

**UNIDAD EDUCATIVA TÉCNICO SALESIANO**

**BRAZO ROBÓTICO ARTICULADO PARA LA MANIPULACIÓN DE OBJETOS DE USO COTIDIANO EN CASAS DE SALUD**

**Para optar al Título de Bachiller de la República del Ecuador, con mención en Aplicaciones Informáticas**

**Autores:**

**Daniel Sebastián Arévalo Arízaga**

**Jonathan Javier Ortega Molina**

**Mateo Sebastián Ortiz Rodas**

**Erick Johann Ureña Torres**

**Tutor:**

**Teodoro Germán Morales Ulloa**

**CUENCA – ECUADOR**

**2015**

# 

# CERTIFICACION

Yo, Mg. Teodoro Morales U., en calidad de tutor del proyecto de grado, Brazo robótico articulado para la manipulación de objetos de uso cotidiano en casas de salud, presentado por los estudiantes Daniel Sebastián Arévalo Arízaga, Jonathan Javier Ortega Molina, Mateo Sebastián Ortiz Rodas y Erick Johann Ureña Torres; certifico el nivel de independencia y creatividad así como la disciplina en el cumplimiento de dicho trabajo. Por lo tanto, por cumplir con los requisitos establecidos autorizo su presentación.

Cuenca, 25 de mayo del 2015

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Mg. Teodoro Morales U.

# PÁGINA DE APROBACIÓN DEL JURADO EXAMINADOR

BRAZO ROBÓTICO ARTICULADO PARA LA MANIPULACIÓN DE OBJETOS DE USO COTIDIANO EN CASAS DE SALUD

Por:

Daniel Sebastián Arévalo Arízaga

Jonathan Javier Ortega Molina

Mateo Sebastián Ortiz Rodas

Erick Johann Ureña Torres

Proyecto de grado aprobado en nombre de la Unidad Educativa Técnico Salesiano, por el siguiente jurado, a los 25 días del mes de mayo de 2015.

|  |  |  |
| --- | --- | --- |
| **Firma** |  | **Firma** |
| **Nombre**…………………………………. | **Nombre**…………………………………. |

|  |
| --- |
| **Firma** |
| **Nombre**…………………………………. |

# DECLARACIÓN DE RESPONSABILIDAD

Los autores declaran que el Proyecto de grado presentado, tanto en su parte práctica como la teórica, es de su autoría; y que han consultado las referencias bibliográficas que se incluyen en este documento.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Daniel Sebastián Arévalo Arízaga

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Jonathan Javier Ortega Molina

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Mateo Sebastián Ortiz Rodas

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Erick Johann Ureña Torres

# DEDICATORIA

Quisiera dedicar todo el esfuerzo y trabajo que significó realizar este proyecto de grado a aquellas personas que a pesar de todos los problemas estuvieron a mi lado. Mis hermanos principalmente por nunca alejarse de mí, sin importar que yo haya sido distante con ellos; a mi familia por ayudarme y apoyarme con todas las decisiones que he tomado; a mis compañeros de trabajo que supieron entender la falta de tiempo con el mismo y finalmente a mis amigos que estuvieron conmigo a pesar de haberles fallado y que sé que estarán ahí cuando yo ya no esté a su lado, quiero que sepan que siempre ocuparán un lugar especial dentro de mi

**Daniel Sebastián Arévalo Arízaga**

Dedico este proyecto principalmente a Dios, por permitirme llegar hasta este punto circunstancial de mi vida y haberme brindado la salud necesaria para cumplir con todos mis objetivos planteados. A mis padres, por los innumerables consejos y su apoyo tanto moral como económico que me brindaron. A mis hermanos, quienes me ayudaron a sacarme una sonrisa en momentos de estrés y tensión. A mi novia Samantha, por todo el amor y paciencia que me dio, por ser mi fiel amiga y compañera y por todo el apoyo emocional que me regalo en el transcurso de este proyecto de grado.

**Jonathan Javier Ortega Molina**

Este proyecto lo dedico a mi familia, que fueron las personas que se preocuparon de mi desde el momento que llegué a este mundo, supieron guiarme, formarme, enseñarme a salir adelante, luchando para salir victorioso de cada adversidad que se ha presentado en mi vida, a las personas que supieron darme una mano en el momento que más lo necesité para poder superar cualquier tropiezo, que supieron darme su tiempo cuando ellos no lo tenían. Las personas de las cuales siempre me he sentido maravillado de que puedan estar a mi lado, a ellas les dedico ésta tesis, a mi familia.

**Mateo Sebastián Ortiz Rodas**

Dedico mi sueño hecho realidad a todas personas que a lo largo de mi vida colegial me brindaron su apoyo incondicional y supieron ganarse mi sincera amistad. Con el mismo sentimiento de gratitud agradezco a mi madre Kety Torres y mis hermanos: Diego y Bryan por haber tenido consideración hacia mi persona en los momentos más difíciles de este camino, que hoy puedo decir que lo he acabado con gran satisfacción.

**Erick Johann Ureña Torres**

# AGRADECIMIENTO

Agradezco a mis compañeros del proyecto de grado, por haberme tenido paciencia y haber entendido los contratiempos que tuve al momento de realizar este proyecto; a mi familia que nunca se opuso a prestarme su ayuda y apoyo; a mi maestro y amigo James Gala, que siempre supo guiarme y enseñarme para que nunca tenga ningún problema en mi vida, y a todas las personas que me ayudaban a levantarme y sonreír cada vez que había sufrido una caída.

**Daniel Sebastián Arévalo Arízaga**

A mis padres ya que gracias a todo su amor, cariño, paciencia, consejos y sobre todo su amistad hoy en día estoy culminando mis estudios en el colegio. A mi hermana Jennifer, quien gracias a su humor pudo sacarme una sonrisa en significantes momentos el transcurso de este año. A mi novia Samantha, quien me ayudo a sobresalir en momentos de dificultad en mi vida con todo su amor y cariño, logrando así que nunca me rinda y poder culminar mi proyecto de grado.

**Jonathan Javier Ortega Molina**

A mis padres, hermanas, que han estado conmigo desde que era un risueño niño, que siempre han estado detrás de mi guiándome para seguir el camino del éxito y conseguir un triunfo más, que les pertenece a ustedes también, a mis amigos, que siempre estuvieron animándome, brindándome un motivo más por el cual sonreír, a mis compañeros de tesis, que sin ellos este proyecto de grado no hubiese podido salir adelante, a todas estas personas quiero agradecerles por todo, no me alcanzan las palabras para expresar el orgullo y lo bien que me siento por tener unas personas tan maravillosas a mi lado.

**Mateo Sebastián Ortiz Rodas**

Me permito agradecer a todas y cada una de las personas que me han acompañado en el transcurso de mis estudios, a mi Madre y Padre, por guiar mi camino y permitirme ser la persona que ahora soy, mis hermanos, quienes con sus constantes e innumerables consejos me han apoyado en todo momento, amigos y familiares por aquella agradable presencia y tiempo compartido. Gracias por acompañar y guiar mi camino.

**Erick Johann Ureña Torres**

# RESUMEN

El proyecto consta de un brazo robótico articulado cuya finalidad es servir de apoyo a personas que tienen dificultades de salud, brindándoles la posibilidad de alcanzar objetos que se encuentran a cierta distancia o poder con este alimentarse a sí mismos.

El proyecto está realizado con servomotores controlados desde un celular y ensamblado en materiales resistentes para ser implementado en casas de salud. La construcción del software se realizó en el IDE App Inventor que usa el lenguaje de programación JAVA. En este se realizó la interfaz que controla el brazo, el cual será mediante sliders, botones y el acelerómetro.

Su parte física consta de servomotores para el movimiento del brazo y de polimetilmetacrilato para la estructura general del mismo, con una base y hombro de inox (acero inoxidable). La unión de las diferentes partes se llevó a cabo con tuercas y pernos.

# ÍNDICE DE CONTENIDOS

[CERTIFICACION II](#_Toc420310328)

[PÁGINA DE APROBACIÓN DEL JURADO EXAMINADOR III](#_Toc420310329)

[DECLARACIÓN DE RESPONSABILIDAD IV](#_Toc420310330)

[DEDICATORIA V](#_Toc420310331)

[AGRADECIMIENTO X](#_Toc420310332)

[RESUMEN XV](#_Toc420310333)

[ÍNDICE DE CONTENIDOS XVI](#_Toc420310334)

[ÍNDICE DE IMÁGENES, FIGURAS Y ANEXOS XX](#_Toc420310335)

[**INTRODUCCIÓN 1**](#_Toc420310336)

[**CAPITULO I 2**](#_Toc420310337)

[1. MARCO TEÓRICO 2](#_Toc420310338)

[1.1. Antecedentes 2](#_Toc420310339)

[1.2. Justificación 2](#_Toc420310340)

[1.3. Objetivos 3](#_Toc420310341)

[1.3.1. Objetivo General 3](#_Toc420310342)

[1.3.2. Objetivos Específicos 3](#_Toc420310343)

[1.4. Fundamentación Teórica 4](#_Toc420310344)

[1.4.1. Servomotores 4](#_Toc420310345)

[1.4.1.1. Motor 4](#_Toc420310346)

[1.4.1.2. Caja Reductora 5](#_Toc420310347)

[1.4.1.3. Circuito de Control (AA51880) 7](#_Toc420310348)

[1.4.1.4. Tipos de Servomotores 9](#_Toc420310349)

[1.4.2. Polimetilmetacrilato (Acrílico) 10](#_Toc420310350)

[1.4.3. Placa Arduino 12](#_Toc420310351)

[1.4.3.1. Sintaxis en Arduino 13](#_Toc420310352)

[1.4.3.1.1. setup() 13](#_Toc420310353)

[1.4.3.1.2. pinMode (pin, mode) 13](#_Toc420310354)

[1.4.3.1.3. Serial.begin(speed) 13](#_Toc420310355)

[1.4.3.1.4. loop() 14](#_Toc420310356)

[1.4.3.1.5. digitalWrite(pin, value) 14](#_Toc420310357)

[1.4.3.1.6. digitalRead(pin) 14](#_Toc420310358)

[1.4.3.1.7. analogWrite(pin, value) 14](#_Toc420310359)

[1.4.3.1.8. analogRead(pin) 14](#_Toc420310360)

[1.4.3.1.9. delay(ms) 14](#_Toc420310361)

[1.4.3.1.10. map (value, fromLow, fromHigh, toLow, toHigh) 15](#_Toc420310362)

[1.4.3.1.11. Serial.read() 15](#_Toc420310363)

[1.4.3.1.12. Serial.print(val) 15](#_Toc420310364)

[1.4.3.1.13. Serial.println(val) 15](#_Toc420310365)

[1.4.4. Resistencias 16](#_Toc420310366)

[1.4.5. Modulo Bluetooth HC-06 17](#_Toc420310367)

[1.4.6. Sistema Operativo Android 19](#_Toc420310368)

[1.4.7. App Inventor 19](#_Toc420310369)

[1.4.8. Acelerómetro 21](#_Toc420310370)

[1.4.9. Sensor de Humedad y Temperatura 24](#_Toc420310371)

[1.4.10. Diodos Emisores de Luz (LED) 26](#_Toc420310372)

[1.4.10.1. Funcionamiento 26](#_Toc420310373)

[1.4.10.2. Partes físicas del LED 28](#_Toc420310374)

[1.4.11. LED RGB 28](#_Toc420310375)

[1.4.11.1. Funcionamiento RGB 29](#_Toc420310376)

[1.4.11.2. Tabla de colores principales 29](#_Toc420310377)

[**CAPÍTULO II 31**](#_Toc420310378)

[2. ANÁLISIS Y DISEÑO 31](#_Toc420310379)

[2.1. Planificación del Proyecto 31](#_Toc420310380)

[2.1.1. Estudio de factibilidades 31](#_Toc420310381)

[2.1.1.1. Factibilidad Operacional 31](#_Toc420310382)

[2.1.1.2. Factibilidad Técnica 31](#_Toc420310383)

[2.1.1.3. Factibilidad Económica 31](#_Toc420310384)

[2.1.2. Recursos 32](#_Toc420310385)

[2.1.2.1. Recursos de Hardware 32](#_Toc420310386)

[2.1.2.2. Recursos de Software 32](#_Toc420310387)

[2.1.2.2.1. Herramientas para la investigación y documento monográfico 32](#_Toc420310388)

[2.1.2.2.2. Herramientas para el desarrollo de software 33](#_Toc420310389)

[2.1.3. Estimaciones 33](#_Toc420310390)

[2.1.3.1. Estimaciones de tiempo 33](#_Toc420310391)

[2.1.3.2. Estimaciones de costo 34](#_Toc420310392)

[2.2. Definición y Alcances del Proyecto 36](#_Toc420310393)

[2.3. Recolección y Análisis de Requisitos 36](#_Toc420310394)

[2.4. Diseño de Software (Ciclo de vida del proyecto) 37](#_Toc420310395)

[2.4.1. Análisis de requisitos 37](#_Toc420310396)

[2.4.2. Diseño del Sistema 37](#_Toc420310397)

[2.4.3. Diseño del Programa 39](#_Toc420310398)

[2.4.3.1. Diseño AppInventor 39](#_Toc420310399)

[2.4.3.2. Diseño Arduino 40](#_Toc420310400)

[2.4.4. Codificación 41](#_Toc420310401)

[2.4.4.1. Codificación en Arduino 41](#_Toc420310402)

[2.4.4.2. Codificación AppInventor 41](#_Toc420310403)

[2.4.5. Diseño Electrónico 41](#_Toc420310404)

[**CAPÍTULO III 43**](#_Toc420310405)

[3. CONSTRUCCIÓN Y PRUEBAS 43](#_Toc420310406)

[3.1. Construcción y Ensamblaje del Hardware 43](#_Toc420310407)

[3.2. Implementación del Sistema 46](#_Toc420310408)

[3.2.1. Envío de datos desde el Hardware 46](#_Toc420310409)

[3.2.2. Recepción de datos desde el Hardware 46](#_Toc420310410)

[3.3. Pruebas 47](#_Toc420310411)

[3.4. Manual de usuario 51](#_Toc420310412)

[**CONCLUSIONES 53**](#_Toc420310413)

[**RECOMENDACIONES 54**](#_Toc420310414)

[**REFERENCIAS BIBLIOGRÁFICAS 55**](#_Toc420310415)

[**ANEXOS 56**](#_Toc420310416)

# ÍNDICE DE IMÁGENES, FIGURAS Y ANEXOS

[Figura 1.1: Motor 4](#_Toc420238505)

[Figura 1.2: Esquema del funcionamiento básico de un motor 5](#_Toc420238506)

[Figura 1.3: Caja reductora 6](#_Toc420238507)

[Figura 1.4: Etapas de reducción de caja reductora 6](#_Toc420238508)

[Figura 1.5: Funcionamiento chip integrado AA51880 8](#_Toc420238509)

[Figura 1.6: Servomotor 10](#_Toc420238510)

[Figura 1.7: Formula esquelética del PMMA (Polimetilmetacrilato) 11](#_Toc420238511)

[Figura 1.8: Placa Arduino UNO y sus partes 12](#_Toc420238512)

[Figura 1.9: Código de colores de resistencia 17](#_Toc420238513)

[Figura 1.10: Módulo bluetooth HC-06 18](#_Toc420238514)

[Figura 1.11: Página de inicio de App Inventor 20](#_Toc420238515)

[Figura 1.12: Página de diseño de aplicaciones App Inventor 21](#_Toc420238516)

[Figura 1.13: Eje cartesiano del funcionamiento acelerómetro 22](#_Toc420238517)

[Figura 1.14: Microchip acelerómetro 22](#_Toc420238518)

[Figura 1.15: Funcionamiento interno acelerómetro 23](#_Toc420238519)

[Figura 1.16: Aplicación en App Inventor para control de acelerómetro 23](#_Toc420238520)

[Figura 1.17: Temperatura vs Tiempo (Sensores DHT11 y DHT22) 24](#_Toc420238521)

[Figura 1.18: Humedad vs Tiempo (Sensores DHT11 Y DHT22) 24](#_Toc420238522)

[Figura 1.19: DHT11 Y DHT22 25](#_Toc420238523)

[Tabla 1.1: Ventajas y Desventajas sensores de humedad 25](#_Toc420238524)

[Tabla 1.2: Especificación técnica sensores de humedad 25](#_Toc420238525)

[Figura 1.20: Diodo Emisor de Luz 26](#_Toc420238526)

[Figura 1.21: Funcionamiento LED 26](#_Toc420238527)

[Figura 1.22: Intensidad vs Voltaje LED 28](#_Toc420238528)

[Figura 1.23: Partes físicas LED 28](file:///D:\Tipitin\Deberes\3°%20de%20Bachillerato\Tesis\MONOGRAFIA.docx#_Toc420238529)

[Figura 1.24: LED RGB 29](#_Toc420238530)

[Tabla 1.3: Cuadro de colores LED RGB 30](#_Toc420238531)

[Tabla 2.1: Estimaciones de tiempo en el aspecto práctico 33](#_Toc420238532)

[Tabla 2.2: Estimaciones de tiempo en el aspecto teórico 34](#_Toc420238533)

[Tabla 2.3: Estimaciones de Costo 35](#_Toc420238534)

[Figura 2.1: Diseño programa AppInventor 39](#_Toc420238535)

[Figura 2.2: Diseño programa Arduino 40](#_Toc420238536)

[Figura 2.3: Diseño electrónico del brazo robótico 42](#_Toc420238537)

[Figura 3.1: Servomotor movimiento hombro 43](#_Toc420238538)

[Figura 3.2: Servomotor movimiento codo 44](#_Toc420238539)

[Figura 3.3: Servomotor movimiento vertical muñeca 44](#_Toc420238540)

[Figura 3.4: Servomotor movimiento rotacional muñeca. 45](#_Toc420238541)

[Figura 3.5: Servomotor apertura pinza 45](#_Toc420238542)

[Figura 3.6: Base de inox 46](#_Toc420238543)

[Figura 3.7: Brazo robótico no energizado 47](#_Toc420238544)

[Figura 3.8: Brazo robótico energizado 47](#_Toc420238545)

[Figura 3.9: Movimiento del hombro del brazo robótico 48](#_Toc420238546)

[Figura 3.10: Moviento de codo del brazo robótico 48](#_Toc420238547)

[Figura 3.11: Movimiento de la muñeca verticalmente del brazo robótico 49](#_Toc420238548)

[Figura 3.12: Movimiento de la muñeca horizontalmente del brazo robótico 49](#_Toc420238549)

[Figura 3.13: Rotación de la base del brazo robótico 50](#_Toc420238550)

[Figura 3.14: Rotación de la base del brazo robótico 50](#_Toc420238551)

[Figura 3.15: Apertura de la pinza del brazo robótico 51](#_Toc420238552)

[Figura 3.16: Manual de Usuario 51](#_Toc420238553)

[Anexo 1: Codificación Arduino 56](#_Toc420238554)

[Anexo 2: Codificación Arduino 56](#_Toc420238555)

[Anexo 3: Codificación Arduino 57](#_Toc420238556)

[Anexo 4: Codificación Arduino 57](#_Toc420238557)

[Anexo 5: Codificación Arduino 58](#_Toc420238558)

[Anexo 6: Codificación Arduino 58](#_Toc420238559)

[Anexo 7: Codificación Arduino 59](#_Toc420238560)

[Anexo 8: Codificación Arduino 59](#_Toc420238561)

[Anexo 9: Codificación AppInventor 59](#_Toc420238562)

[Anexo 10: Codificación AppInventor 60](#_Toc420238563)

[Anexo 11: Codificación AppInventor 60](#_Toc420238564)

[Anexo 12: Diseño pieza antebrazo 61](#_Toc420238565)

[Anexo 13: Diseño pieza brazo 61](#_Toc420238566)

[Anexo 14: Diseño pieza muñeca horizontal 62](#_Toc420238567)

[Anexo 15: Diseño pieza muñeca vertical 62](#_Toc420238568)

[Anexo 16: Diseño pieza hombro 62](#_Toc420238569)

[Anexo 17: Diseño pieza base de la pinza 63](#_Toc420238570)

[Anexo 18: Diseño piezas engranes de la pinza 63](#_Toc420238571)

[Anexo 19: Diseño piezas puntas de la pinza 64](#_Toc420238572)

[Anexo 20: Diseño pinza completa 64](#_Toc420238573)

# INTRODUCCIÓN

Hoy en día la tecnología es una de las herramientas más utilizadas cuando se trata de mejorar la calidad de vida de las personas, por lo tanto en el presente proyecto se ha aprovechado esta tecnología para programar y ensamblar un brazo robótico articulado realizado con servomotores para ayudar a mejorar la calidad de vida de las personas que no se encuentran en un buen estado de salud, permitiéndoles alcanzar objetos que se encuentran a determinada distancia o alimentarse a sí mismos sin la necesidad de ponerse de pie o realizar un gran esfuerzo físico. Para finalizar con éxito este proyecto se tuvo que partir desde el análisis de requisitos en casas de salud, continuando con la respectiva investigación de información y por ultimo aplicar los conocimientos obtenidos para la construcción del proyecto. De esto se obtuvo un prototipo de un brazo robótico funcional destinado a servir a personas con problemas de salud para mejorar su calidad de vida.

# CAPITULO I

# MARCO TEÓRICO

# Antecedentes

Un brazo robótico es un mecanismo articulado, el cual cumple funciones previamente programadas, resultando muy parecido a un brazo humano. Las articulaciones y grados de libertad de estos brazos pueden ser muy variadas, por lo cual buscaremos un diseño que se acomode mejor a nuestras necesidades, basándonos en nuestra problemática.

