

**DISEÑO E IMPLEMENTACION DE UN HARDWARE (TARJETA) PARA
INTERCONECTAR LA PARTE FISICA CON EL COMPUTADOR PARA LA
REALIZACION DE LABORATORIOS REMOTOS**

FRANCISCO ANDRES GOMEZ MORALES

Código: 2009103021

UNIVERSIDAD PEDAGOGICA NACIONAL

FACULTAD DE CIENCIA Y TECNOLOGIA

DEPARTAMENTO DE TECNOLOGIA

LICENCIATURA EN ELECTRONICA

BOGOTA

2016

**DISEÑO E IMPLEMENTACION DE UN HARDWARE (TARJETA) PARA
INTERCONECTAR LA PARTE FISICA CON EL COMPUTADOR PARA LA
REALIZACION DE LABORATORIOS REMOTOS**

FRANCISCO ANDRES GOMEZ MORALES

Código: 2009103021

Director:

DIEGO FERNANDO QUIROGA

Licenciado en Electrónica

UNIVERSIDAD PEDAGOGICA NACIONAL

FACULTAD DE CIENCIA Y TECNOLOGIA

DEPARTAMENTO DE TECNOLOGIA

LICENCIATURA EN ELECTRONICA

BOGOTA

2016

Nota de aceptación:

Firma del Director

Firma del Jurado 1

Firma del Jurado 2

Bogotá, Agosto de 2016

AGRADECIMIENTOS

Agradezco primero que todo a Dios que me ha dado la vida y me ha permitido salir adelante cada día, luego a mis padres quienes me han apoyado tanto moralmente como económicamente en todo este proceso de formación como profesional. Además, agradezco a mis amigos quienes me acompañaron y han ayudado emocionalmente desde el inicio del pregrado, también, a los docentes quienes me formaron profesionalmente para poder cumplir con mi proyecto de vida y optar por el título de licenciado.

DEDICATORIAS

Dedico este logro a todas las personas que han estado apoyándome en todo el proceso para convertirme en profesional, que con su apoyo y compañía me dan fuerzas para seguir adelante con mi proyecto de vida.

Pero muy especialmente se lo dedico a mi novia, aunque no está físicamente conmigo sé que desde el cielo me está apoyando, que con el transcurso de los días me da fuerzas para poder cumplir este sueño el cual es ser profesional.

Francisco Andrés Gómez Morales

RESUMEN ANALITICO EN EDUCACION - RAE

1. Información General	
Tipo de documento	Trabajo de Grado
Acceso al documento	Universidad Pedagógica Nacional. Biblioteca Central
Título del documento	Diseño e implementación de un hardware (tarjeta) para interconectar la parte física con el computador para la realización de laboratorios remotos
Autor(es)	Gómez Morales, Francisco Andrés
Director	Lic. Quiroga Páez Diego Fernando
Publicación	Bogotá. Universidad Pedagógica Nacional, 2016. 81 p.
Unidad Patrocinante	Universidad Pedagógica Nacional UPN
Palabras Claves	PYTHON; RASPBERRY PI; HARDWARE; MOTOR DC; SERVOMOTOR; MOTOR PASO A PASO; SENSOR ULTRASONIDO; PROTOCOLO DE COMUNICACIÓN SSH; PINES GPIO.

2. Descripción
<p>Este trabajo describe de una forma general el diseño de un hardware que controla varios dispositivos usados en la Licenciatura en Electrónica de la Universidad Pedagógica Nacional basado en una aplicación programada en Python. Esta aplicación controla una tarjeta externa por el puerto Ethernet cuya función es controlar un motor DC, servomotor, motor paso a paso y un sensor ultrasonido. El programa se ejecuta por medio de un menú con varias opciones mostrado en la línea de comandos del hardware diseñado.</p>

3. Fuentes
<p>Universidad de Murcia. (s.f.). ¿Qué es el aula virtual? [Fecha de consulta: 6 de mayo de 2015]. Disponible en: <http://www.um.es/aulavirtual/primeros-pasos/que-es-el-aula-virtual/></p> <p>Centro Nacional de desarrollo e investigación en tecnologías libres. (s.f.) Hacia una definición de Hardware libre. [Fecha de consulta: 28 de abril de 2015]. Disponible en: <http://www.cenditel.gob.ve/node/691></p> <p>Aliane N, Fernández J, Martínez A y Ortiz J, Un laboratorio de Ingeniería de Control basado en internet. Información Tecnológica – Vol. 18 Nº 6 – 2007</p> <p>Candelas F, Torres F, Gil P, Ortiz F, Puente S, Pomares J, (2004). Laboratorio virtual remoto para robótica y evaluación de su impacto en la docencia. Revista Iberoamericana de automática e informática industrial, Vol. 1, Núm. 2.</p> <p>Turón R, Picos R, Roca M. Experimentos de electrónica a través de internet. [Fecha de consulta: 21 de abril de 2015]. Disponible en:</p>

<<http://www.upc.edu/euetib/xiicueet/comunicaciones/din/comunicacions/113.pdf>>
 Upton, C, (s.f). *Programming The Raspberry Pi.element14*. www.element14.com
 Gonzales. ((2010)). *Python para todos*. España: Autoedición.
 Downey, A.E. ((2002)). *Aprenda a Pensar como un Programador con Python*. Wellesley, Massachusetts: Green Tea Press
 Gilfillian, I. (s.f.). *La biblia MySQL*. ANAYA MULTIMEDIA.
 Kenkeiras. (s.f.) *De 0 a Python*. Hack x Crack
 Wachenchauser, R. (2011). *Algoritmos y Programación con lenguaje Python*. San Francisco, California, USA.

4. Contenidos

En este trabajo de grado se encontraran con diversos apartados importantes. Aquí solo se mencionan algunos de ellos:

- Hardware
- Python
- Motor DC
- Motor paso a paso
- Servomotor
- Sensor ultrasonido
- Protocolo de comunicación SSH
- Pines GPIO
- Control de los motores
- Interfaz grafica
- Implementación

5. Metodología

Este proyecto se divide en diversas etapas, siendo esta parte fundamental en el desarrollo del software, el cual permite el control de los elementos usados, los cuales permiten evidenciar el proceso de cada una de las etapas para el desarrollo de la tarjeta. La primera etapa es la recopilación de la información, donde se recolecta y se establecen las necesidades del estudiante. La segunda etapa es el diseño y desarrollo de la tarjeta, en la que se realiza la programación de los componentes escogidos que son los más propicios para la elaboración de la tarjeta. Por último la implementación de la tarjeta para verificar la funcionalidad del producto obtenido.

A continuación se describen las etapas de este proyecto:

1. **Análisis:** En esta etapa se analizan los requerimientos que son necesarios para la elaboración de la tarjeta, donde se verifica los materiales necesarios para la creación del hardware; y así evaluar y analizar la información obtenida para su respectiva construcción.
2. **Diseño y construcción:** En esta etapa se toma en cuenta las necesidades del estudiante frente a la tarjeta para su respectivo desarrollo, con los materiales ya definidos para el desarrollo y el manejo de los motores, con una adecuada comunicación con el computador.
- ✓ **Comunicación con el pc:** El estudiante para poder controlar el motor que desee (servomotor, paso a paso o DC), primero que todo tiene que tener ciertos programas instalados (posteriormente se explica en detalle cuales son), usando así el protocolo de comunicación SSH desarrollado para esta tarjeta permitiendo una manipulación adecuada de los motores antes mencionados.

3. **Implementación:** En esta etapa se realizan las pruebas, verificando el funcionamiento correcto de la tarjeta, donde se analiza las necesidades requeridas para así poder satisfacer lo que el usuario vaya a usar.

6. Conclusiones

En definitiva respecto al proyecto se concluye que:

- Este hardware proporciona al estudiante una herramienta para el control de los motores (DC, SERVO, PASO A PASO y SENSOR ULTRASONIDO) que son algunos de los elementos más usados en la licenciatura en electrónica, brindando al profesor una manera diferente e innovadora de realizar laboratorios para las diferentes asignaturas.
- Python es un lenguaje de programación muy compacto, generalmente es mucho más corto que sus similares lenguajes como C, es considerado un lenguaje de programación de muy alto nivel, además es muy versátil, puede ser orientado a objetos y logra interactuar con HTML para visualizar bases de datos, etc.
- El protocolo de comunicación SSH facilita las comunicaciones seguras entre cliente/servidor, además permite al usuario conectarse a un host remotamente, SSH encripta todo lo que envía y recibe proporcionando un nivel alto de seguridad para los hackers.
- La implementación de este hardware incide de una manera favorable el interés por los laboratorios remotos, ya que algunas veces no se puede desarrollar la práctica por diversos motivos ya sean de orden público, falta de materiales o no se encuentra disponible el espacio.
- El raspberry pi como arquitectura hardware por su reducido tamaño, su alta capacidad de proceso, su bajo costo, sus pines GPIO y su lenguaje de programación la convierten en un elemento de suma importancia para la realización de este proyecto, además raspberry pi es prácticamente una computadora funcional, lo cual la hacen el cerebro ideal para el hardware que se diseñó.

Elaborado por:	Gómez Morales, Francisco Andrés
Revisado por:	Lic. Quiroga Páez Diego Fernando

Fecha de elaboración del Resumen:	29	08	2016
--	----	----	------

TABLA DE CONTENIDO

1. INTRODUCCIÓN.....	13
2. ANTECEDENTES.....	15
3. JUSTIFICACIÓN.....	17
4. PROBLEMA DE INVESTIGACIÓN.....	18
5. OBJETIVO GENERAL.....	19
6. OBJETIVOS ESPECÍFICOS.....	19
7. MARCO TEÓRICO.....	20
7.1. Hardware.....	20
7.2. Raspberry Pi 2.....	20
7.3. Python.....	22
7.4. Motor DC.....	22
7.5. Motor Paso a paso.....	23
7.6. Servomotor.....	24
7.7. Sensor Ultrasonido.....	25
7.8. Puente H.....	27
7.9. Protocolo SSH.....	29
8. METODOLOGÍA.....	31
8.1. Análisis.....	32
8.2. Diseño y Construcción.....	33
8.2.1. Pines GPIO.....	36
8.2.2. Protocolo de Comunicación SSH.....	40
8.2.2.1. Configuración de Windows 7 u 8.....	41
8.2.2.2. Configuración dirección estática del raspberry Pi.....	44
8.2.2.3. Verificar la dirección IP en el Raspberry Pi.....	45
8.2.2.4. Verificar si la conexión fue exitosa.....	46
8.2.2.5. Activación del Protocolo SSH en el Raspberry Pi.....	48

8.2.2.6. Instalación del Programa Putty.....	49
8.2.3. Control de los motores y sensor ultrasonido.....	53
8.2.3.1. Control DC.....	53
8.2.3.1.1. Definición de las librerías en Python.....	54
8.2.3.1.2. Definición de las funciones.....	54
8.2.3.2. Control Motor Paso a Paso.....	57
8.2.3.3. Control Servo Motor.....	59
8.2.3.4. Control Sensor Ultrasonido.....	60
8.2.4. Alimentación.....	63
8.3. Implementación.....	63
8.3.1. Interfaz Gráfica.....	63
8.3.1.1. Comando SUDO.....	64
8.3.2. Menú Principal.....	65
8.3.2.1. Opción 1 Motor Paso a Paso.....	67
8.3.2.2. Opción 2 Menú Motor DC.....	68
8.3.2.3. Opción 3 Menú Motor Servo.....	71
8.3.2.4. Opción 4 Lectura Sensor.....	72
9. CONCLUSIONES.....	74
10. ANEXOS.....	75
11. BIBLIOGRAFIA.....	81

LISTA DE FIGURAS

Figura 1. Raspberry Pi 2 model B.....	21
Figura 2. Motor DC.....	23
Figura 3. Motor paso a paso.....	24
Figura 4. Servomotor.....	25
Figura 5. Grafica del Sensor Ultrasonido.....	26

Figura 6. Sensor Ultrasonido.....	27
Figura 7. Circuito puente H.....	28
Figura 8. Puente H con L298.....	34
Figura 9. Driver con ULN2003.....	34
Figura 10. Pines GPIO Raspberry Pi 2 Model B.....	36
Figura 11. Pines Raspberry pi 2 conectado a Puente H L298.....	37
Figura 12. Pines Raspberry pi 2 model b conectado a Driver ULN2003.....	38
Figura 13. Pines Raspberry pi 2 model b conectado a Servomotor.....	39
Figura 14. Pines Raspberry pi 2 model b conectado al Sensor Ultrasonido.....	40
Figura 15. Centro de redes y recursos compartidos de Windows 7.....	41
Figura 16. Propiedades de la ventana Ethernet.....	42
Figura 17. Configuración Dirección IP del Computador.....	43
Figura 18. Características desde la configuración hecha en el Pc.....	44
Figura 19. Ventana que comprueba que las modificaciones fueron guardadas.....	45
Figura 20. Verificación de la Dirección IP en la Raspberry.....	46
Figura 21. Verificación de que la conexión fue exitosa.....	47
Figura 22. Al hacer ping se comprueba que no se pierde ningún dato.....	48
Figura 23. Configuración del Programa Putty.....	49
Figura 24. Dirección IP del PC cargado y guardado en el programa Putty.....	50
Figura 25. Conexión exitosa con el programa Putty.....	51
Figura 26. Dirección IP de la raspberry cargada en el VNCSERVER.....	52
Figura 27. Espacio para digitar la contraseña en el VNCSERVER.....	52
Figura 28. Escritorio remoto de la raspberry listo para empezar a trabajar en Python.....	53
Figura 29. Diagrama de flujo Control Motor DC.....	56
Figura 30. Diagrama de flujo Control Motor paso a paso.....	58

Figura 31. Diagrama de flujo Control Servo Motor.....	60
Figura 32. Diagrama de flujo Lectura Ultrasonido.....	62
Figura 33. Cargador Canaki Raspberry pi 2.....	63
Figura 34. LX Terminal que tiene la raspberry para ejecutar el control.....	64
Figura 35. Ingreso de root y contraseña en el programa Putty.....	65
Figura 36. Menú Principal del Hardware.....	66
Figura 37. Menú Motor paso a paso.....	67
Figura 38. Elección de número de pasos 4 o 8.....	68
Figura 39. Menú Motor DC.....	69
Figura 40. Ingreso valor de PWM de 0 a 100.....	70
Figura 41. Menú Servo Motor.....	71
Figura 42. Menú de Lectura Sensor.....	72
Figura 43. Lecturas hechas por el sensor.....	73

INDICE DE TABLAS

Tabla 1. Configuración de las Sn de un Puente H.....	28
Tabla 2. Características Raspberry Pi 2 model B.....	35

1. INTRODUCCIÓN

Este proyecto gira en torno a varios campos del desarrollo, en este documento nos centramos en la educación, ya que se plantea una opción de realizar laboratorios no presenciales, permitiendo el mismo aprendizaje que se puede llegar con uno presencial, el otro campo es de la electrónica ya que es la base donde se eligió para construir la tarjeta, donde será implementado para fortalecer la licenciatura de la Universidad Pedagógica Nacional.

La educación actual necesita la experimentación práctica, para complementar lo teórico, y una de los caminos más fáciles para que esto sea viable, se realiza mediante los laboratorios presenciales con el avance tecnológico, dejando así la posibilidad de implementar los laboratorios remotos. Los dispositivos de acceso remoto se han utilizado, por mucho tiempo, para experimentar diversas situaciones, por ejemplo, la exploración espacial (satélites) o acceder a lugares a los cuales es difícil llegar.

Los laboratorios remotos son una herramienta demasiado importante para la enseñanza, para la industria y los centros de investigación. Aunque para que el control de los dispositivos sea más eficaz es indispensable el internet, ya que esta presenta una opción para resolver las necesidades de los usuarios al permitir el acceso a equipos físicos desde diversas distancias.

Para la enseñanza, los laboratorios remotos no reemplazan los laboratorios tradicionales, solo ayudan a complementarlos. Poniendo en contexto, a cualquier estudiante que tenga la necesidad de estar presente en el lugar, para poder realizar sus propias prácticas de laboratorio de acuerdo a sus inquietudes, pero para la ejecución de las prácticas tradicionales, el uso de

los laboratorios remotos nos lleva a disminuir los costos y permitir un mejor acceso, ya que un solo equipo puede ser compartido por varios educandos.

Cuando hablamos de hardware libre hacemos referencia a la libertad de poder utilizar cualquier dispositivo y su respectiva documentación, esto no quiere decir que sea de manera gratuita. Aunque comparte filosofía con el software libre, debido a la propia naturaleza de estos componentes físicos, muchos de sus preceptos son limitados, básicamente se considera que un hardware es abierto si cumple una serie de requisitos.

En el presente trabajo se describe de una forma general el diseño de un hardware que controla varios dispositivos usados en la licenciatura en electrónica de la universidad pedagógica nacional basado en una aplicación programada en Python. Esta aplicación controla una tarjeta externa por el puerto Ethernet cuya función es controlar un motor DC, Servo motor, Motor paso a paso y un sensor Ultrasonido. El programa se ejecuta por medio de un menú con varias opciones mostrado en la línea de comandos del hardware diseñado.

2. ANTECEDENTES

- *Sánchez Moreno José, (2001), Análisis, diseño y desarrollo de un laboratorio virtual y remoto para el estudio de la automática a través de Internet. Universidad de Educación a distancia, España.*

Consiste en crear un sistema de apoyo a la enseñanza automática en un laboratorio virtual accesible a través de una red basada en protocolos TCP/IP permitiendo al alumno hacer una práctica de la mejor forma posible como si *estuviera en un laboratorio real.*

- *Eduardo Caicedo Bravo, Bladimir Bacca Cortés, Bayron Andrés Calvache, Jaiber Evelio Cardona, Jaime Alberto Buitrago, (2009), Laboratorio distribuido con acceso remoto para la enseñanza de la robótica. Universidad del Valle y Universidad del Quindío, Colombia.*

Este proyecto permitió el desarrollo de sesiones interactivas con los estudiantes de las dos universidades sobre plataformas móviles, con esto se pretendió enseñar la robótica permitiendo la experimentación, el mejoramiento de la comunicación entre los estudiantes, grupos de investigación y el desarrollo de actividades educativas e investigaciones multidisciplinarias.

