Treball de Fi de Grau

Grau en Enginyeria en Tecnologies Industrials

PERFORMANCE IMPROVEMENT OF A TRANSRADIAL MYOELECTRIC PROSTHESIS

Autor: Pau Adan Simó
Directora: Daniela Tost Pardell
Convocatòria: Abril 2022

ETSEIB

Escola Tècnica Superior
d'Enginyeria Industrial de Barcelona

UPC

Abstract

ARM2u is a UPC team of both graduate and undergraduate Engineering students that strives to develop a transradial myoelectric prosthesis. The team aims to participate in the powered arm prosthetic race of CYBATHLON, held in Zürich, competing against designs from all over the world.

This project compares the prosthesis showcased by the team in December 2021, Andromeda, with the more relevant models of the market, comparing their performance and identifying key improvement areas. This project also reviews the state of Andromeda and the improvements researched and/or implemented since January by ARM2u. It also finds what are the main sources of complaint and rejection among prosthesis users.

Solutions for two of these problems, non-intuitive control of the prosthesis and the lack of tactile feedback, are proposed for Andromeda. A new mode to switch mode is designed and implemented, using voice commands transmitted through a phone app. A feedback system that transmits mechanical feedback to the residual limb proportional to the force exerted by the prosthesis is also designed. Both of these improvements are programmed using a new microcontroller intended to be implemented in future iterations of Andromeda.

Afterwards, testing of those proposals is conducted, and the impact that their implementation could have is briefly discussed.

Summary

Figure index

Table index

Code index

Glossary

- **Transradial prosthesis:** A prosthesis designed to replace the function of missing anatomical segment(s) from the elbow to (and including) the hand.
- **Myoelectric prosthesis:** A prosthesis that is controlled with the electrical tension generated by muscle contractions.
- **Body powered prosthesis:** A prosthesis that is controlled mechanically using other parts of the user body.
- **ARM2u:** ARM2u is a university team founded in 2018 based in Barcelona, currently located at the ETSEIB Campus of the Polytechnical University of Catalonia (U.P.C.). The team consists of a multidisciplinary group of both graduate and undergraduate Engineering students. The current objective of the team is to develop a transradial myoelectric prosthesis to participate in the CYBATHLON competition.
- **CYBATHLON:** The CYBATHLON is a unique championship in which people with physical disabilities compete against each other to complete everyday tasks using state-of-the-art technical assistance systems.
- **EMG:** Electromyography, a technique developed to record muscle activity. Most myoelectric prosthesis are controlled through surface EMG sensors attached to the body of the pilot. Multiple electrodes are needed, because EMGs read the potential difference between two separate electrodes.
- **PHM:** PHM (or Phantom Hand Map) are specific areas on the residual arm where amputees perceive touch as if it was applicated on the missing hand
- **Haptic:** Related to the sense of touch. Haptic technology is the use of tactile sensations to stimulate the sense of touch in a user experience.
- **MSS:** Mechanical surface stimulation, the transmission of mechanical stimulus to the skin, like vibration or pressure.
- **ESS:** Electrical surface stimulation, the transmission of electrical stimulus to the skin.
- **GPIO:** General purpose input/output pins, they can be used to send or receive electrical signals. Signals from these pins usually have only two values: High or Low.
- **PWM:** Pulse width modulation (PWM) is a technique to control analog devices with digital outputs modulating the intensity of the signal.
- **Analog input pins:** Digital input pins measure a range of voltages on a pin, contrary to digital pins, which typically only sense if there is or isn’t voltage in the pin.

1.Introduction

1.1.Origin of the problem

A prosthesis is an artificial device that replaces a missing body part, which may be lost through trauma, disease, or a condition present at birth. The earliest example of a prosthetic hand dates back to 200 BC: Pliny the Elder recorded how Marcus Sergius, a Roman general, received a prosthesis that allowed him to return to battle after losing his hand in the Second Punic War. Since then, prosthetic hands have evolved into revolutionary devices that are growing steadily closer to real hands. Unfortunately, there is still a long way to go: upper limb prosthetics are failing to live up to the patient's expectations, which is shown in their rejection ratio: an average of 30% of myoelectric arm prosthesis users give up the use of their artificial limbs, with peaks of up to 50% of the users [1]. This represents an unacceptable waste of resources, as myoelectric arms price range goes from 20.000$ to 100.000$. [2]

The most important problem that users face is the lackluster performance of their artificial limbs compared to their original arms, which are often unsuited for comfortable daily use. Other key aspects, such as weight or precision, are also considered sources of rejection [3].

1.1.1 ARM2u

ARM2uis a university team founded in 2018 based in Barcelona, currently located at the ETSEIB Campus of the Polytechnical University of Catalonia (UPC). The team consists of a multidisciplinary group of both graduate and undergraduate Engineering students who share the ambition to push the boundaries of the prosthetic industry. With the original intention to participate in the Cybathlon 2020 competition, the team's current focus lies on the design and creation of a transradial (replaces the missing part below the elbow) myoelectric (controlled by the electrical signals generated by our own muscles) arm prosthesis which allows its user to perform basic daily tasks, named Andromeda.

ARM2u's mission is to improve the quality of life of people in need of assistive technologies. One of the team's main milestones is to participate in the powered arm prosthetic race of CYBATHLON competition organized by the ETH Zürich with a self developed myoelectric prosthetic arm. The spirit of the team goes beyond building a prototype to specifically overcome the challenges presented in the competition, but to develop a model that responds to the true needs in the daily lives of people in need of assistive technologies. Similarly, Arm2U’s vision of the future is the consolidation of a team which can create a positive social impact by supporting the diversity and social inclusion of users all around the globe. That is to say, ARM2U intends to have continuity over time and to establish itself as a benchmark for students who want to take part in a real and challenging project with social implications.[4]

1.1.2. Personal motivation

As a member of ARM2u, one of my goals is to use the knowledge obtained during my formative years to help people in need. Using the power of technology to recreate the human body is a fascinating and challenging experience, and using knowledge to better the lives of others is the main goal of science.

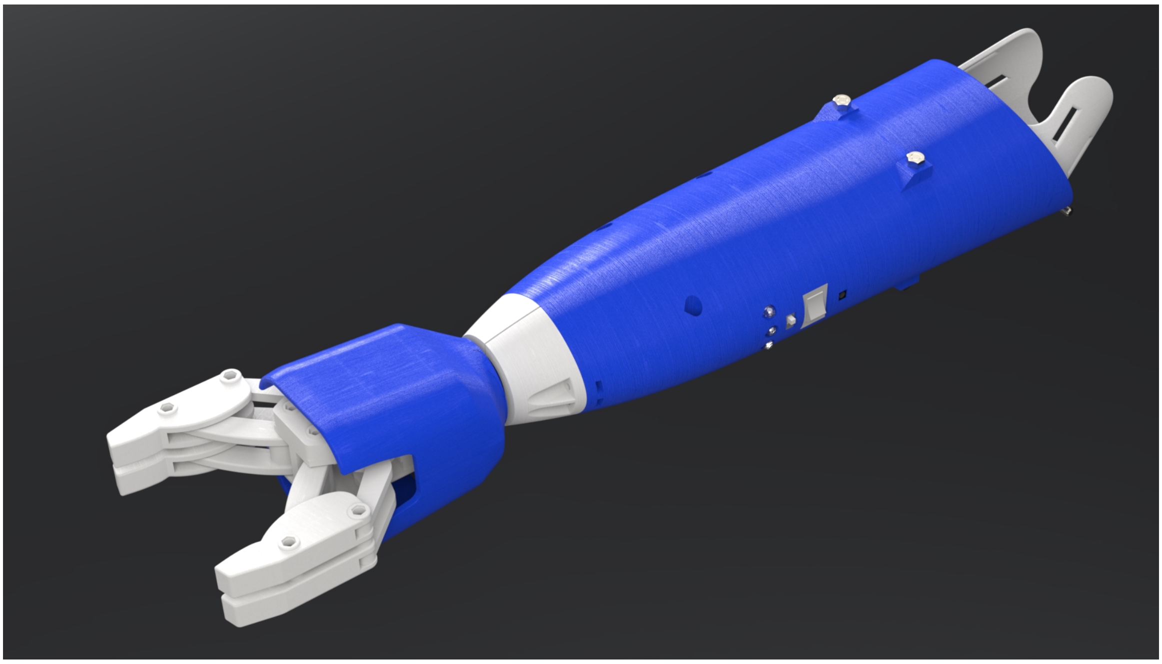


Figure 1.1: Render of a virtual rendition of Andromeda. Source: ARM2u.

1.2.Goals of the project

1.2.1. General goals

The main goal of this project is to contribute to the improvement of the functionalities of the Andromeda prototype designed by ARM2u, providing it with an innovative option to switch modes and a haptic (tactile) feedback system, in order to minimize its rejection risk.

1.2.2. Specific goals

The first goal of the project is to research the field of myoelectric prosthetics in order to discover which are the demands and complaints of prosthetic users. This will also serve to have a better understanding of where Andromeda stands in the competitive field.

The second goal is to analyze ARM2u's prosthesis to ascertain its current capabilities and which potential upgrades would it benefit from. Since continuous improvements in areas such as weight, speed or gripping strength is already being carried out by the departments of the team, this project will focus instead on the research and development of new options for the arm.

The third goal of the project is to improve the mechanism to switch between the different grip modes of the hand, providing an innovative option that is comfortable to the user.

Last but not least, the fourth goal is to increase the accuracy of the hand through haptic (tactile) feedback.

1.3. Scope of the project

The project is designed with its implementation by ARM2u in mind, and therefore the proposed options will take into account the resources, budget and knowledge of the team and its prosthesis by December 2021. For this reason, the solutions will be developed specifically for a transradial myoelectric prosthesis, and options such as invasive surgery or brain-computer interfaces won't be discussed.

2.Problem Analysis

2.1.State of the art

2.1.1. Basis of myoelectric prostheses

A transradial myoelectric prosthesis replicates the functionality of a missing arm from elbow to fingers. In essence, a transradial myoelectric prosthesis consists of three core components: the sensors, the microprocessor and the actuators. The myoelectric sensors are usually placed in the remnants of the user's arm, where they measure the electrical activity of the muscles in response to nerve stimulation. The information is sent to the microcontroller, which in turn sends a signal to the actuators (usually, servo motors), moving them according to the received signal.

Another key aspect of the arm is the socket, which serves as the union between the prosthesis and the residual limb. In recent years, methods to transmit sensory feedback from the hand to the user have become significantly more common, and are beginning to be expected on commercial devices.

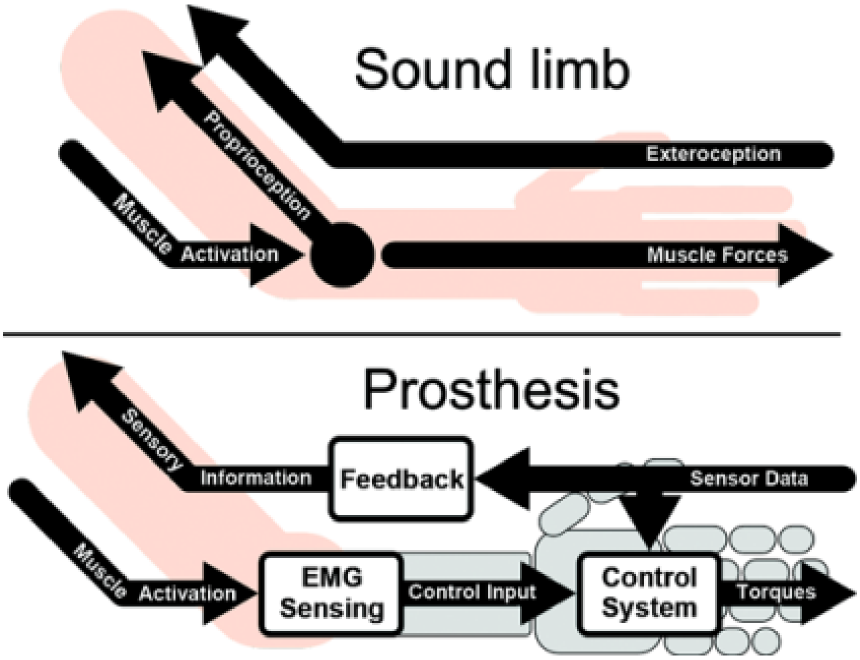


Figure 2.1: Signal flow in a human forearm (above) and in a myoelectric forearm prosthesis (below). Source: [5]

Besides those three elements, a prosthesis usually includes security elements to ensure the safety of the bearer. A power supply is also needed, and some visual cue of the state of the prosthesis (usually, a LED) can be incorporated.

Myoelectric control detects the electrical potential produced during muscle contractions, and consists of four main stages: signal collection, signal amplification, signal filtering and analog-to-digital conversion [6]. Although the most commonly used method of EMGs detections is surface electrodes, those signals can also be recorded by inserting electrodes in the muscle tissue, but that is an invasive method mostly used in clinical analysis. [7] The recorded signals are used to move the hand or switch between modes, although auxiliary control methods are usually also available.

2.1.2. State of the market

Since ARM2u is developing a transradial myoelectric prosthesis, the first step is to take a deeper look into the most representative available or soon to be available options and their capabilities. A literature search was conducted covering the periods 2003-2022 through numerous databases, including Pubmed, SAGE journals, Researchgate and ACRM. These sources were used to complement the information already available in the ARM2u database, and were complemented with data obtained through the web search engine Google Scholar.

Some prostheses were frequently brought up in different articles, such as the Michelangelo hand, the Vincent hand, the TASKA hand or the i-Limb. A deeper research in these models was conducted, using the information available in the manufacturer’s website or datasheets and in the website of Bionics for Everyone [8], which is presented in the table below.

The following properties were included in said table:

- **Hand weight:** It’s imperative to know which weight is commercially acceptable for a myoelectric prosthesis. It’s important to remark that this information refers exclusively to the hands of the prosthetic, since most of these models offer different socket options for below or above the elbow amputations.
- **Grip patterns:** The number of available configurations that the actuators can take in order to replicate hand movements, such as clenching a fist or pointing a finger. In some cases the models offered a virtually limitless number of patterns, but the storage capacity of the device limited the number of modes that could be available at the same time.

- **Lift capacity:** The maximum weight that the hand can carry effectively. This is one of the key indicators of the mechanical performance of the hand.
- **Maximum grip force:** The maximum amount of force that the hand can exert. Similar to lift capacity, this is a key indicator of mechanical performance.
- **Sensory feedback:** Tactile information has been a rising trend for electric-powered prosthesis in recent years, since it was highly demanded by patients. Sensory feedback doesn't only increase the precision of the hand, allowing for more delicate tasks: it also increases the sense of ownership of the prosthesis, reducing its rejection rate.
- **Battery life:** Autonomy is one of the most important factors of a prosthetic device, and therefore it is necessary to know the average life expectancy that Andromeda is expected to provide.
- **Average price:** The average cost of the hand.

Name	Hand weight (Kg)	Grip patterns	Lift capacity (Kg)	Maximum grip force (N)	Sensory feedback (Yes/No)	Battery life (hours)	Average price (€)
Michelangelo	0'51	7	20	70	No	20	60.000
i-limb	0'5	12-36	90	136	No	24	53.000
Bebionic hand	0'525	14	45	140	No	24	32.130
Vincent Evolution	0'4 for the XS option	15	-	-	Yes	-	32.130
TASKA hand	0'640	23	20	-	No	400 grip actions	50.500
Psyonic ability hand	0'47	32	23	-	Yes	24	23.000
Hero arm	0'34	6	8	-	No	24	13.800

MeHand	-	27 grasps + 14 gestures	30	140	Only in the A version	-	32.176 for the A version, 23.000 for the B version
Nexus Hand	0’592	14	32	80	No	-	23.000
Atom Touch *(in development)*	-	Infinite through thought control	20	300	Yes, including heat sensor	-	70.000

Table 2.1: Comparison of the capabilities and prices of different commercially available hand prostheses. The information that is not available is left as “-”. Source: [8]

We can infer from the above table some of the requirements that Andromeda should fulfill in order to be considered a competitive prosthesis. First of all, we can observe that the weight of the hand shouldn’t exceed 600 grams, which is similar to the average weight of an adult male hand (460 grams) and fits within the product requirements of Andromeda (420 grams). We can also appreciate that the gripping force is never below 70 Newtons, which is the typical grip strength required for most daily activities [9]. Battery is supposed to last around 24 hours, and users are expected to recharge it at night. Additionally, according to most fabricants, the batteries of the prostheses should be changed once a year. Last but not least, despite still being uncommon, tactile feedback is more frequent with each generation of prostheses, since that functionality has been highly demanded by the users for years.