# Justificación

En la actualidad existen personas las cuales tienen limitaciones físicas ya sea de forma temporal o permanente, lo cual los vuelve incapaces de movilizarse y manipular objetos de uso cotidiano. El proyecto a realizar servirá principalmente para brindar ayuda a una persona por medio de un brazo mecánico para trabajos que no se puedan realizar dentro de la posibilidad del usuario o que estén fuera de su alcance. Su aplicación se dará principalmente en habitaciones de hospitales, o en el domicilio de las personas que no puedan realizar determinadas actividades debido a su condición médica. La elaboración de este proyecto, será realizada por los estudiantes autores del mismo, y contará con la guía y apoyo de un tutor asignado por la institución. Este proyecto vinculará al usuario con el brazo robótico utilizando tecnología común y de fácil aplicación por tratarse de un dispositivo que maneje el sistema operativo Android, y una tarjeta Arduino UNO.

Se utilizará el acelerómetro del celular para una interacción óptima entre usuario y máquina. El dispositivo Android mediante comunicación bluetooth, enviará señales al Arduino para el control del brazo robótico.

El dispositivo que se diseñará será viable, puesto que su utilización simplificará mucho la realización de ciertas actividades que no sean realizables por el usuario. Esto proporcionará un beneficio mayor en el lugar en donde será utilizado como en un hospital o clínica en la que se encuentre la persona, ya que no se necesitará personal para realizar actividades simples tales como, movilizar un vaso de agua hacia el usuario, o poder alcanzar algún medicamento que sea necesario para el mismo. Además de esto, el hospital ganará mayor prestigio por este servicio adicional que ofrece a sus clientes, y por la tecnología que se utiliza.

El brazo robótico es factible económicamente, puesto que los materiales necesarios para su elaboración son de bajos costos, y la tecnología que utiliza es de fácil aplicación; además de esto, la programación y construcción de los circuitos no conllevan mucho tiempo, por lo cual se podrá finalizar el proyecto en un plazo de aproximadamente 6 meses.

# Objetivos

# Objetivo General

Diseñar y ensamblar un brazo robótico utilizando el acelerómetro de un dispositivo con sistema operativo Android y una tarjeta Arduino, el cual servirá para reemplazar al personal médico en la realización de actividades sencillas.

# Objetivos Específicos

* Diseñar un brazo robótico, utilizando servomotores.
* Determinar la clasificación, funcionamiento, características y aplicación de servomotores que serán utilizados en el proyecto.
* Ensamblar la estructura del brazo robótico utilizando materiales resistentes, y servomotores para su movimiento.
* Realizar la aplicación en Android que será utilizada para el brazo robótico.
* Realizar la programación necesaria para el control del brazo robótico en Arduino.
* Manipular el brazo robótico mediante el acelerómetro de un celular para controlar los movimientos del mismo.
* Implementar el dispositivo realizado en instituciones las cuales brindan servicio médico.

# Fundamentación Teórica

# Servomotores

Los Servomotores, son dispositivos electrónicos cuya finalidad es convertir la energía eléctrica en energía motriz. Su principio de funcionamiento se basa en el del motor eléctrico, con la diferencia de que este es capaz de mantener una posición estática según el voltaje o tensión que reciba. “Los servomotores están constituidos por el motor, una caja reductora y el circuito de control. Es básicamente un actuador mecánico basado en un motor y un reductor. Funciona por impulsos.” (Viloria, 2014, pág. 257)

Como podemos observar, el servomotor se divide en tres partes principales que serán explicadas a continuación.

# Motor

Es un dispositivo capaz de convertir la energía eléctrica en energía motriz, produciéndose un efecto de rotación, el cual es producido por la acción de campos magnéticos. Su funcionamiento es el contrario al de un generador, pues este convierte la energía motriz en eléctrica. “Cuando dos imanes (o electroimanes) se aproximan uno a otro, uno es atraído o repelido con respecto al otro (…) entonces, las fuerzas entre los dos campos magnéticos producen el movimiento.” (Harper, 2000, pág. 1).



Figura 1.1: Motor

Un motor de corriente continua está conformado de tres componentes principales:

* **Estator:** Es la parte fija de un motor eléctrico de corriente continua, y es donde se encuentra el circuito electromagnético, el cual se encarga de producir la fuerza de rotación presente en el rotor.
* **Rotor:** Es el componente rotatorio del motor. Se compone por el eje de rotación y las bobinas. Se mueve gracias a la acción de los imanes del estator, los cuales generan una fuerza de repulsión que afecta al rotor.
* **Escobillas:** La función de estas, es la de alimentar al rotor que al estar en constante movimiento, no puede tener cables conectados. La tensión es enviada directamente por las escobillas, puesto que tienen una composición de grafito.

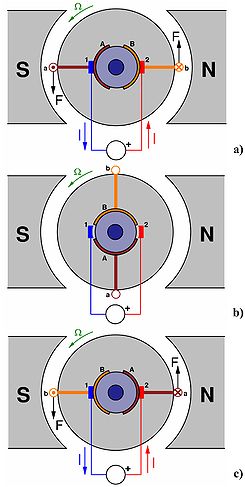


Figura 1.2: Esquema del funcionamiento básico de un motor

# Caja Reductora

Una caja reductora es un conjunto de engranajes que se encargan de reducir la velocidad que posee el motor. El movimiento producido por el motor, atraviesa 4 etapas de reducción, para lograr el movimiento final. “Las cajas de engranajes (o cajas reductoras) están compuestas de trenes de engranajes que poseen ejes intermedios con diferentes etapas de reducción.” (Francisco T. Sánchez Marín, 2006, pág. 55).

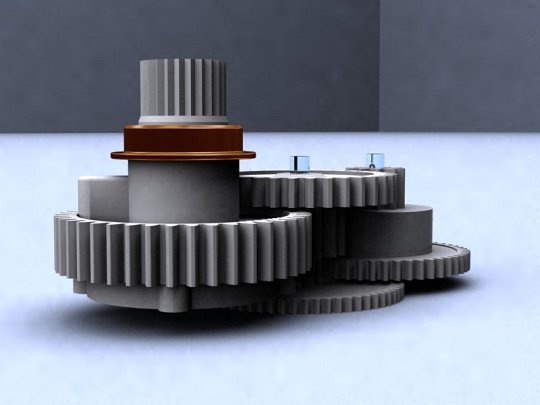


Figura 1.3: Caja reductora

Las etapas de reducción son 4, y se producen por la acción de un conjunto de engranajes, los cuales transmiten el movimiento en el siguiente orden:



Figura 1.4: Etapas de reducción de caja reductora

* **Rojo:** Engranaje movido por el motor.
* **Verde:** 1ra Etapa de reducción.
* **Azul:** 2da Etapa de reducción.
* **Morado:** 3ra Etapa de reducción.
* **Naranja:** 4ta Etapa de reducción.

# Circuito de Control (AA51880)

Este circuito, recibe las señales de pulso provenientes del circuito driver del servomotor. Se encarga de decodificar estas señales y determinar el ángulo en el cual se posicionará el servomotor. El funcionamiento de este CI se da en las siguientes etapas:

1. El voltaje de control ingresa por VIN. Este voltaje tiene una frecuencia determinada en ancho de pulso.
2. El voltaje ingresa a un generador de onda de diente de sierra (Sawtooth Wave & One Shot Generator), el cual genera una onda de forma triangular usando transistores.
3. El componente de posición del servomotor (Servo Position & Over-Time Logic), se encarga de comparar la onda generada por el componente anterior, y la señal de detección de posición del servo (POT). Esta señal se conoce como señal de retroalimentación y sirve para proveer estabilidad al servomotor.
4. En el componente ensanchador de pulso (Pulse Stretcher) se encarga de ensanchar una señal de error, la cual es la diferencia entre la señal de control del servomotor y la señal de retroalimentación. Este ensanchamiento provoca un aumento en el ciclo de trabajo del servomotor, ayudando también a aumentar su fuerza de sujeción.
5. El componente de direccionamiento lógico (Directional Logic Control), se encarga de comparar la onda de VIN y la onda generada por el generador de onda de diente de sierra con la onda de posicionamiento de servo, y envía un voltaje positivo o negativo dependiendo de si la una señal es más larga, o más estrecha que la otra. Esto ocasiona que el motor gire a una dirección u otra, hasta que ambas ondas sean iguales.
6. El control de ancho de pulso (PWM Control) consta de un flip-flop que recibe la señal generada por el direccionamiento lógico, y la envía a un puente H que controla la dirección que seguirá el servomotor. Este bloque ayuda a mejorar el desenvolvimiento del servomotor y su eficiencia.
7. El controlador de salida (Output Driver), controla la dirección del servo utilizando el OUT1 y el OUT2 para cargas ligeras (microservos), y conectando transistores en una configuración de puente H en las salidas EXP1 y EXP2, para manejar cargas más pesadas.

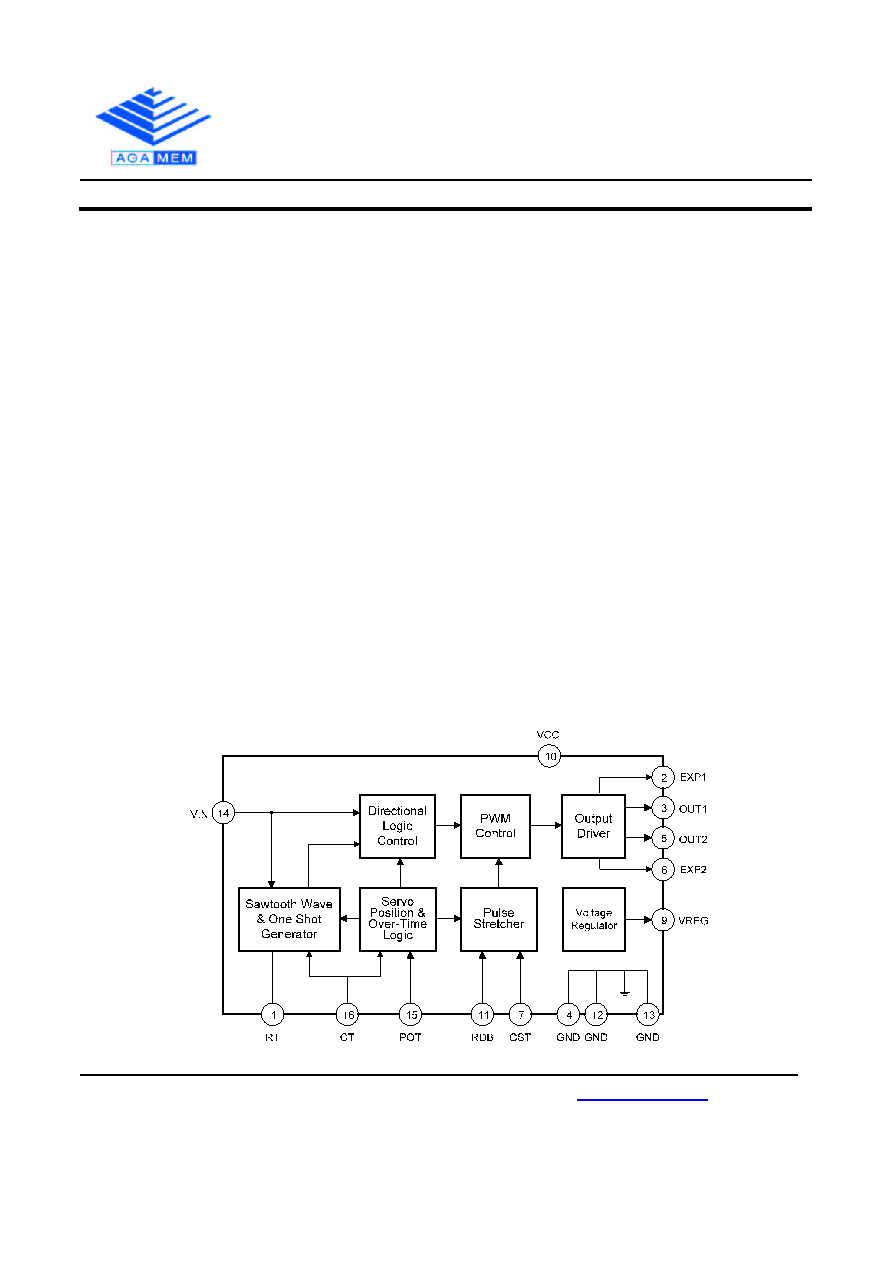


Figura 1.5: Funcionamiento chip integrado AA51880

El servomotor de corriente continua posee las siguientes características:

* Excitación a base de imanes cerámicos permanentes de elevada energía intrínseca y fuerza coercitiva.
* Buena regulación y estabilidad.
* Par elevado.
* Posibilidad de fuertes aceleraciones y deceleraciones.
* Gran estabilidad de marcha incluso a bajas velocidades.
* Amplio campo de variación de giro (hasta 210°).
* Elevada inercia térmica.
* Admite sobrecargas prolongadas.
* Para pequeñas y medianas cargas

# Tipos de Servomotores

* **Servomotores de Corriente Continua:**

DC, por sus siglas en inglés, son servomotores cuyos electrones generadores de corriente se mueven en un solo sentido: del polo negativo al polo positivo, la energía necesaria para el movimiento es mínima y puede generarse con pilas y baterías, por lo que los voltajes requeridos son pequeños.

* **Servomotores de Corriente Alterna:**

AC por sus siglas en inglés, en estos servomotores los electrones cambian de sentido en todo momento (alternan), realizando la transformación de energía mecánica en eléctrica. Este tipo de servomotores admite voltajes más altos, por lo que son ideales para las potencias requeridas por las máquinas al momento de desempeñar el proceso solicitado.

* **Servomotores de Imanes Permanentes o Brushless.**

Es una maquina sincrónica con la frecuencia de alimentación, capaz de desarrollar altos torques (hasta 3 o 4 veces su torque nominal) en forma transitoria para oponerse a todo esfuerzo que trate de sacarla de sincronismo. La denominación brushless proviene del hecho de que este servomotor no posee escobillas y es una forma de diferenciarlo de sus antecesores los servomotores a imán permanente alimentados con corriente continua.

Los servomotores poseen un tamaño estándar para su mayoría de diseños, el cual se representa en el siguiente gráfico.

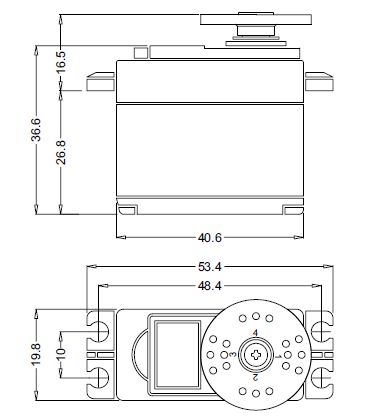


Figura 1.6: Servomotor

A pesar de poseer el mismo tamaño, existen muchos tipos de servomotores dependiendo de la cantidad de peso que son capaces de soportar, el cual puede ir desde los 3Kg hasta los 24Kg. El peso que posee un servomotor se encuentra bordeando los 8 gr.

En la industria, los servomotores han sido vastamente utilizados para realizar movimientos que requieren gran precisión y en muchas ocasiones, gran potencia. Gracias a las propiedades del servomotor, se pueden realizar dispositivos mecánicos como robots con una amplia libertad de movimiento, arañas mecanizadas, brazos robóticos, etc.

# Polimetilmetacrilato (Acrílico)

Es un polímero termoplástico altamente transparente que se obtiene de la polimerización del monómero metilmetacrilato de metilo. Un polímero es una macromolécula que se forma por la unión de monómeros. La polimerización es, como indica (Robert Thornton Morrison, 1987, pág. 330) “la unión de muchas moléculas pequeñas para dar origen a moléculas muy grandes”. Debido a su transparencia, estética y resistencia es considerado como una alternativa al vidrio por lo que se lo conoce como “cristal orgánico” o “cristal acrílico”. Cada monómero de metilmetacrilato se compone de Carbono, Hidrógeno y Oxígeno. Su fórmula es C5H8O2, y se presenta a temperatura ambiente como un líquido incoloro, tóxico e inflamable.

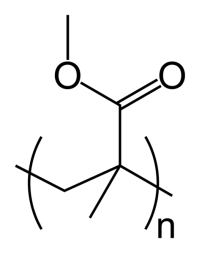


Figura 1.7: Formula esquelética del PMMA (Polimetilmetacrilato)

A pesar de ser llamado “cristal acrílico” no comparte ninguna similitud en su composición química con la del mismo.

Las características principales del PMMA son las siguientes:

* Propiedades ópticas excepcionales.
* Transparencia y acabado brillante.
* Rigidez y estabilidad dimensional.
* Dureza y resistencia a los rasguños.
* Excelente resistencia a los rayos del sol (radiación ultravioleta) y al envejecimiento por las condiciones meteorológicas.
* En el mercado se lo conoce con el nombre de “acrílico”.
* Se las encuentras en forma de láminas o tubos.

El acrílico comercialmente se encuentra a la venta en forma de láminas o planchas, y estas a su vez tienen diferentes tamaños en largo, ancho y espesor. Una lámina de acrílico de 120cm de ancho, por 120cm de largo y 3mm de espesor se encuentra a un valor de 50 USD, mientras que una lámina con las mismas características en lo que se refiere a ancho y largo, pero, con un espesor de 6mm se encuentra a un valor de 95 USD. Para el proyecto se ha usado una lámina de acrílico de espesor de 3mm con un ancho de 210mm y un largo de 160mm en un precio de 30 USD, mientras que una lámina de acrílico de 6mm de espesor con un ancho de 60mm por un largo de 175mm se encuentra en 12 USD. Con estas laminas adquiridas ser realizaron las piezas que conforman el brazo robótico.

# Placa Arduino

Es una tarjeta con hardware abierto que posee entradas y salidas analógicas y digitales, y que contiene un microcontrolador que nos permite comunicar un circuito electrónico con el computador, y programar a este circuito para que trabaje de la manera que nosotros deseamos. Su lenguaje está basado en el lenguaje de programación Processing, el cual a su vez está construido con JAVA y por lo cual posee gran potencial al realizar aplicaciones complejas. Tiene compatibilidad con prácticamente con todo lenguaje de programación, pues su comunicación es por el puerto serial. Al ser un hardware abierto podemos desarrollar cualquier tipo de proyecto sin tener que adquirid algún tipo de licencia.

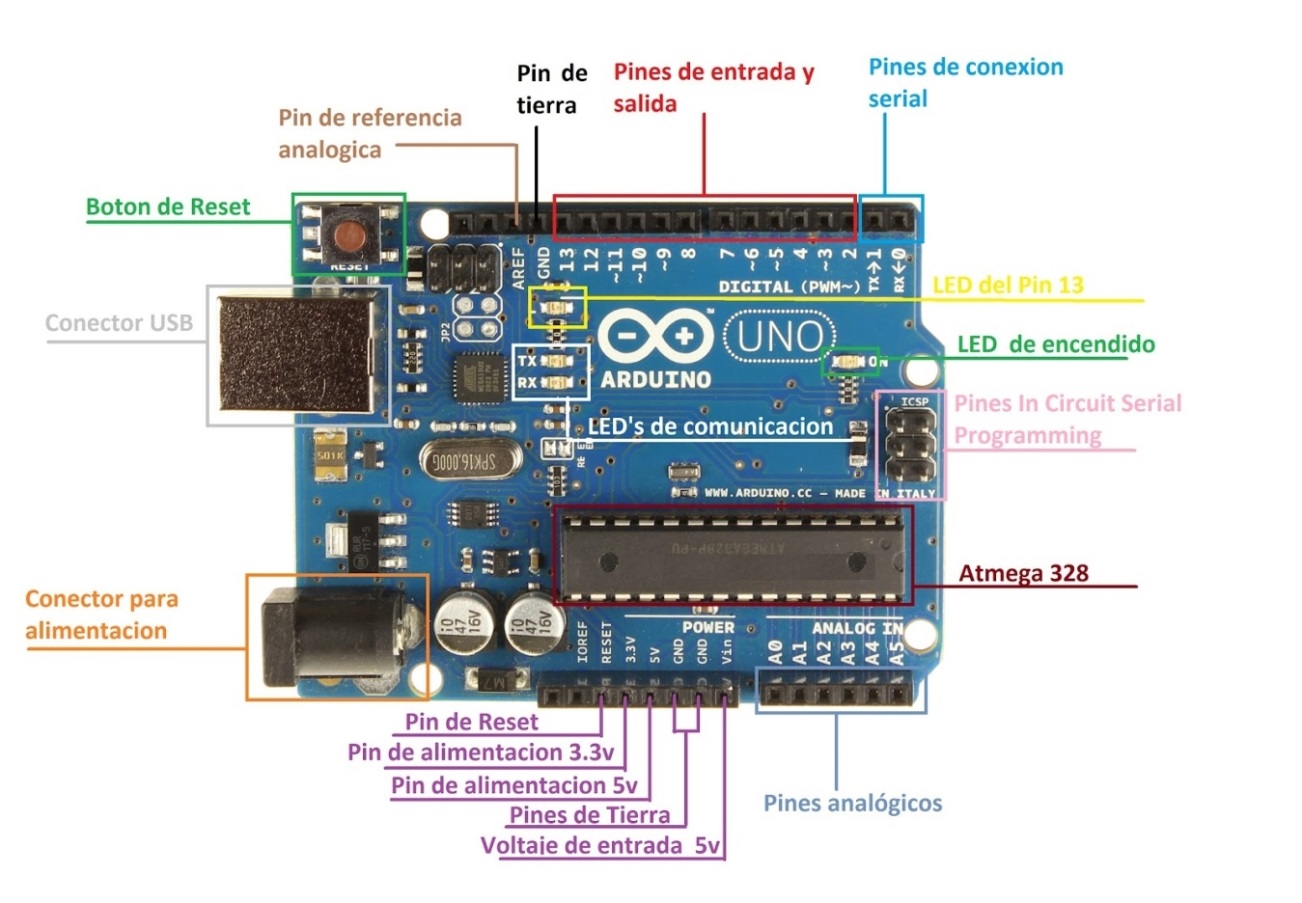


Figura 1.8: Placa Arduino UNO y sus partes

Arduino tiene una gran extensión de tipos de tarjetas. Las que más conocidas son:

* Arduino Uno
* Arduino Zero
* Arduino Micro
* Arduino Esplora
* Arduino Mega
* Arduino Mini
* Arduino Nano

Su tamaño y precio depende del tipo de tarjeta que vayamos a utilizar para nuestra necesidad.