- *Marianela Laura Quisbert, (11 – 10 – 2010), Implementación de un laboratorio Virtual para el aprendizaje de Electrónica, Universidad Autónoma de México (UNAM), México.*

Se desarrolla un sistema que se utiliza varios montajes básicos en electrónica, en este proyecto se usa un microcontrolador que interactúa con el servidor mediante un puerto. En cuanto a la visualización grafica se hace mediante una interfaz gráfica, que

se monta en una página web con bases de datos, permitiendo la interacción entre el estudiante y el profesor encargado del laboratorio virtual.

- *Georman Gerardo Calderón Briceño, (2004), Diseño de tarjeta Controladora de motores DC de altas prestaciones. Universidad Simón Bolívar, Venezuela.*

Este proyecto nació de la necesidad de disponer de una tarjeta que pudiese ser incorporada a un robot caminante y también pueda ser usada para controlar cuatro topologías básicas de brazos mecánicos en el laboratorio de mecatrónica. Se diseñó un circuito impreso que será el soporte de la tarjeta.

- *Rojas Molina A., Herrera Ruiz G., Castañeda Miranda R., Terrazas Lara A., Flores Chaparro J.H., Barón Rodríguez B. y Chaparro Sánchez R., (2010), Diseño de tarjeta electrónica genérica para el control de motores trifásicos. Universidad Autónoma de Querétaro, México.*

En este proyecto se hizo el diseño y el desarrollo de un manejador de potencia con controlador para un motor trifásico de costo reducido y con una amplia gama de aplicaciones en la industria.

- *Carlos Fernando Escobar García, José Alejandro Martínez Peña, German Alberto Téllez González, (2005), Control de un motor Brushless DC con frenado regenerativo. Pontificia Universidad Javeriana, Colombia.*

Este proyecto busco cambiar el concepto de frenado, reemplazando la fricción como forma actual de detener un vehículo, por una en la cual logre frenar sin disipar la energía para así almacenarla y aumentar la eficiencia del motor.

3. JUSTIFICACIÓN

La evolución de la tecnología ha permitido en los últimos años la creación de un nuevo escenario de experimentación: los laboratorios remotos, en comparación a los laboratorios presenciales tradicionales y los laboratorios virtuales, un laboratorio remoto es, como su nombre indica, un laboratorio utilizable a distancia. Este proyecto pretende optimizar el aprendizaje de los estudiantes de la licenciatura en electrónica para la realización de laboratorios remotos a futuro. Los estudiantes juegan un papel importante en el aprendizaje debido a que son los responsables de su propio aprendizaje, en cuanto a los profesores deben buscar los medios más adecuados para que el proceso de aprendizaje sea lo mejor posible y este proyecto es una muy buena manera incentivar el aprendizaje.

4. PROBLEMA DE INVESTIGACIÓN

Con este proyecto se pretende elaborar una tarjeta que permita conectar algunos implementos físicos con el computador creando una forma diferente para la realización de laboratorios remotos, este hardware es una ayuda para el mejoramiento de las clases, en donde los estudiantes y profesores puedan realizar de una manera más cómoda los laboratorios propuestos por los docentes.

Esta tarjeta pretende conectar algunos elementos básicos usados en la licenciatura en electrónica con un protocolo de comunicación que permita conectarse a una interfaz gráfica en el computador, es decir, podrá controlar motores paso a paso, servomotores, motores DC, etc. si el estudiante lo requiere mediante un PWM que será implementado en el hardware; se ubicara en la misma un puerto serial y/o USB para que pueda conectarse fácilmente al computador de la persona que lo esté usando, creando así una manera diferente de realizar los laboratorios a distancia, debido a que no siempre puede estar disponible el espacio del laboratorio en físico o las condiciones externas de la universidad no lo permitan.

La implementación será el resultado final a partir de las pruebas hechas con las posibles tarjetas a desarrollar en este proyecto, es decir, se probará arduino, PIC'S, etc. Dependiendo de los ensayos realizados, se escogerán los componentes más adecuados y eficientes en el cual se pueda construir el hardware deseado.

Este proyecto será una herramienta de gran apoyo para los docentes y estudiantes en las que se integran las tecnologías de la información y la comunicación (tics) como instrumento para el mejoramiento de las actividades propuestas.

5. OBJETIVO GENERAL:

Diseñar una tarjeta que sirva de conexión al computador para ser usado como herramienta en la creación de futuros laboratorios remotos educativos, para las asignaturas disciplinares de la Licenciatura en electrónica de la universidad pedagógica nacional.

6. OBJETIVOS ESPECÍFICOS:

- Crear el hardware que permita conectar físicamente los implementos que use el estudiante directamente al computador y muestre los datos en una interfaz del mismo.
- Evaluar los componentes electrónicos más propicios para la realización de la tarjeta.
- Desarrollar el protocolo de comunicación para que se realice la interacción entre la tarjeta y el interfaz gráfico en el computador.
- Implementar el hardware creado para perfeccionar el laboratorio remoto y poder brindar una alternativa en la ejecución de los mismos.

7. MARCO TEÓRICO

7.1 Hardware:

Es un conjunto de componentes físicos y/o materiales que son añadidos a un dispositivo electrónico. (Fundamentos de Informática, 2013)

En la electrónica el hardware con el software hacen de un elemento vital que hace que cualquier dispositivo, mecanismo o maquina realice varias tareas de una manera automática, en este caso, el hardware es la parte física (tangible), en cambio el software es la parte que no se puede tocar, es decir, el hardware es el cuerpo y el software la esencia.

En la parte computacional, podemos decir que es un conjunto de elementos electrónicos que están interconectados entre sí para cumplir ciertas funciones, también se consideran los dispositivos periféricos que se conectan externamente para realizar una función específica de cada aparato (mouse, teclado, pantalla, etc.).

7.2 Raspberry Pi:

El Raspberry Pi 2 es un micro ordenador personal que permite interactuar con circuitos externos, produciendo una proyección de alta calidad en la pantalla, y soporta varios componentes necesarios en un computador común, permitiendo la inserción de programas para facilitar su implementación de diversas maneras. Permite su ejecución en diversos ámbitos, convirtiéndolos en un centro multimedia.

Estos ámbitos pueden ser:

- El hogar: facilitando la interacción familiar y social, ya que se presta para ser configurada como una consola de video, ya sea de índole reproductora o de juego.

- La escuela: promoviendo el desarrollo académico, pues permite que funcione como procesador y reproductor de video, lo cual favorece su implementación en el aula.

En cualquiera de estos entornos, se facilita su implementación y uso ya que también cuenta con otro tipo de conexiones para ser implementadas en otro tipo de tecnologías, promoviendo el desarrollo emocional y cognitivo, los cuales son favorables en la interacción del niño con la máquina.

Es un dispositivo de 10 cm de longitud aproximadamente que se puede conectar a un televisor o un teclado. Soporta varios componentes para ser un ordenador como tal, puede ser usado para varias cosas que un pc puede realizar, tales como hojas de cálculo, procesadores de texto y algunos juegos.

Anteriormente era más pequeña, tiene varios puertos y entradas, como de USB y otro salida HDMI. Con estos puertos permiten interconectar con otros dispositivos como lo son teclados, pantallas, etc.

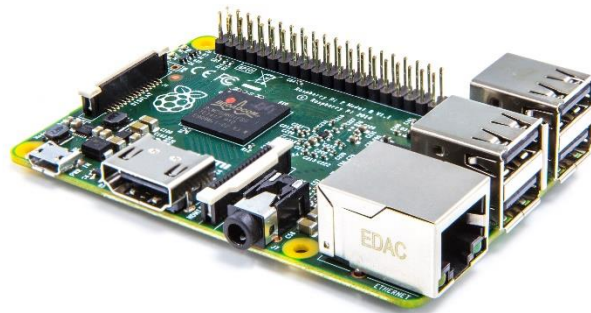


Figura 1. Raspberry Pi 2 model B (Raspberry Pi 2016)

7.3 Python:

Python es un lenguaje de programación de alto nivel similar a C, C++, Perl y Java, pero con una sintaxis muy limpia favoreciendo un código legible. Se trata de un lenguaje interpretado o de script, fuertemente tipado, multiplataforma y orientado a objetos. (Raúl González Duque, 2010)

Las características que hace que Python sea un lenguaje sencillo y muy rápido son: (Raúl González Duque, 2010):

- Su sintaxis es simple, clara y sencilla
- El tipado dinámico.
- Eficiente gestor de memoria
- La gran cantidad de librerías disponibles y la potencia del lenguaje

Python es considerado un lenguaje interpretado porque se ejecuta utilizando un programa intermedio por una interpretadora. Una interpretadora lee un programa de alto nivel y lo ejecuta, lo que significa que lleva a cabo lo que indica el programa. Traduce el programa poco a poco, leyendo y ejecutando cada comando. (Allen Downey, Jerrey Elkner, Chris Meyers, 2002)

7.4 Motor DC:

También llamados máquinas de corriente continua, posee un fácil control de posición y velocidad lo hace una de las mejores opciones para controlar y automatizar sus procesos.

Con la llegada de la electrónica se disminuyó su uso, puesto que los motores de corriente alterna, también son controlados de la misma forma pero estos motores son mucho más

económicos en el mercado. De igual manera, los motores de corriente continua se sigue utilizando en trenes, tranvías, en cuanto a la precisión, se usa para máquinas, micro motores, etc.

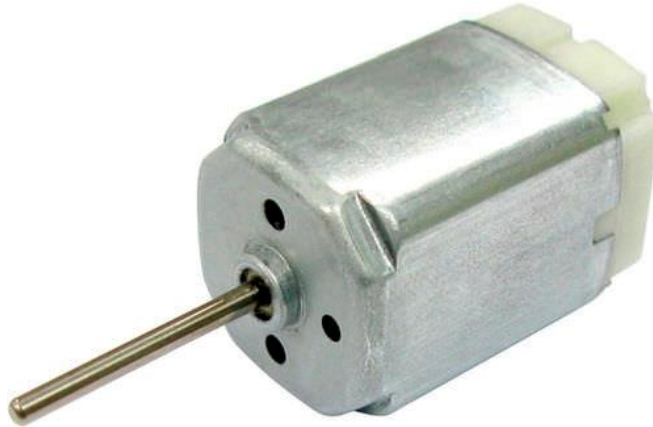


Figura 2. Motor DC (Tomado de <http://www.directindustry.es/prod/johnson-electric/product-665-1217079.html>)

Un motor DC está hecho con imanes y bobinas electromagnéticas, que producen un campo electromagnético. Cuando la corriente pasa a través de la bobina o el alambre, que está ubicada entre los polos norte y sur del imán, el campo que se genera por la armadura interactúa con el campo del imán y genera torsión.

7.5 Motor paso a paso:

Es un dispositivo electromecánico que convierte unos impulsos eléctricos en desplazamientos angulares discretos, en comparación a un motor normal (gira de forma continua), es capaz de avanzar ciertos grados (o pasos) a la vez, esto depende del estado de las entradas de control. Un motor paso a paso tiene un comportamiento igual a un convertidor digital-analógico y es dirigido por impulsos que proceden de sistemas lógicos como microcontroladores y/o ordenadores.



Figura 3. Motor pasó a paso (Tomada de <http://www.dx.com/es/p/28ybj-48-4-phase-5-wires-stepper-motor-blue-silver-dc-5v-184865#.V64tePnhDIU>)

Estos motores son muy buenos para la construcción de mecanismos donde se necesiten movimientos muy precisos. Existen 3 tipos de motores paso a paso, motor de reluctancia variable, motor de magnetización permanente y el motor paso a paso híbrido.

La característica principal de los motores paso a paso es el movimiento de un paso a la vez por cada pulso que se le aplique. Estos pasos pueden variar desde 90 grados hasta movimientos de 1.8 grados, es decir, en 4 pasos para los 90 grados.

7.6 Servomotor:

Un servomotor es un dispositivo que tiene un eje de rendimiento controlado. Puede ser llevado a posiciones angulares específicas al enviar una señal codificada, la señal codificada existe en la línea de entrada, el servomotor mantendrá la posición angular del engranaje, cuando la señal codificada cambia la posición angular de los piñones cambia. Algunos servos son usados para posicionar superficies de control como el movimiento de palancas, ascensores pequeños, etc. Estos motores son usados en radio controles, títeres y robots.

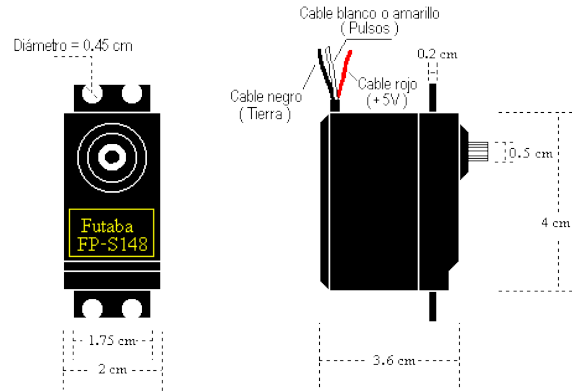


Figura 4. Servomotor (Tomada de <http://www.info-ab.uclm.es/labelec/solar/electronica/elementos/servomotor.htm>)

Los servo motores son pequeños tiene internamente circuitos de control interno y es muy poderoso a pesar de su tamaño. Un servo normal tiene 42 onzas por pulgada o mejor 3 kg por centímetro. En cuanto al torque, es muy fuerte para su dimensión, además entrega una potencia proporcional para cargas mecánicas. No consumen mucha energía, de acuerdo a su composición en el interior se puede observar un juego de piñones, los circuitos de control y la caja, posee tres alambres de conexión externa, uno es para su alimentación de 5 voltios (VCC), el otro es la conexión a tierra (GND), por último el alambre sobrante es el cable de control.

7.7 Sensor Ultrasonido:

Los sensores de ultrasonido tienen una frecuencia mayor que la máxima audible por el oído de las personas. Inician desde los 16 Hz y su límite es superior a los 20 KHz, en cambio se usara un sonido con una frecuencia de 40 KHz, a esta clase de sonido se les conoce como ultrasonido. Su funcionamiento básico es la medición de distancia que se muestra de manera clara en el la siguiente figura, se tiene un receptor que emite un pulso

ultrasonico que rebota sobre un determinado objeto y la reflexión de aquel pulso se detecta por un receptor de ultrasonidos.

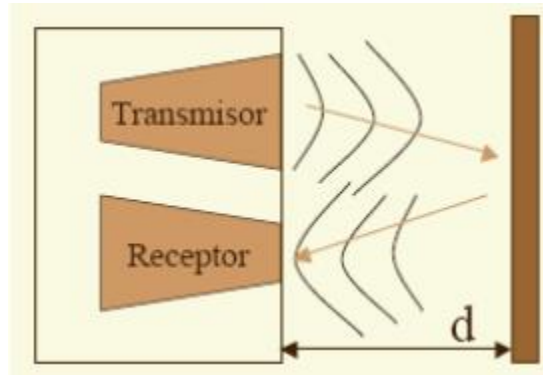


Figura 5. Grafica del Sensor Ultrasonido (Tomado de <http://pacific.fi-p.unam.mx/cursos/sensor%20ultra/sensor.html>)

La gran mayoría de estos sensores tienen un bajo costo debido a que se basan en la emisión de un pulso de ultrasonido cuyo campo de acción es de forma cónica, con este mide el tiempo que transcurre entre la emisión del sonido y la recepción del eco, en el cual se puede establecer una cierta distancia a en la que se encuentra el objeto que ha causado la reflexión de la onda sonora, mediante la siguiente formula:

$$d = \frac{1}{2} V * t$$

Donde V es la velocidad del sonido en el aire y t es el tiempo transcurrido entre la emisión y recepción del pulso.



Figura 6. Sensor Ultrasonido (Tomado de <http://www.kurtsik.org/blog/?p=1212>)

Los sensores ultrasónicos usan un método de medición con una tecnología muy compleja, y solo se emplean como último recurso, como si fuera una solución para aplicaciones especialmente difíciles.

Estos sensores son usados en diversos campos como lo son:

- Herramientas para maquinaria (ingeniería mecánica)
- Alimentación y bebidas
- Trabajos en madera y mobiliario
- Materiales de construcción
- Agricultura
- Diseño
- Pasta y papel
- Manipulación de materiales
- Medición de nivel

7.8 Puente H:

Es un circuito electrónico que permite a un motor DC girar en ambos sentidos, avanzar o retroceder.

Los puentes H ya vienen con algunos circuitos integrados, pero se puede diseñar o construir a partir de componentes más sencillos y económicos.

Generalmente un puente H es construido con 4 interruptores (mecánicos o transistores). De acuerdo a esos interruptores se aplica una tensión positiva en el motor, haciéndolo girar en una sola dirección o invirtiendo el sentido al motor

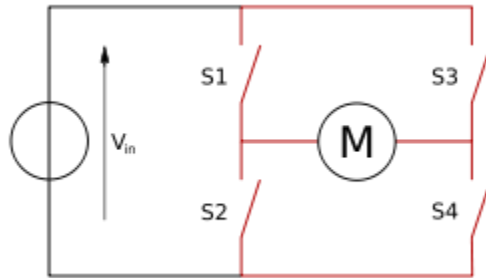


Figura 7. Circuito puente H (Tomado

<http://computointegradoits.blogspot.com.co/2012/05/puente-h.html>)

Un puente H se usa para invertir o frenar de manera brusca el giro de un motor, además puede ser usado para que el motor frene con su propia inercia, cuando es desconectado el motor de la fuente que lo alimenta.

Básicamente se puede hacer esto tomando en cuenta las S_n

S1	S2	S3	S4	Resultado
1	0	0	1	Avanza
0	1	1	0	Retrocede
0	0	0	0	Se detiene bajo inercia propia
1	0	1	0	Frenado brusco

Tabla 1. Configuración de las S_n de un puente H (Tomado de

<http://blutintegrado.blogspot.com.co/2012/05/puente-h.html>)

La forma más sencilla de realizar un puente H es usando transistores, debido a que su tiempo de vida y frecuencia de conmutación son más altas. Cuando se usan convertidores de potencia es muy importante usar interruptores mecánicos.

Los interruptores vienen acompañados de diodos que permite a la corriente circular en sentido inverso al sospechado cada vez que se conmute la tensión puesto que el motor está compuesto por bobinas que durante algunos espacios de tiempo se opondrán a que la corriente varié.