It is important to note that most of the disabled population live in developing countries where advanced prostheses are often impossible to obtain due to the cost and the lack of medical care and technical resources available. In those cases, rudimentary and body-powered solutions are sought after, if they are available, although prostheses created through 3D printing are being considered as affordable solutions for these situations. The incoming Venus Arm developed by eBionics uses its price as a selling point, stating that the cost for the arm will be around 240€ [10]. Unfortunately, this number is tricky, since the user is the one who has to 3D print or buy the pieces of the prosthesis and then assemble the model. Since it is a still developing project, there is no information about the performance of the Venus Arm in daily use, and therefore we can’t ascertain if it satisfies the needs of its users. Another example of low cost prostheses are the ones created by the bioengineering student David Aguilar, who built his own prosthesis when he was 9 years old using Lego sets. He has designed up to 11 different prosthetics for himself, as well as two customized ones for Beknur, the 8 year old son of the consul of Kazakhstan in France. Andromeda could become a project of similar characteristics if it becomes commercially available, although there is still a long way to go.

In terms of control, most prostheses use a direct myoelectric control system, which means that myoelectric sensors (electrodes) are directly placed against the skin regions immediately above the muscle tissue of the residual limb to detect muscle movements. Of the 19 arm prosthesis reviewed on Bionics for Everyone, only five of them (including the Atom Touch from Atom Limbs, which is still a project in development that aims to implement direct thought control) used alternative options.

The Psyonic Ability Hand can be controlled through third party EMGs, linear transducers (which measure displacement in a single direction) or force-sensitive resistors (which read the muscle bulge instead of its electrical signal). These sensors are also used by the Grippy bionic hand instead of EMGs. The LUKE Arm by Mobius Bionics also offers multiple control options to their users besides EMG, such as IMU (Inertial Measurement Units), which allow the user to control their prosthesis with foot movements, reading its tilt like if it was a joystick [11]. Last but not least, the TrueLimb by Unlimited Tomorrow uses a customized sensor system that detects changes in muscle topography.

2.1.3. The problems of prosthetic arms

Despite the capabilities of the already available arms being sufficient in theory, a large number of users eventually abandon their prosthetic devices, preferring to make up for their missing limbs in other ways. The information gathered shows that the trend is shared by both classical body powered prostheses and more advanced electrical ones [12].This problem arises in part due to the lack of adequate testing and researching: Many prostheses easily pass their functionality tests, yet are still rejected by their pilots due to their lack of practicality in their daily use [13]. The main abandonment causes are insufficient functionality and non-intuitive controls, although other inconveniences such as slow reaction time were also mentioned [14]. New options, such as tactile feedback, are also highly demanded.

One challenge in upper-limb prosthetic design is the lack of available and clinically relevant outcome measures. For these reasons, common practices to assess upper-limb prosthetic solutions highlight the need to use multiple outcomes and when not possible, at the very least acknowledge that further investigations would be needed [15]. The tests performed to ascertain the functionality of the prosthesis are often made by able-bodied pilots, and said tests are usually designed with pure mechanical outputs in mind, such as grip speed or strength, while the clinical rehabilitation project takes a more holistic approach. In his 2011 article "*Myoelectrical forearm prostheses: state of the art from user-centered perspective*" Bart Peerdeman suggested that the approach to develop functional requirements should be based on the activities users will perform. The same research stated that the focus should be on validating EMG-sensing results, improving simultaneous control of wrist movements and grasps, deriving optimal parameters for force and position feedback, and taking into account the psychophysical aspects of feedback, such as intensity perception [5].

The user's problems are further worsened by the absence of standardized training for prosthetic limbs. In October 2021 a literature search for training methods for myoelectrical multigrip prosthetic hands covering the 2007-2020 period produced 1528 different peer-reviewed results, which were reduced to only 92 (88 articles and 4 users manuals) after screening the titles and abstracts, and finally trimmed down to 9 [16]. The elimination criteria for the articles in this last phase included:

- No prosthesis use, only training with myo bands (40 articles)
- No myoelectric training described (23 articles)
- No multigrip prosthesis (12 articles)
- Not really peer-reviewed articles (4 articles)
- Foot-controlled prosthesis (2 articles)
- Prototypes not available on the market (2 articles)

The fact that in a period of 13 years only 5 articles and 4 user manuals that could fit the inclusion criteria were published speaks volumes about the lack of available resources that myoelectrical multigrip hand prostheses lack, which leads to inefficient and often insufficient training. Unfortunately Andromeda also falls into said category, which means that ARM2u can have a hard time training their pilots in the future unless new valid sources are published.

It is also worth mentioning that despite the wide array of grasping modes available, most users prefer to consistently use only 3 or 4 of them for all their daily activities, instead of deploying their prostheses to their full potential. The training necessary to skillfully control their hands leaves most users trying to work with what functionalities they have already learned to use.

2.1.4. Haptic technology

Haptic technology uses tactile stimulation to simulate the sense of touch. The ability to interact with the environment and perceive the amount of gripping force exerted with the artificial limb without relying on other senses is essential to improve the life quality of prosthesis users.

The main challenge is to give said feedback to the user in a non-intrusive method that doesn't rely on visual cues. The most viable options are electrical surface stimulation (ESS) and mechanical surface stimulation (MSS), usually through pressure or vibration [17]. Nerve stimulation is also considered a promising field if more intrusive ways are not discarded, but it is beyond the current scope of ARM2u. Atom Touch, mentioned in 2.1, is developing a new non-invasive control system capable of transmitting contact, force, position, speed and even temperature feedback, although said technology hasn't been shown to the public yet. Studies on audible feedback have been conducted by various institutions, but they are deemed inefficient due to the interference of ambient noise and the problems they may create if the user needs to be quiet.

Both MSS and ESS systems are considered valid options within the scope of Andromeda, but ESS presents a potential interference problem between the myoelectrical control system and the feedback signal. Moreover, the detection threshold and the pain threshold signal currents are only 1 mA apart. Therefore, in order to avoid the risk of discomfort, a MSS system will be proposed.

Transmission of tactile feedback in transradial amputations is usually performed directly into the remaining arm, applying the stimulus directly into the phantom hand map, or PHM. The PHM are specific areas on the residual arm where amputees perceive touch as if it was applied on the missing hand, allowing stimulation of the fingers of the phantom hand through the transmission of information from sensors allocated in the prosthetic fingers to specific areas of the skin [18]. This is possible due to rearrangements of the cortical circuits occurring after the first hours after amputations, the location of which can vary between patients. Therefore, each amputee needs to identify the areas of their residual limbs correlated to the fingers of the phantom hand.

In spite of its lack of “fairness”, as it depends on circumstances which the individual has no control over, this method still allows for an increase of performance in patients who only have an incomplete phantom hand (or, in other words, who can only recognize one or two fingers of their PHM) [19]. Despite most of the studies performed being centered around prosthetic users who experienced the loss of an upper body extremity, the phenomenon of phantom limbs can also be experienced by people born without said limbs due to a congenital disease [20], and therefore MSS on the PHM can still be used to transmit sensations to those users. The main drawbacks of this method are that the user usually needs to train in order to have an accurate idea of where the PHM is, and that the connections of the motors must be performed into different points for each patient, but due to the nature of prosthetic work an important degree of customization and direct work with the patient is expected, and therefore these are not considered serious setbacks.

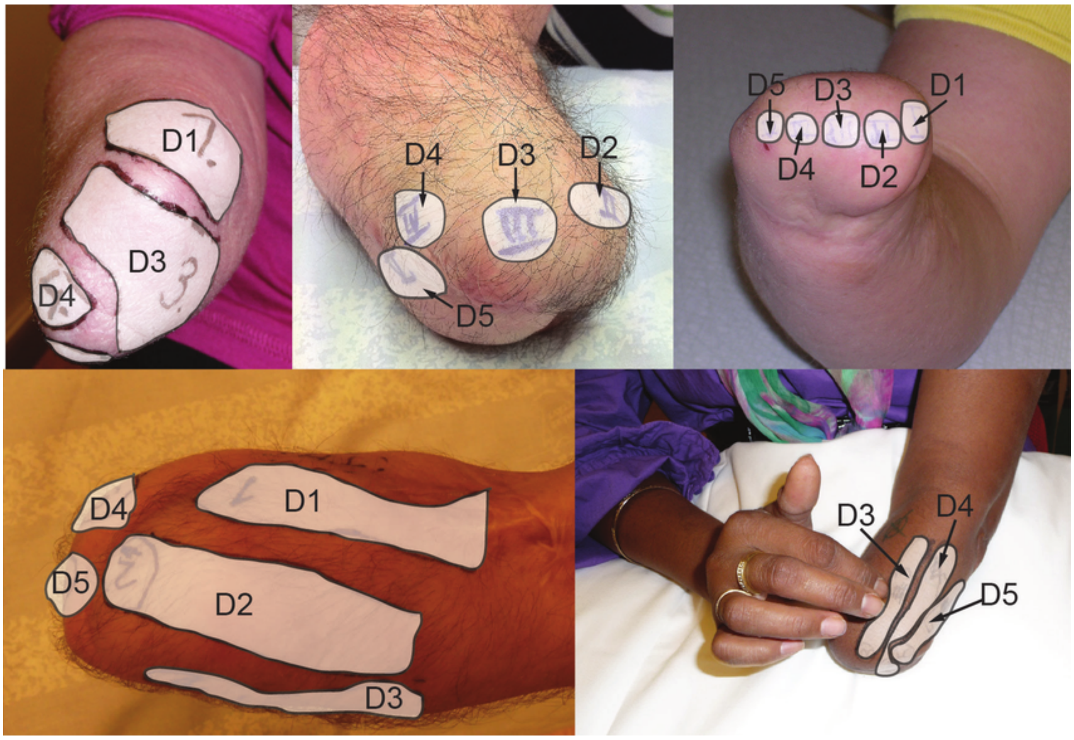


Figure 2.2: Examples of phantom hand maps, with each area corresponding to one of the digits of the original hand (D). Some of them only present a partial phantom hand, like the bottom right example. Source: [21]

2.2.Analysis of the Andromeda prototype

2.2.1. Introduction to Andromeda

In December 2021 ARM2u showcased Andromeda, the first myoelectric arm prosthesis developed by an UPC team, developed using 3D printing technologies. The presentation was held at ETSEIB in front of both students and professionals. The prosthesis was operated through a combination of EMG control and manual regulation by members of the team: a switch on the forearm turned the prosthesi on and off, while another situated next to it allowed the pilot to alternate between the grip and pronosupination modes. The EMG sensors transmitted the muscular contractions of the user to the microcontroller, causing the hand to close if it was on grip mode, or to rotate if it was on pronosupination mode. If no signal was received during an interval of 5 seconds, the hand automatically returned to its original position.

Figure 2.3: Presentation of Andromeda at ETSEIB. Source: ARM2u

2.2.2. Analysis of the prosthesis

Despite the foundations being set, the performance of Andromeda couldn’t meet the goals set by the team. Critical divergences from the requirements in its intended strength, control and weight showcased the need of reconsidering the design of the hand and its actuators, which has been the main goal of the team since the presentation. Through experimental testing, the following properties were found:

Total weight (Kg)	Grip patterns	Lift capacity (Kg)	Maximum grip force (N)	Sensory feedback	Battery life (hours)
4	2	2	18	No	8

Table 2.2: Specifics of Andromeda in December 2021. Source: self.

As we can see, the mechanical properties of the hand were insufficient for the daily needs of the user. Although the number of grip patterns and the existence of sensory feedback could be ascertained at first glance, the rest of the properties were experimentally found.

The weight was calculated using a common electrical scale. Although the measurement instrument lacked precision, the obtained value, 4 Kg, was three times bigger than the weight maximum defined in the product requirements (1.38 Kg), and therefore the measurement error was considered negligible.

Figure 2.4: Lift capacity testing of Andromeda at ARM2u's office, ETSEIB. Source: self.

One critical issue that Andromeda had was that the pincer was almost unable to grab objects of certain geometries. This problem stemmed from the materials used for the 3D printing of the model (PLA, or polylactic acid), which had an extremely low drag. Therefore, the lifting capacity reflected the maximum weight that an object carried by Andromeda could have without slipping from the gripper. Physical experimentation with similar items of different weights (all of them cylindrical, in order to minimize inaccuracies caused by the geometry of the pieces) was conducted. For a weight to be considered “lifted”, the hand needed to grasp and hold it in the air for at least 5 seconds at least 6 out of 10 tries. The heaviest object that passed this test weighed 2 Kg.

The experience using the hand already showcased that it didn’t have the strength to perform most daily activities, but an approximate idea of the force exerted was needed nevertheless. A dynamometer wasn’t available when the test was conducted, therefore the measurements were taken interpolating from the results showcased in the electrical scale when the gripper squeezed it. Although this method lacked precision, it offered a rough estimate of Andromeda’s strength, which was around 18 Newtons.

For the battery, a maximum voltage discharge essay was performed with the help of Members of the team. The data was recorded and monitored in 10 minute intervals until the battery entered critical failure. The battery lasted approximately 15 minutes longer than anticipated in the theoretical calculations. Since the battery passed the test for maximum usage, it was considered that it would also work within the specifications when subjected to normal performance.

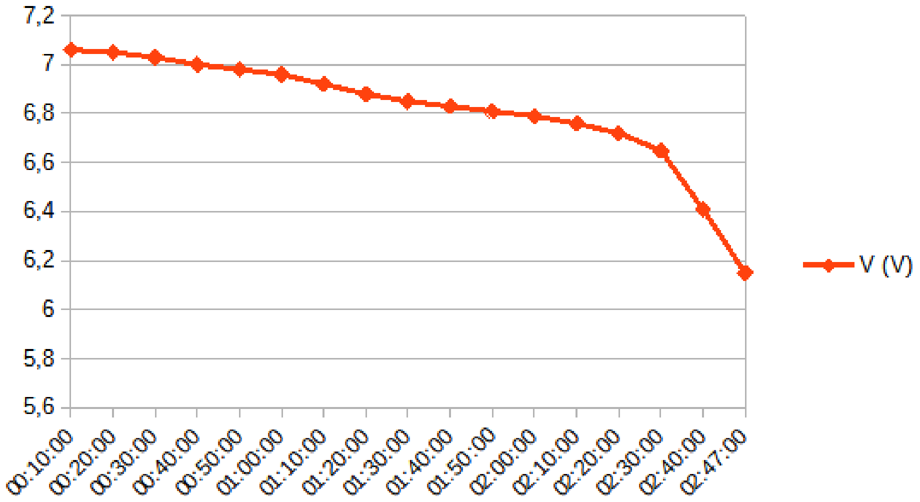


Figure 2.5: Evolution of the voltage through the test. Source: ARM2u.

The Andromeda showcased in December used a gripper as a simpler alternative to a human hand, a solution that can also be found in more basic prosthetic devices, as advanced grip modes were not the highest priority of the team at that moment. Said hand was controlled through two servo motors, each one assigned to one of the hand modes.

A servo motor is an actuator which allows for precise angle and speed control through PWM (pulse width modulation). There are two kinds of servo motors: position and rotation servos. Position servos can rotate until they reach a certain angle (usually, 180º), while rotation servos don't have a limitation to their range of motion, being able to rotate indefinitely as long as they are connected to a power source. Servo motors have 3 wire connections: the control input, which is connected to the output pin that controls the servo (usually, a white or orange wire), the power wire, usually red, and connected to the power supply (usually red) and the ground connection, which is usually brown or black. Most servo motors (including the ones used by Andromeda) use a 5-6V DC power supply. When coding, the input sent to a position servo determines its angle, while the input sent to a rotation servo determines its speed.