# Sintaxis en Arduino

# setup()

Está función se utiliza al inicio del programa para inicializar variables, configurar pines o usar librerías. Se ejecuta una sola vez al encender el Arduino, o presionar el botón reset.

# pinMode (pin, mode)

Este comando configura a los pines para comportarse como entradas o salidas. Sus parámetros son:

* **pin:** El número de pin que deseamos configurar.
* **mode:** el modo que deseamos tenga el pin (INPUT, OUTPUT)

# Serial.begin(speed)

Establece el flujo de bits por segundo para la transmisión serial de datos. Para comunicarse con el computador generalmente se utiliza la velocidad de 9600 b/s. Tiene como parámetro:

* **speed:** bits/seg de la comunicación.

# loop()

Todos los comandos que se programen dentro de esta función, se repetirán de forma indefinida. Se utiliza para controlar activamente a la tarjeta Arduino.

# digitalWrite(pin, value)

Envía señales de HIGH (5 V) o LOW (0 V) al pin determinado. Sus parámetros son:

* **pin:** El número de pin al que se va a enviar la señal.
* **value:** valor que será enviado al pin (HIGH o LOW).

# digitalRead(pin)

Lee el valor de un pin especificado, ya sea HIGH o LOW. Sus parámetros son:

* **pin:** Número de pin del cual se leerá el valor.

# analogWrite(pin, value)

Envía un valor análogo a un determinado pin. Esta señal es enviada por PWM, y posee valores de 0-255. Los pines que soportan PWM son el pin 3, 5, 6, 9, 10 y 11. Sus parámetros son:

* **pin:** Número de pin al cual se enviará la señal.
* **value:** Valor entre 0 (siempre apagado) y 255 (siempre prendido).

# analogRead(pin)

Lee un valor análogo de los pines A0-A5. Su función es leer el voltaje de 0V a 5V y convertirlo a un valor entero de 0 a 1023. Sus parámetros son:

* **pin:** Número del pin de entrada análoga (van del 0 al 5).

# delay(ms)

Detiene el programa por una determinada cantidad de tiempo. Sus parámetros son:

* **ms:** Número de milisegundos que se pausará el programa.

# map (value, fromLow, fromHigh, toLow, toHigh)

Reasigna un número de un rango A, a un rango B. Sus parámetros son:

* **value:** El número a convertir.
* **fromLow:** El valor más bajo del rango actual.
* **fromHigh:** El valor más alto del rango actual.
* **toLow:** El valor más bajo del rango a convertirse.
* **toHigh:** El valor más alto del rango a convertirse.

# Serial.read()

Lee los datos provenientes de la entrada serial.

# Serial.print(val)

Imprime datos al puerto serial. Si el dato enviado es de tipo Integer, devuelve su valor en ASCII, mientras que si es de tipo char o String, imprime el mismo dato enviado. Sus parámetros son:

* **val:** el valor a imprimirse.

# Serial.println(val)

Su funcionamiento es igual al del Serial.print, con la diferencia de que al final, envía un retorno para cambiar de línea.

A continuación se muestra un ejemplo realizado en Arduino usando todos los comandos antes mencionados.

char dato; //Datos que se leerán de la consola serial  
int valores; //Valores que se van a leer de la fotocelda  
void setup()  
{  
 Serial.begin(9600); //Iniciamos la comunicación Serial  
 pinMode(13,OUTPUT); //El pin 13, se asignará como una salida  
}  
void loop()  
{  
 digitalWrite(13,HIGH); //Encendemos el Led  
 dato=Serial.read(); //Leemos datos de la consola Serial  
 if(dato=='a') //Si el dato leído corresponde a la letra "a"  
 {  
 valores=analogRead(A0); //Leemos datos de la fotocelda   
 valores=map(valores, 0,1023, 0,100); //Transformamos los datos  
 Serial.print("Concentración de luminosidad: ");  
 Serial.println(valores); //Imprimimos en porcentaje la luminosidad del ambiente  
 delay(1000); //Detenemos el programa por 1s o 1000ms  
 }  
 if(dato=='b') //Si el dato leído corresponde a la letra "b"  
 {  
 valores=analogRead(A0); //Leemos datos de la fotocelda  
 valores=map(valores, 0,1023, 0,255); //Transformamos los datos  
 analogWrite(A1, valores); //Encendemos un Led por PWM  
 delay(1000); //Detenemos el programa por 1s o 1000ms  
 }  
 digitalWrite(13,LOW); //Apagamos el Led  
}

# Resistencias

Una resistencia es un componente electrónico que se encarga de oponerse al flujo de corriente, reduciendo así el voltaje y la intensidad del circuito electrónico. La resistencia se mide en Ohmios, que se representa con la letra griega omega (Ω). El aparato que se utiliza para medir la resistencia o continuidad es el ohmímetro

“Un conductor cuya resistencia eléctrica es pequeña, conduce bien la corriente eléctrica. Se dice que su conductancia es grande. Si, por el contrario, un conductor tiene una gran resistencia, se dice que su conductancia es pequeña” (Senner, 1994, pág. 28)Entendemos a conductancia como la calidad del elemento para conducir la corriente eléctrica, y es un concepto contrario al de la resistencia.

La resistencia se relaciona directamente con otros dos parámetros de la electrónica que son el voltaje o tensión y la intensidad o corriente. Para esto existe una ecuación matemática descubierta por Georg Simon Ohm, llamada la “Ley de Ohm” que nos indica que la resistencia (R) equivale al voltaje (V) medido en volteos sobre la intensidad (I) medida en amperios:

Físicamente se la encuentra conformado por materiales semiconductores, cubierto por cerámica e identificada por un conjunto de colores que representan la resistencia que posee el mismo. A continuación se muestra una tabla con su código de colores:

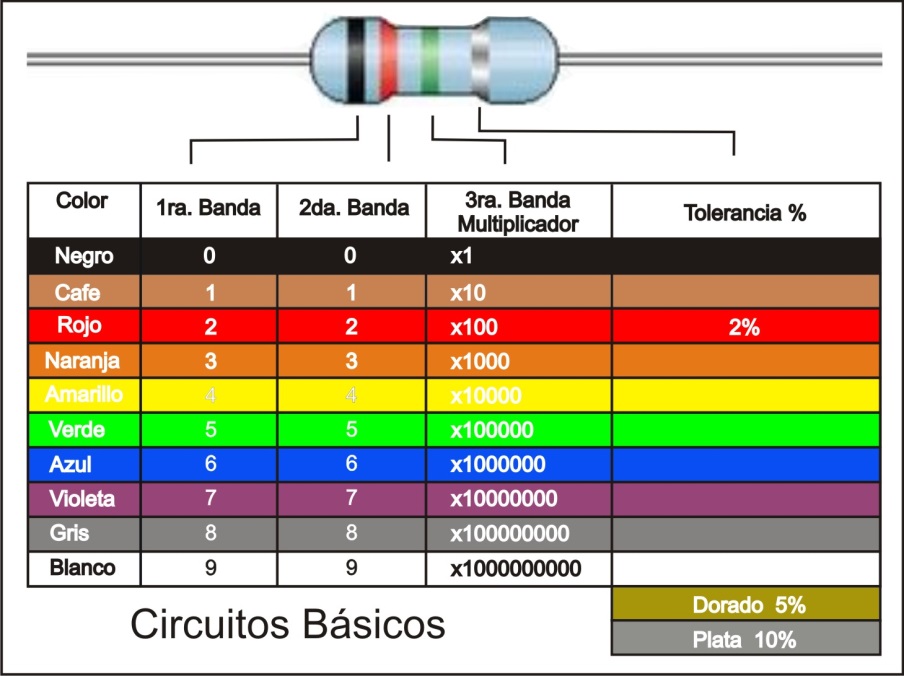


Figura 1.9: Código de colores de resistencia

# Modulo Bluetooth HC-06

Es un dispositivo bluetooth desarrollado por Arduino. Proporciona una señal bluetooth la cual puede ser manipulada por el usuario. “Estos módulos funcionan por lo general como dispositivos esclavos, por lo que deberá ser el otro extremo de la comunicación (normalmente un computador o un teléfono móvil con capacidad Bluetooth) el que funcione como dispositivo maestro” (Artero, 2013, pág. 546)

Estos dispositivos se comunican mediante señales “RX” y “TX”, en donde “RX” es la señal de recepción de datos y “TX” es la señal de transmisión de datos. La función de este módulo es la de recibir señales vía bluetooth, y convertirlas en datos seriales manipulables para Arduino; de igual manera recibe datos seriales desde Arduino y los convierte en señales bluetooth que será enviada al dispositivo maestro.

A pesar de la existencia de otras series de modulo bluetooth, por la simplicidad de uso y las funcionalidades del mismo, el módulo HC-06 es el que mayor ventajas ofrece para el desarrollo de nuestro proyecto. A continuación se muestra el dispositivo físico, con sus respectivos pines.

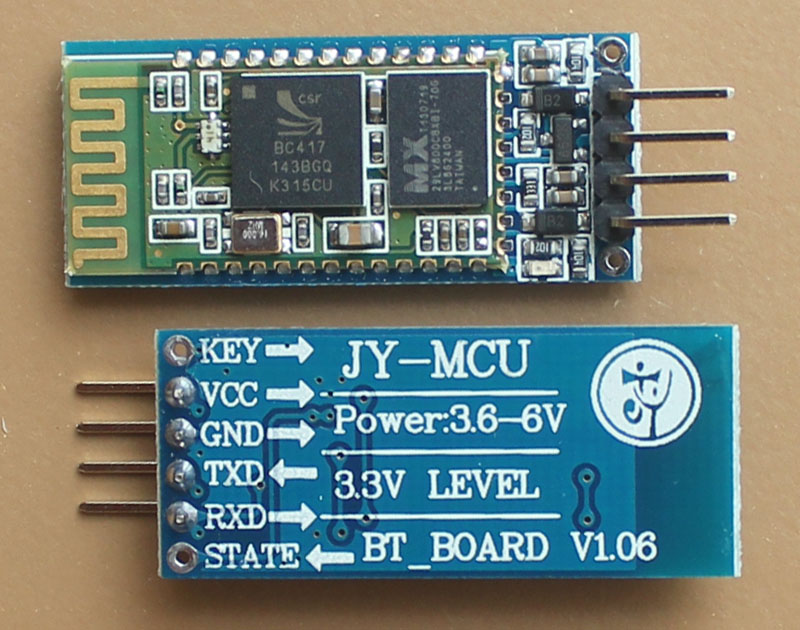


Figura 1.10: Módulo bluetooth HC-06

Para la conexión se utilizan los siguientes pines:

* **VCC:** Se conecta con la alimentación de 5V de la tarjeta Arduino.
* **GND:** Se conecta con el pin de GND de Arduino.
* **TXD:** Es el transmisor del módulo. Se conecta con el pin 0 de Arduino (pin de recepción de datos seriales).
* **RXD:** Es el receptor del módulo. Se conecta con el pin 1 de Arduino (pin de transmisión de datos seriales).

Adicional a esto, el módulo posee un led que indica si el apareamiento del dispositivo no se ha realizado correctamente (led parpadeante), o en su defecto, si se ha realizado correctamente (led encendido).

# Sistema Operativo Android

“Android es un sistema operativo basado en el núcleo Linux diseñado originalmente para dispositivos móviles, tales como teléfonos inteligentes, pero que posteriormente se expandió su desarrollo para soportar otros dispositivos tales como Tablet, reproductores MP3, netbook, PC, televisores, lectores de e-book e incluso, se han llegado a ver en el CES, microondas y lavadoras.” (Mann, 2012, pág. 1). Este sistema operativo, posee una gran interacción con el usuario al poseer una pantalla multitouch, una interfaz gráfica de fácil comprensión y puede implementar sensores adicionales como un acelerómetro, un giroscopio, un sensor de brillo, etc. Hoy en día, es uno de los principales sistemas operativos para Smartphone, siendo su única competencia el sistema operativo iOS, por lo cual un gran porcentaje de usuarios de Smartphones posee este sistema operativo.

# App Inventor

Es una aplicación desarrollada por Google Labs para crear aplicaciones de software exclusivamente para el sistema operativo Android. La manera de programar es muy simple puesto que se realiza mediante la unión de bloques los cuales poseen distintas funciones como matemáticas, lógicas y funciones especiales como son llamar objetos tales como la cámara, el micrófono, acelerómetro, servicios Bluetooth o Wireless.

Una de las grandes ventajas de este entorno es que puede ser manejado mediante su página oficial en la web “*http://appinventor.mit.edu/*” o desde el ordenador teniendo instalado el programa, otra de estas ventajas es que podemos crear aplicaciones sin escribir líneas de código en JAVA, al trabajar en un entorno que utiliza la librería Open Blocks de Java para crear un entorno de bloques y hacer a la aplicación más sencilla de trabajarla a más de que este programa es de código abierto, es decir, que no requiere de la adquisición de una licencia pagada para su uso, y está disponible a todo público.

Para realizar una aplicación utilizando esta herramienta, lo primero que se debe poseer es una cuenta en Gmail, a través de la cual realizaremos nuestro registro en la página. Una vez tenemos un usuario, hacemos clic en el botón Create para acceder al entorno de programación de App Inventor.

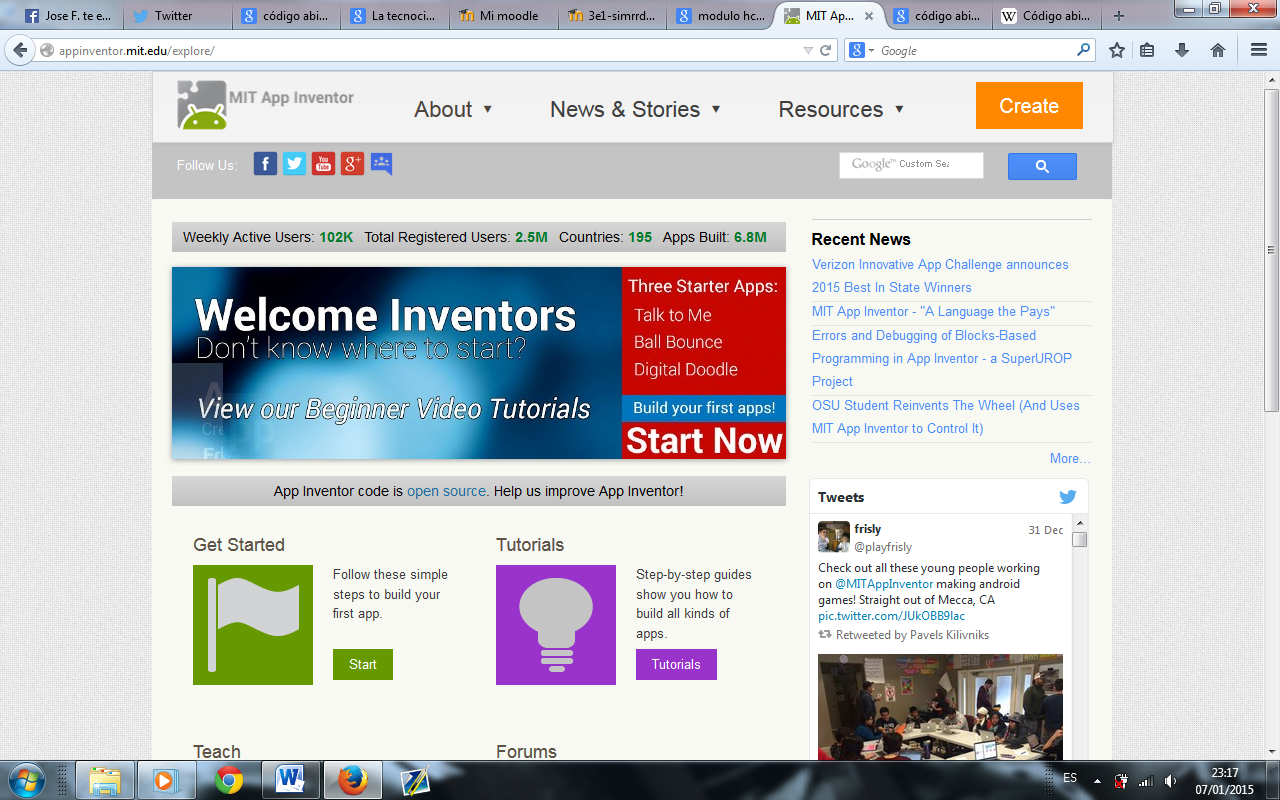


Figura 1.11: Página de inicio de App Inventor

Una vez hecho esto, se nos enviará a un índice donde se guardará todos los proyectos que vayamos creando con nuestra cuenta. Para crear un nuevo proyecto, seleccionamos “Start new project” y damos un nombre al proyecto e inmediatamente nos encontraremos con una ventana que simula la pantalla de Android, y con todos los elementos que podemos usar para nuestra aplicación.

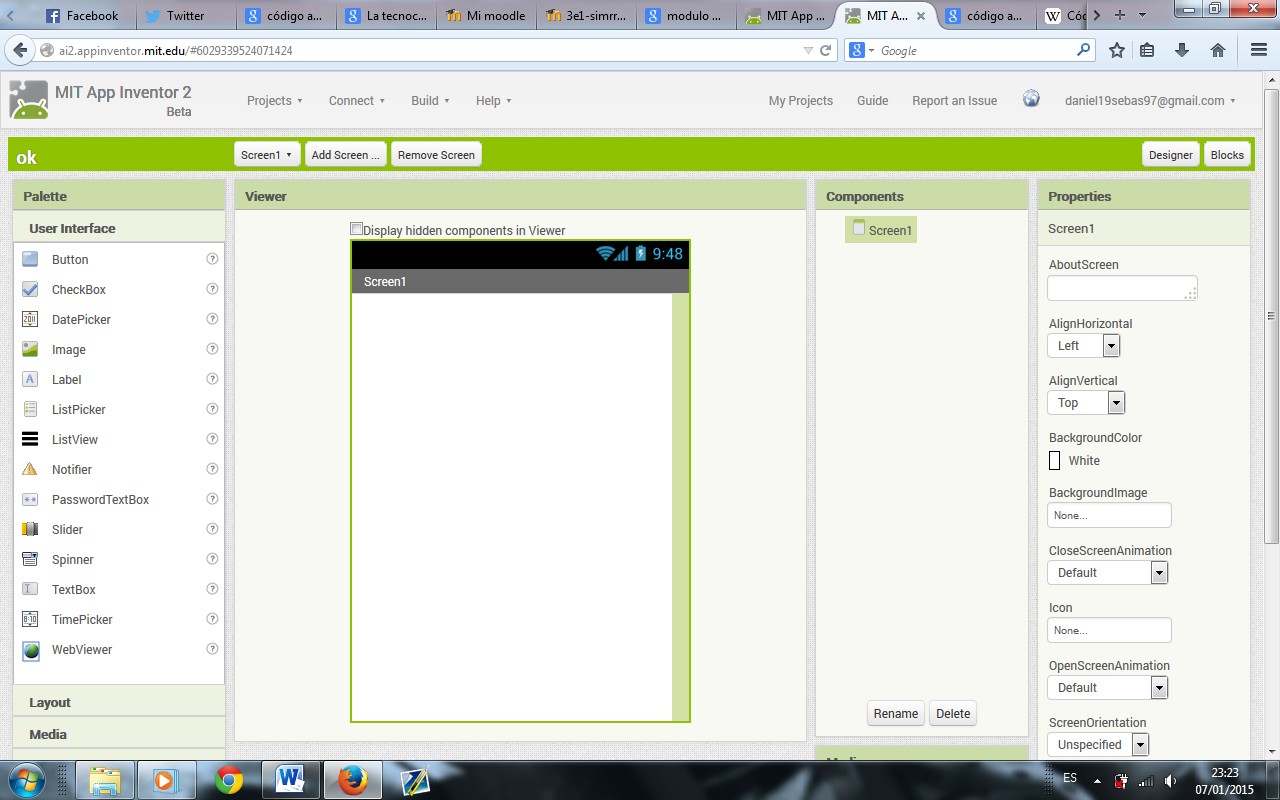


Figura 1.12: Página de diseño de aplicaciones App Inventor

# Acelerómetro

Es un dispositivo que detecta un cambio de aceleración en una masa sísmica, con respecto al dispositivo. Es decir que detecta el peso que sufre esta masa, y nos permite detectar su aceleración en los ejes X, Y, y Z.

“El valor de la fuerza medida está relacionado con el valor de la aceleración por la segunda ley de Newton” (Maloney, 2006, pág. 439)

Medida por un transductor de fuerza

Una cantidad conocida fija de masa

Sus aplicaciones en los Smartphone son principalmente para software que requieran de un sensor de posicionamiento para conocer la orientación del móvil en la dirección deseada en un eje cartesiano de X, Y y Z, que vendría a ser horizontal, vertical y profundidad, respectivamente.

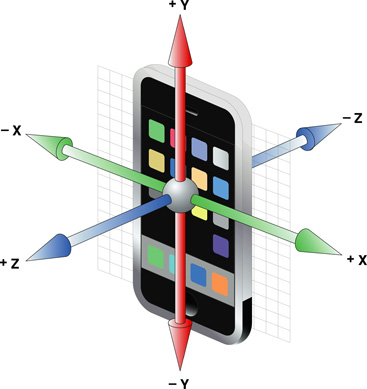


Figura 1.13: Eje cartesiano del funcionamiento acelerómetro

Dentro de un Smartphone se encuentra a este dispositivo como un microchip, el cual es necesariamente muy pequeño para economizar espacio. Se ubica en la parte central del Smartphone.

