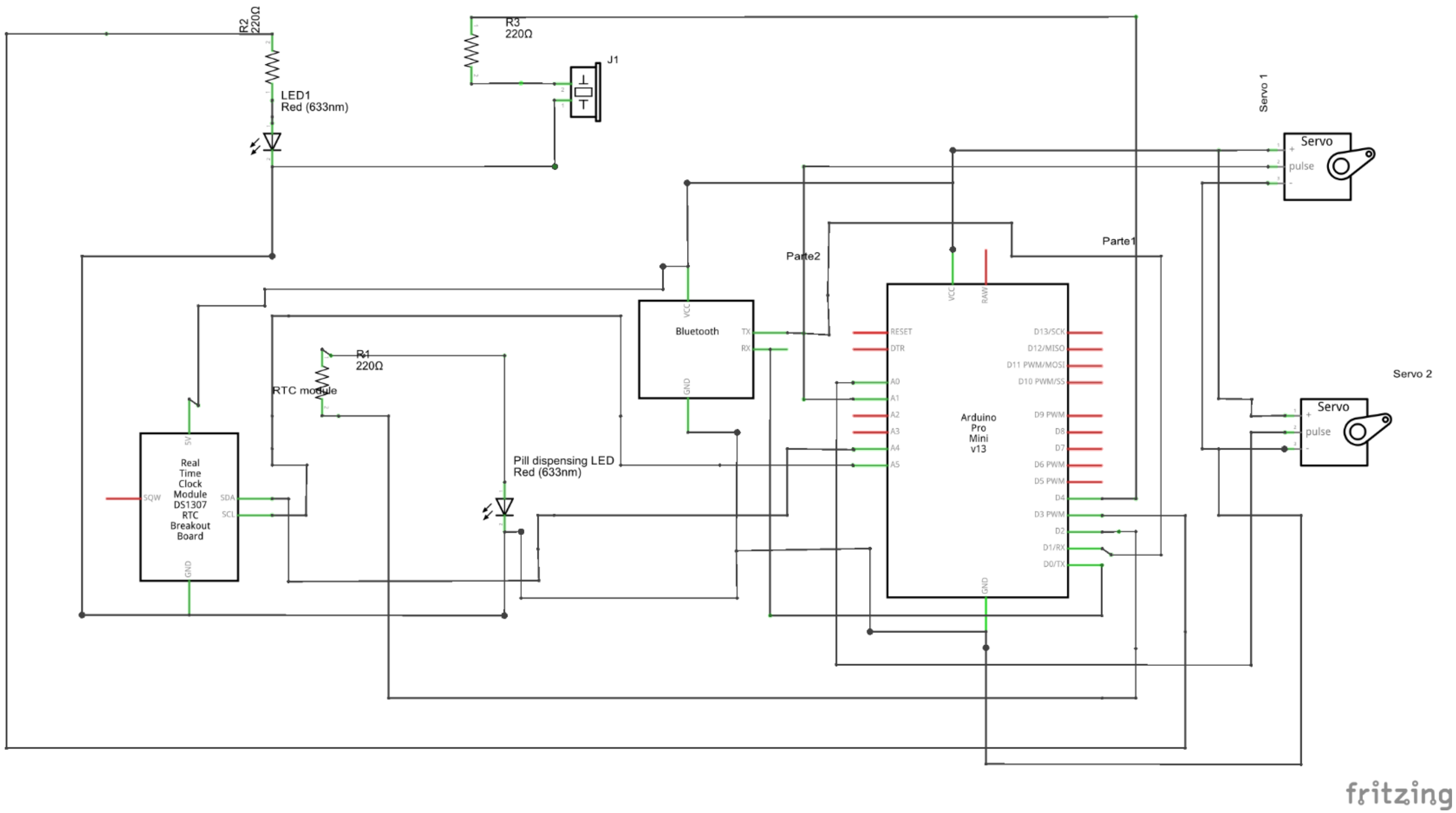


Figure 64: Pillbox circuitry schematic

5.8.4 Code flowchart

A flowchart was the first step needed to create the final version of the code. This flowchart would represent the logic of the pillbox when operating. Although the flowchart is the starting point of designing the code the functionality of the final code may vary. Figure 65 shows the flowchart.

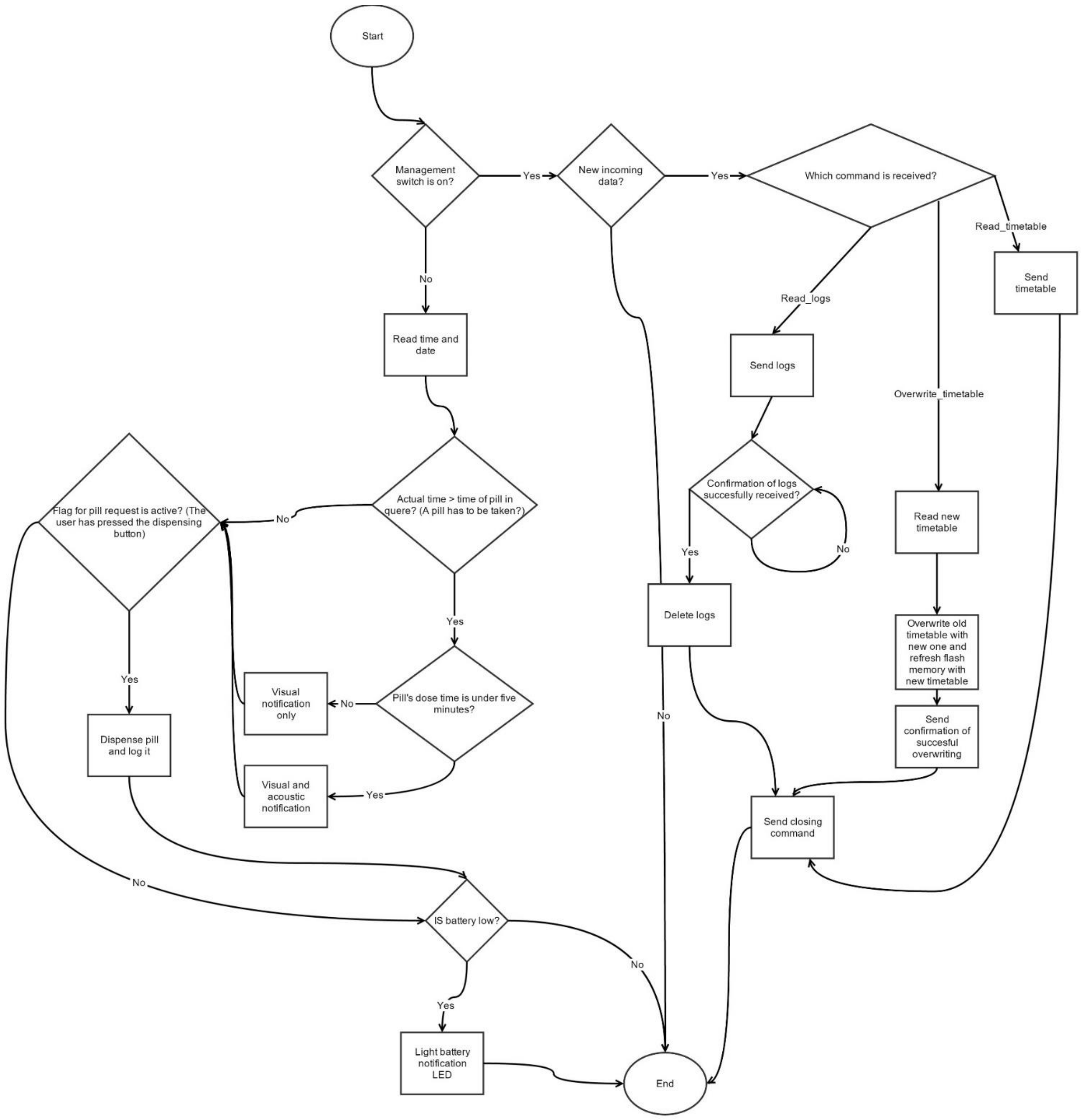


Figure 65: Code flowchart

5.8.5 Pseudocode

In this section the code's pseudocode is shown in Chart 7. Pseudocode consists on a readable versión of what the code will be and it has been based on the flowchart shown in figure 65. The final code will be programmed basing on the pseudocode shown here.

```
int notificationLedPin;//output
int BTswitch; //input
int batteryLedPin; //output
int servo1Pin; //output
int servo2Pin; //output

void setup() {
  //define pins as outputs/inputs
  //initialize
  //position servos in the correct position
  //load pillbox timetable and actual date and time
}
void loop() {
  if(BTswitch == ON){
    if(newIncomingData?){

switch(command){
        case read_timetable:
        //send timetable;
        //send closing command
        break;
        case overwrite_timetable:
        //read timetable;
        //overwrite old timetable with new one;
        //refresh pill dosage for the day up to the actual time;
        //send confirmation;
        //send closing command
        break;
        case read_logs:
        //send logs;
        //wait for confirmation of received logs;
        //delete logs;
        //send closing command
        break;
        default:
        //send closing command
        break;
    }
    }else{
      break;
    }
  }else{
    //read time and date
  if(actualtime > timeoffirstpillinqueue){
    if(actualtime - timeoffirstpillinqueue < 5min){
      //visual and acoustic notifications
    }else{
      //visual notifications
    }
  }
  if(pillflag ==1){
  //dispense pill
  //log dose time
  //remove pill from queue
  }

  }

}
```

Chart 7: Pseudocode

5.8.6 Code

The final code shown below in charts 8 – 12 will be the code compiled and programmed into the Arduino board. Code has an .ino extension and may be compiled and loaded into the board within the Arduino IDE.

```
/***********************/
/*Constant definitions*/
/*********************/
#define dispButton 2
#define switchBT 3
#define dispLed 4
#define speaker 9

/*******************/
/*Libraries needed*/
/*****************/
#include <EEPROM.h>
#include <Servo.h>
#include <Wire.h>
#include "RTClib.h"


Servo upperServo; //Upper servo declaration
Servo lowerServo; //Lower servo declaration
RTC_DS1307 RTC; //Real-Time Clock module declaration


long debouncing_time = 15;           //Debouncing Time in Milliseconds
volatile unsigned long last_micros; //Variable needed for the button debounce
int dispensingFlag = 0;              //Flag to signal if a pill is requested by the user
DateTime now;
int day = 0;
int hour = 0;
int minute = 0;
byte actualDose[4] = {255, 255, 255, 255}; //Array where the next dose to be dispensed will
be stored
int command = 0;
byte doses[256][4];                          //Array where the dose timetable will be loaded
during runtime
byte logs[256][4];                          //Array where the logs will be kept
unsigned long timesinceon;
/**********************************/
/*Different upper servo positions*/
/********************************/
int pillAdisppos = 0; //Pill A dispensing position
int pillBdisppos = 0;//Pill B dispensing position
int pillCdisppos = 0;//Pill C dispensing position
int defaultpillstoragepos = 0;//Default pill storage position
int pillAloadpos = 0;//Pill A loading position
int pillBloadpos = 0;//Pill B loading position
int pillCloadpos = 0;//Pill C loading position
/**********************************/
/*Different lower servo positions*/
/********************************/
int loadpos = 0;//Pill chamber loading position
int releasepos = 0;//Pill chamber releasing position

int finishedrefilling = 120;
```

Chart 8: Final code – part 1

```
void setup() {
  pinMode(dispButton, INPUT);
  pinMode(switchBT, INPUT);
  pinMode(dispLed, OUTPUT);
  pinMode(speaker, OUTPUT);
  upperServo.attach(5);//Attach upper servo to pin 5
  lowerServo.attach(6);//Attach lower servo to pin 6
  digitalWrite(dispLed, LOW);
  attachInterrupt(2, debounceInterrupt, RISING);//Attach an interrupt to pin 2
  Serial.begin(9600);

  Wire.begin();
  RTC.begin();
  loadTimetable(doses);//Load the doses timetable from the EEPROM memory into the doses array
  now = RTC.now();//The actual date is gathered from the RTC module
  day = now.dayOfWeek();
  while(doses[i][0]!=day){
        doses[i][0]=255;
        doses[i][1]=255;
        doses[i][2]=255;
        doses[i][3]=255;

  }
}

void loop() {
  if (digitalRead(switchBT) == HIGH) {//Check if the Bluetooth switch is active
    if (Serial.available() > 0) {//Check if there is Serial data available
      command = Serial.read(); //Read the command sent by the device to the pillbox
      switch (command) {
        case 1:  //Case 1 corresponds to sending the timetable to the device
          Serial.write("Sending the timetable now.");
          byte complete_timetable[256][4];//An array where the complete timetable will be loaded
          loadTimetable(complete_timetable);//Load the timetable from memory into the array
          for (int i = 0; i < 256; i++) {
            for (int j = 0; j < 4; j++) {
              Serial.write(complete_timetable[i][j]);//Writes the array in order to the device
            }
          }
          sendClosingSequence();
          break;

        case 2: //Case 2 corresponds to the device sending a new timetable for the pillbox
          Serial.write("Updating timetable procedure started.");
          byte new_timetable[256][4];
          for (int i = 0; i < 256; i++) {
            for (int j = 0; j < 4; j++) {
              new_timetable[i][j] = Serial.read(); //The new timetable is read byte by byte
            }
          }
          overwrite_timetable(new_timetable);//Overwrite  the  EEPROM  memory  with  the  new
timetable
          loadTimetable(doses);//Load the new timetable from the recently overwritten EEPROM
memory into the doses array
          sendConfirmationSequence();//Send a sequence so that the device knows the transference
has been correctly done
          sendClosingSequence();
          break;

        case 3: //Case 3 sends how much time the device has been powered on for battery saving
causes
          timesinceon = millis();
          Serial.write(timesinceon);
          sendClosingSequence();

          break;

case 4: //Case 4 sends the logs of the times at which the pills were actually taken
          for (int i = 0; i < 256; i++) {
            for (int j = 0; j < 4; j++) {
              Serial.write(logs[i][j]);
            }
          }
          sendClosingSequence();
```