7.9 Protocolo SSH:

Es un protocolo que simplifica las comunicaciones seguras entre dos sistemas usando una arquitectura cliente/servidor permitiendo a los usuarios conectarse a un host remotamente. En contraste de otros protocolos de comunicación tales como FTP o Telnet, SSH encripta la sesión de conexión, haciendo imposible que alguien pueda entrar o hackear contraseñas no encriptadas.

Este protocolo fue diseñado para reemplazar los métodos más antiguos para registrarse remotamente en otro sistema a través de la Shell de comando, como telnet o rsh. Un programa relacionado, el scp, sustituye otros programas diseñados para copiar archivos entre hosts como rcp. Estas aplicaciones arcaicas no encriptan contraseñas entre el cliente y el servidor, evita usarlas mientras le sea posible. El uso de otros métodos para el registro remoto a otros sistemas que reduce los riesgos de seguridad tanto para el sistema cliente como el sistema remoto.

Este protocolo proporciona los siguientes tipos de seguridad:

- Después de la conexión inicial, el cliente puede verificar que se está conectando al mismo servidor al que se conectó anteriormente.
- El cliente transmite su información de autenticación al servidor usando una encriptación robusta de 128 bits.
- Todos los datos enviados y recibidos durante la sesión se transfieren por medio de encriptación de 128 bits, lo cual los hacen extremadamente difícil de descifrar y leer.
- El cliente tiene la posibilidad de reenviar aplicaciones X11 desde el servidor. Esta técnica, llamada *reenvío por X11*, proporciona un medio seguro para usar aplicaciones gráficas sobre una red.

8. METODOLOGÍA

En el siguiente proyecto se divide en varias etapas, siendo fundamental el desarrollo del software en el cual permite el control de los elementos usados, donde se puede evidenciar un proceso en cada una de las etapas del desarrollo de la tarjeta. La primera etapa es la recopilación de la información, donde se recolecta y establece las necesidades del estudiante. La segunda etapa es el diseño y desarrollo de la tarjeta, en la que se realiza la programación de los componentes escogidos que son los más propicios para la elaboración de la tarjeta.

Por último la implementación de la tarjeta para verificar la funcionalidad del producto obtenido.

A continuación se describen las etapas de este proyecto:

4. **Análisis:** En esta etapa se analizan los requerimientos que son necesarios para la elaboración de la tarjeta, donde se verifica los materiales necesarios para la creación del hardware; y así evaluar y analizar la información obtenida para su respectiva construcción.
5. **Diseño y construcción:** En esta etapa se toma en cuenta las necesidades del estudiante frente a la tarjeta para su respectivo desarrollo, con los materiales ya definidos para el desarrollo y el manejo de los motores, con una adecuada comunicación con el computador.
- ✓ **Comunicación con el pc:** El estudiante para poder controlar el motor que desee (servomotor, paso a paso o DC), primero que todo tiene que tener ciertos programas instalados (posteriormente se explica en detalle cuales son), usando así el protocolo de comunicación SSH desarrollado para esta tarjeta permitiendo una manipulación adecuada de los motores antes mencionados.

6. **Implementación:** En esta etapa se realizan las pruebas, verificando el funcionamiento correcto de la tarjeta, donde se analiza las necesidades requeridas para así poder satisfacer lo que el usuario vaya a usar.

8.1 Análisis:

Analizando las asignaturas de la licenciatura en Electrónica de la Universidad Pedagógica Nacional en la cual se usan motores, se escogen tres clases de motores que generalmente son de mayor uso, de los cuales son los siguientes:

- Motor DC
- Servo Motor
- Motor paso a paso

Adicionalmente en algunas de las materias se usan sensores como de proximidad, temperatura, luz, acústico, etc. El sensor que se escogió es el Ultrasonido que es un sensor de proximidad que detecta objetos desde pocos centímetros hasta varios metros. Se escogió porque este emite un sonido y mide el tiempo en el que la señal tarda en regresar, estos los reflejan en un objeto, el sensor recibe el eco producido y lo convierte en señales eléctricas.

Por lo tanto lo que se va a controlar son 4 elementos básicos pero importantes en las asignaturas de la licenciatura.

Teniendo claro cuáles son los elementos a controlar, el siguiente paso es analizar la construcción de la tarjeta, con el fin de poder cumplir con los objetivos propuestos.

Los requisitos mínimos que se necesitan para el correcto funcionamiento de la tarjeta son los siguientes:

- Raspberry pi2 model B
- Python 2.7.4

- Puente H con L298
- Driver ULN2003
- Computador con puerto Ethernet
- Sistema operativo Windows 7, 8, 8.1 o 10
- Cable Ethernet RJ45
- Fuente de Voltaje 12 V
- Adaptador de 5 V

De acuerdo a las pruebas hechas con arduino, pic y raspberry se decidió que el componente principal para hacer la tarjeta será la Raspberry Pi 2, debido a que este elemento es el más adecuado para el control y conexión con los motores, viene con un procesador de 900 MHz de 4 núcleos, con una memoria RAM de 1 GB, trabaja con el sistema operativo Linux pero también puede operar con Windows.

8.2 Diseño y Construcción:

Para realizar el control del motor DC, se usa un módulo puente H L298N es una tarjeta construida al circuito integrado L298N, el cual tiene en su interior de 2 puentes H independientes con una capacidad de soportar 2 amperios y posee dos terminales para conectar hasta cuatro motores al mismo tiempo.

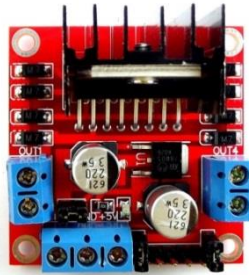


Figura 8. Puente H con L298 (Tomado de <https://www.linio.com.co/p/puente-h-l298-driver-l298-motor-cd-para-arduino-microchip-rojo-tmsj8e>)

Para realizar el control del motor paso a paso se usa un driver ULN2003 que consta de un arreglo de 7 transistores NPN Darlington de alto voltaje y soporta altas corrientes, los transistores en configuración Darlington son de colector abierto y tienen emisores comunes. Cada canal puede soportar hasta 500 mA. Y picos de 600 mA., además tiene diodos de supresión para cargas inductivas, incluye leds para indicar el estado de las salidas.

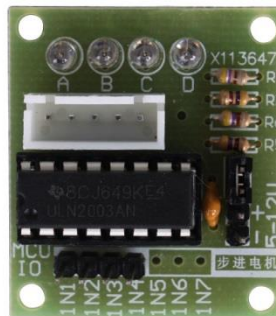


Figura 9. Driver con ULN2003 (Tomado de <http://www.icmasteronline.com/5v-stepper-motor-with-uln2003-driver-252818707>)

Para el servomotor y el sensor ultrasonido no fue necesario usar ningún puente H debido a que se alimentan con un voltaje de 5 voltios proporcionado por la tarjeta raspberry pi 2, esta misma entrega una corriente necesaria dependiendo del elemento que se le conecte.

El raspberry pi 2 consume 5 voltios y 900 mA. por lo que la alimentación que brinda el adaptador debe cumplir con esas condiciones, además posee 40 pines que pueden ser configurados como salidas o entradas, estas conexiones se llaman GPIO.

La raspberry tiene la posibilidad de conectar una cámara y/o un display de pantalla de cristal líquido pero para este proyecto no es necesario utilizarlas.


Features	Raspberry Pi (Model B+)
	
SoC	Broadcom BCM2835
Main processor	700 MHz ARM1176JZF-S single-core CPU (no FPU, no SIMD/DSP) L1: 16 KB I-cache + 16 KB D-cache L2: 128 KB (shared with GPU)
Graphics	Broadcom VideoCore IV
Dedicated video hardware	MPEG-2, VC-1, H.264, MPEG-4 AVC decoder (1080p at 30 fps) MPEG-2, VC-1, H.264, MPEG-4 AVC encoder (1080p at 30 fps)
Memory	512 MB (shared with GPU)
Camera	15-pin MIPI camera interface (CSI) connector
Onboard storage	No internal onboard storage 1 x MicroSD
USB 2.0 ports	4 x USB device (via the on-board 5-port USB hub)
Onboard network	10/100 Mbit/s Ethernet (8P8C) USB adapter
Low-level peripherals	17 x GPIO, SPI bus with two chip selects, UART, I ² C, I ² S audio
Size	85.60 mm x 56 mm

Tabla 2. Características Raspberry Pi 2 model b (Tomado de <http://comohacer.eu/comparativa-y-analisis-raspberry-pi-vs-competencia/>)

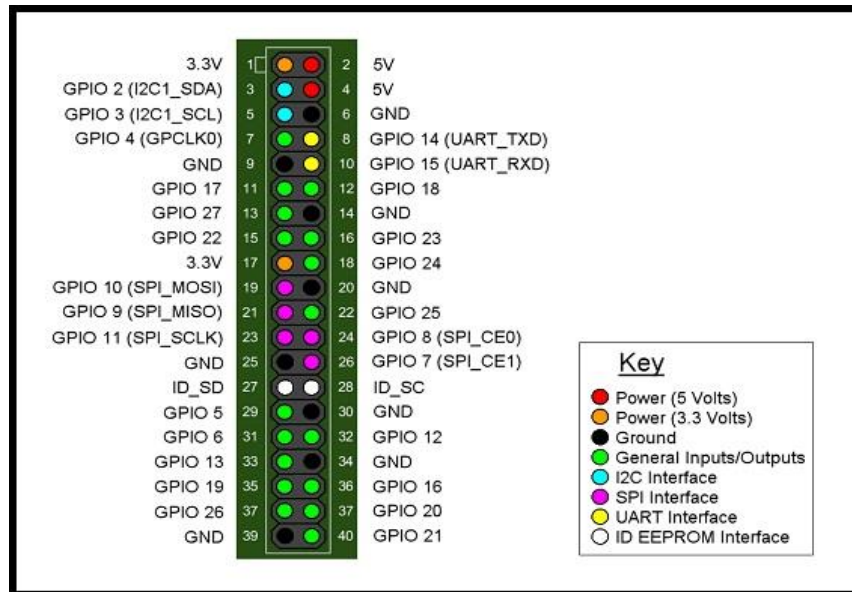


Figura 10. Pines GPIO Raspberry Pi 2 Model B (Tomado de <http://enraged.es/wordpress/>)

8.2.1 Pines GPIO:

Los pines GPIO (pines de entrada o salida digital) son la manera en la cual la raspberry pi 2 puede interactuar con los motores o el sensor. La tensión de trabajo del puerto es de 3.3 v configurado en alto y de 0 v para un estado en bajo y solo puede suministrar una corriente máxima de 16 mA. Todos los pines GPIO pueden ser controlados por medio del lenguaje de programación Python.

Se usaron 19 pines, algunos configurados como VCC o GND, los otros como entradas a los motores y sensor respectivamente.

En el control del motor DC, se asignaron los GPIO 4, GPIO17, GPIO 27 y GPIO 22 como salidas que van conectadas al puente H, que sirve de protección para la raspberry, los pines GPIO 23 y GPIO 24 son los que habilitan el puente para efectuar el debido control que se necesita para el motor anteriormente mencionado, la razón por la cual se usa este puente H

es porque el motor DC necesita una alimentación de 12 v, por lo tanto es sumamente necesario tener protección para la raspberry pi 2 model b.

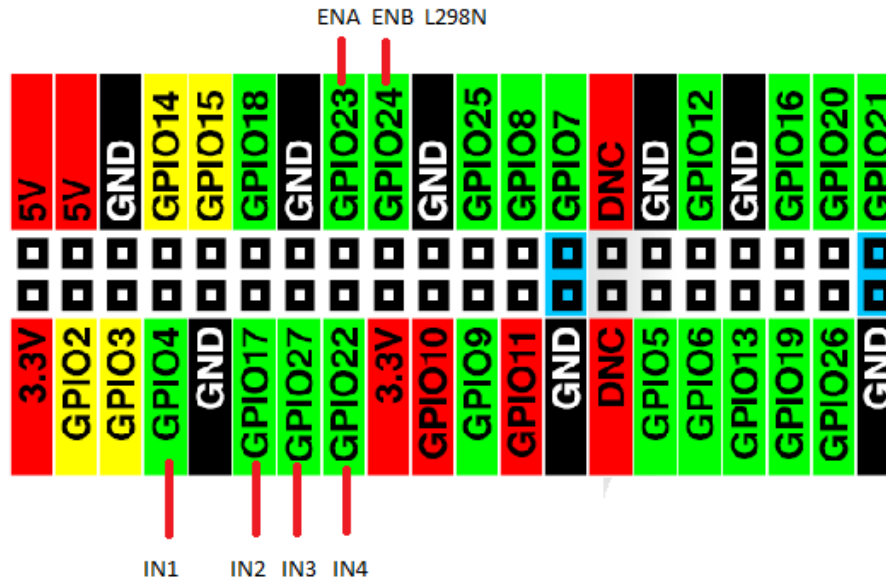


Figura 11. Pines Raspberry pi 2 model b conectado a Puente H L298 (Elaborado por el autor)

Para el control del motor paso a paso, se asignaron los GPIO 5, GPIO 6, GPIO 12 y GPIO 13, estas salidas van conectados a los pines IN1, IN2, IN3 e IN4 respectivamente del driver ULN2003 y a su vez las salidas de este optocoplador están conectados al motor paso a paso cabe aclarar que este driver tiene una alimentación de 5 v que se conecta directamente de la raspberry permitiendo un control más efectivo hacia el motor. Cuando se esté efectuando el control del motor el driver posee unos Leeds, es decir, cuando se mueve el motor estos alumbran de lo contrario no prenden cuando no es posible el movimiento del mismo.

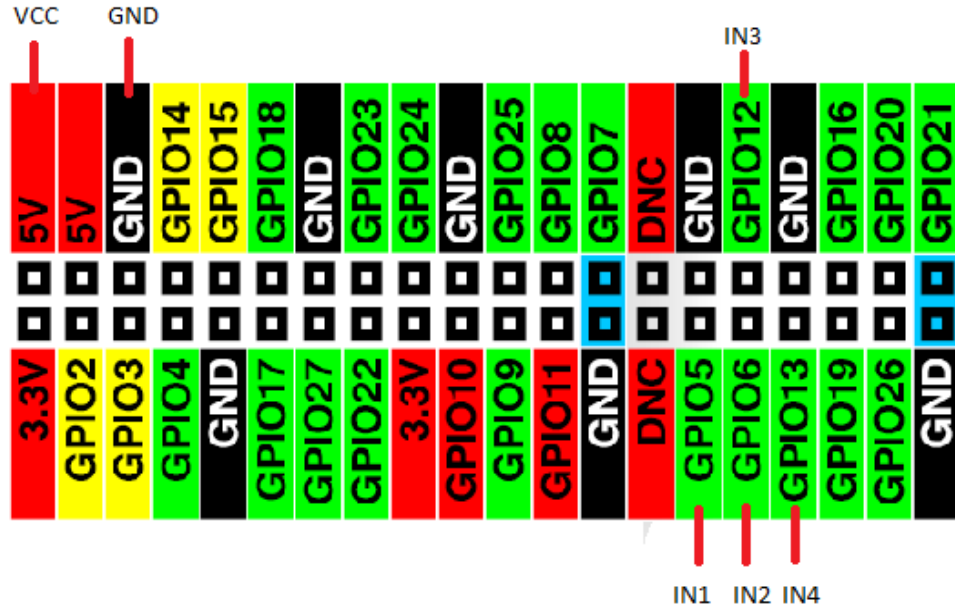


Figura 12. Pines Raspberry pi 2 model b conectado a Driver ULN2003 (Elaborado por el autor)

Con el servo motor, no es necesario un puente H debido que su alimentación es de 5v lo cual puede ser conectado directamente desde la raspberry. Se usa el GPIO 21 como salida para efectuar el control del motor, también se le conecta el pin del VCC y el pin de GND.

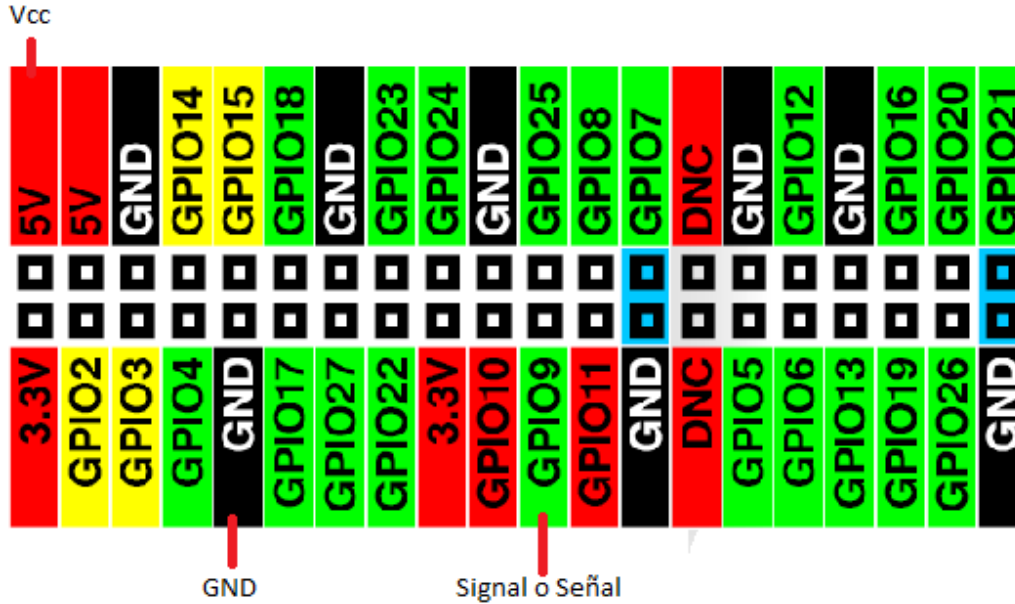


Figura 13. Pines Raspberry pi 2 model b conectado a Servomotor (Elaborado por el autor)

En cuanto al sensor ultrasonido, tampoco es necesario usar un puente H, porque es alimentado por 5 v que es conectado directamente al pin Vcc de la raspberry, el pin TRIG del sensor se conecta al GPIO 20, mientras que el GPIO 21 se conecta a una resistencia y este directamente al pin ECHO del sensor, por último el pin GND del sensor se conecta al pin GND de la raspberry.