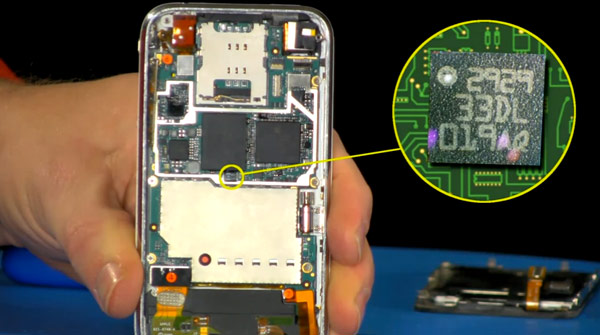


Figura 1.14: Microchip acelerómetro

Su funcionamiento interno se puede representar de forma teórica en la siguiente gráfica:

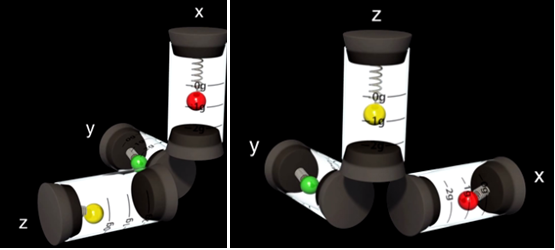


Figura 1.15: Funcionamiento interno acelerómetro

Al rotar el celular, las masas de referencia sufren la alteración de esta rotación, y producen un peso que es detectado por el acelerómetro, indicando así la posición espacial en la que se encuentra el móvil.

Dentro de AppInventor, su programación se encuentra predefinida entre los componentes del celular, y solo es necesario incluirlo en el diseño de la aplicación para poder utilizar sus bloques de control.

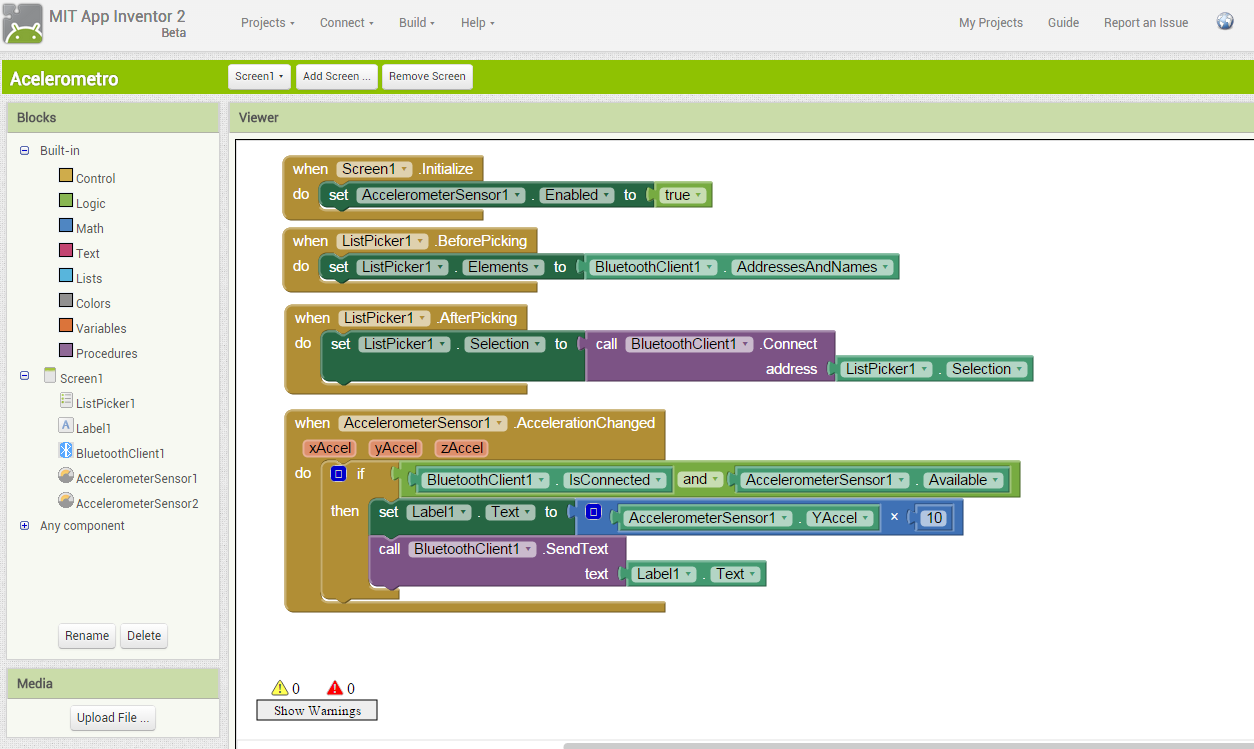


Figura 1.16: Aplicación en App Inventor para control de acelerómetro

# Sensor de Humedad y Temperatura

Es un dispositivo que nos permite conocer, la humedad y temperatura que se encuentra en el ambiente donde se lo implementa.

En el mercado existen muchas variantes de las cuales destacan, por su facilidad de uso, precisión y por la capacidad de medir estos dos parámetros, los modelos DHT11 y DHT22, los cuales son compatibles e incluso existe una librería que ha sido creada para este propósito tal que podremos controlaros desde una interface de Arduino.

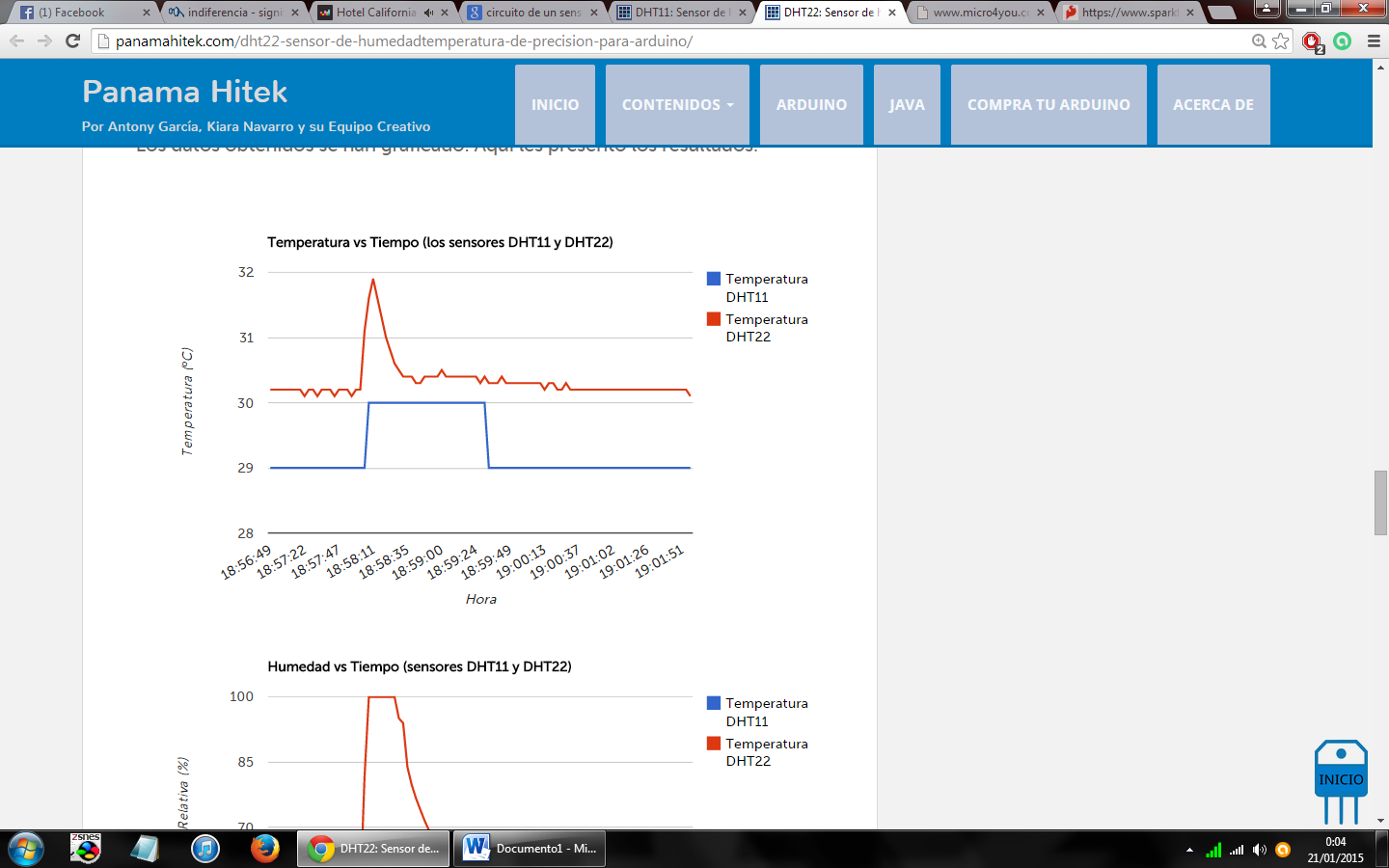


Figura 1.17: Temperatura vs Tiempo (Sensores DHT11 y DHT22)

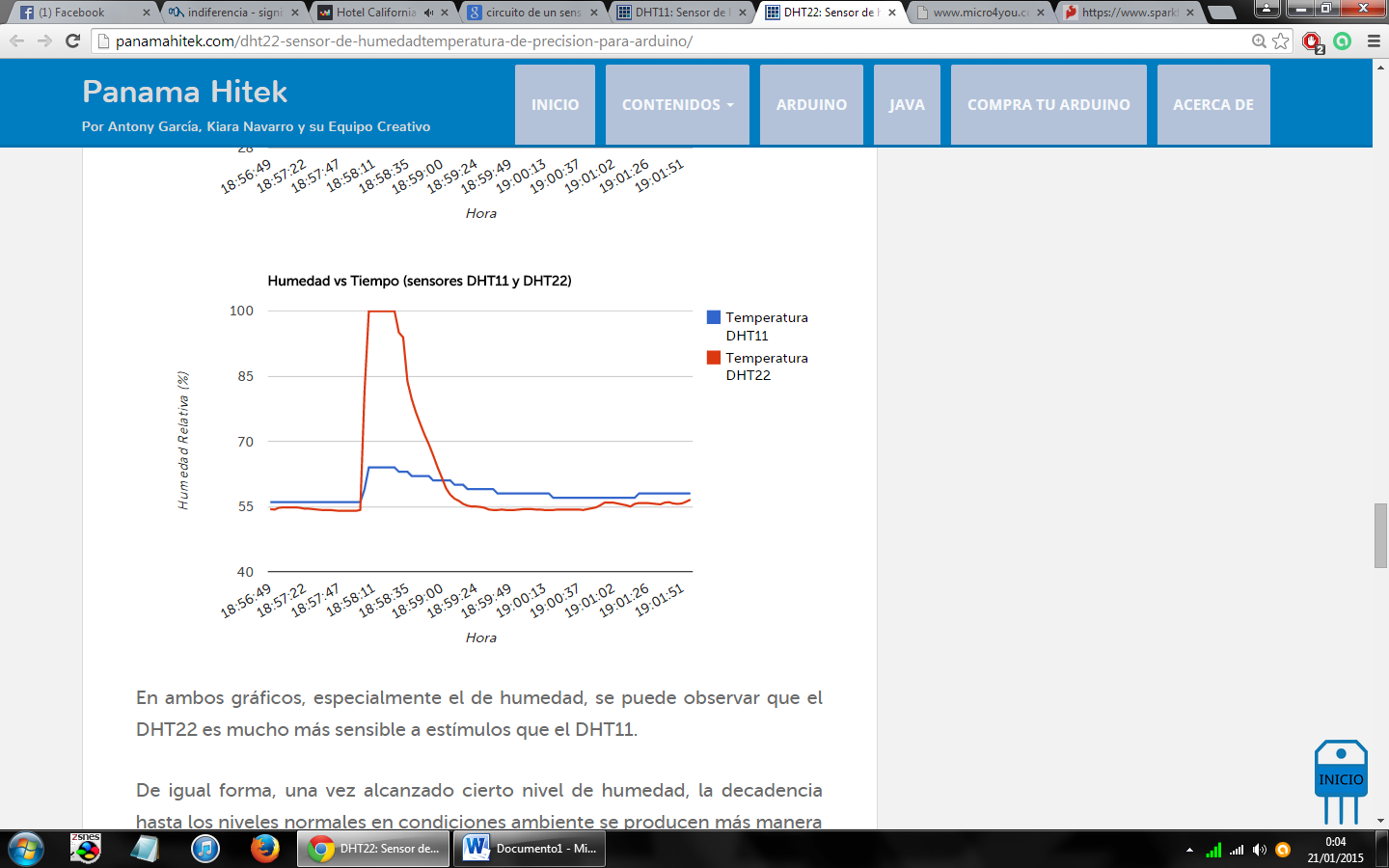


Figura 1.18: Humedad vs Tiempo (Sensores DHT11 Y DHT22)

El DHT22 es mucho más sensible a estímulos que el DHT11, lo cual convierte al DHT22 un dispositivo confiable y estable.

Una vez alcanzado cierto nivel de humedad, la decadencia hasta los niveles normales en condiciones ambiente se producen de una manera más pronunciada en el DHT22 que en el DHT11.

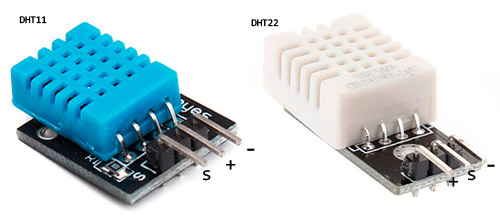


Figura 1.19: DHT11 Y DHT22

Podemos evidenciar que existen también diferencias físicas notables entre su color y tamaño del encapsulado.

A continuación se muestra un cuadro comparativo de ventajas y desventajas entre estos dos dispositivos:

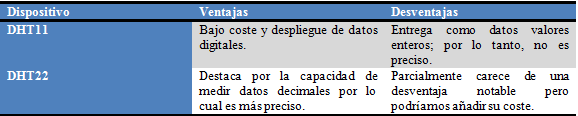


Tabla 1.1: Ventajas y Desventajas sensores de humedad

La especificación técnica de ambos dispositivos son las siguientes:

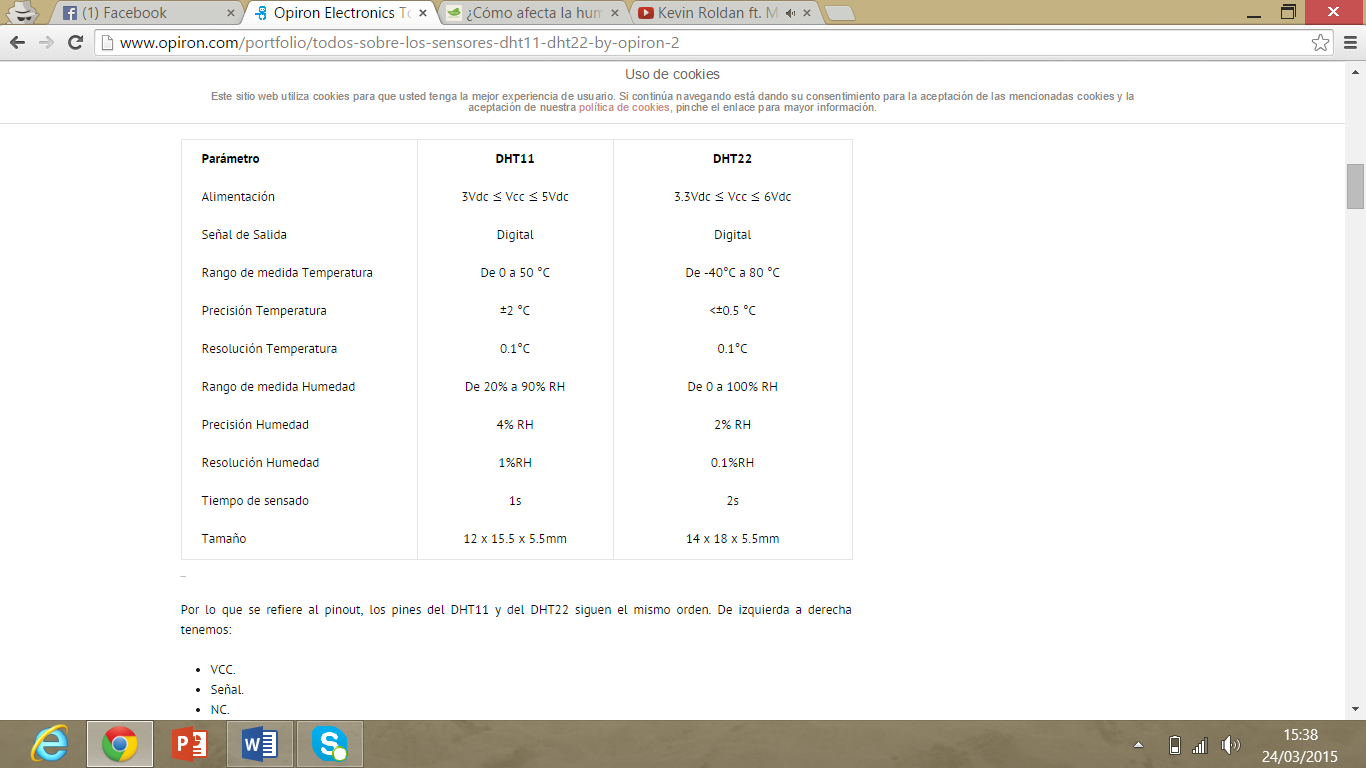


Tabla 1.2: Especificación técnica sensores de humedad

La funcionalidad de este dispositivo en el proyecto consta en ayudar a evitar quemaduras en los pacientes residentes en casas de salud que interactúan con objetos de alta temperatura. El valor de esta temperatura se mostrará al usuario de forma visual utilizando un LED RGB el cual tomará tres tonalidades distintas que serán: frio, tibio, caliente; dependiendo del valor de la temperatura medida por el dispositivo.

Por el mismo motivo hay que tener en cuenta que el exceso de humedad contenida en un cierto ambiente incrementa la posibilidad de padecer dolencias respiratorias, por lo cual la medición de la humedad del ambiente también evitará contraer enfermedades por factores ambientales.

# Diodos Emisores de Luz (LED)

“El diodo emisor de luz (LED) es un diodo capaz de emitir luz visible cuando se energiza” (Robert L. Boylestad, 2003, pág. 40), este es un componente opto electrónico pasivo que consta de un ánodo (pin positivo) y cátodo (pin negativo).

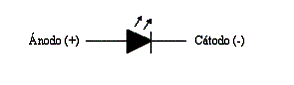


Figura 1.20: Diodo Emisor de Luz

# Funcionamiento

El funcionamiento normal consiste en que, en los materiales conductores, un electrón, al pasar de la banda de conducción a la de valencia, pierde energía; esta energía perdida se manifiesta en forma de un fotón desprendido, con una amplitud, una dirección y una fase aleatoria.

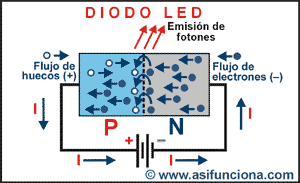


Figura 1.21: Funcionamiento LED

Cuando un electrón pasa de la banda de conducción a la de valencia, se puede manifestar como un fotón desprendido o como otra forma de energía (calor por ejemplo) dependiendo principalmente del tipo de material semiconductor. Cuando un diodo semiconductor se polariza directamente, los huecos de la zona positiva se mueven hacia la zona negativa y los electrones se mueven de la zona negativa hacia la zona positiva; ambos desplazamientos de cargas constituyen la corriente que circula por el diodo.

El dispositivo semiconductor está comúnmente encapsulado en una cubierta de plástico de mayor resistencia que las de vidrio que usualmente se emplean en las lámparas incandescentes. Aunque el plástico puede estar coloreado, es solo por razones estéticas, ya que ello no influye en el color de la luz emitida. Usualmente un LED es una fuente de luz compuesta con diferentes partes, razón por la cual el patrón de intensidad de la luz emitida puede ser bastante complejo.

Para obtener buena intensidad luminosa debe escogerse bien la corriente que atraviesa el LED. Para ello hay que tener en cuenta que el voltaje de operación va desde 1,8 hasta 3,8 voltios aproximadamente (lo que está relacionado con el material de fabricación y el color de la luz que emite) y la gama de intensidades que debe circular por él varía según su aplicación. Los valores típicos de corriente directa de polarización de un LED corriente están comprendidos entre los 10 y los 40 mA.

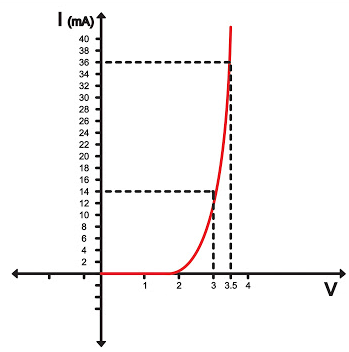


Figura 1.22: Intensidad vs Voltaje LED

# Partes físicas del LED

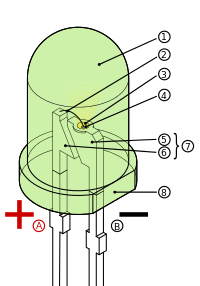
A: ánodo

Figura 1.23: Partes físicas LED

B: cátodo

1: lente/encapsulado epóxico (cápsula plástica).

2: contacto metálico (hilo conductor).

3: cavidad reflectora (copa reflectora).

4: terminación del semiconductor

5: yunque

6: poste

7: marco conductor

8: borde plano

# LED RGB

Este dispositivo electrónico no es más que un LED de 4 pines que contiene tres colores básicos: rojo (Red), verde (Green), azul (Blue); los mismos que dan origen al nombre “RGB” y un cuarto pin que dependiendo del tipo que este sea se conectará a: GND en caso de que sea del tipo “cátodo común”, y a VCC en caso de que este sea del tipo “ánodo común”.

