



**UNIVERSIDAD TECNOLÓGICA ISRAEL**

**TRABAJO DE TITULACIÓN EN OPCIÓN AL GRADO DE:**

**INGENIERO EN ELECTRÓNICA DÍGITAL Y TELECOMUNICACIONES**

**TEMA: IMPLEMENTACIÓN DE UN PROTOTIPO DE MANO ROBÓTICA  
CON MATERIALES INTELIGENTES**

**AUTOR: VÍCTOR HUGO LAVERDE PAUCAR**

**TUTOR: Mg. RENÉ CORTIJO LEYVA**

**AÑO: 2018**

## DECLARACIÓN

Yo, Víctor Hugo Laverde Paucar, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.



.....  
Víctor Hugo Laverde Paucar

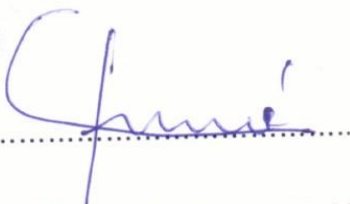
## APROBACIÓN DEL TUTOR

En mi calidad de tutor del trabajo de titulación certifico:

Que el trabajo de titulación "**IMPLEMENTACIÓN DE UN PROTOTIPO DE MANO ROBÓTICA CON MATERIALES INTELIGENTES**", presentado por el Sr. Víctor Hugo Laverde Paucar, estudiante de la carrera de Electrónica Digital y Telecomunicaciones, reúne los requisitos y méritos suficientes para ser sometido a la evaluación del Tribunal de Grado, que se designe, para su correspondiente estudio y calificación.

Quito, Febrero 10 de 2018

TUTOR



Ing. Rene Ernesto Cortijo Leyva, Mg

## **DEDICATORIA**

El proyecto de titulación le dedico a mi esposa y familia por el apoyo brindado en mi formación profesional. A mi madre y padre les agradezco por el esfuerzo realizado para que su hijo tenga una carrera profesional. A mi esposa le agradezco por el apoyo, dedicación y paciencia que ha tenido tanto en mi formación académica, como formación profesional, y finalmente con cariño se lo dedico a Dios por brindarme una grandiosa familia.

VÍCTOR

## **AGRADECIMIENTO**

Este proyecto de titulación requirió de mucha esfuerzo, tiempo y paciencia para su finalización, es por estos motivos que agradezco a mi esposa Veronica Castillo por tener estas cualidades que me ayudo para perseverar en este objetivo de vida.

VÍCTOR

## CONTENIDO

DECLARACIÓN .....	ii
APROBACIÓN DEL TUTOR.....	iii
DEDICATORIA .....	iv
AGRADECIMIENTO.....	v
CONTENIDO .....	vi
ÍNDICE DE FIGURAS .....	x
ÍNDICE DE TABLAS .....	xiii
RESUMEN.....	xv
ABSTRACT .....	xvii
INTRODUCCIÓN .....	xix
Problema de investigación: presentación y argumentación .....	xix
Estado de arte de prototipos de manos robóticas usando aleaciones de memoria de forma .....	xix
Justificación.....	xxi
Objetivos del trabajo de titulación .....	xxi
Objetivo general .....	xxi
Objetivos específicos .....	xxi

DESCRIPCIÓN DE LOS CAPÍTULOS.....	xxii
CAPÍTULO I. FUNDAMENTACIÓN TEÓRICA.....	1
1.1 Fundamentos de aleaciones de memoria de forma SMA (shape-memory alloy) .....	1
1.1.1 Definición.....	1
1.1.2 Propiedad de efecto de memoria de forma de los SMA.....	1
1.1.3 Propiedad de súper elasticidad en los SMA .....	4
1.1.4 Tipos de memoria de forma .....	5
1.1.5 Entrenamiento de los SMA .....	7
1.1.6 Movimientos de los SMA .....	8
1.1.7 NITINOL.....	9
1.1.8 Control de memoria de forma de Nitinol. ....	12
1.1.9 Resortes de aleaciones de memoria Nitinol .....	13
1.1.10 Aleación de memoria de forma escogida .....	15
1.2 Bandas ISM.....	18
1.3 Bluetooth.....	19
1.3.1 Normas Bluetooth .....	19
1.4 Módulos Bluetooth HC-05 .....	20

1.4.1 Características de módulo bluetooth HC-05 .....	20
1.5 Módulo Arduino Nano .....	21
1.6 Sensor de fuerza FSR 402 .....	23
1.7 Sensor de temperatura LM35 .....	24
1.8 Modulación por ancho de pulso PWM.....	25
1.9 Módulo PCA9685 PWM 16 canales .....	25
CAPÍTULO II. PROPUESTA .....	27
2.1 Funcionamiento del prototipo de mano robótica con materiales inteligentes.....	27
2.2 Conformación del sistema de control del prototipo de mano robótica con materiales inteligentes .....	28
2.3 Diseño del circuito de control electrónico .....	28
2.4 Dimensionamiento de fuente.....	32
2.5 Sistema mecánico.....	33
2.6 Diseño de software.....	38
2.7 Comunicación inalámbrica Bluetooth.....	40
2.8 Diseño de programa de módulo Arduino .....	41
CAPÍTULO III. IMPLEMENTACIÓN .....	44
3.1 Desarrollo.....	44



3.1.1 Fuente de alimentación .....	44
3.1.2 Sistema de potencia.....	45
3.1.3 Circuitos acondicionadores de sensores .....	47
3.1.4 Conexiones al módulo Arduino Nano .....	48
3.1.5 Programación de módulo Bluetooth HC-05.....	51
3.1.6 Programación módulo Arduino Nano v3 .....	53
3.1.7 Programación de aplicación de dispositivo Smartphone en APP Inventor 2.....	56
3.2 Implementación.....	61
<b>CAPÍTULO IV PRUEBAS Y ANÁLISIS DE RESULTADOS .....</b>	<b>65</b>
4.1 Pruebas de funcionamiento .....	65
4.2 Presupuesto .....	66
4.3 Análisis de resultados.....	69
<b>CAPÍTULO V CONCLUSIONES Y RECOMENDACIONES .....</b>	<b>71</b>
Conclusiones .....	71
Recomendaciones.....	72
Trabajos futuros .....	74
<b>REFERENCIAS BIBLIOGRÁFICAS .....</b>	<b>75</b>

ANEXOS.....	79
-------------	----

## ÍNDICE DE FIGURAS

Figura 1.1 Fases de cambio de los SMA.....	2
Figura 1.2 Ciclo de histéresis de los SMA.....	4
Figura 1.3 Propiedad de Súper elasticidad en los SMA.....	5
Figura 1.4 Tipos de memoria de forma.....	6
Figura 1.5 Tipos de movimientos de los SMA.....	9
Figura 1.6 Circuito generador de PWM con integrado 555.....	13
Figura 1.7 Resorte de aleación de memoria de forma Nitinol.....	13
Figura 1.8 Ejemplo de relación de estiramiento.....	15
Figura 1.9 Distribución de canales en la banda ISM.....	18
Figura 1.10 Módulo Bluetooth HC-05.....	20
Figura 1.11 Diagrama de conexión de Arduino Nano.....	22
Figura 1.12 Esquema de sensor de fuerza FSR402.....	24
Figura 1.13 Sensor de temperatura LM35.....	25
Figura 1.14 Señal PWM de módulo PCA9685.....	26
Figura 1.15 Módulo PWM PCA9685.....	26

Figura 2.1 Diagrama de bloque del funcionamiento del prototipo de mano robótica....	28
Figura 2.2 Esquema electrónico de opto-acoplador 4N25 .....	29
Figura 2.3 Circuito de polarización de MOSFET IRFZ44N.....	31
Figura 2.4 Articulaciones de modelo ThingVerse .....	34
Figura 2.5 Estructura de articulaciones de la mano .....	34
Figura 2.6 Piezas que conforman estructura de antebrazo .....	35
Figura 2.7 Estructura de antebrazo.....	35
Figura 2.8 Estructura de antebrazo superior.....	36
Figura 2.9 Mecanismo de resorte de Nitinol en serie.....	36
Figura 2.10 Estructura de fijación de resortes de Nitinol.....	37
Figura 2.11 Movimiento de un dedo .....	38
Figura 2.12 Diagrama de flujo de comunicación Bluetooth .....	39
Figura 2.13 Diagrama de flujo de configuración de módulo Bluetooth .....	40
Figura 2.14 Diagrama de flujo de lectura de sensores .....	42
Figura 2.15 Diagrama de flujo de funcionamiento de mano robótica.....	43
Figura 3.1 Fuente de alimentación .....	44
Figura 3.2 Regulador de voltaje a 5 voltios .....	45

Figura 3.3 Módulo PCA9685 .....	46
Figura 3.4 Circuito de conmutación de corriente .....	47
Figura 3.5 Circuito de acondicionador de sensores .....	48
Figura 3.6 Esquemático del módulo Arduino Nano en Proteus .....	49
Figura 3.7 Esquemático del módulo bluetooth HC-05 en Proteus .....	49
Figura 3.8 Circuito del sistema de control de la mano robótica con materiales inteligentes .....	50
Figura 3.9 Selección de placa Arduino .....	51
Figura 3.10 Programa para comunicación serial con Bluetooth HC-05 .....	52
Figura 3.11 Comunicación Serial .....	52
Figura 3.12 Programación de inicialización de librerías .....	53
Figura 3.13 Programación de acondicionamiento de señales de sensores .....	54
Figura 3.14 Programación de selección y generación de movimiento .....	55
Figura 3.15 Aplicación APP inventor 2 .....	56
Figura 3.16 Programación de Screen 1 .....	57
Figura 3.17 Aplicación de prototipo de mano robot .....	57
Figura 3.18 Programación de establecimiento de comunicación inalámbrica .....	59
Figura 3.19 Programación de selección de movimiento .....	60

Figura 3.20 Programación de activación de movimiento.....	61
Figura 3.21 Programación para detener movimiento.....	61
Figura 3.22 Diseño de placa impresa de la fuente de alimentación.....	62
Figura 3.23 Diseño de placa de circuito de potencia.....	62
Figura 3.24 Circuito impreso de fuente de alimentación lado A.....	63
Figura 3.25 Circuito impreso de fuente de alimentación lado B.....	63
Figura 3.26 Circuito impreso de Potencia.....	63
Figura 3.27 Prototipo de mano robótica ensamblado.....	64

## ÍNDICE DE TABLAS

Tabla 1.1 Características del Nitinol.....	11
Tabla 1.2 Características de los resortes de memoria de forma Nitinol.....	14
Tabla 1.3 Características módulo <i>Bluetooth</i> HC-05.....	21
Tabla 1.4 Características de módulo Arduino Nano V3.....	22
Tabla 1.5 Características de módulo PCA9685.....	26
Tabla 2.1 Consumo de energía total del prototipo de mano robótica.....	32
Tabla 3.1 Comandos AT configurados.....	53
Tabla 4.1 Pruebas de funcionamiento.....	66

Tabla 4.2 Costos de materiales.....	68
Tabla 4.3 Costos de importación.....	68
Tabla 4.4 Costo de mano de obra.....	68

## RESUMEN

Para implementar un prototipo de prótesis de mano robótica que genere movimientos articulados, se utilizaron materiales inteligentes de memoria que son elementos formado por la combinación de Níquel y Titanio que tienen propiedades de memoria de forma; dichos materiales inteligentes permitirán asemejarse a los tendones o músculos de la mano y poder generar movimientos casi similares a los movimientos naturales del ser humano en el instante que circule una corriente eléctrica.

Para hacer uso de las propiedades de los materiales inteligentes en la prótesis de mano se tendrá que realizar un aprendizaje de memoria de forma, al someter el material a esfuerzos mecánicos mientras se aplica variaciones de temperatura conocidos como las fases de “Martensita” y “de Austenita”.

Finalizado el proceso de aprendizaje de memoria de forma del material inteligente, se controlará la forma que adoptará a través de un circuito de control mediante opto-acopladores y transistores MOSFET que permita controlar la corriente sobre el actuador sin que genere desgaste en el mismo, de igual manera se puede plantear el uso de un módulo PWM de 16 salidas que permitan controlar la gran cantidad de actuadores a usarse en la prótesis y simplificar espacio físico.

El control electrónico estará a cargo del módulo Arduino Nano V3 y en conjunto con el módulo PCA9685, los cuales se encargarán de controlar y generar una señal PWM que permita el funcionamiento del material inteligente

Una vez generado el movimiento de las articulaciones, se pretende usar sensores de fuerza FSR402 que permitan medir la fuerza ejercida a los objetos, y sensores de temperatura para medir las condiciones del material inteligente al encontrarse en un medio cerrado y que no provoque daños en el mismo.

Los datos obtenidos de fuerza y temperatura serán transmitidos inalámbricamente con módulos *Bluetooth* hacia un ordenador que permita visualizar mediante una aplicación las

condiciones de la prótesis y mandar órdenes de funcionamiento de la mano acorde a los movimientos definidos.

Una vez generado el movimiento de las articulaciones se realizará 5 tipos de movimientos: mano abierta – cerrada, movimiento individual de cada dedo, agarre cilíndrico, agarre de gancho y agarre lateral.

Se realizó pruebas del prototipo finalizado para corrección de problemas.



## ABSTRACT

The implementation of a prototype of robotic hand prosthetic that generates articulated movements, intelligent materials are used that are elements formed by the combination of Nickel and Titanium that have shape memory properties. These intelligent materials make it possible to resemble the tendons or muscles of the hand and to generate movements almost similar to the natural movements of the human being, in the moment that an electric current go through it.

To make use of the properties of the intelligent materials in the hand prosthesis will have to perform a shape memory training, by subjecting the material to mechanical stress while applying temperature variations known as the phases of "Martensite" and "of Austenite".

Once the learning process of shape memory of the intelligent material has been completed, the form that it will adopt through a control circuit will be controlled by means of optocouplers and MOSFET transistors that allow to control the current on the actuator without generating wear on it. In the same way, the use of a PWM module with 16 outputs can be considered, which allows controlling the large number of actuators to be used in the prosthesis and simplifying physical space.

The electronic control will be in charge of the Arduino Nano V3 module and the PCA9685 module, it will be in charge of controlling and generating a PWM signal that allows the operation of the intelligent material.

Once generated the movement of the articulations that intends to use force sensors FSR402 that allows to measure the force exerted to the objects, to use sensors of temperature to measure the conditions of the intelligent material when being in a closed means and that does not cause damages in the same.

The data obtained from the force and the temperature are transmitted wirelessly with *Bluetooth* modules to a computer that allows to visualize by means of an application the conditions of the prosthesis and the operating instructions of the hand with the defined movements.

Once the movement of the joints is generated, 5 types of movements will be performed: open - closed hand, individual movement of each finger, cylindrical grip, hook grip and lateral grip.

Tests of the finished prototype will be performed to correct problems.

**KEYWORDS:** Nitinol, SMA, Smart Material, Robotic Hand, Prosthesis

# INTRODUCCIÓN

## **Problema de investigación: presentación y argumentación**

Se ha realizado una búsqueda dentro del mercado ecuatoriano y se ha identificado que no se han planteado diseños electrónicos para el uso de alternativas de materiales o mecanismo en el desarrollo de prótesis automatizados, que generen nuevas líneas de investigación. Actualmente el uso de prótesis automatizadas se basa en movimientos generados por elementos mecánicos como servomotores que pueden o podrían limitar la movilidad, incrementar el peso de la prótesis y generar un producto que no esté al alcance de la sociedad lo cual provoca problemas sociales relacionados a la movilidad y salud.

Se pretende con este trabajo desarrollar una mano robótica que permita generar movimientos articulados mediante el uso de materiales inteligentes y pueda ser controlado inalámbricamente por un dispositivo móvil.

## **Estado de arte de prototipos de manos robóticas usando aleaciones de memoria de forma**

Uno de los trabajos realizados previamente corresponde a (Villoslada Peciña, 2010), quien realizó la propuesta de “Diseño y aplicación de un actuador SMA en el control de manos robóticas”. En este proyecto se realizó el diseño de un dedo en metacrilato en el cual se fijó los alambres de las aleaciones de memoria de forma y se realizó pruebas de funcionamiento con diferentes tipos arreglos mecánicos y sistemas de control, de igual manera se realizaron pruebas de funcionamiento al exponer las aleaciones a un sistema de refrigeración. [1]

Este estudio fue considerado como éxito en sus objetivos iniciales referente al diseño de un prototipo de sistema de actuación mediante el uso de aleaciones de memoria de forma; acorde a las pruebas realizadas se pudo concretar que un sistema PDI es ineficaz debido a la no linealidad de las materiales inteligentes. En pruebas realizadas con los sistemas de refrigeración se validó que el aceite puede considerarse un buen refrigerante, pero en el uso constante se identificó que el refrigerante eleva su temperatura provocando el comportamiento erróneo de

las aleaciones de memoria de forma. Otras de las falencias detectadas en el sistema mecánico del dedo es que las aleaciones de memoria de forma pueden desacoplarse del sistema de polea, de igual manera demostraron que en el calibre utilizado tiene inconvenientes en la fuerza requerida para la contracción del dedo.

Otro de los trabajos realizados con aleaciones de memoria de forma se detallan en el artículo *Robótica y Prótesis Inteligentes: Una segunda oportunidad de vida* realizado por (Le Vinson Aguilar, 2012), donde se hace mención el estudio realizado del diseño, simulación e implementación de una prótesis de mano robótica con el uso de aleaciones de memoria de forma, por estudiante de la UNAM durante el año 2004.

En este proyecto se diseñó y construyó una mano robótica conformada por un total de 16 piezas y actuadores que permitirían controlar los alambres de memoria de forma, lamentablemente el proyecto fue declarado fallido, en las pruebas y análisis realizados observaron que los alambres SMA recibían el pulso eléctrico necesario, pero el material inteligente no fue capaz de generar ningún tipo de movimiento, y en el peor de los casos el material inteligente incrementó su temperatura hasta el punto de provocar daños en la estructura. Finalmente en el proyecto se concluyó que es necesario realizar un mayor estudio y experimentación para utilizar las aleaciones de memoria en prótesis. [2]

La universidad Saarland en Alemania (ASM International, 2015), se encargó de realizar el proyecto *“Bionic hand based on bundled Nitinol wires activated by electricity”* (Mano biónica basada en alambres de Nitinol agrupados activados por electricidad); el cual es uno de los mejores proyectos realizados; La estructura de la mano biónica fue realizado mediante una impresión 3D, el movimiento de los dedos se encuentra realizado por un grupo de alambres de memoria de forma, colocados en paralelo con el fin de generar mayor fuerza y el control de las aleaciones de memoria de forma fue realizado mediante un sistema de control de posición mediante la variación resistiva del material. Lamentablemente no se han hecho publicaciones de mejoramientos del prototipo realizado con el fin de llevarlo a ser un producto comercial. [3]

Con la información expuesta, el presente proyecto tiene como finalidad la creación de una mano robótica mediante el uso de resortes de aleaciones de memoria de forma, con el objetivo

de evaluar el funcionamiento de los materiales inteligentes y poder tener un prototipo base sobre el cual pueda partir investigaciones de desarrollo de nuevas tecnologías.

## **Justificación**

Implementar un prototipo de prótesis de mano robótica, que genere movimientos articulados mediante el uso de actuadores basados en materiales inteligentes podría permitir generar movimientos semejantes al ser humano y tener una presentación más amigable para una persona con discapacidad.

## **Objetivos del trabajo de titulación**

### **Objetivo general**

Implementar un prototipo de prótesis de mano robótica que genere 5 movimientos articulados.

### **Objetivos específicos**

- Establecer los materiales inteligentes adecuados para realizar la mano.
- Diseñar el sistema de control automático mediante módulo Arduino de acuerdo a los movimientos definidos para la mano.
- Crear el prototipo articulado de la mano incorporando los sensores que permitan medir las variables de fuerza, temperatura y variación resistiva por la variación de posicionamiento.
- Generar 5 tipos de movimientos con la mano: mano abierta – cerrada, movimiento individual de cada dedo, agarre cilíndrico, agarre de gancho y agarre lateral.
- Crear software en un dispositivo móvil para el control de los movimientos definidos de la mano de forma inalámbrica.
- Realizar pruebas de funcionamiento y validación.

## DESCRIPCIÓN DE LOS CAPÍTULOS

Para el desarrollo del proyecto inicialmente en el Capítulo 1 se tienen los fundamentos teóricos de las herramientas utilizadas, como son el estudio de la definición y funcionamiento de las aleaciones de material inteligente SMA, las características principales del módulo de comunicación inalámbrica *Bluetooth*, módulo Arduino, los sensores de temperatura y fuerza.

El capítulo 2 se encuentra orientado a la propuesta realizada para el diseño del prototipo de mano robótica a nivel de la electrónica utilizada y el software de funcionamiento.

En el capítulo 3 está orientado a la implementación del proyecto, donde se describe el funcionamiento de cada circuito realizado en el software de simulación, la fabricación de los circuitos electrónicos, la implementación sobre la estructura de la prótesis de mano y pruebas de funcionamiento.

Finalmente se presentan las conclusiones y recomendaciones que se dedujeron durante el desarrollo del proyecto los cuales permitirán establecer nuevas líneas de investigación o trabajos futuros del proyecto propuesto.

# CAPÍTULO I. FUNDAMENTACIÓN TEÓRICA

## 1.1 Fundamentos de aleaciones de memoria de forma SMA (shape-memory alloy)

### 1.1.1 Definición

Una aleación de memoria de forma SMA<sup>1</sup> es un metal conformado por dos materiales en distinta proporción, contiene propiedades de súper elasticidad<sup>2</sup> y de memoria de forma, permitiendo a la aleación tener la capacidad de recordar una forma incluso después de ser sometido a varios esfuerzos físicos provocando la deformación en su estructura, y tener la capacidad de recuperar su forma original al ser expuesta a variaciones de temperatura.<sup>3</sup>

Debido que las aleaciones de memoria de forma tienen una respuesta singular a los cambios físicos y de temperatura se los han clasificado como materiales inteligentes, esencialmente utilizados para generar fuerza, movimiento o almacenar energía.<sup>4</sup>

### 1.1.2 Propiedad de efecto de memoria de forma de los SMA

La propiedad de memoria de forma en un material inteligente es posible debido a las fases de Martensita<sup>5</sup> y Austenita<sup>6</sup> presentados en la aleación al ser sometido a diferentes rangos de temperatura, donde la temperatura puede tener origen en una fuente calorífica externa o ser generado al momento de estar sometido a una corriente continua.

---

<sup>1</sup> SMA: shape-memory alloy

<sup>2</sup> Súper elasticidad: Propiedad física que permite deformar un material en gran medida, hasta un 10 %.

<sup>3</sup> (Wikipedia, 2017)

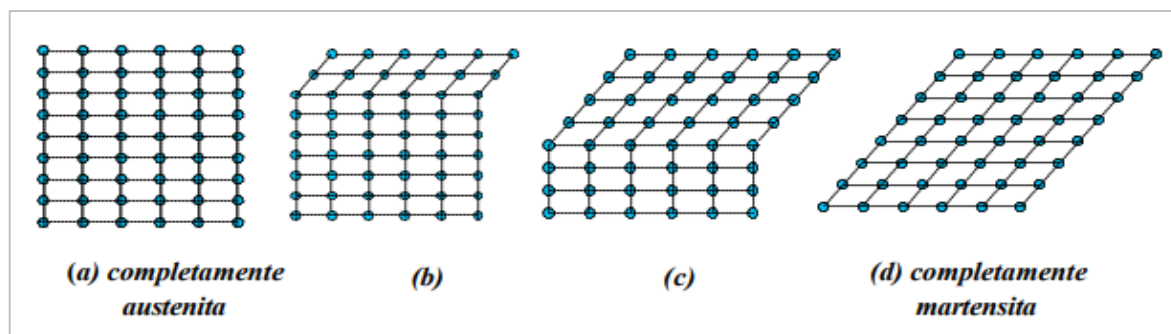
<sup>4</sup> Fuente: (Wikipedia, 2017), (DYNALLOY, 2018)

<sup>5</sup> Martensita: Nombre que recibe la fase cristalina en aleaciones ferrosas.

<sup>6</sup> Austenita: Es una forma de ordenamiento específica de los átomos de hierro y carbono

El estado de Martensita se produce en el momento que el material es sometido a una variación de temperatura baja, donde la estructura cristalina de la aleación se vuelve flexible para permitir que se vuelva moldeable o deformable, mientras en el estado de Austenita al ser sometido a una temperatura alta, la estructura cristalina de las aleaciones vuelve a organizarse acorde a su estructura establecida previamente como un material en estado sólido o rígido.

En la Figura 1.1 se encuentra el proceso de cambio de fase de Austenita a Martensita, donde la estructura cristalina cambia conforme a la variación de temperatura, permitiendo a la aleación tener distintos comportamientos. En la imagen (a) se encuentra la estructura cristalina ordenada en la fase de Austenita, en las imágenes (b) y (c) se observa la transición o cambio de la estructura cristalina ordenada a la desordenada, mientras que en la imagen (c) se visualiza la estructura cristalina de forma desorganizada en la fase de Martensita.



**Figura 1.1 Fases de cambio de los SMA**  
Fuente: (Salinas Chávez, 2011)

La compresión de las propiedades dentro de la fase de Martensita se puede realizar desde las perspectivas de transformación microscópica y macroscópica.<sup>7</sup>

---

<sup>7</sup> Fuente: (Salinas Chávez, 2011)



Desde un enfoque microscópico los SMA se comportan como transformaciones por desplazamiento, donde existe un movimiento de átomos en un rango no muy amplio para formar nuevas estructuras cristalinas sin la necesidad de cambiar la estructura química de la matriz de la aleación, lo que permite una rápida transformación entre fases sin depender del tiempo en que se encuentra sometido a una temperatura específica sino del rango de temperatura en que se encuentra.

Mientras que en un enfoque macroscópico en la Austenita y Martensita se producen diferencias en sus propiedades conforme se atraviesa el punto de transformación entre fases mediante la variación de las temperaturas.

En la figura 1.2 la transición entre la fase de Austenita y Martensita forma un ciclo o lazo de histéresis<sup>8</sup> el cual se encuentra caracterizado por 4 temperaturas denominados temperaturas características o de transformación:

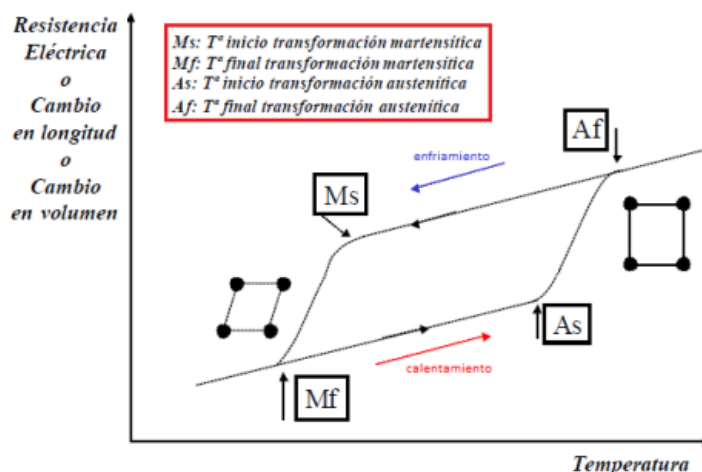
- Inicio de la fase Martensita (Ms)
- Final en la fase Martensita (Mf)
- Inicio de fase Austenita (As)
- Final de fase Austenita (Af)

En el lazo de histéresis formado por las variaciones de temperatura en el SMA, no existe una linealidad entre las variaciones de temperatura y los cambios de resistencia o cambio de longitud, dificultando el proceso de control de la aleación en función de variaciones de temperatura<sup>9</sup>.

---

<sup>8</sup> Histéresis: La histéresis es la tendencia de un material a conservar una de sus propiedades, en ausencia del estímulo que la ha generado.

<sup>9</sup> Fuente (Cano Sánchez , 2010)



**Figura 1.2 Ciclo de histéresis de los SMA**

Fuente: (Gómez Lopez, 2011)

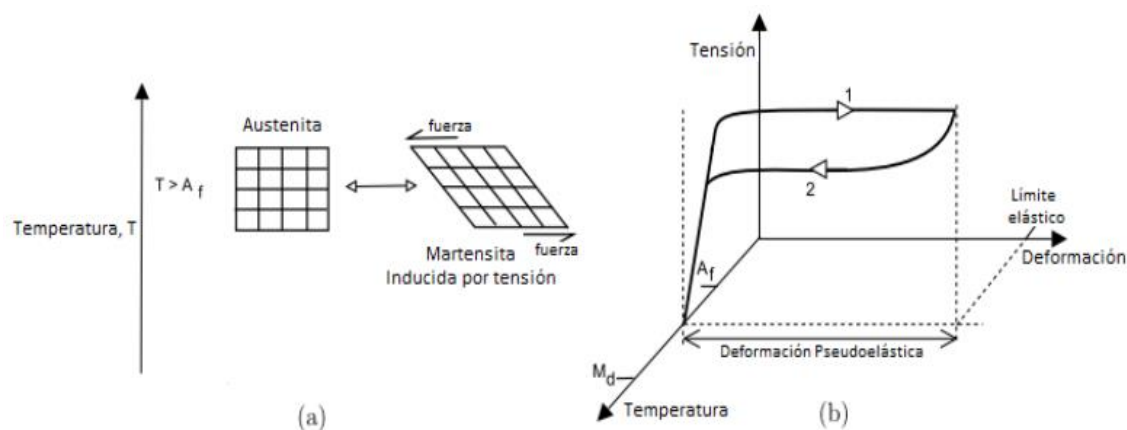
### 1.1.3 Propiedad de súper elasticidad en los SMA

Al referirnos a súper elasticidad en un metal, se puede considerar como la característica que permite generar grandes deformaciones a través de una carga y tener la capacidad de recuperarse después de eliminar la carga.

Con lo indicado anteriormente la súper elasticidad en una aleación SMA es la capacidad para inducir una transformación de fase Martensita mediante una tensión, produciendo que la aleación retorne a su estado inicial, transformando la Martensita producida en Austenita.

La súper elasticidad en un SMA ocurre en un rango de temperatura que inicia en la temperatura final de fase Austenita  $A_f$  hasta la temperatura máxima para transformación de Martensita por tensión  $M_d$ , en este rango de temperaturas, la fase de Martensita en la aleación se estabiliza al aplicar una tensión, pero se comporta de forma inestable al dejar de estar sometido a la tensión antes expuesta. Como se observa en la figura 1.3 si la aleación se encuentra en una temperatura mayor a  $M_d$  mientras se aplica una carga, se dejará de producir el efecto de Martensita inducida por tensión y provocará que la

aleación sufra una deformación plástica es decir, se producirá una deformación permanente impidiendo que regrese a su estado anterior<sup>10</sup>.



**Figura 1.3 Propiedad de Súper elasticidad en los SMA**  
Fuente: (Moreno Martínez, 2014)

#### 1.1.4 Tipos de memoria de forma

Las aleaciones de memoria de forma pueden dividirse en dos clases:

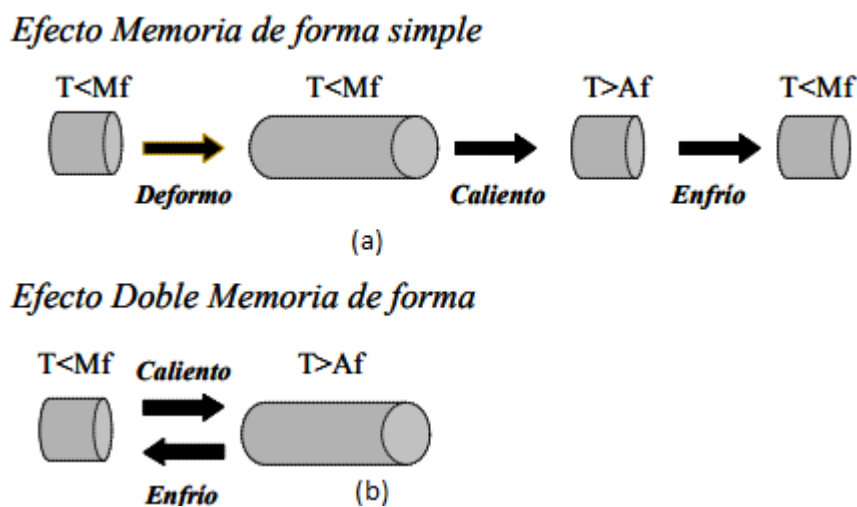
- Memoria Simple
- Memoria Doble

En las aleaciones de memoria simple, la forma de estructura definida (forma aprendida previamente) se encuentra en la fase de Austenita y se llegará a este, al ser sometido a una temperatura elevada. Para poder llegar a otra forma de estructura se deberá realizar por medio de un esfuerzo físico o tensión aplicada a la aleación en su fase de Martensita, este esfuerzo físico debe estar dentro de los rangos de estrés que se puede generar en la aleación sin provocar daños en el mismo.

<sup>10</sup> Fuente: (Moreno Martínez, 2014)

En las aleaciones de memoria doble se dispone de dos formas de estructura, una en la fase Martensita y otra en la Austenita. Al ser sometida a una temperatura alta la aleación toma una estructura establecida previamente mientras que al estar sometida a una temperatura menor o se enfría el material inteligente, tomará una estructura de forma totalmente diferente a la presentada en la fase de Austenita.<sup>11</sup>

En la Figura 1.4 se define la comparación entre los dos efectos de memoria mencionados anteriormente. En la imagen (a) el material se encuentra en su estado de Martensita y posterior a ser sometida a un esfuerzo físico regresa a su forma establecida previamente, mientras que en la imagen (b) se refleja el comportamiento de transformación a nivel de su estructura acorde a la temperatura en la que se encuentra sometida; contrario a lo observado en la memoria simple, en la memoria doble no se requiere de un esfuerzo físico para llevar al material de una forma a otra.