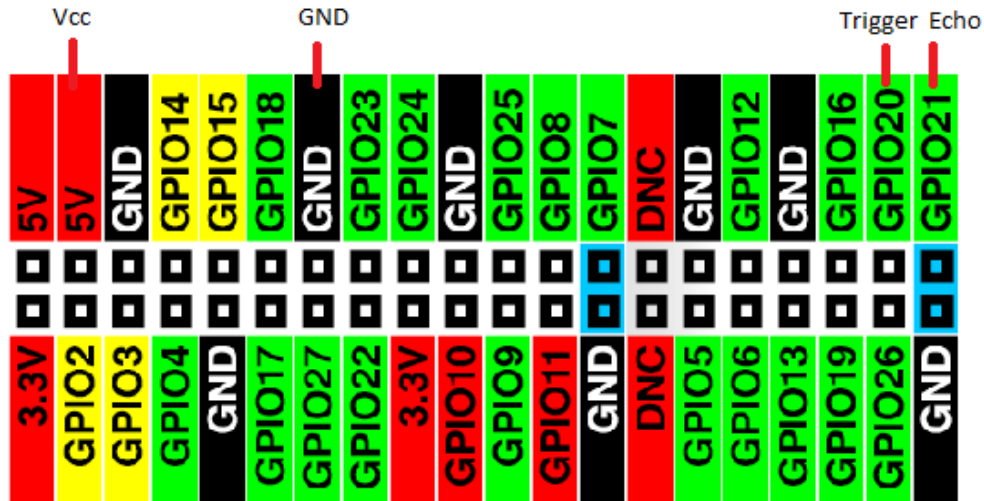


Figura 14. Pines Raspberry pi 2 model b conectado al Sensor Ultrasonido *(Elaborado por el autor)*

8.2.2 Protocolo de Comunicación SSH:

SSH (Secure Shell) es un protocolo de comunicación que permite la conexión remota con diferentes servidores o tipos de máquinas siempre y cuando lo soporten, según en la página del creador algunos países está prohibido usar ese protocolo, debido a que prácticamente es la entrada para hackear servidores. En este caso, será utilizado de manera local para este proyecto y es muy usado en varios lugares del mundo. Este programa permite hacer conexiones seguras de una forma remota a la Raspberry Pi 2 model B, es decir, no será necesario usar una segunda pantalla, teclado o mouse ya que se podrá operar remotamente la Raspberry.

La conexión se hace directamente entre el Raspberry al computador mediante un cable de red RJ45. La velocidad de conexión es de hasta 100 Mbps, para hacer dicha conexión se necesitan los siguientes materiales:

- Computador o portátil con Windows 7, 8 o 10

- Puerto de red libre en el portátil o PC
- Cable de red RJ-45
- Raspberry Pi 2 Model b

8.2.2.1 Configuración de Windows 7 u 8

Se conecta el conector RJ -45, las luces del puerto de red tiene que encender tanto del PC como en el Raspberry Pi, cabe aclarar que algunas tarjetas de red en los PCs no cuentan con LED que indiquen la respectiva conexión.

Primero que todo tenemos que configurar el centro de redes y recursos compartidos del PC, para esto realizamos los siguientes pasos:

- Es necesario compartir la conexión Wi-Fi o la conexión Ethernet que se usa para acceder a la red, para esto se necesita activar el uso compartido de internet en las dos interfaces disponibles, se hace clic derecho en el icono de red (📶 o 🖨️). Luego abrimos el centro de redes y recursos compartidos.

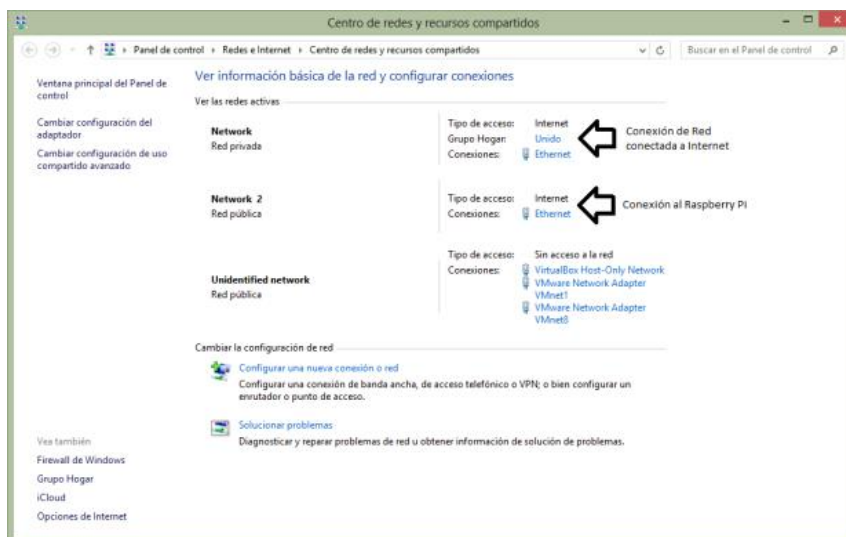


Figura 15. Centro de redes y recursos compartidos de Windows 7(Elaborado por el autor)

- Se identifica y selecciona la conexión primaria a internet y hacer clic sobre el link al frente de conexiones. Aquí se abre la ventana de “estado de Ethernet”. Después se le hace clic en el botón propiedades, se despliega una ventana, damos clic en la pestaña llamada “Uso compartido” y activamos la selección “permitir que todos los usuarios de otras redes se conecte a través de la conexión a internet de este equipo”. Luego en la opción de selección buscamos conexión de red que tenga el Raspberry Pi que debe ser la del puerto Ethernet que está libre, por ejemplo aquí se llama “Ethernet 2”.

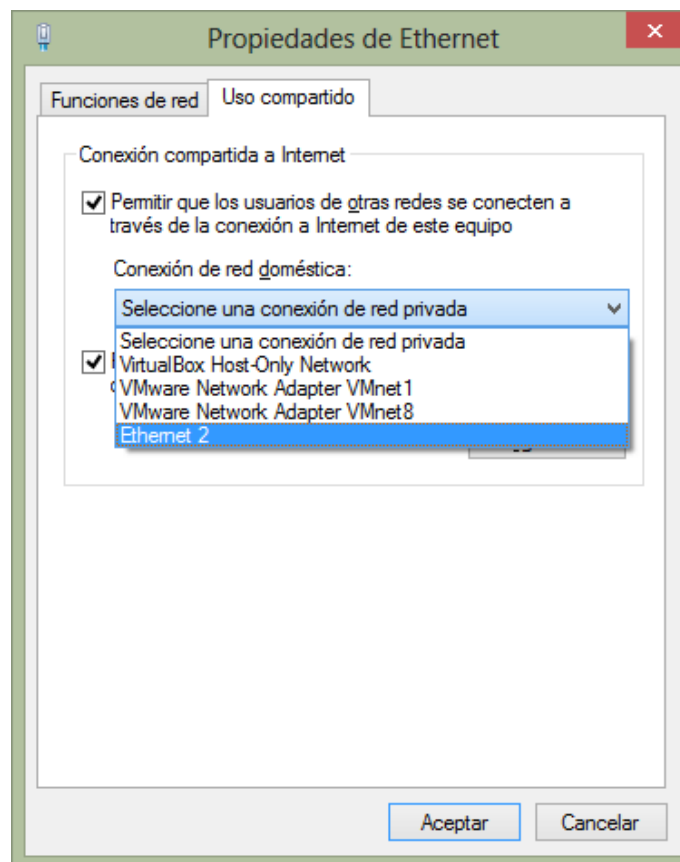


Figura 16. Propiedades de la ventana Ethernet (Elaborado por el autor)

- Se configura la dirección estática del PC en el puerto en el que se encuentra conectado el Raspberry, se repite el primer paso, ahora seleccionamos el puerto donde la

raspberry está conectada, como esta en la figura anterior el puerto “Ethernet 2”. Hacemos clic en el botón propiedades, luego seleccionamos “esta selección usa los siguientes elementos” buscamos la sección “Protocolo de Internet versión 4”, seleccionamos este campo y luego clic en propiedades. En este caso, como ejemplo se asigna la red 192.168.137.1 a este puerto de conexión, pero en el hardware de todo este proyecto se usó la red 192.168.137.156.

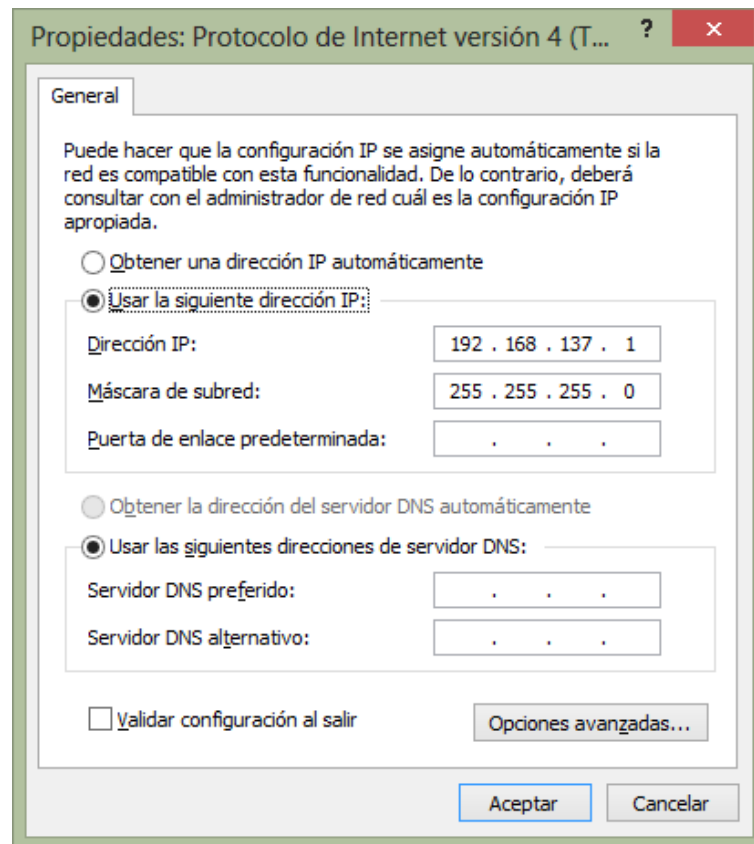


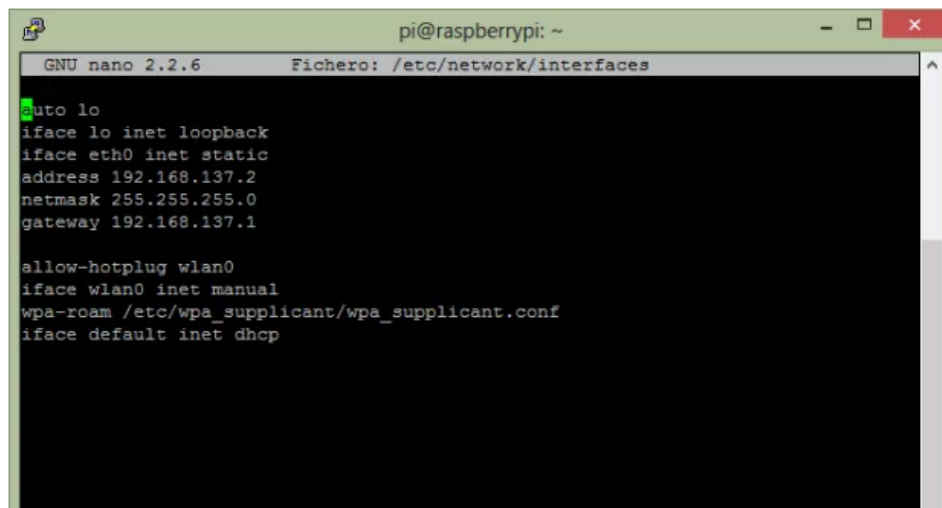
Figura 17. Configuración Dirección IP del Computador *(Elaborado por el autor)*

8.2.2.2 Configuración dirección estática del Raspberry Pi:

Ingresamos a la línea de comando de Linux. Una vez allí, ingresamos el siguiente comando, el cual abrirá una ventana de edición para configurar la dirección estática del Raspberry Pi.

Sudo nano/ etc/network/interfaces

- Cuando ya esté abierto el archivo se necesita modificar la tercera columna dhcp por static y asignamos la dirección IP 192.168.137.2 al dispositivo. Para que queden de las mismas características que la configuración hecha en el PC, los valores deber ser iguales a la siguiente imagen.



```

pi@raspberrypi: ~
GNU nano 2.2.6 Fichero: /etc/network/interfaces
auto lo
iface lo inet loopback
iface eth0 inet static
address 192.168.137.2
netmask 255.255.255.0
gateway 192.168.137.1

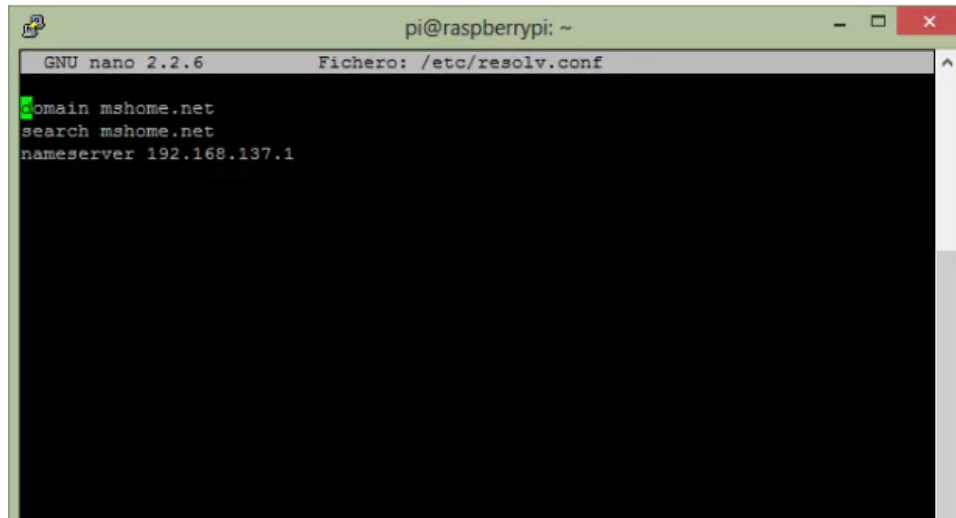
allow-hotplug wlan0
iface wlan0 inet manual
wpa-roam /etc/wpa_supplicant/wpa_supplicant.conf
iface default inet dhcp
  
```

Figura 18. Características desde la configuración hecha en el Pc (*Elaborado por el autor*)

- Este paso es opcional. Pero sirve para cerciorarse que toda la configuración está de acuerdo a la configuración deseada. Editamos el fichero /etc/resolv.conf y verificamos que el name server es la misma dirección IP que el que asignamos al PC (en este caso usamos la dirección IP 192.168.137.1 pero en el proyecto del hardware

asignamos la dirección IP 192.168.137.156). Así como se muestra en la siguiente imagen. Para guardar las modificaciones hechas, usamos este comando

```
Sudo nano /etc/resolv.conf
```

A screenshot of a terminal window on a Raspberry Pi. The window title is "pi@raspberrypi: ~". The terminal shows the GNU nano 2.2.6 editor editing the file /etc/resolv.conf. The content of the file is: domain mshome.net, search mshome.net, and nameserver 192.168.137.1. The cursor is at the end of the first line.

```
pi@raspberrypi: ~
GNU nano 2.2.6 Fichero: /etc/resolv.conf
domain mshome.net
search mshome.net
nameserver 192.168.137.1
```

Figura 19. Ventana que comprueba que las modificaciones fueron guardadas

(Elaborado por el autor)

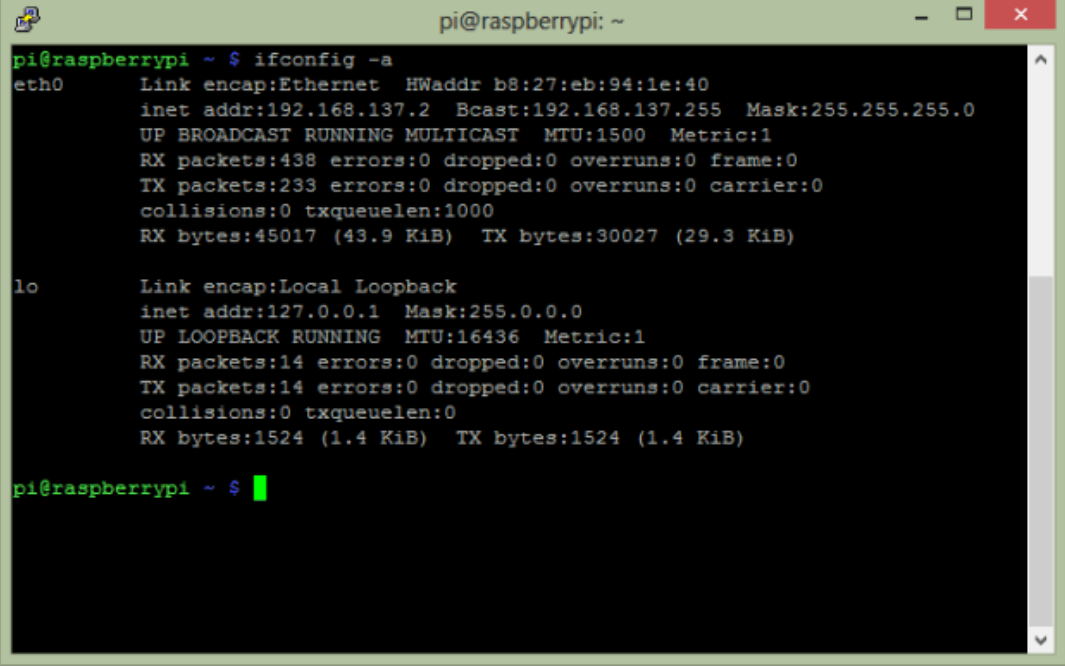
- Reiniciamos el raspberry Pi para que los cambios hagan efecto. Para esto usamos el siguiente comando:

```
Sudo reboot
```