Chart 9: Final code - part 2

```
case 5: //Case 5 activates the refilling sequence
          Serial.write("Refilling sequence activated.");
          int movement;
          movement = Serial.read();//The device sends a number, this number will either be on
of the pill refilling position or a number interpreted as a command to stop the refilling
sequence
          while (movement != finishedrefilling) {//While the number sent is not the one which
stops the refilling sequence the pill storage will rotate to the sent position
            upperServo.write(movement);
            delay(1000);
            movement = Serial.read();
          }
          upperServo.write(defaultpillstoragepos);//After finishing the refilling sequence the
pill storage will be rotated to its default positions
          delay(1000);
          sendClosingSequence();
          break;

        default:
          sendClosingSequence();
          break;
      }
    } else {

    }
  } else {
    now = RTC.now();//The actual date is gathered from the RTC module
    day = now.dayOfWeek();
    hour = now.hour();
    minute = now.minute();
    loadNextDose();//The next dose in line will be loaded into the actualDose variable.

    if (day == actualDose[0] && hour >= actualDose[1] && minute >= actualDose[2]){ //Checks the
time of the next dose with the actual time
      if (minute - actualDose[3] < 5 ) {//If the next pill was to be taken less than five minutes
ago visual and acoustic signals are sent
        digitalWrite(dispLed, HIGH);
        alarm();
      } else {
        digitalWrite(dispLed, HIGH);//Only visual notifications are activated if the pill had
to be taken more than five minutes ago
      }
    }

    if (dispensingFlag == 1) {//If the flag to dispense a pill was pressed
      int pill = actualDose[3];
      dispensePill(actualDose[3]);//The dispensing procedure is activated for the pill type
which has to be taken
      for (int i = 0; i < 256; i++) {//A for loop is created to log the pill dose time
        if (logs[i][0] == 255) {
          logs[i][0] = day;
          logs[i][1] = hour;
          logs[i][2] = minute;
          logs[i][3] = pill;
        }
      }


    }

  }
}
```

Chart 10: Final code - part 3

```
/***********************************************************/
/*loadTimetable:                                          */
/*This function recieves an array of bytes as an argument*/
/*and loads the dose timetable from the EEPROM memory    */
/*into the array                                        */
/******************************************************/
void loadTimetable(byte array_to_fill[][4]) {
  int address = 0;
  for (int i = 0; i < 256; i++) {
    for (int j = 0; j < 4; j++) {
      array_to_fill[i][j] = EEPROM.read(address);
      address++;
    }
  }
}
/****************************************************************/
/*debounceInterrupt:                                          */
/*The method which prevents the dispensing button press from */
/*debouncing and then activates the flag to dispense a pill.*/
/************************************************************/
void debounceInterrupt() {
  if ((long)(micros() - last_micros) >= debouncing_time * 1000) {
    dispensingFlag = 1;
    last_micros = micros();
  }
}
/*****************************************************************/
/*loadNextDose:                                                */
/*This function retrieves the next dose to be taken from      */
/*the doses array and stores it into the actualDose variable.*/
/************************************************************/
void loadNextDose() {
  int dosetocompare;
  for (int i = 0; i < 256; i++) {
    dosetocompare = (int)doses[i];
    if (dosetocompare != 255) {
      for (int j = 0; j < 4; j++) {
        actualDose[j] = doses[i][j];
        doses[i][j] = 255;
      }
      break; //para salirnos del for loop
    }
  }
}
/******************************************************************************/
/*dispensePill:                                                              */
/*This function starts the pill dispensing procedure to                     */
/*dispense the pill type which is notified with the function's argument   */
/*and rotates the servos in turn to dispense the pill                     */
/************************************************************************/
void dispensePill(int typeOfPill) {
  if (typeOfPill != 255) { //Esto comprueba que no haya fallos o que la pildora cargada sea correcta
    upperServo.write(typeOfPill);
    delay(200);
    lowerServo.write(loadpos); //posicion de carga
    delay(200);
    upperServo.write(defaultpillstoragepos);
    delay(200);
    lowerServo.write(releasepos);
    delay(200);
    digitalWrite(dispLed, LOW);


  }
}
```

Chart 11: Final code - part 4

```
/**********************************************************************/
/*sendConfirmationSequence:                                           */
/*This function sends to the other device a                          */
/*preestablished confirmation sequence to indicate the new timetable*/
/*has been correctly received                                       */
/*******************************************************************/
void sendConfirmationSequence() {
  byte confirmationsequence[5] = {200, 200, 200};
  int i = 0;
  for (int i = 0; i < 3; i++)
    Serial.write(confirmationsequence[i]);
}

/************************************************************/
/*sendClosingSequence:                                      */
/*This function sends to the other device a                */
/*preestablished closing sequence to indicate the correct*/
/*termination of the bluetooth command                    */
/*********************************************************/
void sendClosingSequence() {
  byte closingsequence[5] = {255, 254, 255};
  int i = 0;
  for (int i = 0; i < 3; i++)
    Serial.write(closingsequence[i]);
}
/***********************************************************/
/*overwrite_timetable:                                     */
/*This function recieves an array of bytes as an argument*/
/*and overwrites the timetable stored in the EEPROM     */
/*memory with the one in the array                     */
/*******************************************************/
void overwrite_timetable(byte new_timetable[][4]) {
  int address = 0;
  for (int i = 0; i < 256; i++) {
    for (int j = 0; j < 4; j++) {
      EEPROM.put(new_timetable[i][j], address);
      address++;
    }
  }
}
/*****************************************************/
/*alarm:                                              */
/*This function emits an alarm through the speaker  */
/*to signal the acoustic notification of a new dose*/
/**************************************************/
void alarm() {
tone(speaker, 440, 2000);
}
```

Chart 12: Final code - part 5

6 SYNTHESIZING THE PILLBOX

6.1 PRINTER DESCRIPTION AND ASSEMBLY

The selected printer which was decided to be acquired for this project was the Prusa i3 Steel. This printer is the Prusa's third revision with a steel frame to confer more stability and robustness which mainly helps when the printer is printing an object, as it may have vibrations which could slightly displace the printer's extruder. The steel frame avoids these vibrations as it is a stronger frame than the original one which is usually made out of aluminium.

Below there is a photographic comparison between both models shown in figure 66.

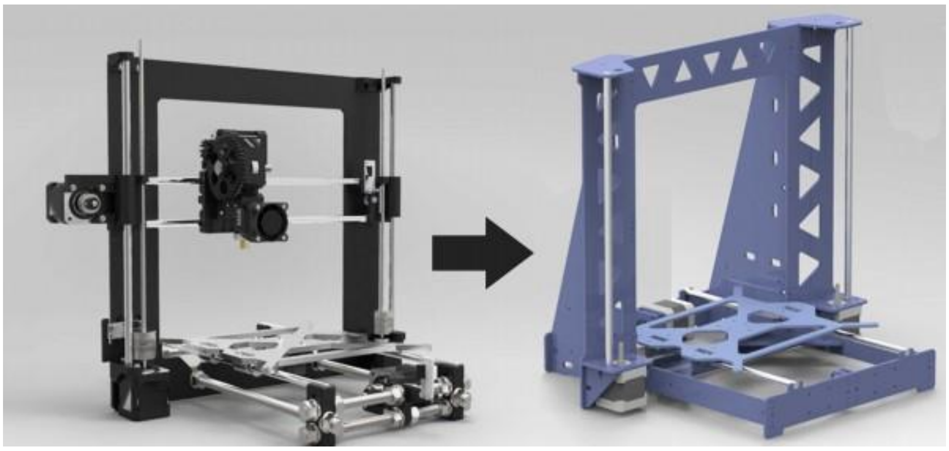


Figure 66: Prusa 3rd iteration with alumnium (left) and steel (right) frame

The printer was acquired via a shop located in Madrid [36] which sends a box with all the necessary pieces for arming the printer inside a box. It is up to the user to assembly the printer functionally. The selected printer came at a cost of 409€. Below an image of how the box comes with all the prepackaged pieces is shown in figure 67 taken from an unboxing video of KitPrinter3D [36]. The main advantage of this kit was that the Arduino comes with the firmware already preloaded and the stepper motors calibrated and ready for use, which saves a lot of time in getting the printer ready to print.

Figure 67: KitPrinter3D 3D printer kit inside the box

This printer consists on a metal frame which holds a platform (which holds the printing bed) capable of moving in the Y axis and a plastic part capable of moving in the X and Z axis. This part will be holding the extruder alongside with a mechanism which feeds the plastic filament from the roll into the extruder at different speeds. The axis of coordinates is located at the left hand side corner which is nearer the front of the printer. The printer uses endstops, a mechanical switch, to know when the platform and the extruders head has reached its maximum or minimum values. Below a representation of the coordinates in relation to the printer is shown in figure 68.

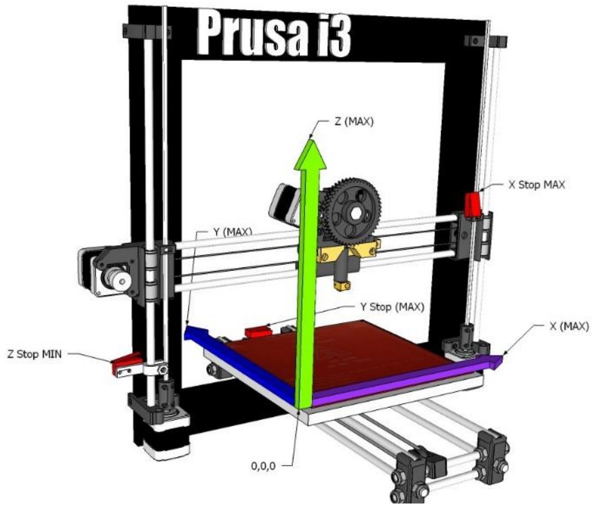


Figure 68: Prusa i3 axes diagram

The printer uses a total of five stepper motors to achieve the movement of the extruder's head and platform along the axes as well as the filament feed into the extruder. Power is also needed for heating up the extruder till the desired temperature at which the plastic filament will melt and to heat up the printing bed (it is a heated bed as this helps reinf All the system is controlled with an Arduino Mega carrying a special shield called RAMPS (RepRap Arduino Mega Pololu Shield). This shield can hold for the stepper drivers (needed to control the stepper motors from the Arduino) as well as manage all the needed connections for the printer. The whole system is powered by a power source which can give up to 30A and 12V. The connection schematic of the system's electronics is presented in figure 69.