**Figura 1.4 Tipos de memoria de forma**  
Fuente: (Salinas Chávez, 2011)

<sup>11</sup> Fuente: (Salinas Chávez, 2011)

### 1.1.5 Entrenamiento de los SMA

Tanto las aleaciones de memoria simple y doble tienen que ser sometidas a un ciclo de esfuerzo en cada variación de temperatura, con el propósito de que la aleación recuerde una nueva forma. A este tipo de ciclo se lo conoce como entrenamiento.

A continuación se mencionan algunas de las técnicas utilizadas para el entrenamiento de los materiales inteligentes:

- Deformación de la aleación en la fase de Austenita con el fin de obtener deformaciones permanentes en la estructura como son las dislocaciones.
- Aplicación de tensiones externas con la meta de que el material inteligente obtenga una fase de Martensita inducida por tensión.
- Mediante la utilización de ciclo de memoria de forma que consiste en recuperar en varias ocasiones la forma definida del SMA después de sufrir deformaciones.
- Mediante el entrenamiento de tensión continua durante la fase de Austenita con el fin de obtener una cantidad de Martensita inducida por tensión.

Con las técnicas mencionadas, el proceso para que una aleación adquiriera una nueva forma, es colocar el material en la nueva forma deseada el mismo que estará sometido a una tensión fija en la fase de Martensita y posteriormente someterle a una temperatura dentro del rango de Austenita; Una vez alcanzada una alta temperatura se debe someter al material inteligente inmediatamente a una temperatura baja.

Si en el proceso de aprendizaje el material inteligente excede al rango de temperatura de la fase Austenita, producirá que la aleación pierda sus propiedades y de acuerdo al diámetro de la aleación puede producirse puntos de ruptura, por lo que siempre debe ser sometida a una temperatura uniforme.<sup>12</sup>

---

<sup>12</sup> Fuente: (Moreno Martínez, 2014)

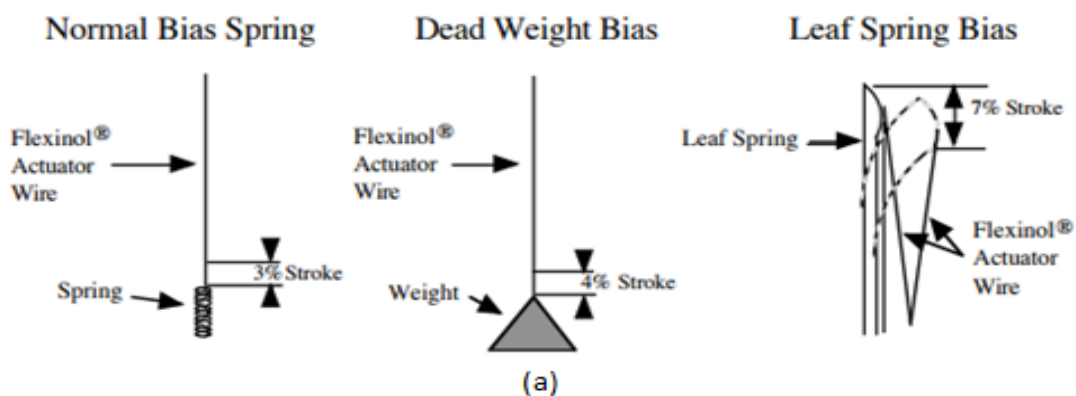
### 1.1.6 Movimientos de los SMA

Antes de hacer mención a los movimientos recomendados en los materiales inteligentes se iniciará con la aclaración del término estrés.

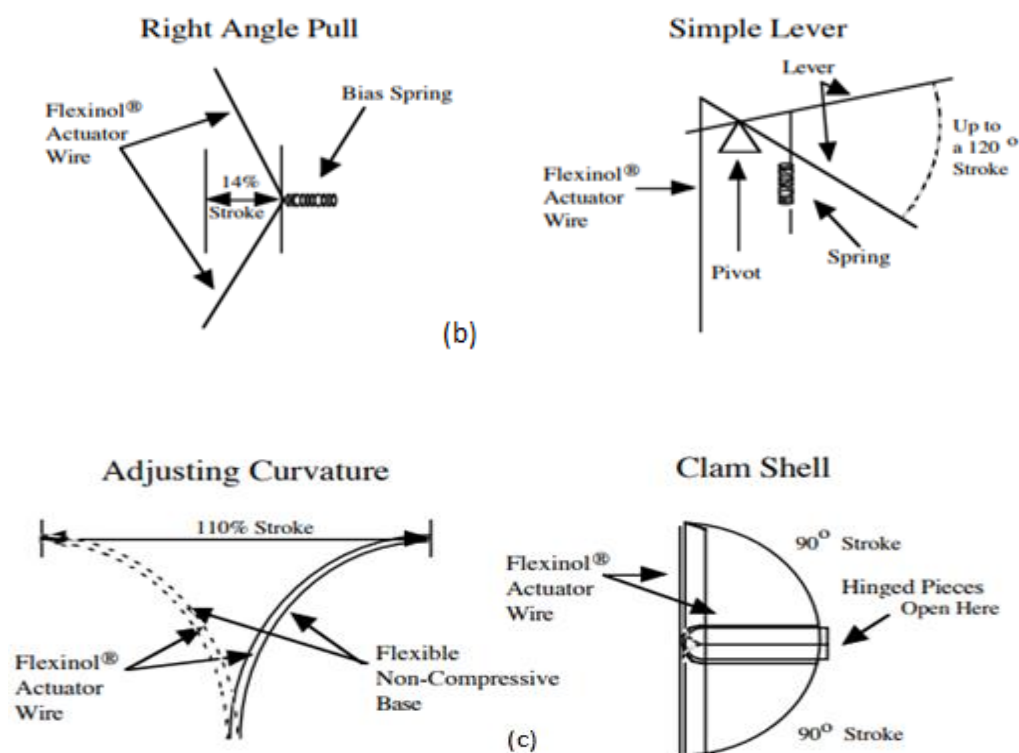
El estrés presentado en los materiales inteligentes se puede considerar como el nivel máximo de deformación que puede sufrir sin comprometer la propiedad de memoria de forma, es decir que el exceso de ciclos de deformación donde el estrés es mucho mayor al establecido producirá que la vida útil del SMA disminuya.

Con lo indicado anteriormente la variación del movimiento de los SMA se mide como un porcentaje de la longitud del alambre utilizado y la cantidad de estrés que se utiliza para restablecer el alambre a su forma establecida, o para estirarlo en su fase de Martensita.

En la figura 1.5 se presentan los tipos de movimiento y el estrés máximo que se debe aplicar en cada uno de los movimientos de acuerdo al fabricante Dynalloy<sup>13</sup>.



<sup>13</sup> DYNALLOY, Inc. es un fabricante de una clase específica de aleaciones de memoria de forma basadas en NiTi,



**Figura 1.5 Tipos de movimientos de los SMA**  
Fuente: (DYNALLOY, 2018)

### 1.1.7 NITINOL

Nitinol es una aleación de memoria de forma conformado de un 55%-56% níquel y 44%-45% titanio, eléctricamente se comporta como una resistencia eléctrica que varía según la longitud y diámetro. Sus aplicaciones van desde la ortodoncia, medicina, actuadores y robótica.<sup>14</sup>

El Nitinol fue desarrollado en 1961 por el laboratorio Naval Ordnance. El nombre del Nitinol proviene de la composición del Níquel, Titanio y del laboratorio en el que fue desarrollado (Nitinol)

<sup>14</sup> Fuente: (Mucslwires®, 2017)

A continuación se nombran algunas de las ventajas y desventajas que dispone el Nitinol en relación con otros materiales inteligentes compuestos por dos elementos.

#### Ventajas

- Dispone de mayor capacidad de memoria en proporción de un 8% en relación a otras aleaciones.
- La resistividad eléctrica del Nitinol es mayor en comparación con otras aleaciones, lo que permite mayor facilidad en la activación del material al ser inducida una corriente eléctrica.
- Dispone de mejor resistencia a la corrosión producido por elementos ambientales.
- Las aleaciones de Nitinol son más estables térmicamente, lo que permite tener un mayor margen de intervalos de temperatura para el cambio de fases.

#### Desventajas

- Su valor económico es mayor a otras aleaciones y más difíciles de mecanizar.
- Los ciclos de histéresis producida en las aleaciones de Nitinol son más pronunciado provocando que a pequeños cambios de temperatura se produzcan grandes deformaciones impidiendo un control proporcional a variaciones de temperatura.

En la tabla 1.1 se muestra las relaciones del tipo de cables Nitinol acorde a su diámetro, longitud y tipo (LT *low* temperatura, HT *high* temperatura).

Diámetro (um)	Resistencia ohms/inch (ohms/m)	Fuerza de recuperación* (gr)	Fuerza de deformación* (gr)	Corriente típica en 1 segundo de contracción (mA)	Tiempo de enfriamiento 158° F, 70°C "cable LT" (Seg)	Tiempo de enfriamiento 194° F, 90°C "cable HT" (Seg)
0.001 (030µm)	36.2 (1425)	0.02 (8.9)	0.008 (3.6)	45	0.18	0.15



<b>0.0015 (037µm)</b>	22.6 (890)	0.04 (20)	0.016 (8)	55	0.24	0.20
<b>0.002 (050µm)</b>	12.7 (500)	0.08 (36)	0.032 (14)	85	0.4	0.3
<b>0.003 (075µm)</b>	5.9 (232)	0.18 (80)	0.07 (32)	150	0.8	0.7
<b>0.004 (100µm)</b>	3.2 (126)	0.31 (143)	0.12 (57)	200	1.1	0.9
<b>0.005 (125µm)</b>	1.9 (75)	0.49 (223)	0.20 (89)	320	1.6	1.4
<b>0.006 (150µm)</b>	1.4 (55)	0.71 (321)	0.28 (128)	410	2.0	1.7
<b>0.008 (200µm)</b>	0.74 (29)	1.26 (570)	0.50 (228)	660	3.2	2.7
<b>0.010 (250µm)</b>	0.47 (18.5)	1.96 (891)	0.78 (356)	1050	5.4	4.5
<b>0.012 (300µm)</b>	0.31 (12.2)	2.83 (1280)	1.13 (512)	1500	8.1	6.8
<b>0.015 (375µm)</b>	0.21 (8.3)	4.42 (2250)	1.77 (900)	2250	10.5	8.8
<b>0.020 (500µm)*</b>	0.11 (4.3)	7.85 (3560)	3.14 (1424)	4000	16.8	14.0

**Tabla 1.1 Características del Nitinol**  
**Fuente: (DYNALLOY, INC, 2018)**

Como se observa en la tabla 1.1 existe una relación entre la fuerza que se produce en el Nitinol y el diámetro de su estructura; la fuerza que se produce y la corriente requerida; Concluyendo que a mayor sección de área del Nitinol se producirá una mayor fuerza sin

depender de su longitud. Pero a mayor área se va a requerir de una mayor corriente que permita al Nitinol alcance su temperatura de cambio de estado.<sup>15</sup>

### **1.1.8 Control de memoria de forma de Nitinol.**

Desde el descubrimiento del Nitinol se han propuesto varios tipos de sistemas para el control de posicionamiento desde sistemas complejos de PDI<sup>16</sup>, sistemas de control de posición en función a la temperatura del material, sistemas de control de posición en función a la variación resistiva del material inteligente, y sistemas electrónicos de control mediante señales PWM<sup>17</sup>, siendo este último el más aceptado debido a la facilidad de implementación de circuitos electrónicos para generar las señales PWM, como también mayor facilidad en el manejo de su posición al aplicar pulsos con un ancho de pulso predefinido.

En la figura 1.6 se puede identificar el circuito propuesto por el fabricante Dynalloy donde el control del alambre de Nitinol, se lo realiza mediante pulsos PWM generados por el circuito oscilador.

La alimentación del circuito es de 5 voltios y el diámetro que se puede utilizar es de 100  $\mu\text{m}$ , motivo por el cual se utiliza una resistencia en serie para limitar la corriente que circula por el Nitinol; El valor de la resistencia dependerá del diámetro del material inteligente y el voltaje de alimentación del circuito.

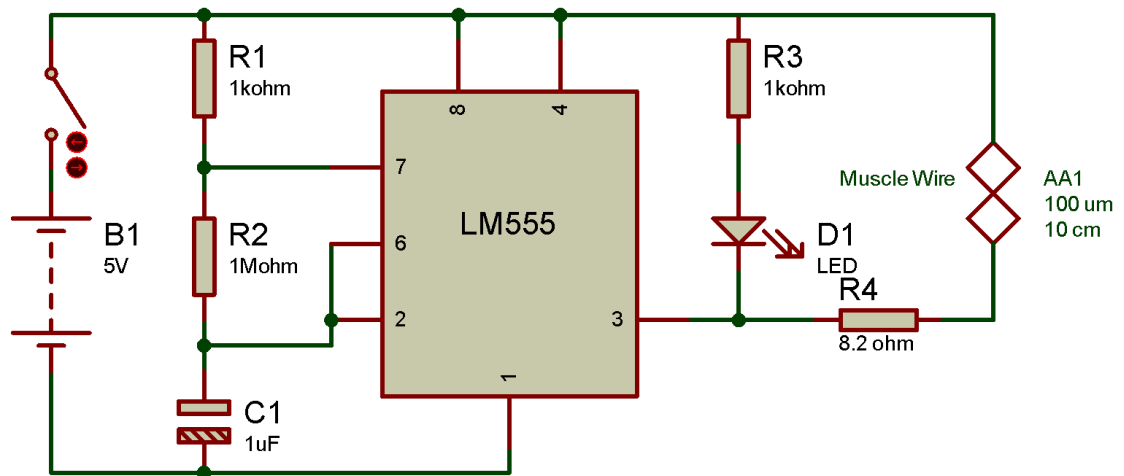
---

<sup>15</sup> Fuente: (DYNALLOY, INC, 2018)

<sup>16</sup> PDI: Un controlador PID es un mecanismo de control por realimentación ampliamente usado en sistemas de control industrial

<sup>17</sup> PWM: Pulse-Width Modulation (Modulación por Ancho de Pulso).

### A Simple Oscillator Circuit



**Figura 1.6 Circuito generador de PWM con integrado 555**  
Fuente: (DYNALLOY, Inc, 2017)

#### 1.1.9 Resortes de aleaciones de memoria Nitinol

Los resortes de Nitinol son una forma de presentación física de los alambres de Nitinol, donde la aleación adopta la forma de resorte pero es sometido a un proceso que impide que la nueva forma adoptada regrese a su forma de alambre, el resorte de Nitinol constará con las mismas propiedades de los SMA y de igual manera adoptará las propiedades de los resortes, permitiendo que el material produzca movimientos lineales en dos sentidos.

En la figura 1.7 se puede apreciar la forma física de los resortes de Nitinol, mientras que en la tabla 1.2 se puede observar las características de los resortes que es semejante a la tabla 1.1, pero se diferencia por la característica de relación de estiramiento.



**Figura 1.7 Resorte de aleación de memoria de forma Nitinol**  
Fuente. Elaborado por el autor

Díámetro de alambre de resorte in(mm), diámetro externo in(mm)	SR en Caliente , SR en Frío	Resistencia ohms/inch (ohms/meter )	Fuerza de recuperación n* (gramos)	Fuerza de deformación * (gramos)	Corriente típica en 2 segundo de contracción (mA)	Tiempo de enfriamiento o 194° F, 90°C "cable HT" (segundos)
<b>0.020 (0.51)</b> <b>0.136 (345)</b>	16, 7	0.11 (4.33)	0.536 (243.3)	0.215 (97.32)	3.4	15.0
<b>0.015 (0.381)</b> <b>0.10 (2.54)</b>	15, 6	0.21 (8.27)	0.307 (139.3)	0.122 (55.72)	1.9	9.0
<b>0.008 (0.203)</b> <b>0.054 (1.37)</b>	14, 5	0.74 (29.13)	0.089 (39.3)	0.035 (15.94)	0.7	3.0

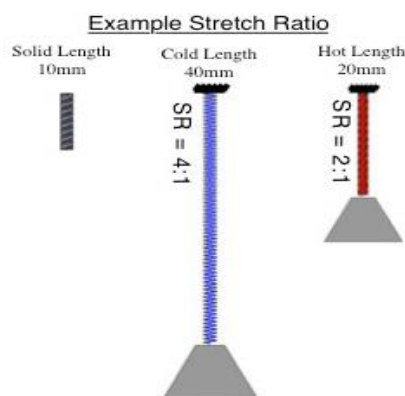
**Tabla 1.2 Características de los resortes de memoria de forma Nitinol**  
Fuente: (DYNALLOY, INC., 2018)

La relación de estiramiento “SR” indicada en la tabla 1.2 de características se define como relación directa de cualquier longitud “L” e inversamente proporcional a la longitud del resorte en su forma sólida original “SL”, esta característica se puede presentar tanto en el estado de Martensita o Austenita.

$$SR = \frac{L}{SL} \quad (1)$$

En la figura 1.8 se tiene un ejemplo de la relación de estiramiento donde se tiene una longitud sólida de 10 mm y un SR de 4 en frío y 2 en caliente, conforme la ecuación (1) se puede deducir la longitud de estiramiento en 40 mm en frío y 20 mm en caliente.<sup>18</sup>

<sup>18</sup> Fuente: (DYNALLOY, INC., 2018)



**Figura 1.8 Ejemplo de relación de estiramiento**  
**Fuente: (DYNALLOY, INC., 2018)**

### 1.1.10 Aleación de memoria de forma escogida

Con la información expuesta en este documento, donde se consideró la información propuesta por el fabricante, las características innatas de las aleaciones de memoria de forma y la disponibilidad comercial, se ha escogido a los resortes de Nitinol como actuadores para generar el movimiento individual de cada dedo de la mano.

Las consideraciones realizadas se mencionan a continuación:

- Fuerza de recuperación, es un elemento a considerar debido a la fuerza opositora que se presenta en cada articulación para generar el movimiento, en este caso esta fuerza estará conformada por el peso de la articulación y la fuerza de fricción que se opone al movimiento ubicado en las uniones de las articulaciones.

$$Fd = masa * gravedad \quad (2)$$

$$Fd = Wd \quad (2.a)$$

Donde  $Fd$  es considerado la fuerza producida por el peso de cada articulación  
 $Wd$

- Las fuerzas de fricción o rozamiento que impiden el movimiento de la articulación y se encuentra definido por la siguiente ecuación:

$$F_R = \mu * N \quad (3)$$

Donde:

- Fr es la fuerza de fricción producida entre las articulaciones de la mano robótica que se oponen al movimiento generado por los resortes de Nitinol.
- $\mu$  es la constante de fricción del material (0,60 plástico)
- N se considera como el vector de fuerza Normal.

El resorte de Nitinol usado es de diámetro de  $750\mu\text{m}$  (0,75 milímetros). Sus características se describen a continuación.

- Diámetro de cable:  $750\mu\text{m}$
- Fuerza de deformación de 6 Newtons
- Variación de longitud de 14 cm aplicando una fuerza de 6 Newtons
- Fuerza de recuperación de 8.4 Newtons
- Rango de temperatura de trabajo en fase Austenita:  $75^\circ\text{C} - 85^\circ\text{C}$ .
- Diámetro exterior de la bobina: 6mm
- Contracción en 1.8 segundos con una corriente de 2A.

Con las consideraciones expuestas se aplica la siguiente sumatoria de fuerzas

$$F_{niti} = 8.4 \text{ Newtons}$$

$$F_{niti} > F_d + F_r \quad (4)$$

$$F_d = m * g \quad (5) \quad F_r = \mu * m * g \quad (6)$$

$$F_{total} = (1 + \mu) * m * g \quad (7)$$

$$F_{total} = (1 + 0.60) * (m) * 9.8 \frac{m}{s^2} \quad (7.a)$$

La masa promedio de cada dedo es de 88 gramos (0,088Kg), con el cual se reemplaza en la ecuación (7.a) y se obtiene la fuerza total que se opone al movimiento del resorte inteligente es de 1.37 newtons.

$$F_{total} = (1 + 0.60) * (0.088 \text{ Kg}) * 9.8 \text{ m/s}^2$$

$$F_{total} = 1.37 \text{ N}$$

$$8.4 \text{ N} > 1.37 \text{ N}$$

Acorde a los resultados obtenidos el resorte de Nitinol de 750  $\mu\text{m}$  dispone la fuerza necesaria para generar los movimientos de cada dedo de la mano.

Con las características del resorte de Nitinol se puede obtener el coeficiente de elasticidad, el mismo que depende de la fuerza ejercida y el desplazamiento producido.

$$F = K * \Delta X \quad (8)$$

$$k = \frac{F}{\Delta X} \quad (9)$$

Donde:

- $F = 6$  newtons
- $\Delta x = 14$  cm (0,14 m)

Reemplazando los datos proporcionados por el fabricante en la ecuación (9) se obtiene la constante de elasticidad K.

$$K = \frac{6 \text{ N}}{0,14 \text{ m}}$$

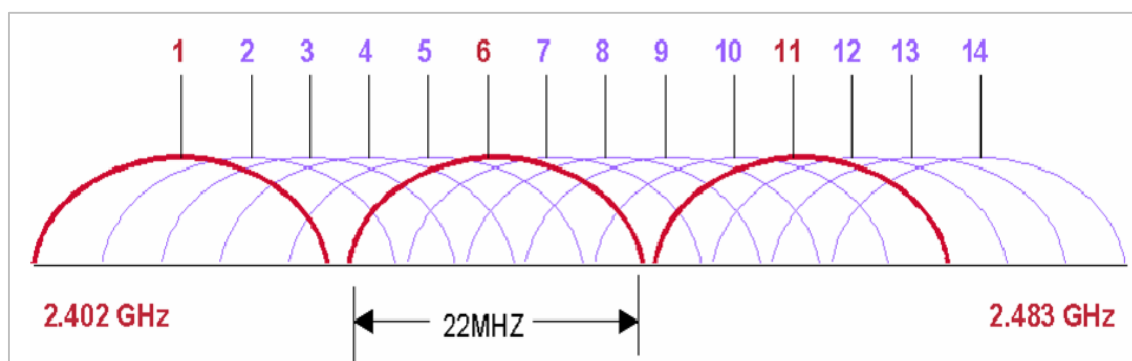
$$K = 42,85 \text{ N/m}$$

El coeficiente de elasticidad para el resorte de 750  $\mu\text{M}$  es de 42,85  $\text{N}/\text{m}$

## 1.2 Bandas ISM

Las bandas ISM<sup>19</sup> son consideradas en radiofrecuencia como bandas de uso libre en las telecomunicaciones, orientadas principalmente para las aplicaciones médicas, científicas, industriales, y tecnológicas, siendo este último el campo más usado para las conexiones inalámbricas en las frecuencias de 2.4 GHz y 5 GHz, utilizados por dispositivos móviles inteligentes como teléfonos, computadores y dispositivos inalámbricos que funcionan bajo los estándares IEEE<sup>20</sup> 802.15.1 (WPAN<sup>21</sup> y dispositivos *Bluetooth*), IEEE 802.11 (WLAN<sup>22</sup>), IEEE 802.15.4.

Dependiendo del lugar geográfico, la banda de 2,4 GHz utiliza de 7 a 14 canales, donde se tiene una frecuencia central, una inferior y una superior con una separación entre canales de 5 MHz, para evitar la interferencia entre canales adyacentes se realiza una separación de 5 canales por lo que tradicionalmente se utilizan los 1, 6 y 11 o los 1, 5, 9 y 13, según el estándar como se muestra en la Figura 1.9<sup>23</sup>



**Figura 1.9 Distribución de canales en la banda ISM**  
Fuente: (Simons, 2012)

19 ISM: Industrial, Scientific and Medical

20 IEEE: Instituto de Ingeniería Eléctrica y Electrónica/ estandarización mediante protocolos

21 WPAN: Red inalámbrica de área personal

22 WLAN: Red de área local inalámbrica

23 Fuente: (Simons, 2012)



## 1.3 Bluetooth

*Bluetooth* es considerado como una tecnología de comunicación inalámbrica de corto alcance que aprovecha el espectro de radio frecuencia para establecer una conexión entre dispositivos sin la necesidad de usar un medio alámbrico y permitir el intercambio de datos provenientes de dispositivos como impresoras, teléfonos móviles, auriculares inalámbricos, entre otros.

La tecnología *Bluetooth* utiliza el rango de frecuencia la banda libre ISM 2.4 GHz a 2,485 GHz, permite establecer una velocidad máxima de datos aproximadamente hasta 20000 Kbps en un rango de alcance de diez metros para un transmisor de segunda clase y aproximadamente a 100 metros para un transmisor de primera clase.<sup>24</sup>

### 1.3.1 Normas Bluetooth

El estándar de comunicación de los dispositivos electrónicos *Bluetooth* se puede clasificar en las siguientes normas según la IEEE:

- Estándar 802.15.1 (Bluetooth versión 3), permite a los dispositivos lograr velocidades de transferencia de hasta 1000 Kbps.
- Estándar 802.15.3 (Bluetooth versión 4), es una norma que se encuentra en desarrollo, permitirá a los dispositivos *Bluetooth* obtener velocidades de 20000 Kbps lo cual se puede considerar como banda ancha.
- Estándar 802.15.4 (Bluetooth versión 4), al igual que la norma anterior aún se encuentra en desarrollo para el manejo de aplicaciones de baja velocidad<sup>25</sup>.

---

24 Fuente: (Carlos, 2016)

25 Fuente: (Simons, 2012)

## 1.4 Módulos Bluetooth HC-05

El módulo electrónico *Bluetooth* HC-05 permite conectar varios dispositivos de forma inalámbrica. Incorpora la configuración de fábrica que permite trabajar en modo maestro o modo esclavo dependiendo del tipo de aplicación a realizarse. En el modo maestro permite que el módulo se pueda comunicar con otros módulos *Bluetooth*, es decir permitirá al micro-controlador seleccionar a otros dispositivos *Bluetooth* que se encuentran a la espera de peticiones, mientras que en el modo esclavo únicamente escucha por peticiones de conexión.

La configuración del módulo se realiza mediante un lenguaje llamado AT, el cual permite mediante una comunicación serial entre una laptop y el módulo *Bluetooth* cambiar las configuraciones del dispositivo como son: nombres del dispositivo *Bluetooth*, contraseña, potencia de TX, modo de trabajo maestro o esclavo, etc.<sup>26</sup>

### 1.4.1 Características de módulo bluetooth HC-05

En la figura 1.10 se puede visualizar los pines de conexión del módulo *Bluetooth* HC-05. Los pines de conexión *State* y *Enable* permite el ingreso al modo de configuración AT, mientras que los pines RXD y TXD establecen la comunicación serial con otros dispositivos micro-controladores como es el módulo Arduino, considerar que la conexión de los pines RXD y TXD se tiene que invertir en el dispositivo al que se conecte.



**Figura 1.10** Módulo Bluetooth HC-05  
Fuente: (Naylamp Mechatronics SAC, 2018)

---

<sup>26</sup> Fuente: (ELECTRONICA60NORTE, 2017)

Aunque el módulo HC-05 permite una alimentación de 5 voltios, es recomendable que la alimentación sea de 3.3 voltios para evitar daños en el módulo.

En la tabla 1.3 se puede visualizar las características del módulo *Bluetooth*.

CARACTERÍSTICAS	
Voltaje de polarización:	3.3V – 5V.
Voltaje recomendado de operación:	3.3V.
Tasa de transmisión <i>Baud rate</i> [baudios]:	1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200.
Dimensiones:	(4.4 x 1.6 x 0.7) cm
Consumo de corriente:	< 40 mA
Corriente en modo sueño:	< 1mA

**Tabla 1.3 Características módulo *Bluetooth* HC-05**

**Fuente: (Electronilab, 2017)**

## 1.5 Módulo Arduino Nano

Arduino es considerada como una multiplataforma de software y hardware que integra dispositivos electrónicos de lenguaje abierto (*open-source*) permitiendo flexibilidad y fácil manejo para el desarrollo de proyectos electrónicos.

Cuando se habla de multiplataforma se quiere decir que el software de Arduino se puede ejecutar en ordenadores con sistemas operativos como Windows, Macintosh OSX y GNU/Linux. Su ambiente de programación es bastante intuitivo permitiendo la eficiencia y flexibilidad para realizar el desarrollo de aplicaciones debido que está basado en lenguaje de programación C++ que puede ser expandible mediante librerías.

El Arduino Nano es una placa electrónica con un micro-controlador ATmega328, dispone de 14 pines entradas/salidas digitales (6 salidas PWM), 8 puertos analógicas, 1 puerto de comunicación serial UART, un cristal oscilador de cuarzo de 16 MHz, para la conexión a un

ordenador o alimentación externa se dispone de un puerto mini USB tipo B.<sup>27</sup> En la tabla 1.4 se visualiza las características del módulo Arduino Nano V3, y en la figura 1.11 se visualiza el esquema de conexión del módulo.

CARACTERÍSTICAS	
Micro-controlador:	ATmega328
Voltaje de alimentación <i>input</i> :	5V
Pines digitales:	14 (6 con PWM)
Puertos analógicos:	16
Corriente suministrada por pin:	40 mA
Corriente máxima suministrada para el pin 3.3V:	50 mA
Memoria flash:	32 KB
Memoria SRAM:	8 KB
Memoria EEPROM:	2 KB
Velocidad de reloj:	16 MHz

Tabla 1.4 Características de módulo Arduino Nano V3  
Fuente: (Arduino, 2017)

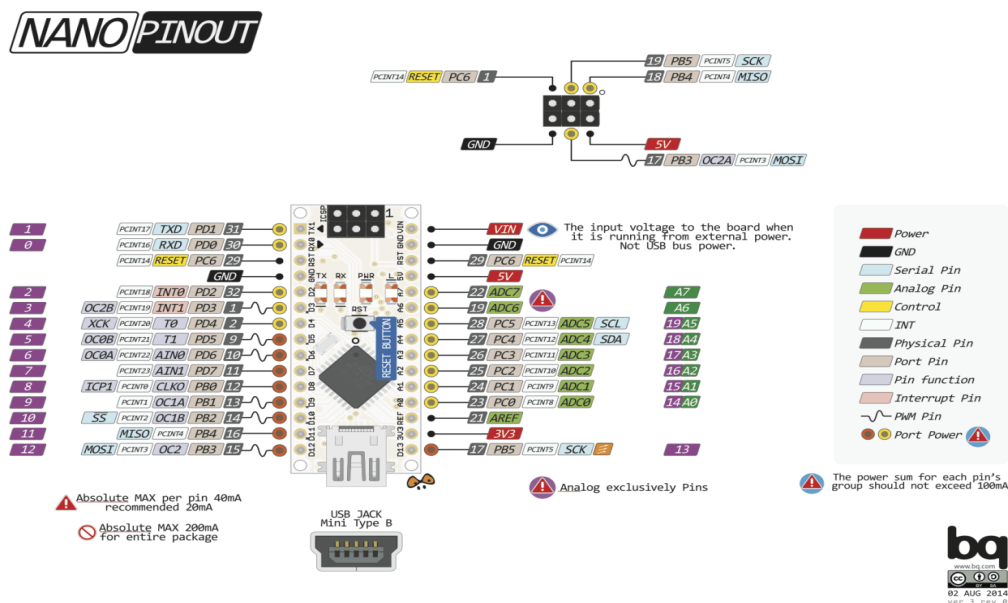


Figura 1.11 Diagrama de conexión de Arduino Nano  
Fuente: (Electronilab, 2018)

<sup>27</sup> Fuente: (Arduino, 2017)

## 1.6 Sensor de fuerza FSR 402

El sensor FSR es considerado como un dispositivo netamente resistivo que consta de dos filamentos con una resistencia que varía en proporción directa a la fuerza aplicada. Cuando el sensor conforma un circuito por división de voltaje al aplicar una fuerza disminuirá la resistencia del sensor y por ende producirá una variación en el voltaje de salida.

La variación de voltaje en el FSR se puede expresar en la ecuación (10):

$$V_{out} = \frac{R_M \cdot V_{cc}}{(R_M + R_{FSR})} \quad (10)$$

Donde:

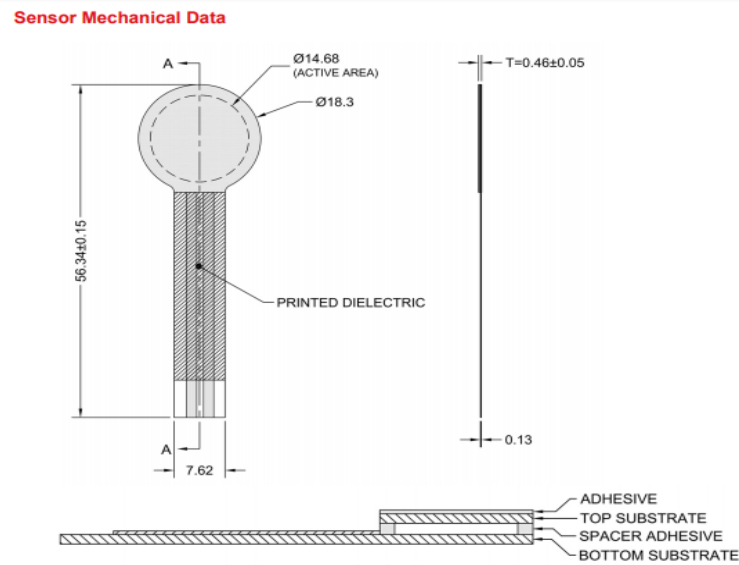
- $V_{out}$  es el voltaje de salida del divisor de voltaje
- $R_M$  es una resistencia fija conocida
- $R_{FSR}$  es la variación de la resistencia del FSR
- $V_{cc}$  el voltaje de alimentación de 5V

Con el proceso indicado anteriormente permitirá acondicionar la señal obtenida del sensor a una forma de voltaje el mismo que será introducido a un micro-controlador para realizar una relación entre el voltaje obtenido y su equivalencia de fuerza medida en newton<sup>28</sup>.