8.2.2.3 Verificar la dirección IP en el Raspberry Pi

Después de reiniciar el raspberry Pi es necesario verificar si las modificaciones hechas hicieron efecto, se necesita verificar la dirección IP asignada al dispositivo, por lo tanto se usa el siguiente comando:

```
Ifconfig -a
```



```

pi@raspberrypi ~ $ ifconfig -a
eth0      Link encap:Ethernet  HWaddr b8:27:eb:94:1e:40
          inet addr:192.168.137.2  Bcast:192.168.137.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:438 errors:0 dropped:0 overruns:0 frame:0
          TX packets:233 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:45017 (43.9 KiB)  TX bytes:30027 (29.3 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:14 errors:0 dropped:0 overruns:0 frame:0
          TX packets:14 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:1524 (1.4 KiB)  TX bytes:1524 (1.4 KiB)

pi@raspberrypi ~ $ █

```

Figura 20. Verificación de la Dirección Ip en la Raspberry *(Elaborado por el autor)*

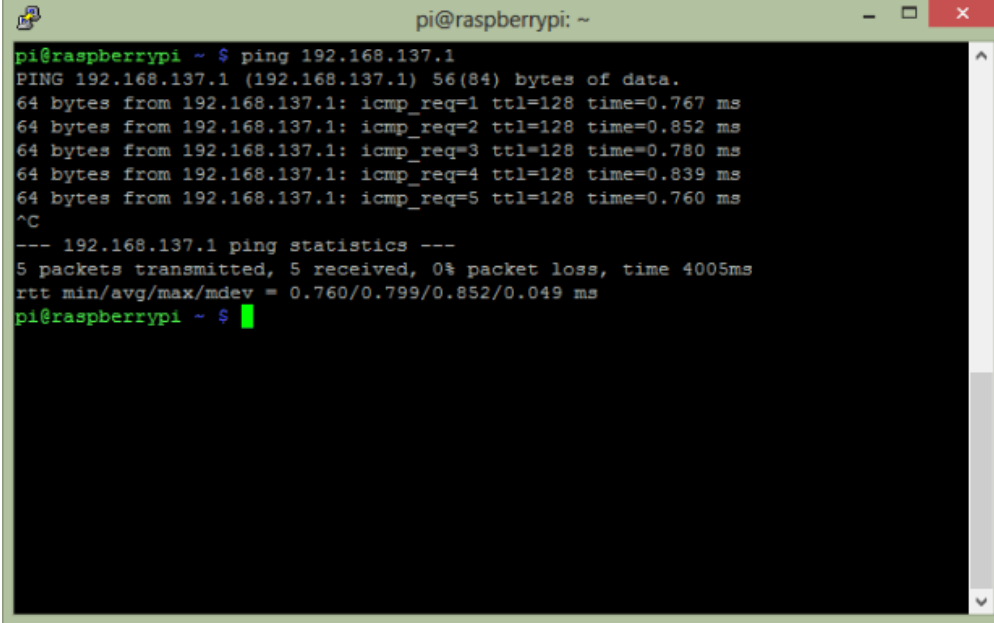
8.2.2.4 Verificar si la conexión fue exitosa:

Para verificar si la configuración ha sido exitosa, realice un ping desde la dirección IP del Raspberry Pi a la dirección del computador. En este proyecto, la dirección que se asignó al Raspberry es de 192.168.137.2 y la del PC es 192.168.137.156

Usamos el siguiente comando en el Raspberry para verificar la conexión, si esta es exitosa debe devolver algo muy parecido a lo que se muestra:

Ping 192.168.137.156

Después de recibir 4 o 5 paquetes de datos con éxito, oprima Ctrl+c para cancelar, si todo fue un éxito debe aparecer un resultado como este:

A screenshot of a terminal window titled "pi@raspberrypi: ~". The terminal shows the execution of a ping command: "pi@raspberrypi ~ \$ ping 192.168.137.1". The output displays five successful ping responses, each showing 64 bytes of data, an ICMP request number, a TTL of 128, and a response time in milliseconds. The response times are 0.767 ms, 0.852 ms, 0.780 ms, 0.839 ms, and 0.760 ms. After the fifth response, the user presses the Ctrl-C key (indicated by "^C"). The terminal then displays the ping statistics: "--- 192.168.137.1 ping statistics ---", "5 packets transmitted, 5 received, 0% packet loss, time 4005ms", and "rtt min/avg/max/mdev = 0.760/0.799/0.852/0.049 ms". The prompt "pi@raspberrypi ~ \$" is visible at the bottom of the terminal window.

```
pi@raspberrypi ~ $ ping 192.168.137.1
PING 192.168.137.1 (192.168.137.1) 56(84) bytes of data.
64 bytes from 192.168.137.1: icmp_req=1 ttl=128 time=0.767 ms
64 bytes from 192.168.137.1: icmp_req=2 ttl=128 time=0.852 ms
64 bytes from 192.168.137.1: icmp_req=3 ttl=128 time=0.780 ms
64 bytes from 192.168.137.1: icmp_req=4 ttl=128 time=0.839 ms
64 bytes from 192.168.137.1: icmp_req=5 ttl=128 time=0.760 ms
^C
--- 192.168.137.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4005ms
rtt min/avg/max/mdev = 0.760/0.799/0.852/0.049 ms
pi@raspberrypi ~ $
```

Figura 21. Verificación de que la conexión fue exitosa *(Elaborado por el autor)*

Al realizar el ping desde el PC, es necesario ir al cmd. Para esto se oprime la tecla Windows+R y en el espacio escribir cmd y luego enter. Después de esto, se abre una nueva ventana de línea de comandos. Luego se realiza un ping hacia la dirección IP del raspberry, en este caso es 192.168.137.2, si la conexión es todo un éxito, se espera el siguiente resultado, donde no hay paquetes perdidos.

```

C:\WINDOWS\system32\cmd.exe
C:\>ping 192.168.137.2

Haciendo ping a 192.168.137.2 con 32 bytes de datos:
Respuesta desde 192.168.137.2: bytes=32 tiempo=1ms TTL=64
Respuesta desde 192.168.137.2: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.137.2: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.137.2: bytes=32 tiempo<1m TTL=64

Estadísticas de ping para 192.168.137.2:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (<0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 1ms, Media = 0ms

C:\>_

```

Figura 22. Al hacer ping se comprueba que no se pierde ningún dato

(Elaborado por el autor)

Después de haber hecho todos estos pasos, ya tenemos las direcciones IP tanto del raspberry Pi como del PC configuradas, además el PC ya está listo para usar el protocolo SSH para que la conexión con el hardware sea todo un éxito. Ahora vamos a activar el protocolo SSH en el raspberry PI.

8.2.2.5 Activación del protocolo SSH en el raspberry Pi

Para activar este protocolo, se ejecuta estos comandos:

- Sudo apt-get install ssh

Luego se inicia el servicio con el siguiente comando:

- Sudo /etc/init.d/ssh start

Y para que pueda ser ejecutado automáticamente al iniciar la raspberry Pi, se debe ingresar este código:

- Sudo update-rc.d ssh defaults

8.2.2.6 Instalación del programa Putty:

Se necesita instalar el cliente SSH en el computador, para esto se recomienda instalar el programa Putty, el cual se puede descargar por internet, este programa contiene todos los programas necesarios para hacer la conexión remota con el raspberry Pi y otros en el cual incrementa el nivel de seguridad. Después de la instalación del software, se hará la conexión con el raspberry Pi; para esto debemos ejecutar el programa Putty.

Se verá una ventana como la siguiente:

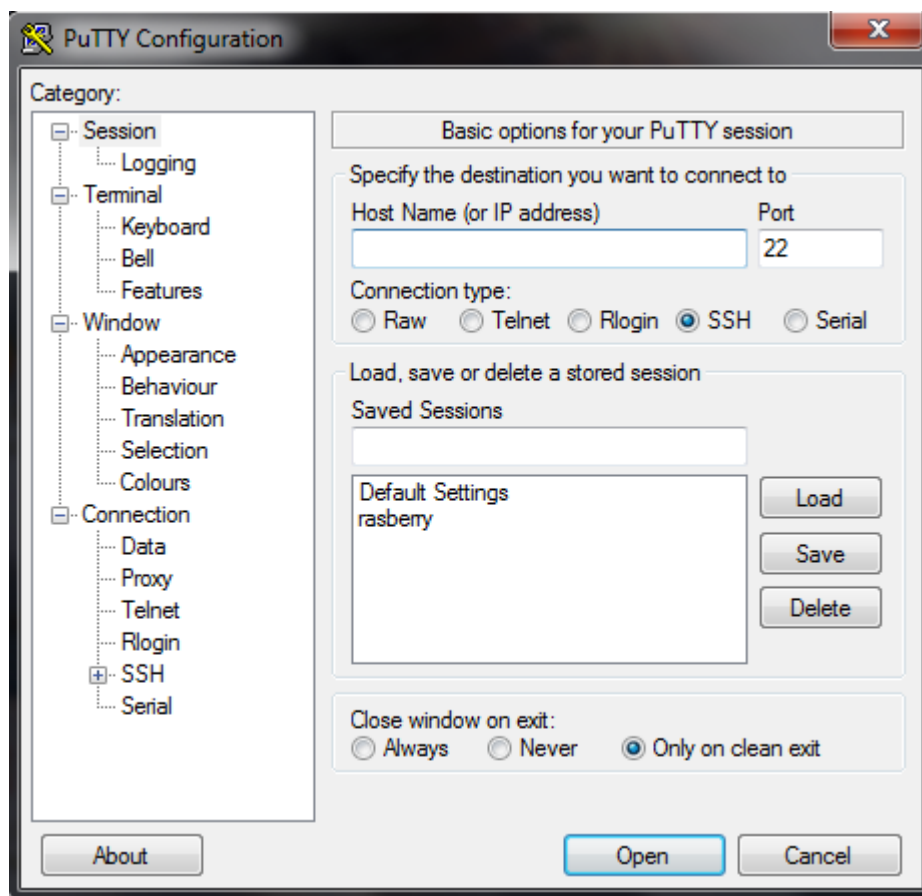


Figura 23. Configuración del Programa Putty (Elaborado por el autor)

En el espacio “Host Name” ingrese la dirección del Raspberry Pi, en este espacio se ha configurado con la dirección 192.168.137.2. Cabe aclarar que el puerto es el 22. Se le debe dar un nombre a la conexión (en este caso se le llamo raspberry) en el campo “saved sessions”

y se oprime “saved”, de esta forma se guarda la dirección IP, con eso no se volverá a escribir nuevamente cada vez que se conecte nuevamente. La próxima vez que se conecte simplemente se le da clic en el nombre de la conexión y luego en “Load” para poder conectarse al raspberry.

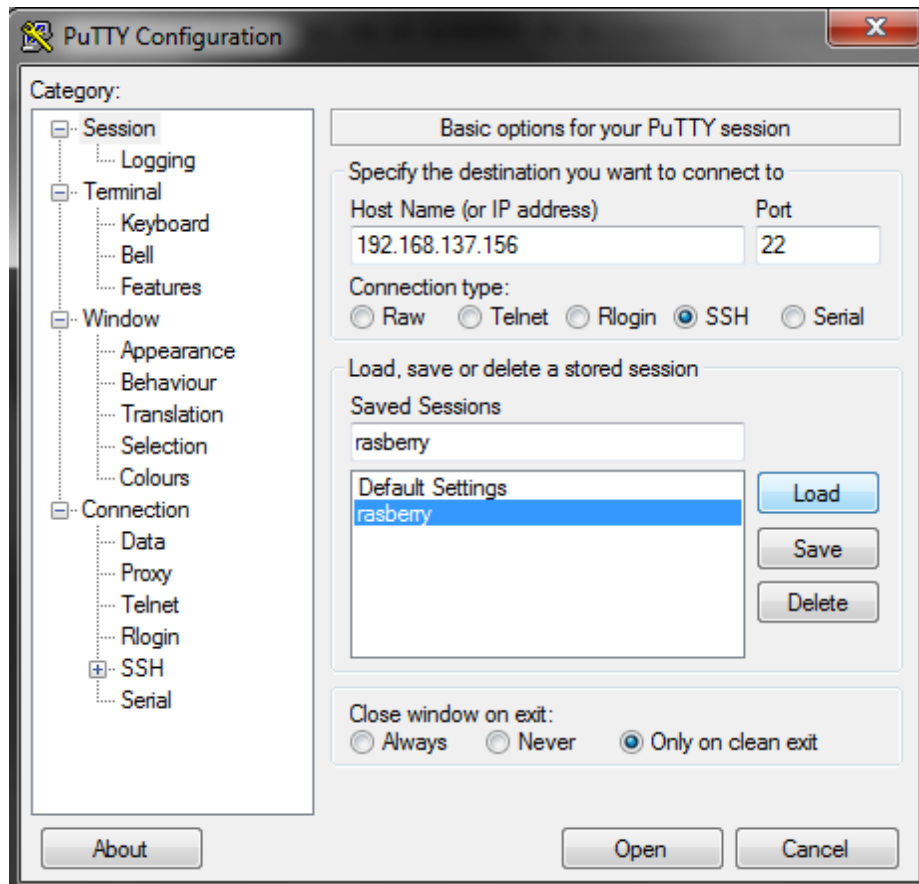
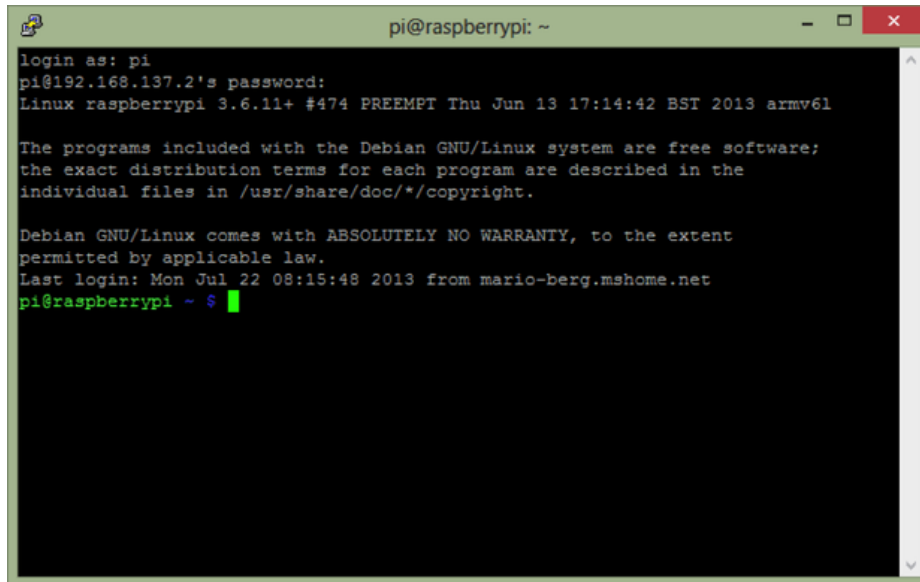


Figura 24. Dirección Ip del PC cargado y guardado en el programa Putty

(Elaborado por el autor)

Luego para empezar la conexión se le da clic en “Open”, si la conexión es un éxito aparecerá una ventana pidiendo el nombre del usuario y la contraseña. Saldrá una línea de comandos de Linux, esto quiere decir que ya está conectado el raspberry.



```
pi@raspberrypi: ~
login as: pi
pi@192.168.137.2's password:
Linux raspberrypi 3.6.11+ #474 PREEMPT Thu Jun 13 17:14:42 BST 2013 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Jul 22 08:15:48 2013 from mario-berg.mshome.net
pi@raspberrypi ~ $
```

Figura 25. Conexión exitosa con el programa Putty *(Elaborado por el autor)*

Ahora para que se muestre la interfaz y poder hacer la conexión completa entre el pc y el raspberry pi, se necesita descargar el programa VNC SERVER (Virtual network Computing). Es un software cliente-servidor, para explicar el funcionamiento de este servicio envía “fotos o imágenes” del escritorio al computador remoto, varias veces por segundo permitiendo ver el escritorio del computador remoto.

La primera vez que se ejecuta vncserver en el raspberry, le pedirá una contraseña para dar al acceso remoto, esta será solicitada cada vez que se vaya a acceder al escritorio. Lo más recomendado es usar la misma contraseña del raspberry o algo fácil de recordar ya que este no tiene la opción de recordar contraseña como lo tiene Facebook, el correo o diferentes páginas web que brindan esta opción.

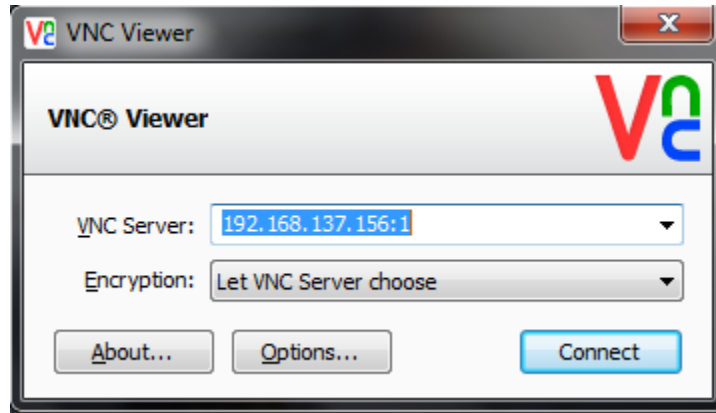


Figura 26. Dirección IP de la raspberry cargada en el VNCSERVER

(Elaborado por el autor)

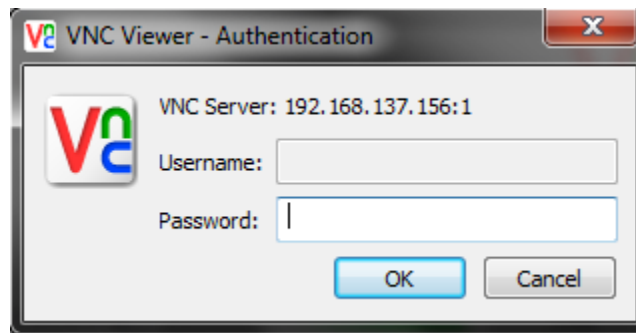


Figura 27. Espacio para digitar la contraseña en el VNCSERVER *(Elaborado*

por el autor)

Ahora si entramos al sistema operativo de raspberry por medio del protocolo de comunicación SSH, con esto ya podemos por medio del lenguaje Python hacer el control de los implementos escogidos.