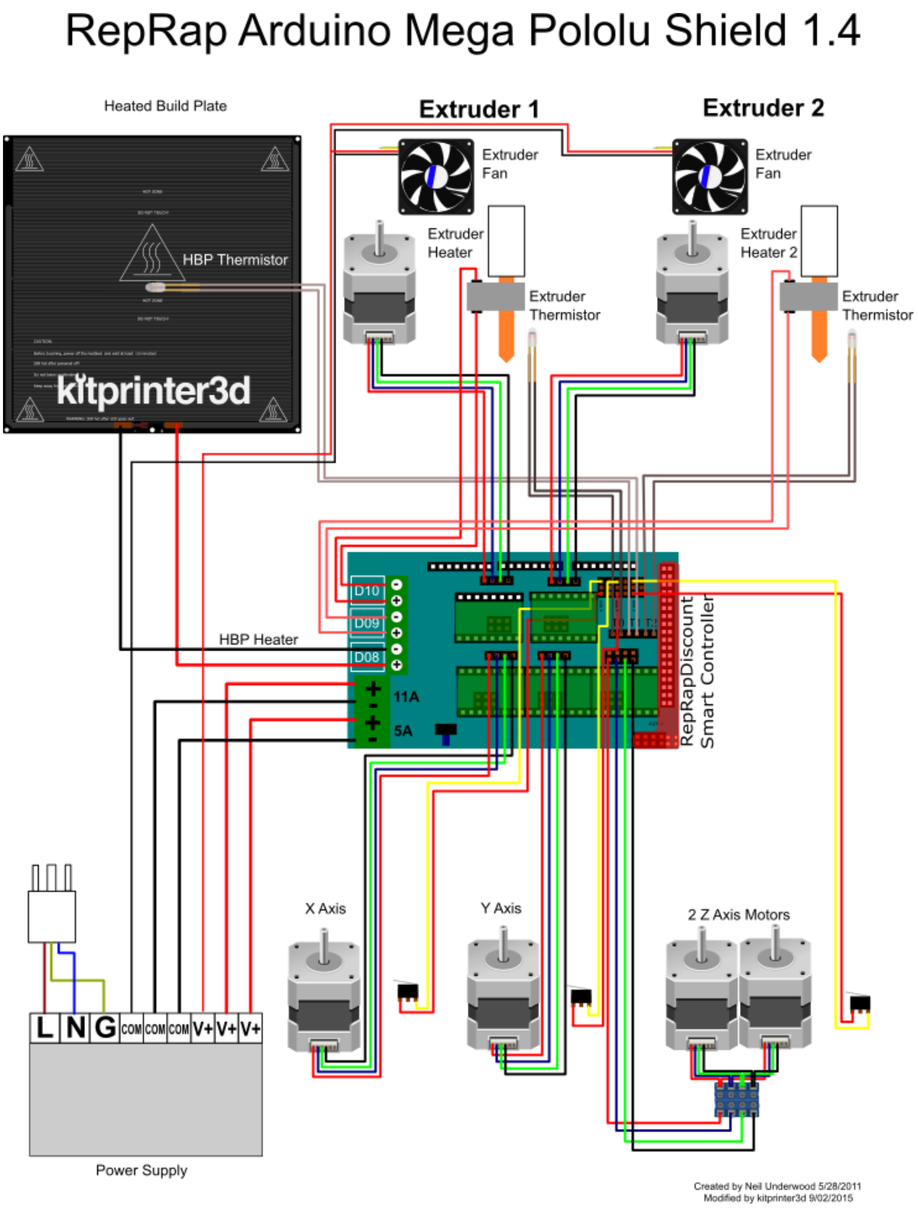


Figure 69: Connection schematics of the prusa i3 steel

This schematic includes a second extruder, which will not be present in the 3D printer which was built for this project.

Assembling the printer was mostly made with screws and it was needed to file some of the 3D printed parts used so that they would fit into their selected parts. The final result of the printer once built can be seen in figure 70.

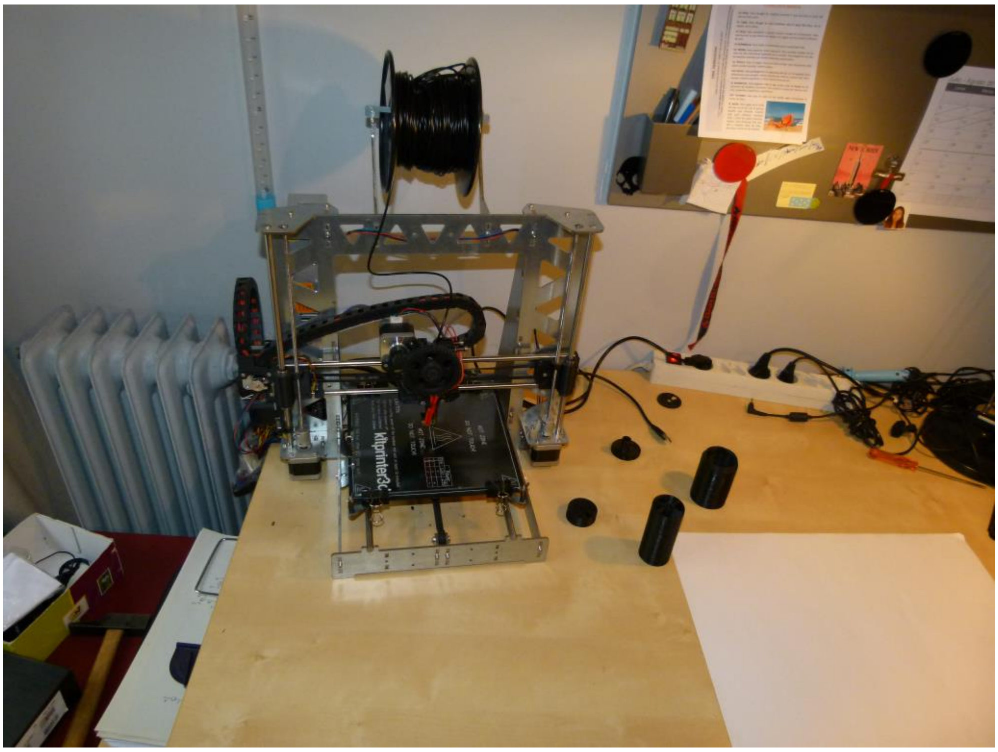


Figure 70: Prusa i3 steel used to print the pillbox parts once assembled and functional, already printed parts of the pillbox can be seen beside the printer

6.2 Printing software

A specific open-source software called PrintRun is used to control the 3D printer via a USB cable which is connected directly to the 3D printer's Arduino. The two main used programs will be Pronterface, a GUI host from which to control the printer and send the printing orders, and Slic3r, a program to convert the digital model into slices as well as setting the preferences of the print (such as support structures, infill patterns, etc…) and finally converting all the data into code which will be used by the printer to print the object. Pronterface GUI is shown in figure 71 and Slic3r GUI is shown in figure 72.

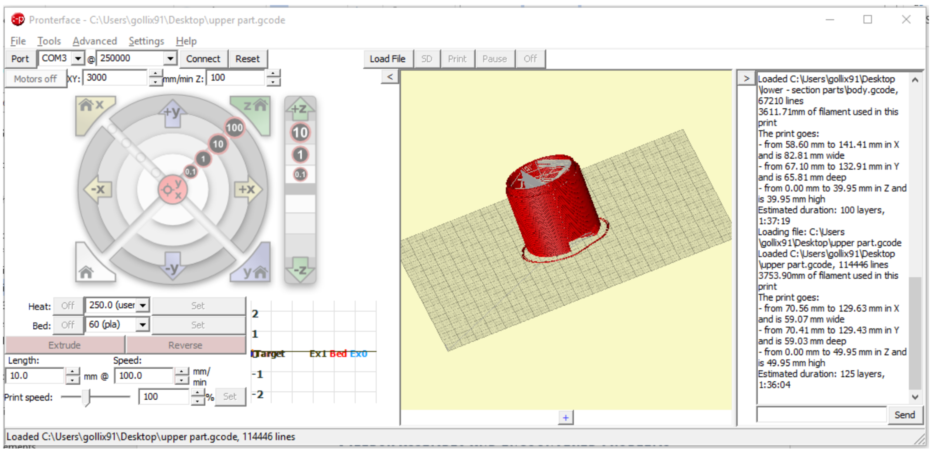


Figure 71: Pronterface GUI

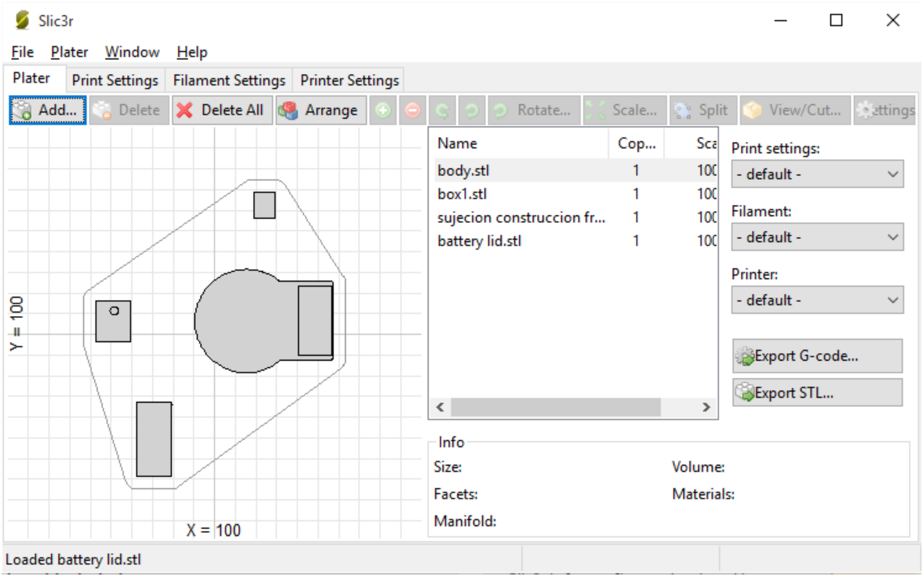


Figure 72: Slic3r GUI

6.3 Printing in 3D

Printing successfully a part in 3D using a Prusa can be challenging at first as there are some steps to follow between having the digital model and the physical one with a reasonable quality. The necessary steps followed to get the pillbox's parts will now be detailed.

The first step to take is to convert the digital model into an STL format, as there are many different extensions from which to work with in 3D modelling but the Slicer program will accept .stl files. The program which has been used to create the pillbox has been FreeCad which works with .FCSTD extension but the same program has an option to export the models to STL. Before exporting it is necessary to ensure the object is positioned in such a way that it will be easier for the printer to construct it, ensuring that there are as few parts hanging in the air possible (a cone can be easily constructed if it is positioned in an upright position while it can be extremely difficult to print if it is flat on its side as shown in figure 73, needing many support structures as shown).

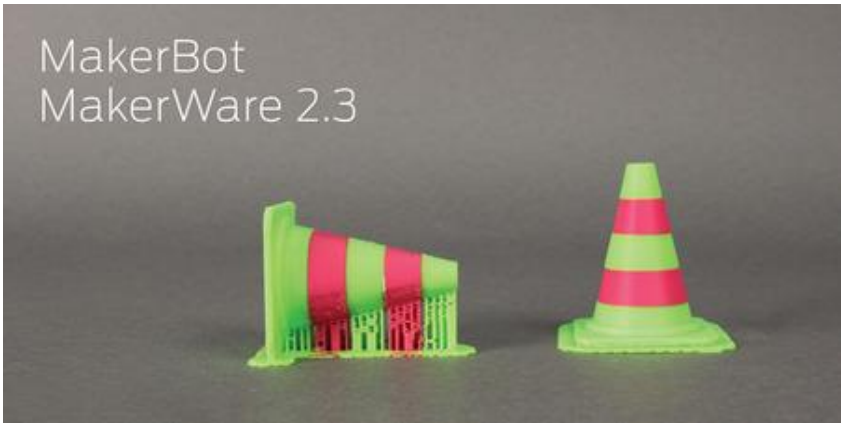


Figure 73: Example of printing a cone in a standing position and in a side position, the standing position does not need any extra support structures while the sideways position does

Next step to take is open the Slic3r program and load the .stl model. The part must be positioned in the desired place of the printing bed with the help of the left hand side panel of the program and all the preferences must be carefully chosen to ensure the part will be printed correctly. The main settings which must be correctly specified are:

- Filament diameter: 3 mm.
- Layer height: 0.4 mm.
- Infill density and pattern: 40% and honeycomb pattern (this settings specify the density and the pattern of the infill of the object, which defines how the interior parts of the object (which are not visible or accessible for a person) will be constructed).
- Support material: yes (support material is created when there are parts without support from the base as shown in Figure 72.

The Slic3r program will now create a .gcode file, this file contains a set of instructions which will be needed by the printer to print the final object, below an extract of a .gcode file to print one of the pillbox's parts is shown in chart 13.

```
; generated by Slic3r 1.1.7 on 2015-09-14 at 11:38:36

; perimeters extrusion width = 0.40mm
; infill extrusion width = 0.42mm
; solid infill extrusion width = 0.42mm
; top infill extrusion width = 0.42mm

G21 ; set units to millimeters
M107
M104 S205 ; set temperature
G28 ; home all axes
G1 Z5 F5000 ; lift nozzle

M109 S205 ; wait for temperature to be reached
G90 ; use absolute coordinates
G92 E0
M82 ; use absolute distances for extrusion
G1 F1800.000 E-1.00000
G92 E0
G1 Z0.350 F7800.000
G1 X79.147 Y79.209 F7800.000
G1 E1.00000 F1800.000
G1 X80.636 Y77.815 E1.06420 F1080.000
G1 X81.040 Y77.475 E1.08081
G1 X82.219 Y76.527 E1.12840
G1 X83.886 Y75.353 E1.19260
G1 X85.743 Y74.234 E1.26083
G1 X86.123 Y74.036 E1.27431
G1 X87.588 Y73.293 E1.32601
G1 X88.956 Y72.698 E1.37295
G1 X89.556 Y72.463 E1.39324
```

Chart 13: GCode set of instructions for the 3D printer.