En la figura 1.12 se puede visualizar la estructura del sensor FSR402, donde se visualiza el área activa del sensor

---

<sup>28</sup> Fuente: (Interlink Electronics, 2017)



**Figura 1.12 Esquema de sensor de fuerza FSR402**  
Fuente: (Interlink Electronics, 2017)

## 1.7 Sensor de temperatura LM35

El sensor LM35 es un integrado, que posee internamente un circuito electrónico de control que permite obtener un voltaje de salida que directamente proporcional a la temperatura medida en la entrada del sensor debido a su linealidad, es decir un incremento de voltaje en una razón de 10 milivoltios por grado centígrado

El rango de lectura del sensor LM35 es de  $-55^{\circ}\text{C}$  a  $150^{\circ}\text{C}$ , con una precisión de  $0,5^{\circ}$  centígrados.<sup>29</sup>

En la figura 1.13 se visualiza el esquema de conexión del integrado del sensor LM35.

<sup>29</sup> Fuente (LLamas, 2015)



**Figura 1.13 Sensor de temperatura LM35**  
Fuente: (LLamas, 2015)

## 1.8 Modulación por ancho de pulso PWM

La modulación PWM se considera como el procesamiento de una señal de voltaje que permite enviar información codificada en niveles de voltaje, o mediante circuitos electrónicos permita variar la porción de energía recibida por una carga en un rango de tiempo llamado “Período”.

Las señales PWM son utilizadas en la electrónica digital donde se necesitan simular una señal analógica. Son de tipos cuadradas o sinusoidales donde es posible cambiar el ancho del pulso efectivo referente a su propio período; la resultante de este proceso de cambio es llamado *duty cycle* (*ciclo de trabajo*) y su unidad es representado como un valor porcentual.<sup>30</sup>

## 1.9 Módulo PCA9685 PWM 16 canales

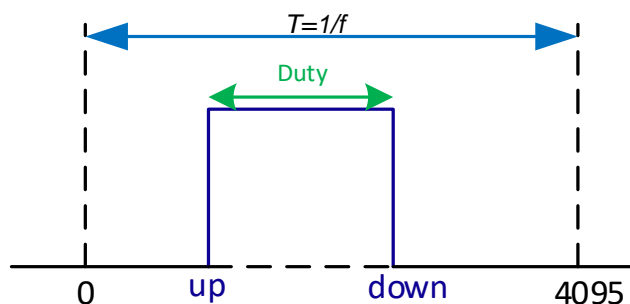
El módulo PCA9685 es una placa electrónica que permiten generar 16 canales de salidas PWM con resolución de 12 bits y únicamente un canal de comunicación mediante I2C a un micro-controlador<sup>31</sup>.

---

<sup>30</sup> Fuente: (Arduino UTFSM RACSO, 2014)

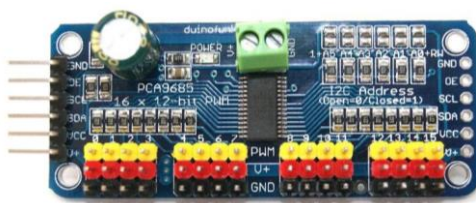
<sup>31</sup> Fuente: (Naylamp Mechatronics, 2016)

En la figura 1.14 se puede identificar el *duty cycle* generado acorde a la resolución de 12 bits (4096 niveles) y el período T. Si la alimentación del módulo PCA9685 es de 5 voltios, el período de 1 segundo y el pulso de trabajo es de una resolución de 2047 niveles, estaríamos hablando que el pulso útil tiene un voltaje RMS de 2.5 voltios, como conclusión se puede variar el pulso *duty cycle* y obtener variaciones del voltaje y corriente en una carga.



**Figura 1.14 Señal PWM de módulo PCA9685**  
Fuente: (Naylamp Mechatronics, 2016)

En la figura 1.15 se detallan las conexiones físicas y en la tabla las características del módulo.



**Figura 1.15 Módulo PWM PCA9685**  
Fuente: (ARMZTEC, 2018)

Características	
<b>Frecuencia</b>	1000 Hz
<b>Número de Canales</b>	16 Canales
<b>Resolución</b>	12 Bits
<b>Voltaje</b>	DC 3 - 5 voltios
<b>V Max +</b>	6 voltios
<b>Tamaño</b>	60*25 mm

**Tabla 1.5 Características de módulo PCA9685**  
Fuente: (Electronilab, 2017)



## CAPÍTULO II. PROPUESTA

### 2.1 Funcionamiento del prototipo de mano robótica con materiales inteligentes

El capítulo II iniciará con una descripción del funcionamiento del prototipo de mano robótica con materiales inteligentes.

Inicialmente se tendrá una estructura que se asemeje a una mano humana que permita realizar movimientos articulados, el cual estará sujeto a un sistema mecánico conformado por resortes del material inteligente de forma llamado Nitinol.

Para el funcionamiento del circuito del proyecto se utilizó una batería de litio de 11 voltios 10 amperios hora que permitió energizar el sistema de control del material inteligente, energizar el módulo Arduino y los sensores de temperatura y fuerza.

En el instante que los circuitos se encuentren energizados, el módulo Arduino inicia sus parámetros de funcionamiento e inicia también la medición de temperatura y fuerza hasta que reciba la orden del dispositivo Smartphone. En el dispositivo Smartphone se realizó de forma manual la vinculación de los dispositivos *Bluetooth* tanto del circuito de la prótesis como del teléfono para poder comenzar la transferencia de información.

Una vez que se recibe una señal inalámbrica del módulo *Bluetooth*, que se encuentra conectado a los puertos seriales del módulo Arduino Nano, este realiza la verificación del tipo de dato recibido y elige el tipo de movimiento seleccionado en el dispositivo Smartphone, inmediatamente se espera la señal que contenga la orden de generar el movimiento de forma manual o automática.

Posterior a la señal recibida, el micro-controlador envía la orden del canal PWM que será utilizado, la frecuencia y pulso efectivo de la señal al módulo PCA9685 mediante una conexión I2C.

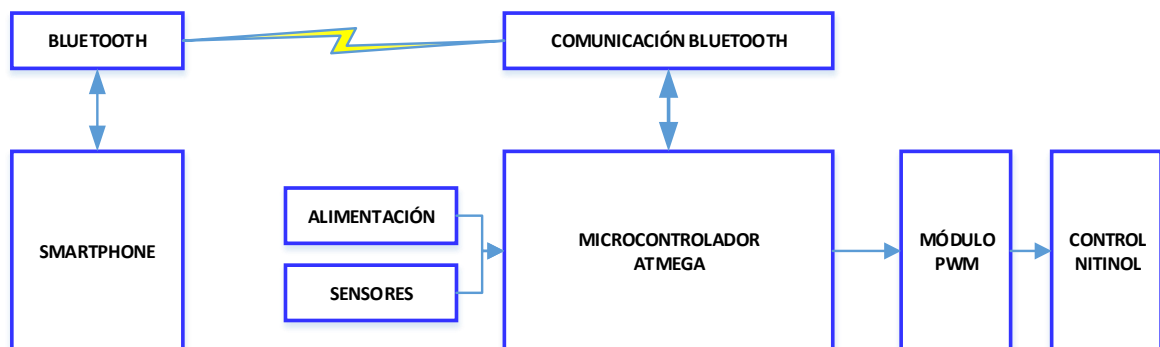
Recibida las órdenes del micro-controlador el módulo PCA9685 produce el PWM que es emitida al circuito de control del material inteligente de Nitinol.

El sistema del control del material inteligente está conformado por circuitos opto-acopladores que permite separar el sistema electrónico de alta y baja potencia, transistores MOSFET que funcionan como un interruptor *on/off*.

Finalmente recibida la señal en el sistema de control, se energizará el material inteligente Nitinol lo que permitirá que el resorte se contraiga y produzca el movimiento de las articulaciones de la prótesis.

## 2.2 Conformación del sistema de control del prototipo de mano robótica con materiales inteligentes

En la figura 2.1 se presenta el esquema de bloque que refleja los elementos usados en el prototipo de mano robótica.



**Figura 2.1 Diagrama de bloque del funcionamiento del prototipo de mano robótica**

Fuente: Elaborado por el autor

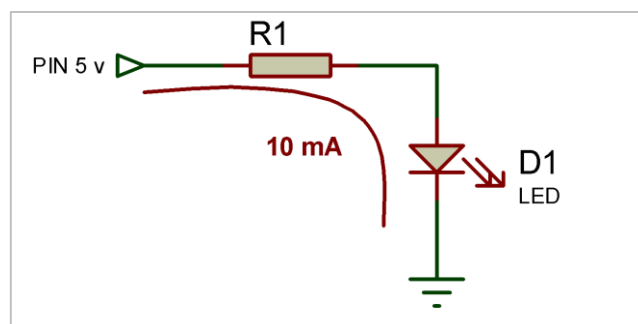
## 2.3 Diseño del circuito de control electrónico

El diseño del circuito de control electrónico partió de evaluaciones realizadas con la estructura de la prótesis de mano, con el fin de complementar un esquema electrónico con la parte mecánica de los movimientos de las articulaciones.

Como fue mencionado en el Capítulo I, las aleaciones de memoria de forma funcionan en una relación de temperatura y movimiento que puede ser controlado por el flujo de corriente que atraviesa la aleación, partiendo de este hecho se implementó un circuito que maneja el flujo de energía por medio de una señal PWM generado por el módulo Arduino o un módulo externo, en este caso se usó el módulo PCA9685 lo que permitió generar señales PWM con una amplitud de 5 V, pulso variable y frecuencia variable.

Para la alimentación del proyecto se utilizó una fuente de 11 voltios con alto amperaje, motivo por el cual fue necesario separar los circuitos digitales de los de potencia. Desde un enfoque tecnológico existen varias formas de separar el circuito de potencia (relés, transistores, opto-acopladores), pero se utilizó los opto-acopladores por su estructura interna donde existe un diodo led y un fototransistor asegurando la independencia de los circuitos electrónicos. Para el diseño del circuito de potencia se escogió el uso de opto-acopladores 4N25 que permite trabajar con voltajes de hasta 30 voltios.

De acuerdo al fabricante los integrados 4N25 trabajan con una corriente de polarización  $I_F$  de 10 miliAmperios (mA) para polarizar el diodo led interno; el módulo Arduino permite suministrar una corriente de 40 mA por pin, por lo que se requiere limitar la corriente a través de una resistencia que se calcula a continuación:



**Figura 2.2 Esquema electrónico de opto-acoplador 4N25**  
Fuente: Elaborado por el autor

$$V_{R1} = 5V - 1.7V \quad R1 = \frac{V_{R1}}{10mA} \quad (11)$$

$$R1 = \frac{3.3V}{10mA} \quad (11.a) \quad R1 = 330\text{ ohm}$$

- $VRI$  es la caída de tensión en la resistencia
- $RI$  es la resistencia calculada

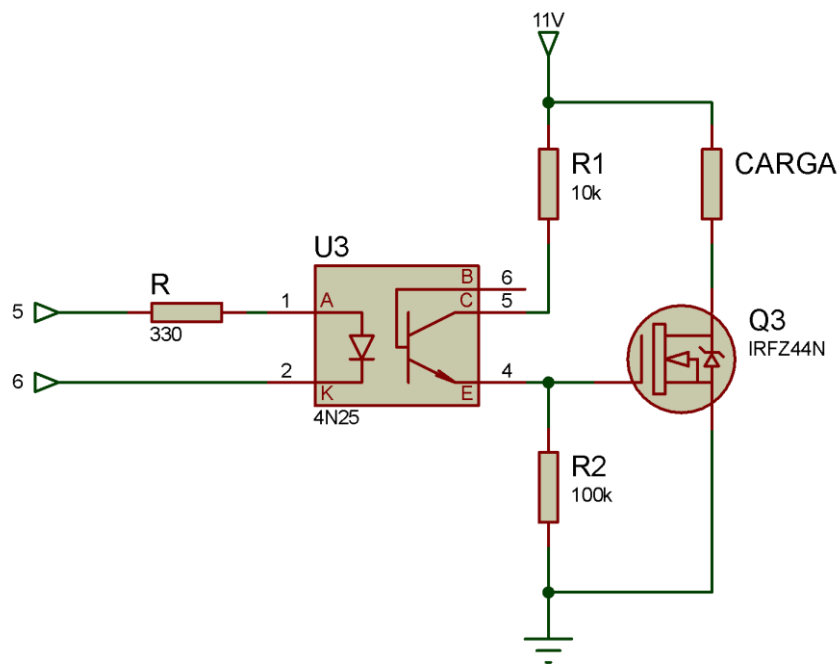
La resistencia calculada que se debe colocar a la entrada digital de cada opto-acoplador es de 330 ohmios.

El circuito de potencia debe funcionar como un *switch* que permita la rápida conmutación de la señal eléctrica acorde a la señal PWM recibida, esto se lo puede realizar mediante un transistor MOSFET. El transistor escogido es el modelo IRFZ44N, debido a sus características indicadas en la hoja de datos del fabricante.

- Rápida conmutación *on/off*
- Resistencia *Drain-Source* baja  $R_{ds}$  de 15 miliohmios

Para el uso del transistor MOSFET se ha utilizado el circuito de polarización que permita trabajar en las zonas de saturación y corte; en el transistor MOSFET IRFZ44N para que pueda funcionar en saturación se requiere que el voltaje de compuerta sea de 10 voltios con el fin de asegurar una corriente de drenador fuente de hasta 15 amperios. Para obtener el voltaje de compuerta de 10 voltios se utilizó un divisor de voltaje que se deriva del uso del opto-acoplador.

En la figura 2.3 se visualiza el esquema del circuito de potencia con los opto-acopladores, al momento que el opto-acoplador es polarizado funciona como un circuito cerrado permitiendo que las resistencias R1 y R2 formen un circuito de divisor de voltaje que activa el transistor MOSFET IRFZ44N.



**Figura 2.3 Circuito de polarización de MOSFET IRFZ44N**  
Fuente: Elaborado por el autor

Los valores de las resistencias R1 y R2 se obtendrá de la siguiente forma:

$$V_{out} = V_{cc} \left( \frac{R2}{R1 + R2} \right) \quad (12)$$

$$R1 = 10 \text{ Kohm valor asumido}$$

Despejando R2 se obtendría la siguiente ecuación:

$$R2 = \frac{R1}{\left( \frac{V_{cc}}{V_{out}} - 1 \right)} \quad (12. a) \quad R2 = \frac{10 \text{ Kohm}}{\left( \frac{11v}{10v} - 1 \right)}$$

$$R2 = 100 \text{ Kohm}$$

Anteriormente se habló que uno de los motivos por lo que se escogió el transistor MOSFET IRFZ44N es por su resistencia drenador-fuente de 17.5 miliohmios, coeficiente que es esencial al momento de calcular la potencia que disipará el transistor al instante que circule altas corrientes. El SMA trabajará en rangos de hasta 3 amperios, es decir, por el

transistor MOSFET circulará una corriente máxima de 3 amperios y la potencia que se disipa en forma de temperatura será de 52.5 mili Watts, acorde a la ecuación (13), con lo que se evita el sobrecalentamiento de los circuitos.

$$P = R_{ds} * I_{Ds}^2 \quad (13)$$

$$P = 17.5 \text{ m}\Omega * 3 \text{ A} \quad P = 52.5 \text{ mW}$$

## 2.4 Dimensionamiento de fuente

Para el presente proyecto se utilizó como fuente una batería de tipo LIPO que permite tener una mayor duración en comparación a otros tipos de fuente portátil.

En la tabla 2.1 se detalla el consumo de los componentes utilizados.

DISPOSITIVO	Consumo de corriente	Voltaje de alimentación	Potencia W
Módulo Arduino Nano	15 mA	5	0,075
Módulo PCA9685	35 mA	5	0,175
Módulo Bluetooth HC05	50mA	3,3	0,165
Sensor Lm35	60uA	5	0,0003
Sensor FSR 402	0,35 mA	5	0,00175
4n25	60 mA	1,15	0,069
Nitinol	2 A	0,08 ohm	(0,32*5)
Regulador de voltaje	160,41mA	11.1	1,78

**Tabla 2.1 Consumo de energía total del prototipo de mano robótica**  
Fuente: Elaborado por el autor

El consumo del circuito es de 3,86 watts más una tolerancia del 10% dando una potencia de 4,24 watts del cual se partió para realizar el cálculo para la capacidad de la batería en el peor de los casos, bajos los siguientes parámetros.

- Autonomía por 20 horas
- Profundidad de la descarga del 30%
- Temperatura media de la batería 25°C

- Tensión de batería 11 voltios

$$P = 4.24 \text{ watts} * 20 \text{ horas} \quad (14)$$

$$P = 84.93 \text{ watts}$$

$$PD30\% = 84,93 \text{ watts} * 0,30 \quad (15)$$

$$PD30\% = 25,47 \text{ watts}$$

$$P = 84,93 + 25.47 \text{ watts} \quad (16)$$

$$P = 110,409 \text{ watts}$$

$$A 25^\circ C \quad 110,409 \text{ watts} * 1 = 110,409 \text{ watts} \quad (17)$$

$$\text{Capacidad} = \frac{110,409 \text{ watts}}{11 \text{ voltios}} \quad (18) \quad \text{Capacidad} = 10.03 \text{ AH}^{32}$$

Acorde a la capacidad obtenida se buscó en el mercado una batería que suministre la capacidad necesaria, por lo cual se escogió la batería de litio LIPO de 11.1 voltios 10000mA.

## 2.5 Sistema mecánico

Las articulaciones que conforman el prototipo de la mano están realizados a partir de un modelo 3D obtenido en la página ThingVerse<sup>33</sup>, el modelo permite ser modificado acorde a las necesidades de movimiento mediante el programa 3D Builder. En la figura 2.4 se observa

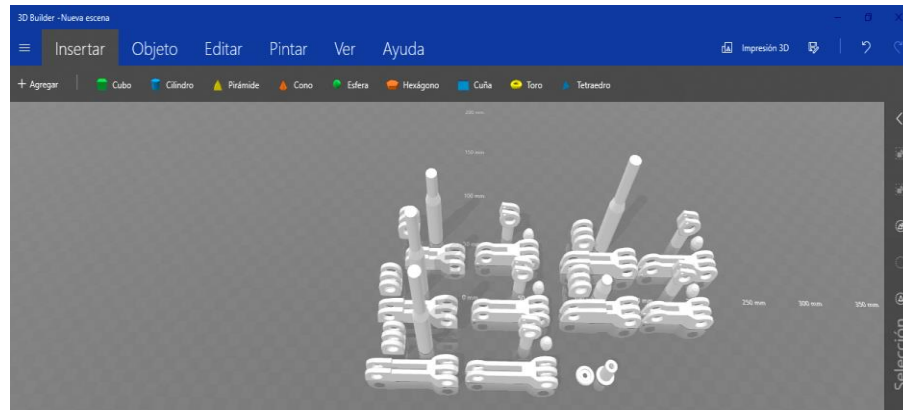
---

<sup>32</sup> Fuente: (Cavadevices, 2018)

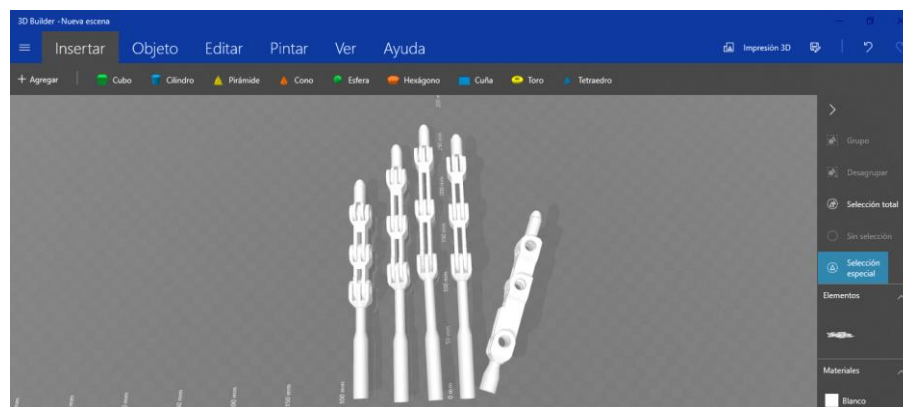
Fuente: <http://www.cavadevices.com/archivos/FOLLETOS/calculo%20de%20bateria.pdf>

<sup>33</sup> <https://www.thingiverse.com/thing:1568293>

las piezas que conforman cada dedo y en la figura 2.5 se tiene una vista previa de las articulaciones armadas.



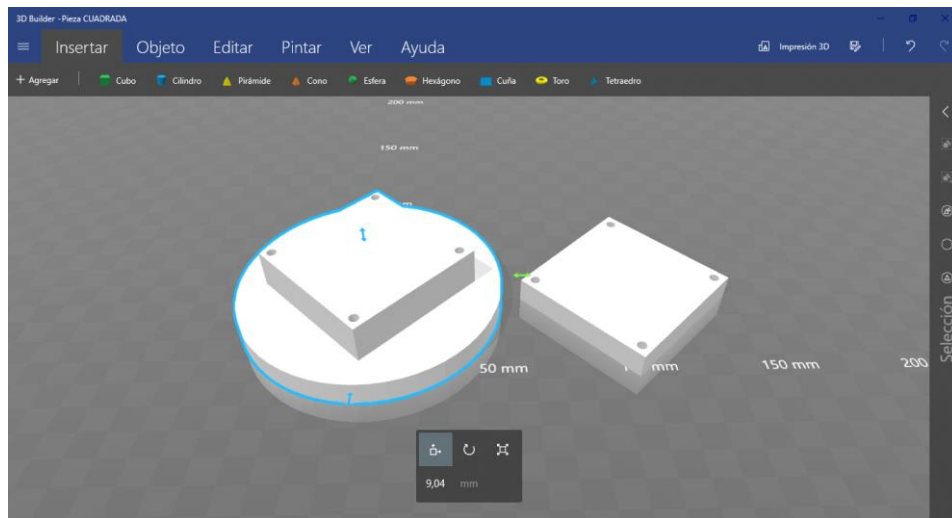
**Figura 2.4 Articulaciones de modelo ThingVerse**  
Fuente: Elaborado por el autor



**Figura 2.5 Estructura de articulaciones de la mano**  
Fuente: Elaborado por el autor

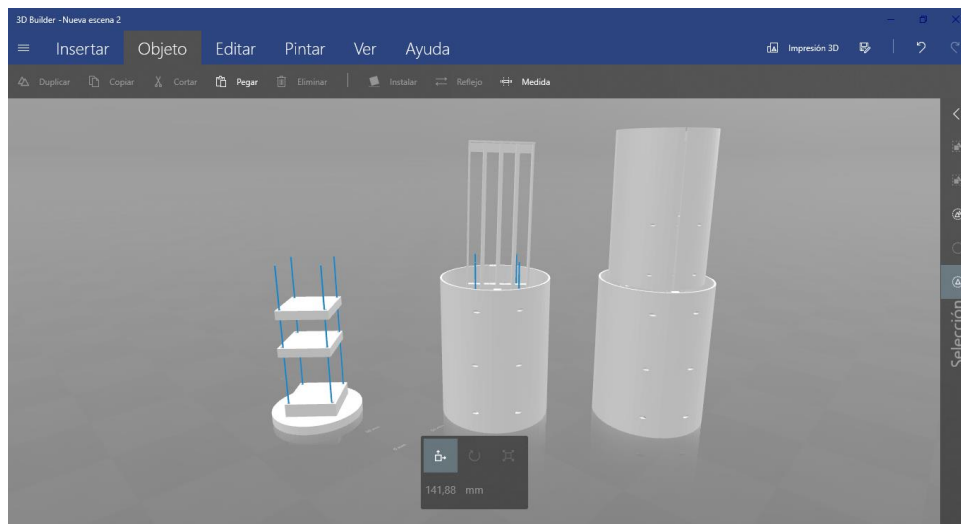
En la figura 2.6 se encuentran las piezas que conforman la base del antebrazo donde se colocan las placas de los circuitos impresos. La primera pieza está conformado por una circunferencia de 10 centímetros de diámetro y un cuadrado con arista de 5 centímetros, el grosor que conforma las dos piezas es de dos centímetros, la segunda pieza es un cuadrado con arista de 5 centímetros y un grosor de 1 centímetro. Las dos piezas están conformados por 4 orificios de 2 milímetros de diámetro cuya finalidad es de unir las piezas indicadas, separadas 12 centímetros.





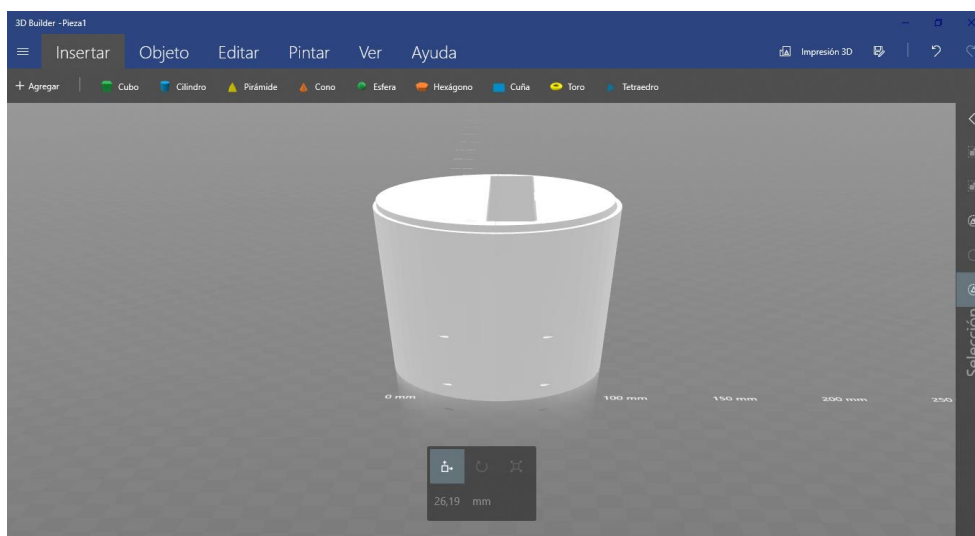
**Figura 2.6** Piezas que conforman estructura de antebrazo  
**Fuente:** Elaborado por el autor

Una vez colocados los circuitos impresos en la estructura de la base del antebrazo, se instaló una protección que separa la parte externa e interna, y este se encuentra conformado por un cilindro con un diámetro de 12.5 centímetros y una altura de 16 centímetros el grosor del cilindro es de dos milímetros. En la figura 2.7 se puede visualizar la estructura de la base y la protección que la recubre.



**Figura 2.7** Estructura de antebrazo  
**Fuente:** Elaborado por el autor

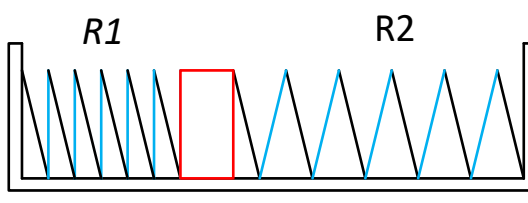
En la figura 2.8 se visualiza el segundo cilindro creado con un diámetro de 10 centímetros y una altura de 16 centímetros, este cilindro se encarga de proporcionar la protección del sistema de movimiento creado con las aleaciones de memoria de forma.



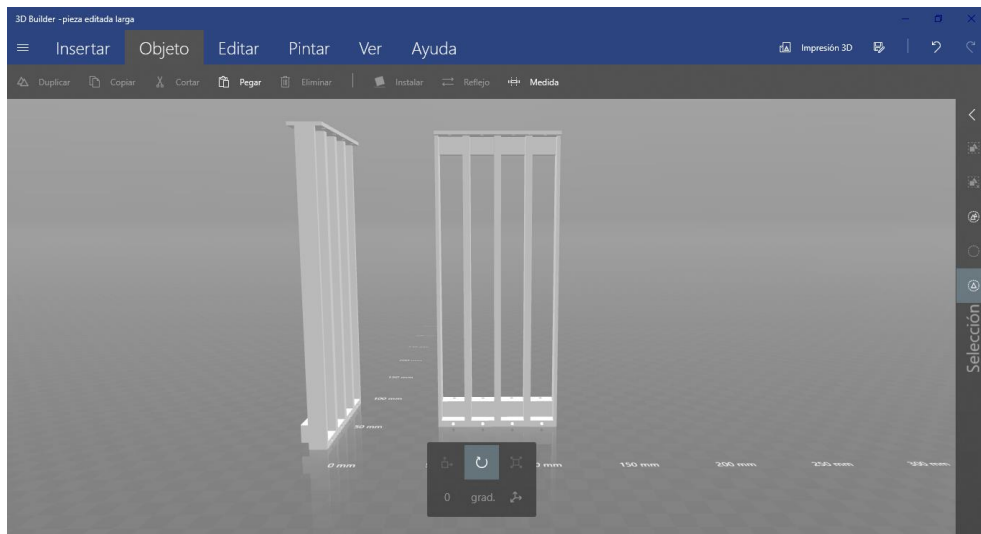
**Figura 2.8 Estructura de antebrazo superior**  
**Fuente: Elaborado por el autor**

El movimiento de las articulaciones será generado por el desplazamiento lineal provocado por el material inteligente Nitinol, cada desplazamiento tiene una relación de (1.5 a 2) cm en cada articulación o dedo. Con el fin de crear este desplazamiento lineal se escogió el uso de resortes hechos de Nitinol por el fabricante Dynalloy.

Con los datos de fuerza proporcionado por el fabricante que fueron mencionados en el Capítulo 1, se diseñó el mecanismo que permita generar el movimiento lineal de la articulación con los resortes de Nitinol como se indicada en la figura 2.9, donde existe un desplazamiento vertical producido por el resorte 1 por el efecto de su memoria permitiendo que la articulación se contraiga, mientras que el resorte 2 se desplaza una distancia igual al desplazamiento del resorte 1; debido que los resortes de Nitinol solo disponen de una sola memoria de forma, se requiere de otra fuerza que permita regresar el resorte 1 a su forma de la fase Martensita y esto será producido al momento que se active la memoria de forma del resorte 2, produciendo el movimiento contrario al indicado inicialmente.



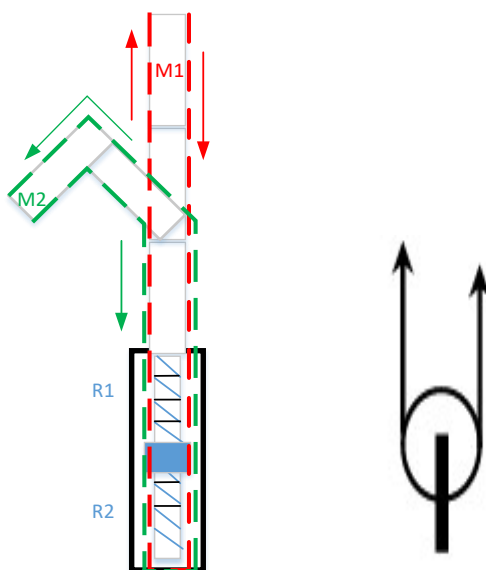
**Figura 2.9 Mecanismo de resorte de Nitinol en serie**  
**Fuente: Elaborado por el autor**



**Figura 2.10 Estructura de fijación de resortes de Nitinol**  
**Fuente: Elaborado por el autor**

Con las consideraciones indicadas anteriormente se diseñó la estructura sobre el cual se coloca los resortes de Nitinol como se indica en la figura 2.10, sus dimensiones son de 17.5 centímetros de alto y 7 centímetros de ancho.

En la figura 2.11 se visualiza el funcionamiento del sistema mecánico conformado por la estructura, los resortes de Nitinol y un filamento de nylon que se encuentra sujeto en la parte frontal y posterior del extremo de la articulación y recorre los puntos de guía de la estructura formando un sistema semejante a las poleas. En el momento que se active el resorte R2 se produce un movimiento vertical M2 que permite la contracción del dedo y al mismo tiempo el filamento de nylon realiza un recorrido igual a la distancia desplazada del resorte R2; En el momento que se active el resorte R1 permitirá que el resorte R2 regrese a su estado inicial y simultáneamente producirá un desplazamiento M1 que permitirá que la articulación regrese a su estado original.



**Figura 2.11 Movimiento de un dedo**  
Fuente: Elaborado por el autor

## 2.6 Diseño de software

Para el control inalámbrico del prototipo de mano robótica se ha desarrollado una aplicación móvil en una plataforma libre como es Android, la aplicación móvil fue desarrollada mediante el programa APP Inventor versión 2, el cual es una aplicación online gratuita al cual se puede acceder mediante una cuenta Gmail. Una de las ventajas por el cual se ha escogido este software es por su portabilidad que permite acceder al software en diferentes puntos geográficos.

El diseño del software se basó en las siguientes preguntas:

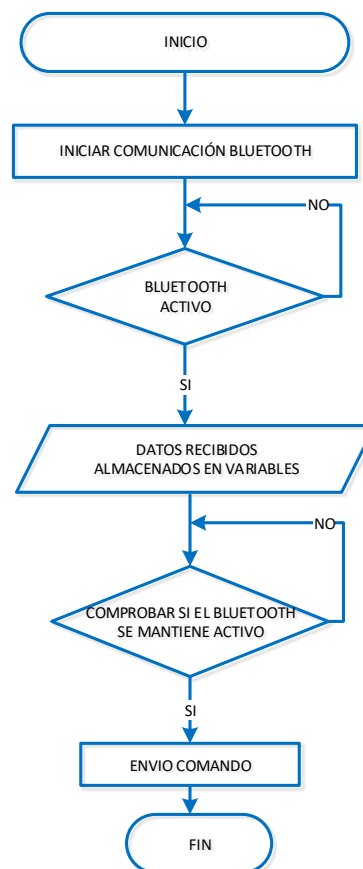
- ¿Qué tipo de comunicación inalámbrica se utilizará para recibir y enviar información?
- ¿La información recibida tiene que ser visualizada en la aplicación?
- ¿Qué tipo de información se debe enviar?

Para el intercambio de información entre el módulo Arduino y el dispositivo Smartphone se utilizó la tecnología *Bluetooth*.

La información que se recibirá del módulo de Arduino es de los sensores de temperatura y fuerza que permitirán al usuario tomar decisiones por lo que es importante visualizar la información, de igual manera se requiere enviar comandos que el micro-controlador intérprete como instrucciones predefinidas.