Andromeda's gripper mode was controlled by a position servo, while its pronosupination was controlled by a rotation servo. The current project uses a switch to change between the two available modes, making the process uncomfortable for the user, who needs to reach for the prosthesis with their other arm each time they want to switch modes.

The control of the arm was also deemed lackluster, with no clear correlation in various cases between the intended movements of the user and the actions performed by Andromeda. After different members of the team tried to use the prototype, it was noticed that the grade of precision greatly diverged between users, indicating that this issue was partially caused by lack of training, although even the users with the best control still couldn't manipulate objects with precision using the gripper. The analysis of the EMG sensors by the electronics team after the presentation showcased damage due to welding errors and wear and tear, which left the connections vulnerable, causing severe ambient noise (which was deemed as the cause of the irregularities experienced controlling the arm).

Finally, ARM2u currently uses the members of the team itself (all of them able-bodied pilots) in order to assess the functionality of the prosthesis in its early stages. In order to perform the tasks the members use electrodes to transmit their muscular signals to the prosthesis. The team also has a pilot who performs the final tests in person. The testing process is both time-consuming and tedious, with many tests consisting of repetitive tasks, some of which can take up to 8 hours.

2.2.3. Improvements

After the presentation, the Mechanics department focused most of its resources in increasing the drag of the gripper. A silicone cover on the “fingertips” of the pincer was implemented, with outstanding results. Simultaneously, a more anatomically correct hand was designed. After numerous redesigns, the mechanics department is developing a humanoid hand made with flexible PLA, which will be controlled through tensors and servo motors to allow precise and independent control of each finger.

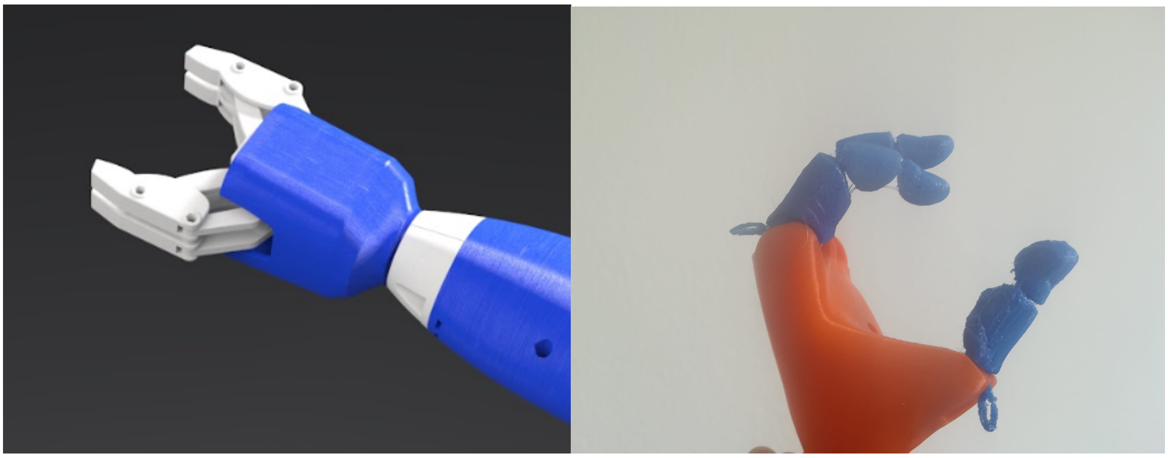


Figure 2.6: Comparison between a render of the gripper used in December (left) and the future hand (right). Sources: ARM2u (left) and self (right)

Since the new hand is still under development, as stated in scope, this project will use the hand showcased in December as reference. The team strives to develop a multi grip hand, which will allow the user to choose between different modes of movement, picking the one that better suits their needs. The current objective is to have the following modes: cylindrical grasp, hook, pronosupination, lateral, tip (two fingers), palmar (three fingers) and point (one finger, designed to allow the use of touch devices and keyboard writing).

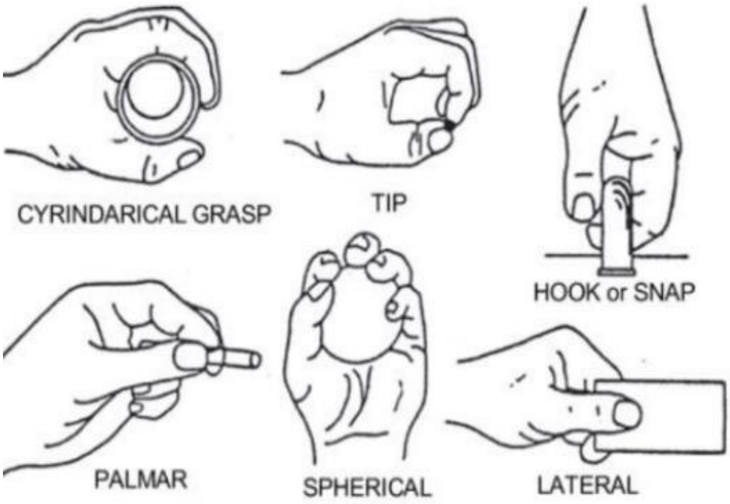
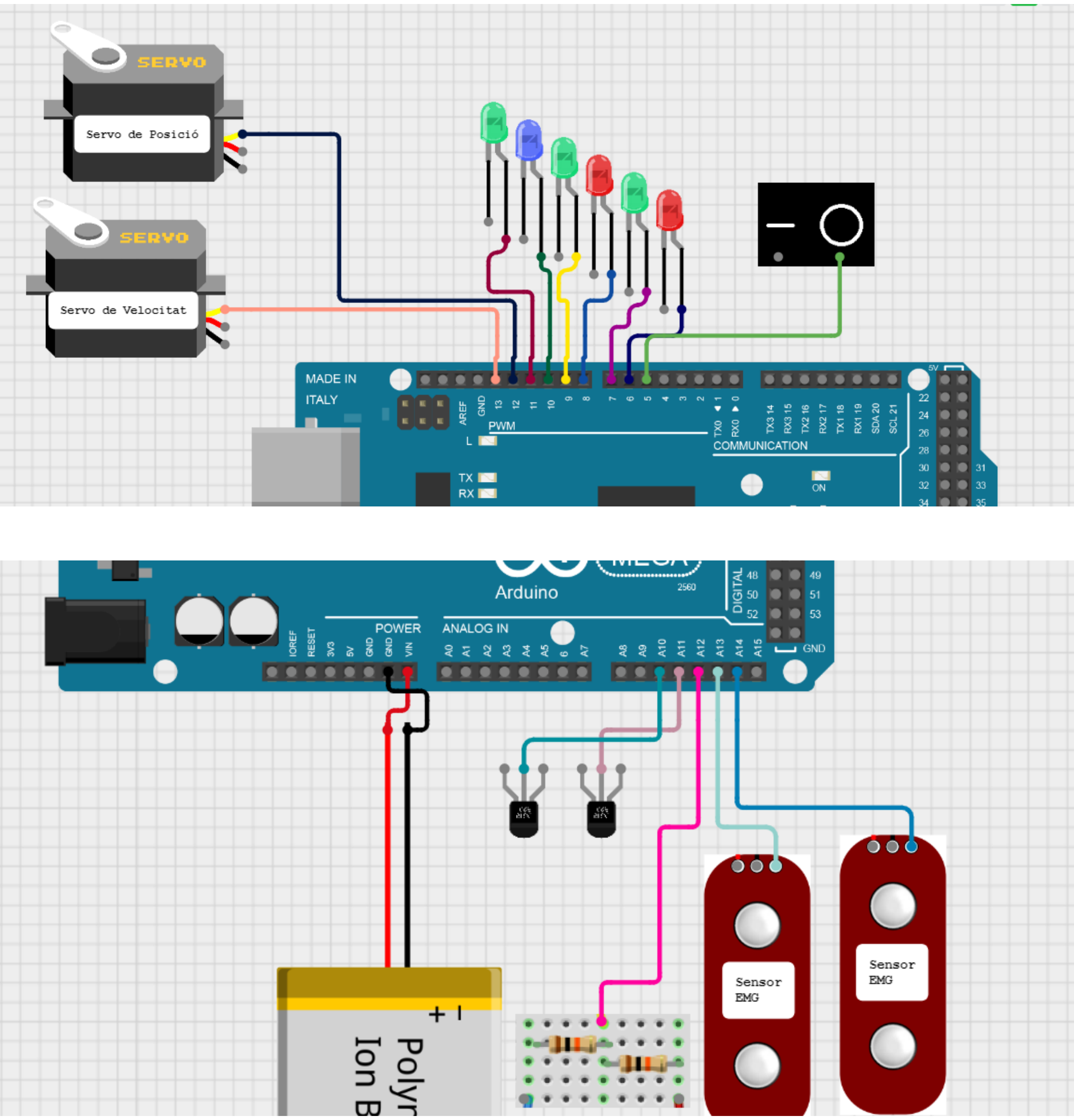


Figure 2.7: Six different types of grasp. Source: ARM2u

In order to correct the mistakes of the control system, a new set of EMGs with additional safety elements will be used in the next iterations, including a connection system that doesn't require welding.

The electrical components of the prosthesis are also being revised, with devices being incorporated or excluded from the connection map. One of the main examples is the drastic reduction of the LEDs in the prosthesis, which will be taken down from six to one. These devices were originally intended to serve as visual indicators of the status of Andromeda and the potential risks it could suffer (like overheat or electrical malfunction), but their physical implementation on the device was not practical. Therefore, for future iterations only a single LED will remain, which will simply indicate if the prosthesis is active with a green light or if it has a technical problem with a red one. The implementation of a LCD screen to give the user detailed information of the state of their prosthesis is being studied by the members of the electronics department.



Figures 2.8 and 2.9: Detail of the digital and analog connections of Andromeda by December 2021. The elements with a free pin (LEDs and ON/OFF switch) are also connected to the ground, while elements with two free pins (EMG sensors, temperature sensors [pins A10, A11] and servos) are connected to the ground and the 5V input. Source: self.

2.3.Summay

The world of myoelectric arms is still a field in development, where constant innovation is sought in order to give the users a prosthesis with equal or better performance than an organic arm. In the world of prostheses, the myoelectric models are in the higher end of the available products, with advanced functionalities that are highly sought after. This is reflected in their price, which turns them into a luxury that many potential customers can't afford. Due to the complexity of their creation, it stands to reason that these devices were typically reserved to the wealthy, although fortunately the research of new technologies is slowly making myoelectric arms more affordable, especially thanks to 3D printing. ARM2u can find its own space here, as an economic yet highly functional option for those in need. Unfortunately, the team still has a long road ahead of it, although the first steps have been taken, and ARM2u has learned from its past experiences and is currently building a promising prototype for upcoming competitions.

But the lack of satisfaction among users is something that even the high end models suffer. The reviews of the prosthesis mentioned in 2.1 showed that a better method to switch modes is needed, because in spite of most prostheses having more than 10 different grip modes most of the users only used between 4 and 6 in their daily lives, prefering to adapt to the use of a few of them even when their use was suboptimal instead of utilizing the whole potential of their prosthesis. Therefore, innovative methods to control the modes of the hand could incite users to deploy their prostheses at their full potential, increasing their performance and with that, the satisfaction of the clients and their confidence.

Another interesting improvement is the integration of sensory feedback, which can increase the precision of Andromeda and the user's sense of ownership.

3.Proposed solutions

3.1.Improvement of the mode selection

3.1.1. Board research

Andromeda originally used the Arduino Mega microcontroller, but the size of the board proved to be problematic when the physical prototype was ensembled. The current iteration is expected to work with Arduino Uno, which offers all the functionalities Andromeda needs yet occupies almost half the space of the previous board, although it is only intended as a temporary solution. A research for new boards that can offer a better performance while occupying less space has been conducted, using the following criteria:

- The board size must be inferior to 101x53mm (the dimensions of the Arduino Mega). Smaller sizes than Arduino Uno are prefered, as they are easier to integrate into the prosthesis.
- Due to the number of signals that the prosthesis will receive the microcontroller must have a clock speed of at least 48 MHz, although as stated before higher values are prefered, since the increase of complexity in the program could demand higher processing speeds.
- A mínimum of 10 analog input pins will be required. Despite the previous iteration employing a total of 5 analogical input pins for the EMGs (electromyographic sensors) and security elements, at least one additional pin will be needed for a pressure sensor, with a total of four more (one per sensor attached to each finger) eventually being necessary. Two or more additional pins are considered highly recommended for future EMG or security additions, although they are not mandatory.
- Eventually at least 11 PWM (pulse width modulation) pins will be needed: 6 for the servo motors in charge of the movements of each finger and the wrist, and 5 for the vibration motors used in tactile feedback. LEDs and switches can be connected to either PWM or GPIO (general purpose input or output) pins, and therefore offer more flexibility in terms of requirements, with only 2 required for battery life LED and ON/OFF switch.

- The current program doesn't use EEPR (or EEPROM, electronically-erasable programmable read-only memory), but requires 578 bytes (approx, 0.6KB) of SRAM (static random access memory) and 9926 bytes (approx, 1 KB) of Flash memory, although future iterations are expected to require more space. Therefore, the bare minimum memory required will be 2KB Flash and 1 KB SRAM.
- Integrated WiFi and/or Bluetooth modules are highly desired, since they save further space in the prosthetic.
- They must be compatible with the Arduino IDE.

In the following page there is a table with some of the boards that were found, along with its main characteristics. Cells highlighted in yellow indicate that despite being within the parameters the capabilities of the board may not suffice the needs of Andromeda, cells highlighted in red outrightly fail to meet the minimum requirements exposed above.

Name	**Size (mm)**	**Operating & input voltage (V)**	**CPU Speed (MHz)**	**Analog In/Out**	**Digital Io/PWM**	**EEPR/ SRAM/ Flash (KB)**	**WiFi/ Bluetooth module**	**Average price (€)**
Arduino Mega 2560Rev3	101x 53	5/7-12	16	16/0	54/15	4/8/256	External	35
Arduino Micro	48x18	5/7-12	16	12/0	20/7	1/2.5/32	External	18
Arduino Zero	68x53	3.3-7/12	48	6/1	20/10	16/32/ 256	External	33
Arduino Nano RP2040 Connect	45x18	3.3-5/21	133	8/0	20/20	448/520 /16000	Integrated	21

Arduino Uno WiFi REV2	69x53	5/7-12	16	6/0	14/5	0.256/0.0061/48	Integrated	39
Arduino MKR WiFi 1010	62x25	3.3-5	48	7/1	8/13	0/32/256	Integrated	28
Teensy 4.1	61x18	3.3-5	600	18/0	55/35	4/1024/7936	External	27
Particle Photon	37x20	3.3-3.6/5.5	120	8/1	20/7	64/128/1000	Integrated	19
Seeduino nano V4.2	43x18	5-7/12	20	8/0	14/6	1/2/32	External	8
ESP32 DevKitC 32D	48x28	3.3-5	240	15/0	34/16	448/520/4000	Integrated	11
ESP8266 NodeMCU	49x26	3.3-4.5/10	80	1/0	11/4	0/64/4000	Integrated	9
Thing plus ESP32 WROOM	59x23	2.3/3.6-5	240	13/2	21/16	448/520/16000	Integrated	23

Table 3.1: Comparison of the capabilities and prices of different microcontroller boards. Sources: [21, 22, 23, 24, 25, 26, 27]

Looking at the table above, we can see that many of the boards fail to meet the requirements. Of the two options we are left with, ESP32 DevKit 32D by Espressif has been chosen due to its greater number of pins, slightly lesser size and more affordable cost. Therefore, and as the proposed board for future iterations for Andromeda, the rest of the project will use it as reference.