Next and final step will be to boot the Pronterface program and establish a connection with the 3D printer via a USB cable. It is mandatory to calibrate the extruder's Z position in respect to the printing bed to ensure a quality print. This can be accomplished by setting the extruder at the origin of the Z axis with the Pronterface program and adjust the screws at the bottom of the printing bed so that a sheet of paper can be slipped between the bed and the extruder easily while feeling the extruder pressing lightly onto the paper sheet. Once the extruder's position is calibrated the .gcode file can be loaded into the Pronterface program where a preview of the print will be shown in the screen. The object can now be sent to the printer to be printed.

The main problem while trying to print the first parts was that the part became messed up because it didn't stick properly to the printing bed, after consulting in Internet forums the solution was found to be to apply common hairspray to the printing bed. "Nelly" hairspray was the most recommended hairspray for the 3D prints and it was proven thay it made the first printed layers strongly stick to the bed avoiding the object to move while printing. [37]

6.4 PRINTED PARTS

Printing the parts was a slow process in which many parts were either printed wrongly or just as test structures. Discarded parts can be seen in figure 74 as a consequence of failed prints.

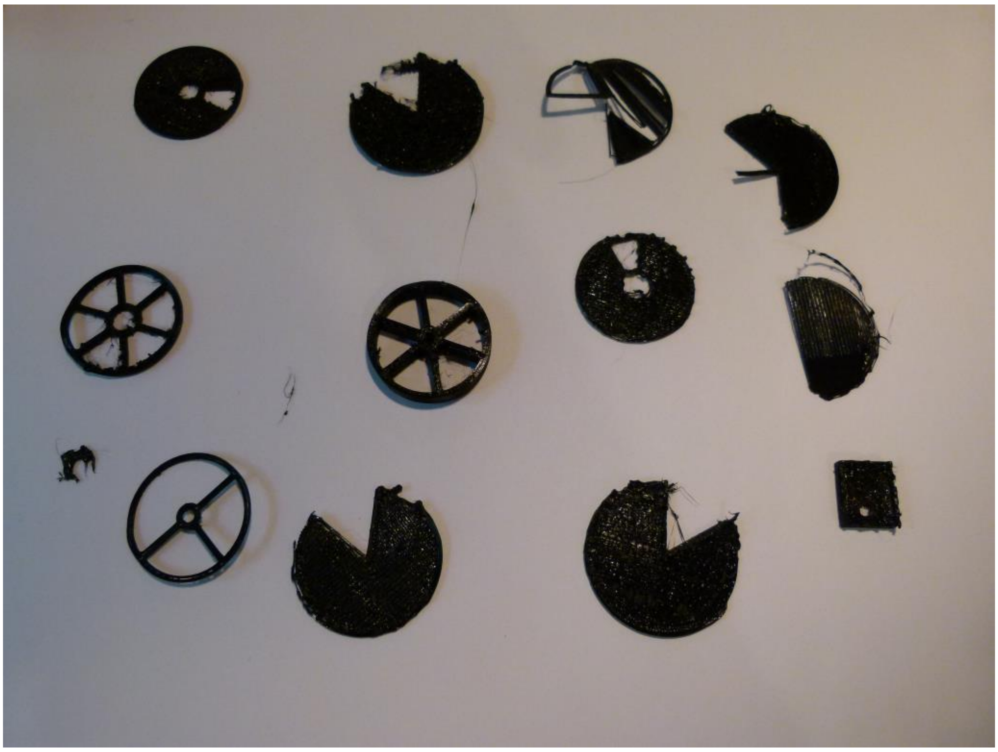


Figure 74: Discarded parts due to their lack of quality or just printed as test parts.

Even though many hours were spent in trial and error finally parts with the necessary quality were oprinted at last. Below some of the final parts which will be used in the prototype can be seen in figure 75 and 76.

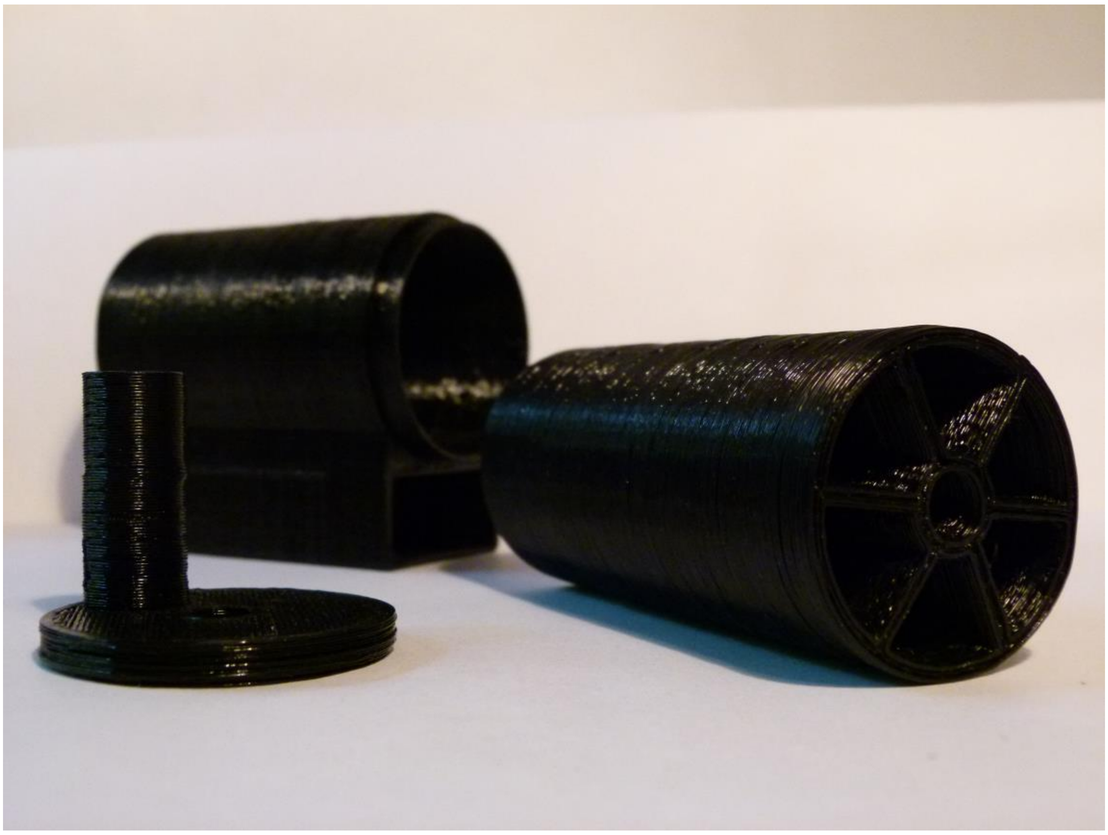


Figure 75: Parts with the necessary quality which will be used to assemble the prototype: pill storage, middle body and pill pipe can be seen.

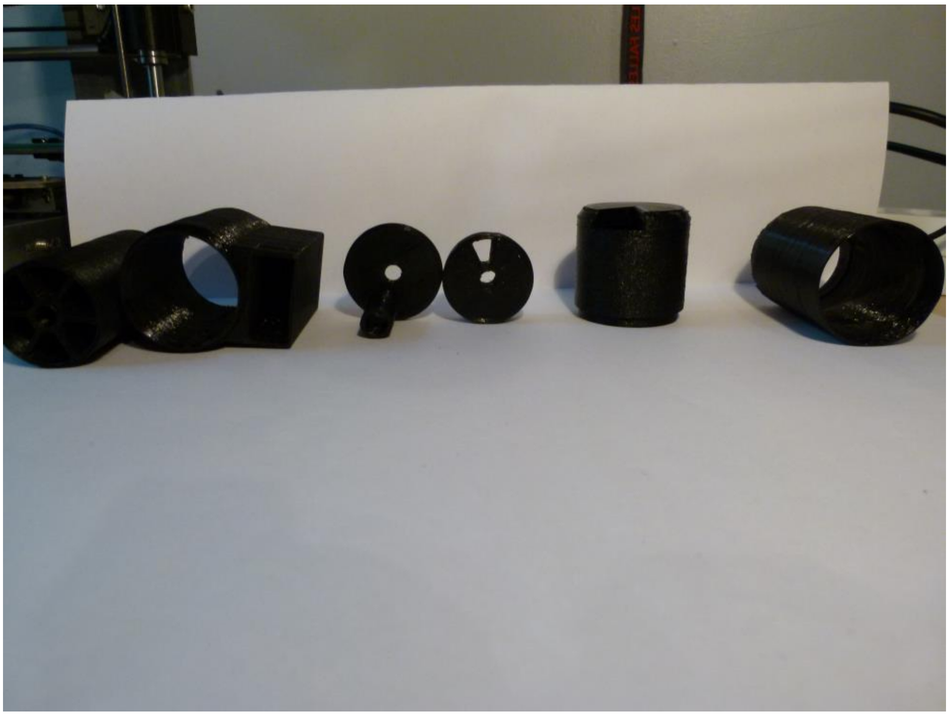


Figure 76: More final parts are shown in the image

6.5 PILLBOX ASSEMBLY AND ENCOUNTERED PROBLEMS

Assemblying the pillbox involves assemblying together the different 3D printed parts alongside with the circuits or components and some glue or screws where needed. The used glue has beenSuperGlue (cyano acrylate glue), using it to securely hold the prototype's parts together. Tests will be conducted later on to check the durability of this material and make any necessary changes if it is necessary. The pillbox has been assembled following the simplified scheme.

After testing the pillbox integrity [see Testing the pillbox section]. It has been necessary to use sandpaper to sand some imperfections in some of the parts and interconnect them successfully. In some of the parts a void was created were two parts should have been touching but due to printing imperfections there weren't touching. If the hole was small it was filled with the same glue used to glue the parts together. If the hole was of considerable dimensions the hole were filled and the parts welded together using printable plastic strands found in thingiverse and printed with the 3D printer created and designed for that purpose. These strands were welded together with the pillbox's parts using a soldering iron. [38] The finished pillbox after the assembly is shown in figures 77 and 78.

Figure 77: Fully assembled pillbox upper view

Figure 78: Fully assembled pillbox full view

7 TESTING THE PILLBOX

Throughout this section the pillbox will be put through some tests to ensure both its operation is carried out successfully and its strength and general cohesion is the expected one. When a test is failed it will be noted the solution taken and if necessary the document will be changed to reflect those changes (for example, if the problem is in the code the code presented in this document will be the corrected version).

These tests include the name of the test in which the area to test is indicated, the procedure to carry out the test and the expected result which would make the pillbox pass the test. If the test is passed it will be indicated in the Result column with a “Success” mark, on the other hand if the test is failed the coum will indicate what the outcome of the test was and a proposed solution to fix the problem and make the pillbox pass the test will be supplied in the last column.

7.1 MISCELLANEOUS TESTS

Tests shown in chart 14 are not related to any particular area of the pillbox as they are tests conducted in areas which concern to the pillbox itself independently of the functioning mode it is working on.

Test number	Test	Procedure	Expected result	Result	Proposed solution
1	Switch on	Connect Arduino board inside the pillbox to a power source	Pillbox switches on normally	Success	-
2	General cohesion	Exert minor force trying to move the fixed parts	Parts will not move	Fail: applying minor pressure made some of the pillbox parts become loose	Sanding some plastic connections and refilling holes with glue and/or melted plastic solved this problem. [See Pillbox assembly and encountered problems.]
3	Fall test	Drop the pillbox from 50 cm to the floor	Pillbox will not break in any part	Partial success, the pillbox sustained the fall although it is not recommendable for the pillbox to be subjected to being dropped to the fall.	The pillbox, as well as most of other electronical devices can be sensible and it is not recommended at all that it is frequently dropped.

Chart 14: Miscellaneous tests

7.2 TESTS FOR THE DEFAULT STAGE

The tests conducted in chart 15 are tests perfomed while the pillbox is in the default stage when the pillbox is found in an idle state with no pill notifications and no pills pending to be intaken and the Bluetooth switch being initially in the off position.

Test number	**Test**	**Procedure**	**Expected result**	**Result**	**Proposed solution**
4	Push button	Push the button when there are no notifications	Nothing happens	Fail, the pillbox dispensed the next pill although it wasn't time yet for the pill to be dispensed.	A line of code was added to the code which checks if the dispensing flag is on and that the pill has to be dispensed before commencing the dispensing procedure.
5	Switch usage	Switch the BT switch from the off to the on position	BT LED comes on and pillbox prepares to connect via BT	Success	-
6	Connect to BT (switch off)	Try to connect from an outer device to the pillbox while the BT switch is in the Off position	Pillbox may pair but ignore any orders	Failure, the pillbox paired and started processing some of the orders given while it shouldn't if the switch is off	Part of the code was wrapped in an if-else statement so that if the BT switch was off it, the Arduino board completely ignores anything received by the Serial communication (which is connected to the Bluetooth module).

Chart 15: Default stage tests

7.3 TESTS FOR THE MANAGEMENT STAGE

Tests performed in chart 16 are tests conducted when the pillbox was in the management stage, with the Bluetooth switch in the on position and the pillbox paired in most cases with an external device. To conduct these tests an Android phone was used with the BlueTerm [41] application, an application which mainly consists on a terminal in which to send via Bluetooth commands to a Bluetooth device.