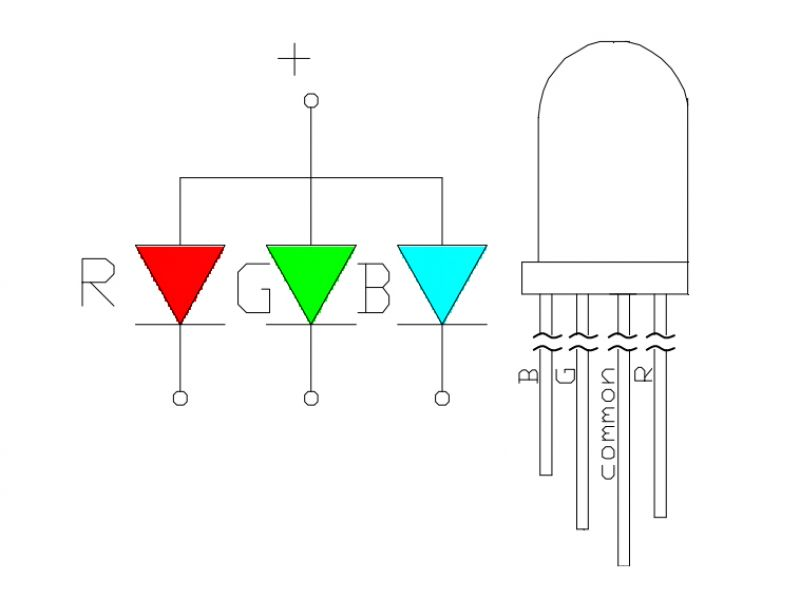


Figura 1.24: LED RGB

# Funcionamiento RGB

Es un modelo de color basado en la síntesis aditiva, con el que es posible representar un color mediante la mezcla por adición de los tres colores de luz primarios

En el caso del LED RGB se energiza el color que se desea entre un rango de 0 a 255 para definir la intensidad de brillo de cada uno y con la mezcla de estos colores se formara uno nuevo.

# Tabla de colores principales

|  |  |  |  |
| --- | --- | --- | --- |
| Color | HTML/CSS Name | Hex Code  #RRGGBB | Decimal Code  (R,G,B) |
|  | Negro | #000000 | (0,0,0) |
|  | Blanco | #FFFFFF | (255,255,255) |
|  | Rojo | #FF0000 | (255,0,0) |
|  | Lima | #00FF00 | (0,255,0) |
|  | Azul | #0000FF | (0,0,255) |
|  | Amarillo | #FFFF00 | (255,255,0) |
|  | Cyan / Aqua | #00FFFF | (0,255,255) |
|  | Magenta / Fuchsia | #FF00FF | (255,0,255) |
|  | Plata | #C0C0C0 | (192,192,192) |
|  | Gris | #808080 | (128,128,128) |
|  | Marrón | #800000 | (128,0,0) |
|  | Oliva | #808000 | (128,128,0) |
|  | Verde | #008000 | (0,128,0) |
|  | Purpura | #800080 | (128,0,128) |
|  | Teal | #008080 | (0,128,128) |
|  | Navy | #000080 | (0,0,128) |

Tabla 1.3: Cuadro de colores LED RGB

# CAPÍTULO II

# ANÁLISIS Y DISEÑO

# Planificación del Proyecto

# Estudio de factibilidades

# Factibilidad Operacional

Por parte de la institución Unidad Educativa Técnico Salesiano, se está recibiendo el apoyo necesario, la supervisión y la aprobación para la correcta realización del proyecto. Actualmente el proyecto es un prototipo de un brazo robótico para casas de salud, realizado con la visión de proporcionar ayuda a las personas con dificultades motrices y que necesiten de un dispositivo adicional para realizar tareas cotidianas, que ellos solos tendrían dificultades en realizar. Este prototipo, podría llegar a ser de interés para ciertos hospitales y clínicas, las cuales no tienen grandes cantidades de personal y no pueden abastecer a varios pacientes o casas de salud que tienen demasiada cantidad de personal y quieren hacer un corte de personal sin dejar de buscar el bienestar del paciente.

# Factibilidad Técnica

La tecnología que se adquirirá e implementará en el proyecto, se conseguirá en distintos almacenes especializados en componentes electrónicos. Entre los componentes que se utilizarán para el funcionamiento del proyecto se encuentran: Servomotores, Arduino UNO, módulo bluetooth HC-06 y otros componentes electrónicos, además de que se necesitará un dispositivo móvil que tenga incluido un acelerómetro para controlar los movimientos que realizará el brazo robótico, el cual también poseerá conexión bluetooth, necesaria para establecer conexión con el módulo HC-06 para la transmisión de datos y comunicación entre el software y el hardware.

# Factibilidad Económica

Todos los gastos generados durante el desarrollo del proyecto serán financiados por los mismos estudiantes. En el caso de que el producto fuese implementado ayudaría a agilizar el tráfico de personas en una casa de salud, logrando así disminuir el número de personal y aumentando la agilidad de atención, además de que ayudaría financieramente a la casa de salud debido a que esta deberá ejercer menos gastos en salarios para los empleados.

# Recursos

# Recursos de Hardware

La máquina con la cual se desarrollará el proyecto tendrá las siguientes especificaciones: procesador de 2.10 GHz, memoria RAM de 4GB. En esta máquina específicamente se realizará la investigación y el documento monográfico.

Además, contaremos con una segunda máquina con las siguientes características: procesador 2.40GHz, memoria RAM de 4GB. Con esta se realizará el software del proyecto y los distintos diseños del periférico motriz.

Entre los componentes físicos que utilizará el brazo robótico se encuentran: una tarjeta Arduino UNO necesaria para la programación de los movimientos y control del brazo robótico; un dispositivo móvil para que el usuario pueda enviar los comandos que requiere que el brazo realice; servomotores para ejecutar todos los movimientos de rotación del brazo y otros componentes electrónicos. La estructura del brazo robótico constará de piezas diseñadas y realizadas en acrílico las cuales serán de 3mm y 5mm de grosor, necesarias para proveer resistencia y estabilidad al brazo.

Para la ejecución de la aplicación, es necesario poseer un celular con el sistema operativo Android 1.5 o superior; además de que el dispositivo móvil debe poseer de conexión bluetooth y acelerómetro para poder utilizar todas las funciones que requiere la aplicación.

# Recursos de Software

# Herramientas para la investigación y documento monográfico

* Sistema Operativo Windows 7 Ultimate
* Navegador Web Mozilla Firefox
* Microsoft Office 2010 o superior

# Herramientas para el desarrollo de software

* MIT App Inventor 2
* Arduino 1.0.6 o superior
* AutoCAD 2010 o superior

# Estimaciones

# Estimaciones de tiempo

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Actividades** | **Sept** | | **Octubre** | | | | **Noviembre** | | | | **Diciembre** | | | | **Enero** | | | | **Febrero** | | | | **Marzo** | | | | **Abril** | | | | **Mayo** | | | |
| **1** | **2** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** |
| Anteproyecto |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Diseño aplicación |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Diseño electrónico |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Ensamblaje brazo robótico. |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Implementación de la aplicación en Arduino |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Implementación de la aplicación en AppInventor |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Programación para comunicación entre AppInventor y Arduino |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Pruebas funcionamiento |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Tabla 2.1: Estimaciones de tiempo en el aspecto práctico

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Actividades** | **Sept** | | **Octubre** | | | | **Noviembre** | | | | **Diciembre** | | | | **Enero** | | | | **Febrero** | | | | **Marzo** | | | | **Abril** | | | | **Mayo** | | | |
| **1** | **2** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** |
| Búsqueda Bibliográfica del protocolo |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Protocolo |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Búsqueda Bibliográfica del Capítulo 1, 2, 3. |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo Capítulo 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo Capítulo 2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo Capítulo 3 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Tabla 2.2: Estimaciones de tiempo en el aspecto teórico

# Estimaciones de costo

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Item** | **Denominación** | **Cantidad** | **P. Unitario** | **P. Total** |
| 01 | Servomotor TowerPro MG995 | 3 | 23.21 | 69.64 |
| 02 | Servomotor HiTec HS-311 | 3 | 13.84 | 41.52 |
| 03 | Módulo Bluetooth HC-06 | 1 | 14.29 | 14.29 |
| 04 | Motor DC con caja reductora | 1 | 31.25 | 31.25 |
| 05 | Estructura de acrílico | 1 | 35.71 | 35.71 |
| 06 | Cable flexible para Arduino | 5 | 2.23 | 11.16 |
| 07 | Pines de conexión para Arduino | 40 | 0.27 | 10.71 |
| 08 | Módulo Sensor de Temperatura y Humedad AM2301 | 1 | 17.81 | 17.81 |
| 09 | Fuente de Corriente Continua 5V-5A | 1 | 35.71 | 35.71 |
| 10 | Tarjeta Arduino UNO | 1 | 31.25 | 31.25 |
| 11 | Base de Acrílico para Arduino | 1 | 10.71 | 10.71 |
| 12 | Circuito Integrado LM2309 | 1 | 1.79 | 1.79 |
| 13 | Led RGB | 1 | 0.36 | 0.36 |
| 14 | Base de Acero Inoxidable | 1 | 178.57 | 178.57 |
| 15 | Piezas de Acero Inoxidable para hombro. | 2 | 4.47 | 8.93 |
| **Subtotal** | | | | **499.41** |
| **IVA** | | | | **59.94** |
| **Total** | | | | **559.35** |

Tabla 2.3: Estimaciones de Costo

El Software Arduino 1.0.6 es un producto con licencia libre, por lo cual no es necesario estimar un costo de software para comprar licencias.

App Inventor 2 es una plataforma IDE otorgado por google que se encuentra en internet, por lo que está disponible para cualquier usuario, haciéndolo a éste un software con licencia libre, gracias a esto no se tiene que estimar un costo de licencia.

Mientras que AutoCAD es un producto con licencia privada, por lo que éste tiene un costo estimado de €360, lo que equivale a 396 USD.

# Definición y Alcances del Proyecto

Nuestro periférico motriz, será capaz de manipular objetos de dimensiones aproximadas a los 5 cm de ancho, y con una masa aproximada a los 150 gr. Esto debido a que este proyecto será un prototipo, el cual podría llegar a construirse a mayor escala, pero que no se realizará de esta manera debido a limitaciones tanto en la factibilidad económica como técnica.

El brazo robótico, tendrá un movimiento libre en dos ejes (X y Y), al mismo tiempo que podrá girar en su propio eje, gracias a una base rotatoria, lo cual lo permitirá trasladarse en el eje Z, dándole un buen alcance en un radio de aproximadamente 50 cm.

El periférico que estamos planeando realizar, se limitará a movilizarse sobre un eje de rotación dentro de una base, es decir, el alcance que tenga el brazo se limitará a la longitud de su estructura. La base del mismo no ha sido planificada para que pueda desplazarse de su sitio, sino simplemente para permitir una rotación del periférico.

# Recolección y Análisis de Requisitos

Para este punto se utilizó el método de observación gracias a lo cual se logró conocer los principales inconvenientes que existen en las distintas casas de salud, los cuales pueden llegar a ser solucionados con la ayuda del proyecto que se está realizando.

Dentro de las casas de salud se puede comprobar que existe una gran cantidad de personal para la atención de los pacientes. Muchas veces estos pacientes requieren realizar tareas sencillas y necesitan de la ayuda de un auxiliar médico, lo que provoca que haya una aglomeración de personas de forma innecesaria.

Al analizar estos inconvenientes, se puede concluir como finalidad del proyecto la utilización del brazo robótico como una ayuda para las personas con dificultades médicas. Gracias a la estructura que posee el brazo robótico, es posible para el usuario alcanzar objetos de dimensiones bajas y manipularlas, evitándose la ayuda de otras personas, solucionándose de esta manera la problemática observada anteriormente.

# Diseño de Software (Ciclo de vida del proyecto)

# Análisis de requisitos

Las personas con problemas de salud, muchas veces encuentran su movilidad limitada, lo cual no les permite realizar ciertas tareas cotidianas, tales como alimentarse, beber líquidos, tomar un medicamento, etc. Para solucionar esta problemática, hemos planteado desarrollar un proyecto que consiste en el ensamblaje y sistema de control de un brazo robótico articulado el cual poseerá tres articulaciones y tendrá la finalidad de realizar todas las actividades que los pacientes no puedan hacer por cuenta propia que ya han sido mencionados.

# Diseño del Sistema

El proyecto constará principalmente de cuatro elementos:

En primer lugar se realizará la estructura física del brazo, la cual se conformará de una base, hombro, antebrazo, brazo, muñeca, y pinza. La mayor parte de esta estructura estará realizada a base de polimetilmetacrilato (acrílico) el que dará una gran resistencia al brazo, además de reducir el peso del mismo; el único elemento que no se realizará con este material será la base del brazo, la cual será de acero inoxidable para balancear el peso de toda la estructura, además de su resistencia y dureza.

Acoplado a la estructura se encontrará el siguiente elemento, el circuito electrónico que brindará movilidad al brazo. Este circuito constará de 6 servomotores y un motor con caja reductora, los cuales se conectarán a cada articulación del brazo distribuidas de la siguiente manera: En la base se encontrará el motor con caja reductora conectado a un puente H, el cual nos permitirá una movilidad de 360° pudiendo así manipular objetos que se encuentren cualquier dirección alrededor del brazo; en el hombro se utilizarán dos servomotores Tower Pro MG 995 con un torque de 10 kg/cm ya que en este punto será de vital importancia la fuerza para poder levantar objetos pesados; de la misma manera se utilizará este servomotor en el codo del brazo, debido a que es otro punto que soporta gran cantidad de peso; en la muñeca se utilizarán dos servomotores HS-311 con un torque de 3.7 kg/cm, uno para el movimiento vertical, y otro para la rotación de la misma; finalmente en la pinza se utilizará un Micro Servo Tower Pro SG90 con un torque de 1.8 kg/cm, que servirá para abrir y cerrar la pinza del brazo.

En tercer lugar viene el circuito y programa de control, para el cual utilizaremos una tarjeta Arduino UNO. Todos los pines de control de los servomotores se conectarán a las salidas PWM del Arduino, mientras que el puente H se conectará a dos salidas digitales del mismo. Otro elemento que se incluirá en el circuito de control es el módulo Bluetooth HC-06, que se conectará a los pines de recepción y transmisión serial del Arduino; con este elemento se comunicará a la aplicación de Android con el programa de Arduino, para el control del circuito de movilidad del brazo. El programa que manejara a la tarjeta Arduino, recibirá todos los datos enviados desde el móvil Android, los procesará y los enviará a los distintos dispositivos, lo cual permitirá el correcto funcionamiento del brazo.

Por último, el programa destinado al usuario final se realizará con App Inventor, el cual se maneja con lenguaje de programación JAVA, esta aplicación contendrá la interfaz gráfica con la cual el usuario manejará el brazo robótico de forma sencilla. Este programa se encargará de interpretar las solicitudes del usuario y las enviará al módulo bluetooth para la comunicación con la tarjeta Arduino. Las solicitudes del usuario se realizarán mediante elementos como sliders, botones y acelerómetro para controlar la posición de los servomotores, y check box para escoger la parte del brazo que se va a mover.

# Diseño del Programa

# Diseño AppInventor

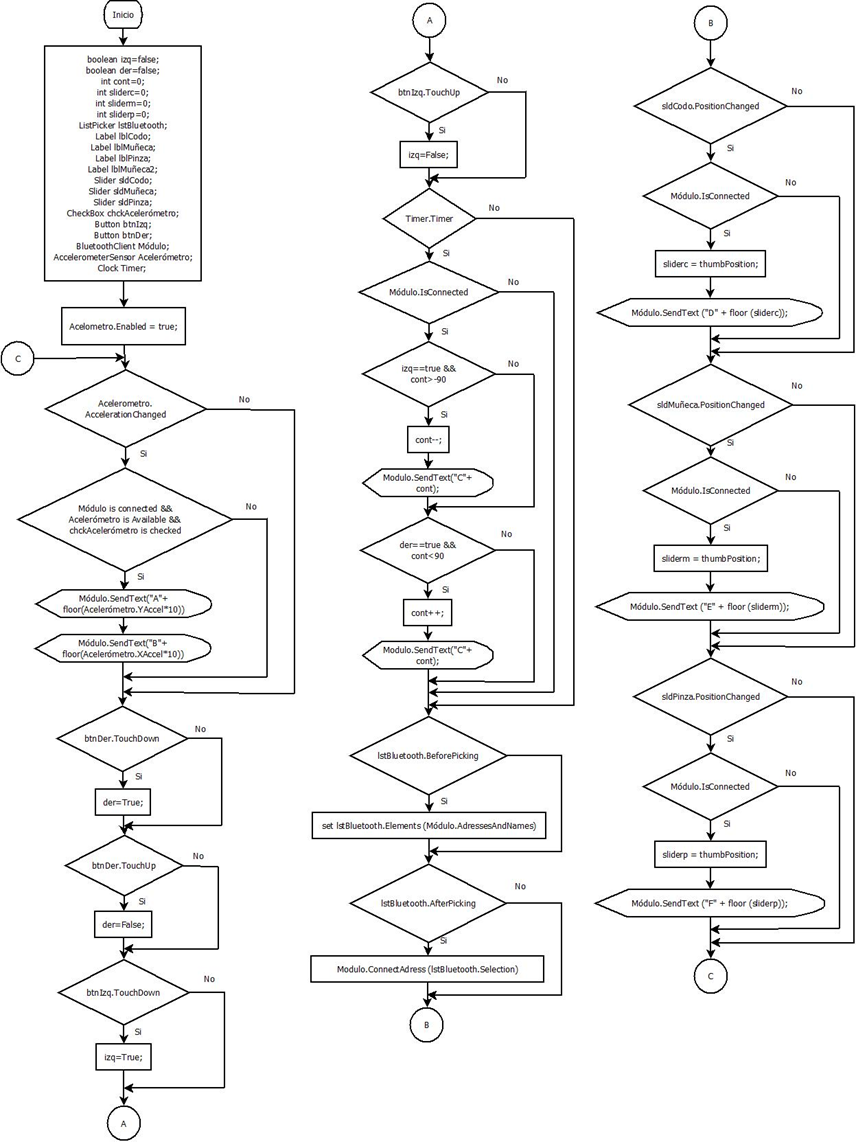


Figura 2.1: Diseño programa AppInventor

# Diseño Arduino

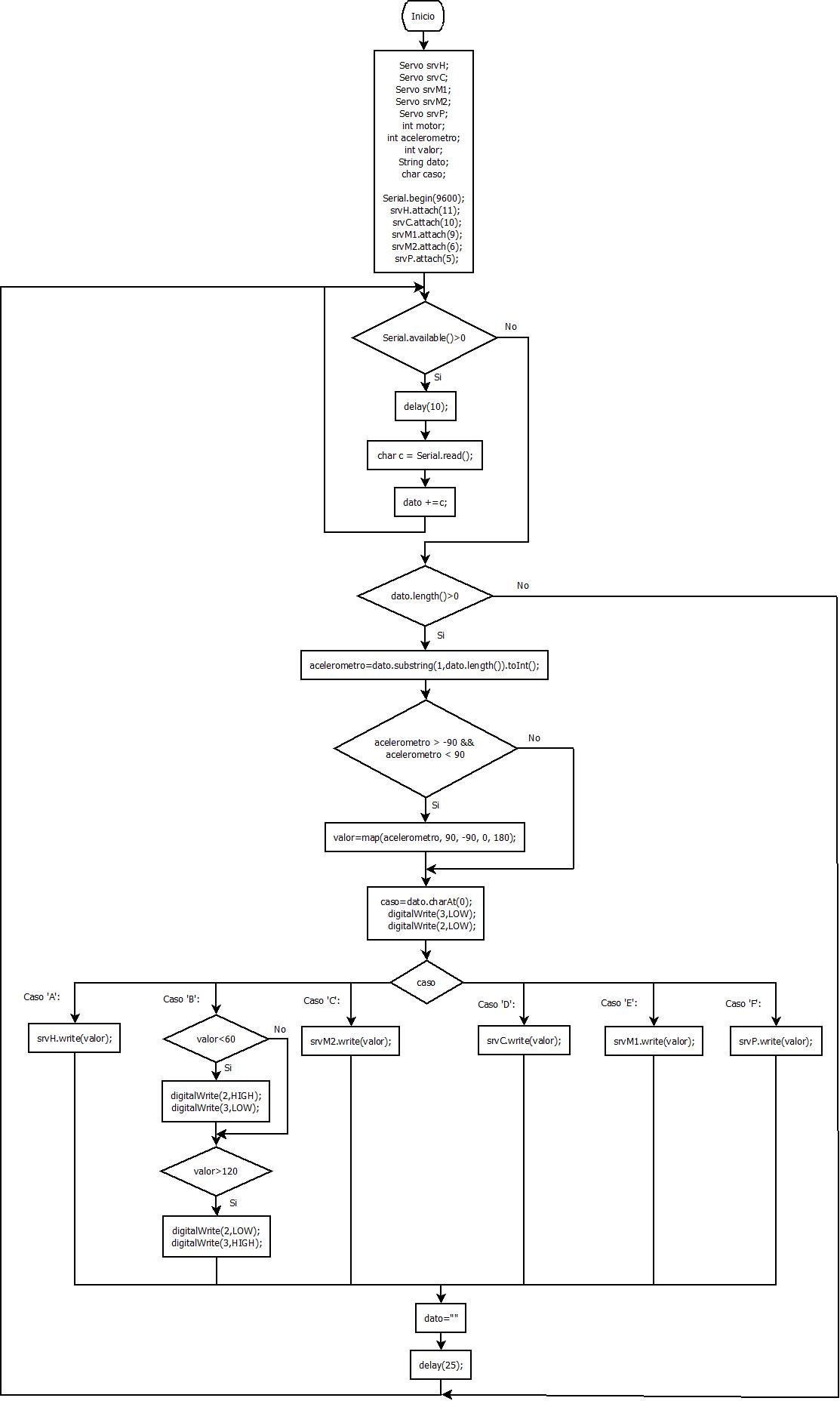


Figura 2.2: Diseño programa Arduino

# Codificación

# Codificación en Arduino

El programa realizado en Arduino principalmente cumplirá la función de lectura de datos mediante la transmisión serial, para recibir los datos enviados desde la aplicación realizada en AppInventor. Una vez obtenidos los datos, está aplicación los descompondrá en dos, el un dato simbolizará el servomotor que deseamos sea movido, y el otro dato el ángulo al cual deseamos que se mueva. Adicional a esto, poseerá una función de lectura y transmisión de datos del sensor de temperatura y humedad, y comandos para la utilización del LED RGB *(Véase anexo 1, 2, 3, 4, 5, 6, 7, 8)*

# Codificación AppInventor

La aplicación desarrollada en AppInventor, cumplirá el rol de recibir todas las solicitudes del usuario para enviarlas a Arduino. Su interfaz gráfica está creada para ser usada de forma intuitiva, mediante la aplicación de sliders, botones y checkbox. Al realizar cualquier acción en estos elementos, existirá un envío de los valores cambiados a Arduino, para modificar el movimiento del brazo. Adicionalmente, se encarga de leer los datos de temperatura y humedad emitidos por Arduino mediante conexión Bluetooth, para que sean visualizados de forma textual por el usuario. *(Véase anexo 9, 10, 11)*

# Diseño Electrónico

Para la realización del circuito de conexión, se utilizará principalmente las salidas digitales con Modulación de Ancho de Pulso (PWM) de Arduino, las mismas que permitirán el movimiento angular de los servomotores. Adicionalmente se ha utilizado una fuente de alimentación externa con un elevado amperaje con el fin de que no existan problemas con la fuerza de torque de los servomotores. En los pines de Transmisión y Recepción Serial se ha conectado un Módulo Bluetooth HC-06 para la transmisión de datos, y finalmente se ha conectado un sensor de temperatura y humedad con un led RGB para indicar la temperatura y humedad del ambiente.