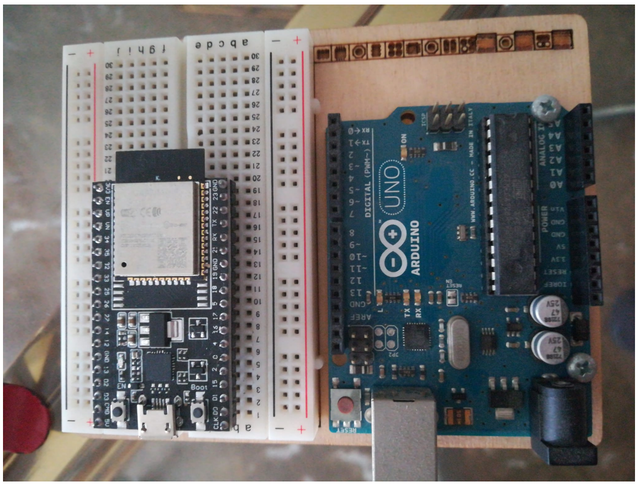


Figure 3.1: Size comparison between ESP32 DevKit 32D and Arduino Uno. Source: self.

In addition to the above mentioned characteristics, further investigation of the ESP32 revealed that it allows us to work with BLE, or Bluetooth Low Energy, which drastically reduces the battery consumption. Considering that the prosthesis will be operative during most of (if not all) the waking hours of the user, this factor will not only drastically increase the battery life expectancy, but will also increase the autonomy of the user.

3.1.2. Improved switch mode

As seen in points 2.1 and 2.2, despite current myoelectric prosthetic arms using myoelectric control to switch modes (which may be aided with a phone app), Andromeda uses a manual switch in order to alternate between different modes, which causes several complications. Due to the inconveniences that prosthesis users suffer when they try to use myoelectrical control systems, an alternative method has been proposed in the form of voice recognition control through a wireless device. The end goal is to give Andromeda users the option to switch between modes both through the prosthesis voice control and through a phone app, which will be useful for situations in which silence is required or ambient noise makes operating the prosthesis through voice commands a hassle.

In order to implement a voice control method, an app that sets the groundwork for a future Andromeda application will be programmed, which will be able to send verbal commands to the ESP32. Implementation of a recording device into Andromeda has been left out as a task for future iterations due to current redesigns in the arm structure, although adaptation of the program shouldn't be a major inconvenience.

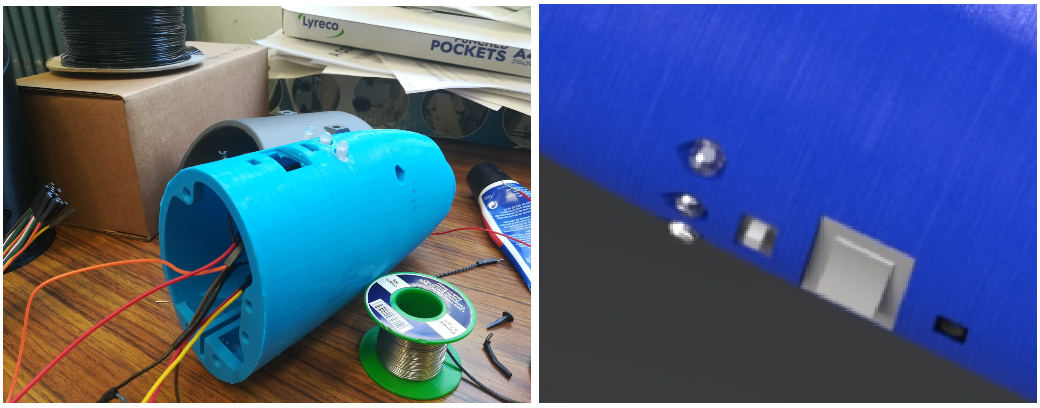


Figure 3.2: Andromeda's socket before the switches were incorporated (left) and close-up of the switch mode button on a render, between the LEDs and the ON/OFF switch. Source: self (left) and ARM2u (right).

3.1.3. Implementation of the new switch mode

3.1.3.1. Introduction to ESP32

ESP32 can be programmed using an Arduino IDE with C++ or in MicroPython. Due to the pre-existing programs developed by ARM2u, the Arduino IDE is prefered. In order to work with the ESP32 using the Arduino IDE, the first step is to add the ESP32 development boards using the Arduino IDE Board Manager, following the steps found on Dronebot Workshop [28].

After the new board was installed, some test programs were run in order to get a first taste, using breadboard circuits to physically test the programs. The first ESP32 program tested was a basic LED example. It was highlighted that for ESP32 the serial monitor, a tool which allows the display of text messages sent between the computer and ESP32, is usually set up with the "Serial.begin(115200);" line. The baud rate of the ESP32 (which represents the maximum number of bits per second that can be transmitted) is 115.200, while Arduino typically works with a baud rate of 9600 bps (although it can perform at higher rates). This means that on average the ESP32 can transmit 12 times more information per second.

3.1.3.2. Servo motor control through ESP32

After running other test programs, the servo motors were tackled. Servo motors are controlled through the microcontroller using PWM (pulse width modulation). Arduino IDE comes with a built-in servo motor library, but it doesn't work with ESP32. Fortunately, many free use libraries are available. For the purposes of this project, the *ESP32 Servo* library designed by Kevin Harrington was chosen and installed using the Library manager, following dronebot's article on ESP32 and servo motors [29].

While some ESP32 servo libraries only support a limited number of servos, the one we are using supports the 16 PWM pins of the board, although some of the GPIO pins that support PWM may also be needed for other applications of Andromeda in the future. If that were the case, this issue could be solved with the PCA9685, a 16 channel PWM controller, allowing even more servos or other PWM controlled devices. It is also necessary to remark that since the prosthesis is currently experiencing a remodeling process the code will be adapted: The received orders will not only switch, but also activate the modes integrated.

The ESP32 offers a 3.3V output, while most servo motors need a 5-6V output. This can easily be solved by creating a level shifter with transistors, but for the purposes of the prototype an already available 6V battery was used instead. Andromeda also uses a battery as a power source, although its rated voltage is 5V. Therefore, In order to reduce the electrical load on the equipment and to recreate operating conditions similar to the ones experimented with Andromeda, a voltage divider was implemented. A voltage divider is a passive linear circuit that produces an output voltage lower than its input voltage. This circuit can already be seen as one of the security elements connected to the A12 pin in the figure 2.6*: Detail of the analog connections of Andromeda by December 2021*.

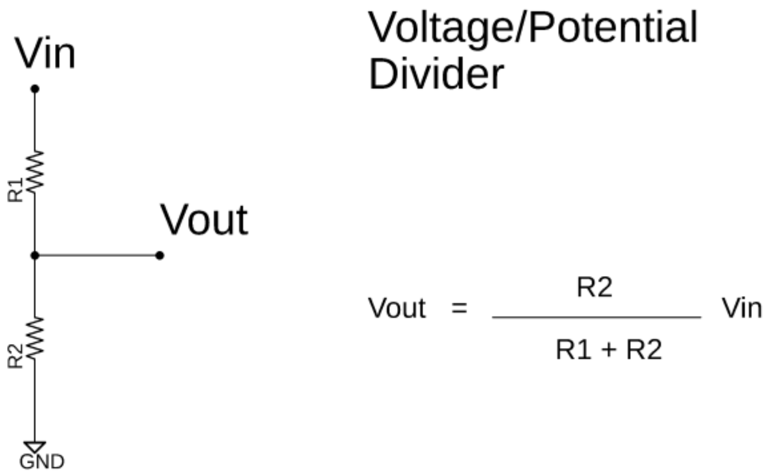


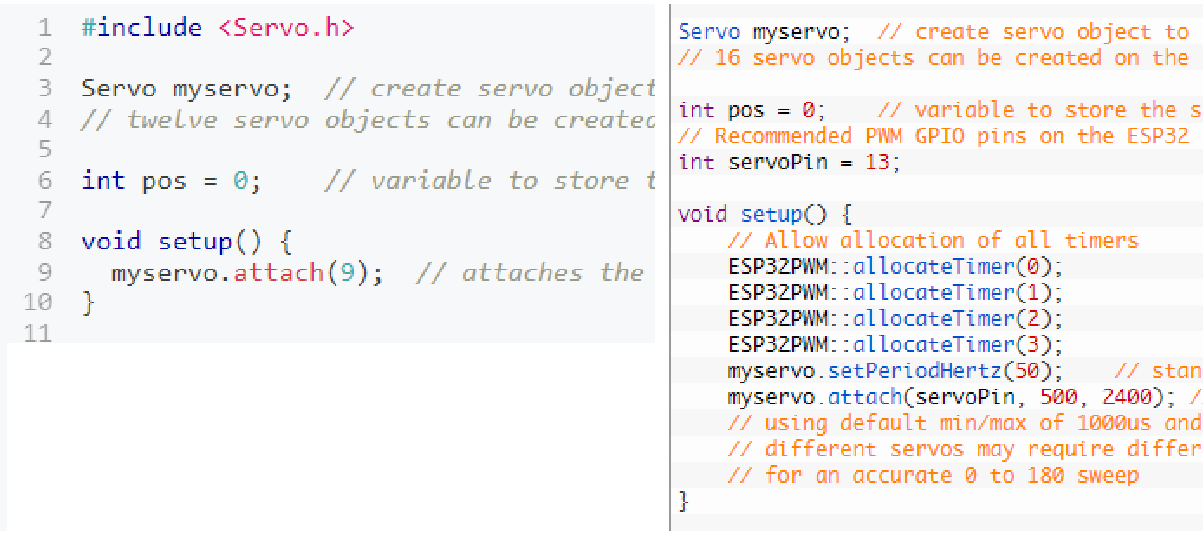
Figure 3.3: Representation and output calculation of a voltage divider. Source: [30]

As showcased in the image, the only thing we need to create a voltage divider are two resistors and a power source. Using the multimeter available at ARM2u's office the real voltage of the battery was rated between 6.5 and 6.7 V, which could seriously damage the components. For a desired 5V output, the equation becomes (Vin-Vout)*R2-Vout*R1 = 0, which roughly means R2 = 3.125*R1. We will try to find a suitable option among the resistors that we already have, taking into account the lectures of the multimeter, which gives a desired range for R2/R1 of [2.94, 3.33], favoring the numbers which give us higher protection against the voltage. Looking at the available resistors, the following options were found:

Color code	Value
Red/Red/Brown/Gold	220 Ω
Green/Blue/Brown/Gold	560 Ω
Brown/Black/Red/Gold	1 KΩ
Yellow/Brown/Red/Gold	4.1 KΩ
Gray/Red/Black/Brown/Brown	8.2 KΩ
Brown/Black/Orange/Gold	10 KΩ
Brown/Green/Orange/Gold	15 KΩ
Brown/Gray/Green/Gold	1.8 MΩ
Brown/Black/Blue/Gold	10 MΩ

Table 3.2: Available resistors. Source: self.

The highlighted values offer the closest fit: 15/4.1 = 3.658. With R2 = 15 and R1 = 4.1 the output value is approximately 5.2V, which is way better for the equipment than the previous voltage. After the divider was physically ensembled, two of the example programs available at ESP32Servo (*Sweep* and *Knob*) were tested in order to get familiar with the control of servos through ESP32 and check that all the components worked properly. These two examples were taken from modified classical Arduino sketches and adapted for ESP32.



```
#include <Servo.h>

Servo myservo;  // create servo object
// twelve servo objects can be create

int pos = 0;    // variable to store t

void setup() {
  myservo.attach(9);  // attaches the
}

```

```
Servo myservo;  // create servo object to
// 16 servo objects can be created on the

int pos = 0;    // variable to store the s
// Recommended PWM GPIO pins on the ESP32
int servoPin = 13;

void setup() {
    // Allow allocation of all timers
    ESP32PWM::allocateTimer(0);
    ESP32PWM::allocateTimer(1);
    ESP32PWM::allocateTimer(2);
    ESP32PWM::allocateTimer(3);
    myservo.setPeriodHertz(50);    // stan
    myservo.attach(servoPin, 500, 2400); /
    // using default min/max of 1000us and
    // different servos may require differ
    // for an accurate 0 to 180 sweep
}
```

Code 3.1: Comparison between the Arduino (Left) and ESP32 (Right) Sweep examples. Source: [29]

The main difference between both codes is the need to allocate the timers of the PWM when using the ESP32 Servo library. Another key difference is in the attach command, which is followed by two numerical values. These values represent the minimum and maximum pulse width, which represent the 0º and 180º position respectively. Their values must be found experimentally. For the servo motors used in Andromeda, those values were close to 500 and 2500. It is interesting to note that these parameters were adjusted in the Sweep sketch, and the parameters remained the correct values when the Knob program was tested.

3.1.3.3. Wireless control

In order to remotely control the program through a portable device (more specifically, a mobile phone), various methods were tested using the ESP32 WiFi and Bluetooth modules. ESP32 can act as an access point which we could use to switch between the modes of Andromeda, but it does not provide connection to the internet or other existing networks, limiting the functionality of our phone when we use it to control the prosthesis. It is also possible to use an already existing internet connection to program the remote control of the ESP32, but it can only share information with devices connected to the same network. Since the WiFi capabilities of the ESP32 limit the microcontroller to domestic use, the proposed solution uses its built-in Bluetooth capabilities instead.

A test code to switch between modes, designed to mimic the options implemented in Andromeda in December 2021 (grip and pronosupination), was tested. The servo motors were controlled through the Serial Bluetooth Terminal app, although any serial Bluetooth terminal controller should be able to effectively manipulate the prosthesis as long as it is connected to the ESP32.

In order to switch between modes a simple alphanumeric code consisting of four commands designed to replicate the functions of Andromeda was programmed: “open” and “close” replicated the options to open and close the hand to a set angle, while “turn” activates the continuous pronosupination. Finally, the “stop” command turned off all active servos. The orders must be written in the Serial Bluetooth App screen, and in order to offer an increased speed the input commands were simplified to “o” (open), “c” (close), “t” (turn), “s” (stop). Due to the structure of the code, the addition of new commands is a trivial endeavor, but this solution is still lacking in a key aspect: Despite providing a basic way to easily switch between modes, the user still needs to use their hand to do so. The Serial Bluetooth app is unable of voice recognition and, therefore, a new control app had to be designed.

3.1.3.4. Voice control and app design

An example voice-controlled application to switch a LED on and off for ESP32 developed using MIT’s App Inventor, a visual programming environment, served as the groundwork to build Andromeda’s voice control app. The base app used Google’s voice recognition to give three different orders to a LED: turn on, turn off and blink. Unfortunately, the ESP32 program for this app used WiFi, requiring several adaptations.

MIT App Inventor also allows us to design an Android app to connect to the ESP32 using the Bluetooth of our phone. After a basic sketch to control LEDs was implemented, a code to integrate the voice control of the previous application was developed. The first problem found was that the bluetooth connection and the voice recognition program were triggered as soon as the App started, causing an overlap problem in which the Bluetooth connection prevented the voice recognition from triggering again. To temporarily halt the problem the triggers for the voice recognition were changed, using a virtual button instead. This allowed a fast verification of the connection between devices.