En base a los cuestionamientos indicados anteriormente se ha elaborado el diagrama lógico mostrado en la figura 2.12.

Al momento que el programa se ejecuta, se inicializa los parámetros de comunicación *Bluetooth*, si la comunicación inalámbrica se encuentra activa permitirá que el dispositivo reciba información del dispositivo asociado y lo almacenara en variables caso contrario el programa espera hasta que se encuentre la comunicación activa. Por otro lado si la comunicación inalámbrica permanece activa el dispositivo *Bluetooth* permitirá el envío de información.

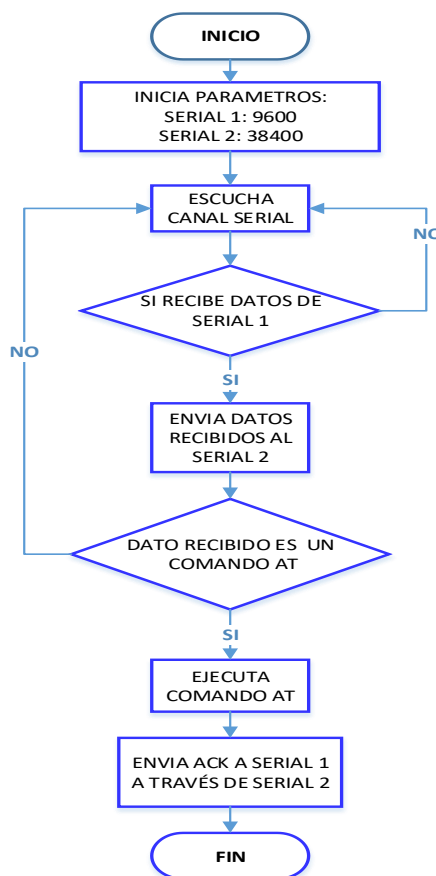


**Figura 2.12 Diagrama de flujo de comunicación Bluetooth**  
**Fuente: Elaborado por el autor**

## 2.7 Comunicación inalámbrica Bluetooth

El dispositivo *Bluetooth* HC-05 viene con una programación de fábrica, el cual puede ser modificado a través de un módulo Arduino en el modo de comando AT.

Para iniciar con la configuración del módulo *Bluetooth* se debe realizar una conexión física en cualquier de los puertos seriales disponibles del módulo Arduino. Una vez realizado las conexiones, se debe iniciar el software de programación del Arduino que permita establecer la comunicación serial tanto del dispositivo *Bluetooth* como de la computadora, posteriormente se ejecutará un programa que permita leer las instrucciones recibidas del computador y ser retransmitidas al dispositivo *Bluetooth* para ser leídos como un comando llamado AT. En la figura 2.13 se pone a consideración el diagrama de bloques que representa al programa para comunicación con el módulo *Bluetooth*.



**Figura 2.13 Diagrama de flujo de configuración de módulo Bluetooth**  
Fuente: Elaborado por el autor

En el momento que se ejecuten los parámetros de comunicación serial, el módulo Arduino escuchará los dos canales seriales establecidos entre el computador y el dispositivo *Bluetooth* HC-05, si se recibe información del puerto serial 1 del computador se retransmitirá hacia el puerto serial del bluetooth quien se encarga de validar si la información recibida es un comando AT, si el comando AT es válido se ejecutará el comando y se envía una señal ACK del serial 2 al serial 1 que confirma la ejecución del comando.

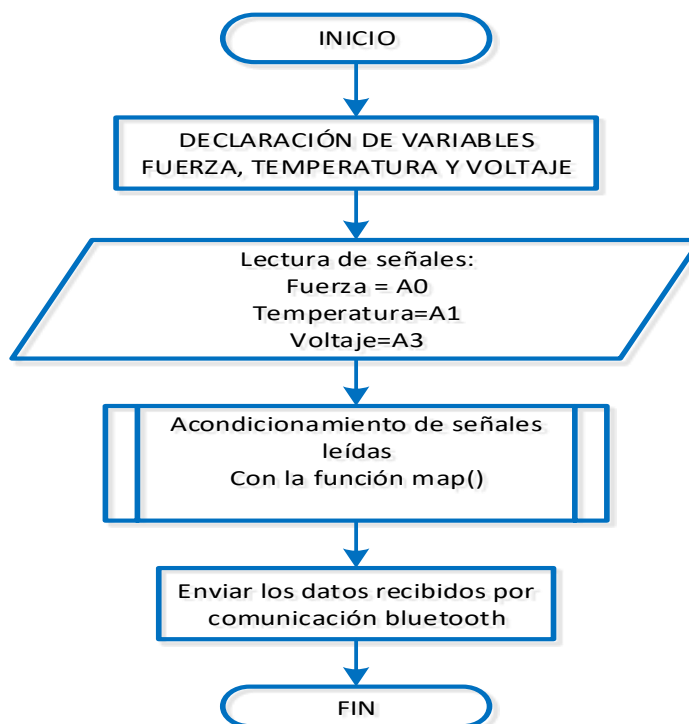
## **2.8 Diseño de programa de módulo Arduino**

Para el diseño de la programación del módulo Arduino se ha definido 2 etapas:

- Sensores
- Control.

En la etapa de sensores se consideró las variables a medir, en este caso: fuerza, temperatura y voltaje. Estas variables serán leídas por el microcontrolador para ser posteriormente acondicionadas y permitan una rápida interpretación de lectura por el usuario.

En el momento que se inicien los parámetros de ejecución del módulo Arduino donde se definen variables y otros, se procederá con la lectura de las señales de los sensores en los pines A0, A1 y A2 y serán almacenados en las variables de fuerza, temperatura y voltaje respectivamente. Una vez obtenidos los datos de la señal se procederá con el acondicionamiento mediante un algoritmo matemático para ser almacenados en nuevas variables. La información almacenada en nuevas variables será transmitida en una comunicación inalámbrica por los puertos seriales. En la figura 2.14 se visualiza el proceso de lectura y acondicionamiento de las señales recibidas por los sensores que conforman el programa principal.

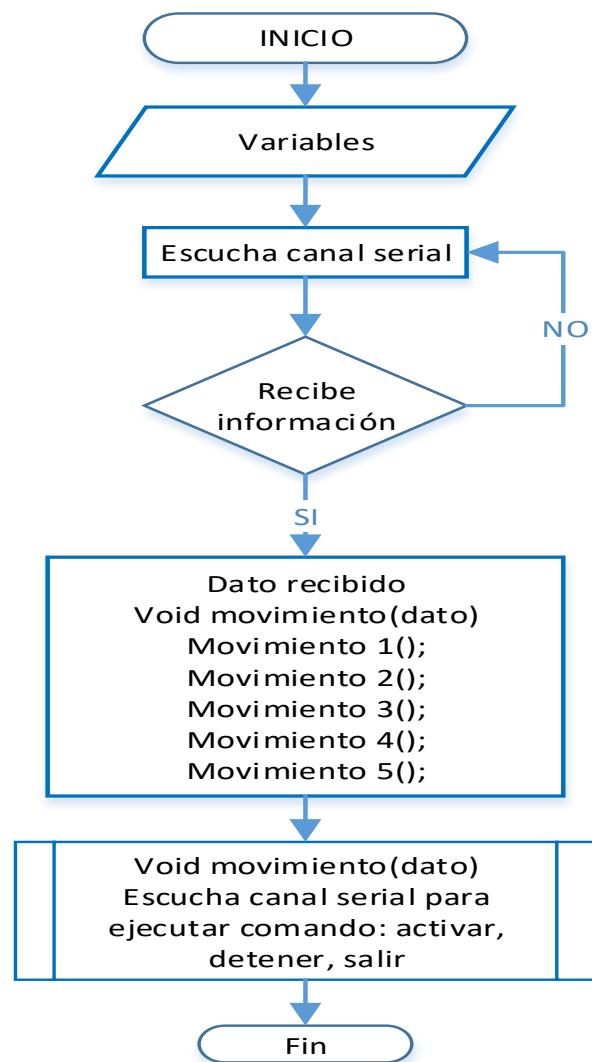


**Figura 2.14 Diagrama de flujo de lectura de sensores**  
**Fuente: Elaborado por el autor**

Para la etapa de control se ha definido la habilitación de comunicación serial para realizar el intercambio de información con el dispositivo Smartphone y el módulo Arduino. Una vez inicializado las librerías y variables en el microcontrolador, el programa espera a recibir una señal por comunicación serial, si recibe alguna información compara con el valor con los datos almacenado en una variable, si los datos coinciden se inicia las funciones acorde a cada movimiento.

A continuación se visualiza en la figura 2.15 el diagrama de bloque que representa el programa del microcontrolador.





**Figura 2.15 Diagrama de flujo de funcionamiento de mano robótica**  
**Fuente: Elaborado por el autor**

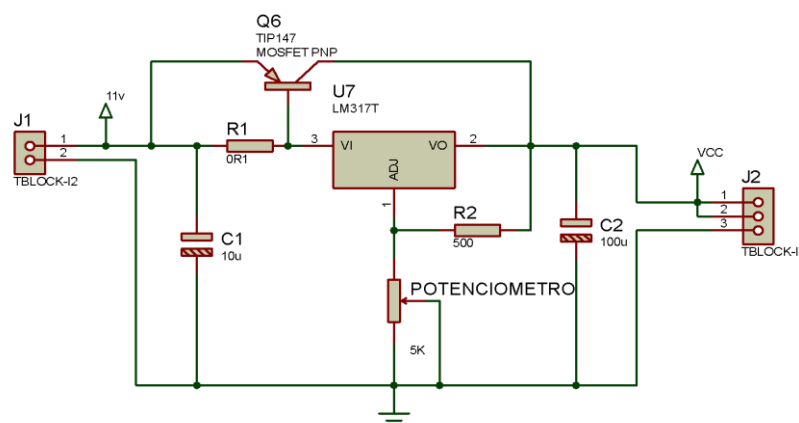
## CAPÍTULO III. IMPLEMENTACIÓN

### 3.1 Desarrollo

#### 3.1.1 Fuente de alimentación

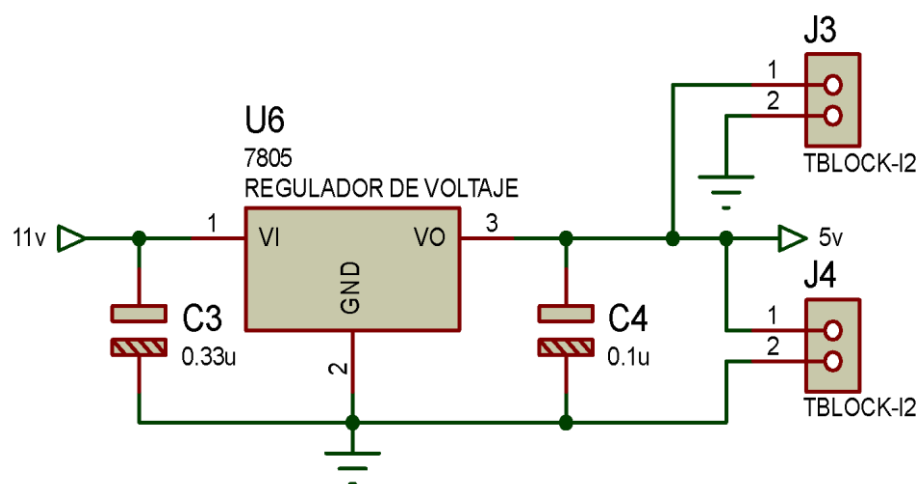
La fuente de alimentación del prototipo es mediante una batería de litio llamada LIPO que permite energizar los dispositivos electrónicos que conforman el sistema de manejo del material inteligente Nitinol, micro-controlador y los elementos periféricos

En la figura 3.1 se puede visualizar el esquema general de alimentación el cual inicia con la batería de litio de 11 voltios 10 amperios hora, el cual ingresa a un regulador de voltaje variable conformado por el integrado LM317T y los capacitadores (10 y 100)  $\mu\text{F}$  que permite generar un voltaje de salida en un rango de 1.5 voltios a 11 voltios que puede ser controlado con un potenciómetro de 5 kilohmios. Aunque se obtiene un voltaje variable, el LM317T entrega una corriente máxima de 1.5 Amperios, motivo por el cual se utilizó un esquema de bypass transistorizado que está conformado por una resistencia de 1 ohmio 10 watts permitiendo energizar el transistor MOSFET PNP TIP147 para que circule una corriente de emisor colector de 10 amperios que será entregado a la salida del regulador de voltaje.



**Figura 3.1 Fuente de alimentación**  
Fuente: Elaborado por el autor

Debido que se tienen varias etapas en el sistema del prototipo que funcionan con un valor determinado de voltaje, se estableció una segunda etapa de alimentación de 5 voltios para energizar los dispositivos que se polarizan con dicho valor. En la figura 3.2, se tiene un regulador de voltaje fijo a 5 voltios que está conformado por el integrado 7805 y dos capacitores (0.33 y 0,1)  $\mu\text{F}$  los cuales son valores propuestos por el fabricante. En el diagrama, J3 representa a los terminales de alimentación para un mini multímetro que es utilizado para censar el voltaje de la fuente variable.



**Figura 3.2 Regulador de voltaje a 5 voltios**  
Fuente: Elaborado por el autor

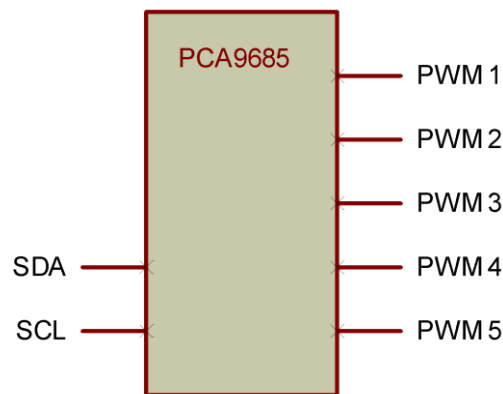
Una vez obtenido el voltaje de referencia de 5 V, se lo utilizó como fuente de alimentación tanto para el módulo Arduino Nano, sensor de temperatura LM35, sensor de fuerza FSR402, módulo PCA9685.

### 3.1.2 Sistema de potencia

El sistema de potencia está conformado por tres etapas:

- Módulo PCA9685.
- Circuito Opto-acopladores.
- Circuito de conmutación de voltaje.

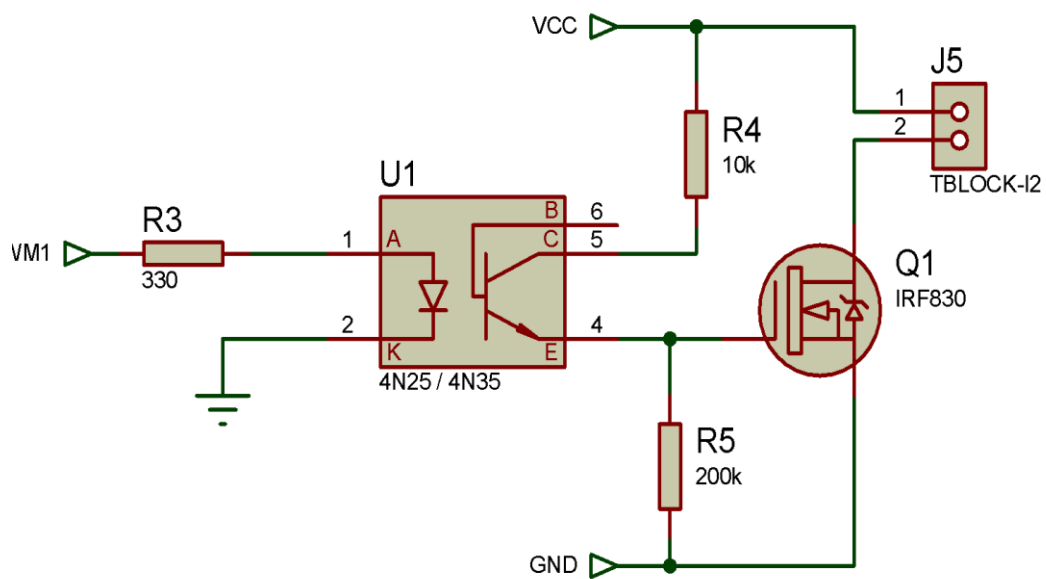
En la figura 3.3, se observa el esquemático del módulo PCA9685, como se indicó en el marco teórico el módulo permite establecer una comunicación I2C al módulo Arduino usando los pines SCL y SDA presente en los dos circuitos.



**Figura 3.3 Módulo PCA9685**  
Fuente: Elaborado por el autor

En la figura 3.4 se visualiza las etapas de los circuitos opto-acopladores y de conmutación de corriente. En la primera etapa se encuentra el opto-acoplador 4N25 que se encuentra conectados mediante una resistencia de 330 ohmios al módulo PCA9685, donde al recibir una señal de 1 lógico provoca que internamente en el opto-acoplador el led se encienda, activando el fototransistor NPN permitiendo que el opto-acoplador entre en saturación, provocando la circulación de una corriente a la etapa de conmutación de potencia.

Acorde al diagrama presentado en la figura 3.4, la etapa de conmutación está conformado por un circuito de alimentación de compuerta común del MOSFET IRFZ44N. En el instante que el circuito de polarización no circula una corriente en el pin de compuerta permite que el transistor MOSFET se comporte como un circuito abierto entre los pines fuente y drenador, impidiendo la circulación de corriente en el resorte de Nitinol ubicado entre la fuente variable y el pin drenador del MOSFET; si el opto-acoplador se satura, entrega una corriente a la compuerta del MOSFET que permite que el MOSFET se comporte como un circuito cerrado, y el resorte de Nitinol inicie su proceso de contracción acorde a la memoria establecida.



**Figura 3.4 Circuito de conmutación de corriente**  
Fuente: Elaborado por el autor

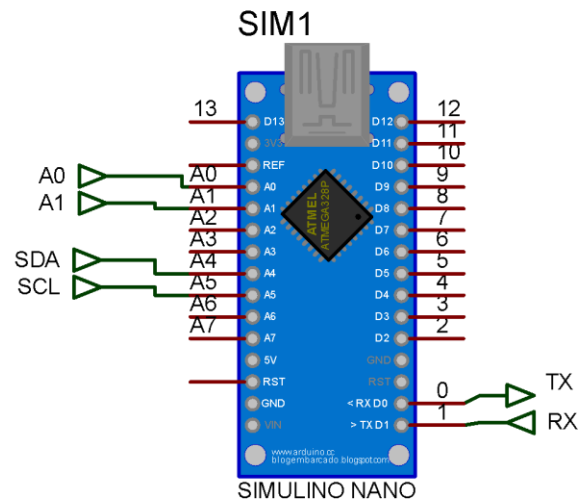
### 3.1.3 Circuitos acondicionadores de sensores

En la figura 3.5 se encuentra las conexiones y circuitos acondicionadores de los sensores de temperatura LM35 y el sensor de fuerza FSR402.

En el caso del sensor de temperatura el integrado LM35 trabaja con una polarización de 5 voltios y acorde al funcionamiento del integrado LM35, permite una conexión directa al módulo Arduino sin la necesidad realizar un circuito que acondicione la señal. Durante el muestreo de los datos obtenidos del sensor se visualizó una alta oscilación de valores medidos el cual se corrigió con la colocación de una resistencia de 10 kilohmios entre el pin de salida y tierra del integrado LM35

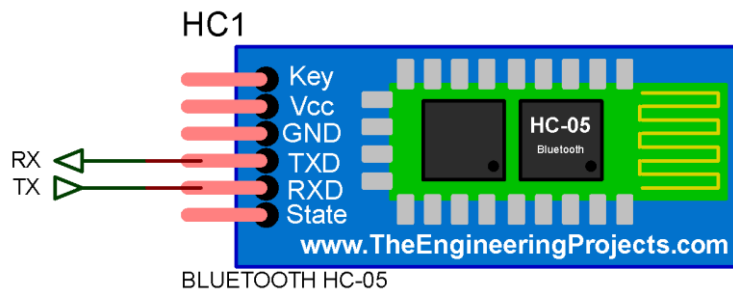
Para el sensor de fuerza FSR402 se acondiciona la señal realizando un circuito de un divisor de voltaje entre el sensor y una resistencia de 10 kilohmios (valor establecido por el fabricante).





**Figura 3.6** Esquemático del módulo Arduino Nano en Proteus

Fuente: Elaborado por el autor



**Figura 3.7** Esquemático del módulo bluetooth HC-05 en Proteus

Fuente: Elaborado por el autor

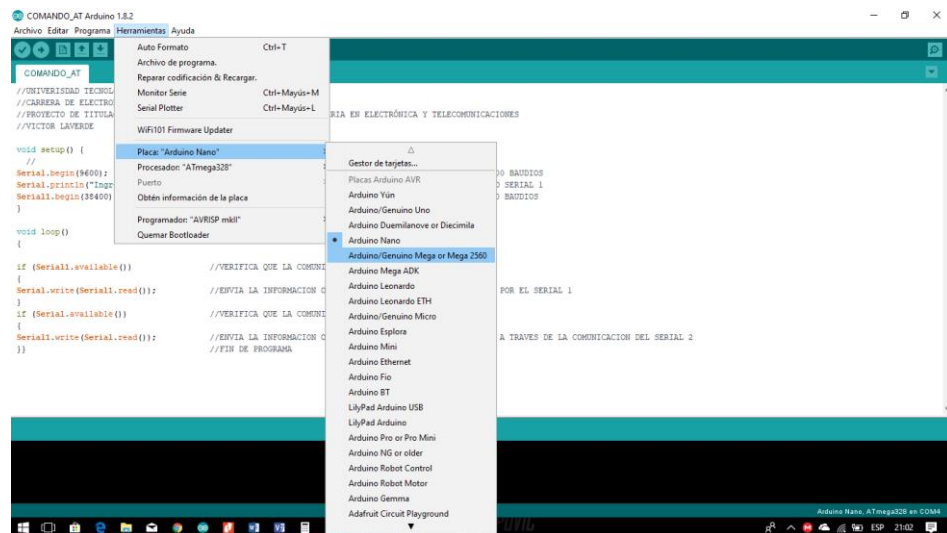
En la Figura 3.8 se muestran el esquema electrónico conformado por el sistema de potencia, de comunicación inalámbrica, microcontrolador y sensores, realizados en el programa de simulación Proteus versión 8.





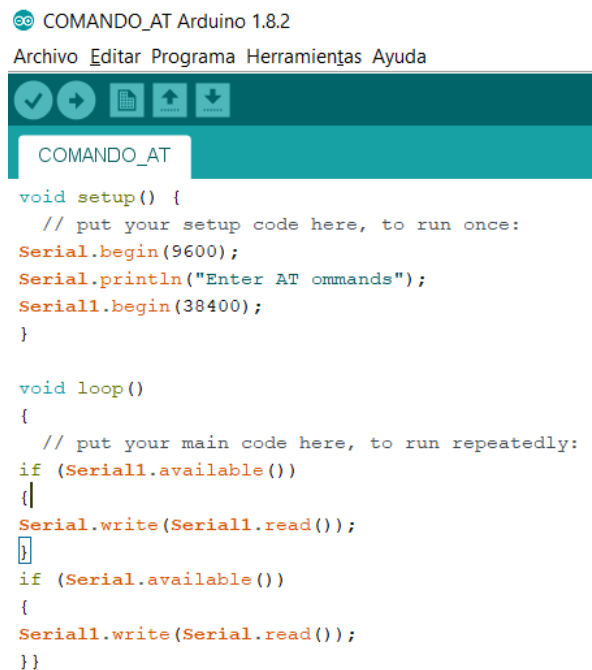
### 3.1.5 Programación de módulo Bluetooth HC-05

Antes de iniciar con la programación del módulo Arduino se seleccionó el tipo de placa a usar, esto se lo realizó desde la opción de Herramientas en la sección de placas, donde se selecciona la placa Arduino Mega el cual es utilizado únicamente para la programación del módulo *Bluetooth* debido que este dispone de 4 puertos de comunicación serial.



**Figura 3.9 Selección de placa Arduino**  
Fuente: Elaborado por el autor

En la figura 3.10 se puede visualizar el programador de fábrica de Arduino con el programa y en la figura 3.11 se muestra la comunicación serial de la laptop al módulo Arduino, donde se puede enviar los comandos AT al dispositivo *Bluetooth*.



```

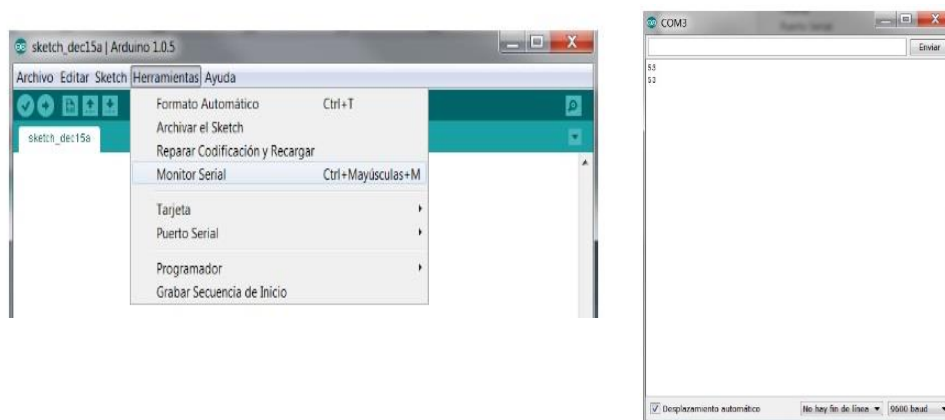
COMANDO_AT Arduino 1.8.2
Archivo Editar Programa Herramientas Ayuda

void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);
  Serial.println("Enter AT ommands");
  Serial1.begin(38400);
}

void loop()
{
  // put your main code here, to run repeatedly:
  if (Serial1.available())
  {
    Serial.write(Serial1.read());
  }
  if (Serial.available())
  {
    Serial1.write(Serial.read());
  }
}

```

**Figura 3.10 Programa para comunicación serial con Bluetooth HC-05**  
**Fuente: Elaborado por el autor**



**Figura 3.11 Comunicación Serial**  
**Fuente: Elaborado por el autor**

En la tabla 3.1 se enumera los parámetros básicos cambiados en el módulo *Bluetooth*, cada cambio de parámetro se tiene un mensaje de retorno que confirma la acción realizada o en su defecto si existe un error.

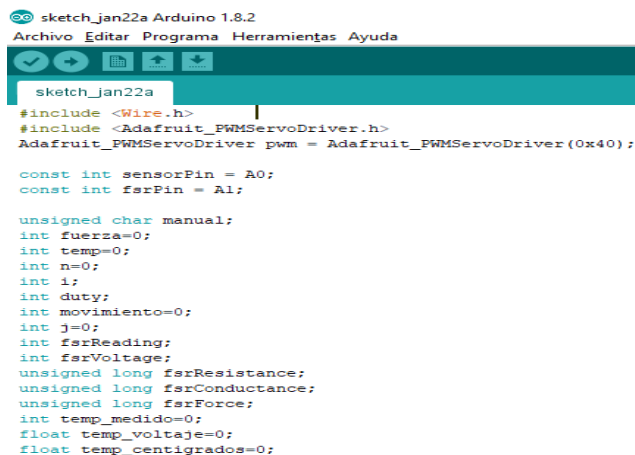
LISTA DE COMANDOS AT	
COMANDO	FUNCIÓN
AT	Test de conexión UART
AT+RESET	Resetear dispositivo
AT+VERSION	Query firmware versión
AT+ORGL	Restaurar configuraciones de fabrica
AT+ADDR	Consulta la dirección del dispositivo Bluetooth
AT+NAME Mano_Robot	Consulta o configura el nombre del dispositivo Bluetooth
AT+PIN Mano1234	Consulta o configura la contraseña del dispositivo Bluetooth

Tabla 3.1 Comandos AT configurados  
Fuente: Elaborado por el autor

### 3.1.6 Programación módulo Arduino Nano v3

La programación del módulo Arduino está basado en el diagrama de bloques presentado en la figura 2.14 y 2.15, el programa inicia con la declaración de variables, puertos analógicos, puertos digitales, inicialización de librerías.

Las librerías a inicializarse son para la comunicación I2C y generación de señales PWM. El tipo de variables a usarse serán del tipo INT, *unsigned long*, *char* y del tipo *float* para realizar el acondicionamiento de las señales recibidas por los sensores, en los puertos analógicos declarados.



```

sketch_jan22a Arduino 1.8.2
Archivo Editar Programa Herramientas Ayuda
sketch_jan22a
#include <Wire.h>
#include <Adafruit_PWMServoDriver.h>
Adafruit_PWMServoDriver pwm = Adafruit_PWMServoDriver(0x40);

const int sensorPin = A0;
const int fsrPin = A1;

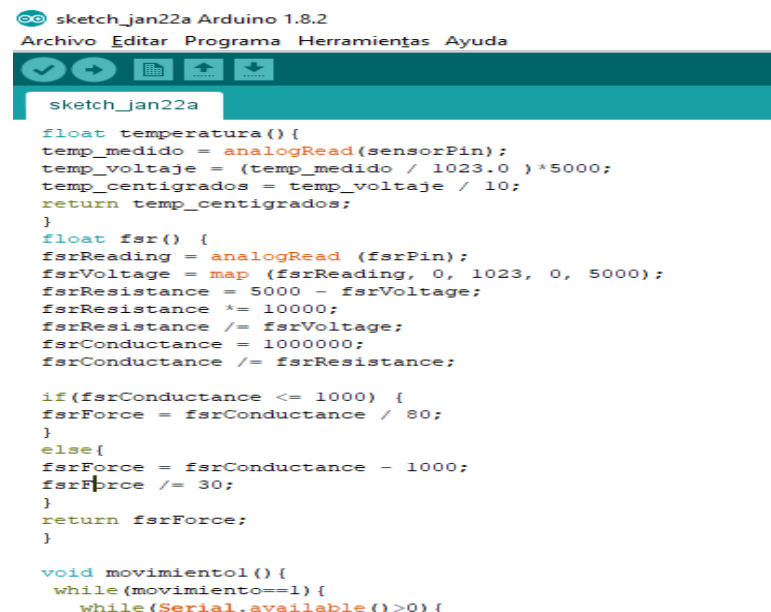
unsigned char manual;
int fuerza=0;
int temp=0;
int n=0;
int i;
int duty;
int movimiento=0;
int j=0;
int fsrReading;
int fsrVoltage;
unsigned long fsrResistance;
unsigned long fsrConductance;
unsigned long fsrForce;
int temp_medido=0;
float temp_voltaje=0;
float temp_centigrados=0;

```

Figura 3.12 Programación de inicialización de librerías  
Fuente: Elaborado por el autor

Posteriormente se inicia los parámetros de comunicación serial, I2C, asignación de frecuencia a 2 Hz para la señal PWM y designación de los puertos usados como entrada y salidas.

En el cuerpo del programa se basa en el diagrama de flujo de la figura 2.9, se realiza la llamada a funciones referente a la toma de lectura de las señales de los sensores y acondicionamiento de los mismos.



```

sketch_jan22a Arduino 1.8.2
Archivo Editar Programa Herramientas Ayuda
sketch_jan22a
float temperatura() {
  temp_medido = analogRead(sensorPin);
  temp_voltaje = (temp_medido / 1023.0 ) * 5000;
  temp_centigrados = temp_voltaje / 10;
  return temp_centigrados;
}
float fsr() {
  fsrReading = analogRead (fsrPin);
  fsrVoltage = map (fsrReading, 0, 1023, 0, 5000);
  fsrResistance = 5000 - fsrVoltage;
  fsrResistance *= 10000;
  fsrResistance /= fsrVoltage;
  fsrConductance = 1000000;
  fsrConductance /= fsrResistance;

  if (fsrConductance <= 1000) {
    fsrForce = fsrConductance / 80;
  }
  else {
    fsrForce = fsrConductance - 1000;
    fsrForce /= 30;
  }
  return fsrForce;
}

void movimientol() {
  while (movimiento==1) {
    while (Serial.available() > 0) {

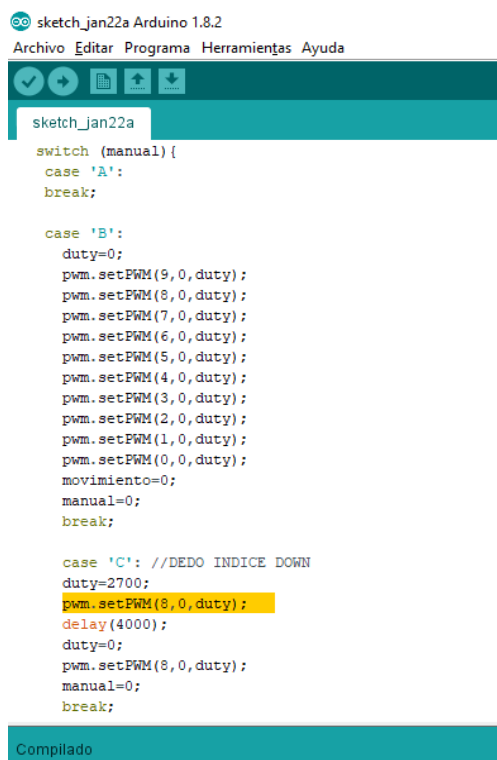
```

**Figura 3.13 Programación de acondicionamiento de señales de sensores**  
Fuente: Elaborado por el autor

Una vez que se toma la lectura de la señal entregada por los sensores de temperatura, fuerza y voltaje se realiza el acondicionamiento de las señales recibidas; en el caso de la función de temperatura la señal recibida se muestrea con un valor en el rango de 0 a 1023 niveles, debido que el valor recibido no es comprensible, se utiliza una fórmula para que el valor medido se exprese como un voltaje referencia entre 0 a 5000 milivoltios y posteriormente se pueda manifestar como una temperatura tomando en cuenta la resolución de 10 milivoltios por grado centígrado indicado en la hoja del fabricante. En el caso de la señal recibida del sensor de fuerza se optó por usar la función “*map()*” de Arduino que permite transformar el rango medido entre 0 a 1023 en otro rango en este caso de 0 a 5000. Obtenido el valor requerido se acondiciona la señal para obtener la

resistividad y conductividad del FSR, se aplica una función de condición (If <= 1000) para determinar la fuerza en una medida de Newtons.