Test number	**Test**	**Procedure**	**Expected result**	**Result**	**Proposed solution**
7	Connect to BT (switch on)	Connect to pillbox while BT switch is on	Pillbox pair's and accepts incoming commands	Success	-
8	Start refilling procedure	Send the command to start the refilling sequence	The pillbox successfully initiates the refilling procedure by notifying to it the device	Success	-
9	Send closing command without having inserted any	Send closing command once in the refilling sequence	The pillbox must close the refilling procedure correctly	Success	-
10	Insert more pills than there is space for	Insert physically more pills than the ones allowed in the pillbox's storage space	The pill must be clear it doesn't fit into the storage space and will fall out of the pillbox and not move around in the pillbox	Fail, it wasn't clear if the pill compartment was full and there was space for the pill to fall inside the pillbox.	The distance between the lid and the compartment was reduced so that the compartment was more visible and less space was left for the pill to fall around the pillbox.
11	Send different numbers once in the refilling stage	Once in the refilling different numbers (so that the storage pill part rotates)	The pill storage must rotate correctly and align with the pillbox's entry hole the different pill compartments	Success	-

		must be sent via BT to access the different pill compartments			
12	Send closing command after having refilled correctly the pillbox	Send the closing command after having refilled the pillbox	The pillbox must shutdown the pill refilling sequence	Success	-
13	Request logs	Send the requesting logs command while in BT management mode	The logs must be correctly sent by the pillbox	Partial success, the pillbox did send the logs back but they had a confusing and incomprenhensible format	The code was modified so that the logs were sent byte by byte so that the receiving device could process them in time.
14	Request for the uptime of the pillbox	Send the command to request the pillbox's uptime	The time since the pillbox was switched on must be sent by the pillbox	Success	-
15	Request for the current dose timetable	Send the command to request the dose timetable the pillbox actually has	The pillbox must send the actual timetable it has	Success	-
16	Overwrite the dose timetable	Send a command and a new timetable to the pillbox so that the pillbox	The pillbox must receive and overwrite the timetable and send a confirmation command to the device after so	Success	-

		overwrites the timetable with the new one			
17	Request the pillbox's timetable to check if it has been updated after test number 16	Send the request for the timetable	The pillbox must send the timetable which was sent in test number 16	Success	-

Chart 16: Management stage tests

7.4 TESTS FOR THE DISPENSING STAGE

Tests in chart 17 are tests conducted or related to the dispensing stage in the pillbox, when the pillbox is prepared to dispense a pill (with the pillbox light notifying a pending pill to be dispensed) or while the inner parts are rotating to dispense the pill once the dispensing procedure has started.

Test number	Test	Procedure	Expected result	Result	Proposed solution
18	Start dispensing procedure for each one of the pills	When the pillbox is notifying for a pill then the button is pressed so that the dispensing procedure starts.	Once the button has been pressed the pillbox must start the dispensing procedure and a pill must be dispensed.	Success	-
19	Switch BT on while dispensing procedure is ongoing	When the pillbox is dispensing a pill the switch will be changed to on.	The position change of the switch must not affect the dispensing procedure.	Success	-
20	Press button again while dispensing procedure is ongoing	When the pillbox is dispensing a pill the button will be pushed.	The button press must not affect the dispensing procedure.	Fail, the pillbox would dispense a pill that wasn't due yet after dispensing the current pill	This bug was the same that caused test number 4 to fail, once corrected it solved both tests.

Chart 17: Dispensing stage test

8 Planning and budget

In this section the planning for the project's work has been outlined as well as the budget needed to complete the project.

8.1 Planning

On this planning the hours have been accounted for in the table shown below. Due to constraints in the college's timetable while working on the project it has been assessed that each day the work on the project was of 5 hours a day. Chart 18 shows the amount of hours which have been invested to produce the prototype of the pillbox. Charts 19 and 20 show the Gantt chart of the different activities with the amount of days for each activity.

Concept	Amount of hours	Amount of days
Research and documentation	50,00	10
Conceptual design	20,00	4
Part models design	60,00	12
Electronics design	30,00	6
Software design	40,00	8
Assemblying 3D printer	45,00	9
Learning 3D printer basics	10,00	2
Printing the parts	40,00	8
Pillbox assembly	20,00	4
Pillbox testing	30,00	6
Memory writing	80,00	16
Total	425	85

Chart 18: Planning schedule

Concept	Amount of hours	Amount of days	Day 1 - 5	Day 6 - 10	Day 11 -15	Day 16 -20	Day 21 - 25	Day 26 - 30	Day 31 - 35	Day 36 - 40	Day 41 - 45
Research and documentation	50,00	10									
Conceptual design	20,00	4									
Part models design	60,00	12									
Electronics design	30,00	6									
Software design	40,00	8									
Assemblying 3D printer	45,00	9									
Learning 3D printer basics	5,00	1									
Printing the parts	40,00	8									
Pillbox assembly	20,00	4									
Pillbox testing	30,00	6									
Memory writing	80,00	16									

Chart 19: Gantt chart, 1st part

Concept	Amount of hours	Amount of days	Day 46 - 50	Day 51 - 55	Day 56 - 60	Day 61 - 65	Day 66 - 70	Day 71 - 75	Day 76 - 80	Day 81 - 84
Research and documentation	50,00	10								
Conceptual design	20,00	4								
Part models design	60,00	12								
Electronics design	30,00	6								
Software design	40,00	8								
Assemblying 3D printer	45,00	9								
Learning 3D printer basics	5,00	1								
Printing the parts	40,00	8								
Pillbox assembly	20,00	4								
Pillbox testing	30,00	6								
Memory writing	80,00	16								

Chart 20: Gantt chart 2nd part

8.2 BUDGET

Costs of this project has been calculated with all the electronic and material costs along with the work hours, paying 20€ for each worked hour. Chart 21 shows the amount of money the total costs amount to and the separate costs of each of the different parts involving the project.

Concept	Quantity	Cost	Total
Work hours	425	20€	8500€
Computer	1	500€	500€
3D printer	1	400€	400€
Funduino Pro Mini	1	5.80€	5.80€
USB to TTL serial converter	1	2.99€	2.99€
Bluetooth module	1	11.13€	11.13€
LEDs, switch,button and cables	1	1.20€	1.20€
Piezo speaker	1	1.89€	1.89€
Real-Time Clock module	1	2.66€	2.66€
Servos	2	3€	6€
3D plastic filament roll	1	20€	20€
Total			9.426 €

Chart 21: Budget chart

9 CONCLUSIONS AND FUTURE WORK

This project's aim was to produce an electronic device which would cover a problem occurring in the present day and offer a solution which could help to save lives as it deals with in a medical area. The project's aims have been successfully fulfilled with the pillbox's fully operational prototype, which was the scope of this project.

The final outcome has been a pillbox prototype made out of SLA plastic non-toxic to humans and perfectly capable of storing up to 18 pills of three different types at a time. The total length of the pill is of 23 cm and 5 cm wide except in the battery part where the pillbox measures 6.5 cm from front to back. The net weight of the pillbox without pills nor battery is 240 grams. The pillbox is expected to have a life of around 1.5 / 2 years without any maintenance.

On future work the creation of the software program needed to manage the pillbox with an easy-to-use interface as well as improving the pillbox design. It will be necessary to prioritize in those areas which were originally intended for this prototype but due to a lack of memory on the Arduino board or other constraints haven't been implemented. Main future improvements in these area include:

- A remote notification system for the pillbox to notify on real-time when the user does not take a pill.
- A battery monitoring circuit is an important improvement to work on as well as the pillbox notifying when the pillbox's battery is low.
- Designing a charger which allows the pillbox to charge its battery while powered on and without the need to remove its battery.
- A slimmer and lighter design which will allow the users to carry the pillbox around more easily without losing pill storage space.
- Increasing the amount of pill types available.

During the development of this project I have been able to reinforce many different areas learnt thorugh my college degree such as electronics, microprocessors and software programming while having to learn many different ones not covered such as 3D printing technologies and 3D design. One of the most valuable things I personally acquire from this project was the challenge of conceiving, designing and producing an electronic device from scratch up to its final stages with a positive outcome as the scope intended for this project was successfully achieved. This has allowed me to see the complexity of having to cover all the different areas in creating a prototype, as it became a much harder task than at first was envisioned it would be, and consequently a more rewarding feeling when it was successfully accomplished.

10 REFERENCES

[1] Christina Caron, Nurse Gives Patient Paralytic Instead of Antacid, ABCNews, Nov. 21, 2011 (Retrieved Sep. 25, 2015 from http://abcnews.go.com/Health/nurse-patient-paralytic-antacid/story?id=14997244 , Last accessed: 25/09/2015).

[2] Jane E. Brody, Too Many Pills for Aging Patients, The New York Times, Apr. 16, 2012, (Retrieved Sep. 25, 2015 from http://well.blogs.nytimes.com/2012/04/16/too-many-pills-for-aging-patients/).

[3]Candy Sagon, Too Many Pills: Are We Over-Medicated As We Age?, AARP, Apr.25, 2012, (Retrieved Sep. 25, 2015 from http://blog.aarp.org/2012/04/25/too-many-pills-are-we-overmedicated-as-we-age/).

[4]Sabriya Rice, Are you taking too many meds?, CNN, May 31, 2011, (Retrieved Sep. 25, 2015 from http://edition.cnn.com/2011/HEALTH/05/31/med.nation.too.many.meds/).

[5]Servicios de Farmacia de Hospitales Comarcales de Cataluña, "Manual para la evaluación de nuevos fármacos en el hospital, Documentos XIIIa Jornada Servicios de Farmacia de Hospitales Comarcales de Cataluña, Nov. 2003, (Retrieved Sep. 25, 2015 from http://www.elcomprimido.com/FARHSD/ManualOlotCapitulo1_cast_Version3.PDF).

[6] Adhere Tech, (Last accessed Sep. 25, 2015 in http://adheretech.com/).

[7] iMediPac, (Last accessed Sep. 25, 2015 in https://en.medissimo.fr/en/imedipac/).

[8] TinyLogics, (Last accessed Sep. 25, 2015 in http://tinylogics.com/).

[9] My-Ubox, (Last accessed Sep. 25, 2015 in http://my-ubox.com/).

[10] GlowCaps, (Last accessed Sep. 25, 2015 in http://www.glowcaps.com/).

[11] MedMinder, (Last accessed Sep. 25, 2015 in http://www.medminder.com/).

[12] ePill, (Last accessed Sep. 25, 2015 in http://www.epill.com/).

[13] Phillips Medication Dispenser, (Last accessed Sep. 25, 2015 in http://www.managemypills.com/content/).

[14] P. Michael Ho, MD, PhD; Chris L. Bryson, MD, MS; John S. Rumsfeld, MD, PhD; Medication Adherence: Its importance in Cardiovascular Outcomes; Circulation: 2009; 119: 3028-3035; doi: 10.1161/CIRCULATIONAHA.108.768986, (Retrieved Sep. 25, 2015 from http://circ.ahajournals.org/content/119/23/3028.full).

[15] Arduino Pro Mini reference, (Retrieved Sep. 25, 2015 from https://www.arduino.cc/en/pmwiki.php?n=Main/ArduinoBoardProMini).

[16] Raspberry Pi, (Retrieved Sep. 25, 2015 from https://www.raspberrypi.org/).

[17] WiPy, (Retrieved Sep. 25, 2015 from http://wipy.io/).

[18] FreeCad, (Retrieved Sep. 25, 2015 from http://www.freecadweb.org/).

[19] Beatriz G. Portalatín, ¿Dónde van a parar los 'viejos' aparatos médicos?, El Mundo, Nov. 11, 2012, (Retrieved Sep. 25, 2015 from http://www.elmundo.es/elmundosalud/2012/11/07/tecnologiamedica/1352316792.html).

[20] M. E. Molina Corchado Enmedio, "Envase para pastillas", Nº pub.: 1 044 714, Nº sol. : U 009902981, (Retrieved Sep. 25, 2015 from http://www.oepm.es/pdf/ES/0000/000/01/04/47/ES-1044714_U.pdf).

[21] J. Pérez Adsuar, "Pastillero", Nº pub.: 1 068 942, Nº sol. : U 200802049, (Retrieved Sep. 25, 2015 from http://www.oepm.es/pdf/ES/0000/000/01/06/89/ES-1068942_U.pdf).

[22] GRAFIA, S.A, M. Fernández Alonso, "Pastillero avisador", Nº pub.: 1 064 960, Nº sol. : U 200700451, (Retrieved Sep. 25, 2015 from http://www.oepm.es/pdf/ES/0000/000/01/06/49/ES-1064960_U.pdf).

[23] TechTarget, Bluetooth Definition, Nov. 2006, (Retrieved Sep. 25, 2015 from http://searchmobilecomputing.techtarget.com/definition/Bluetooth).