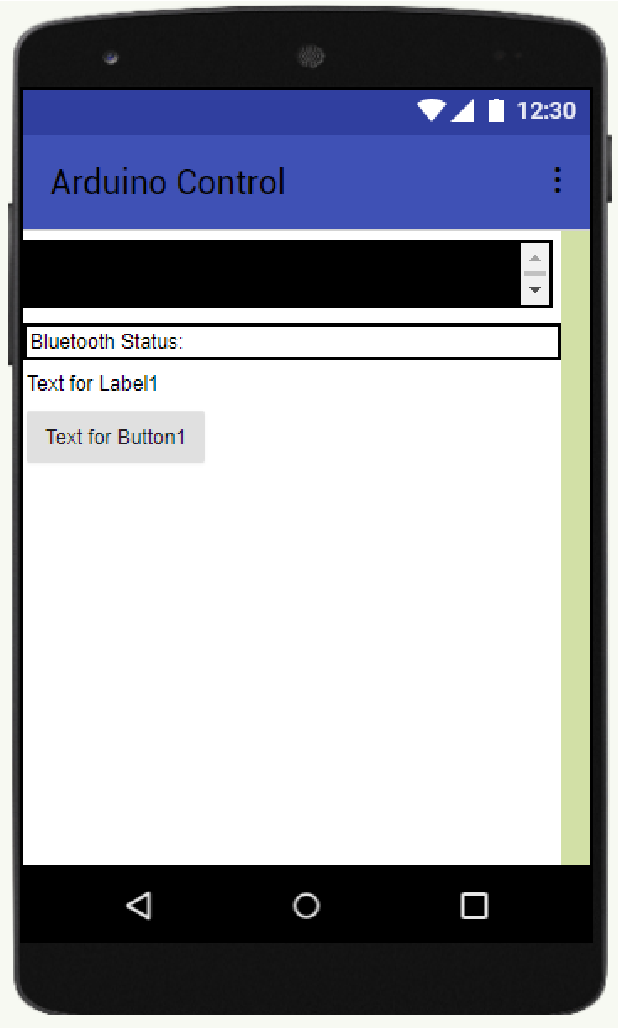


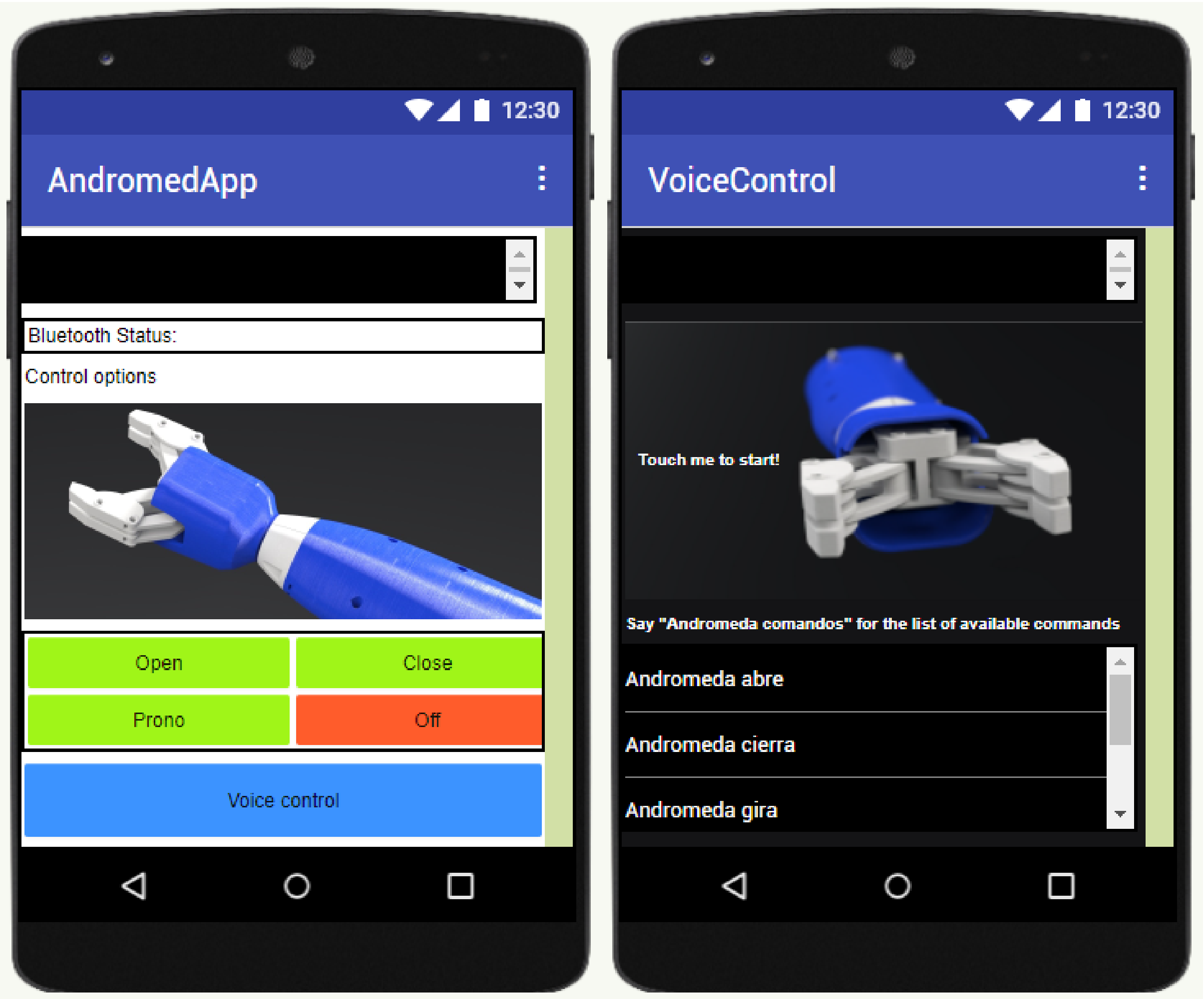
Figure 3.4: First iteration of the voice recognition App.

During said trials it was noticed that despite the application accurately recording the voice commands (thanks to a text programmed to showcase the recorded messages in the app), those weren't sent in the correct format to the ESP32. The microcontroller received an apparent random number instead of words, which upon further inspection were revealed to be a numerical equivalent of the recorded orders in an unknown code. The option to simply modify the ESP32 code to work upon receiving those numerical equivalents instead of the pronounced words was studied, but was deemed a solution for the short term that would cause greater problems in the long run, as the need of finding out new equivalences for new commands would become a bigger hindrance as time progressed.

A study of potential functions that could solve the problem was carried out, from which "*Serial.readStringUntil()*" adapted to "*ESP_BT.readStringUntil('\r')*" was considered the best option. This function reads the information from ESP_BT (the Bluetooth Serial object of the ESP32 code) as a String until it reaches the Carriage Return (‘\r’), which marks the end of the line. With this function, successful verbal control of the LED through the App was achieved. Afterwards, the LED control was switched to a Servo control, allowing us to give the orders “cierra” (close), “abre” (open), “gira” (rotate) and “para” (stop). In order to reduce the risk of ambient noise accidentally triggering one of the commands, they were modified to “Andromeda cierra”, “Andromeda abre”, “Andromeda gira” and “Andromeda para”.

Now that basic voice recognition was ready, the next step was programming the method to manually switch between the modes of Andromeda through the App, which was quickly done with new virtual buttons that sent a string with the words to trigger the new mode when pressed. Once non-verbal control through the app was designed, a way to get constant voice recognition was implemented. The button to trigger voice recognition was modified to send the user to a new window, which would trigger a loop that constantly asked for new orders. This new window also showcased the available commands on screen, which the user could access by tapping into the screen, in case the user needed to remember any of them. This effectively stopped the constant voice recognition, therefore a new button to re-trigger it was installed, which showcased a picture of the prosthesis and the message “Touch me to start!”. The commands “Andromeda comandos” and “Andromeda regresa” were included to verbally stop voice recognition and showcase the list of commands and to go back to go back to the main screen, respectively.

The last step, after multiple iterations, was to make the user interface aesthetically pleasant and consistent with the brand of ARM2u. The app was called AndromedApp, using the logo of ARM2u as the icon for the application. Consequently, the App used mostly the colors of the team and the arm (white, dark gray and blue) for the interface. Images of the prosthesis were also introduced in both screens, with the one on the voice control screen acting as a trigger to restart the recognition system. The app was successfully installed on the phone device using a QR code generated by the website of the App inventor.

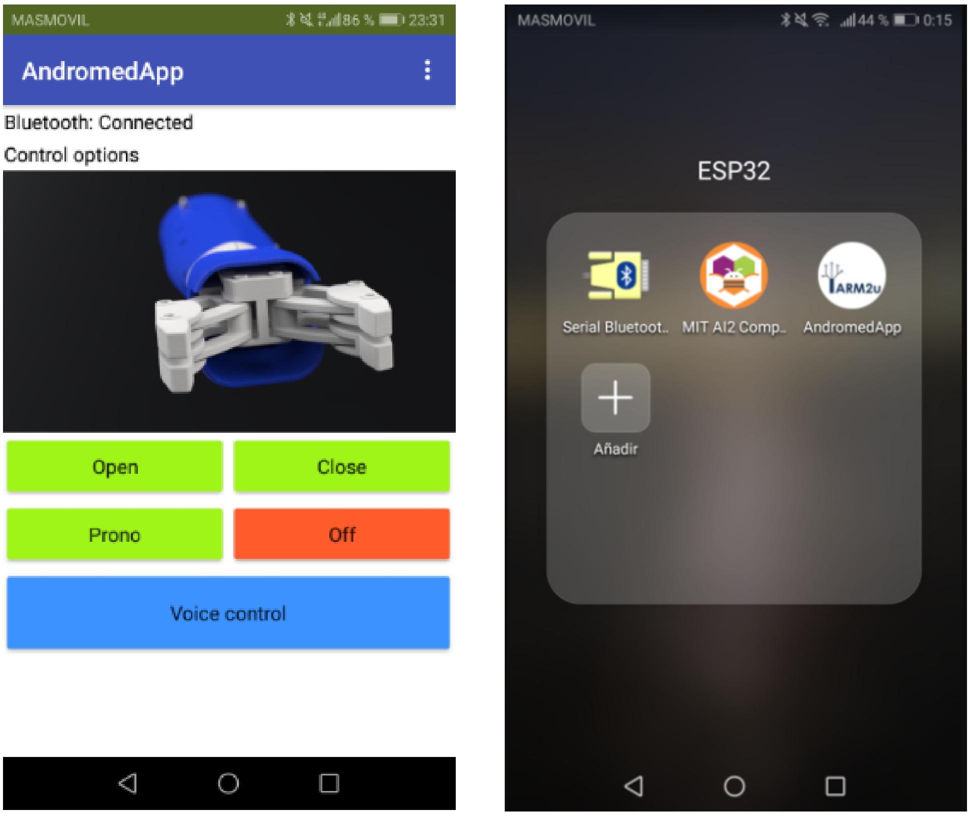


Say "Andromeda comandos" for the list of available commands

Figure 3.5: Main menu (left) and voice control screen (right) of the second iteration of the voice recognition App. Source: self

Usage of multiple windows had an unexpected drawback: The Bluetooth needed to be re-linked everytime the user switched between screens. The option to put the constant voice recognition option on the same screen was implemented, modifying the values used to trigger the loop in order to start or close it at will, bringing the definitive version.

During the installation process we found that most devices will show a warning when trying to install the app, since it comes from an outside source, although it is a safe one and no problems should arise from it. For future plans of brand expansion, it is also necessary to remark that MIT allows to update the apps developed through their App Inventor to google play, making them accessible to everyone who uses the arm.



Bluetooth: Connected

Control options

Figure 3.6: User interface of the final App (left) and the App installed in a user's phone with ARM2u's logo (right). Source: self.

The App operates in the following way: When it is started, the variable “key” is created and assigned the value “0”. Simultaneously, as the user interface (Screen1) is initialized, it showcases a list with all the Bluetooth devices linked to the user's phone. When the Andromeda microcontroller is chosen, if it is able to connect the list of devices disappears, showcasing the user interface. If there is a mistake with the connection or no devices are available, the list of devices will continue to block the interface until a valid device is picked.

Said interface showcases five buttons: four distributed in the middle that allow the user to switch between the previously indicated modes, and a fifth one in the bottom which activates the voice recognition of the App. When pressing a button or issuing a voice command, the App sends a text message to the microcontroller with either a predefined command or the transcription of the user's words, causing the servo motors to react.

The voice recognition application works through a recursive loop. When the “key” variable defined at the start has a value of 1, google’s voice recognition is triggered automatically, and after every command is issued the value of “key” becomes 1. The voice recognition is also triggered when the bottom button (voice control) is pressed, which initiates the loop. When google’s voice recognition is launched, “key” becomes 0 again until the command is issued, and therefore the loop ends if the screen is touched, until the blue button is pressed again.

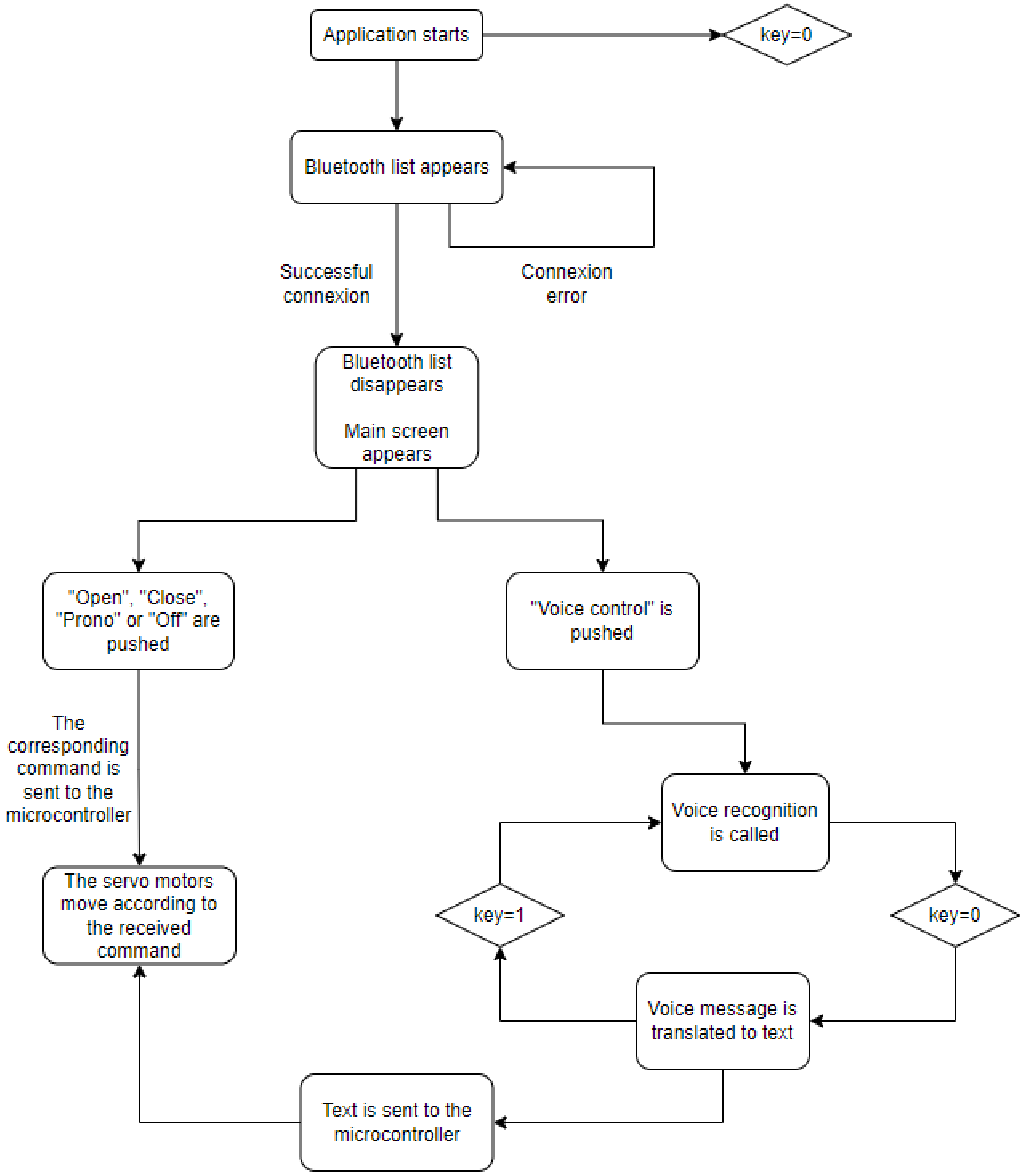
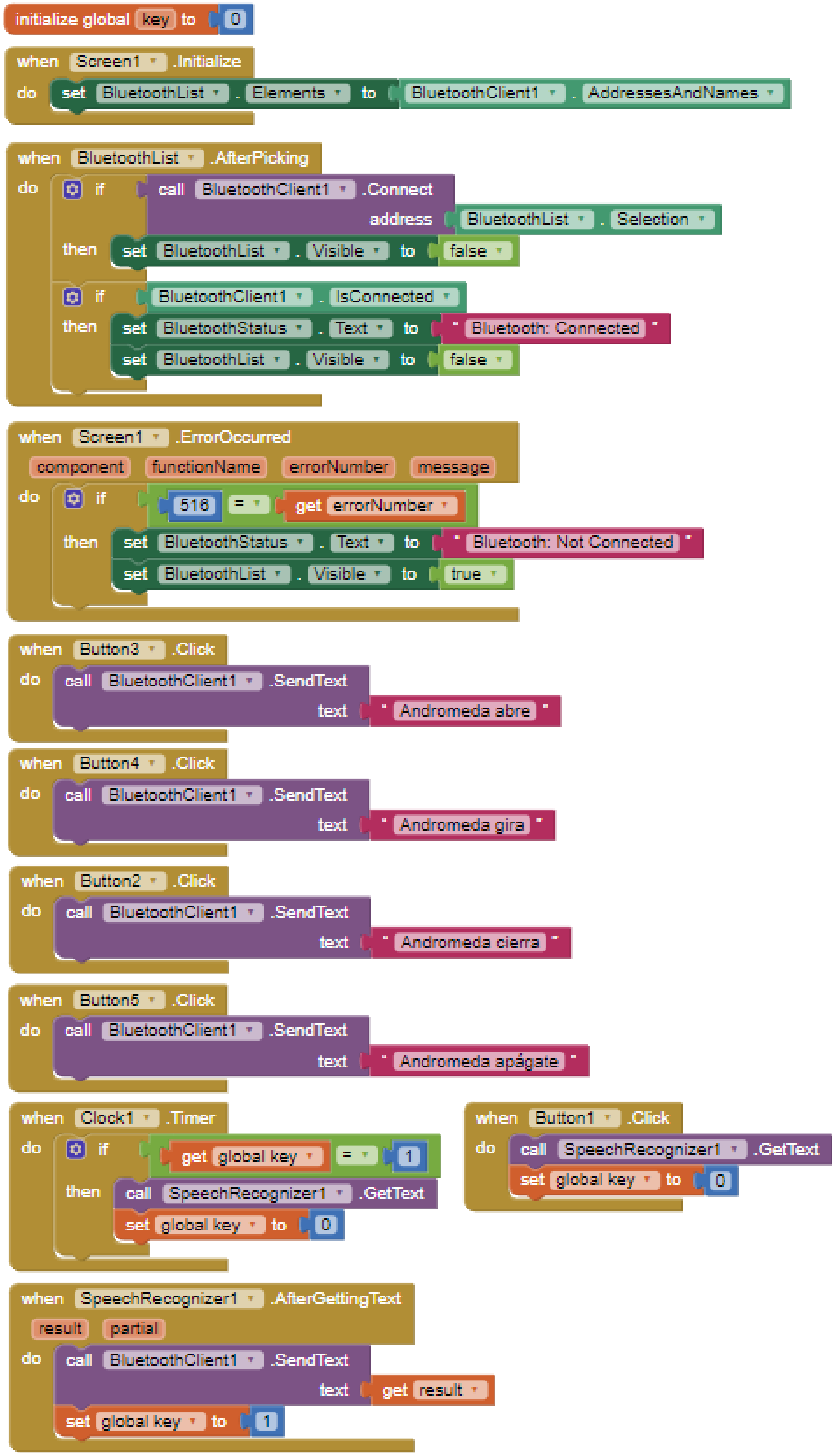