Dentro del cuerpo del programa se aplica una condición de *While* que permite escuchar el canal de comunicación serial, en el momento de obtener datos en el buffer de entrada se almacena la información recibida en una variable, se compara y determina a qué tipo de movimiento pertenece mediante la función de *switch case* para posteriormente hacer una llamada de función acorde al movimiento. De acuerdo al proyecto propuesto se realizó 5 tipos de movimientos: libre, cerrado, gancho, circular y lateral; en cada movimiento se escucha nuevamente el canal de comunicación serial para determinar la ejecución del movimiento mediante la función `pwm.setPWM(7,0,duty)` donde 7 es el número de canal del módulo PCA9685 el mismo que varía entre 0 y 15 canales, 0 y la variable “*duty*” determinan el flanco de inicio y fin del ancho del pulso PWM.



```

sketch_jan22a Arduino 1.8.2
Archivo Editar Programa Herramientas Ayuda
sketch_jan22a
switch (manual){
  case 'A':
    break;

  case 'B':
    duty=0;
    pwm.setPWM(9, 0, duty);
    pwm.setPWM(8, 0, duty);
    pwm.setPWM(7, 0, duty);
    pwm.setPWM(6, 0, duty);
    pwm.setPWM(5, 0, duty);
    pwm.setPWM(4, 0, duty);
    pwm.setPWM(3, 0, duty);
    pwm.setPWM(2, 0, duty);
    pwm.setPWM(1, 0, duty);
    pwm.setPWM(0, 0, duty);
    movimiento=0;
    manual=0;
    break;

  case 'C': //DEDO INDICE DOWN
    duty=2700;
    pwm.setPWM(8, 0, duty);
    delay(4000);
    duty=0;
    pwm.setPWM(8, 0, duty);
    manual=0;
    break;
}
Compilado

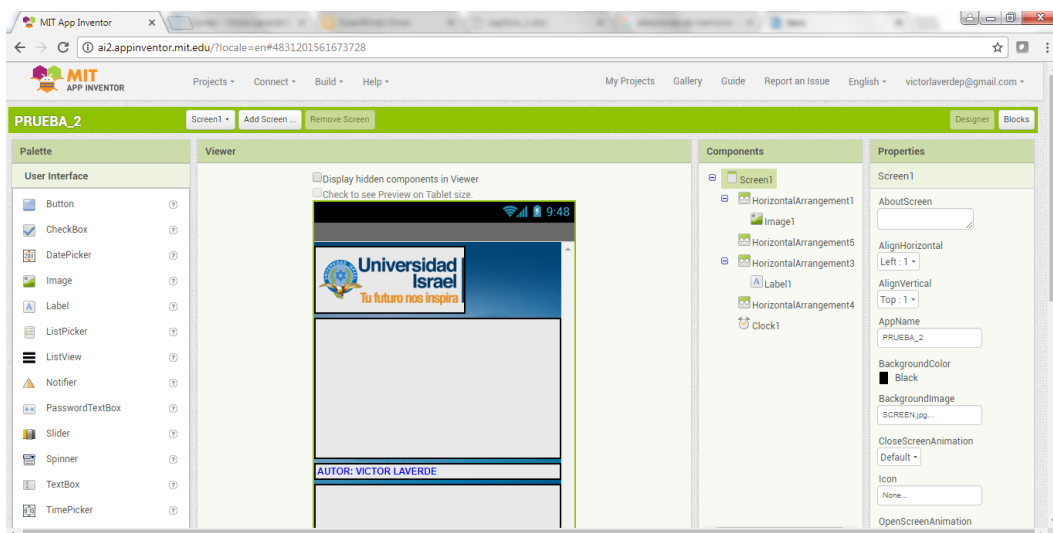
```

**Figura 3.14 Programación de selección y generación de movimiento**  
Fuente: Elaborado por el autor

### 3.1.7 Programación de aplicación de dispositivo Smartphone en APP Inventor 2

La aplicación desarrollada para el dispositivo *Smartphone* o teléfono inteligente fue creada en la plataforma APP Inventor 2, el mismo que es un software libre presentado por la empresa Google. Una vez ingresado en la plataforma web se debe crear un nuevo proyecto con el nombre de la aplicación, en el cual consta con una opción gráfica que permite agregar texto, iconos, imágenes y funciones que desempeña la aplicación.

En la figura 3.15 se dispone de una plantilla base que tiene como objetivo la presentación de la aplicación y posteriormente iniciar con la aplicación principal, el tiempo que toma pasar de una plantilla a otra es realizado por la función reloj que se encuentra configurado con un tiempo previamente establecido a 5 segundos.

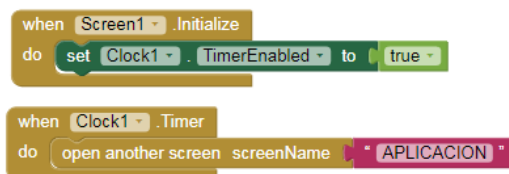


**Figura 3.15 Aplicación APP inventor 2**  
Fuente: Elaborado por el autor

La programación de la pantalla de inicio *screen 1* se encuentra los elementos que conforman la estructura del programa formado por bloques, cada bloque puede representar una función matemática, lógica, específica o variable.

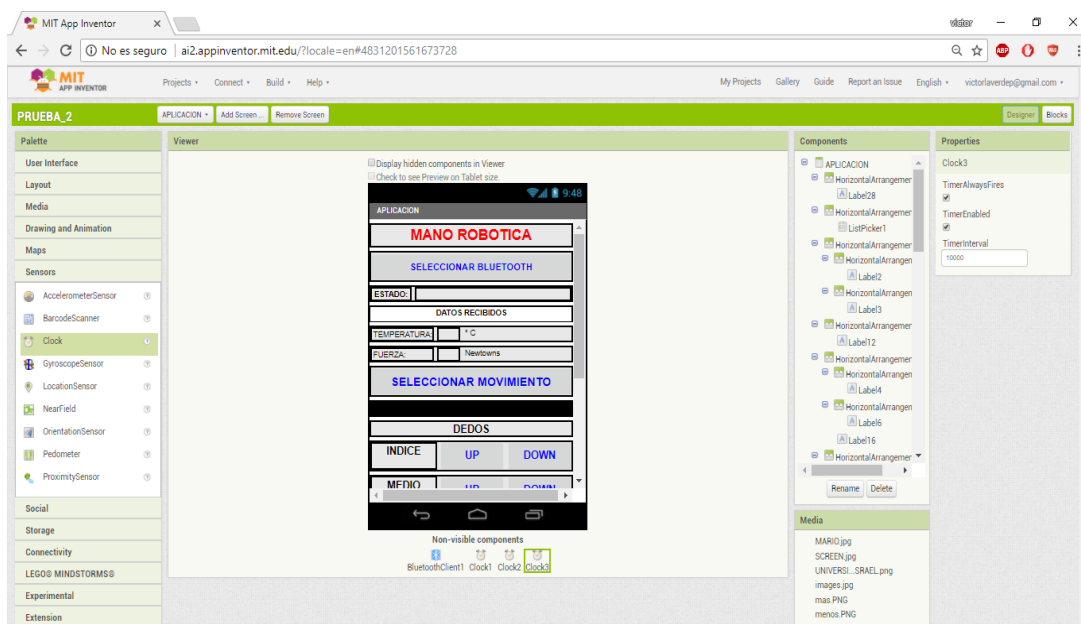
A continuación se detalla la programación del *screen 1*:

- Cuando se inicializa la pantalla screen1, valida que la función reloj se encuentra habilitado.
- Cuando la función reloj culmine su tiempo establecido anteriormente, abre otro *screen* llamado aplicación.



**Figura 3.16 Programación de Screen 1**  
Fuente: Elaborado por el autor

En la figura 3.17 se puede visualizar la aplicación principal de la mano robot donde se encuentra conformado el menú de herramientas, *screen* (aplicación en la parte central de la pantalla), el listado de componentes y sus propiedades.



**Figura 3.17 Aplicación de prototipo de mano robot**  
Fuente: Elaborado por el autor

Como se observa la pantalla de la figura 3.17, se ubican en el lado derecho el menú de herramientas, que se utilizan para el diseño de una aplicación, en este caso el menú se encuentra separado por varios sub categorías, entre las más esenciales se encuentran:

- *Use interface*: En esta categoría se encuentran los elementos activos que permiten generar botones, iconos, cuadros de texto, diálogos.
- *Layout*: En esta categoría se encuentran bloques que permiten realizar la separación vertical u horizontal entre los elementos activos como botones, cuadros de texto, de igual manera se encargan de proporcionar un orden al momento de crear el *screen*.
- *Sensors*: En esta categoría se realiza el uso de los sensores internos del teléfono o dispositivos internos, entre los cuales se encuentran *Bluetooth*, reloj, acelerómetro, etc.

Con la información indicada, se menciona a continuación los elementos utilizados para la creación del *screen* aplicación:

- *Button*: Permite crear botones
- *Label*: Permite ingresar textos
- *ListPicker*: Permite crear listas
- *Layout Horizontal*: Permite realizar separaciones
- *Bluetooth*: Activa la función de Bluetooth
- *Clock*: Permite generar contadores

Al momento de seleccionar cualquiera de los elementos mencionados anteriormente, se realizó el cambio en sus propiedades, entre los cuales se encuentran:

- Cambio de color de fondo
- Cambio de tamaño de texto
- Cambio de tamaño del elemento
- Cambio de color de texto



- Opción de insertar gráficos
- Opción de ingreso de texto

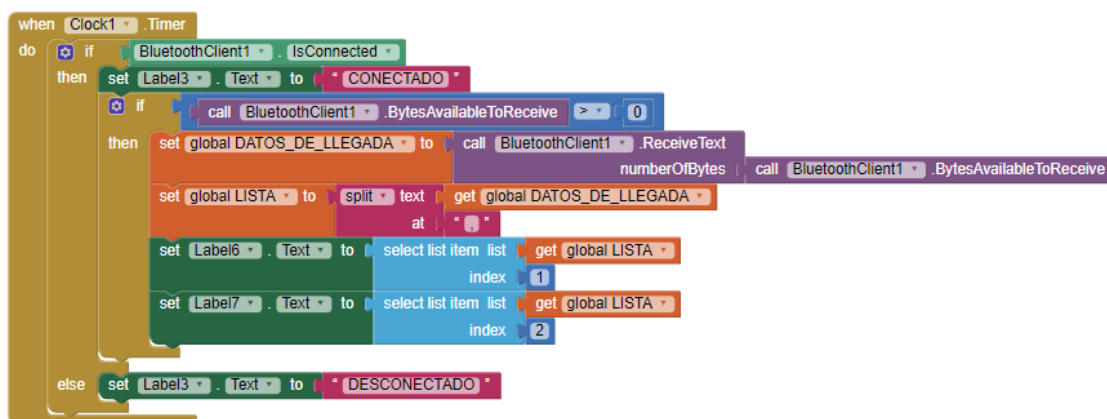
En el instante que es presionado el botón *listPiker* se establecen las funciones de reloj y *Bluetooth* que permite establecer la comunicación entre dos dispositivos inalámbricos que se encuentran activos.

Como se observa en la figura 3.18, se dispone de la programación por bloques del funcionamiento de la aplicación, el cual se detalla a continuación:

Mientras se encuentre enlazados los dispositivos *Bluetooth*, se refleja el estado de comunicación como “Conectado”, caso contrario se visualizará en la aplicación como “Desconectado”.

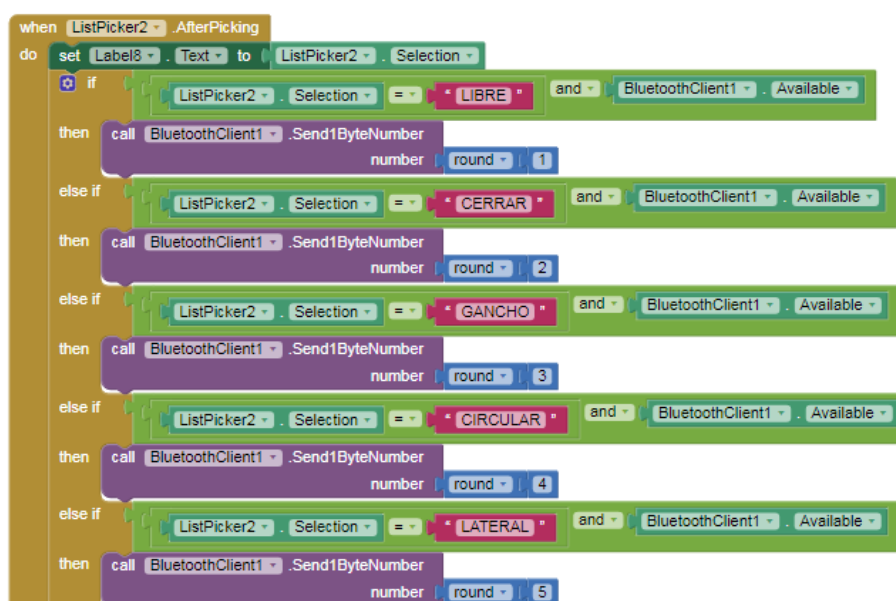
En el instante que se recibe datos (*bytes*) se guarda la información en una variable llamada “Datos\_de\_llegada”, posteriormente se segmenta la información de la variable en el instante que se dispone del símbolo establecido “,” y se almacena en una nueva variable llamada “lista”.

Los datos de la variable “lista” se muestran como texto en la aplicación.



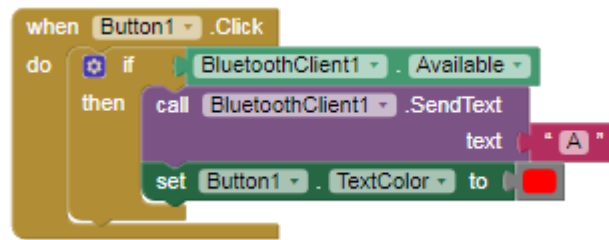
**Figura 3.18 Programación de establecimiento de comunicación inalámbrica**  
**Fuente: Elaborado por el autor**

Una de las funciones de la aplicación es seleccionar 5 diferentes tipos de movimiento que se quiere generar en la mano robótica. En el momento que se selecciona el movimiento de la lista de *ListPiker2* el programa se encarga de validar si la comunicación *Bluetooth* se encuentra activa y el movimiento que fue escogido, si la comparación es verdadera entonces la función *Bluetooth* se encarga de enviar información como un número decimal, como se muestra en la figura 3.19.

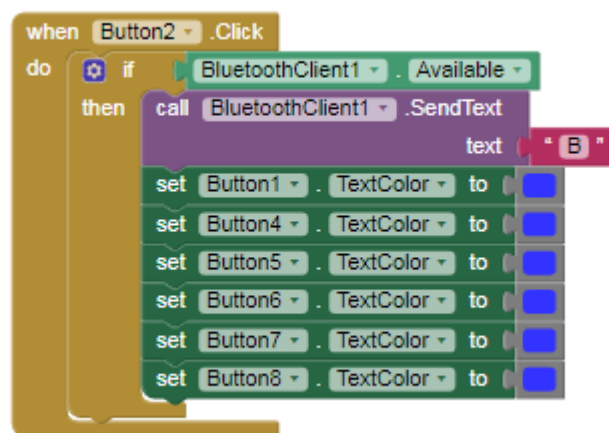


**Figura 3.19 Programación de selección de movimiento**

Una vez seleccionado el movimiento se dispone de dos botones en la aplicación móvil que permiten activar el movimiento de las aleaciones de memoria de forma o en su defecto las detenga. En las figuras 3.20 y 3.21 se dispone la programación referente a las acciones que realiza los botones indicados anteriormente, en el instante que es presionado el botón de activar se valida si el *Bluetooth* es activo mediante la función *if then*, y si este lo está se encarga de enviar el texto “A” mediante la función *blueooth*, por otro lado si se presiona el botón detener (figura 3.21) se valida si el *Bluetooth* esta activo y si este se encuentra activo se envía el texto “B”.



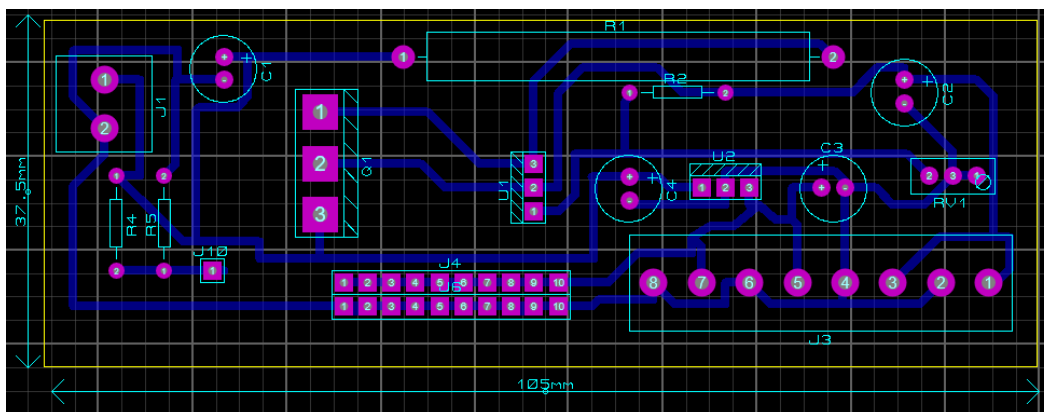
**Figura 3.20 Programación de activación de movimiento**  
Fuente: Elaborado por el autor



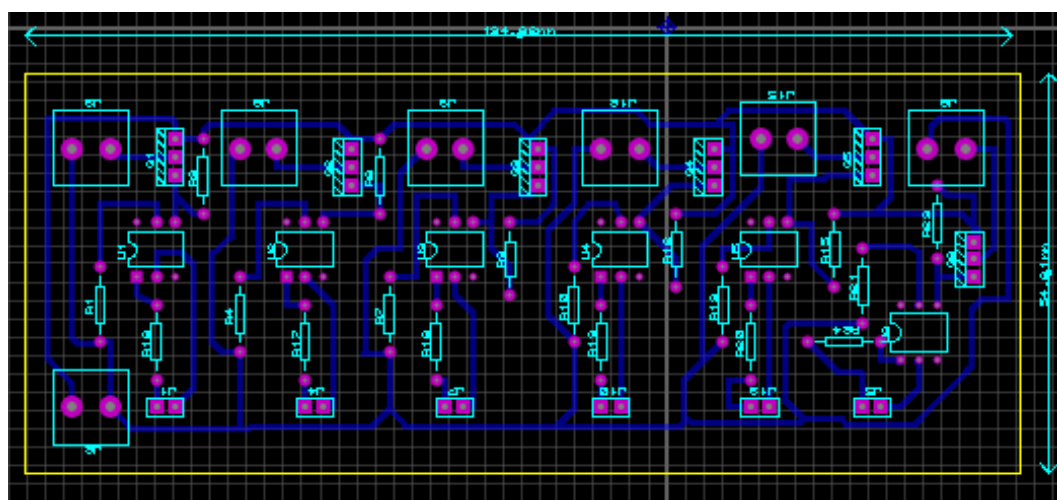
**Figura 3.21 Programación para detener movimiento**  
Fuente: Elaborado por el autor

### 3.2 Implementación

Para la implementación de las placas electrónicas se realizó el diseño de los circuitos impresos con el programa Ares de Proteus versión 8, en las consideraciones que se realizó durante el diseño es el espacio libre entre los componentes, con el propósito de evitar que las pistas puedan causar cortocircuitos entre sí; de igual manera se consideró la ubicación de los elementos que requieren de disipación de temperatura, por lo general se requiere la utilización de disipadores de calor por lo cual se requiere de espacio adicional en la placa del circuito.

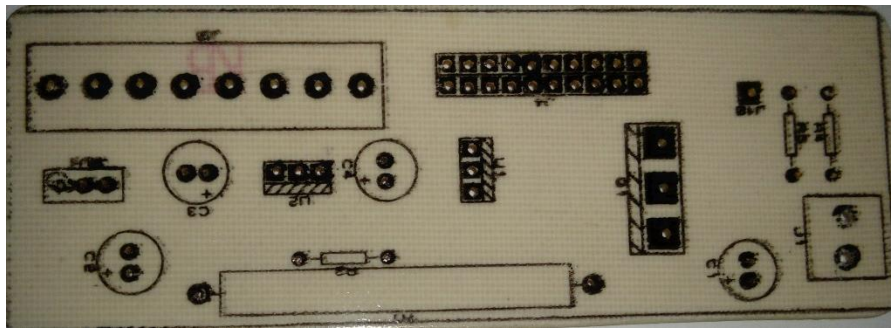


**Figura 3.22** Diseño de placa impresa de la fuente de alimentación  
Fuente: Elaborado por el autor

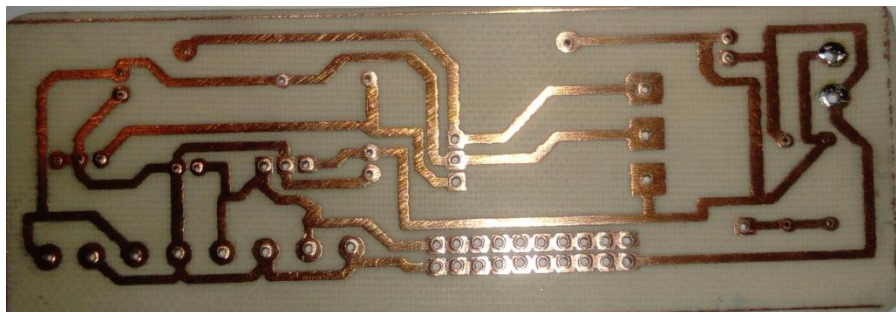


**Figura 3.23** Diseño de placa de circuito de potencia  
Fuente: Elaborado por el autor

En las siguientes figuras se presentan las fotografías que hacen referencia a los circuitos realizados en PCB de la fase de alimentación y de potencia. Los circuitos fueron realizados en un tamaño de (10.5 x 3.5) centímetros y (13.5 x 5) centímetros correspondientemente, el método utilizado para la fabricación de las placas electrónicas fue mediante la técnica de transferencia de calor y dilución en ácido férrico. Como se observa en las siguientes figuras 3.24 y 3.25 se encuentran la placa impresa del circuito que regula el voltaje de la batería de litio.

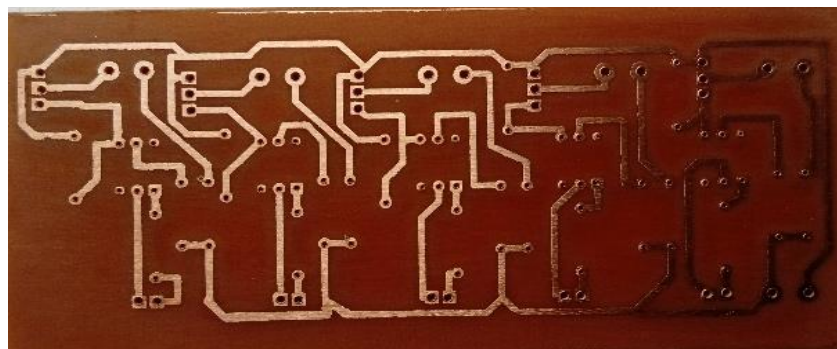


**Figura 3.24 Circuito impreso de fuente de alimentación lado A**  
Fuente: Elaborado por el autor



**Figura 3.25 Circuito impreso de fuente de alimentación lado B**  
Fuente: Elaborado por el autor

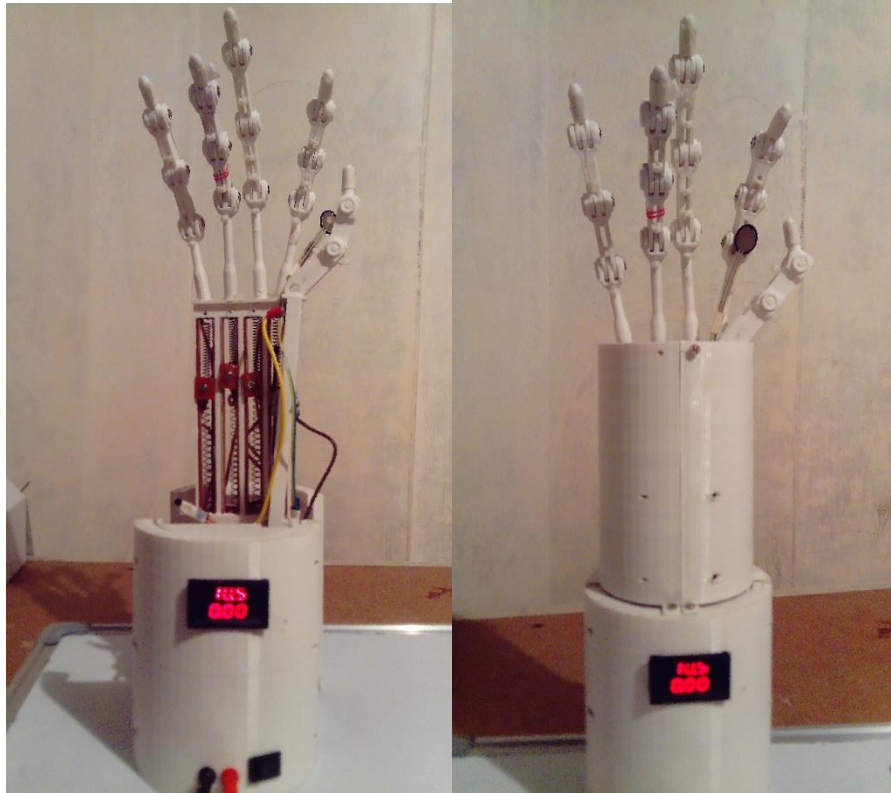
En la figura 3.26 se encuentra la placa del circuito de potencia que permite controlar los resortes de Nitinol con el voltaje entregado por la fuente de alimentación.



**Figura 3.26 Circuito impreso de Potencia**  
Fuente: Elaborado por el autor

Los circuitos electrónicos y el módulo Arduino son implementados en dos secciones de la estructura que simula el antebrazo presentado en el capítulo del diseño (figura 3.27) y en

la figura 3.28 se visualiza el prototipo de mano robótica finalizado con la colocación de la estructura que protege a los circuitos y resortes de Nitinol



**Figura 3.27 Prototipo de mano robótica ensamblado**  
**Fuente: Elaborado por el autor**

## CAPÍTULO IV PRUEBAS Y ANÁLISIS DE RESULTADOS

### 4.1 Pruebas de funcionamiento

Acorde a la información proporcionada por el fabricante, los resortes de Nitinol pueden tener una contracción en un tiempo menor a 1 segundo al ser sometido a una corriente de 2 amperios, durante las pruebas realizadas la contracción del resorte dependió del pulso aplicado; pero la corriente eficaz aplicado por el pulso PWM fue menor que una corriente suministrada constantemente provocando el incremento de tiempo en la contracción del resorte de Nitinol.

Una vez implementado los circuitos electrónicos a la estructura de la prótesis de mano, se realizó pruebas de funcionamiento mediante la variación de frecuencia y pulso PWM.

En la tabla 4.1 se expone el desplazamiento en comparación a variaciones en el pulso y frecuencia utilizada.

Articulación	Frecuencia (Hz)	Ancho de Pulso	Ancho de Pulso %	Tiempo de Contracción (Seg)	Desplazamiento (mm)
Índice	2	1024	25%	5	20
Índice	2	2048	50%	5	20
Índice	2	4000	97,70%	5	20
Índice	2	1024	25%	10	20
Índice	2	2048	50%	10	20
Índice	2	4000	97,70%	10	20
Índice	100	1024	25%	5	20
Índice	100	2048	50%	5	20
Índice	100	4000	97,70%	5	20
Índice	100	1024	25%	10	20
Índice	100	2048	50%	10	20
Índice	100	4000	97,70%	10	20
Índice	350	1024	25%	5	20
Índice	350	2048	50%	5	20
Índice	350	4000	97,70%	5	20
Índice	350	1024	25%	10	20

<b>Índice</b>	350	2048	50%	10	20
<b>Índice</b>	350	4000	97,70%	10	20

**Tabla 4.1 Pruebas de funcionamiento**

**Fuente: Elaborado por el autor**

Acorde a la tabla 4.1 obtenida del desplazamiento en relación a la variación del pulso con una frecuencia de 2 Hz se puede tener mayor control del desplazamiento producido por los resortes de Nitinol, mientras que la variación del ancho de pulso permitió disminuir el tiempo de contracción del resorte de Nitinol permitiendo mayor velocidad al momento de generar movimientos, pero el incremento de velocidad en los resortes de Nitinol implica un mayor consumo de corriente e incremento de temperatura produciendo que el período de enfriamiento del Nitinol para regresar a la fase de Martensita sea mayor.

Entre los problemas presentados y que provocó el daño en los resortes de Nitinol fue la adaptación de una nueva memoria de forma no deseada en los resortes, debido a tensiones producido en el material inteligente mientras se aplicaba un ancho de pulso de 3000 a 4000, por tal motivo se optó por usar un ancho de pulso con un flanco máximo de 2500 con el fin de evitar daños en los resortes de Nitinol durante la transición de la fase Austenita a Martensita.

Acorde al propósito del presente proyecto se realizó pruebas de funcionamiento del prototipo de mano robótica para generar distintos movimientos, donde se alcanzó un porcentaje de similitud del 50%, valor esperado para la primera etapa o fase del prototipo de mano robótica fabricado con materiales inteligentes.

## **4.2 Presupuesto**

Como parte del desarrollo del proyecto se pone a consideración el presupuesto de la implementación del prototipo de mano robótica referente a materiales, costos de importación y el costo de mano de obra.



DESCRIPCIÓN	ESPECIFICACIÓN	UNIDADES	COSTO UNITARIO \$	VALOR TOTAL \$
Módulo Arduino Nano	ATmega 328	1	11	11
Módulo PWM	PCA 9685	1	20	20
Módulo Bluetooth	HC-05	1	13	10
Sensor de Fuerza	FSR402	1	11.16	11.16
Sensor de temperatura	LM35	1	3	3
Opto-acopladores	4N-25	10	0,35	0,32
Transistor MOSFET	IRFZ44N	10	1,4	14
Regulador de voltaje 5 V	LM7805	7	1,02	7,14
Regulador de voltaje variable	LM317	1	0,9	0,9
Regulador de corriente		1	4,85	4,85
Terminal	Ojo Rojo	15	0.30	4,5
Borneras	2 puntos	17	0,28	4,76
Conectores		1	0,5	0,5
Resistencia 330	330 ohm ¼ W	10	0,05	0,5
Resistencia 100	100 kohm ¼ W	10	0,05	0,5
Resistencia 10	10 kohm ¼ W	10	0,05	0,5
Resistencia 1	1 ohm 10 Watts	1	0,49	0,49
Resistencia de precisión	5 kohm	1	0,49	0,49
Capacitador	0,33 µF	2	0,1	0,2
Capacitador	0,1 µF	2	0,1	0,2
Placa PBC		2	8,5	17
Placa PBC		1	13,9	13,9
Impresión 3D	Estructura	1	217	217
Resortes de Nitinol		4	39,95	159,8
Batería Litio Lipo	10 Ah 3C	1	86,99	86,99
Cable multi-filar	3 metros	1	3	3
Cables protoboard		2	2,1	4,2
Mini multímetro	100 V - 10 A	1	13	13
Cargador	Cargador Batería LIPO IMAX	1	69	69
Varios			150	150

TOTAL	\$ 817,74
-------	-----------

**Tabla 4.2 Costos de materiales**  
Fuente: Elaborado por el autor

En la tabla 4.3 se hace mención a los costos de importación de los materiales inteligentes y la batería LIPO.

DESCRIPCIÓN	ESPECIFICACIÓN	UNIDADES	COSTO UNITARIO \$	VALOR TOTAL \$
Costo de Shipping EEUU	Importación materiales inteligentes	2	7	14
Costo de importación	Importación método 4x4	1	45	45
Pago de impuesto	Aduana	1	5	5
TOTAL				\$ 64,00

**Tabla 4.3 Costos de importación**  
Fuente: Elaborado por el autor

En la tabla 4.4 se hace mención a los costos de mano de obra que conforman el esfuerzo físico y mental que fueron requeridos para el desarrollo e implementación del proyecto.

DESCRIPCIÓN	ESPECIFICACIÓN	VALOR TOTAL \$
Diseño	Diseño estructural y electrónico	250
Implementación	Diseño estructural y electrónico	250
TOTAL		\$ 500,00

**Tabla 4.4 Costo de mano de obra**  
Fuente: Elaborado por el autor

Los costos presentados hacen referencia a la inversión requerida para realizar un prototipo de primera fase; para futuras mejoras o desarrollo de prototipos puede implicar la disminución del costo de fabricación.

### **4.3 Análisis de resultados**

Finalizado el proyecto para realizar un prototipo de mano robótica mediante materiales inteligentes, se ha realizado un análisis referente al funcionamiento del prototipo y los inconvenientes presentados, con el fin de proponer mejoras en el proyecto a futuro como son:

- Cambio estructural para disminución del tamaño del prototipo y pueda ser utilizado por una persona.
- Utilización de nuevas técnicas para disminución de las placas electrónicas con el fin de apoyar a la disminución del tamaño y peso en los nuevos proyectos.
- Utilización una combinación de materiales inteligentes del tipo alambre y resorte para mejorar el tipo de movimiento.

A continuación se hace mención a los aspectos no resueltos durante la realización del proyecto.

La fuente de alimentación que se encuentra conformado por la batería de LIPO y las etapas de regularización de voltaje se evidenció que los tamaños de los elementos al ser demasiados grandes provocan problemas en el tamaño total y peso en la estructura de la mano robótica provocando dificultad en la portabilidad del mismo, motivo por el cual se requiere el desarrollo de alternativas que permitan la disminución del espacio de la etapa de alimentación

Con el uso de los resortes de Nitinol se identificó que pueden generar grandes fuerzas pero delimitan al momento de generar movimientos más suaves o precisas, por el cual se debe realizar un segundo prototipo mediante la combinación del uso de resortes de Nitinol y alambres de Nitinol con el fin de generar nuevos movimientos más precisos.