[24] Bluetooth SIG, Bluetooth Basics, Mar. 2011, (Retrieved Sep. 25, 2015 from http://www.bluetooth.com/Pages/Basics.aspx).

[25] *Enabling Timekeeping Function and Prolonging Battery Life in Low Power Systems*, NXP Semiconductors, 2011, (Retrieved Sep. 25, 2015 from http://www.digikey.com/en/articles/techzone/2011/dec/enabling-timekeeping-function-and-prolonging-battery-life-in-low-power-systems).

[26] I2C-Bus, Detailed Introduction, Primer, (Retrieved Sep. 25, 2015 from http://www.i2c-bus.org/).

[27] Fritzing, (Retrieved Sep. 25, 2015 from http://fritzing.org/home/).

[28] 3D Printing, What is 3D printing? , (Retrieved Sep. 25, 2015 from http://3dprinting.com/what-is-3d-printing/).

[29] How Stereolithography (3-D Layering) Works from HowStuffWorks.com, (Retrieved Sep. 25, 2015 from http://computer.howstuffworks.com/stereolith.htm/printable).

[30] Rain Noe, What is the difference between Selective Laser Sintering, Direct Metal Laser Sintering, Laser Melting and Laser Cusing?, Core 77, Feb. 18, 2014, (Retrieved Sep. 25, 2015 from http://www.core77.com/posts/26457/production-methods-whats-the-difference-between-selective-laser-sintering-direct-metal-laser-sintering-laser-melting-and-lasercusing-26457).

[31] Renee Eaton, What's the Difference Between SLS and SLM - and Why Care?, RapidMade, June 30, 2014, (Retrieved Sep. 25, 2015 from http://www.rapidmade.com/rapidmade-blog/2014/6/30/ycjnxytvpt8n85gqutk5wj67cmx4t7).

[32] Buy3DPrinter, Laminated Object Manufacturing – LOM, Buy3DPrinter, April 2013, (Retrieved Sep. 25, 2015 from http://www.buy3dprinter.org/3dprintingtechnologies/laminated-object-manufacturing-lom/).

[33] Rob Martinez, When to choose hobby vs. commercial level 3D printers – Materials, SupplyBetter, Nov. 1, 2013, (Retrieved Sep. 25, 2015 from https://www.supplybetter.com/blog/choosing-fff-vs-fdm.html).

[34] Luke Chilson, The Difference Between ABS and PLA for 3D Printing, ProtoParadigm, Jan. 26, 2013, (Retrieved Sep. 25, 2015 from http://www.protoparadigm.com/news-updates/the-difference-between-abs-and-pla-for-3d-printing/).

[35] 3DPrintingForBeginners, What Material Should I Use For 3D Printing?, 3DPrintingForBeginners, Feb. 23, 2015, (Retrieved Sep. 25, 2015 from http://3dprintingforbeginners.com/filamentprimer/).

[36] KitPrinter3D, (Retrieved Sep. 25, 2015 from http://www.kitprinter3d.com/es/).

[37] David Monterde, Como evitar el warping, Aug. 17, 2014, (Retrieved Sep. 25, 2015 from http://elfilamento.com/2014/08/17/como-evitar-el-warping/).

[38] maurerpe, PLA Welding: Join PLA like a boss, MakerBotThingiverse, May 24, 2012, (Retrieved Sep. 25, 2015 from http://www.thingiverse.com/thing:23606).

[39] Jeelab, RTCLib, Adafruit, (Retrieved Sep. 25, 2015 from https://github.com/adafruit/RTClib).

[40] Pymasde.es, BlueTerm, June 24 2013, (Retrieved Sep. 25, 2015 from https://play.google.com/store/apps/details?id=es.pymasde.blueterm&hl=es)

PDS documentation: College Of Engineering-University of Wisconsin, Product Design Specifications, Apr. 26 2009, (Retrieved Sep. 25, 2015 from http://homepages.cae.wisc.edu/~me349/lecture_notes/product_design_spec.pdf).

Conceptual design documentation: Department of Electrical and Electronic Engineering, CONCEPTUAL DESIGN REPORT, Middle East Technical University, Dec. 17, 2013, (Retrieved Sep. 25, 2015 from http://www.eee.metu.edu.tr/~design/doku.php?id=conceptual_design).

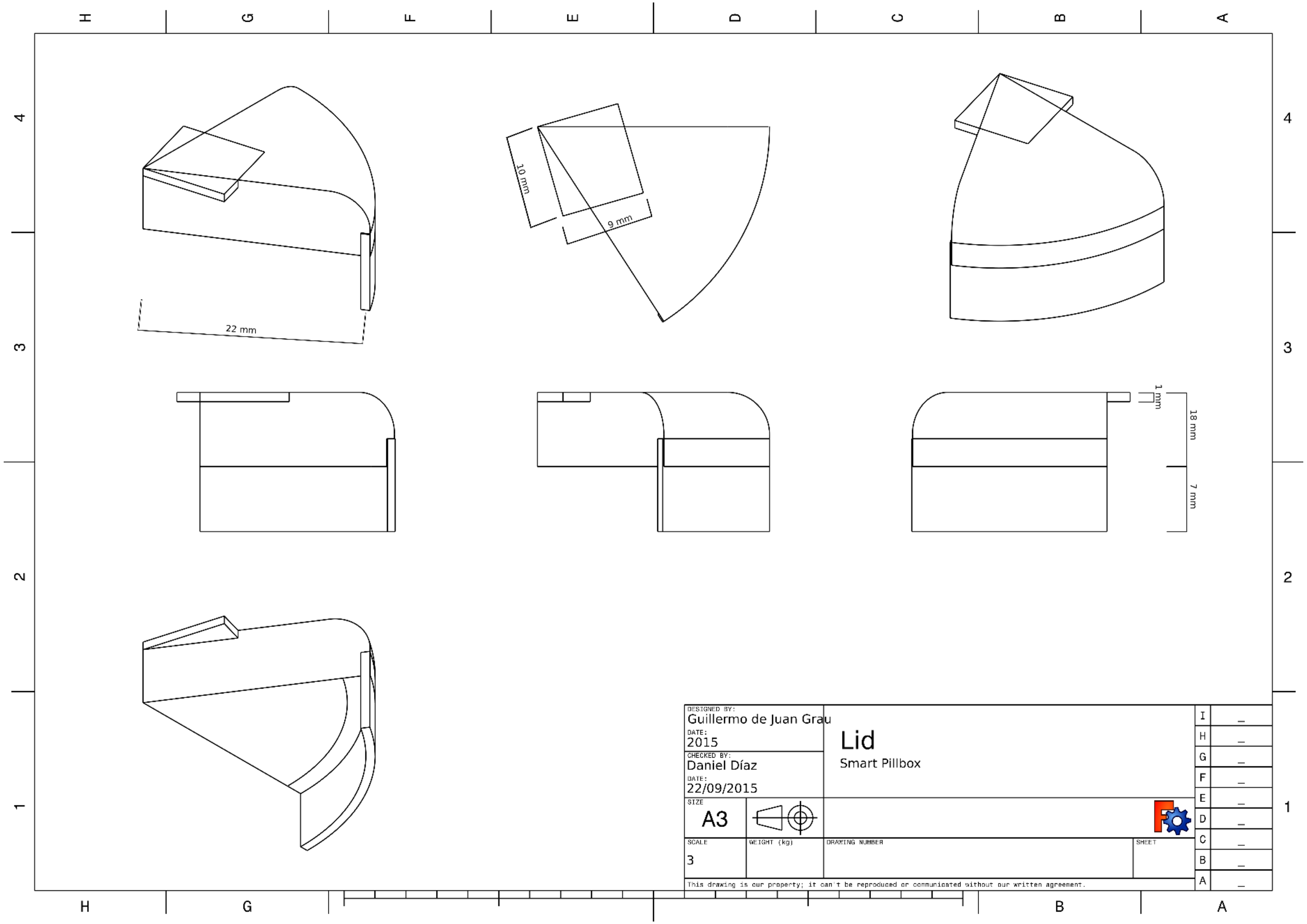
Block diagram creation: Draw.io, (Last accessed Sep. 25, 2015 in http://www.draw.io).

Code flowchart creations: Gliffy, (Last accessed Sep. 25, 2015 in http://www.gliffy.com).

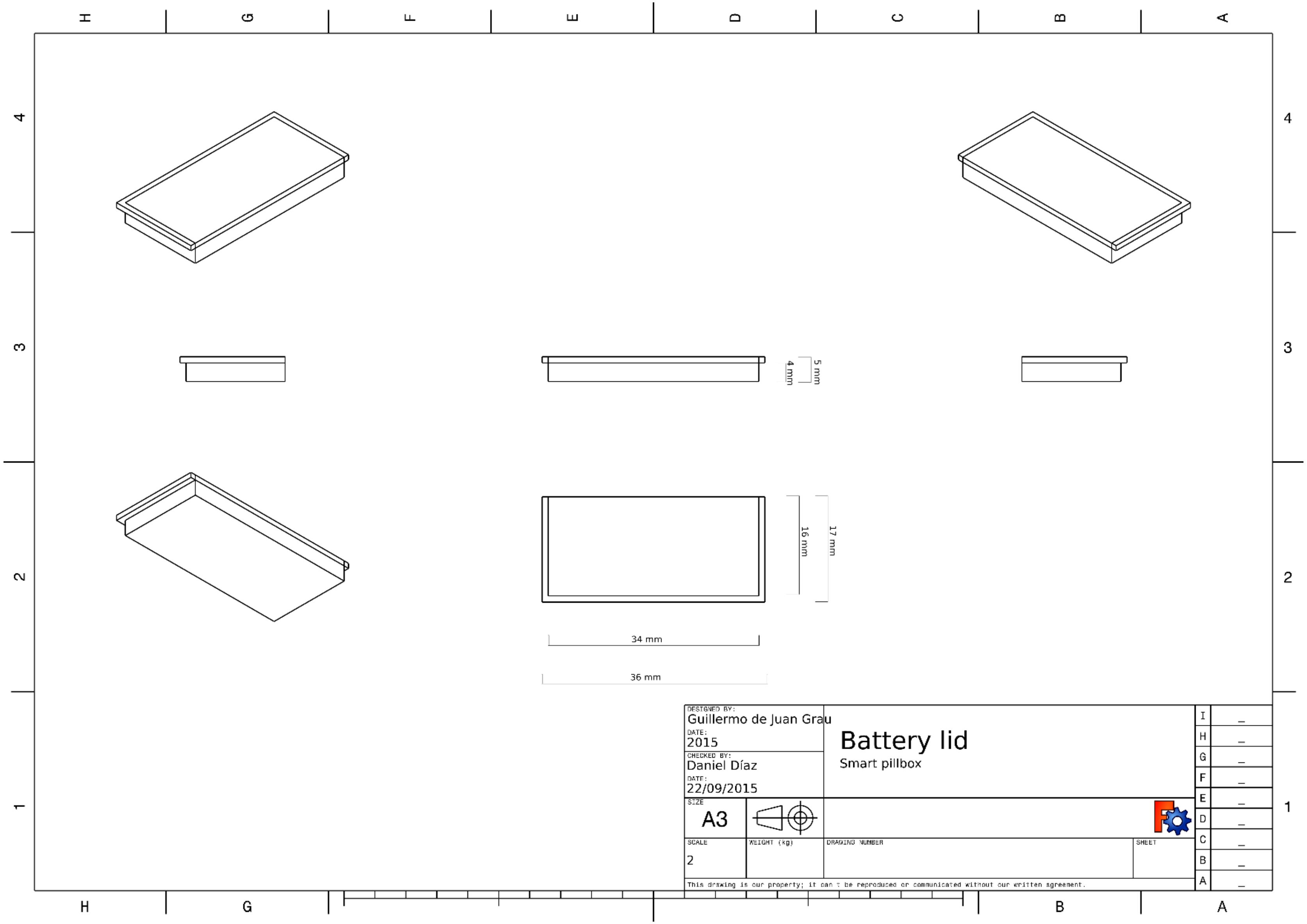
Circuit schematic creation: Fritzing, (Last accessed Sep. 25, 2015 in http://www.fritzing.org).