Figure 3.7: Flowchart of AndromedApp. Source: self.

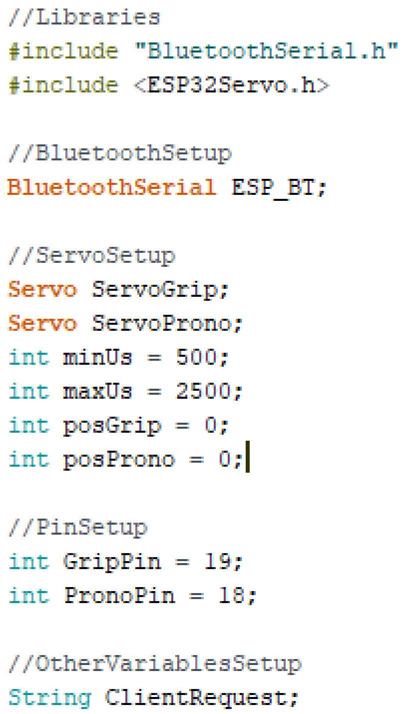


```
initialize global key to 0

when Screen1 .Initialize
do  set BluetoothList . Elements to BluetoothClient1 . AddressesAndNames

when BluetoothList .AfterPicking
do  if    call BluetoothClient1 .Connect
                                address BluetoothList . Selection
    then  set BluetoothList . Visible to false
    if    BluetoothClient1 . IsConnected
    then  set BluetoothStatus . Text to " Bluetooth: Connected "
          set BluetoothList . Visible to false

when Screen1 .ErrorOccurred
  component  functionName  errorNumber  message
do  if    516 = get errorNumber
    then  set BluetoothStatus . Text to " Bluetooth: Not Connected "
          set BluetoothList . Visible to true

when Button3 .Click
do  call BluetoothClient1 .SendText
                         text " Andromeda abre "

when Button4 .Click
do  call BluetoothClient1 .SendText
                         text " Andromeda gira "

when Button2 .Click
do  call BluetoothClient1 .SendText
                         text " Andromeda cierra "

when Button5 .Click
do  call BluetoothClient1 .SendText
                         text " Andromeda apágate "

when Clock1 .Timer
do  if    get global key = 1
    then  call SpeechRecognizer1 .GetText
          set global key to 0

when Button1 .Click
do  call SpeechRecognizer1 .GetText
    set global key to 0

when SpeechRecognizer1 .AfterGettingText
  result  partial
do  call BluetoothClient1 .SendText
                         text get result
    set global key to 1
```

Code 3.2: Block configuration of AndromedApp on MIT App Inventor. Source: self.

The ESP32 code works in the following way: The first thing it does is call the libraries it is going to use (bluetooth and servo) and define the global variables. Global variables are variables defined outside of any function but can still be called by all of them. When defining these variables we must indicate what kind of object represents (such as string or integer). In this case, the following variables are defined:

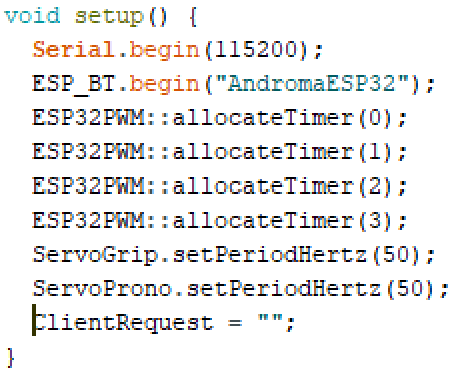
- **ESP_BT:** A bluetooth object, necessary for the exchange of information with the mobile phone.
- **Servo grip and Servo motor:** Servo objects, representing the two servo motors that control the movements of the gripper.
- **minUs and maxUs:** Two integers that will determine the minimum and maximum pulse width, set to 500 and 2500, the values found experimentally before.
- **posGrip and posProno:** Two integer variables used to store the position of the servos, started at 0.
- **GripPin and PronoPin:** The pins that will control the two servo motors. 19 and 18 respectively, although other pins can be assigned when the whole prosthesis is ensembled.
- **ClientRequest:** A string used to record the sent commands.



```
//Libraries
#include "BluetoothSerial.h"
#include <ESP32Servo.h>

//BluetoothSetup
BluetoothSerial ESP_BT;

//ServoSetup
Servo ServoGrip;
Servo ServoProno;
int minUs = 500;
int maxUs = 2500;
int posGrip = 0;
int posProno = 0;

//PinSetup
int GripPin = 19;
int PronoPin = 18;

//OtherVariablesSetup
String ClientRequest;
```

Code 3.3: Definition of global variables. Source: self.

The next step is the definition of void setup. The code written in this section will only run one time, as soon as the program starts. As seen in previous code, the Monitor Series is called here at a baud rate of 115200 bits per second. It also activates the Bluetooth of the microcontroller, and allocates the timers of the servomotors, as well as their periods, following again the examples previously tested. Last but not least, the setup gives “ClientRequest” the value of an empty string, “”.

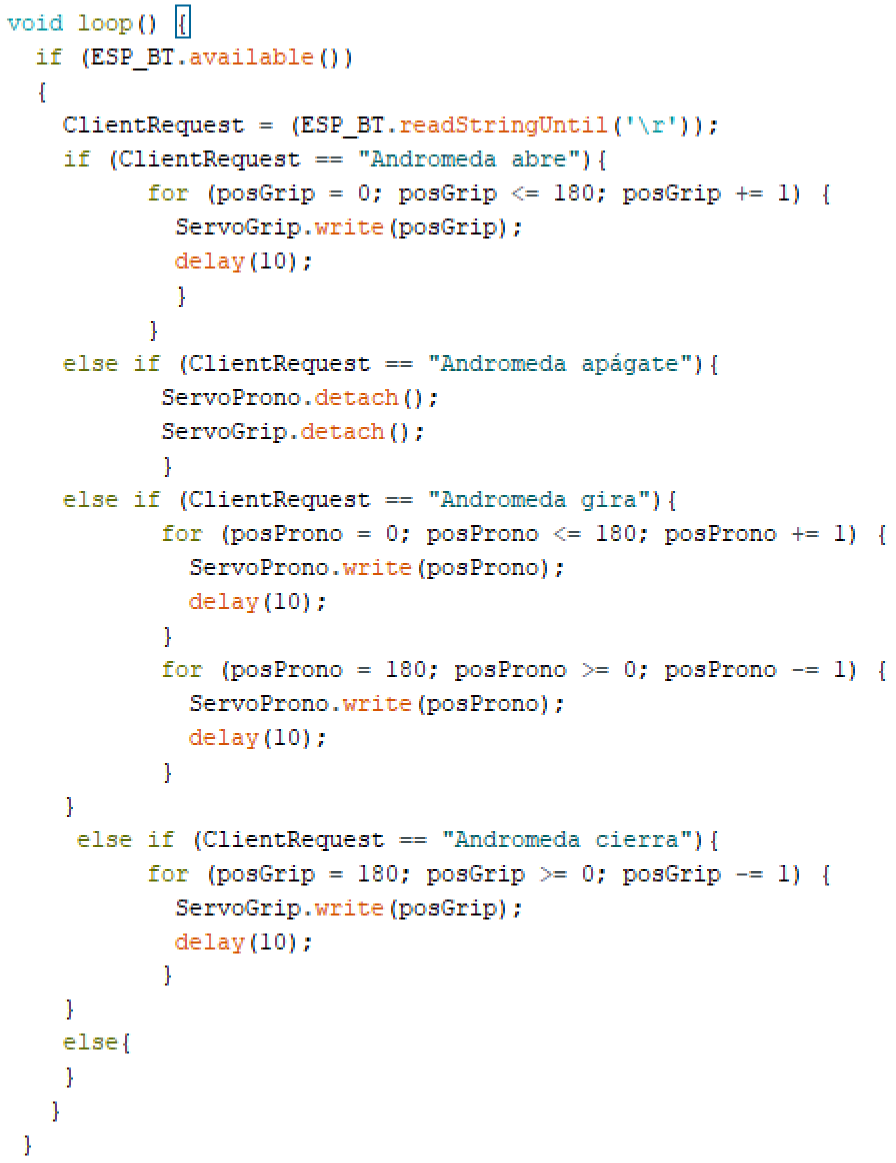


```
void setup() {
  Serial.begin(115200);
  ESP_BT.begin("AndromaESP32");
  ESP32PWM::allocateTimer(0);
  ESP32PWM::allocateTimer(1);
  ESP32PWM::allocateTimer(2);
  ESP32PWM::allocateTimer(3);
  ServoGrip.setPeriodHertz(50);
  ServoProno.setPeriodHertz(50);
  ClientRequest = "";
}
```

Code 3.4: Void setup. Source: self.

Finally, the loop checks if the device is connected via Bluetooth. If the answer is yes, then it attaches the servos to their respective pins and reads the messages sent either verbally or after pressing a button through the connected device. If ClientRequest is one of the four programmed commands, an order is given to the servomotors:

- If the command is “Andromeda abre”, the position servo performs a 180º rotation, fully opening the gripper.
- If the command is “Andromeda cierra”, the opposite happens: The position servo turns back to the 0º position.
- If the command is “Andromeda gira”, the speed servo is activated, providing continuous rotation.
- If the command is “Andromeda apágate” both servos are disconnected, stopping all their functions until a new command is issued.



```
void loop() {
  if (ESP_BT.available())
  {
    ClientRequest = (ESP_BT.readStringUntil('\r'));
    if (ClientRequest == "Andromeda abre"){
          for (posGrip = 0; posGrip <= 180; posGrip += 1) {
            ServoGrip.write(posGrip);
            delay(10);
            }
          }
    else if (ClientRequest == "Andromeda apágate"){
           ServoProno.detach();
           ServoGrip.detach();
           }
    else if (ClientRequest == "Andromeda gira"){
           for (posProno = 0; posProno <= 180; posProno += 1) {
             ServoProno.write(posProno);
             delay(10);
           }
           for (posProno = 180; posProno >= 0; posProno -= 1) {
             ServoProno.write(posProno);
             delay(10);
           }
    }
     else if (ClientRequest == "Andromeda cierra"){
          for (posGrip = 180; posGrip >= 0; posGrip -= 1) {
            ServoGrip.write(posGrip);
            delay(10);
           }
    }
    else{
    }
   }
 }
```

Code 3.5: Void loop. Source: self.

3.2.Tactile Feedback

3.2.1. Proposed solutions

Doctor Ulrika Wijk, from the University of Lund (Sweden), published in July of 2020 a study on an artificial hand that allowed a certain degree of tactile feedback [31]. The device used air-mediated pressure in order to relay the strength exerted by the user's fingers. A silicone glove with bulbs in every fingertip was made and applied on a single degree-of-freedom prosthetic hand. When the silicon bulbs in the fingertips were pressed, the air was transferred via plastic tubes that reached actuators inside the prosthetic socket and gave pressure (mechano tactile feedback) on the skin corresponding with the PHM zones [32].

The results of the study showed an increase in positive subjective experiences linked to body ownership and experiences of sensory feedback from the prosthesis, but did not improve the performance with the prosthesis. The participants of the study expressed a desire for stronger feedback. The bulbs were also deemed impractical, as their size made them bulky for fine manipulation, and too soft to hold certain objects.

An initial proposal to adapt this system to ARM2u's prosthesis was studied, but it was deemed not feasible. Usage of a hydraulic feedback system instead of a pneumatic one to increase the pressure of the feedback was proposed, but the potential of leaks that could damage the prosthesis or damage the user, their clothes and their environment demanded an alternative solution.

In addition to these circumstances, the ARM2u prosthesis uses a slip ring to allow continuous pronosupination (or constant wrist turn, a movement that would not be feasible with an organic hand) without risking the integrity of the electrical circuit. The risk of cables being damaged due to that same constant pronosupination makes this solution unfeasible, but fortunately it's not the only one.

The next proposed solution was a vibrotactile stimulation system. Vibrotactile stimulation is when tactile sensation is evoked by a mechanical vibration of the skin, typically at frequencies of 10–500 Hz [32]. A simple design using small low-power vibration motors (the same that can be found in smartphones) could allow the transmission of tactile feedback directly into the phantom hand map. Most prosthetic hand tactile feedback systems experience problems to detect grip force, specially when covered with cosmetic gloves. Therefore, this problem should be taken into account when designing the feedback system for the ARM2u prosthesis.

The feedback system was designed to be directly implemented on the gripper showcased in December. It consists of a pressure sensor allocated in one of the pincers and a coin vibration motor (similar to the ones found in mobile phones) integrated in the socket that will generate a stimulus proportional to the force applied directly to the skin of the user. Due to divervenges between the gripper and an anatomically correct hand only one sensor and motor will be used in the physical implementation, which will allow the user to feel the intensity of the gripping strength.