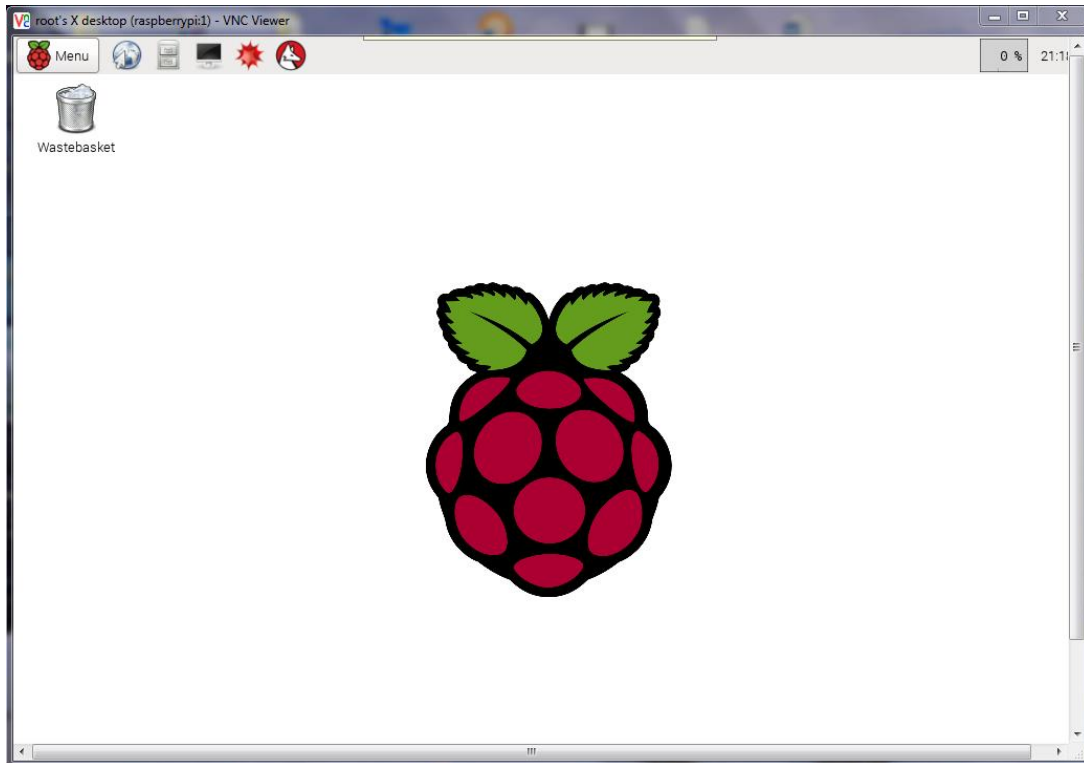


Figura 28. Escritorio remoto de la raspberry listo para empezar a trabajar en Python

(Elaborado por el autor)

8.2.3 Control de los motores y sensor ultrasonido:

8.2.3.1 Control DC:

La base de este control es el modulo integrado L298, el cual es puente H, que es capaz de manejar niveles altos de voltaje y corriente, está diseñado para soportar cargas inductivas como relés, motores de corriente continua y motores paso a paso. Tiene dos puentes para habilitar o deshabilitar las salidas de las señales de entrada. Este puente H es alimentado con 12 v para tener un buen control y un correcto desarrollo del motor teniendo en cuenta la velocidad programada y el consumo de corriente del mismo.

Las características de este puente H son las siguientes:

- Interfaz de potencia 7v a 46v

- Corriente máxima 2 Amperios
- Voltaje de control 5v
- Corriente de control 36 mA.
- Potencia de salida 25w
- Luces indicando encendido, control y dirección

8.2.3.1.1 Definición de las librerías en Python:

- **Import time:** Este módulo ofrece varias funciones relacionadas con el tiempo, permitiendo asignar los tiempos en el motor.

- **Import os:** Este módulo suministra de una interfaz a la API del sistema operativo para poder controlar el motor con una entrada del teclado del computador.

- **Import RPi.GPIO as gpio:** Este módulo permite utilizar los pines de la raspberry en la programación en Python-

8.2.3.1.2 Definición de las funciones:

- **Función def:** Las funciones descritas Speed y Menú son generales ya que con estas se puede tener una mejor definición de las demás funciones que operan dentro del menú, es decir, no se ejecuta hasta que no sea invocada. Para invocar dichas funciones, se les llama por su nombre, por ejemplo

```
def speed():
```

```
def menú():
```

- **Raw_input:** Esta función hace Python guarde lo que sea dentro de un string, cabe aclarar que esto es de gran ayuda para enviar datos del teclado del ordenador al raspberry PI mediante el protocolo de comunicación desarrollado SSH

- **Os.system ('clear')**: Este comando limpia la pantalla en Python con esa función, este es válido para cualquier sistema operativo: Windows, Linux, DOS, etc. Lo primero que hay que hacer, ya estando trabajando en cualquier entorno de desarrollo, es importar el modulo os. Ya que de esta manera cada vez que se ejecute el menú con sus diferentes opciones siempre se limpiara la pantalla.

- **GPIO.cleanup ()**: Proporciona la función de limpiar todos los puertos que se ha utilizado en este proyecto, cabe aclarar que solo afecta a cualquier puerto que se configura en el programa actual. Con esto se restablece cualquier puerto que se usa en este programa de nuevo modo de introducción. Esto previene cualquier daño, por ejemplo, una situación en la que tiene un puerto configurado como una salida ALTA y conecta a GND (BAJO), lo que por consiguiente haría un corto circuito en el puerto provocando daños.

- **Time.sleep ()**: Con esta función permite controlar el tiempo que estará encendido el motor, por ejemplo, `time.sleep (5)`, es decir tiempo de 5 segundos: `time.sleep (.500)`, quiere decir, tiempo de 300 milisegundos.

- **Python PWM**: La modulación de ancho de pulso controla la potencia, es decir, controla la cantidad de energía que va al motor y por lo tanto la rapidez que gira.

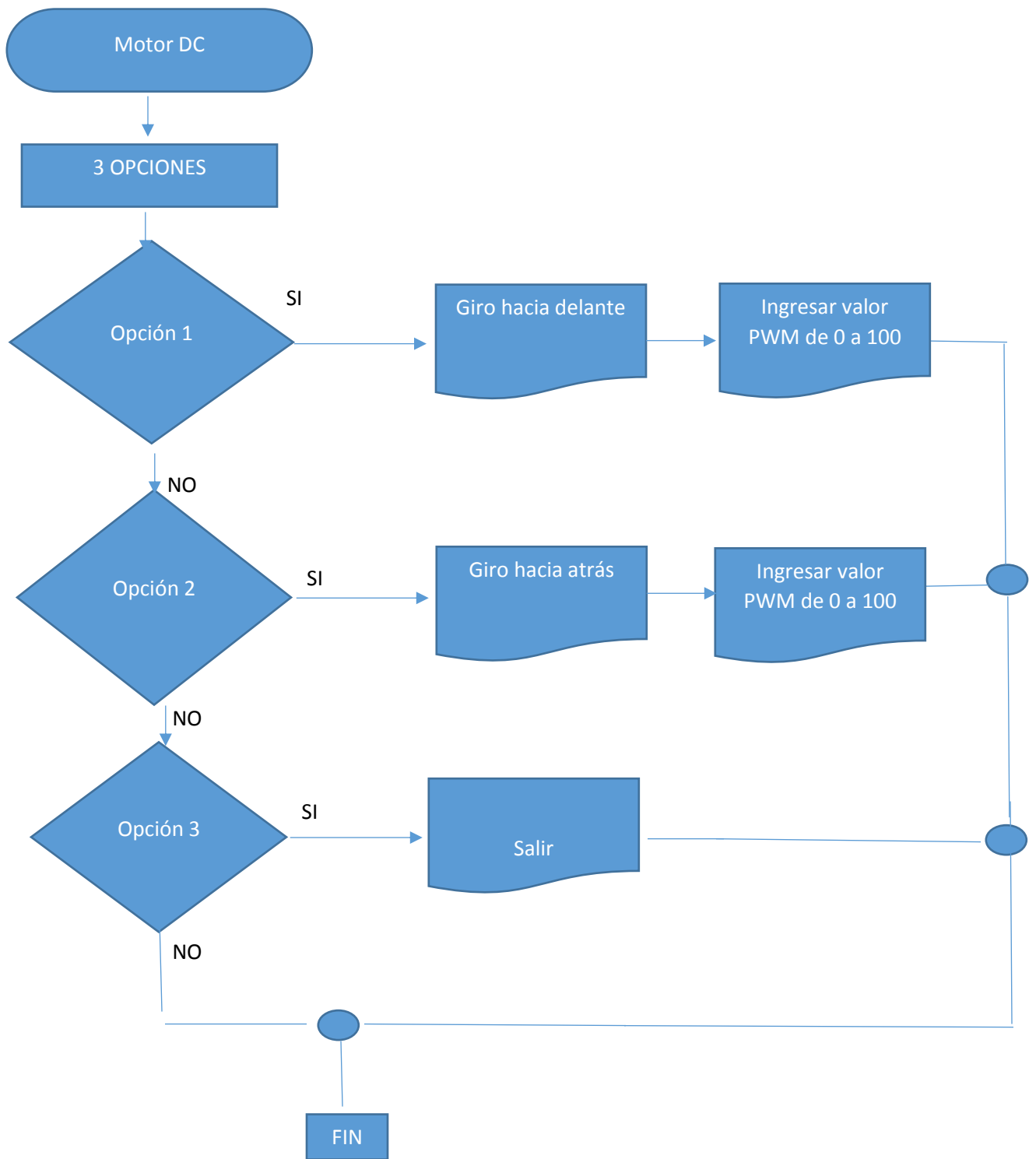


Figura 29. Diagrama de Flujo del Control Motor DC *(Elaborado por el autor)*

8.2.3.2 Control Motor Paso a Paso:

Este motor se controla con 5 cables:

- 5v
- Entrada 1
- Entrada 2
- Entrada 3
- Entrada 4

Para el control de cada una de las fases se utiliza el driver que contiene el integrado ULN2003, el cual proporciona 8 salidas en colector abierto que se controla por 8 entradas digitales.

Se usan las mismas funciones y librerías que se describieron en el párrafo anterior solo que se acondiciona el código para que los GPIO se activen en la secuencia adecuada para su funcionamiento en 8 o 4 pasos.

Se hacen las llamadas sucesivas del 1 al 8 para que el motor gire en un sentido o del 8 al 1 para que gire al revés, por lo tanto, esta secuencia se repite cada 8 llamadas siempre y cuando no haya un cambio de sentido. En cuanto, a la secuencia de 4 pasos se hacen 4 llamadas solo que esta vez son de nivel superior que harán girar el motor en los pasos deseados.

El valor de entrada representa el número de pasos a aplicar y el signo establece el sentido del giro, es decir, con los valores positivos el motor girara a la derecha, mientras que los valores negativos girara hacia la izquierda, cuando finaliza la secuencia se llama a la función con un valor de 0 para desactivar las bobinas.

Al momento de ejecutar el menú que se despliega, este pide el número de giros y el giro, por lo tanto cuando el estudiante o usuario ingrese un signo negativo y un número cualquiera el motor paso a paso girara a la derecha los pasos que el estudiante haya ingresado, en cambio, si digita un numero positivo girara al otro lado.

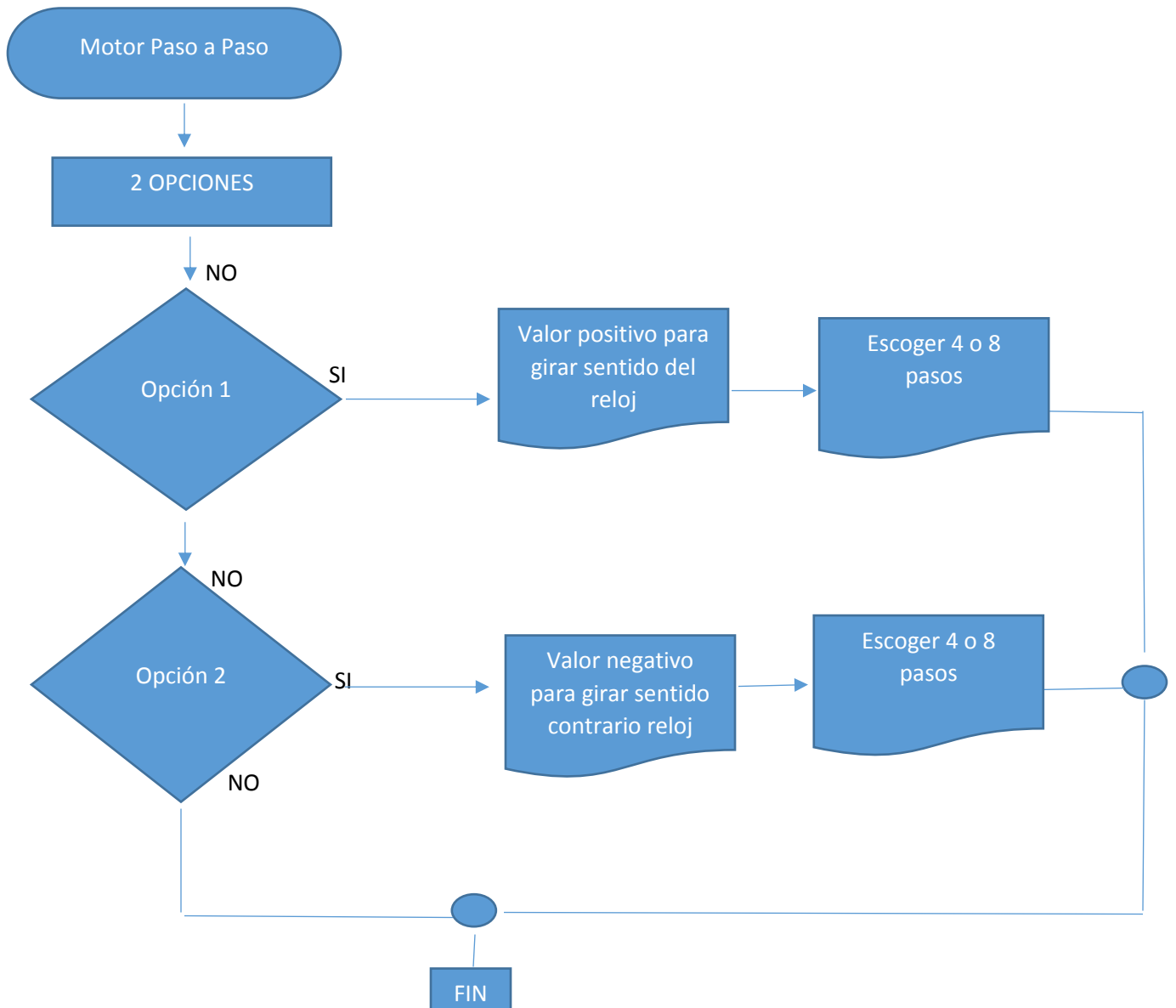


Figura 30. Diagrama de flujo Control Motor paso a paso (Elaborado por el autor)

8.2.3.3 Control Servo Motor:

El servomotor que se uso es el TOWER PRO SG 90, es un servo motor muy sencillo y económico, de fácil adquisición por cualquier estudiante, consta de una caja de reductora con engranajes, un motor eléctrico y un circuito electrónico de control.

Los cables que tiene este motor son 3 de diferentes colores, el color marrón corresponde al GND, el color rojo corresponde a VCC y el color amarillo corresponde a SIGNAL o SEÑAL. Este servomotor se alimenta de 5 voltios.

Para poder controlar el servo se usa la librería RPi.GPIO e import time, se pone el pin 21 como salida y después en modo pwm.

Primero que todo se importa el tiempo para poder usar time.sleep, luego la raspberry se pone en modo Board, el pin 21 se define como salida y a su vez en modo pwm y se envía 50 pulsos por segundo, al mismo tiempo se envía un pulso del 7.5% para poder centrar el servomotor.

Con un While se inicia un loop infinito, dentro de este se envía un pulso de 4.5% para poder girar hacia la izquierda, esperamos cierto tiempo en milisegundos, luego enviamos un pulso de 10.5% para poder girar el servo hacia la derecha, esperamos otro tiempo en milisegundos, además se envía un pulso de 7.5% para centrar el motor nuevamente y esperamos en milisegundos. Si el usuario pulsa Control + C se detiene el servo y por último los pines GPIO de la raspberry se limpian.

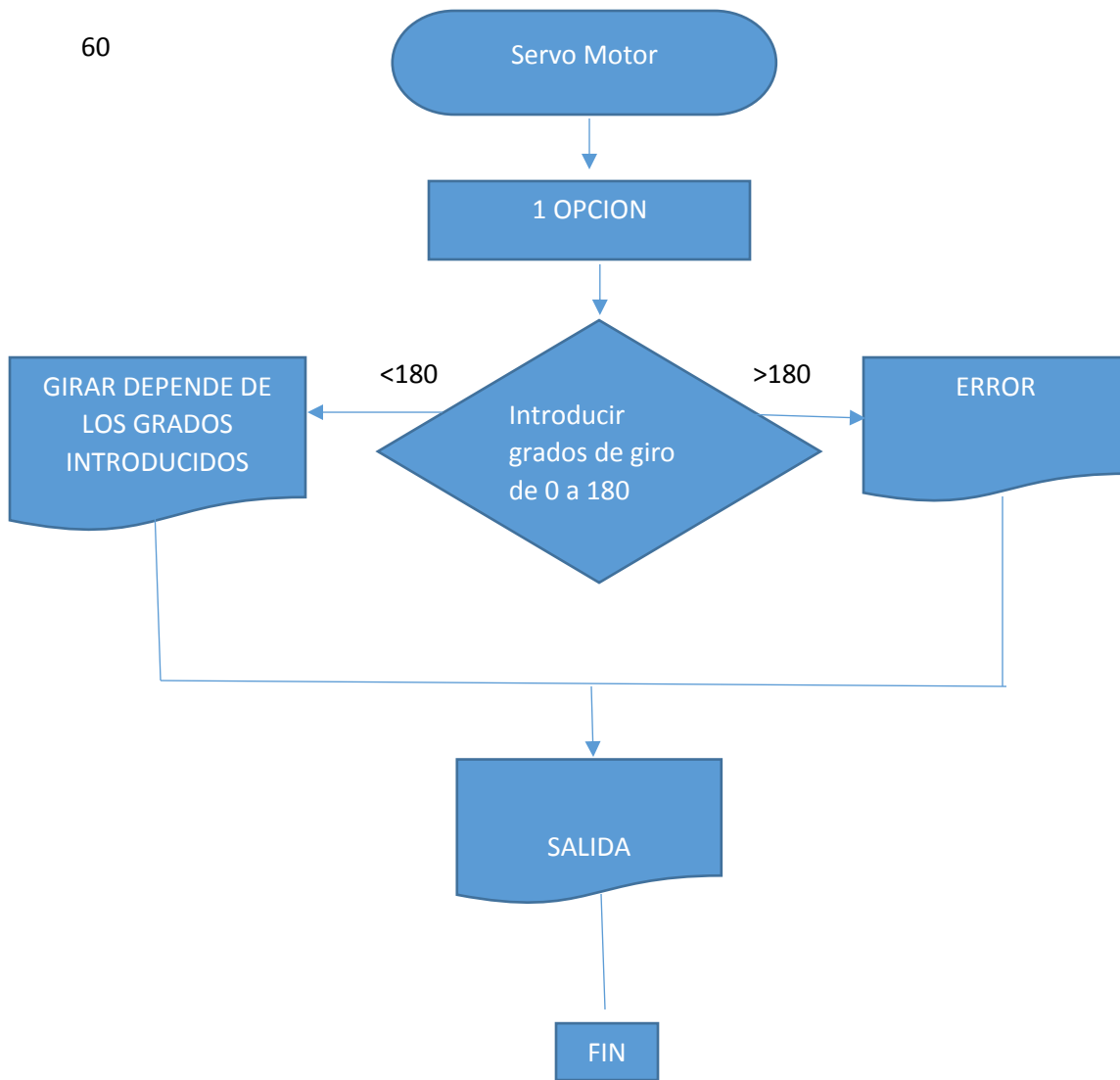


Figura 31. Diagrama de flujo Control Servo Motor (Elaborado por el autor)

8.2.3.4 Control Sensor Ultrasonido:

Para este control se importan las funciones de los módulos necesarios que son RPi.GPIO, TIME, el pin GPIO donde se conecta el activador (TRIGGER), es el 23, el pin de la salida (ECHO) del sensor es el 24.