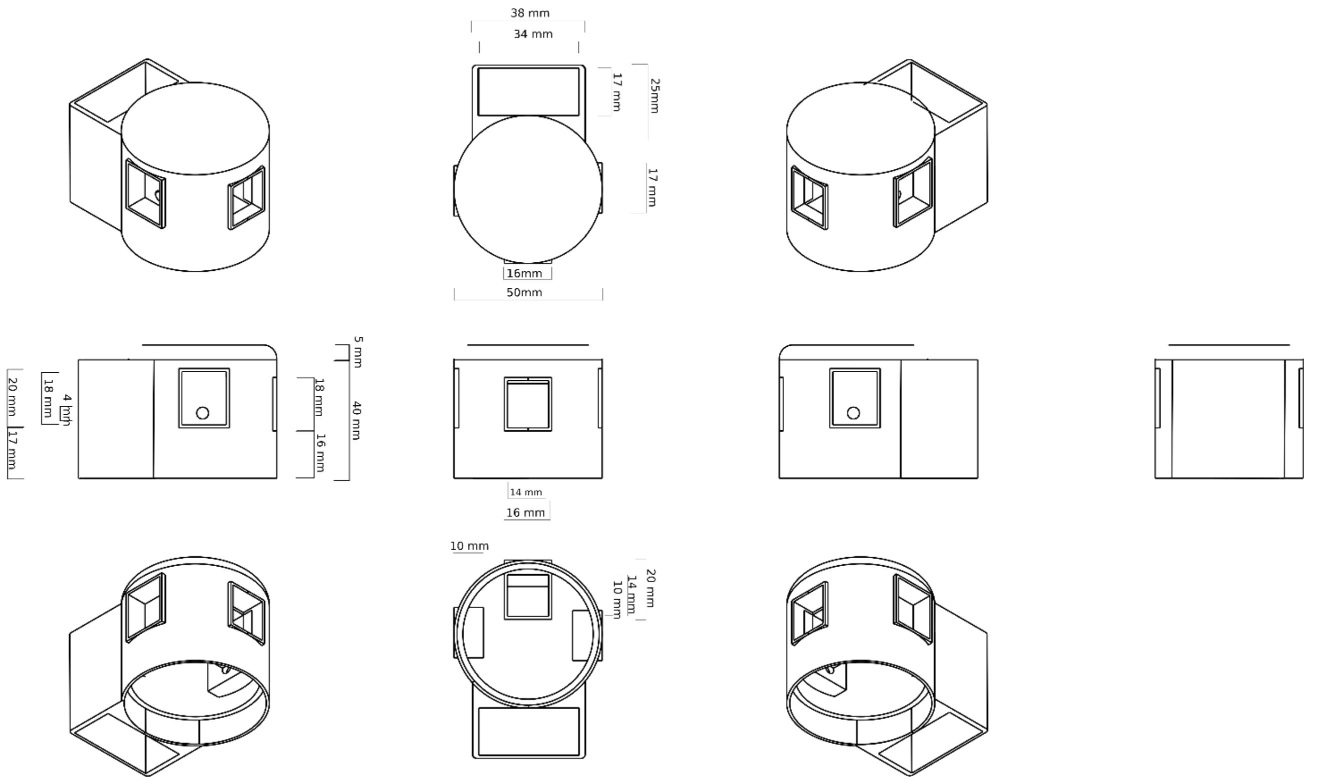
3D model design: FreeCad, (Last accessed Sep. 25, 2015 in http://www.freecadweb.org).

11 Annex



22 mm
10 mm
9 mm
1 mm
18 mm
7 mm
DESIGNED BY:
Guillermo de Juan Grau
DATE:
2015
CHECKED BY:
Daniel Díaz
DATE:
22/09/2015
SIZE
A3
Lid
Smart Pillbox
SCALE
3
WEIGHT (kg)
DRAWING NUMBER
SHEET
This drawing is our property; it can't be reproduced or communicated without our written agreement.

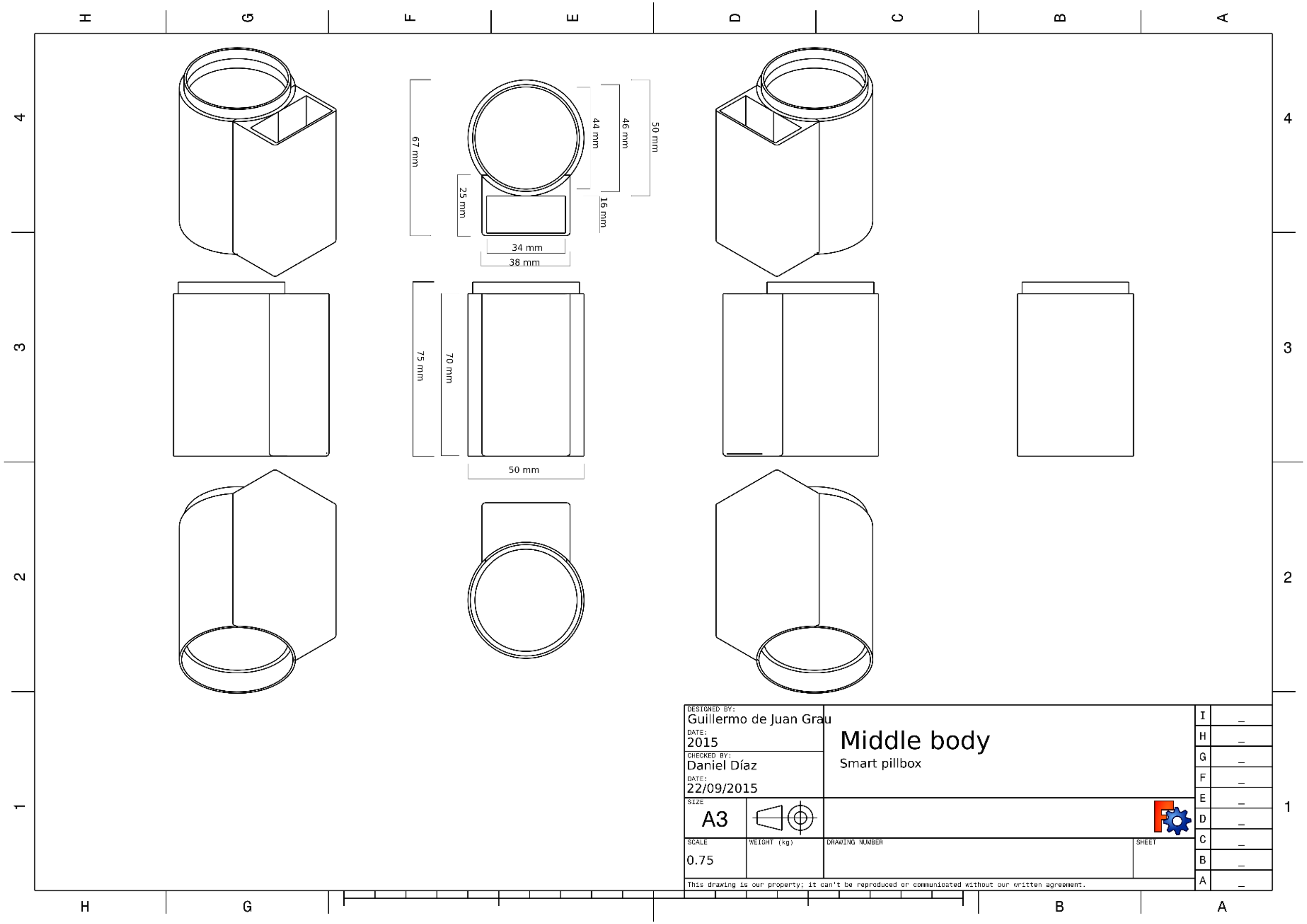

5 mm
4 mm
16 mm
17 mm
34 mm
36 mm
DESIGNED BY:
Guillermo de Juan Grau
DATE:
2015
CHECKED BY:
Daniel Díaz
DATE:
22/09/2015
Battery lid
Smart pillbox
SIZE
A3
SCALE
2
WEIGHT (kg)
DRAWING NUMBER
SHEET
This drawing is our property; it can't be reproduced or communicated without our written agreement.

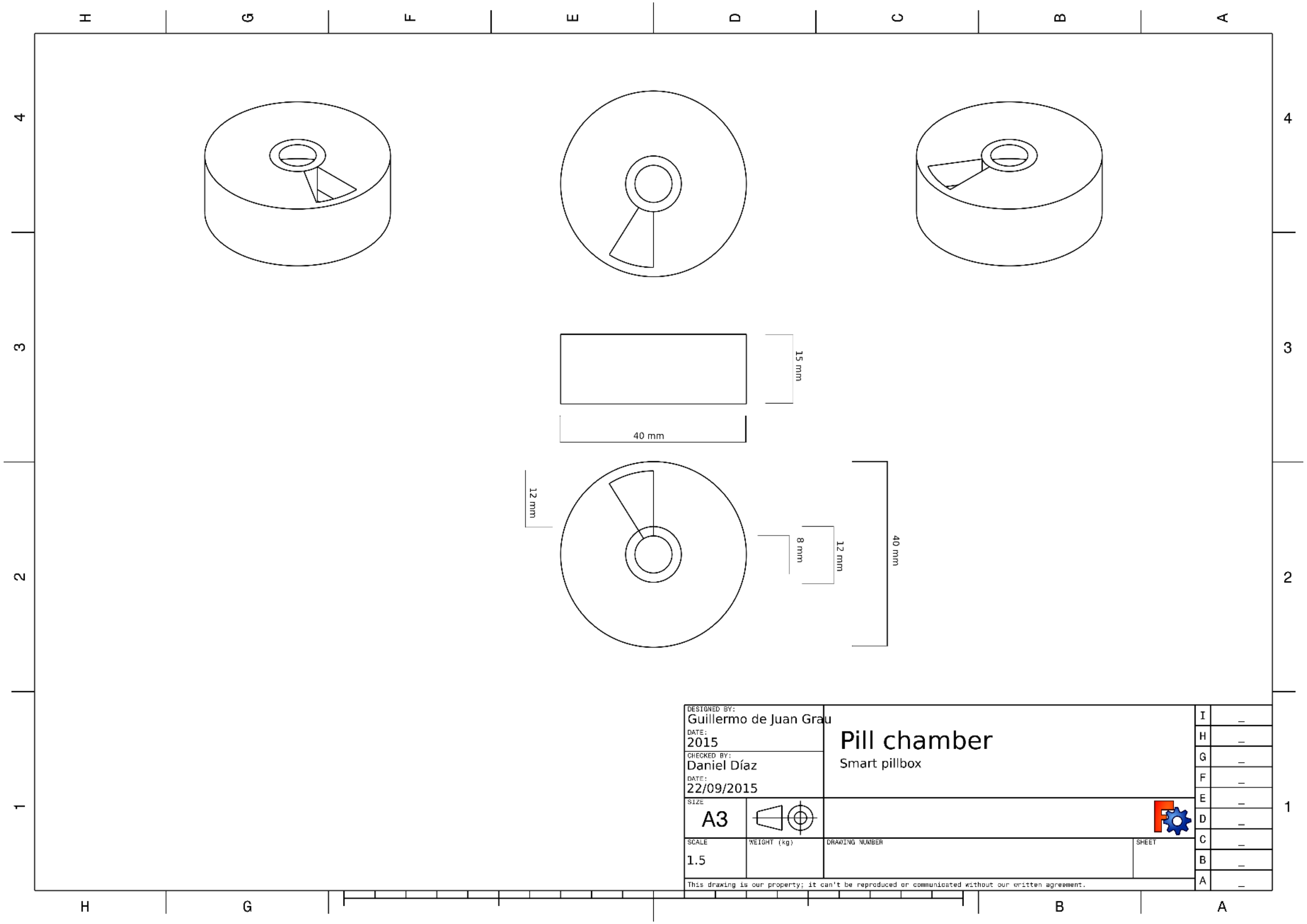


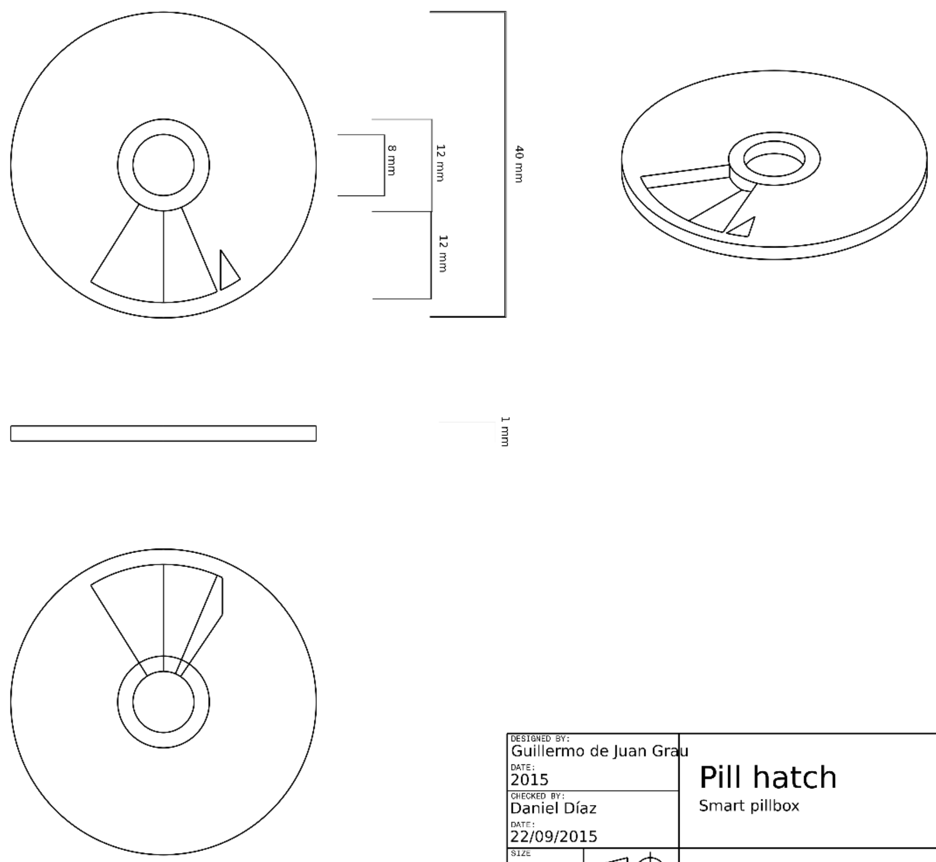
DESIGNED BY: Guillermo de Juan Grau
DATE: 2015
CHECKED BY: Daniel Díaz
DATE: 22/09/2015
SIZE: A3
SCALE: 0.75
WEIGHT (kg):
DRAWING NUMBER:
SHEET:

Lower body

Smart pillbox


67 mm
25 mm
44 mm
46 mm
50 mm
16 mm
34 mm
38 mm
75 mm
70 mm
50 mm
DESIGNED BY:
Guillermo de Juan Grau
DATE:
2015
CHECKED BY:
Daniel Díaz
DATE:
22/09/2015
SIZE
A3
SCALE
0.75
WEIGHT (kg)
DRAWING NUMBER
SHEET
Middle body
Smart pillbox
This drawing is our property; it can't be reproduced or communicated without our written agreement.