This has been decided because the gripper can only perform one range of motions (open-close) and therefore the only feedback considered relevant was the amount of pressure exerted when the gripper is holding an object. The three available prostheses showcased in the table 2.1.2.1 (the Ability hand from Psyonic [33], the MeHand from MaxBionic [34] and the Vincent Evolution by Vincent Systems [35]) all use a single motor to provide sensory feedback, and therefore this solution is considered appropriate from a commercial perspective, although the use of a higher number of sensors and motors and the mapping of the user's phantom hand map could be implemented for increased accuracy in more advanced iterations.

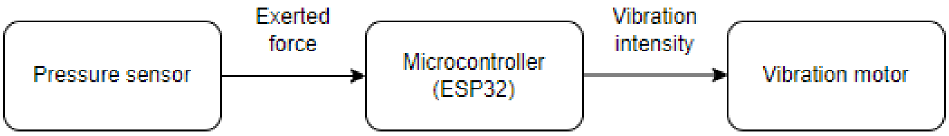


Figure 3.8: Flowchart of the feedback system. Source: self.

Pressure sensors detect the force exerted on a fixed surface as an input and turn it into an electrical output. Those sensors consist of a pressure sensitive element (usually called diaphragm) which deflects when pressure is applied, and a transduction element which turns the sensed deflection into the electrical output that will proportionally increase or decrease according to the pressure change. The conversion is usually done thanks to piezoresistive elements, which change their electrical resistance proportionally to the pressure they experience, increasing and decreasing when tensile and compressive strains are applied, respectively [36]. A search for viable sensors was conducted, using the following criteria:

- The sensor must withstand a maximum force of 70N, taken from the average maximum male fingertip pinch strength of males [37]. This value matches the values showcased on 2.1.2 for the minimum strength needed to perform most daily activities.
- The diaphragm must have a contact surface equal or smaller than the pincers and the fingertips of the hand. The pincertips have a surface of 3x2’5cm, while the fingers of the flexible PLA hand have a surface of roughly 1x1cm.
- The sensor must be structurally compatible with both the pincer and the fingers. That means, it must be sufficiently small or flexible to be integrated.
- The sensor must work with 3.3 V DC.
- The price and weight of the sensors must be as small as possible.

The research yielded various sensors that worked within the requirements. Among those, the DF9-40 was chosen, a flexible sensor that can be incorporated in the inner surface of the fingers. Multiple maximum force values were available (between 2 and 20 Kg). Due to its intended constant use and the difference in price ranges, which was less than 5€, the highest capacity option was chosen. The sensor has the following characteristics:

Force range	0-20 Kg
Length	40 mm
Thickness	0.25 mm
Diameter	9 mm
Accuracy	+/- 2.5%
Resistance	10 MOhms without load 100-1 KOhms with load
Response time	1 ms
Restoration time	15 ms
Operating voltage	3.3 DC

Force range	0-20 Kg
Length	40 mm
Thickness	0.25 mm
Diameter	9 mm
Operating temperature	Min: -20ºC Max: 60ºC
Weight	Lesser than 1 gram

Table 3.3: Properties of the DF9-40 sensor. Source: [38]

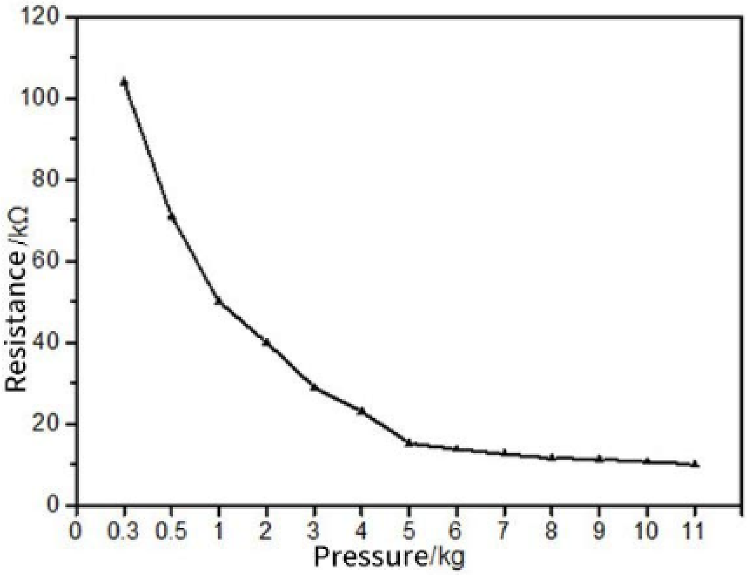


Figure 3.9: Evolution of the sensor's resistance with increasing loads. Source: [38]

Contrary to LEDs, we don't need to differentiate between the pins of the force sensor when we are making the connections, although the sensor needs to be connected to the 3V3 pin of the ESP32, besides the ground and the analog pin to read its value. The sensor only has two pins. Therefore, the ground and analog pins will share a terminal strip.

For the motor we will use a simple vibration coin motor directly allocated in the interior of the socket. Vibration motors convert electrical input into a mechanical output of variable intensity, and can be controlled either through PWM or a simple GPIO pin. Since regulable vibration is desired, the control will be through PWM.

The chosen model is the ROB-08449 by SparkFun, which has the following characteristics:

Rated voltage	3V DC
Operating voltage	2.3-3.6V DC
Starting voltage	2V DC
Rated current	60 mA
Rated speed	13000 +/- 3000 rpm (216,67+/-50 Hertz)
Operating temperature	Min: -30ºC Max: 60ºC
Maximum noise	50dB, 30 at rated voltage and load
Diameter	10 +/- 0.1 mm
Weight	1.179 grams

Table 3.4: Properties of the ROB-08449 motor. Rated voltage indicates the ideal voltage applied, operating voltage indicates the actual voltage applied that the equipment supports while performing safely within the specifications, and starting voltage is the minimum voltage required for the motor to start. Source: [39]

Both components work within the operational temperature ranges of Andromeda (-10/+50 ºC).

3.2.2. Programming of the vibration motor

The first step is to get familiar with the coding used for controlling the vibration of the motors. The test code showed that the vibration motors worked in the same fashion as the LEDs, and therefore could be controlled in a similar way. Like the LEDs, a correct connection of the motors is fundamental: The blue cable (which represents the negative lead) must be connected to the ground, or otherwise the coin motor won't vibrate.

After the first installment of the code, an upgraded version to modulate the intensity of the vibration was designed, using a PWM pin. This allowed control of the vibration intensity with a variable input. Following previous Servo examples, the control method used was a potentiometer regulated through a control knob. It's important to note that the analogWrite(Pin,Value) used to command analogical devices is not part of the implemented functions of ESP32. Fortunately, the servo library downloaded before has it, and therefore we only need to call it at the beginning of the test sketch to use it. The tests were deemed a success, allowing us to regulate the intensity of the vibration through the knob.

The input from the potentiometer had a range of values of [0,4095], which was proportionally converted to the motor. For the first run, the range of values of the motor was set to [0,255], taking the LED Brightness sketch as a base. Higher maximum values were tried in the next iterations, but no perceptible changes in the intensity of the vibration were noticed when the input value for the motor suprassed 255, and therefore the range of the first run was maintained.

3.2.3. Sensor programming

The next step was to test the force sensor in order to turn its outputs into the new control mechanism. A 10K resistor was implemented to the circuit in order to protect the sensor. The monitor series showed that the maximum value recorded by the sensor when direct pressure using the fingertips was applied was 3749, which was considered equivalent to the value showcased when pressure with the silicone fingertips was applied (3743).

Therefore, for the implementation of the sensor into the gripper we will use the same silicone used to increase the drag of said gripper. The silicone won't be directly applied over the sensor. Instead, the sensor will be fixed under the already existing “silicone globe” that covers the fingers thanks to the sensor's flexibility and dimensions.

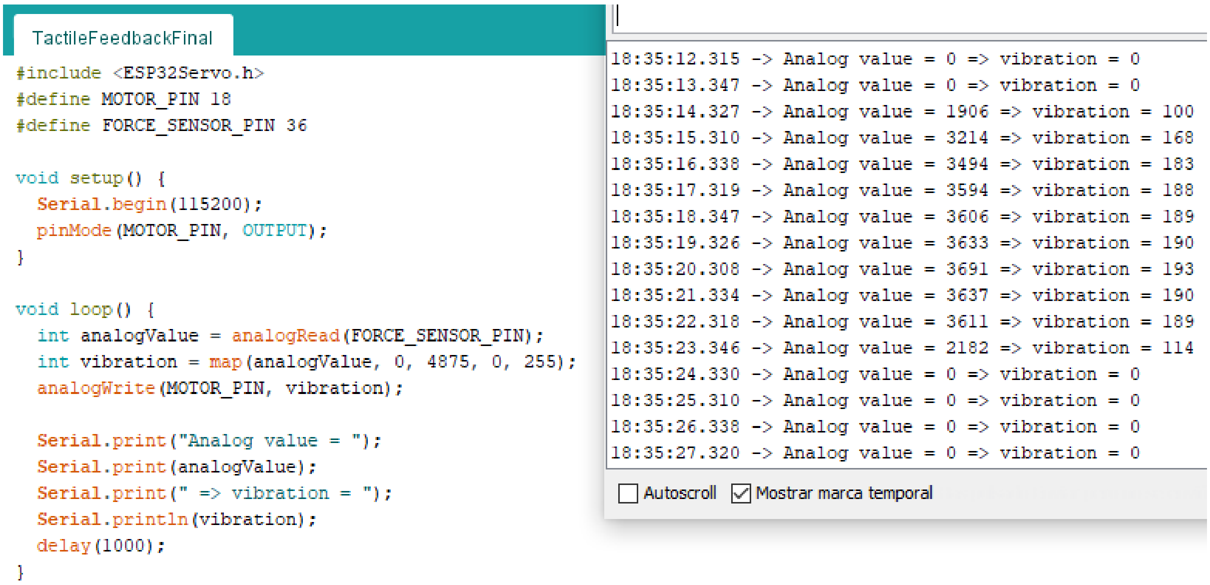
Naked values around maximum pressure	Cover value around maximum pressure
2767	2033
3411	3499
3625	3666
3668	3697
3670	3722
3632	3718
3654	3743
3749	3743
3353	3373

Table 3.5: Comparison of the digital outputs of the pressure sensor without and with silicone covers (left and right, respectively). Source: self.

All that was left to have the first iteration of the tactile feedback program was using the pressure sensor as input instead of the potentiometer, adjusting the map function accordingly.

The calculations were made using direct stimulation from an organic hand instead of the force exerted by Andromeda because, as stated in 2.2, the maximum force that Andromeda can yield is significantly smaller than what an average person can exert. Therefore, adjusting the outputs using it as a reference was detrimental, since small stimuli would trigger high intensity vibrations that would give a false sense of the applied strength.

Although in theory higher values than 3800 could be attainable, it is considered unlikely that Andromeda will be able to consistently reach them in the near future. Nevertheless, in order to reduce the potential strain on the motor and the noise of the vibrations, 4875 will be set as the maximum output value for the map function, which roughly equals 1.3 times the maximum obtained value. With the adjusted values, the code for the feedback system is finished.



```
#include <ESP32Servo.h>
#define MOTOR_PIN 18
#define FORCE_SENSOR_PIN 36

void setup() {
  Serial.begin(115200);
  pinMode(MOTOR_PIN, OUTPUT);
}

void loop() {
  int analogValue = analogRead(FORCE_SENSOR_PIN);
  int vibration = map(analogValue, 0, 4875, 0, 255);
  analogWrite(MOTOR_PIN, vibration);

  Serial.print("Analog value = ");
  Serial.print(analogValue);
  Serial.print(" => vibration = ");
  Serial.println(vibration);
  delay(1000);
}
```

```
18:35:12.315 -> Analog value = 0 => vibration = 0
18:35:13.347 -> Analog value = 0 => vibration = 0
18:35:14.327 -> Analog value = 1906 => vibration = 100
18:35:15.310 -> Analog value = 3214 => vibration = 168
18:35:16.338 -> Analog value = 3494 => vibration = 183
18:35:17.319 -> Analog value = 3594 => vibration = 188
18:35:18.347 -> Analog value = 3606 => vibration = 189
18:35:19.326 -> Analog value = 3633 => vibration = 190
18:35:20.308 -> Analog value = 3691 => vibration = 193
18:35:21.334 -> Analog value = 3637 => vibration = 190
18:35:22.318 -> Analog value = 3611 => vibration = 189
18:35:23.346 -> Analog value = 2182 => vibration = 114
18:35:24.330 -> Analog value = 0 => vibration = 0
18:35:25.310 -> Analog value = 0 => vibration = 0
18:35:26.338 -> Analog value = 0 => vibration = 0
18:35:27.320 -> Analog value = 0 => vibration = 0
```

Code 3.6: Code of the Tactile feedback with Monitor Series for value comparison

As we can see, the only needed libraries and variables are ESP32 servo and the two pins, respectively. The setup sets the pin assigned to the motor as an output, while the loop constantly reads the value of the sensor and reduces it to accurately control the vibration. The program also writes the registered and sent values, in order to study the pressure variation during the testing.

4.Validation

4.1.App tests

4.1.1. Test 1 - Command validation

This test will ascertain the precision of the programmed commands. The test must be performed once using the buttons of the app and once using the voice recognition function. In order to validate this test the following conditions must be met:

- **Open/Close:** The pincer makes a full range of motion, leaving an opening of less than 2 mm when the close command is issued.
- **Pronosupination:** The hand performs uninterrupted rotation at constant speed until new valid commands are issued.
- **Off:** Both servo motors stop.

Both methods conveyed the commands accurately, bringing the same results. Therefore, the test was considered valid.

4.1.2. Test 2 - Background noise

The objective of this test is to ascertain the background noise range in which the voice command is deemed operative. For this purpose, an example voice recognition app that wrote down everything it registered, developed using MIT App Inventor, was used. Simultaneously, the sound measurement “Sonómetro” App, developed by Splen Apps, was used to measure the level of ambient noise.

The voice recognition started to fail around 60 dB, the average sound level of a restaurant or an office, although a moderate tone was kept during the whole recording. Therefore, it is possible to give commands above said level in an emergency. It is necessary to remark that this test highly depends on the phone in which the app is installed, and therefore slight variations between users are expected.

4.2.Feedback tests

4.2.1. Sensitivity test

This test will determine the accuracy of the feedback system when discerning different textures. In order to do so, a total of 10 objects were used, 5 considered soft (sponge, t-shirt, plushie, cardboard box and bag) and 5 considered hard (glass, piggy bank, wooden plank, hardcover book and metal bottle). The testing required a volunteer, who had the pager motor fixed to his arm. The ten items were showcased to the volunteer, after which they were squeezed with the gripper, in order to give him a first feeling of what their haptic stimulus felt like. The gripper had the DF9-40 sensor fixed to one of its pincers under the silicone fingertip cover. The hand was not attached to the rest of the arm or a servo motor. Instead, it was manually controlled by direct pressure applied to both pincers, in order to obtain a force reading of higher magnitude.

To ensure that the volunteer relied only on his sense of touch, he kept his eyes closed and faced an opposite direction to the testing bench.

When conducting the test, pressure was exerted on each item in a random order, in order to see if the user could differentiate between them. This process was repeated for three consecutive rounds, with feedback and a new round of squeezing being conducted at the end of each round. The results showcased that despite the user differentiating between soft and hard objects with a 70% accuracy (out of 30 trials, the user correctly identified the category of the object in 21 occasions), individual recognition of each object was lower than desired: only a 36.67% of accuracy was achieved. An analysis of the results also show that the detection accuracy skyrocketed at the end of each round, evidentiating that a significant part of it was achieved due to deductive reasoning when fewer objects were left. A fourth round of 15 grasps in which objects could be grasped on more than one occasion was conducted. Despite object recognition managing to still maintain an accuracy close to 50% (53.33%), object recognition plummeted down to 13.33%, only 2 objects.

After the results were further examined, it was concluded that the level of accuracy desired requires a more advanced system, and possibly the use of multiple sensors and a precise phantom hand map. Therefore, the scope of the feedback system was reduced in order to enhance its accuracy, since differentiation between soft and hard textures has been demonstrated as possible.