Luego indicamos el esquema de los números de GPIO con el siguiente comando: GPIO.setmode (GPIO.BCM), después establecemos el trigger como un canal de salida, y el pin ECHO como un canal de entrada, se cuentan las mediciones que llevamos. Iniciamos con un ciclo infinito para terminar, para terminar el programa se oprime Ctrl + C, en seguida se

apaga el pin activador y permite un par de segundo para que se estabilice, más tarde se prende el activador por 10 micro segundos y después se vuelve apagar; en este momento el sensor envía una cantidad de pulsos de 40 Hz (ultrasónicos) y coloca su salida ECHO en alto ya que se debe detectar este proceso e iniciar un conteo de tiempo.

La salida ECHO se mantendrá en alto hasta recibir el eco reflejado por el obstáculo, en ese momento el sensor pondrá ECHO en bajo y termina de contar el tiempo. La duración del eco se mide en segundos, teniendo en cuenta que distancia = velocidad * tiempo, por lo tanto la velocidad del sonido es 343 m/s, el cual necesitamos dividir el valor resultante entre dos por que la medición corresponde al tiempo que tarda el pulso ultrasónico en llegar al obstáculo y regresar nuevamente al sensor.

$$\text{Distancia} = (34300 * \text{duración_eco})/2$$

Se imprime el resultado, luego incrementamos el número de mediciones realizadas y se muestra el resultado en pantalla. Por último, se reinicia todos los canales GPIO.

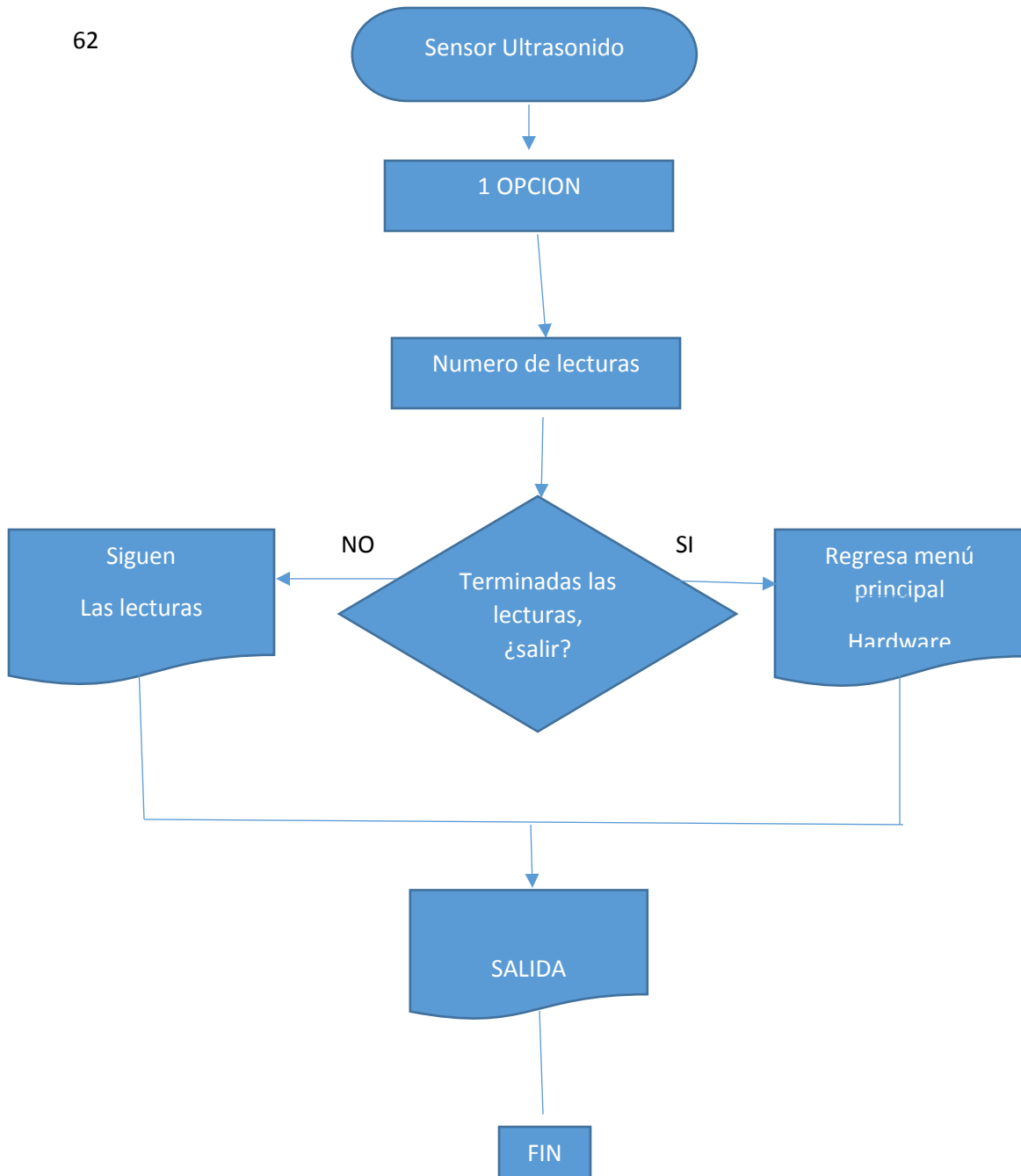


Figura 32. Diagrama de flujo Lectura Ultrasonido (Elaborado por el autor)

8.2.4 Alimentación:

Se utiliza un adaptador con un voltaje de entrada AC de 100 – 240 voltios y un voltaje de salida de 5 voltios, proporcionando una corriente de 2.5 amperios, el cual son suficientes para alimentar el hardware y así poder garantizar el correcto funcionamiento de toda la tarjeta.



Figura 33. Cargador Canaki Raspberry pi 2 (*Raspberry pi 2016*)

8.3 IMPLEMENTACION:

8.3.1 Interfaz gráfica:

La línea de comandos en un equipo posibilita al usuario un control sobre su sistema, en este caso el raspberry pi. Los usuarios que usen Windows pueden llegar a esta pantalla a través del siguiente comando Command Prompt o Powershell. Estas herramientas permiten al usuario manipular directamente si sistema a través del uso de comandos.

El LXTERMINAL es la aplicación de terminal que tiene por defecto el Raspberry Pi, se le conoce como un emulador de terminal, es decir, que emula las antiguas terminales de video en un entorno gráfico. Esta ventana se puede encontrar en el escritorio de las Raspberry pi y cuando inicia se ve esto:

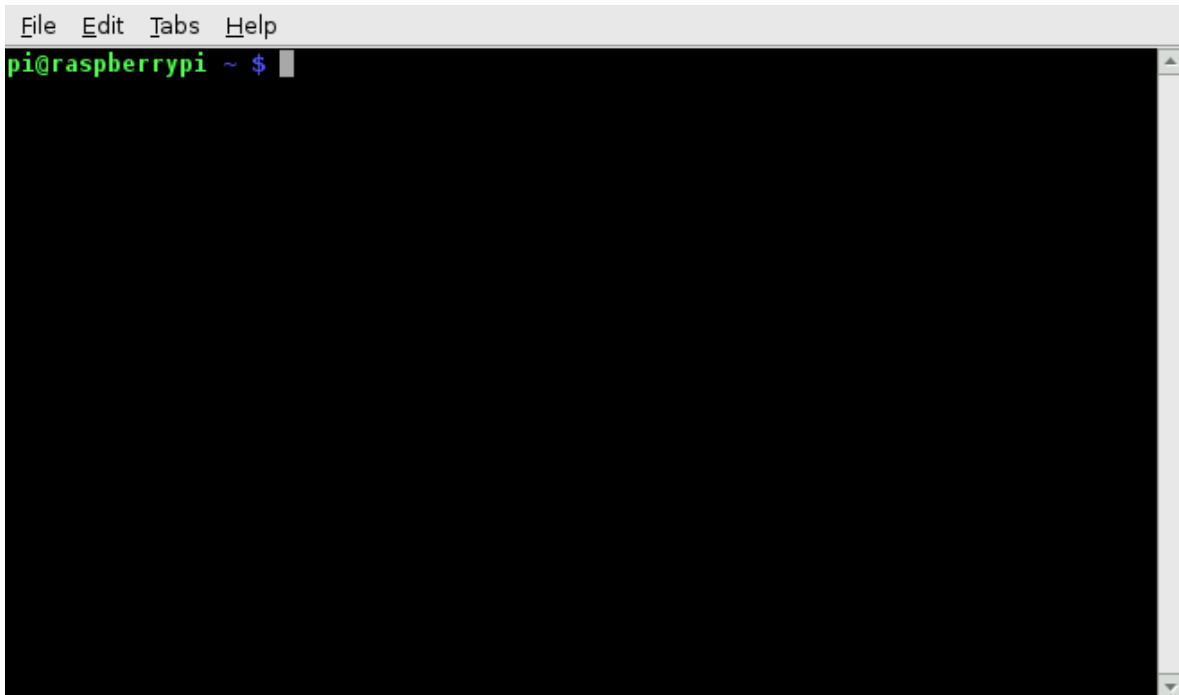


Figura 34. LX Terminal que tiene la raspberry para ejecutar el control

(Elaborado por el autor)

El primer mensaje que se ve es este:

```
pi@raspberrypi ~$
```

Aquí se muestra el nombre de usuario y el nombre del host de la tarjeta. En este caso, el usuario es pi y el nombre de host es raspberrypi.

8.3.1.1 Comando SUDO:

Existen comandos en el que se hacen cambios permanentes en su sistema y que requiere que el usuario tenga privilegios de root para ejecutar. Este comando sudo da cuenta de forma temporal la capacidad de ejecutar estos comandos, siempre y cuando su nombre de usuario está en una lista de usuarios. Cuando se le añade el sudo al inicio de un comando y oprime

enter, le pedirá una contraseña y todo es correcto entonces el comando que desea ejecutar se ejecuta con privilegios root.

8.3.2 Menú Principal:

Después de hacer la conexión entre el PC y el hardware ya hecho. Se ejecuta el programa putty, aparece un mensaje así:

Login as:

Se escribe root para poder entrar al hardware, se espera 5 segundos y este pedirá una contraseña, la cual es 1234 (es fácil de recordar) así debe aparecer:

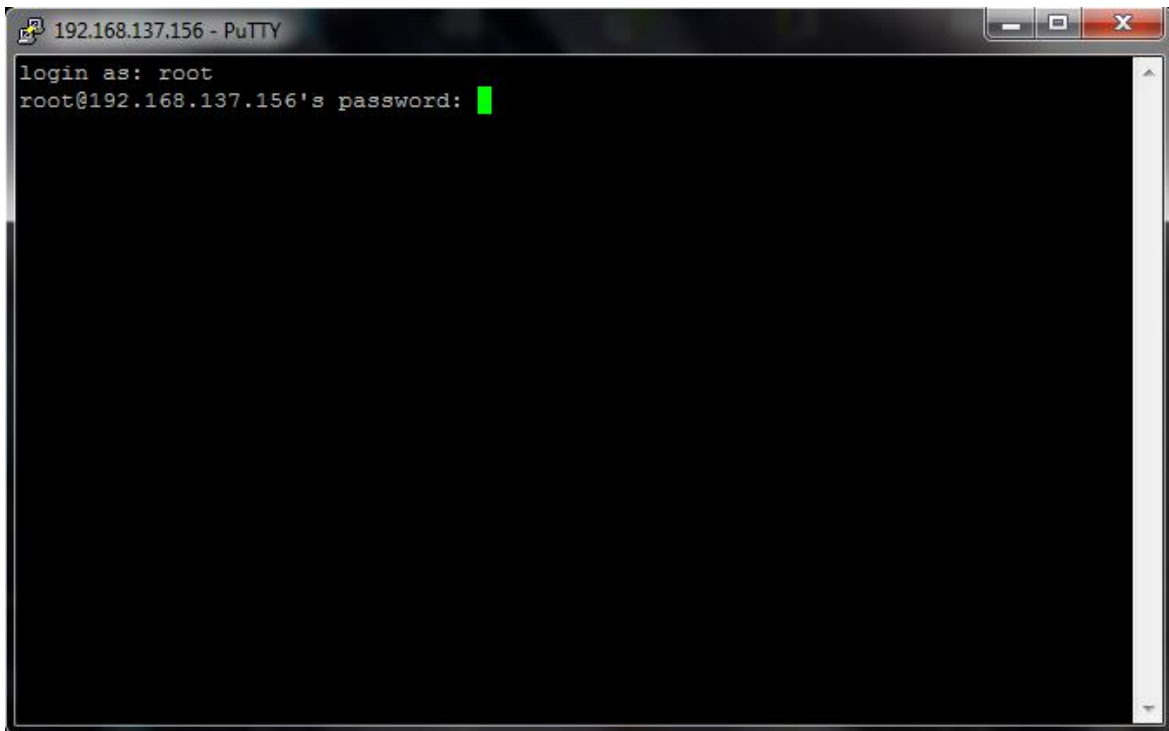


Figura 35. Ingreso de root y contraseña en el programa Putty (Elaborado por el autor)

Luego de ingresar la contraseña, la interfaz dará acceso a la raspberry pi.

Para entrar a la raspberry, hay que ejecutar el menú principal, para esto se tiene que dar acceso a donde está dicho archivo, a continuación se muestra como se hace:

- `cd pacho`: este comando da acceso a la carpeta llamada pacho donde está guardado el archivo a ejecutar
- Luego se escribe `Python iniciar.py`: este comando entra al archivo y lo ejecuta
- Aparecerá el menú principal del hardware, posee 4 opciones, las cuales son el control de los motores DC, Servomotor, Paso a paso y la Lectura del Sensor, como en la siguiente lo muestra:

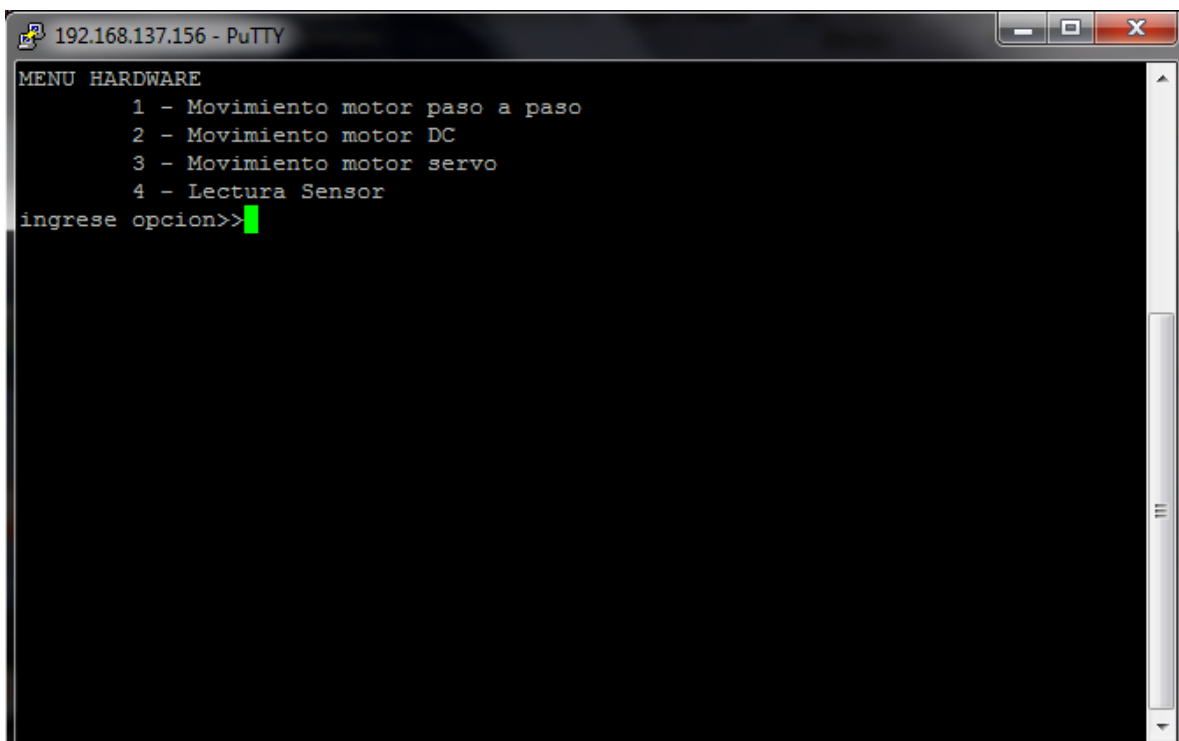
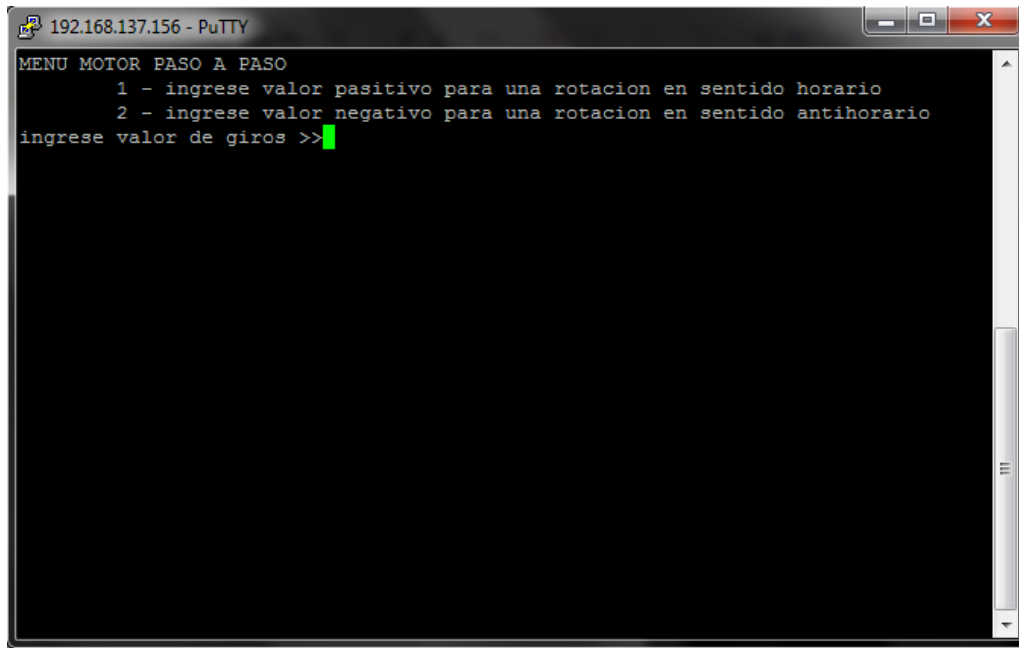
A screenshot of a PuTTY terminal window. The title bar reads "192.168.137.156 - PuTTY". The terminal content shows a menu titled "MENU HARDWARE" with four numbered options: "1 - Movimiento motor paso a paso", "2 - Movimiento motor DC", "3 - Movimiento motor servo", and "4 - Lectura Sensor". Below the menu, the prompt "ingrese opcion>>" is displayed with a green cursor.