15 mm
40 mm
12 mm
8 mm
12 mm
40 mm
DESIGNED BY:
Guillermo de Juan Grau
DATE:
2015
CHECKED BY:
Daniel Díaz
DATE:
22/09/2015
Pill chamber
Smart pillbox
SIZE
A3
SCALE
1.5
WEIGHT (kg)
DRAWING NUMBER
SHEET
This drawing is our property; it can't be reproduced or communicated without our written agreement.



DESIGNED BY:
Guillermo de Juan Grau
DATE:
2015
CHECKED BY:
Daniel Díaz
DATE:
22/09/2015

Pill hatch

Smart pillbox

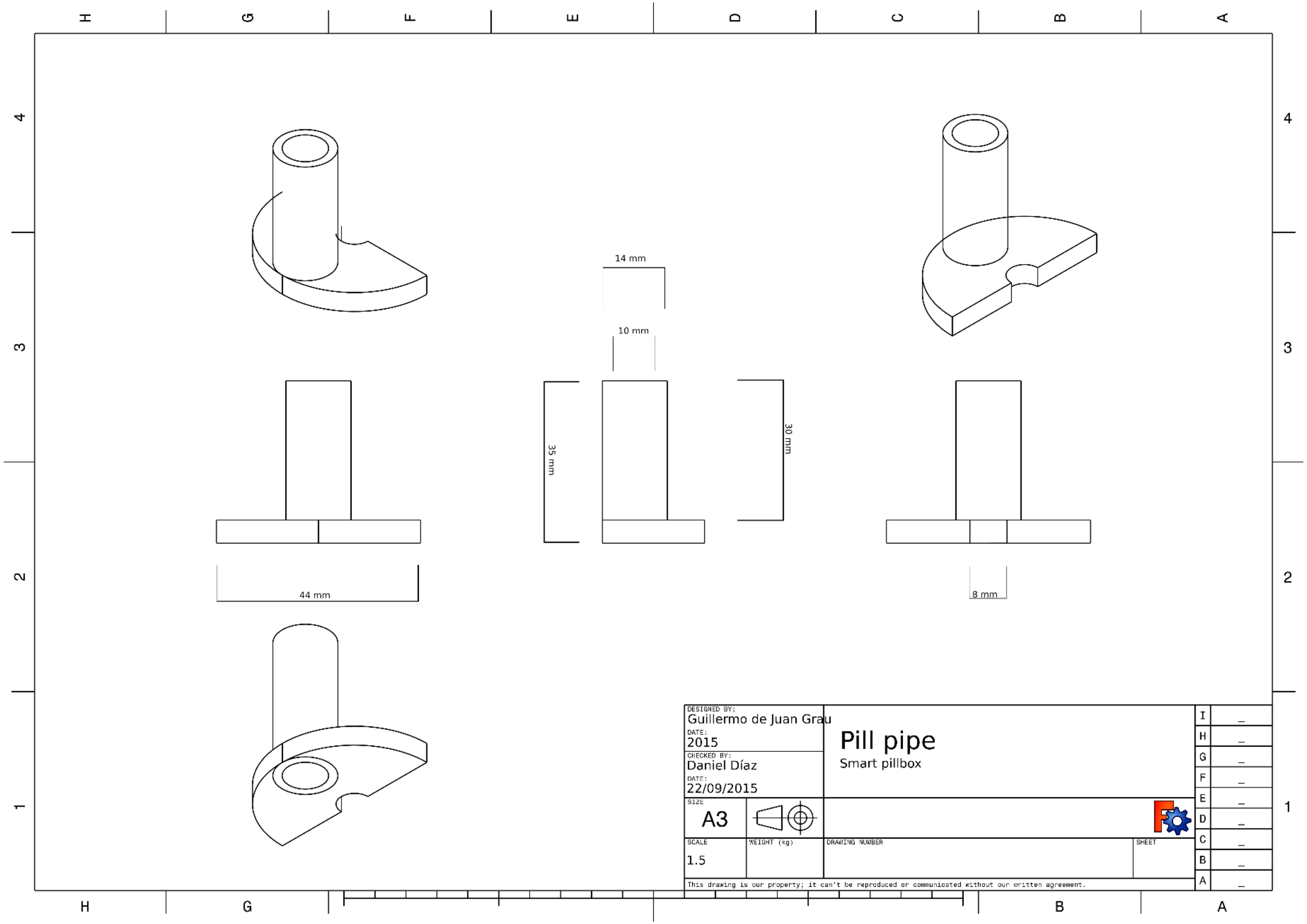
SIZE: A3

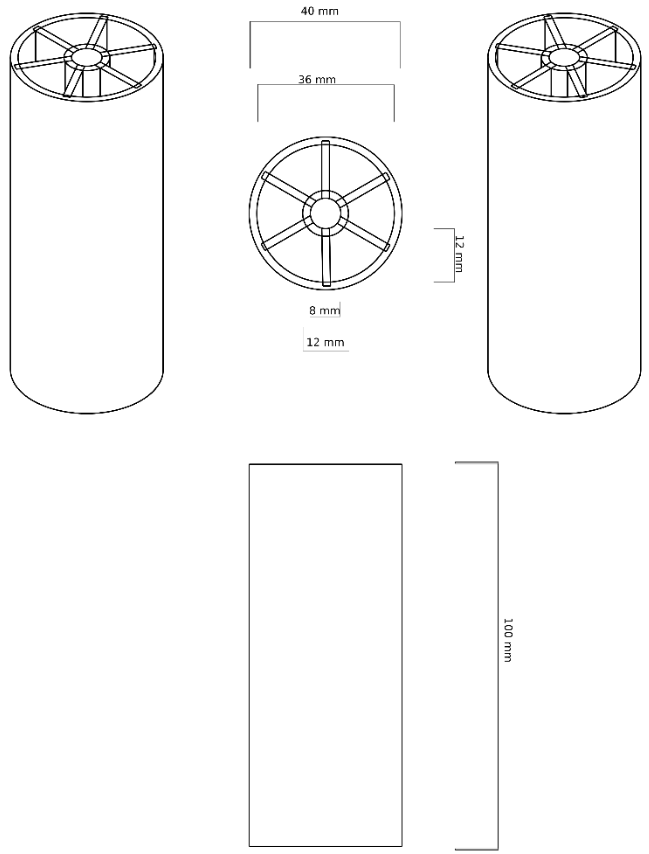
SCALE: 2

WEIGHT (kg):

DRAWING NUMBER:

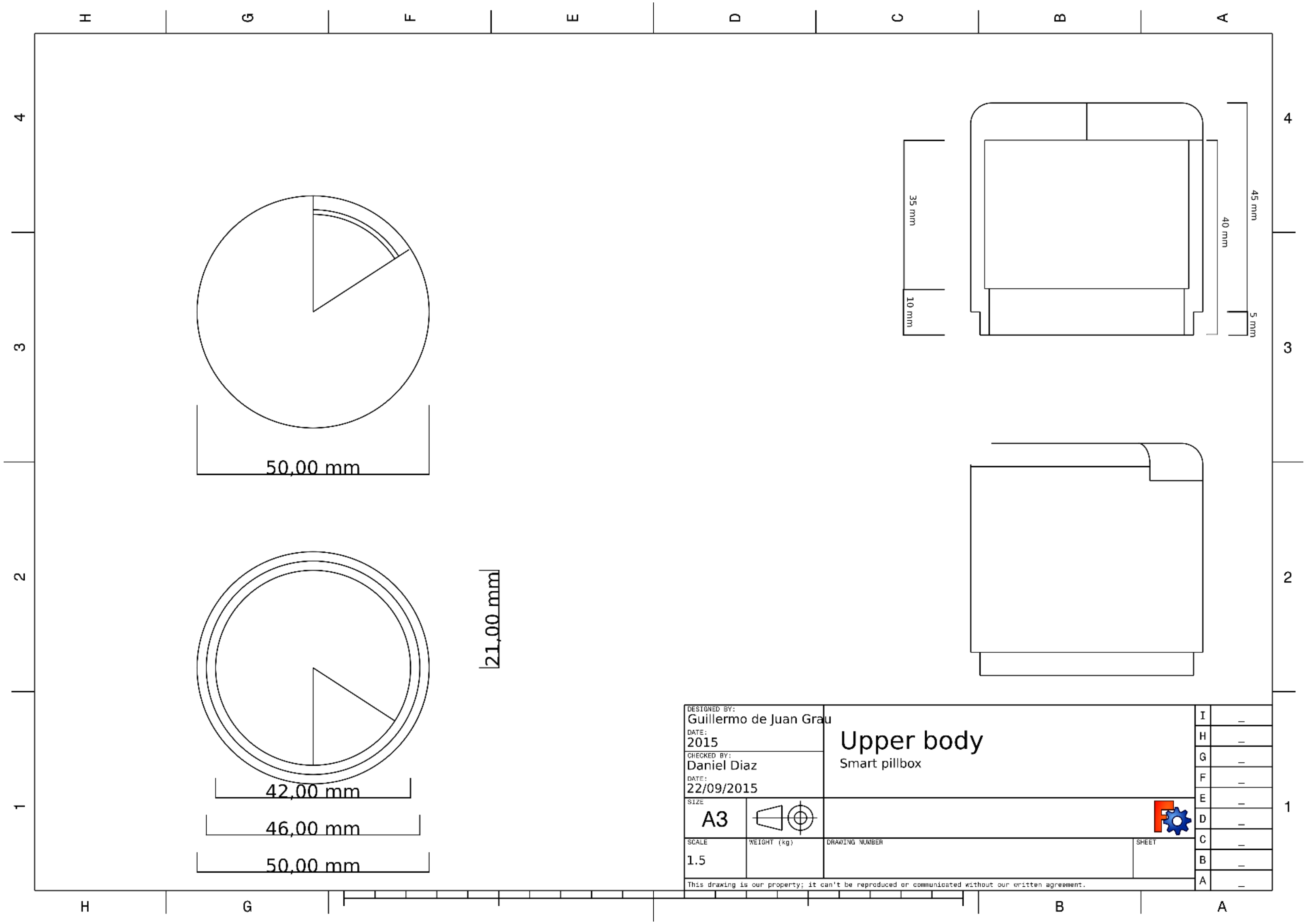
SHEET:


14 mm
10 mm
35 mm
30 mm
44 mm
8 mm
DESIGNED BY:
Guillermo de Juan Grau
DATE:
2015
CHECKED BY:
Daniel Díaz
DATE:
22/09/2015
Pill pipe
Smart pillbox
SIZE
A3
SCALE
1.5
WEIGHT (kg)
DRAWING NUMBER
SHEET
This drawing is our property; it can't be reproduced or communicated without our written agreement.



DESIGNED BY: Guillermo de Juan Grau DATE: 2015 CHECKED BY: Daniel Díaz DATE: 22/09/2015	**Pill storage** Smart pillbox
SIZE: A3	
SCALE: 1	WEIGHT (kg):
DRAWING NUMBER:	SHEET:

I	_
H	_
G	_
F	_
E	_
D	_
C	_
B	_
A	_



50,00 mm
21,00 mm
42,00 mm
46,00 mm
50,00 mm
35 mm
10 mm
40 mm
45 mm
5 mm
DESIGNED BY:
Guillermo de Juan Grau
DATE:
2015
CHECKED BY:
Daniel Diaz
DATE:
22/09/2015
Upper body
Smart pillbox
SIZE
A3
SCALE
1.5
WEIGHT (kg)
DRAWING NUMBER
SHEET
This drawing is our property; it can't be reproduced or communicated without our written agreement.