A redesign of the feedback program will be conducted once the rest of the tests are finished, in case no further remodeling needs arise. In this new iteration, instead of using a proportional value to the input of the sensor a range of values will be defined. According to which range of values is perceived by the sensor, a different fixed value will be sent to the motor. The monitor series showcased the following range of values during the tests:

	Max (hard)	**Min (hard)**	**Max (soft)**	**Min (soft)**
Sensor	3513	2459	2822	1691
Motor	184	129	148	88

Table 4.1: Experimental values of the first haptic feedback test. Ref: self.

Taking the above information into consideration, the program will follow the next parameters:

Stimulus	**Value ranges (sensor)**	**Output (motor)**
No	[0,200]	0
Light touch	(200,1500]	50
Soft pressure	(1500,2450]	120
Hard pressure	(2450,3800]	200
Maximum pressure	>3800	255

Table 4.2: Value ranges for the next iteration of the haptic feedback program. Source: self.

4.2.2. Vibration limitation test

Intense vibration can be uncomfortable or outrightly harmful to the user. In order to ensure that the user is safe, a test to detect the maximum vibration speed with which the user is comfortable will be performed.

The test was performed by applying direct contact between the motor and the residual limb of the pilot and using the knob sketch to gradually increase the intensity, until the maximum vibration of the motor was reached. The pilot expressed no discomfort during the whole process, but expressed his desire to have punctual vibration when pressure was detected (similar to a warning) instead of constant vibration. Additionally, the user expressed no discomfort due to the acoustic effects of the vibration, thanks to the insulation provided by the socket.

Following his desires, the test will be modified to have a minimum required variation in the force detected to trigger a new vibration.

5.Budget and planification

The below tables indicate the time invested in the project and its total budget. It’s important to remark that this only covers the improvements that will be implemented upon Andromeda, and not the whole process of design and creation of the prosthesis.

Activity	**Time (hours)**	**Cost (€/hour)**	**Final cost (€)**
Investigation and formation	150	40	6.000
Market studies	40	40	1.600
Team meetings	30	40	1.200
Coding	20	40	800
Testing	20	40	800
Writing the report	60	40	2.400
Total	320	40	12.800

Table 5.1: Cost of the activities performed during this project. Source: self.

Product	**Provider**	**Price (€/unit)**	**Units**	**Total (€)**
ESP32-DevKitC-32D	Mouser	9’10	1	9’10
Flexible pressure sensor DF9-40 20Kg	Amazon	12’69	1	12’69
Vibration motor ROB-08449	Sparkfun	2’12	1	2’12
Total				23’87

Table 5.2: Price of the components bought for the project. Source: self.

ARM2u buys components for the prosthesis in trimestral orders. The components used have been considered a part of those orders, and therefore shipping prices are not included. It is also necessary to remark that most of the components used for the testing circuits (resistors, potentiometers, wires, servo motors, battery, breadboard, gripper…) are already available to the team, and therefore weren’t considered part of the cost.

6.Environmental and social impact

Improvement of the performance of low cost myoelectric prosthetic arms can have a drastic effect in the quality of life of millions of people. In 2008 it was estimated that 3 million people suffered from arm amputation worldwide, with 2.4 million being from developing countries with scarce access to the medical, technological and economical resources necessary to obtain those prosthesis [40]. We can’t forget that this problem even affects people from birth: Only in the U.S., an average of 1900 babies are born with an upper or lower limb difference each year [41].

People who suffer a limb loss don’t face only physical challenges, but also psychological ones. Depression, trauma and anxiety, among others, are common among amputees. The Liner Ward Amputee Community survey published in April 2022 got inputs from more than 650 participants. Between 35 and 45% of them, depending on their age range, often faced mental-health challenges [42].

Increasing both the affordability and functionality of their available models could help them face those issues, increasing their sense of worth and their opportunities to return to an active lifestyle in which they can perform the same activities they could do before their amputation (or, in the case of congenital amputees, allowing them to perform those activities for the first time in their lives). This in return allows them to take a more active role in society, augmenting their employability and reducing the economical load on healthcare systems.

The use of 3D printing technologies to create those devices is also a significant improvement both in price and in environmental impact. With 3D printers being steadily more available, the possibility of having the necessary materials to build the prosthesis already at hand significantly decreases the carbon footprint of the product, with only electrical components needing transportation. As mentioned in 2.1, the incoming Venus arm by eBionics uses this to reduce their price below 240€.

7.Conclusions

The world of prosthetics is filled with opportunities for improvement and innovation. In the past, high quality prostheses were prohibitive to the majority of the population, and although there is still a long way to go, initiatives to make them more affordable to everyone are being started by students all across the globe. Is in this growing sector where ARM2u finds its reason to be.

Despite their prices and advanced functionalities, high end prostheses are still failing to live up to their user's expectations. Research and development of new methods to increase the functionality of the prosthetic limbs and the integration with the user are being studied, but the literature on testing or training processes is usually scarce, leading to incomplete tests that overlook key aspects of the daily use of the prosthesis and ineffective training programs.

Nevertheless, significant progress is still being made, and new models offer improvements that prostheses users have been demanding for a long time. This project lays the foundations for the implementation of two of those improvements in Andromeda: An improved control mode, and sensory feedback. Those additions enhance the experience of the user, increasing their comfort and performance. Since prosthesis are devices that the user is expected to use daily during extended periods of time, these upgrades can have a major impact on their lives, increasing the number of activities they can perform and making them feel comfortable while doing so.

The proposed improvements also increase the value of Andromeda, making the project more attractive to organizations seeking a partnership with ARM2u and students interested in joining the team. For this reason, these systems will be implemented and exposed at ARM2u's stand at Forum ETSEIB.

Thanks

To my mother, for always being there for me.

To the ARM2u team, who has given me countless opportunities to experience the world of biomedical engineering.

To Dani, for her constant support and guidance through the whole project.

Bibliography (MLA format)

1. Vujaklija, Ivan et al. “New developments in prosthetic arm systems.” *Orthopedic research and reviews* vol. 8 31-39. 7 Jul. 2016, doi:10.2147/ORR.S71468
2. "Bionic Hand Price List". *Bionics For Everyone*, 2022, https://bionicsforeveryone.com/bionic-hand-price-list/.
3. Vujaklija, Ivan et al. “New developments in prosthetic arm systems.” *Orthopedic research and reviews* vol. 8 31-39. 7 Jul. 2016, doi:10.2147/ORR.S71468
4. "Arm2u – ETSEIB – Grasp The World!". *Arm2u-Etseib.Upc.Edu*, 2022, https://arm2u-etseib.upc.edu/wp/.
5. Peerdeman, Bart et al. “Myoelectric forearm prostheses: state of the art from a user-centered perspective.” *Journal of rehabilitation research and development* vol. 48,6 (2011): 719-37. doi:10.1682/jrrd.2010.08.0161
6. B. Rodríguez-Tapia, I. Soto, D. M. Martínez and N. C. Arballo, "Myoelectric Interfaces and Related Applications: Current State of EMG Signal Processing–A Systematic Review," *IEEE Access*, vol. 8, pp. 7792-7805, 2020, doi: 10.1109/ACCESS.2019.2963881.
7. M. A. Cavalcanti Garcia & Vieira, T. M. M. “Surface electromyography: Why, when and how to use it“ Revista andaluza de medicina del deporte. In *Revista Andaluza de Medicina del Deporte* Vol. 4, Issue 1. Elsevier.
8. "Current Options For Bionic Hands". *Bionics For Everyone*, 2022, https://bionicsforeveryone.com/current-options-for-bionic-hands/.
9. "Ottobock Michelangelo Hand". *Bionics For Everyone*, 2022, https://bionicsforeveryone.com/ottobock-michelangelo-hand/#lift-capacity-strength.
10. "FAQ - Ebionics". *Ebionics*, 2022, https://ebionics.org/faq/.
11. "LUKE Arm Detail Page – Mobius Bionics". *Mobiusbionics.Com*, 2022, https://www.mobiusbionics.com/luke-arm/#section-three.
12. Yamamoto, Michiro MD, PhD; Chung, Kevin C. MD, MS; Sterbenz, Jennifer BS; Shauver, Melissa J. MPH; Tanaka, Hirotaka MD, PhD; Nakamura, Takashi PO; Oba, Jumpei OTR; Chin, Takaaki MD, PhD‖, Hirata, Hitoshi MD, PhD “Cross-sectional International Multicenter Study on Quality of Life and Reasons for Abandonment of Upper Limb Prostheses”, *Plastic and Reconstructive Surgery - Global Open*: May 2019 - Volume 7 - Issue 5 - p e2205

doi: 10.1097/GOX.0000000000002205

13. Haverkate, Liz, et al. “Assessment of Body-Powered Upper Limb Prostheses by Able-Bodied Subjects, Using the Box and Blocks Test and the Nine-Hole Peg Test.” *Prosthetics and Orthotics International*, vol. 40, no. 1, Feb. 2016, pp. 109–116, doi:10.1177/0309364614554030.
14. Peerdeman, Bart et al. “Myoelectric forearm prostheses: state of the art from a user-centered perspective.” *Journal of rehabilitation research and development* vol. 48,6 (2011): 719-37. doi:10.1682/jrrd.2010.08.0161
15. Latour, Debra OTD, MEd, OTR/L; Passero, Thomas CP “Assessment of Functionality of Multifunction Prosthetic Hands”, *Journal of Prosthetics and Orthotics*: July 2019 - Volume 31 - Issue 3 - p 165-166 doi: 10.1097/JPO.0000000000000269
16. Widehammar, Cathrine; Lidström Holmqvist, Kajsa; Hermansson, Liselotte “Training for users of myoelectric multigrip hand prostheses: a scoping review”, *Prosthetics and Orthotics Internationa*l: October 2021 - Volume 45 - Issue 5 - p 393-400
doi: 10.1097/PXR.0000000000000037
17. Pylatiuk, Christian MD; Kargov, Artem MEng; Schulz, Stefan MEng “Design and Evaluation of a Low-Cost Force Feedback System for Myoelectric Prosthetic Hands”, *JPO Journal of Prosthetics and Orthotics*: April 2006 - Volume 18 - Issue 2 - p 57-61
18. Björkman, Anders et al. “Sensory qualities of the phantom hand map in the residual forearm of amputees.” *Journal of rehabilitation medicine* vol. 48,4 (2016): 365-70. doi:10.2340/16501977-2074
19. Antfolk, Christian & D'Alonzo, Marco & Controzzi, Marco & Lundborg, Göran & Rosén, Birgitta & Sebelius, Fredrik & Cipriani, Christian. “Artificial Redirection of Sensation From Prosthetic Fingers to the Phantom Hand Map on Transradial Amputees: Vibrotactile Versus Mechanotactile Sensory Feedback”. *IEEE transactions on neural systems and rehabilitation engineering : a publication of the IEEE Engineering in Medicine and Biology Society*. 21. 10.1109/TNSRE.2012.2217989.
20. "Even Non-Amputees Can Feel A Phantom Limb". *Livescience.Com*, 2022, https://www.livescience.com/28694-non-amputees-feel-phantom-limb.html.
21. "Boards". *Arduino Official Store*, 2022, http://store.arduino.cc/collections/boards?view=view-48&grid_list=list-view.
22. "Teensy® 4.1". *Pjrc.Com*, 2022, https://www.pjrc.com/store/teensy41.html.
23. "Photon Datasheet | Datasheets". *Docs.Particle.Io*, 2022, https://docs.particle.io/datasheets/wi-fi/photon-datasheet/.

24. "Seeeduino Nano". *Seeedstudio.Com*, 2022, https://www.seeedstudio.com/Seeeduino-Nano-p-4111.html.
25. "ESP32-Devkitc V4 Getting Started Guide - ESP32 - — ESP-IDF Programming Guide Latest Documentation". *Docs.Espressif.Com*, 2022, https://docs.espressif.com/projects/esp-idf/en/latest/esp32/hw-reference/esp32/get-started-devkitc.html.
26. "Placa Nodemcu V3 - ESP8266 / CH340G Wifi Bricogeek CH340-V3 | Bricogeek.Com". *Tienda.Bricogeek.Com*, 2022, https://tienda.bricogeek.com/wifi/1033-nodemcu-v3-wifi-esp8266-ch340.html.
27. "Sparkfun Thing Plus - ESP32 WROOM Sparkfun WRL-14689 | Bricogeek.Com". *Tienda.Bricogeek.Com*, 2022, https://tienda.bricogeek.com/placas-adafruit-feather/1298-sparkfun-thing-plus-esp32-wroom.html.
28. "Getting Started With The ESP32 - Using The Arduino IDE". *Dronebot Workshop*, 2022, https://dronebotworkshop.com/esp32-intro/.
29. "Using servo motors with the ESP32". *Dronebot Workshop*, 2022, https://dronebotworkshop.com/esp32-servo/.
30. Fahad, Engr. "What Is A Voltage Divider Or Potential Divider, Formula And Practical Uses?". *Electronic Clinic*, 2022, https://www.electroniclinic.com/what-is-a-voltage-divider-or-potential-divider-formula-and-practical-uses/.
31. Wijk, Ulrika et al. “Sensory Feedback in Hand Prostheses: A Prospective Study of Everyday Use.” *Frontiers in neuroscience* vol. 14 663. 7 Jul. 2020, doi:10.3389/fnins.2020.00663
32. C. Cipriani, F. Zaccone, S. Micera and M. C. Carrozza, "On the Shared Control of an EMG-Controlled Prosthetic Hand: Analysis of User–Prosthesis Interaction," in *IEEE Transactions on Robotics*, vol. 24, no. 1, pp. 170-184, Feb. 2008, doi: 10.1109/TRO.2007.910708.
33. Akhtar, A., J. Cornman, J. Austin, and D. Bala. “Touch Feedback and Contact Reflexes Using the PSYONIC Ability Hand”. *MEC20 Symposium*, July 2020, https://conferences.lib.unb.ca/index.php/mec/article/view/26.
34. "Mehand From Maxbionic". *Bionics For Everyone*, 2022, https://bionicsforeveryone.com/maxbionic-mehand/#sensory-feedback.

35. "Vincent Evolution". *Bionics For Everyone*, 2022, https://bionicsforeveryone.com/vincent-evolution/
36. "Pressure Measurement Sensor | How It Works | FUTEK". *Futek.com*, 2022, https://www.futek.com/pressure-measurement.
37. Stegink Jansen C.W., Kocian Simper V., Stuart H.G., and Pinkerton H.M. “Measurement of Maximum Voluntary Pinch Strength: Effects of Forearm Position and Outcome Score.” Journal of Hand Therapy 16: 326-336, 2003.
38. "SENSOR FLEX DF9-40 10KG RESISTIVO – Grupo Electrostore". *Grupoelectrostore.com*, 2022, https://grupoelectrostore.com/shop/sensores/flex-resistivos/sensor-flex-df9-40-10kg-resistivo/.
39. “ROB-08449 SparkFun | Mouser España”. Mouser.es, 2022, https://www.mouser.es/ProductDetail/SparkFun/ROB-08449?qs=WyAARYrbSnZj6jJCAHtRrQ%3D%3D&mgh=1&vip=1&gclid=CjwKCAjwsJ6TBhAIEiwAfl4TWDivh3OgK65nYI_W-hG06A5jPhBTWyDxTrvInhej_JwFYZR_fm4O1RoCQMwQAvD_BwE
40. Maurice LeBlanc. “Give hope - Give a hand - The LN4 prosthetic hand”. Stanford university, 2008. https://web.stanford.edu/class/engr110/2011/LeBlanc-03a.pdf
41. "Hanger Clinic Blog | Limb Loss And Limb Difference: Facts, Statistics, & Resources". *Hanger Clinic*, 2022, https://hangerclinic.com/blog/prosthetics/limb-loss-and-limb-difference-facts-statistics-resources/.
42. “Amputee Community survey 2022: The aging curve”. *Amplitude*, 2022, https://livingwithamplitude.com/amputee-statistics-community-survey-2022-age-curve/