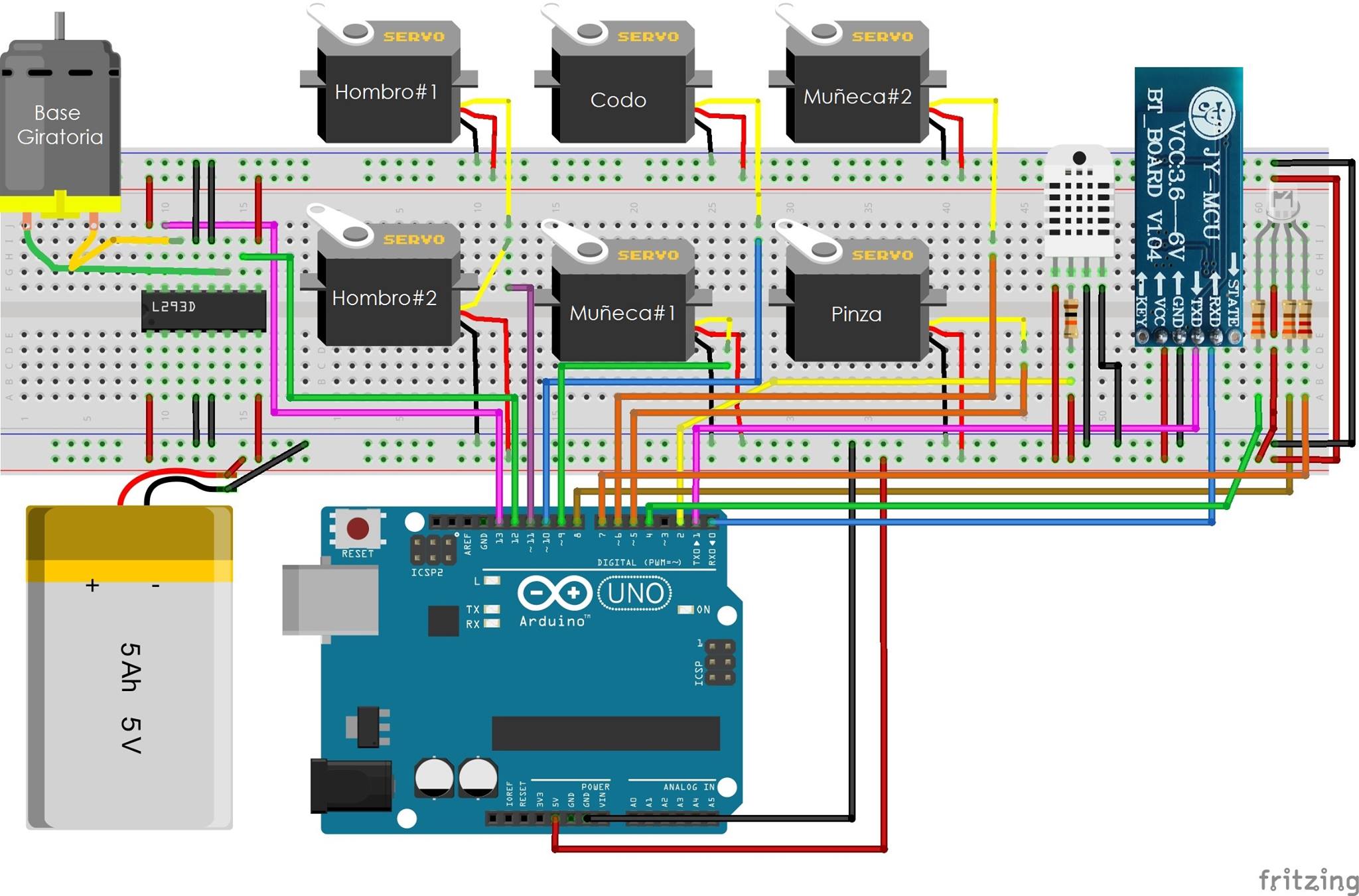


Figura 2.3: Diseño electrónico del brazo robótico

# CAPÍTULO III

# CONSTRUCCIÓN Y PRUEBAS

# Construcción y Ensamblaje del Hardware

Para la estructura del brazo, es necesaria la realización de los diseños de todos los elementos del brazo. Estos diseños han sido realizados en AutoCAD 2010, para su posterior elaboración en acrílico.

Con los diseños realizados, se prosiguió con la realización de las piezas en polimetilmetacrilato. Las piezas fueron realizadas en grosores de 3mm y 5mm, y se realizaron en un centro de diseño gráfico de corte a láser llamado “Trazos Design”, por lo cual las dimensiones fueron iguales a las del diseño. Las piezas en acrílico fueron las siguientes: *(Ver anexo 12, 13, 14, 15, 16, 17)*

Para el movimiento articulado del brazo robótico, se utilizaron los siguientes servomotores:

* 2 Servomotores MG 995 para el movimiento del hombro.

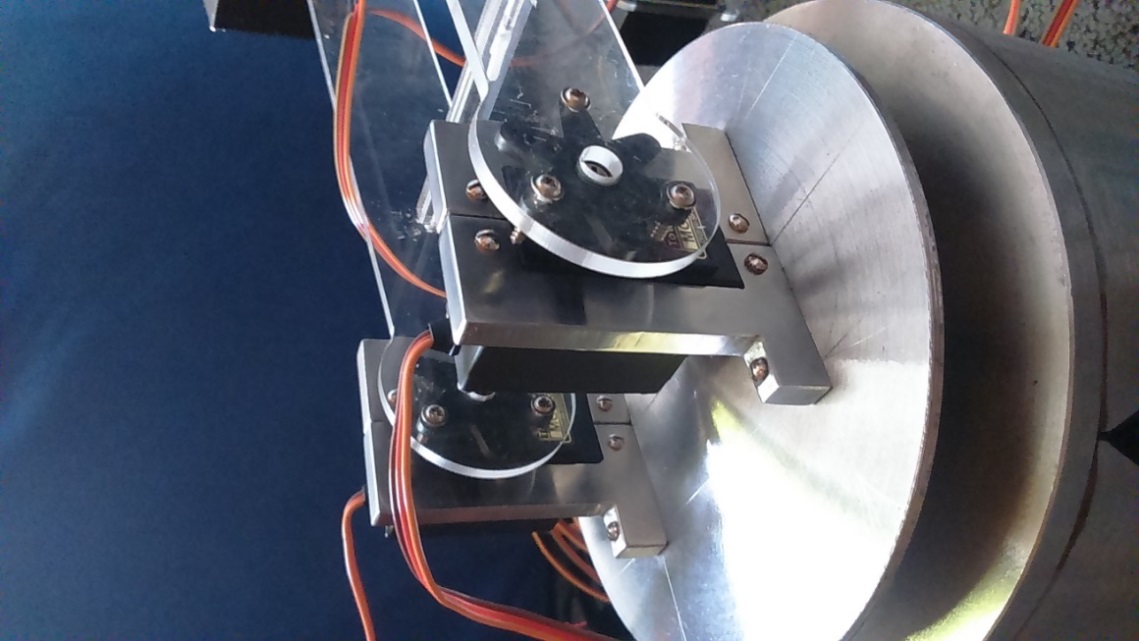


Figura 3.1: Servomotor movimiento hombro

* 1 Servomotor MG 995 para el movimiento del codo.

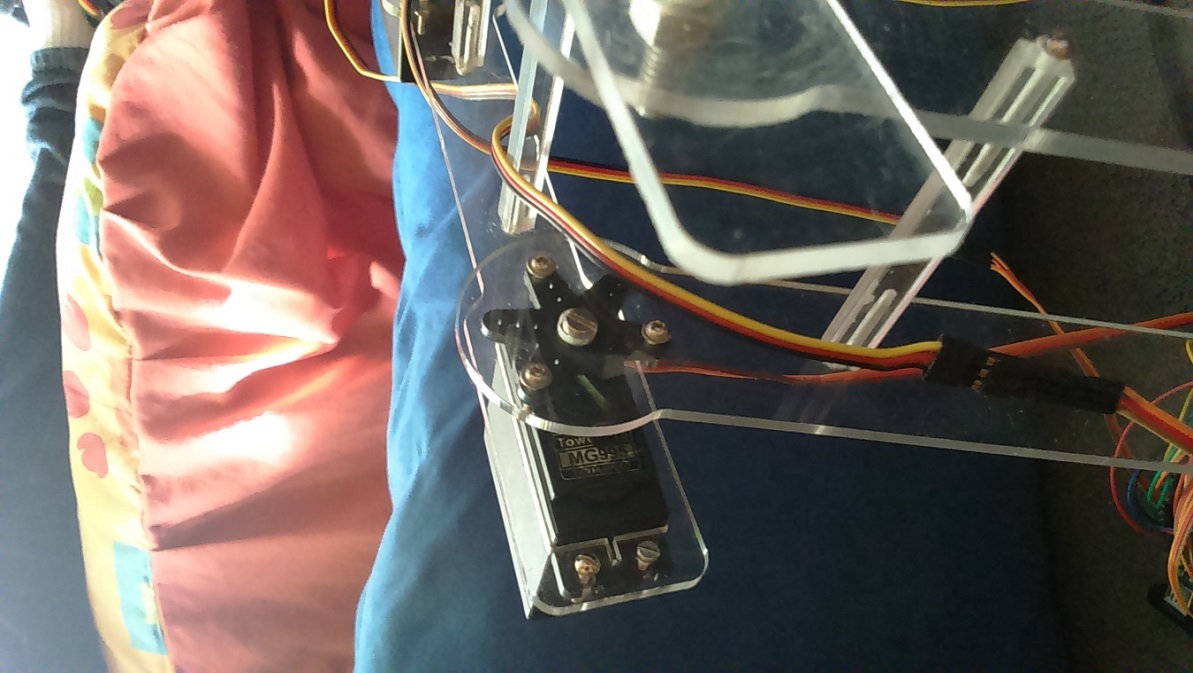


Figura 3.2: Servomotor movimiento codo

* 1 Servomotor HS 311 para el movimiento vertical de la muñeca.



Figura 3.3: Servomotor movimiento vertical muñeca

* 1 Servomotor HS 311 para el movimiento rotacional de la muñeca.

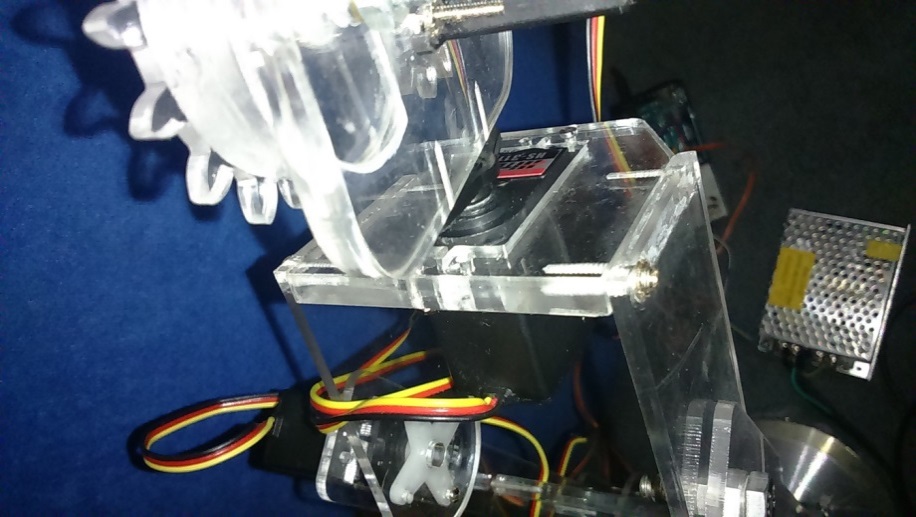


Figura 3.4: Servomotor movimiento rotacional muñeca.

* 1 Servomotor HS 311 para la apertura de la pinza.



Figura 3.5: Servomotor apertura pinza

La base del brazo robótico fue realizada en acero inoxidable (inox) para conseguir un mayor peso y soporte para el brazo. Este elemento posee de un sistema de rodamiento conectado a un motor de caja reductora el cual produce un movimiento de rotación de 360° sobre el eje central del brazo.

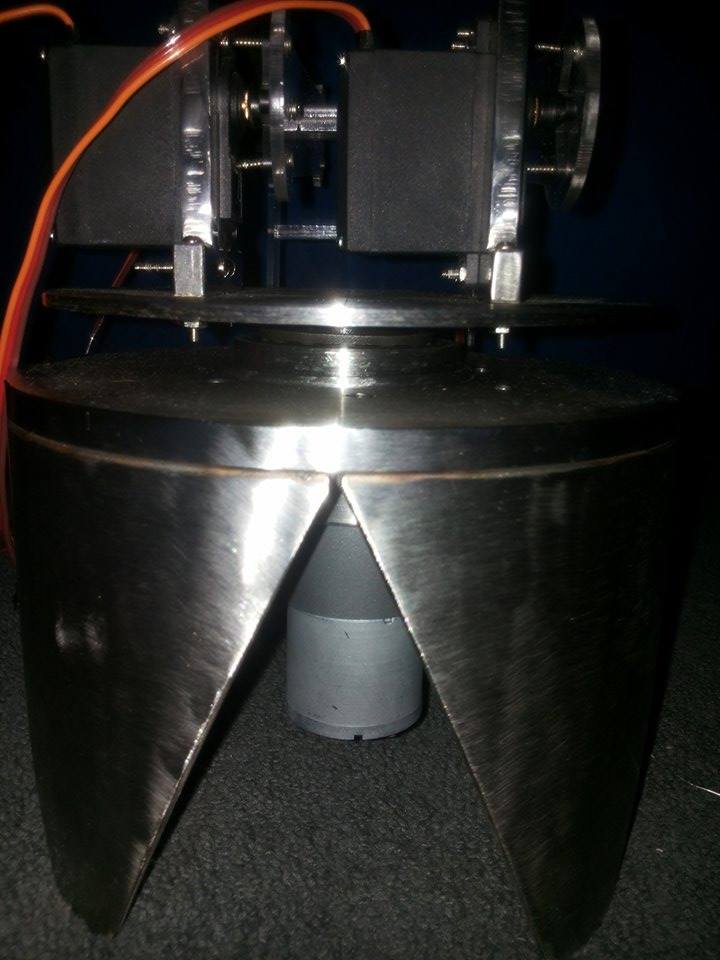


Figura 3.6: Base de inox

La finalidad principal de este elemento es incrementar el alcance del brazo, permitiéndole alcanzar objetos a aproximadamente 50 cm de radio del brazo.

# Implementación del Sistema

# Envío de datos desde el Hardware

La transmisión de datos del hardware, se produce mediante un módulo de comunicación bluetooth HC-06, el cual utiliza los canales de recepción y transmisión serial de Arduino para establecer conexión con el dispositivo móvil con sistema operativo Android.

Los datos que envía el periférico indican el valor de temperatura y humedad en el ambiente; esto se realiza utilizando el módulo DHT22 con el fin de notificar al usuario las condiciones climáticas en la que se encuentra.

# Recepción de datos desde el Hardware

El mecanismo de recepción de datos desde el hardware es mismo utilizado para el envío de datos. Los datos recibidos desde la aplicación en Android, son los que indican la posición de los servomotores. El dato recibido se compone de dos elementos, un carácter que indica el servomotor a moverse, y un número para el ángulo del servomotor.

El otro valor que recibe el periférico, es el valor del acelerómetro para el movimiento rotacional de la base del brazo, y para el movimiento del hombro.

# Pruebas

****

Figura 3.7: Brazo robótico no energizado

****

Figura 3.8: Brazo robótico energizado

****

Figura 3.9: Movimiento del hombro del brazo robótico

****

Figura 3.10: Moviento de codo del brazo robótico

****

Figura 3.11: Movimiento de la muñeca verticalmente del brazo robótico

****

Figura 3.12: Movimiento de la muñeca horizontalmente del brazo robótico

****

Figura 3.13: Rotación de la base del brazo robótico

****

Figura 3.14: Rotación de la base del brazo robótico

****

Figura 3.15: Apertura de la pinza del brazo robótico

# Manual de usuario



Figura 3.16: Manual de Usuario

1. **Botón de conexión:** Éste botón establece la conexión bluetooth con el brazo robótico. En el caso de que la conexión se realizó correctamente, éste mismo sirve para la desconexión.
2. **Indicador de temperatura:** En ésta etiqueta se puede visualizar la temperatura y humedad del ambiente.
3. **Indicador del codo:** Esta barra de desplazamiento indica la posición en la ubicará el servomotor del codo.
4. **Indicador de la muñeca:** Esta barra de desplazamiento indica la posición en la ubicará el servomotor vertical de la muñeca.
5. **Indicador de la pinza:** Esta barra de desplazamiento permite abrir y cerrar la pinza.
6. **Casillero del hombro:** Al seleccionarlo permite subir o bajar el hombro mediante el movimiento del celular (Uso del acelerómetro)
7. **Casillero de la base:** Al seleccionarlo permite girar 360° mediante el movimiento del celular (Uso del acelerómetro)
8. **Botones de la muñeca:** Al ser presionados modifican el ángulo del servomotor horizontal de la muñeca.
9. **Botón de salida:** Al presionarlo cierra automáticamente la aplicación.

# CONCLUSIONES

* Conseguimos diseñar y construir el brazo robótico utilizando servomotores, a pesar de tener problema con el torque de los mismos, puesto que al momento de manipular objetos pesados, existe un desequilibrio en el brazo.
* Se realizó toda la investigación teórica necesaria para construir el proyecto; esto incluye la investigación acerca de servomotores, Arduino, acrílico, AppInventor, Android, etc. Todo esto fue necesario al momento de determinar qué tipo de servomotores utilizar, y que componentes serán necesarios para el brazo robótico.
* Se logró ensamblar toda la estructura del brazo en acrílico, utilizando servomotores en cada articulación y tornillos para las sujeciones del mismo, a pesar de haber tenido varios inconvenientes al momento de atornillar ciertas piezas, puesto que en muchas ocasiones se trizaron o la medida no fue exacta.
* Construimos la aplicación para Android en AppInventor, la cual permite al usuario establecer el ángulo en el que desea que se ubiquen los servomotores. Adicionalmente poseía un botón de conexión con el dispositivo, y una etiqueta que permitía visualizar la temperatura y humedad ambiente.
* Logramos controlar al brazo robótico utilizando un programa realizado en Arduino. Dentro de la programación, se tuvo que modificar en varias ocasiones al programa, para lograr movimientos más fluidos, suavizados y sobre todo delimitados, pero finalmente se obtuvo un movimiento que superó nuestras expectativas
* Se consiguió un funcionamiento correcto en el movimiento de los servomotores utilizando el acelerómetro; sin embargo, este resultó muy poco preciso por lo cual no se utilizó en todas las partes del brazo.
* Uno de los objetivos no alcanzados, fue la implementación en casas de salud. Esto sucedió debido a la falta de tiempo, y la falta de personas interesadas en este producto. Debido a la calidad de los componentes, el precio final del brazo resultó muy elevado, lo cual no resulta conveniente al momento de realizar un producto para casas de salud.

# RECOMENDACIONES

* Se recomienda utilizar servomotores del mismo torque en cada articulación del brazo, esto con la finalidad de que todas las articulaciones del brazo tengan una fuerza similar, y no exista problemas al levantar objetos pesados.
* Se debe prestar mucha atención con los componentes antes de comprarlos. Es muy útil utilizar los conocimientos adquiridos en la investigación para determinar qué tipo de elementos son los más idóneos en el proyecto.
* De igual forma, se debe analizar de forma detallada las medidas y perforaciones que deben tener las piezas de acrílico, antes de trabajar con ellas. Esto podría evitar problemas al momento de armar la estructura.
* Se recomienda realizar una interfaz gráfica muy amigable con el usuario, para que cualquier persona sea capaz de manipular el brazo robótico sin mayor dificultad.
* Realizar pruebas de funcionamiento al momento de realizar la programación en Arduino, es una excelente forma de asegurarnos que el movimiento resulte de la forma esperada, puesto que podría llegar a ser muy rápido o muy brusco, y se debe buscar tener movimientos suaves y fluidos.
* No se debe depender únicamente del acelerómetro, debido a su falta de precisión. A pesar de ser una herramienta muy novedosa y que ayuda a realizar movimientos más fácilmente, puede resultar incómodo al momento de que el usuario controle el dispositivo.
* Una sugerencia es la de utilizar materiales de menor precio, para lograr construir un dispositivo similar pero más económico. Una construcción masiva de este producto podría significar la implementación de forma más sencilla en clínicas y casas de salud.