En el momento que se activa el movimiento de cerrado donde se pretendía el movimiento simultaneo de los 5 dedos de la mano, se presentó problemas con la generación del movimiento producido por la señal PWM.

## CAPÍTULO V CONCLUSIONES Y RECOMENDACIONES

Finalmente este capítulo abarca la culminación del proyecto de titulación referente a la implementación de un prototipo de mano robótica con materiales inteligentes incorporando las conclusiones y recomendaciones realizado durante su elaboración.

### Conclusiones

- Se investigó los diferentes tipos de materiales inteligentes que permitieran generar movimientos al estar sometidos por una corriente eléctrica, entre los cuales estuvo el Nitinol que es una aleación con memoria de forma que pudo ser controlado mediante modulación de ancho de pulso y con ello facilitar el esquema electrónico que controlaría a las aleaciones de Nitinol.
- Se diseñó un sistema de control mediante el uso del módulo Arduino y circuitos electrónicos como opto-acopladores y transistores mosfet que permitió controlar los movimientos definidos para la mano, mediante el uso de señales PWM.
- Se implementó un prototipo de mano robótica mediante materiales inteligentes, módulos de comunicación inalámbrica, módulos micro-controladores y dispositivos electrónicos.
- Se logró establecer 5 tipos de movimientos en la mano robótica pero no con el mismo grado de similitud con una mano real debido a las limitaciones físicas de la estructura superior de la mano, motivo por el cual la mejora del diseño estructural y la combinación de otros tipos de materiales inteligentes podrían permitir alcanzar mayor definición en los movimientos.
- Se desarrolló un algoritmo computacional que permite comunicar un dispositivo móvil con los módulos Arduino mediante comunicación inalámbrica, permite validar

la flexibilidad que existe en los diferentes tipos de dispositivos electrónicos de diferentes fabricantes que se pueden integrar para conformar un prototipo de mano robótica, permitiendo la posibilidad de escalabilidad y fácil reemplazo de la tecnología usada para realizar mejoras continuas en el prototipo.

- Finalizado la implementación del prototipo de mano robótica, se realizó con éxito el proceso de pruebas con el prototipo y los algoritmos desarrollados, Durante este proceso se identificó las falencias en el movimiento de las articulaciones al no tener un método de refrigeración de las aleaciones de Nitinol, que permita un rápido enfriamiento y el material inteligente pueda tomar una nueva forma.
- Mediante el uso de sensores se pudo tener un control de las variaciones de temperatura que se producían internamente en las fases de alimentación como en la fase mecánica conformada por las aleaciones de Nitinol, esto fue fundamental para poder generar nuevos movimientos y no causar daños en los materiales inteligentes en el momento que se encuentran en la fase de Austenita por el incremento de temperatura.

## **Recomendaciones**

Finalmente las recomendaciones que se darán a continuación servirán para ampliar el campo aplicativo de nuestro proyecto, así como errores que deben evitarse:

- El proyecto que se implementó permitirá el desarrollo de múltiples aplicaciones tanto en la industria como en la robótica, permitiendo realizar mayores investigaciones del uso de materiales inteligentes en el campo tecnológico.
- Como proceso de mejoramiento del prototipo de mano robótica con materiales inteligentes, se recomienda el cambio del diseño de la estructura de la base del material inteligente con el fin de poder colocar más resortes en paralelo y aumentar la fuerza de cada articulación de la mano.

- El proceso de selección del calibre del material inteligente se encuentra ligado a la fuerza que se opone al movimiento en el momento de que la aleación recupere su forma en la fase de Austenita, por lo cual se tiene que considerar que la fuerza que se hace referencia está conformado como el peso mínimo de cada dedo y la fuerza de fricción entre las articulaciones, el cual hay que superar para generar movimientos, de igual manera se debe considerar que en el aumento del calibre del material seleccionado implicará un incremento de corriente proporcionado por la fuente, con dicha aclaración se puede recomendar el uso de más alambres de Nitinol en paralelo que permitan generar mayor fuerza de recuperación sin la necesidad de recurrir a un exceso de corriente suministrado por la fuente.
- Como se hizo mención en el presente documento, someter las aleaciones de memoria de forma a un sistema de refrigeración es indispensable para que las aleaciones puedan llegar a la fase de Martensita en un menor tiempo, pero se debe considerar que si el material inteligente aún se encuentra continuamente en la fase de Austenita, se aplique una tensión opuesta al movimiento de recuperación de forma y en ese momento se aplica el proceso de refrigeración puede provocar que el material inteligente adopte una nueva memoria de forma no deseado.
- Como alternativa en la creación del prototipo de la prótesis se elaboró la estructura mediante PVC y como se indica en el presente documento se colocó los resortes de Nitinol en la parte interna de la estructura, en este caso ambos elementos al encontrarse en contacto directo pueden provocar daños en la estructura por las temperaturas que se alcanzan, por lo que se recomienda usar disipadores en los extremos o puntos de sujeción del material inteligente con el fin de evitar daños en la estructura.

- En el caso de tener la necesidad que el material inteligente adopte nuevas formas se debe considerar que la temperatura al que se debe someter la aleación debe ser proporcional en todo el elemento con el fin de evitar puntos de ruptura o formas no deseadas.

## **Trabajos futuros**

La elaboración del prototipo de mano robótica con materiales inteligentes es la base para la realización de proyectos bajo la linealidad de la robótica y medicina. Como primera fase el prototipo permite generar movimientos básicos por medio del Nitinol, con el desarrollo de una segunda fase se puede reducir tamaño, peso e incremento de la fuerza. La implementación de sensores musculares mioeléctricos permitiría realizar movimientos voluntarios de la mano robótica controlados por el usuario sin la necesidad de usar dispositivos externos.



## REFERENCIAS BIBLIOGRÁFICAS

- Arduino. (5 de Noviembre de 2017). *Arduino*. Obtenido de ARDUINO MEGA 2560 REV3: <https://store.arduino.cc/usa/arduino-mega-2560-rev3>
- Arduino UTFSM RACSO. (21 de Mayo de 2014). *Arduino UTFSM*. Obtenido de Modulación por ancho de pulso (PWM): <http://www.arduino.utfsm.cl/modulacion-por-ancho-de-pulso-pwm/>
- ARMZTEC. (5 de Enero de 2018). *ARMZTEC*. Obtenido de Controlador Servomotores PWM 16CH PCA9685: <http://www.armztec.com/2016/06/controlador-servomotores-pwm-16ch.html>
- ASM International. (5 de Octubre de 2015). *ASM International*. Obtenido de Bionic hand based on bundled nitinol wires activated by electricity: [https://www.asminternational.org/web/smst/newswire/-/journal\\_content/56/10180/25825248/NEWS;jsessionid=06A1A076FFC51DAC5EA70D078FFBB01B?p\\_p\\_id=webcontentresults\\_WAR\\_webcontentsearchportlet\\_INSTANCE\\_Y38TZYxJGwq9&p\\_p\\_lifecycle=0&p\\_p\\_state=normal&p\\_p\\_mode=view](https://www.asminternational.org/web/smst/newswire/-/journal_content/56/10180/25825248/NEWS;jsessionid=06A1A076FFC51DAC5EA70D078FFBB01B?p_p_id=webcontentresults_WAR_webcontentsearchportlet_INSTANCE_Y38TZYxJGwq9&p_p_lifecycle=0&p_p_state=normal&p_p_mode=view) [3]
- Cano Sánchez , A. (2010). *ESTUDIO E IMPLEMENTACIÓN DE ACTUADORES*. MADRID: UNIVERSIDAD CARLOS III DE MADRID.
- Carlos, V. (12 de Octubre de 2016). *Introducción a la tecnología Bluetooth*. Obtenido de CCM: <http://es.ccm.net/contents/70-bluetooth>
- Cavadevices. (01 de Enero de 2018). *Cavadevices.com*. Obtenido de ¿Cómo calcular la capacidad necesaria de batería para almacenar energía ?: <http://www.cavadevices.com/archivos/FOLLETOS/calculo%20de%20bateria.pdf>

DYNALLOY. (26 de Enero de 2018). *DYNALLOY*. Obtenido de Technical Characteristics of Flexinol® Actuator Wires F1140Rev I.2: <http://www.dynalloy.com/pdfs/TCF1140.pdf>

DYNALLOY, Inc. (15 de Octubre de 2017). *DYNALLOY, Inc Makers of Dynamic Alloys*. Obtenido de DYNALLOY, Inc Makers of Dynamic Alloys: <http://www.dynalloy.com/nitinol.php>

DYNALLOY, INC. (7 de Enero de 2018). *DYNALLOY Makers of Dynamic Alloys*. Obtenido de Flexinol® Actuator Wire Technical and Design Data: [http://www.dynalloy.com/tech\\_data\\_wire.php](http://www.dynalloy.com/tech_data_wire.php)

DYNALLOY, INC.. (7 de Enero de 2018). *DYNALLOY*. Obtenido de FLEXINOL® Actuator Spring Technical and Design Data: [http://www.dynalloy.com/tech\\_data\\_springs.php](http://www.dynalloy.com/tech_data_springs.php)

ELECTRONICA60NORTE. (5 de Noviembre de 2017). *electronica60norte*. Obtenido de DATASHEET BLUETOOTH TO SERIAL PORT MODULE HC05: <http://www.electronica60norte.com/mwfls/pdf/newBluetooth.pdf>

Electronilab. (5 de Noviembre de 2017). *ELECTRONILAB*. Obtenido de Módulo Bluetooth HC-05 Serial Rs232: <https://electronilab.co/tienda/modulo-bluetooth-hc-05-serial-rs232/>

Electronilab. (2017). *Electronilab Ingeniería y Diseño Electrónico* . Obtenido de PCA9685 – Controlador 16 Canales PWM Servomotores – I2C.

Electronilab. (26 de Enero de 2018). *Electronilab.co*. Obtenido de Arduino Nano V3 – ATmega328 5V + Cable USB Compatible: <https://electronilab.co/tienda/arduino-nano-v3-atmega328-5v-cable-usb/>

Gómez Lopez, A. (2011). *Olimpia*. Obtenido de Nitinol, un biomaterial con memoria de forma:

[http://olimpia.cuautitlan2.unam.mx/pagina\\_ingenieria/mecanica/mat/mat\\_mec/m6/Nitinol\\_un%20biomaterial.pdf](http://olimpia.cuautitlan2.unam.mx/pagina_ingenieria/mecanica/mat/mat_mec/m6/Nitinol_un%20biomaterial.pdf)

Interlink Electronics. (11 de Noviembre de 2017). *Interlink Electronics*. Obtenido de FSR 400 Series Data Sheet:

[https://www.interlinkelectronics.com/datasheets/Datasheet\\_FSR.pdf](https://www.interlinkelectronics.com/datasheets/Datasheet_FSR.pdf)

Le Vinson Aguilar, L. (2012). *aidias*. Obtenido de Robótica y Prótesis Inteligentes: Una segunda:

[https://aidias.weebly.com/uploads/4/6/4/0/46406533/levinson\\_1\\_artdivulgacion.pdf](https://aidias.weebly.com/uploads/4/6/4/0/46406533/levinson_1_artdivulgacion.pdf)  
[2]

LLamas, L. (15 de Julio de 2015). *Luis Llamas Ingeniería, Informática y Diseño*. Obtenido de MEDIR TEMPERATURA CON ARDUINO Y SENSOR LM35:  
<https://www.luisllamas.es/medir-temperatura-con-arduino-y-sensor-lm35/>

Moreno Martínez, G. (2014). *Control de posición usando Shape Memory Alloy SMA*. Cartagena: Universidad Politécnica de Cartagena.

Mucslwires®. (15 de Octubre de 2017). *Actuator Wire Tech Data Sheet*. Obtenido de Muscle Wire: [http://store-musclewires-com.3dcartstores.com/Actuator-Wire-Tech-Data-Sheet\\_c\\_111.html](http://store-musclewires-com.3dcartstores.com/Actuator-Wire-Tech-Data-Sheet_c_111.html)

Naylamp Mechatronics. (6 de Diciembre de 2016). *Naylamp Mechatronics*. Obtenido de Tutorial Módulo Controlador de servos PCA9685 con Arduino:  
[http://www.naylampmechatronics.com/blog/41\\_Tutorial-M%C3%B3dulo-Controlador-de-servos-PCA9685.html](http://www.naylampmechatronics.com/blog/41_Tutorial-M%C3%B3dulo-Controlador-de-servos-PCA9685.html)

Naylamp Mechatronics SAC. (3 de Enero de 2018). *Naylamp Mechatronics SAC*. Obtenido de Configuración del módulo bluetooth HC-05 usando comandos AT:

[http://www.naylampmechatronics.com/blog/24\\_configuracion-del-modulo-bluetooth-hc-05-usa.html](http://www.naylampmechatronics.com/blog/24_configuracion-del-modulo-bluetooth-hc-05-usa.html)

Salinas Chávez, J. A. (1 de Julio de 2011). *eprints.uanl.mx*. Obtenido de CARACTERIZACIÓN DE ALEACIONES NiTi Y NiTiCu ELABORADAS POR:  
<http://eprints.uanl.mx/2800/1/1080221584.pdf>

Simons, E. (9 de Octubre de 2012). *microondasism.blogspot*. Obtenido de BANDA ISM:  
<http://microondasism.blogspot.com/2012/10/bandaism-lasbandas-ism-industrial.html>

Villoslada Peciña, Á. (2010). *Diseño y aplicación de un actuador SMA en*. Madrid: UNIVERSIDAD CARLOS III. [1]

Wikipedia. (15 de Octubre de 2017). *Shape-memory alloy*. Obtenido de Wikipedia:  
<https://differentSmoke.files.wordpress.com/2009/01/flexinol.pdf>

## **ANEXOS**

**ANEXO No. 1**

**CRONOGRAMA**

Id	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
1	<b>Definición del plan de proyecto</b>	<b>26 días</b>	<b>mar 11/4/17</b>	<b>mar 16/5/17</b>		
2	Investigar el tema de proyecto a realizar	6 días	mar 11/4/17	mar 18/4/17		Victor Laverde
3	Elaboración del documento del plan de proyecto	6 días	mié 19/4/17	mié 26/4/17	2	Victor Laverde
4	Rectificación y corrección del plan de proyecto	6 días	jue 27/4/17	jue 4/5/17	3	Victor Laverde
5	Elaboración del cronograma de tareas	6 días	vie 5/5/17	vie 12/5/17	4	Victor Laverde
6	Entrega y aprobación de plan	0 horas	lun 15/5/17	lun 15/5/17	5	Victor Laverde
7	<b>Establecer las tecnologías y condiciones de diseño</b>	<b>23 días</b>	<b>mar 16/5/17</b>	<b>jue 15/6/17</b>		
8	Investigar los dispositivos controladores para el proyecto	4 horas	mar 16/5/17	mar 16/5/17		Victor Laverde
9	Definir los dispositivos electronicos asociados al diseño	1 día	mar 16/5/17	mié 17/5/17	8	Victor Laverde
10	Definir la aplicación para el dispositivo móvil	4 horas	mié 17/5/17	mié 17/5/17	9	Victor Laverde
11	Adquisición de materiales y dispositivos electrónicos	21 días	jue 18/5/17	jue 15/6/17	10	Victor Laverde
12	<b>Pruebas de funcionamiento de los dispositivos</b>	<b>4 días</b>	<b>jue 15/6/17</b>	<b>mar 20/6/17</b>		
13	Acondicionamiento de forma de material inteligente de prueba	1 día	jue 15/6/17	jue 15/6/17		
14	Acondicionamiento de forma de 15 metros de material inteligente	3 días	vie 16/6/17	mar 20/6/17	13	Victor Laverde
15	<b>Diseño del sistema de control electrónico</b>	<b>1 día</b>	<b>mié 21/6/17</b>	<b>mié 21/6/17</b>		
16	Diseño del circuito electrónico para control del material inteligente	1 día	mié 21/6/17	mié 21/6/17		
17	Diseño del circuito regulador de voltaje 5V y 3.3 v	4 horas	mié 21/6/17	mié 21/6/17	16	Victor Laverde
18	Diseño del circuito acondicionador de señal de sensores	0,5 días	mié 21/6/17	mié 21/6/17	17	Victor Laverde
19	<b>Diseño de la aplicación de monitoreo y administración</b>	<b>5 días</b>	<b>mié 21/6/17</b>	<b>mar 27/6/17</b>		

abr '17  
3 | 10 | 17 | 24



Proyecto: CRONOGRAMA  
 Fecha: jue 1/2/18

Tarea		Resumen inactivo		Tareas externas	
División		Tarea manual		Hito externo	
Hito		solo duración		Fecha limite	
Resumen		Informe de resumen manual		Progreso	
Resumen del proyecto		Resumen manual		Progreso manual	
Tarea inactiva		solo el comienzo			
Hito inactivo		solo fin			

Id	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos	abr '17			
							3	10	17	24
20	Diseño de aplicación APP en Android para visualizar datos y control inalámbrico de la mano robotica	2 días	mié 21/6/17	jue 22/6/17						
21	Implementacion de aplicación APP en dispositivo android	3 días	vie 23/6/17	mar 27/6/17	20	Victor Laverde				
22	<b>Construcción de sistema de control electrónico</b>	<b>3 días</b>	<b>mié 28/6/17</b>	<b>vie 30/6/17</b>						
23	Creación de circuito impreso del control del material inteligente	3 días	mié 28/6/17	vie 30/6/17		Victor Laverde				
24	Creación del circuito impreso del regulador de voltaje de 5v y 3.3 v	3 días	mié 28/6/17	vie 30/6/17						
25	Creación del circuito impreso de acondicionador de señal de sensores	3 días	mié 28/6/17	vie 30/6/17						
26	Impresión 3D de estructura de mano	2 días	sáb 1/7/17	lun 3/7/17						
27	<b>Implementación del Sistema</b>	<b>23 días</b>	<b>mar 4/7/17</b>	<b>jue 3/8/17</b>						
28	Programacion de módulo Arduino Mega para generar movimientos definidos	15 días	mar 4/7/17	lun 24/7/17						
29	Programacion de módulo de bluetooth HC-05	5 horas	mar 25/7/17	mar 25/7/17	28	Victor Laverde				
30	Implementación del material inteligente en la estructura de mano	2 días	mié 26/7/17	jue 27/7/17	29	Victor Laverde				
31	Implementación del sistema de control y sensores en la estructura de la mano	4,38 días	vie 28/7/17	jue 3/8/17	30	Victor Laverde				
32	<b>Pruebas de funcionamiento y validación</b>	<b>6 días</b>	<b>vie 4/8/17</b>	<b>vie 11/8/17</b>						
33	Prueba de calibración del grado de movilidad de las articulaciones	3 días	vie 4/8/17	mar 8/8/17						
34	Pruebas del control inalámbrico de la prótesis	3 días	mié 9/8/17	vie 11/8/17	33	Victor Laverde				
35	<b>Reestructuración de prototipo de mano robotica version 2</b>	<b>30 días</b>	<b>lun 14/8/17</b>	<b>vie 22/9/17</b>		<b>Victor Laverde</b>				
40	<b>Reestructuración de prototipo de mano robotica version 3</b>	<b>45 días</b>	<b>sáb 23/9/17</b>	<b>jue 23/11/17</b>	35	<b>Victor Laverde</b>				

Proyecto: CRONOGRAMA  
 Fecha: jue 1/2/18

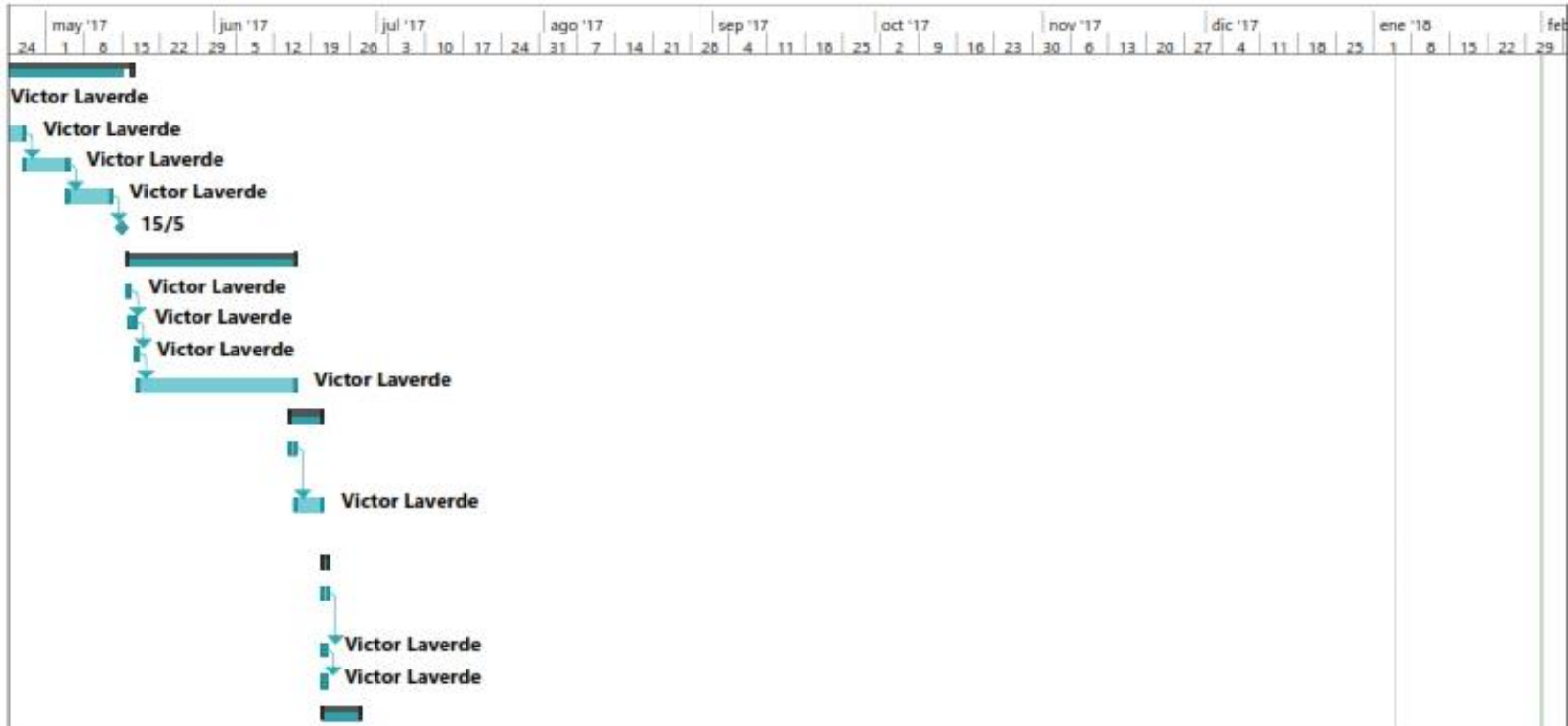
Tarea		Resumen inactivo		Tareas externas	
División		Tarea manual		Hito externo	
Hito		solo duración		Fecha limite	
Resumen		Informe de resumen manual		Progreso	
Resumen del proyecto		Resumen manual		Progreso manual	
Tarea inactiva		solo el comienzo			
Hito inactivo		solo fin			



Id	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos	abr '17	3	10	17	24
46	<b>Reestructuración de prototipo de mano robotica version 4</b>	<b>30 días</b>	<b>vie 24/11/17</b>	<b>jue 4/1/18</b>	<b>40</b>	<b>Victor Laverde</b>					
47	Diseño de estructura interna version final	15 días	vie 24/11/17	jue 14/12/17		Victor Laverde					
48	Impresión 3D de estructura interna de mano versión final	2 días	vie 15/12/17	lun 18/12/17	47	Victor Laverde					
49	Implementación de estructura interna	7 días	mar 19/12/17	mié 27/12/17	48	Victor Laverde					
50	Pruebas de funcionamiento	6 días	jue 28/12/17	jue 4/1/18	49	Victor Laverde					
51											

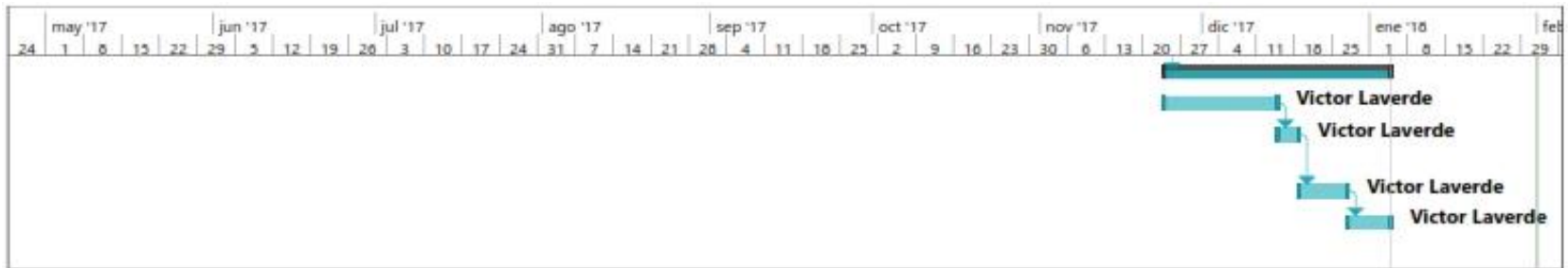
Proyecto: CRONOGRAMA  
 Fecha: jue 1/2/18

Tarea		Resumen inactivo		Tareas externas	
División		Tarea manual		Hito externo	
Hito		solo duración		Fecha limite	
Resumen		Informe de resumen manual		Progreso	
Resumen del proyecto		Resumen manual		Progreso manual	
Tarea inactiva		solo el comienzo			
Hito inactivo		solo fin			



Proyecto: CRONOGRAMA Fecha: jue 1/2/18	Tarea		Resumen inactivo		Tareas externas	
	División		Tarea manual		Hito externo	
	Hito		solo duración		Fecha limite	
	Resumen		Informe de resumen manual		Progreso	
	Resumen del proyecto		Resumen manual		Progreso manual	
	Tarea inactiva		solo el comienzo			
	Hito inactivo		solo fin			





Proyecto: CRONOGRAMA Fecha: jue 1/2/18	Tarea		Resumen inactivo		Tareas externas	
	División		Tarea manual		Hito externo	
	Hito		solo duración		Fecha limite	
	Resumen		Informe de resumen manual		Progreso	
	Resumen del proyecto		Resumen manual		Progreso manual	
	Tarea inactiva		solo el comienzo			
	Hito inactivo		solo fin			

## **ANEXO No. 2**

### **PROGRAMACIÓN DE MÓDULO ARDUINO NANO V3**

```

//UNIVERSIDAD TECNOLOGICA ISRAEL
//CARRERA DE ELECTRONICA Y TELECOMUNICACIONES
//PROYECTO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO
INGENIERIA EN ELECTRÓNICA Y TELECOMUNICACIONES
//VICTOR LAVERDE

#include <Wire.h> //LIBRERIA PARA
COMUNICACION I2C
#include <Adafruit_PWMServoDriver.h> //LIBRERIA PARA
GENERAR SEÑALES PWM
Adafruit_PWMServoDriver pwm = Adafruit_PWMServoDriver(0x40); //DEFINE LA
VARIABLE pwm PARA MANEJO DE LA SEÑAL EN LA DIRECCION 0X40 DEL
MODULO PCA9685

const int sensorPin = A0; //DEFINE AL PUERTO
ANALOGICO A0 EN LA VARIABLE sensorPin DEL SENSOR DE
TEMPERATURA LM35
const int fsrPin = A1; //DEFINE AL PUERTO
ANALOGICO A1 EN LA VARIABLE fsrPin DEL SENSOR DE FUERZA FSR

unsigned char manual; //SE DEFINE VARIABLES A
USAR EN EL PROGRAMA
int fuerza=0;
int temp=0;
int n=0;
int i;
int duty;
int movimiento=0;
int j=0;
int fsrReading;
int fsrVoltage;
unsigned long fsrResistance;
unsigned long fsrConductance;
unsigned long fsrForce;
int temp_medido=0;
float temp_voltaje=0;
float temp_centigrados=0;

//FIN DE DECLARACION DE VARIABLES

void setup() {

pinMode(fsrPin, INPUT);
pinMode(sensorPin, INPUT);
Serial.begin(9600); //SE ESTABLECE LA
COMUNICACION SERIAL 1 EN 9600 BAUDIOS
//SE ESTABLECE LA COMUNICACION SERIAL 2 EN
9600 BAUDIOS

```

```

pwm.begin(); //SE INICIA LA COMUNICACION CON
MODULO PCA9685
pwm.setPWMFreq(3); //SE CONFIGURA LA
FRECUENCIA DE LA SEÑAL PWM EN 2 HZ
}
void loop() { //CUERPO DEL PROGRAMA
temp = temperatura(); //SE LLAMA A LAS FUNCIONES
temperatura()
fuerza = fsr(); //SE LLAMA A LA FUNCION fsr()
Serial.print(temp); //SE ENVIA LA INFORMACION
ALMACENADO EN TEMP POR LA COMUNICACION SERIAL
Serial.print(","); //SE ENVIA EL SEPARADOR DE
DATOS "," POR COMUNICACION SERIAL
Serial.print(fuerza); //SE ENVIA LA INFORMACION
ALMACENADO EN FUERZA POR LA COMUNICACION SERIAL
Serial.print(","); //SE ENVIA EL SEPARADOR DE DATOS "," POR
COMUNICACION SERIAL
delay(5000);

if(Serial.available()>0){ //VERIFICA SI EN EL BUFFER SE
RECIBE ALGUNA INFORMACION POR COMUNICACION SERIAL
movimiento=Serial.read(); //SI SE TIENE UN INFORMACION
EN EL BUFFER SE ALMACENA EN LA VARIABLE MOVIMIENTO
}

switch(movimiento){ //DE ACUERDO AL VALOR
ALMACENADO EN LA VARIABLE MOVIMIENTO SE SELECCIONA EL TIPO DE
MOVIMIENTO
case 1: //MOVIMIENTO LIBRE
movimiento1();
break;
case 2: //MOVIMIENTO CERRAR
movimiento2();
break;
case 3: //MOVIMIENTO GANCHO
movimiento3();
break;
case 4: //MOVIMIENTO CIRCULAR
movimiento4();
break;
case 5: //MOVIMIENTO LATERAL
movimiento5();
break;
case 6: //SALIR DE SELECCION DE MOVIMIENTO
default:
break;
}

```

```
}  
}
```

```
float temperatura(){ //FUNCION PARA MEDIR LA  
TEMPERATURA DEL SENSOR LM35  
temp_medido = analogRead(sensorPin); //FORMULA PARA  
ACONDICIONAR SEÑAL  
temp_voltaje = (temp_medido / 1023.0 )*5000;  
temp_centigrados = temp_voltaje / 10;  
return temp_centigrados; //EL RESULTADO SE  
ALMACENA EN LA VARIABLE TEMP_CENTIGRADOS Y SE RETURNO EL  
VALOR EN EL LLAMAD DE FUNCION  
}  
float fsr() { //FUNCION PARA MEDIR LA FUERZA  
DEL SENSOR FSR402  
fsrReading = analogRead (fsrPin);  
fsrVoltage = map (fsrReading, 0, 1023, 0, 5000); //SE UTILIZA LA  
FUNCION MAP PARA CONVERTIR EL NUMERO DE MUESTRAS MEDIDAS EN  
UN VALOR DE 5000 MILIVOLTIOS  
fsrResistance = 5000 - fsrVoltage; //DETERMINA LA  
RESISTENCIA DEL SENSOR DE FUERZA A PARTIR DEL VALOR MEDIDO EN  
VOLTIOS  
fsrResistance *= 10000;  
fsrResistance /= fsrVoltage;  
fsrConductance = 1000000; //DETERMINA LA  
CONDUCTANCIA DEL SENSOR DE FUERZA A PARTIR DEL VALOR MEDIDO  
fsrConductance /= fsrResistance;  
//SE UTILIZA CMO REFERENCIA LAS  
GRAFICAS DEL FSR PARA APROXIMAR LA FUERZA EN NEWTONS  
if(fsrConductance <= 1000) {  
fsrForce = fsrConductance / 80;  
}  
else{  
fsrForce = fsrConductance - 1000;  
fsrForce /= 30;  
}  
return fsrForce; //EL RESULTADO SE ALMACENA EN  
LA VARIABLE TEMP_CENTIGRADOS Y SE RETURNO EL VALOR EN EL  
LLAMAD DE FUNCION  
}
```

```
void movimiento1(){ //FUNCION DE MOVIMIENTO 1  
while(movimiento==1){ //SE CREA UN BUCLE INFINITO  
MIENTRAS LA VARIABLE MOVIMIENTO SEA IGUAL A 1  
while(Serial.available())>0){ //SE ESCUCHA EL BUFFER DE  
ENTRADA DE LA COMUNICACION SERIAL POR UNA INSTRUCCION
```



```

    manual=Serial.read();                //SE ALMACENA LOS DATOS DEL
BUFFER EN LA VARIABLE MANUAL
}
switch (manual){                        //SE DEFINE A QUE INSTRUCCION
PERTENECE ACORDE AL DATO RECIBIDO
    case 'A':                            //SI SE RECIBE LA INSTRUCCION "A"
NO REALIZA ACCION
        break;

    case 'B':                            //SI SE RECIBE LA INSTRUCCION "B"
SE ASIGNA EL VALOR 0 A LA VARIABLE DUTY
        duty=0;
        pwm.setPWM(9,0,duty);            //SE ENVIA POR
COMUNICACION I2C AL MODULO PCA9685 EL ANCHO DE PULSO DUTY EN
CADA UNO DE LOS CANALES (0 - 9)
        pwm.setPWM(8,0,duty);
        pwm.setPWM(7,0,duty);
        pwm.setPWM(6,0,duty);
        pwm.setPWM(5,0,duty);
        pwm.setPWM(4,0,duty);
        pwm.setPWM(3,0,duty);
        pwm.setPWM(2,0,duty);
        pwm.setPWM(1,0,duty);
        pwm.setPWM(0,0,duty);
        movimiento=0;                    //SE DEVINE NUEVO VALOR A LA
VARIABLE MOVIMIENTO PARA SALIR DE BUCLE INFINITO MOVIMIENTO1
        manual=0;                        //SE ENCERA LA VARIABLE MANUAL
        break;

    case 'C': //DEDO INDICE UP           //SI SE RECIBE LA
INSTRUCCION "C" SE ASIGNA EL ANCHO DE PULSO 2700 A LA VARIABLE
DUTY
        duty=2700;
        pwm.setPWM(8,0,duty);            //SE ENVIA POR COMUNICACION
I2C AL MODULO PCA9685 EL ANCHO DE PULSO DUTY EN EL CANAL 8
PARA GENERAR EL MOVIMIENTO CERRADO
        delay(4000);                     //SE ESPERA 4 SEGUNDOS MIENTRAS
SE GENERA LA SEÑAL PWM
        duty=0;                          //SE ENCERA EL ANCHO DE PULSO A 0
        pwm.setPWM(8,0,duty);            //SE ENVIA POR COMUNICACION
I2C EL ANCHO DE PULSO 0 PARA DETENER LA SEÑAL PWM
        manual=0;                        //SE ENCERA LA VARIABLE MANUAL
        break;

    case 'D': //DEDO INDICE DOWN        //SE GENERA EL
MOVIMIENTO DE ABIERTO DEL DEDO INDICE