# REFERENCIAS BIBLIOGRÁFICAS

Artero, Ó. T. (2013). Arduino: Curso práctico de formación. Madrid: RC Libros.

Autero, Ó. T. (2013). Arduino: curso practico de formación. Madrid, España: Grupo RC.

Francisco T. Sánchez Marín, A. P. (2006). Mantenimiento Mecánico de Máquinas. Castellón de la Plana: Publicacions de la Universitait Jaume.

Harper, G. E. (2000). El libro práctico de los generadores, transformadores y motores eléctricos. Mexico D.F.: Limusa.

Maloney, T. J. (2006). Electrónica industrial moderna. México: Pearson Educación.

Mann, M. (2012). Introducción a Android. Mikel Angoar.

Robert L. Boylestad, L. N. (2003). Electrónica: teoría de circuitos y dispositivos electrónicos. México: PEARSON EDUCACIÓN.

Robert Thornton Morrison, R. N. (1987). Química Orgánica. Mexico: Pearson.

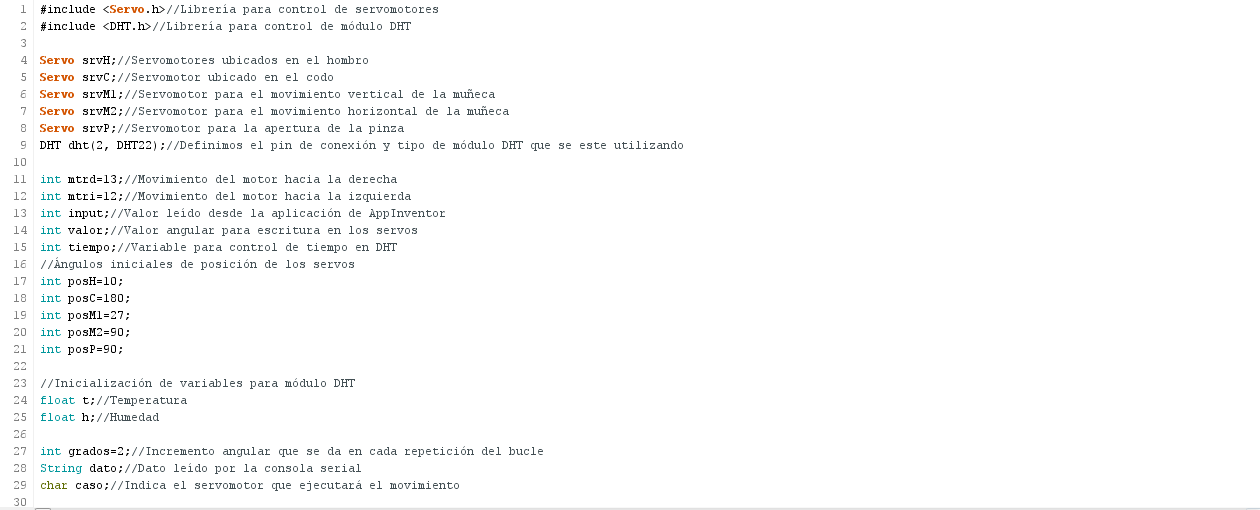
Senner, A. (1994). Principios de electrotecnia. En A. Senner, *Principios de electrotecnia* (pág. 28). Barcelona: Reverté.

Viloria, J. R. (2014). Motores de Corriente Continua. Madrid: Paraninfo.

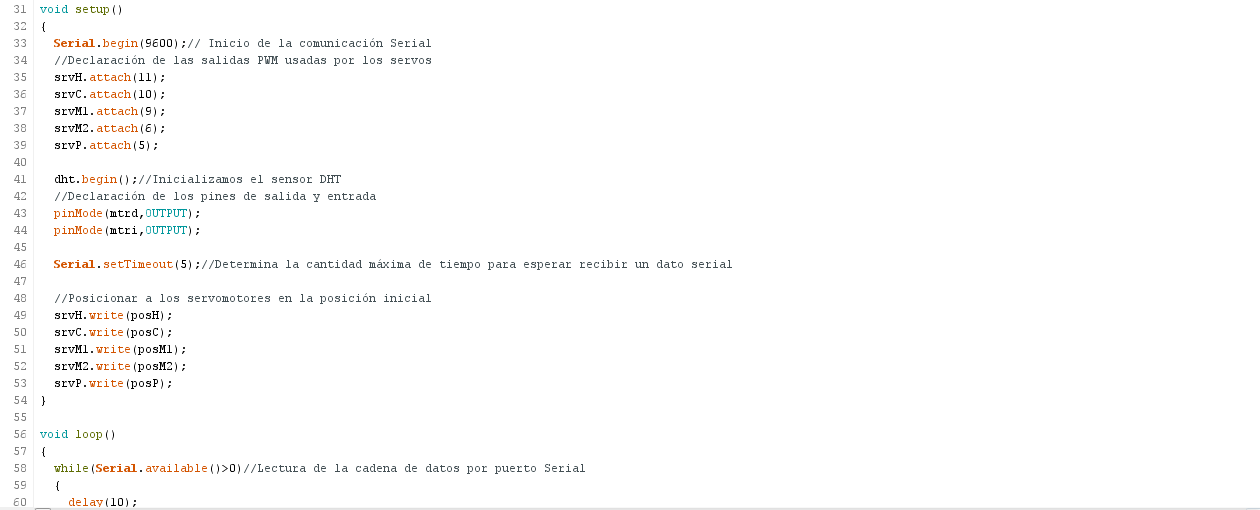
# 

# ANEXOS

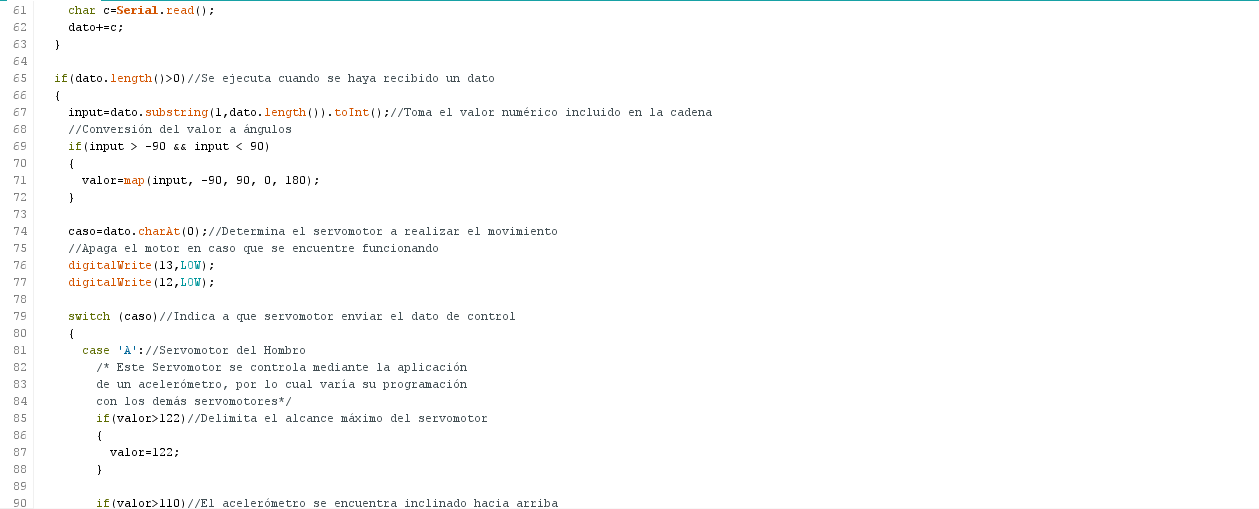
Anexo 1: Codificación Arduino



Anexo 2: Codificación Arduino



Anexo 3: Codificación Arduino



Anexo 4: Codificación Arduino



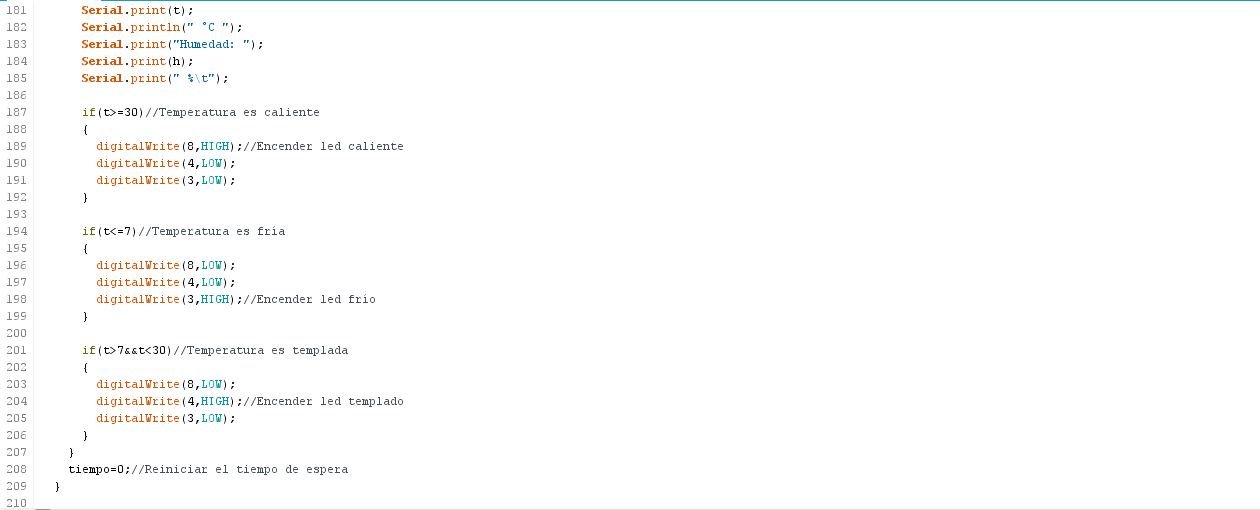
Anexo 5: Codificación Arduino



Anexo 6: Codificación Arduino



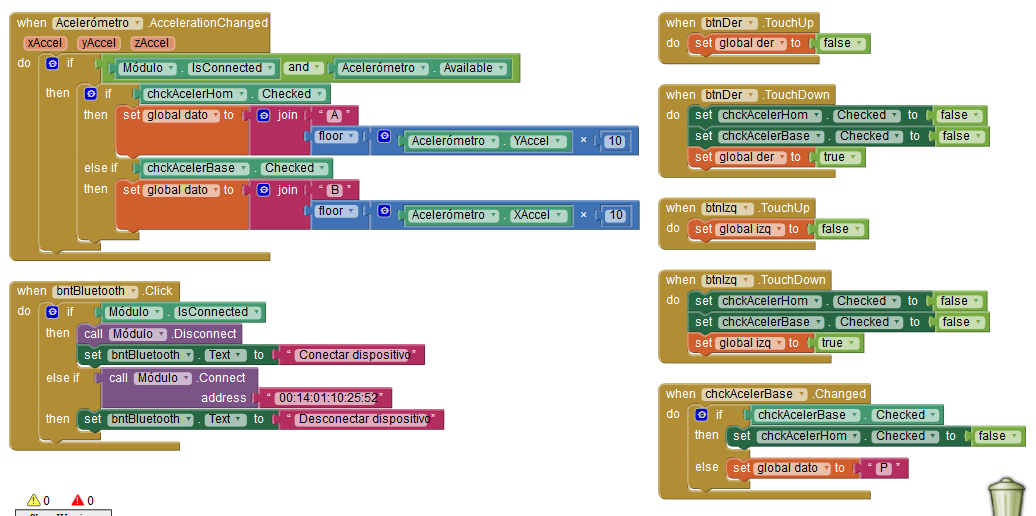
Anexo 7: Codificación Arduino



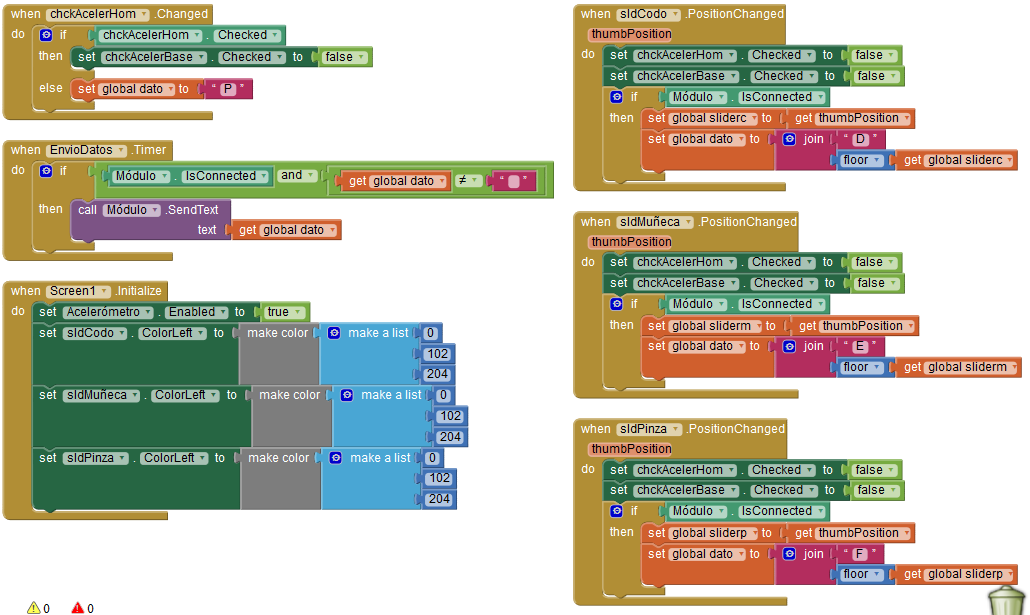
Anexo 8: Codificación Arduino

http://i.gyazo.com/9ceb1758ee161fb73db6c577dbd5daae.png

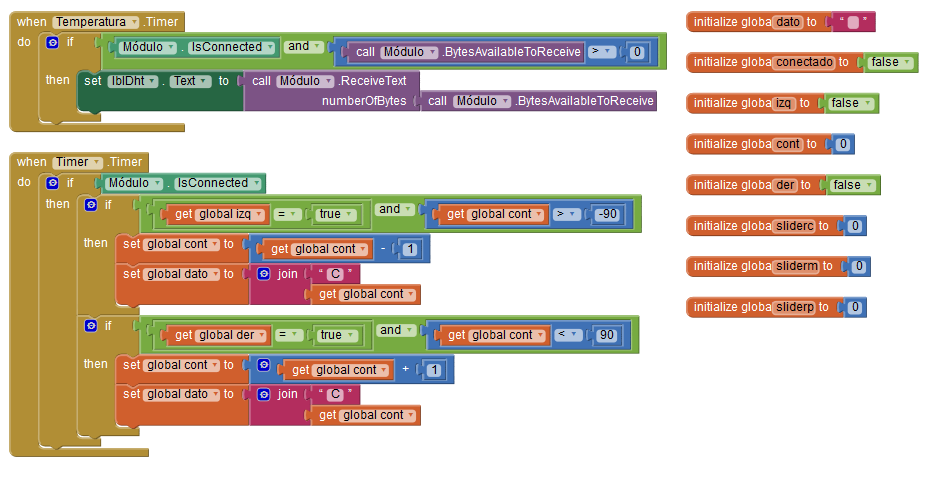
Anexo 9: Codificación AppInventor



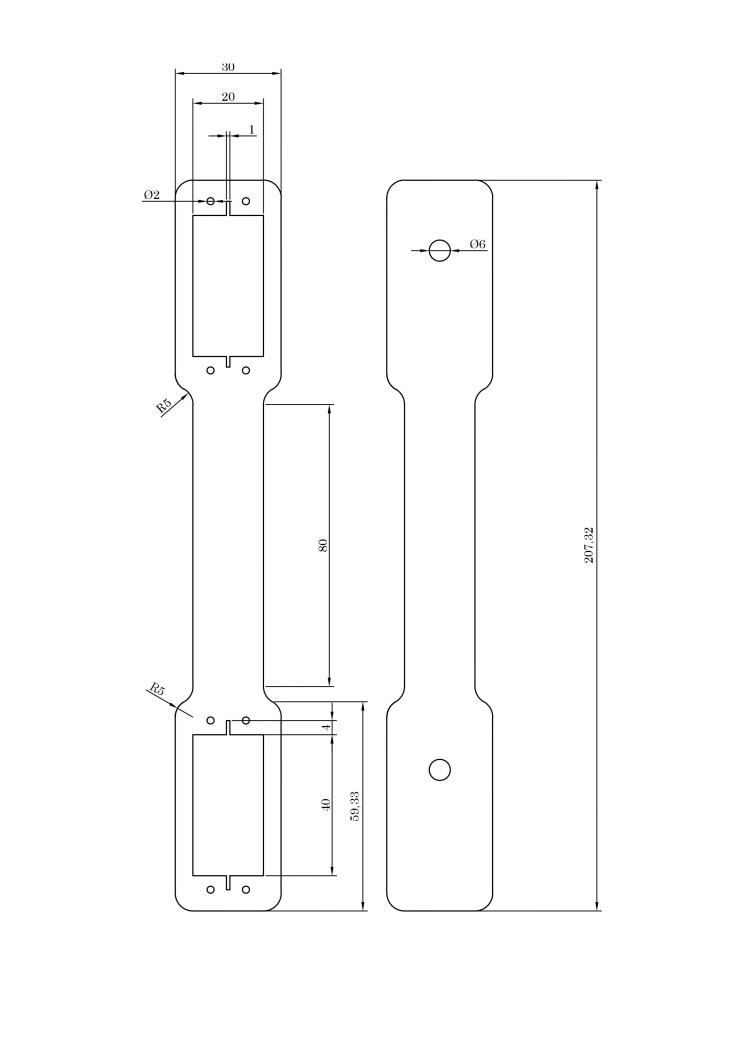
Anexo 10: Codificación AppInventor



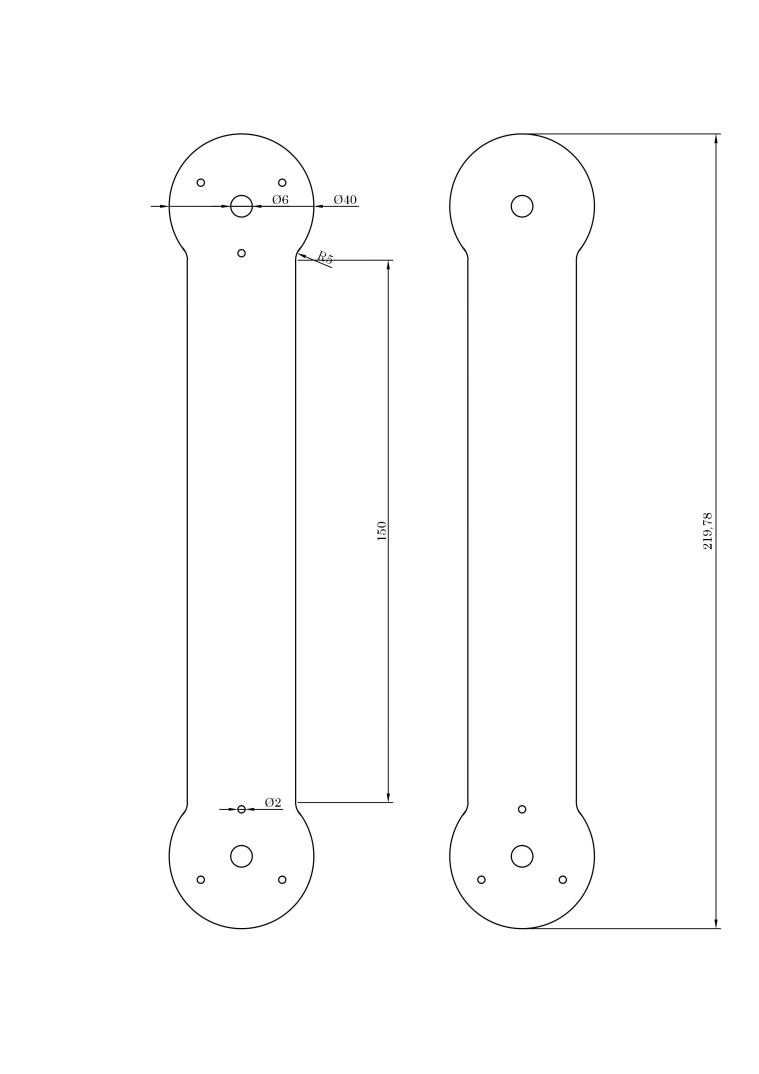
Anexo 11: Codificación AppInventor



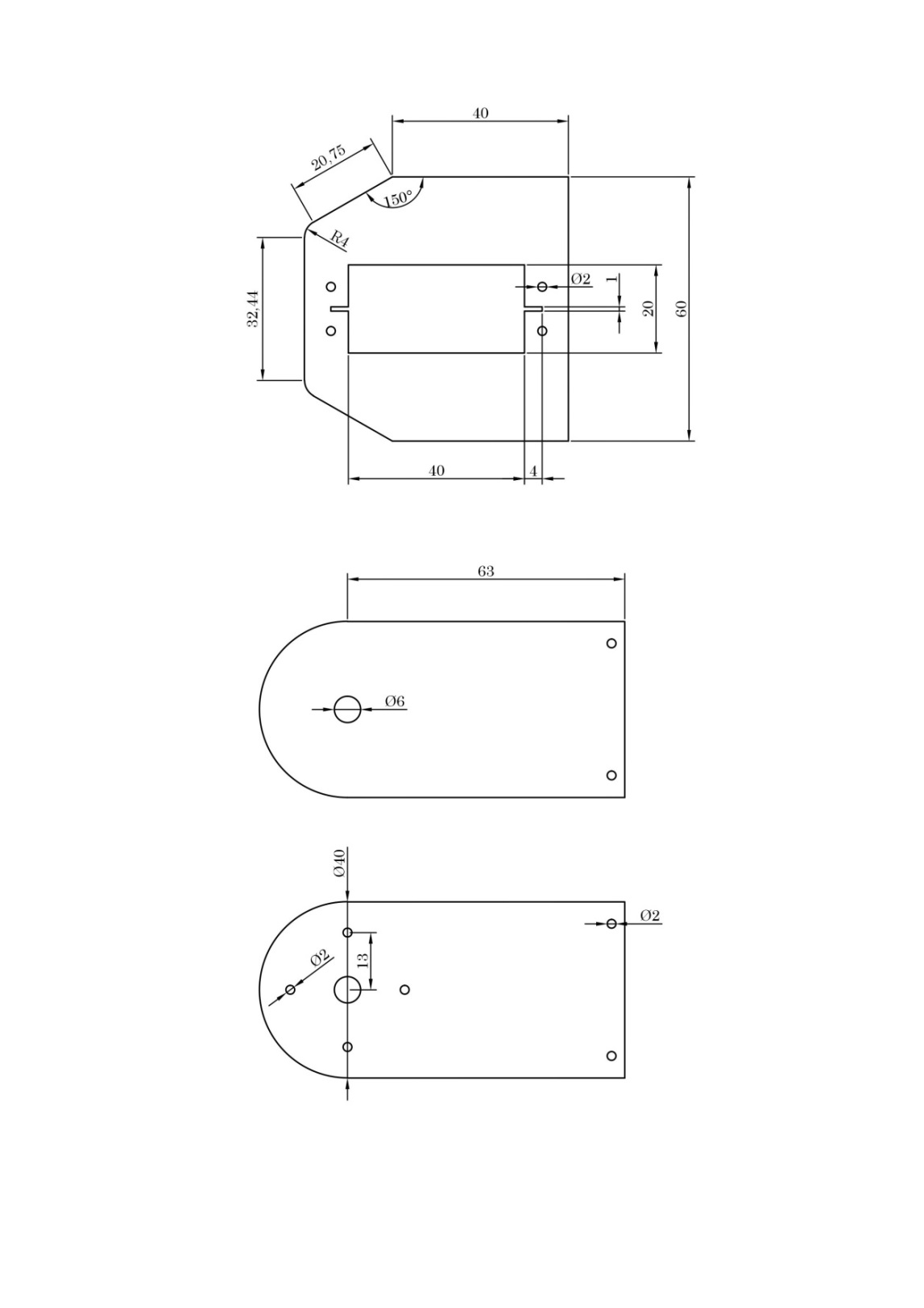
Anexo 12: Diseño pieza antebrazo



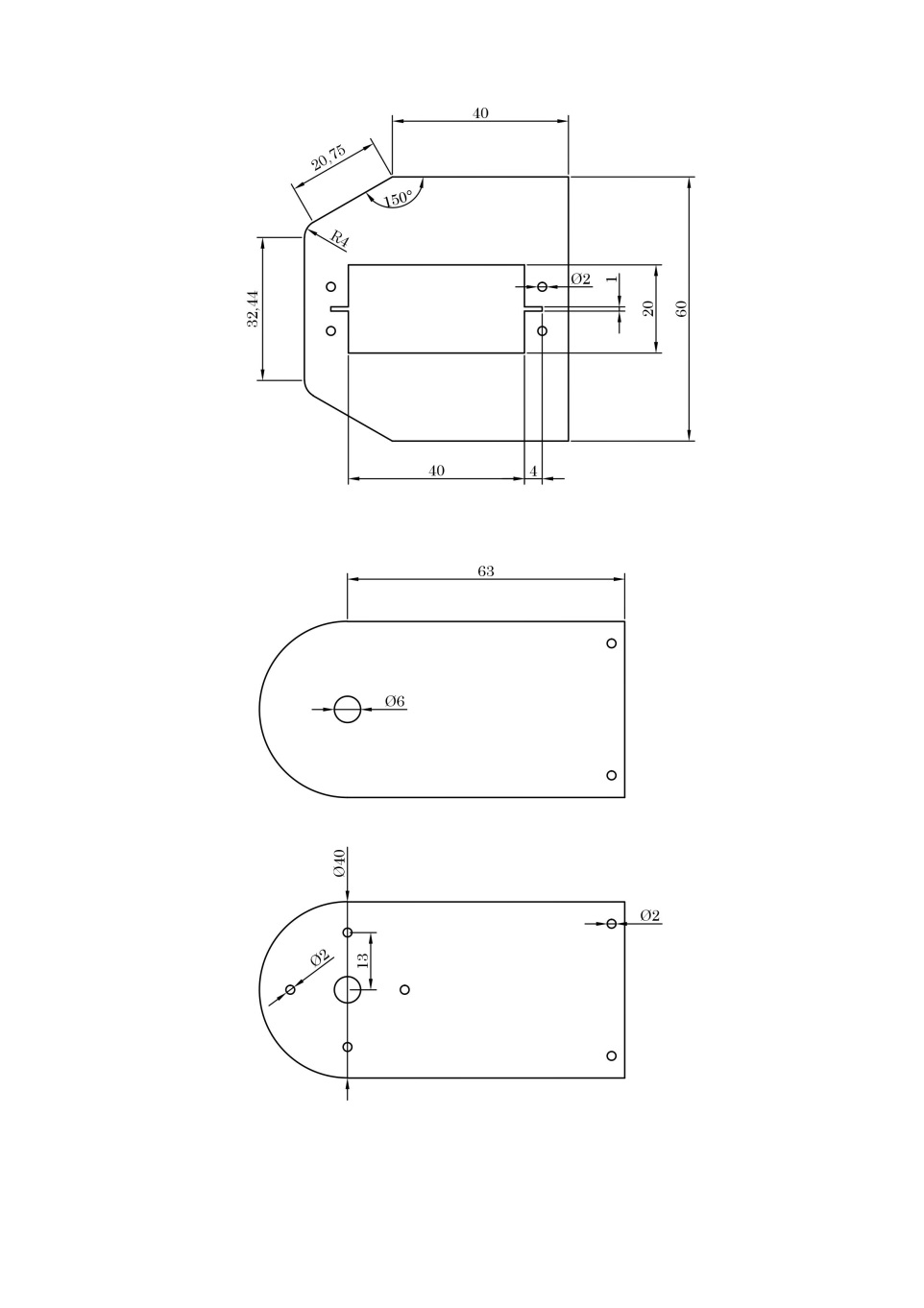
Anexo 13: Diseño pieza brazo



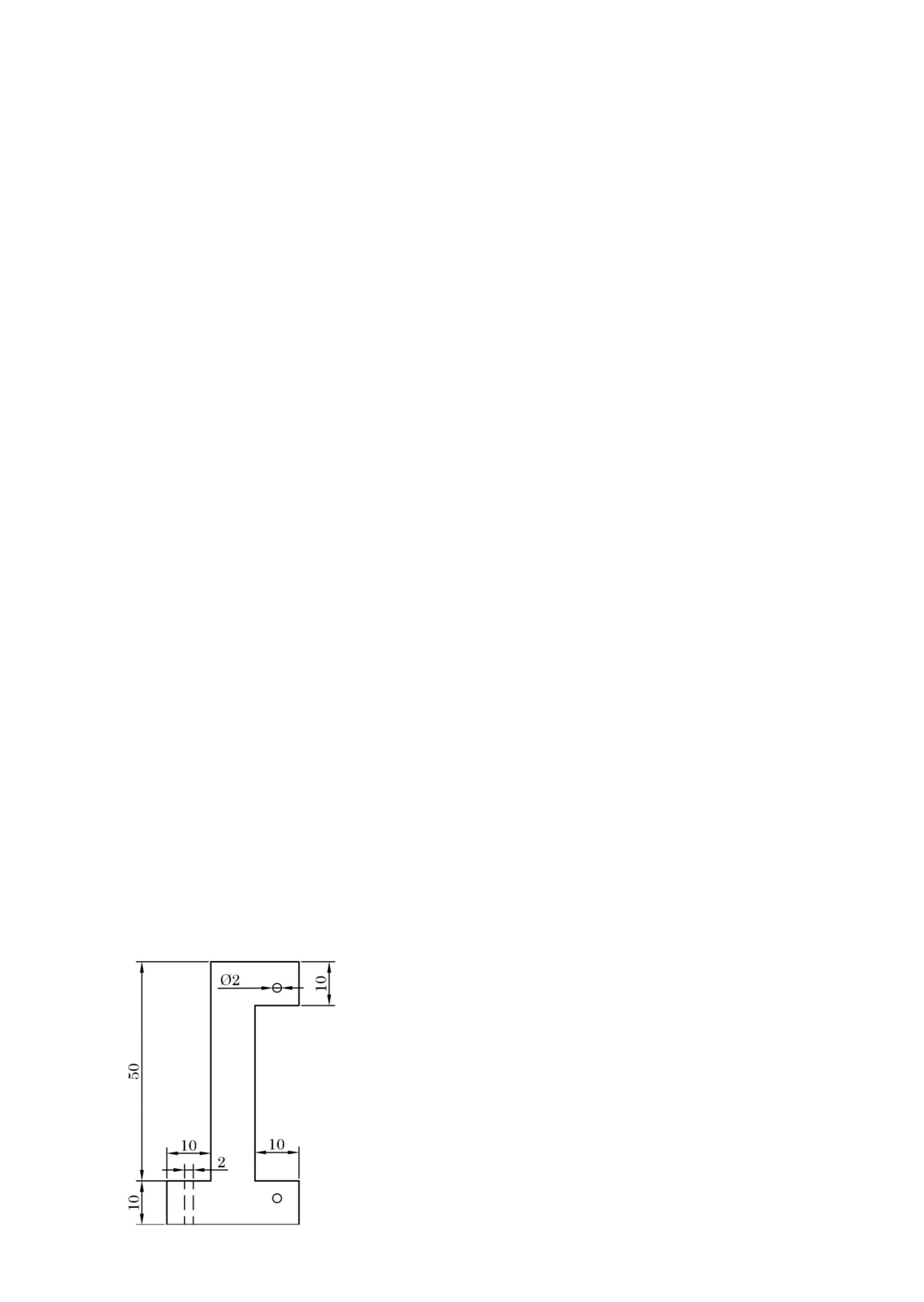
Anexo 14: Diseño pieza muñeca horizontal



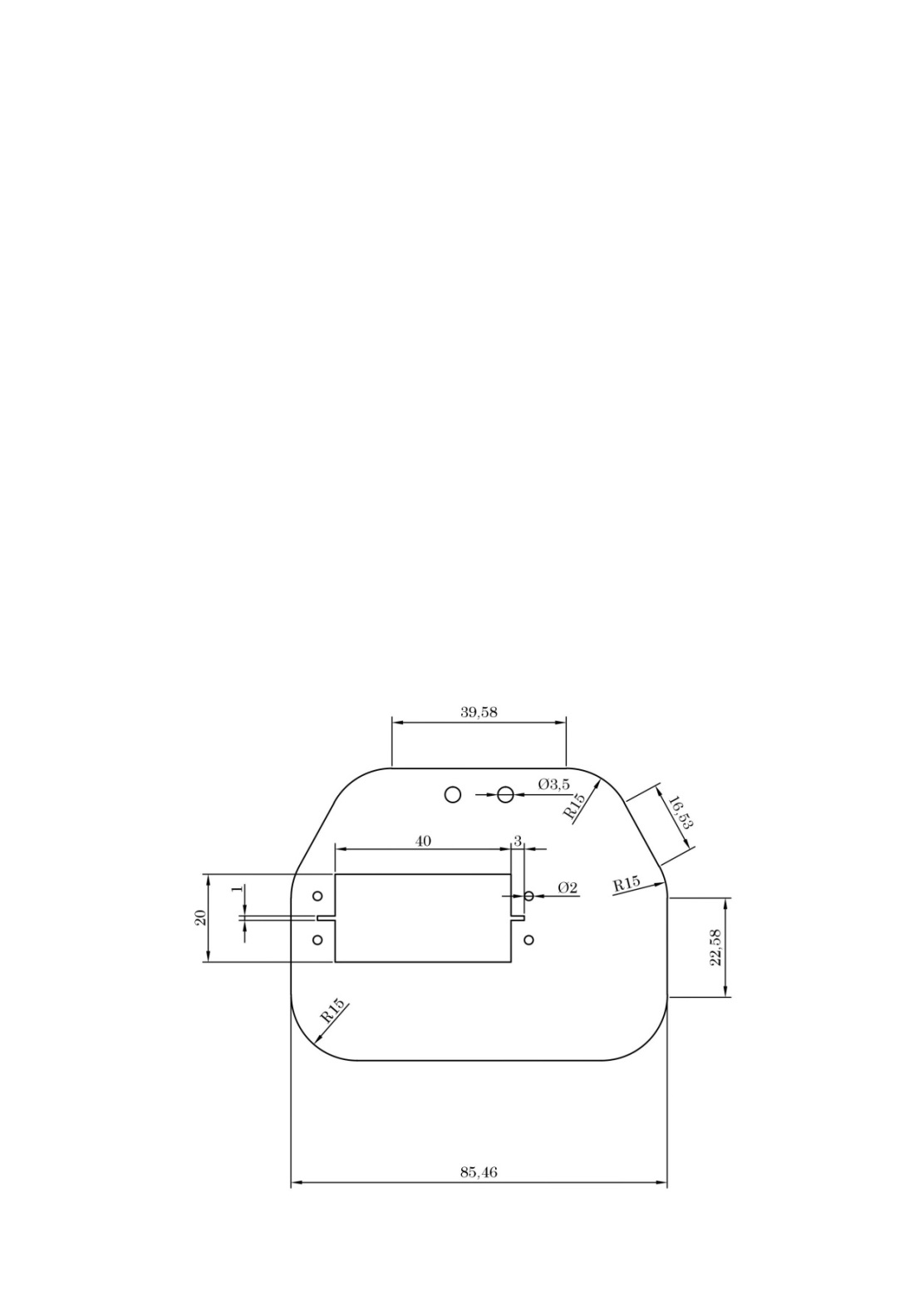
Anexo 15: Diseño pieza muñeca vertical



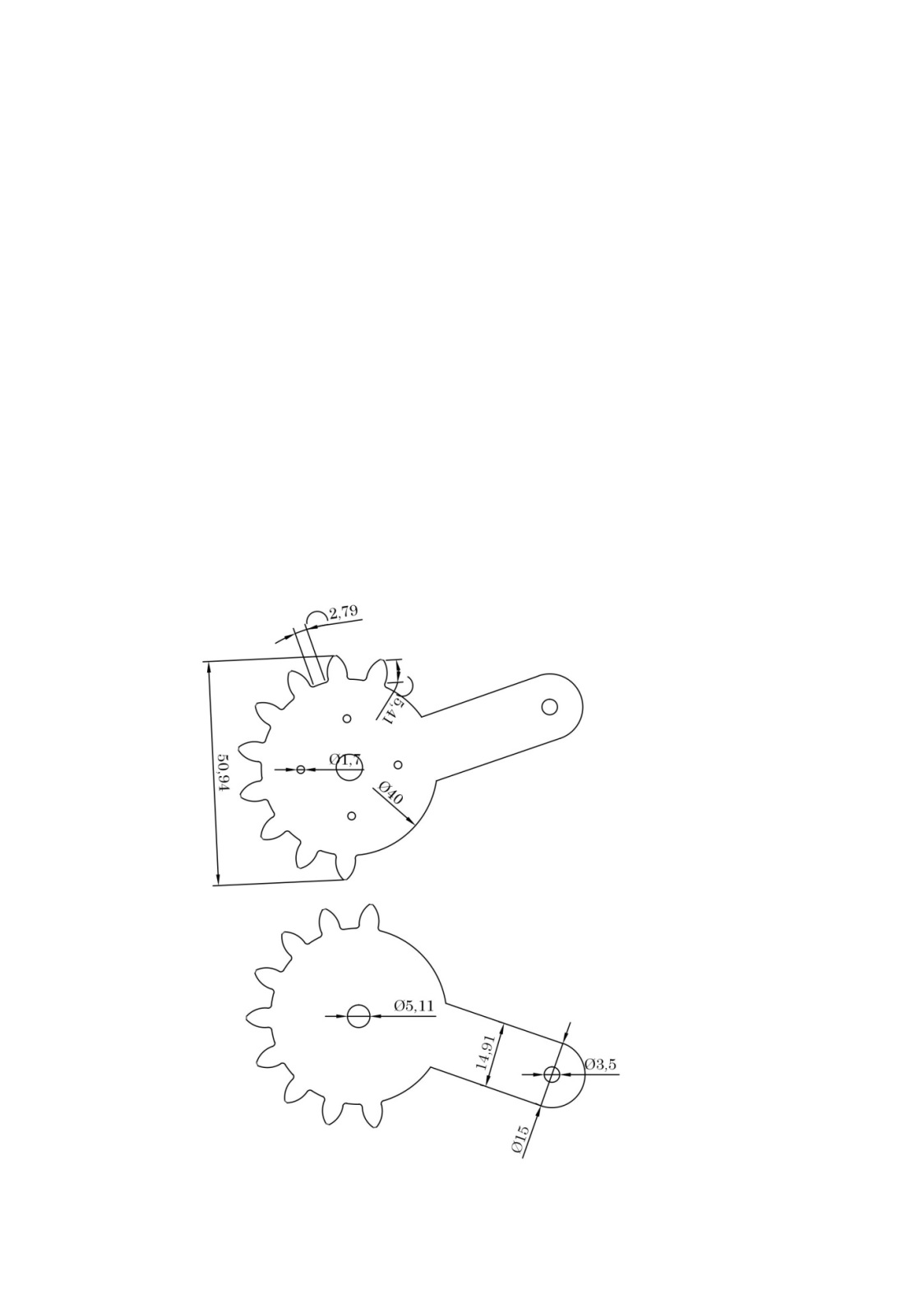
Anexo 16: Diseño pieza hombro



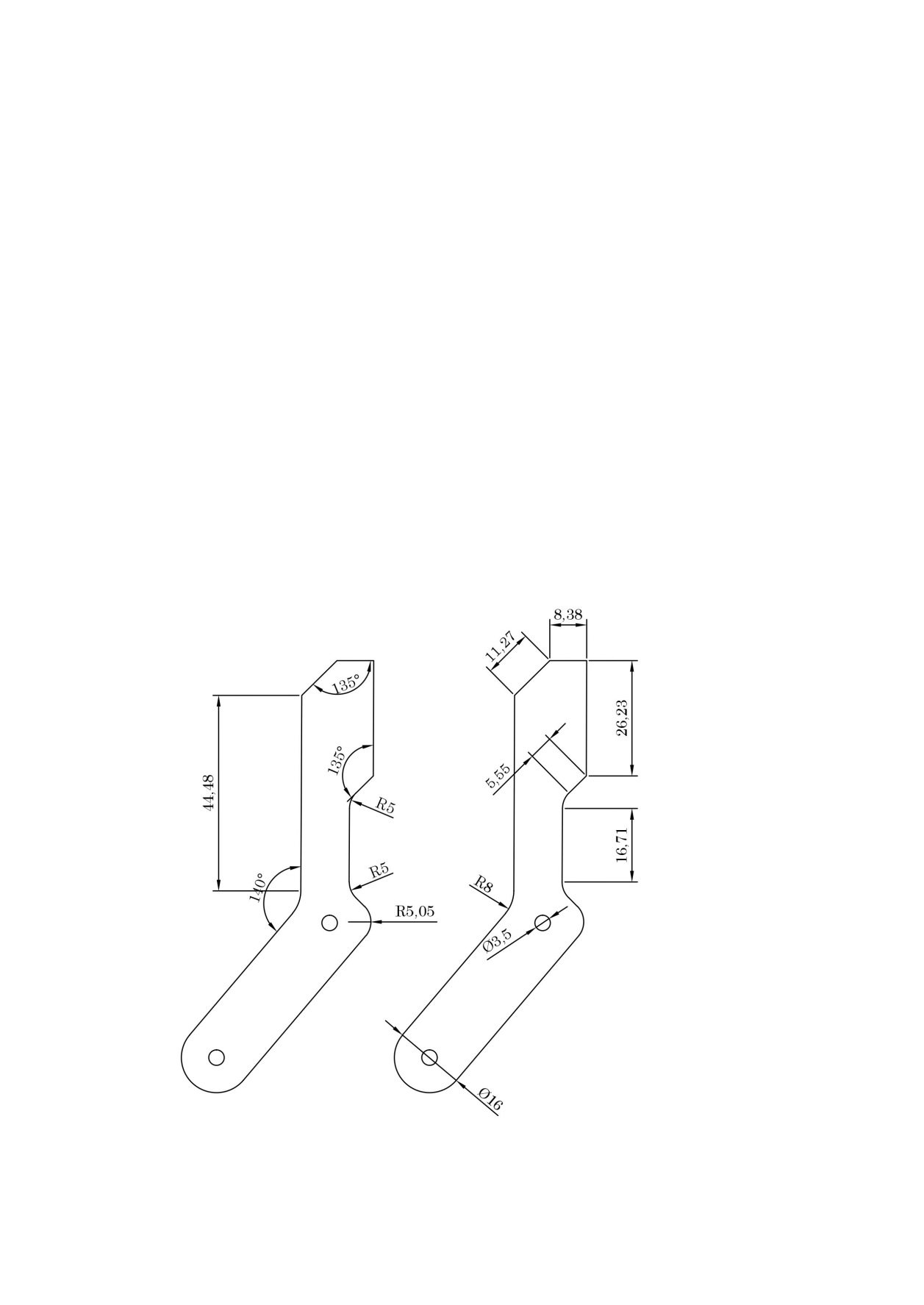
Anexo 17: Diseño pieza base de la pinza



Anexo 18: Diseño piezas engranes de la pinza



Anexo 19: Diseño piezas puntas de la pinza



Anexo 20: Diseño pinza completa