```

```
duty=2700;
pwm.setPWM(9,0,duty);
delay(4000);
duty=0;
pwm.setPWM(9,0,duty);
manual=0;
break;
```

```
case 'E': //DEDO MEDIO DOWN //SE GENERA EL
MOVIMIENTO CERRADO DEL DEDO MEDIO
```

```
duty=2700;
pwm.setPWM(6,0,duty);
delay(4000);
duty=0;
pwm.setPWM(6,0,duty);
manual=0;
break;
```

```
case 'F': //DEDO MEDIO UP //SE GENERA EL MOVIMIENTO
ABIERTO DEL DEDO MEDIO
```

```
duty=2700;
pwm.setPWM(7,0,duty);
delay(4000);
duty=0;
pwm.setPWM(7,0,duty);
manual=0;
break;
```

```
case 'G': //DEDO ANULAR DOWN //SE GENERA EL
MOVIMIENTO CERRADO DEL DEDO ANULAR
```

```
duty=2700;
pwm.setPWM(4,0,duty);
delay(4000);
duty=0;
pwm.setPWM(4,0,duty);
manual=0;
break;
```

```
case 'H': //DEDO ANULAR UP //SE GENERA EL MOVIMIENTO
ABIERTO DEL DEDO ANULAR
```

```
duty=2700;
pwm.setPWM(5,0,duty);
delay(4000);
duty=0;
pwm.setPWM(5,0,duty);
manual=0;
```

```

break;

    case 'I': //DEDO MEÑIQUE DOWN                //SE GENERA EL MOVIMIENTO
CERRADO DEL DEDO MEÑIQUE
    duty=2700;
    pwm.setPWM(2,0,duty);
    delay(4000);
    duty=0;
    pwm.setPWM(2,0,duty);
    manual=0;
    break;

    case 'J': //DEDO MEÑIQUE UP                //SE GENERA EL MOVIMIENTO
ABIERTO DEL DEDO MEÑIQUE
    duty=2700;
    pwm.setPWM(3,0,duty);
    delay(4000);
    duty=0;
    pwm.setPWM(3,0,duty);
    manual=0;
    break;

    case 'K': //DEDO PULGAR UP                //SE GENERA EL MOVIMIENTO
CERRADO DEL PULGAR
    duty=2500;
    pwm.setPWM(0,0,duty);
    delay(2000);
    duty=0;
    pwm.setPWM(0,0,duty);
    manual=0;
    break;

    case 'L': //DEDO PULGAR DOWN                //SE GENERA EL
MOVIMIENTO ABIERTO DEL PULGAR
    duty=2500;
    pwm.setPWM(1,0,duty);
    delay(2000);
    duty=0;
    pwm.setPWM(1,0,duty);
    manual=0;
    break;
default:
    break;
}
}}
```

```

void movimiento2(){          //MOIMIENTO MANO CERRADA
while(movimiento==2){
while(Serial.available()>0){
  manual=Serial.read();
}
switch (manual){
case 'A':
  duty=2700;
  pwm.setPWM(9,0,duty); //DEDO INDICE
  delay(4000);
  duty=0;
  pwm.setPWM(9,0,duty); //DEDO INDICE
  duty=2700;
  pwm.setPWM(7,0,duty); //DEDO MEDIO
  delay(4000);
  duty=0;
  pwm.setPWM(7,0,duty); //DEDO MEDIO
  duty=2700;
  pwm.setPWM(5,0,duty); //DEDO ANULAR
  delay(4000);
  duty=0;
  pwm.setPWM(5,0,duty); //DEDO ANULAR
  duty=2700;
  pwm.setPWM(3,0,duty); //DEDO MEÑIQUE
  delay(4000);
  duty=0;
  pwm.setPWM(3,0,duty); //DEDO MEÑIQUE
  duty=2500;
  pwm.setPWM(1,0,duty); //DEDO PULGAR
  delay(1000);
  duty=0;
  pwm.setPWM(1,0,duty); //DEDO PULGAR

  manual=0;
  break;

case 'B':

  duty=2500;
  pwm.setPWM(0,0,duty); //DEDO PULGAR
  delay(1500);
  duty=0;
  pwm.setPWM(0,0,duty); //DEDO PULGAR

  duty=2700;

```

```

pwm.setPWM(8,0,duty); //DEDO INDICE
delay(4000);
duty=0;
pwm.setPWM(8,0,duty); //DEDO INDICE
duty=2700;
pwm.setPWM(6,0,duty); //DEDO MEDIO
delay(4000);
duty=0;
pwm.setPWM(6,0,duty); //DEDO MEDIO
duty=2700;
pwm.setPWM(4,0,duty); //DEDO ANULAR
delay(4000);
duty=0;
pwm.setPWM(4,0,duty); //DEDO ANULAR
duty=2700;
pwm.setPWM(2,0,duty); //DEDO MEÑIQUE
delay(4000);
duty=0;
pwm.setPWM(2,0,duty); //DEDO MEÑIQUE
movimiento=0;
manual=0;
break;
}
}
}

void movimiento3() { //MOVIMIENTO GANCHO
while(movimiento==3){
while(Serial.available()>0){
manual=Serial.read(); }

switch (manual){
case 'A':

duty=2700;
pwm.setPWM(9,0,duty); //DEDO INDICE
delay(1500);
duty=0;
pwm.setPWM(9,0,duty); //DEDO INDICE
duty=2700;
pwm.setPWM(7,0,duty); //DEDO MEDIO
delay(1500);
duty=0;
pwm.setPWM(7,0,duty); //DEDO MEDIO
duty=2700;
pwm.setPWM(5,0,duty); //DEDO ANULAR

```

```
delay(1500);
duty=0;
pwm.setPWM(5,0,duty);//DEDO ANULAR
duty=2700;
pwm.setPWM(3,0,duty);//DEDO MEÑIQUE
delay(1500);
duty=0;
pwm.setPWM(3,0,duty);//DEDO MEÑIQUE
manual=0;
break;
```

```
case 'B':
```

```
duty=2700;
pwm.setPWM(8,0,duty); //DEDO INDICE
delay(1500);
duty=0;
pwm.setPWM(8,0,duty); //DEDO INDICE
duty=2700;
pwm.setPWM(6,0,duty);//DEDO MEDIO
delay(1500);
duty=0;
pwm.setPWM(6,0,duty);//DEDO MEDIO
duty=2700;
pwm.setPWM(4,0,duty);//DEDO ANULAR
delay(1500);
duty=0;
pwm.setPWM(4,0,duty);//DEDO ANULAR
duty=2700;
pwm.setPWM(2,0,duty);//DEDO MEÑIQUE
delay(1500);
duty=0;
pwm.setPWM(2,0,duty);//DEDO MEÑIQUE
movimiento=0;
manual=0;
break;
```

```
}
}}
```

```
void movimiento4(){ //MOVIMIENTO CIRCULAR
while(movimiento==4){
while(Serial.available()>0){
manual=Serial.read();
}
switch (manual){
case 'A':
```

```
duty=2700;
pwm.setPWM(7,0,duty);//DEDO MEDIO
delay(4000);
duty=0;
pwm.setPWM(7,0,duty);//DEDO MEDIO
```

```
duty=2700;
pwm.setPWM(5,0,duty);//DEDO ANULAR
delay(4000);
duty=0;
pwm.setPWM(5,0,duty);//DEDO ANULAR
```

```
duty=2700;
pwm.setPWM(9,0,duty); //DEDO INDICE
delay(4000);
duty=0;
pwm.setPWM(9,0,duty); //DEDO INDICE
```

```
duty=2500;
pwm.setPWM(1,0,duty);//DEDO PULGAR
delay(1000);
duty=0;
pwm.setPWM(1,0,duty); //DEDO PULGAR
```

```
duty=2700;
pwm.setPWM(3,0,duty);//DEDO MEÑIQUE
delay(4000);
duty=0;
pwm.setPWM(3,0,duty);//DEDO MEÑIQUE
```

```
manual=0;
break;
```

```
case 'B':
```

```
duty=2500;
pwm.setPWM(0,0,duty);//DEDO PULGAR
delay(2000);
duty=0;
pwm.setPWM(0,0,duty); //DEDO PULGAR
```

```
duty=2700;
pwm.setPWM(2,0,duty);//DEDO MEÑIQUE
delay(4000);
duty=0;
```

```

pwm.setPWM(2,0,duty);//DEDO MEÑIQUE

duty=2700;
pwm.setPWM(6,0,duty);//DEDO MEDIO
delay(4000);
duty=0;
pwm.setPWM(6,0,duty);//DEDO MEDIO

duty=2700;
pwm.setPWM(4,0,duty);//DEDO ANULAR
delay(4000);
duty=0;
pwm.setPWM(4,0,duty);//DEDO ANULAR

duty=2700;
pwm.setPWM(8,0,duty); //DEDO INDICE
delay(4000);
duty=0;
pwm.setPWM(8,0,duty); //DEDO INDICE

movimiento=0;
manual=0;
break;
default:
break;
}
}
}
void movimiento5(){

while(movimiento==5){
  while(Serial.available()>0){
    manual=Serial.read();
  }
switch (manual){
case 'A':

  duty=2700;
  pwm.setPWM(9,0,duty); //DEDO INDICE
  delay(2500);
  duty=0;
  pwm.setPWM(9,0,duty);//DEDO INDICE
  pwm.setPWM(1,0,duty);//DEDO PULGAR
  delay(1500);
  duty=0;

```



```
pwm.setPWM(1,0,duty);//DEDO PULGAR

manual=0;
break;

case 'B':

    duty=2700;
    pwm.setPWM(8,0,duty); //DEDO INDICE
    delay(2500);
    duty=0;
    pwm.setPWM(8,0,duty);//DEDO INDICE
    pwm.setPWM(0,0,duty);//DEDO PULGAR
    delay(1500);
    duty=0;
    pwm.setPWM(0,0,duty);//DEDO PULGAR

    movimiento=0;
    manual=0;
    break;
    default:
    break;

}}}
```

## **ANEXO No. 3**

# **PROGRAMACIÓN DE CONFIGURACIÓN DE MÓDULO BLUETOOTH HC-05**

```

//UNIVERSIDAD TECNOLOGICA ISRAEL
//CARRERA DE ELECTRONICA Y TELECOMUNICACIONES
//PROYECTO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO
INGENIERIA EN ELECTRÓNICA Y TELECOMUNICACIONES
//VICTOR LAVERDE
void setup() {
Serial.begin(9600);           //SE INICIA LA COMUNICACION SERIAL 1 A
UNA VELOCIDAD DE 9600 BAUDIOS
Serial.println("Ingresar Comandos AT"); // SE ENVIA EL MENSAJE DE
INGRESAR COMANDOS AT POR EL PUERTO SERIAL 1
Serial1.begin(38400);        //SE INICI LA COMUNICACION SERIAL 2 A
UNA VELOCIDAD DE 38400 BAUDIOS
}
void loop() {                // CUERPO DEL PROGRAMA
if (Serial1.available()){    //VERIFICA QUE LA COMUNICACION SERIAL 2
ESTE HABILITADA
Serial.write(Serial1.read()); } //ENVIA LA INFORMACION OBTENIDA DE LA
COMUNICACION SERIAL 2 POR EL SERIAL 1
if (Serial.available()) {    //VERIFICA QUE LA COMUNICACION SERIAL 1
ESTE HABILITADA
Serial1.write(Serial.read()); //ENVIA LA INFORMACION OBTENIDA DE LA
COMUNICACION SERIAL 1 A TRAVES DE LA COMUNICACION DEL SERIAL 2
}}                            //FIN DE PROGRAMA

```

## **ANEXO No. 4**

# **PROGRAMACIÓN DE APLICACIÓN EN APP INVENTOR**

```
when ListPicker1 . BeforePicking
do set ListPicker1 . Elements to BluetoothClient1 . AddressesAndNames
```

```
when ListPicker1 . AfterPicking
do set Label3 . Text to ListPicker1 . Selection
   set ListPicker1 . Selection to call BluetoothClient1 . Connect
                                   address ListPicker1 . Selection
```

```
when Clock1 . Timer
do if BluetoothClient1 . IsConnected
   then set Label3 . Text to "CONECTADO"
       if call BluetoothClient1 . BytesAvailableToReceive > 0
           then set global DATOS_DE_LLEGADA to call BluetoothClient1 . ReceiveText
                                                       numberOfBytes call BluetoothClient1 . BytesAvailableToReceive
                set global LISTA to split text get global DATOS_DE_LLEGADA
                                       at " "
                set Label6 . Text to select list item list get global LISTA
                                       index 1
                set Label7 . Text to select list item list get global LISTA
                                       index 2
           else set Label3 . Text to "DESCONECTADO"
```

```
initialize global DATOS_DE_LLEGADA to ""
```

```
initialize global LISTA to create empty list
```

```
when Clock2 . Timer
do set Clock2 . TimerInterval to Clock2 . TimerInterval
   set global DATOS_DE_LLEGADA to ""
   set global LISTA to create empty list
```

```
when ListPicker2 .AfterPicking
do
  set Label8 .Text to ListPicker2 .Selection
  if ListPicker2 .Selection = "LIBRE" and BluetoothClient1 .Available
  then call BluetoothClient1 .Send1ByteNumber
    number round 1
  else if ListPicker2 .Selection = "CERRAR" and BluetoothClient1 .Available
  then call BluetoothClient1 .Send1ByteNumber
    number round 2
  else if ListPicker2 .Selection = "GANCHO" and BluetoothClient1 .Available
  then call BluetoothClient1 .Send1ByteNumber
    number round 3
  else if ListPicker2 .Selection = "CIRCULAR" and BluetoothClient1 .Available
  then call BluetoothClient1 .Send1ByteNumber
    number round 4
  else if ListPicker2 .Selection = "LATERAL" and BluetoothClient1 .Available
  then call BluetoothClient1 .Send1ByteNumber
    number round 5
  else if ListPicker2 .Selection = "DETENIDO" and BluetoothClient1 .Available
  then set Label8 .Text to " "
    call BluetoothClient1 .Send1ByteNumber
      number round 0
```

```
when Button1 .Click
do
  if BluetoothClient1 .Available
  then call BluetoothClient1 .SendText
    text " A "
  if Button1 .TextColor = blue
  then set Button1 .TextColor to red
  else set Button1 .TextColor to blue
```

```
when Button2 .Click
do
  if BluetoothClient1 . Available
  then
    call BluetoothClient1 .SendText
      text " B "
    set Button1 . TextColor to blue
    set Button4 . TextColor to blue
    set Button5 . TextColor to blue
    set Button6 . TextColor to blue
    set Button7 . TextColor to blue
    set Button8 . TextColor to blue
```

```
when Button4 .Click
do
  if ListPicker2 . Selection = " LIBRE " and BluetoothClient1 . Available
  then
    call BluetoothClient1 .SendText
      text " C "
    set Button5 . Enabled to false
    set Button4 . TextColor to red
```

```
when Button5 .Click
do
  if ListPicker2 . Selection = " LIBRE " and BluetoothClient1 . Available
  then
    call BluetoothClient1 .SendText
      text " D "
    set Button4 . Enabled to false
    set Button5 . TextColor to red
```

```
when Button6 .Click
do
  if ListPicker2 . Selection = " LIBRE " and BluetoothClient1 . Available
  then
    call BluetoothClient1 .SendText
      text " E "
    set Button7 . Enabled to false
    set Button6 . TextColor to red
```

```
when Button7 .Click
do
  if ListPicker2 . Selection = "LIBRE" and BluetoothClient1 . Available
  then
    call BluetoothClient1 . SendText
      text " F "
    set Button6 . Enabled to false
    set Button7 . TextColor to red
```

```
when Button8 .Click
do
  if ListPicker2 . Selection = "LIBRE" and BluetoothClient1 . Available
  then
    call BluetoothClient1 . SendText
      text " G "
    set Button9 . Enabled to false
    set Button8 . TextColor to red
```

```
when Button9 .Click
do
  if ListPicker2 . Selection = "LIBRE" and BluetoothClient1 . Available
  then
    call BluetoothClient1 . SendText
      text " H "
    set Button8 . Enabled to false
    set Button9 . TextColor to red
```

```
when Button10 .Click
do
  if ListPicker2 . Selection = "LIBRE" and BluetoothClient1 . Available
  then
    call BluetoothClient1 . SendText
      text " I "
    set Button11 . Enabled to false
    set Button10 . TextColor to red
```



```
when Button11 .Click
do
  if ListPicker2 . Selection = "LIBRE" and BluetoothClient1 . Available
  then
    call BluetoothClient1 . SendText
      text "J"
    set Button10 . Enabled to false
    set Button11 . TextColor to red
```

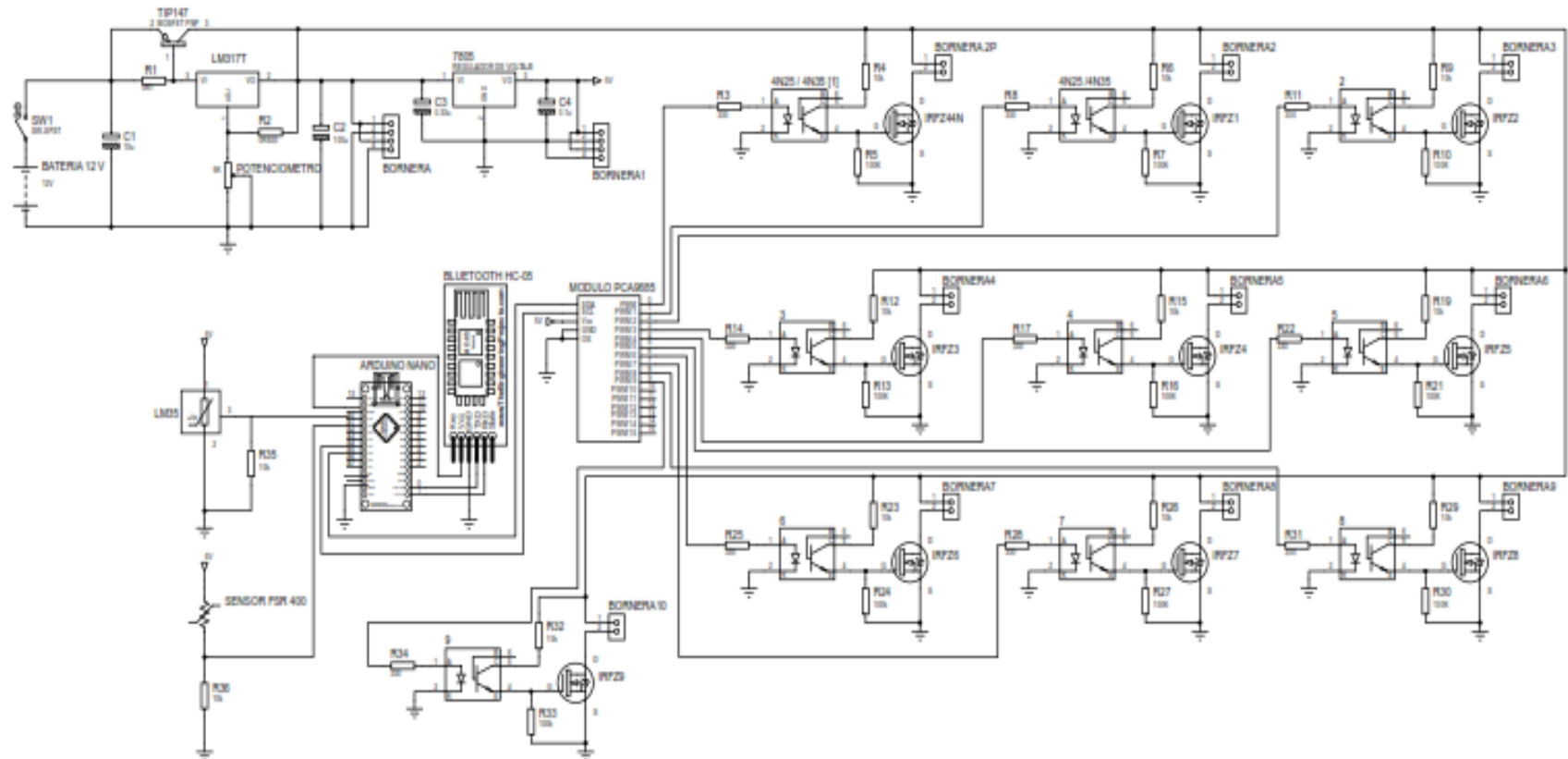
```
when Button12 .Click
do
  if ListPicker2 . Selection = "LIBRE" and BluetoothClient1 . Available
  then
    call BluetoothClient1 . SendText
      text "K"
    set Button13 . Enabled to false
    set Button12 . TextColor to red
```

```
when Button13 .Click
do
  if ListPicker2 . Selection = "LIBRE" and BluetoothClient1 . Available
  then
    call BluetoothClient1 . SendText
      text "L"
    set Button12 . Enabled to false
    set Button13 . TextColor to red
```

```
when Clock3 .Timer
do
  set Button4 . Enabled to true
  set Button4 . TextColor to blue
  set Button5 . Enabled to true
  set Button5 . TextColor to blue
  set Button6 . Enabled to true
  set Button6 . TextColor to blue
  set Button7 . Enabled to true
  set Button7 . TextColor to blue
  set Button8 . Enabled to true
  set Button8 . TextColor to blue
  set Button9 . Enabled to true
  set Button9 . TextColor to blue
  set Button10 . Enabled to true
  set Button10 . TextColor to blue
  set Button11 . Enabled to true
  set Button11 . TextColor to blue
  set Button12 . Enabled to true
  set Button12 . TextColor to blue
  set Button13 . Enabled to true
  set Button13 . TextColor to blue
```

**ANEXO No. 5**

**DIAGRAMA CIRCUITAL**



- |                                |  |                            |
|--------------------------------|--|----------------------------|
| 1. MÓDULO ARDUINO NANO V3 [5V] | 7. RESISTENCIA VARIABLE DE PRECISION [5KOHM] | 13. RESISTENCIA [500 OHM]  |
| 2. BLUETOOTH HC-05 [3.3V]      | 8. RESISTENCIA CERAMICO [1 OHM - 10W]        | 14. RESISTENCIA [330 OHM]  |
| 3. MÓDULO PCA9685 [3.3V]       | 9. CAPACITOR ELECTROLITICO [10 uF - 25V]     | 15. RESISTENCIA [10 KOHM]  |
| 4. MOSFET DARLINGTON TIP147    | 10. CAPACITOR ELECTROLITICO [100 uF - 25V]   | 16. RESISTENCIA [100 KOHM] |
| 5. REGULADOR DE VOLTAJE LM7805 | 11. CAPACITOR ELECTROLITICO [0.1 uF - 25V]   | 17. MOSFET [IRF244N]       |
| 6. REGULADOR DE VOLTAJE LM7805 | 12. CARACITOR ELECTROLITICO [0.33 uF - 25V]  | 18. OPTOCOPLADOR 4N25      |

TITULO: CIRCUITO DEL SISTEMA DE CONTROL DE PROTESIS ROBOTICA CON MATERIALES INTELIGENTES

ESC	NOMBRE DEL DOCUMENTO: CIRCUITO DE CONTROL
FECHA: Mar 25 de Julio, 2017	AUTOR: Victor Laverde
Hoja 1 de 1	

**ANEXO No. 6**

**MANUAL DE USUARIO**

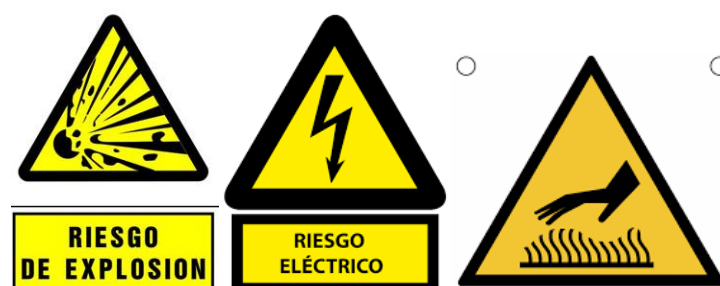
# MANUAL DE USUARIO

AUTOR VICTOR LAVERDE

## Contenido

CONSEJOS DE SEGURIDAD .....	3
Batería .....	3
Resortes de Nitinol.....	3
Dispositivos electrónicos.....	3
CONTENIDO DEL PROYECTO .....	4
CONEXIONES BÁSICAS.....	5
MANEJO DE APLICACIÓN MÓVIL .....	6
Requerimientos técnicos para el uso de aplicación móvil .....	6
Instalación de aplicación móvil .....	6
Antes de iniciar la aplicación.....	6
Uso de aplicación.....	8
TIPOS DE MOVIMIENTOS.....	10
Movimiento Libre:.....	10
Movimiento Cerrado.....	10
Movimiento Gancho .....	10
Movimiento Circular .....	10
Movimiento Lateral.....	11
SOPORTE TÉCNICO .....	11

## CONSEJOS DE SEGURIDAD



El proyecto de mano robótica con materiales inteligentes contiene elementos que pueden causar corto circuito, incendio, riesgo de explosión o quemaduras. Se exige al usuario leer el presente manual antes de iniciar su manipulación.

### Batería

La fuente de alimentación es una batería LIPO de 11 voltios 10 Ah, tome las precauciones necesarias para que NUNCA se junten los terminales positivo y negativo de la batería, se puede producir un cortocircuito que puede provocar que la batería se incendie o explote.

En caso de hinchazón o derrame de líquido de la batería; desconecte la batería del prototipo y aléjese una distancia prudente por posible explosión o incendio de la batería.

### Resortes de Nitinol

Los resortes de Nitinol ubicados en la parte interna de la estructura del proyecto, pueden alcanzar temperaturas en rangos de 55°C – 85°C. NO manipule el material inteligente durante y después de energizar el circuito, la exposición directa puede causar quemaduras.

### Dispositivos electrónicos

En la estructura interna de la prótesis de mano se encuentra dispositivos electrónicos que pueden sufrir daños a la exposición de líquidos, evite derramar líquidos sobre el prototipo.



## CONTENIDO DEL PROYECTO



Prototipo de mano



Conector Banana  
- Terminal T



Cable USB Macho - Macho



Batería LIPO 11 V  
10000 mA



Aplicación Móvil

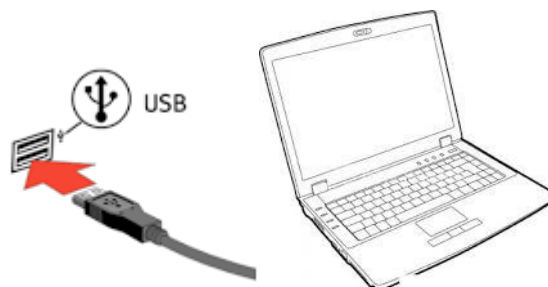


Cargador Balanceador para  
Batería LIPO

## CONEXIONES BÁSICAS



Conectar la batería LIPO al prototipo mediante el cable Banana – T, en los terminales designados según la polaridad de positivo (rojo) y negativo (negro).



Para cambios de configuración conectar el prototipo a un ordenador mediante el cable USB Macho – Macho en el terminal USB Hembra.

# MANEJO DE APLICACIÓN MÓVIL

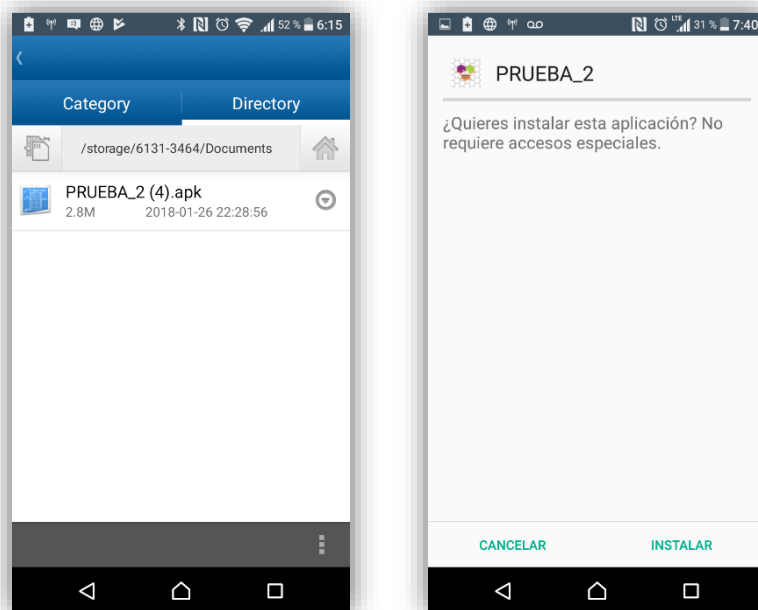
## Requerimientos técnicos para el uso de aplicación móvil

- Dispositivo móvil con sistema operativo Android mayor a la versión 2.
- Comunicación *Bluetooth*
- Memoria necesaria:

## Instalación de aplicación móvil

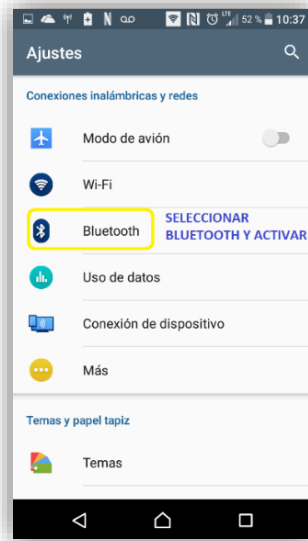
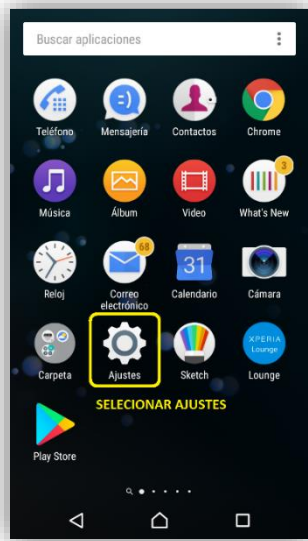
Para realizar la instalación de la aplicación deberá transferir el archivo prueba\_2.APK al directorio interno del dispositivo inteligente.

Una vez transferido el archivo ejecutable, presione sobre el archivo y seleccione la opción de “Instalar”.



## Antes de iniciar la aplicación

Para iniciar con el manejo del prototipo de mano robótica se requiere previamente emparejar los dispositivos *Bluetooth* de la mano robot y del teléfono inteligente, esto se lo realiza con los siguientes pasos, mientras la mano robótica se encuentra encendido.



1

Ir a Ajustes / Bluetooth.

Activar Bluetooth del teléfono inteligente.

En la opción de dispositivos disponibles, seleccionar la opción de "Mano\_Robotica".

Para sincronizar los dispositivos se debe ingresar la clave: Mano1234



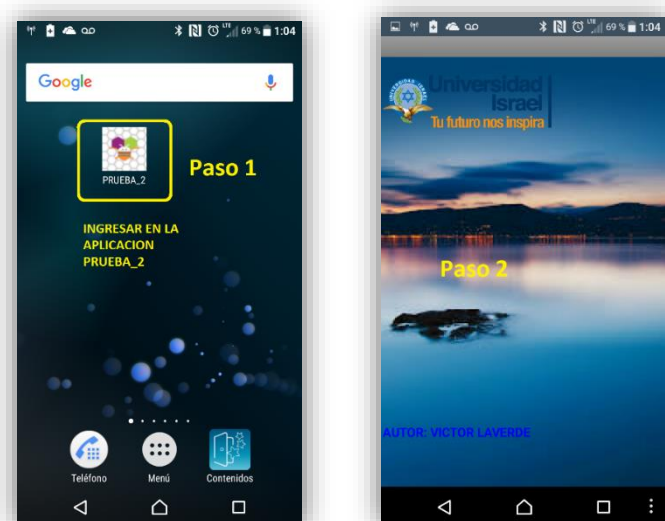
## Uso de aplicación

Finalizado la instalación de la aplicación se dispondrá del ícono en el menú del dispositivo con el nombre de PRUEBA\_2.

A continuación seguir los siguientes pasos para realizar el manejo de la mano robótica:

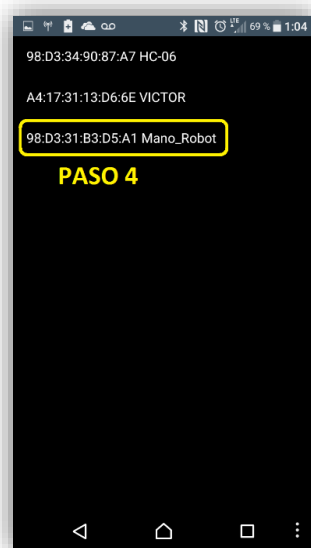
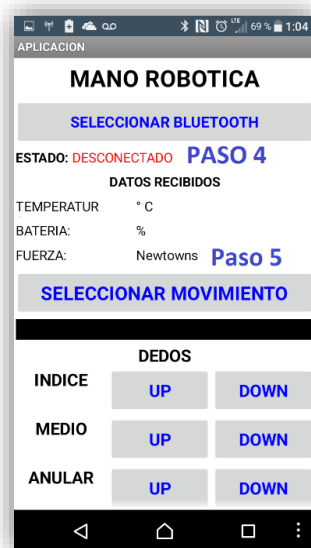
1

Ingresar en la aplicación  
Esperar 5 segundos mientras se inicia la aplicación .



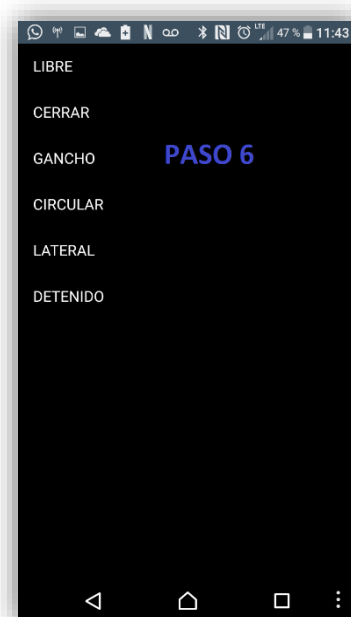
2

Presionar el botón Seleccionar *Bluetooth*.  
En el nuevo menú desplegado seleccionar "Mano\_Robotica".



3

Presionar el botón Seleccionar Movimiento para escoger el tipo de movimiento a generar en la mano robótica.



## TIPOS DE MOVIMIENTOS

Los movimientos que se pueden generar con el prototipo de mano robótica son 5, los mismos que se mencionan a continuación:

- Libre
- Cerrado
- Gancho
- Circular
- Lateral

### **Movimiento Libre:**

En el movimiento Libre se podrá manejar el movimiento de abierto y cerrado de cada dedo de la mano del prototipo, como se hace referencia en la siguiente imagen cada dedo dispone de dos botones para generar los movimientos de abierto (*UP*) y cerrado (*down*), en el caso de desear el cambio de movimiento se debe presionar primero el botón de “Detener” para salir del movimiento Libre y poder seleccionar un nuevo movimiento.

### **Movimiento Cerrado**

En el movimiento Cerrado solo se encuentra habilitados los botones de Activar y Detener, los mismos que permiten únicamente generar el movimiento de cerrado de cada dedo de forma secuencial. Para salir de este movimiento lo realizará de forma automática al momento de presionar detener.

### **Movimiento Gancho**

En el movimiento Gancho solo se encuentra habilitados los botones de Activar y Detener, los mismos que permiten únicamente generar el movimiento de semi-cerrado de cada dedo de forma secuencial para poder sujetar un objeto. Para salir de este movimiento lo realizará de forma automática al momento de presionar el botón detener.

### **Movimiento Circular**

En el movimiento Circular solo se encuentra habilitados los botones de Activar y Detener, los mismos que permiten únicamente generar el movimiento de cerrado de cada dedo de forma secuencial para poder sujetar un objeto. Para salir de este movimiento lo realizará de forma automática al momento de presionar detener.

### **Movimiento Lateral**

En el movimiento Lateral solo se encuentra habilitados los botones de Activar y Detener, los mismos que permiten únicamente generar el movimiento de cerrado del dedo índice y pulgar de forma secuencial para poder sujetar un objeto. Para

salir de este movimiento lo realizará de forma automática al momento de presionar detener.

## **SOPORTE TÉCNICO**

En el caso de inconvenientes en el prototipo, daños en los circuitos o reemplazo de las aleaciones de memoria de forma, comunicarse mediante correo electrónico su requerimiento.

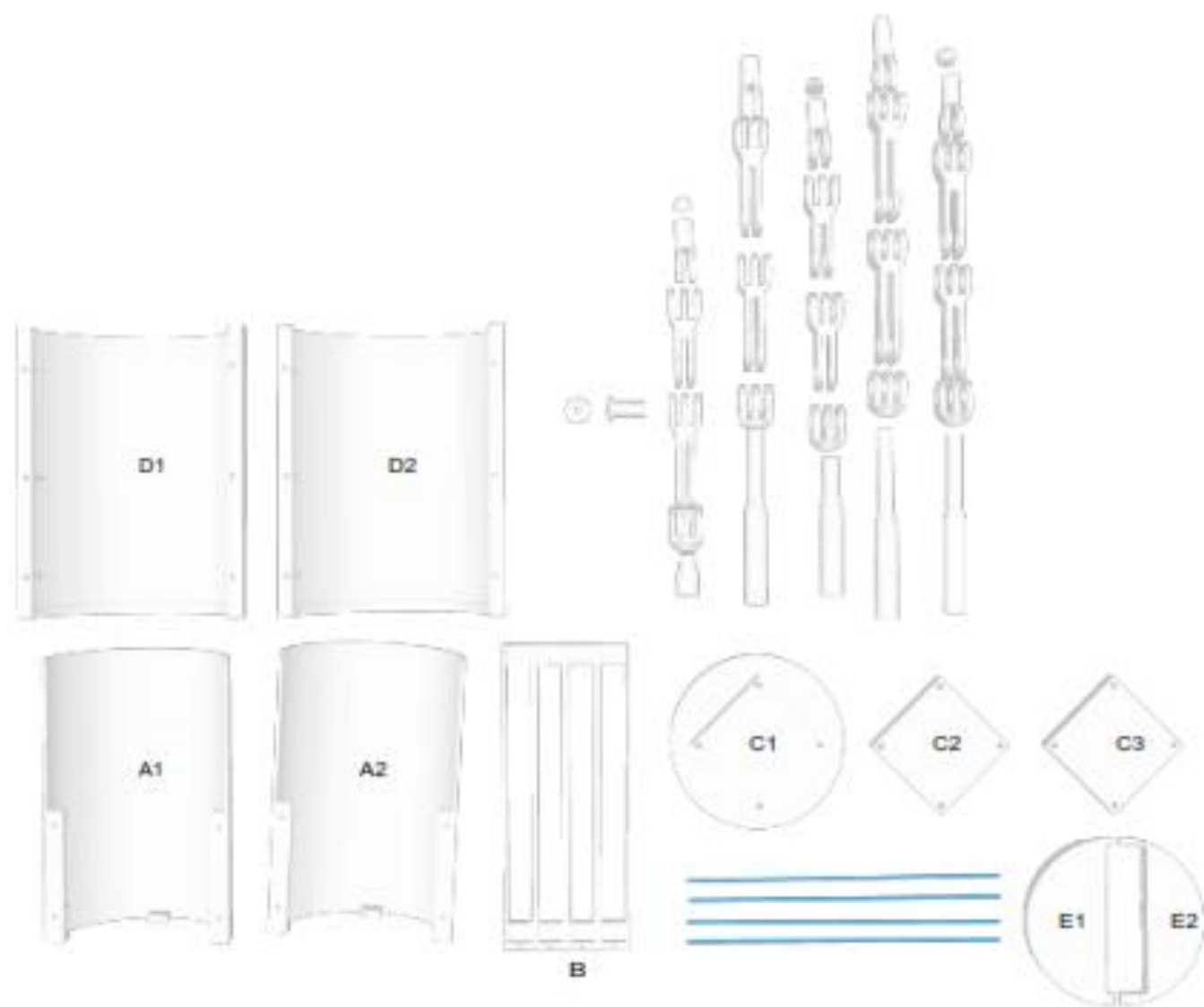
Email: [victorlaverdep@hotmail.com](mailto:victorlaverdep@hotmail.com)

Teléfono: 0992576617



**ANEXO No. 7**

**DIAGRAMA DE CONEXIÓN DE ESTRUCTURA**



Nueva esfera

**ANEXO No. 8**

**HOJA DE DATOS**



**DYNALLOY, Inc.**  
Makers of Dynamic Alloys

## Technical Characteristics of



## Actuator Wires

*Flexinol® Actuator Wires are small diameter wires which contract like muscles when electrically driven. Smaller than motors or solenoids, cheaper and generally easier to use, these wires perform physical movement across an extremely wide variety of applications.*

### Table of Contents

Physical Properties Summary	2
Introduction	3
Section 1. Movement	4
Section 2. Electrical Guidelines	6
Section 3. Cycle Time	7
Section 4. Miscellaneous	9
Section 5. Underlying Technology	11



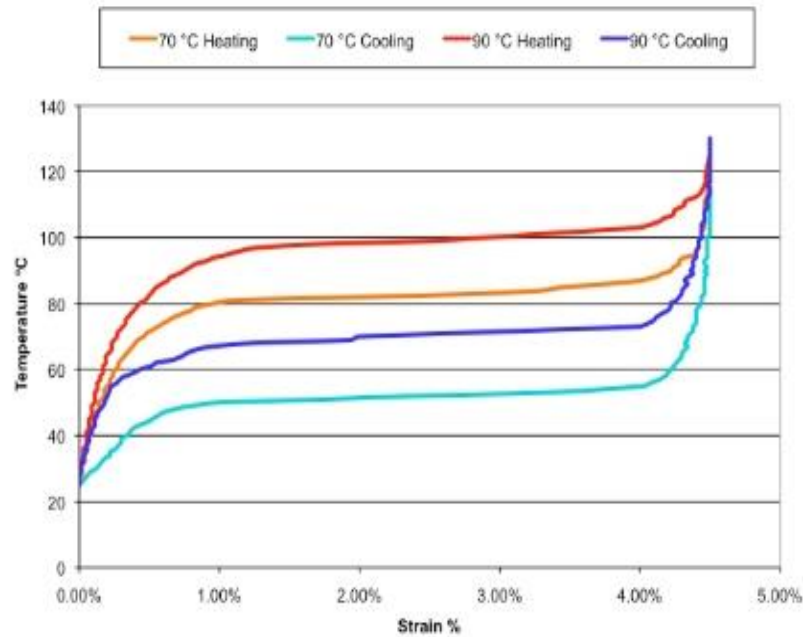
FLEXINOL®

**DYNALLOY, Inc.**  
Makers of Dynamic Alloys

## NICKEL - TITANIUM ALLOY PHYSICAL PROPERTIES

1. Density	0.235 lb/in <sup>3</sup> (6.45 g/cm <sup>3</sup> )
2. Specific Heat	0.20 BTU/lb * °F (0.2 cal/g * °C)
3. Melting Point	2370 °F (1300 °C)
4. Latent Heat of Transformation	10.4 BTU/lb (5.78 cal/g)
5. Thermal Conductivity	10.4 BTU/hr * ft * °F (0.18 W/cm * °C)
6. Thermal Expansion Coefficient	
Martensite	3.67x10 <sup>-6</sup> /°F (6.6x10 <sup>-6</sup> /°C)
Austenite	6.11x10 <sup>-6</sup> /°F (11.0x10 <sup>-6</sup> /°C)
7. Poisson Ratio	0.33
8. Electrical Resistivity (approx.)	
Martensite:	32 micro-ohms * in (80 micro-ohms * cm)
Austenite:	39 micro-ohms * in (100 micro-ohms * cm)

1562 Reynolds Avenue, Irvine, California 92614 USA 714-436-1206 714-436-0511 Fax <http://www.dynalloy.com>



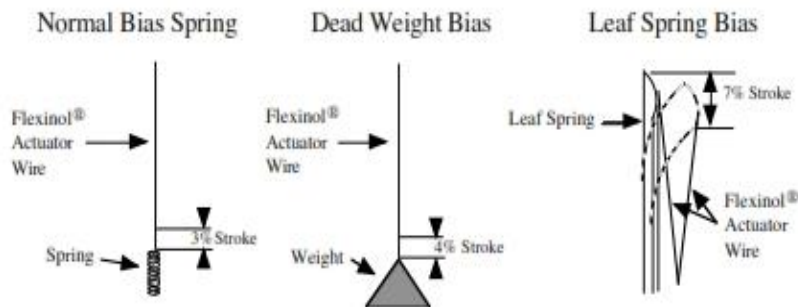
Typical Temperature vs. Strain Characteristics for Dynalloy's standard 158°F (70°C) "LT" and 194°F (90°C) "HT" Austenite start temperature alloys, at 172 MPa



## Section 1. Movement

The movement or stroke of Flexinol® actuator wire is measured as a percentage of the length of the wire being used and is determined, in part, by the level of stress one uses to reset the wire, or to stretch it in its low temperature phase. This opposing force, used to stretch the wire, is called the bias force. In most applications, the bias force is exerted on the wire constantly, and on each cycle as the wire cools, this force elongates it. If no force is exerted as the wire cools, very little deformation or stretch occurs in the cool, room temperature state and correspondingly very little contraction occurs upon heating. Up to a point the higher the load the higher the stroke. The strength of the wire, its pulling force and the bias force needed to stretch the wire back out are a function of the wire size or cross sectional area and can be measured in pounds per square inch or "psi". If a load of 5,000 psi (34.5 MPa) is maintained during cooling, then about 3% memory strain will be obtained. At 10,000 psi (69 MPa), about 4% results, and with 15,000 psi (103 MPa) and above, nearly 5% is obtained. However, there is a limit to how much stress can be applied.

Far more important to stroke is how the wire is physically attached and made to operate. Dynamics in applied stress and leverage also vary how much the actuator wires move. While normal bias springs that increase their force as the Flexinol® actuators contract have only 3-4% stroke, reverse bias forces which decrease as the actuator wires contract can readily allow the wire to flex up to 7%. Mechanics of the device in which it is used can convert this small stroke into movements over 100% of the wires' length and at the same time provide a reverse bias force. The stress or force exerted by Flexinol® actuator wires is sufficient to be leveraged into significant movement and still be quite strong. Some basic structures, their percent of movement, and the approximate available force they offer in different wire sizes are as follows:

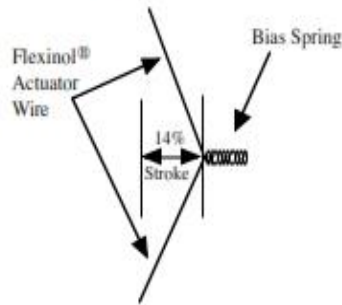




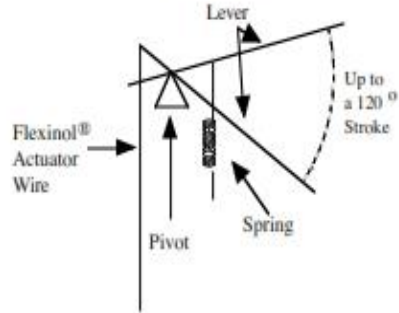
FLEXINOL®

**DYNALLOY, Inc.**  
Makers of Dynamic Alloys

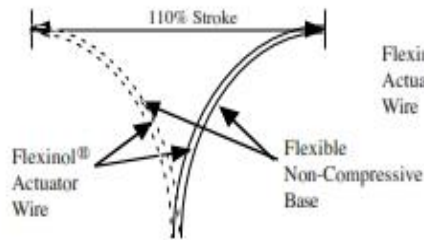
### Right Angle Pull



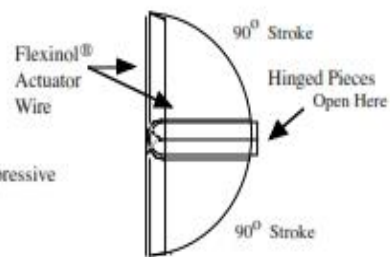
### Simple Lever



### Adjusting Curvature



### Clam Shell



1562 Reynolds Avenue, Irvine, California 92614 USA 714-436-1206 714-436-0511 fax <http://www.dynalloy.com>

### Stroke and Available Force Table

	Approx. Stroke	0.003" Wire (0.076 mm)	0.006" Wire (0.15 mm)	0.010" Wire (0.25 mm)
Normal Bias Spring	3%	0.18 lb (80 g)	0.73 lb (330 g)	2.05 lb (930 g)
Dead Weight Bias	4%	0.18 lb (80 g)	0.73 lb (330 g)	2.05 lb (930 g)
Leaf Spring Bias	7%	0.18 lb (80 g)	0.73 lb (330 g)	2.05 lb (930 g)
Right Angle Pull	14%	0.04 lb (20 g)	0.18 lb (83 g)	0.51 lb (232 g)
Simple Lever (6:1 ex)	30%	0.024 lb (11 g)	0.10 lb (47 g)	0.29 lb (133 g)
Adjusting Curvature	110%	0.006 lb (3 g)	0.026 lb (12 g)	0.075 lb (34 g)
Clam Shell	100%	0.007 lb (3.2 g)	0.028 lb (13 g)	0.082 lb (37 g)





FLEXINOL®

**DYNALLOY, Inc.**  
Makers of Dynamic Alloys

## Section 2. Electrical Guidelines

If Flexinol® actuator wire is used in the appropriate conditions, then obtaining repeatable motion from the wire for tens of millions of cycles is reasonable. If higher stresses or strains are imposed, then the memory strain is likely to slowly decrease and good motion may be obtained for only hundreds or a few thousands of cycles. The permanent deformation that occurs in the wire during cycling is heavily a function of the stress imposed and the temperature under which the actuator wire is operating. Flexinol® wire has been specially processed to minimize this straining, but if the stress is too great or the temperature too high, some permanent strain will occur. Since temperature is directly related to current density passing through the wire, care should be taken to heat, but not overheat, the actuator wire. The following charts give rough guidelines as to how much current and force to expect with various wire sizes.

Diameter Size inches (mm)	Resistance ohms/inch (ohms/meter)	Pull Force* pounds (grams)	Cooling Deformation Force* pounds (grams)	Approximate** Current for 1 Second Contraction (mA)	Cooling Time 158°F, 70°C "LT" Wire *** (seconds)	Cooling Time 194°F, 90°C "HT" Wire *** (seconds)
0.001 (0.025)	36.2 (1425)	0.02 (8.9)	0.008 (3.6)	45	0.18	0.15
0.0015 (0.038)	22.6 (890)	0.04 (20)	0.016 (8)	55	0.24	0.20
0.002 (0.050)	12.7 (500)	0.08 (36)	0.032 (14)	85	0.4	0.3
0.003 (0.076)	5.9 (232)	0.18 (80)	0.07 (32)	150	0.8	0.7
0.004 (0.10)	3.2 (126)	0.31 (143)	0.12 (57)	200	1.1	0.9
0.005 (0.13)	1.9 (75)	0.49 (223)	0.20 (89)	320	1.6	1.4
0.006 (0.15)	1.4 (55)	0.71 (321)	0.28 (128)	410	2.0	1.7
0.008 (0.20)	0.74 (29)	1.26 (570)	0.50 (228)	660	3.2	2.7
0.010 (0.25)	0.47 (18.5)	1.96 (891)	0.78 (356)	1050	5.4	4.5
0.012 (0.31)	0.31 (12.2)	2.83 (1280)	1.13 (512)	1500	8.1	6.8
0.015 (0.38)	0.21 (8.3)	4.42 (2004)	1.77 (802)	2250	10.5	8.8
0.020 (0.51)	0.11 (4.3)	7.85 (3560)	3.14 (1424)	4000	16.8	14.0

\* The Heating pull force is based on 25,000 psi (172 MPa), which for many applications is the maximum safe stress for the wire. However, many applications use higher and lower stress levels. This depends on the specific conditions of a given design. The cooling deformation force is based on 10,000 psi (70 MPa), which is a good starting point in a design. Nonetheless, this value can also vary depending on how the material is used.

\*\* The contraction time is directly related to current input. The figures used here are only approximate since room temperatures, air currents, and heat sinking of specific devices vary. On small diameter wires (<= 0.006" diameter) currents that heat the wire in 1 second can typically be left on without over-heating it.

\*\*\* Approximate cooling time, at room temperature in static air, using a vertical wire. The last 0.5% of deformation is not used in these approximations.



# HOJA DE DATOS DE SENSOR DE FUERZA FSR402



## FSR® 400 Series Data Sheet

Force Sensing Resistors®

### Features and Benefits

- Actuation force as low as 0.2N and sensitivity range to 20N
- Cost effective
- Ultra thin
- Robust; up to 10M actuations
- Simple and easy to integrate

### Description

Interlink Electronics FSR® 400 Series is part of the single zone Force Sensing Resistor® family. Force Sensing Resistors, or FSR's, are robust polymer thick film (PTF) devices that exhibit a decrease in resistance with increase in force applied to the surface of the sensor. This force sensitivity is optimized for use in human machine interface devices including automotive electronics, medical systems, industrial controls and robotics.

The FSR 400 Series sensors come in seven different models with four different connecting options. A battery operated demo is available. Call us for more information at +1 805-484-8855.



FSR® 400 Short  
5mm Circle x 20mm



FSR® 400  
5mm Circle x 38mm



FSR® 402 Short  
13mm Circle x 25mm



FSR® 402  
13mm Circle x 56mm



FSR® 404  
20mm Donut with 5.5mm hold



FSR® 406  
38mm Square x 83mm



FSR® 408-xxx  
10mm Wide x xxx mm strip  
xxx = 50, 100, 200, 300, 400, 500mm

PIN: FDS-10004-C

Human-Machine Interface Solutions for a Connected World

# FSR<sup>®</sup> 400 Series Data Sheet

Force Sensing Resistors<sup>®</sup>

## Device Characteristics

Actuation Force*	-0.2N min
Force Sensitivity Range*	-0.2N – 20N
Force Resolution	Continuous (analog)
Force Repeatability Single Part	+/- 2%
Force Repeatability Part to Part	+/- 6% (Single Batch)
Non-Actuated Resistance	>10 Mohms
Hysteresis	+10% Average (RF+ - RF-)/RF+
Device Rise Time	< 3 Microseconds
Long Term Drift 1kg load, 35 days	< 5% log10(time)
Operating Temperature Performance Cold: -40°C after 1 hour Hot: +85°C after 1 hour Hot Humid: +85°C 95RH after 1 hour	-5% average resistance change -15% average resistance change +10% average resistance change
Storage Temperature Performance Cold: -25°C after 120 hours Hot: +85°C after 120 hours Hot Humid: +85°C 95RH after 240 hours	-10% average resistance change -5% average resistance change +30% average resistance change
Tap Durability Tested to 10 Million actuations, 1kg, 4Hz	-10% average resistance change
Standing Load Durability 2.5kg for 24 hours	-5% average resistance change
EMI	Generates no EMI
ESD	Not ESD Sensitive
UL	All materials UL grade 94 V-1 or better
RoHS	Compliant

Specifications are derived from measurements taken at 1000 grams, and are given as (one standard deviation/mean), unless otherwise noted.  
\*Typical value. Force dependent on actuation interface, mechanics, and measurement electronics.

# FSR<sup>®</sup> 400 Series Data Sheet

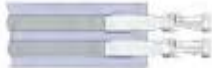
Force Sensing Resistor<sup>®</sup>

## Connector Information

### Bare Tail



Female Tin Contacts  
PN: TE 2-487406-4



Female Tin Contacts with 2 Pin Housing  
PN: TE 2-487408-4  
PN: TE 487378-1



Solder Tabs  
PN: TE 1-88997-2



Other Available Part Numbers:  
Hardware Development Kit, PN 54-78247

## Application Information

For specific application needs please contact Interlink Electronics support team. An Integration Guide and Hardware Development Kit (HDK) are also available. FSR's are two-wire devices with a resistance that depends on applied force. Below is a force vs. resistance graph that illustrates a typical FSR<sup>®</sup> response characteristic. Please note that the graph values are reference only and actual values are dependent upon actuation system mechanics and sensor geometry.

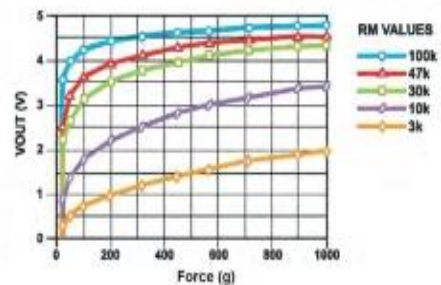
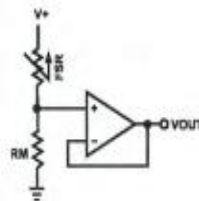
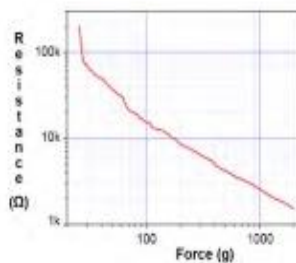
For simple force-to-voltage conversion, the FSR device is tied to a measuring resistor in a voltage divider (see figure below) and the output is described by the following equation.

$$V_{OUT} = \frac{R_M V +}{(R_M + R_{FSR})}$$

In the configuration shown, the output voltage increases with increasing force. If  $R_{FSR}$  and  $R_M$  are swapped, the output swing will decrease with increasing force. The measuring resistor,  $R_M$ , is chosen to maximize the desired force sensitivity range and to limit current. Depending on the impedance requirements of the measuring circuit, the voltage divider could be followed by an op-amp.

A family of force vs.  $V_{OUT}$  curves is shown on the graph below for a standard FSR in a voltage divider configuration with various  $R_M$  resistors. A  $V+$  of 5V was used for these examples. Please note that the graph values are for reference only and will vary between different sensors and applications.

Refer to the FSR Integration Guide for more integration methods and techniques.



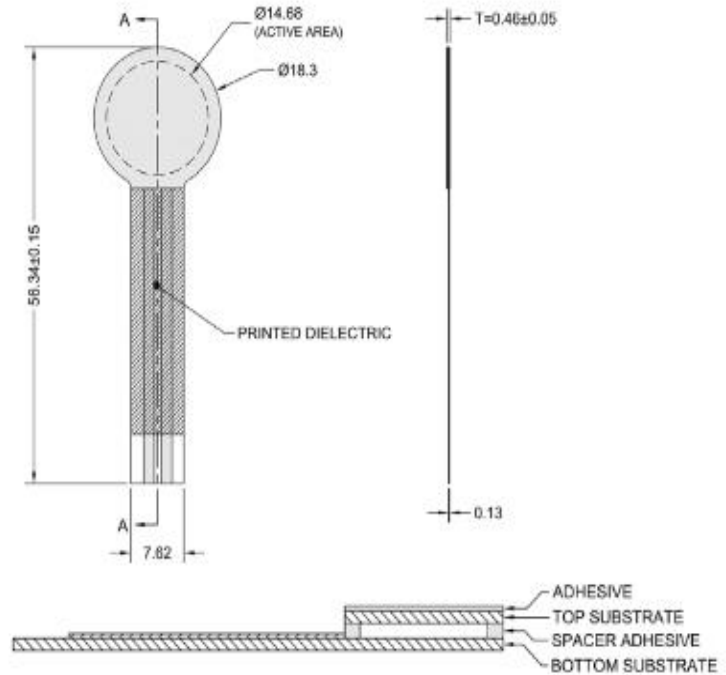
**Model 402:**

Active Area:  $\varnothing 14.68\text{mm}$   
 Nominal Thickness:  $0.46\text{mm}$   
 Switch Travel:  $0.15\text{mm}$

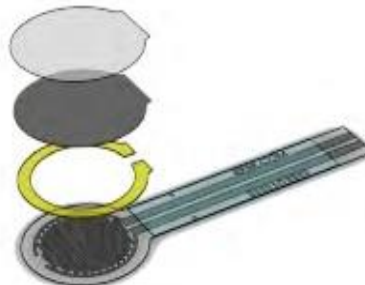
**Available Part Numbers:**

- PN: 44-29103 Model 402  
- No contacts or solder tabs
- PN: 34-00012 Model 402  
- with female contacts
- PN: 34-00001 Model 402  
- with female contacts and housing
- PN: 30-81794 Model 402  
- with solder tabs

**Sensor Mechanical Data**



**Exploded View**





## DATASHEET BLUETOOTH TO SERIAL PORT MODULE HC05



### Overview

HC-05 module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup.

Serial port Bluetooth module is fully qualified Bluetooth V2.0+EDR (Enhanced Data Rate) 3Mbps Modulation with complete 2.4GHz radio transceiver and baseband. It uses CSR Bluecore 04-External single chip Bluetooth system with CMOS technology and with AFH (Adaptive Frequency Hopping Feature). It has the

footprint as small as 12.7mmx27mm. Hope it will simplify your overall design/development cycle.

[www.electronica60norte.com](http://www.electronica60norte.com)  
[electronica60norte@hotmail.com](mailto:electronica60norte@hotmail.com)

# Specifications

## Hardware features

- Typical -80dBm sensitivity.
- Up to +4dBm RF transmit power.
- Low Power 1.8V Operation, 3.3 to 5 V I/O.
- PIO control.
- UART interface with programmable baud rate.
- With integrated antenna.
- With edge connector.

## Software features

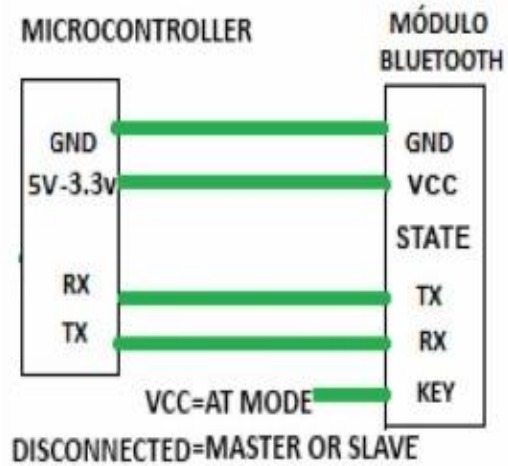
- Slave default Baud rate: 9600, Data bits:8, Stop bit:1,Parity:No parity.
- PIO9 and PIO8 can be connected to red and blue led separately. When master and slave are paired, red and blue led blinks 1time/2s in interval, while disconnected only blue led blinks 2times/s.
- Auto-connect to the last device on power as default.
- Permit pairing device to connect as default.
- Auto-pairing **PINCODE:"1234"** as default.
- Auto-reconnect in 30 min when disconnected as a result of beyond the range of connection.

www.electronica60norte.com  
electronica60norte@hotmail.com

## Pin out configuration

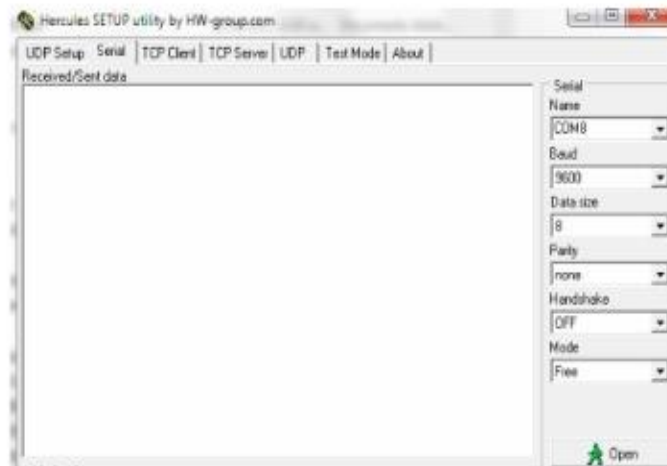


## Typical Application Circuit



www.electronica60norte.com  
electronica60norte@hotmail.com

Open connection and you will be ready to send and receive data from module Bluetooth like Serial Port COM



## AT COMMANDS

### How to get to AT COMMAND mode

- 1: Connect KEY pin to VCC.
- 2: Supply power to module. Then the module will enter into AT MODE. In this mode you have to use baud rate at 38400. In this way, user should change the baud rate for SLAVE AND MASTER mode.

### How to set this module as "Master - Host" role

- 1: Input high level to KEY.
- 2: Supply power to the module. And the module will enter to AT COMMAND.
- 3: Set the parameters of the hyper terminal or the other serial tools (baud rate: 38400, data bit:8, stop bit:1, no parity bit, no Flow Control).
- 4: Sent the characters "AT+ROLE=1\r\n" through serial, then receive the characters "OK\r\n". Here, "\r\n" is the CRLF.
- 5: Sent the characters "AT+CMODE=1\r\n" through serial, then receive the characters "OK\r\n". Here, "\r\n" is the CRLF.
- 6: Default factory password passkey is: 1243, this must be the same in the Bluetooth slave module if you want to pair it.  
To read passkey use this command: "AT+PSWD?".  
To Reset the password command sent the characters "AT+PSWD=XXXX".  
The password must be 4-bits.



## DECLARACIÓN Y AUTORIZACIÓN

Yo, Víctor Hugo Laverde Paucar, CI 1723504013 autor del trabajo de graduación:

**Implementación de un Prototipo de Mano Robótica con Materiales Inteligentes**, previo a la obtención del título de **Ingeniería en Electrónica Digital y Telecomunicaciones** en la UNIVERSIDAD TECNOLÓGICA ISRAEL.

1.- Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de difundir el respectivo trabajo de graduación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la SENESCYT a tener una copia del referido trabajo de graduación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Quito, Marzo del 2018

**Atentamente.**



**Víctor Hugo Laverde Paucar.**  
C.I. 1723504013



MY.PLAG.ES

## REPORTE DE PLAGIO

EL REPORTE CERTIFICA QUE EL DOCUMENTO ADJUNTO

***TESIS VICTOR LAVERDE.docx***

FUE REVISADO CON EL SERVICIO DE PREVENCIÓN DE PLAGIO MY.PLAG.ES

Y TIENE:

SIMILITUD

**2%**

PARAFRASEA

**0%**

CITAS INCORRECTA

**0%**



MY.PLAG.ES

2%

TESIS REVISADA VICTOR LAVERDE.docx

0%

0%

14

TESIS REVISADA VICTOR LAVERDE.docx

hace 1 día

2%

Similitud

0%

Referencias

0%

Citas incorrectas

14

Coincidencias

BAJO RIESGO DE PLAGIO

Nombre del archivo: TESIS REVISADA VICTOR LAVERDE.docx

Archivo verificado: 2018-02-14

Reporte generado: 2018-02-15

<https://my.plag.es/file/TESIS-REVISADA-VICTOR-LAVERDE-docx-2>



*Handwritten signature*