Figura 36. Menú Principal del Hardware (Elaborado por el autor)

Como lo podemos ver en la anterior imagen, el usuario o estudiante selecciona la opción que desee.

8.3.2.1: Opción 1 Motor paso a paso:

Cuando se selecciona la opción 1 que es el Menú motor paso a paso, aparece esto:



```
192.168.137.156 - PuTTY
MENU MOTOR PASO A PASO
  1 - ingrese valor positivo para una rotacion en sentido horario
  2 - ingrese valor negativo para una rotacion en sentido antihorario
ingrese valor de giros >>
```

Figura 37. Menú Motor paso a paso *(Elaborado por el autor)*

Nuevamente se despliega otro menú donde hay dos opciones, una es que el usuario ingrese un valor positivo para que el motor gire en sentido de las manecillas del reloj (horario). La otra opción pide que el usuario ingrese un valor negativo para que el motor gire en sentido contrario de las manecillas del reloj (anti horario).

A screenshot of a PuTTY terminal window titled "192.168.137.156 - PuTTY". The terminal displays a menu in Spanish: "MENU MOTOR PASO A PASO", "1 - opcion de pasos 4 o 8", and "ingrese nuemro de pasos>>". A green cursor is positioned at the end of the last line, ready for input. The terminal window has standard Windows-style window controls (minimize, maximize, close) in the top right corner.

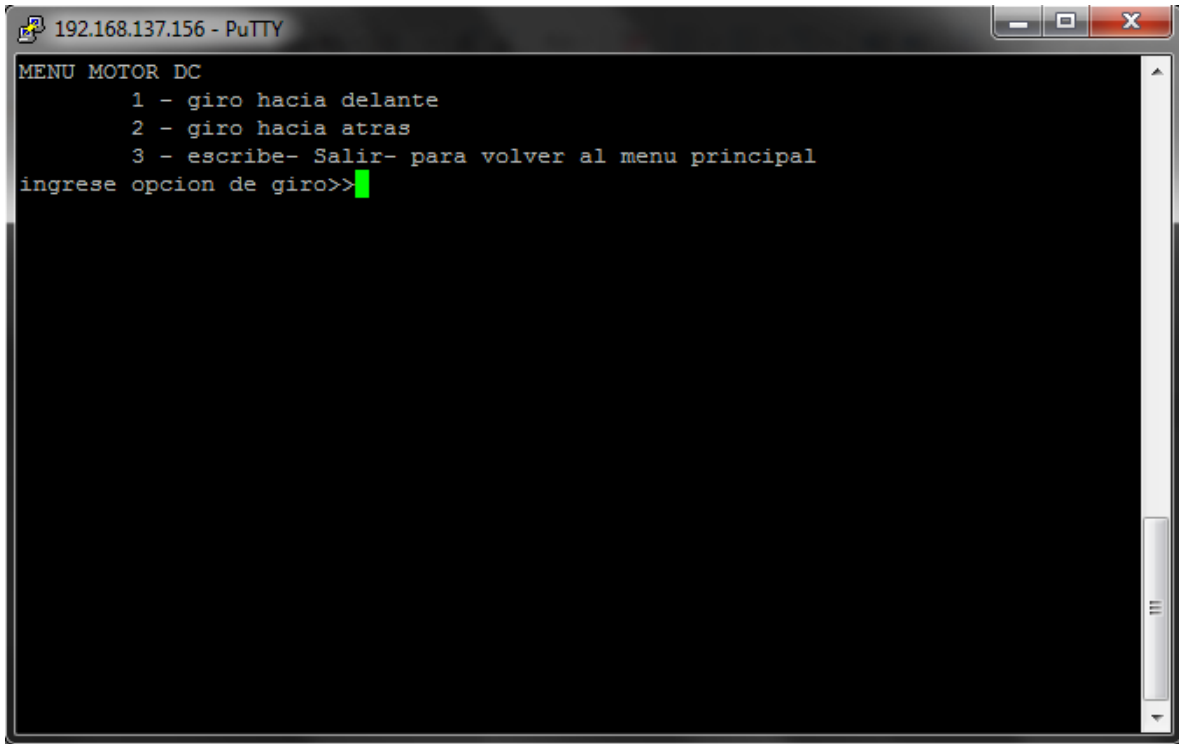
Figura 38. Elección de número de pasos 4 o 8 (*Elaborado por el autor*)

Independientemente de la opción que escoja el usuario ya sea el sentido horario o anti horario, el hardware pide el número de pasos que quiera dar el motor sean 4 o 8 pasos.

Al haber realizado la orden del usuario, automáticamente regresa al menú del motor paso a paso, pidiendo nuevamente el sentido del giro del motor, si el usuario ingresa salir regresa al menú principal del hardware.

8.3.2.2 Opción 2 Menú Motor DC:

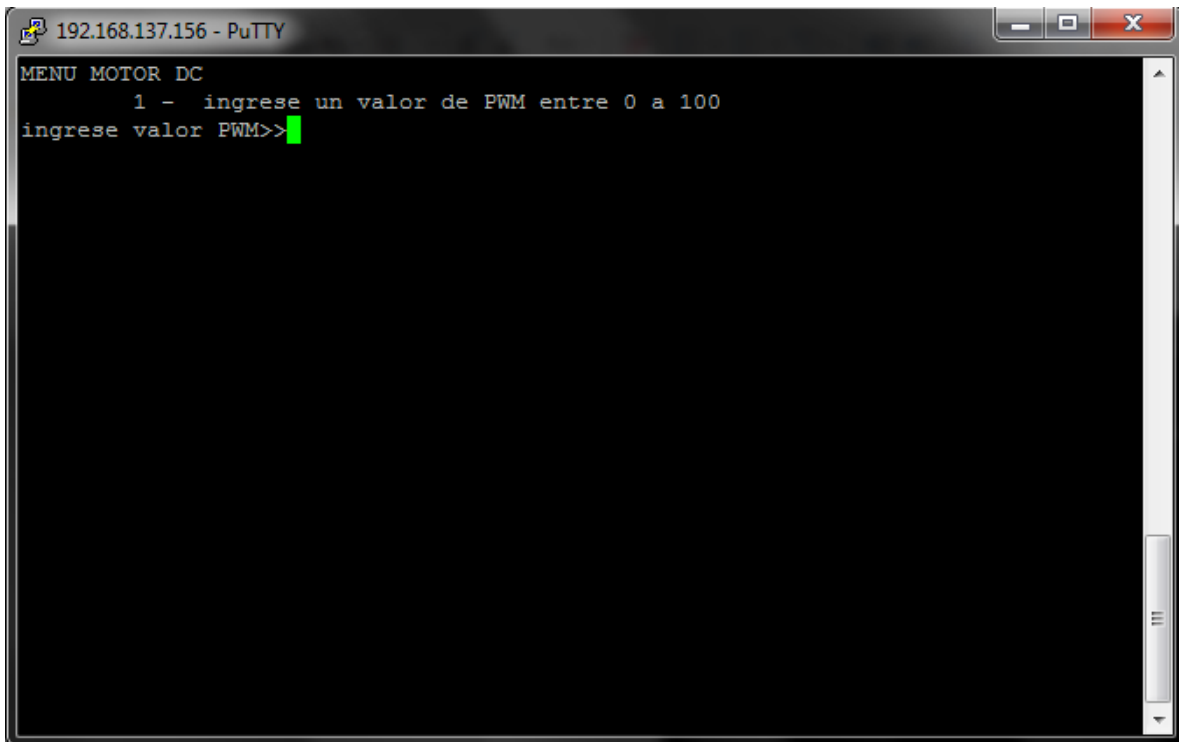
La segunda opción Movimiento Motor DC:

A screenshot of a PuTTY terminal window titled "192.168.137.156 - PuTTY". The terminal displays a menu for a DC motor with three options: "1 - giro hacia delante", "2 - giro hacia atras", and "3 - escribe- Salir- para volver al menu principal". Below the menu, the prompt "ingrese opcion de giro>>" is shown with a green cursor. The terminal window has standard Windows-style window controls (minimize, maximize, close) in the top right corner.

```
192.168.137.156 - PuTTY
MENU MOTOR DC
  1 - giro hacia delante
  2 - giro hacia atras
  3 - escribe- Salir- para volver al menu principal
ingrese opcion de giro>>
```

Figura 39. Menú Motor DC *(Elaborado por el autor)*

Aquí se muestran 3 opciones, la primera el giro hacia delante, lo cual hace que el motor gire normalmente, la segunda hacia atrás, en esta opción frena el motor y hace que el motor gire al revés, se puede evidenciar cuando se ejecuta este código que el motor se fuerza para cumplir con la orden del usuario.



```
192.168.137.156 - PuTTY
MENU MOTOR DC
  1 - ingrese un valor de PWM entre 0 a 100
ingrese valor PWM>>
```

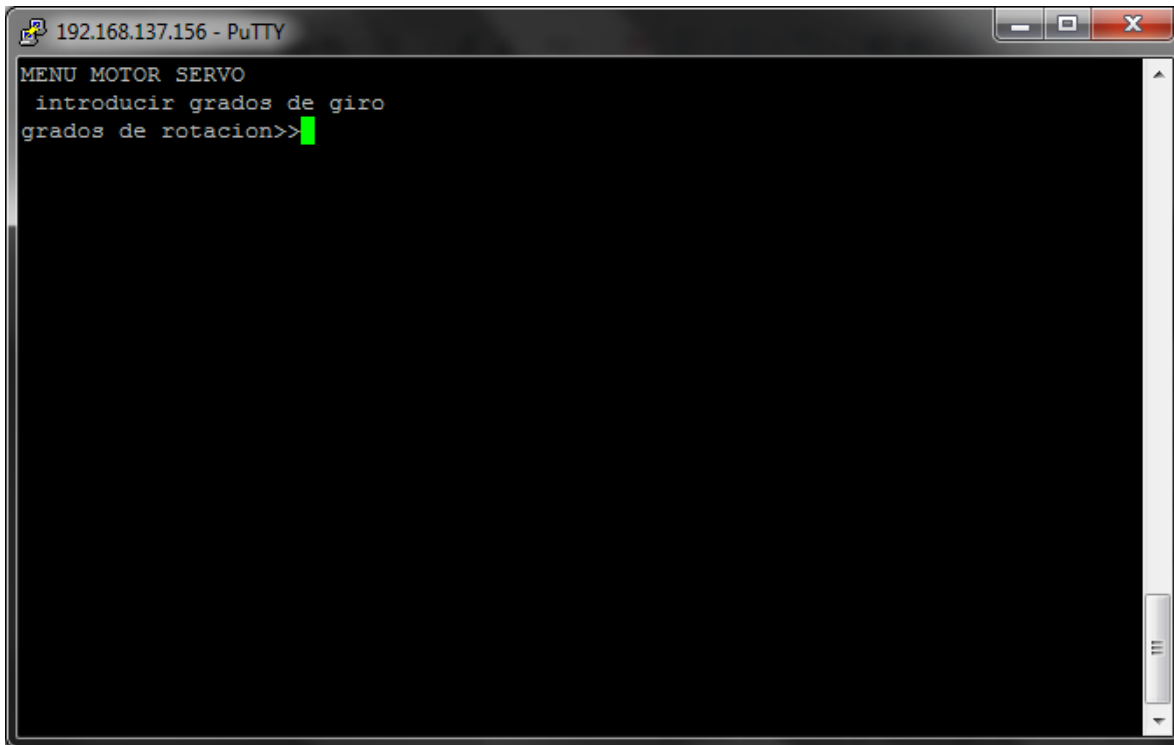
Figura 40. Ingreso valor de PWM de 0 a 100 *(Elaborado por el autor)*

Independiente de la opción que se escoja ya se la opción 1 o la opción 2, el menú muestra una nueva ventana pidiendo un valor de pwm entre 0 a 100, esto quiere decir, que el valor máximo de pwm es 100, por otra parte, si el usuario ingresa un valor mayor a 100, el código está programado para generar un error, el cual sirve de protección para la raspberry, debido a que si se fuerza mucho más el motor puede que la raspberry entre en corto o peor aún se quemara uno de los puertos GPIO.

Cuando se ejecuta la opción que el usuario haya escogido vuelve automáticamente al menú del Motor DC, para regresar al menú principal del hardware se escribe salir.

8.3.2.3 Opción 3 Menú Motor Servo:

La tercera opción Movimiento Servo Motor:



```
192.168.137.156 - PuTTY
MENU MOTOR SERVO
  introducir grados de giro
grados de rotacion>> █
```

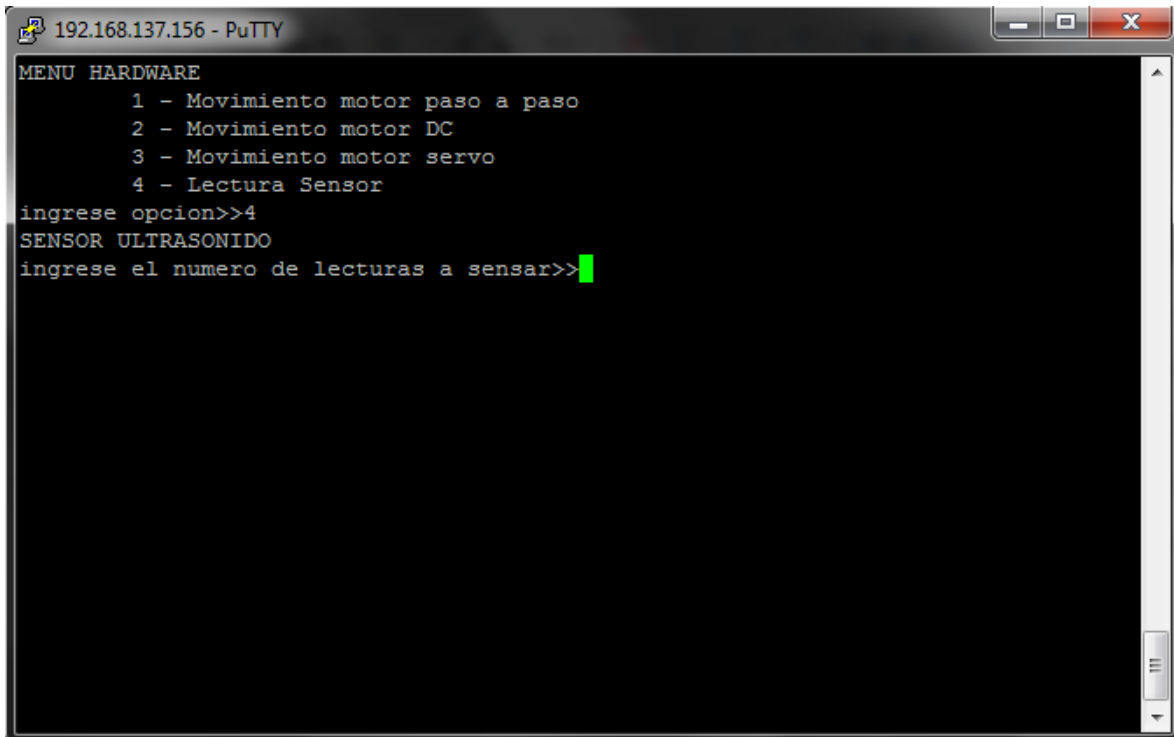
Figura 41. Menú Servo Motor *(Elaborado por el autor)*

Como se muestra en la imagen anterior, el código pide el valor de los grados para que el servomotor gire, cabe aclarar que el grado máximo que soporta ese servomotor es de 180 grados, el motivo por el cual esa limitación es por fabrica ya que el juego de engranajes no gira más de ese valor en grados desde su posición inicial.

Para regresar al menú principal del hardware se escribe salir.

8.3.2.4: Opción 4 Lectura Sensor:

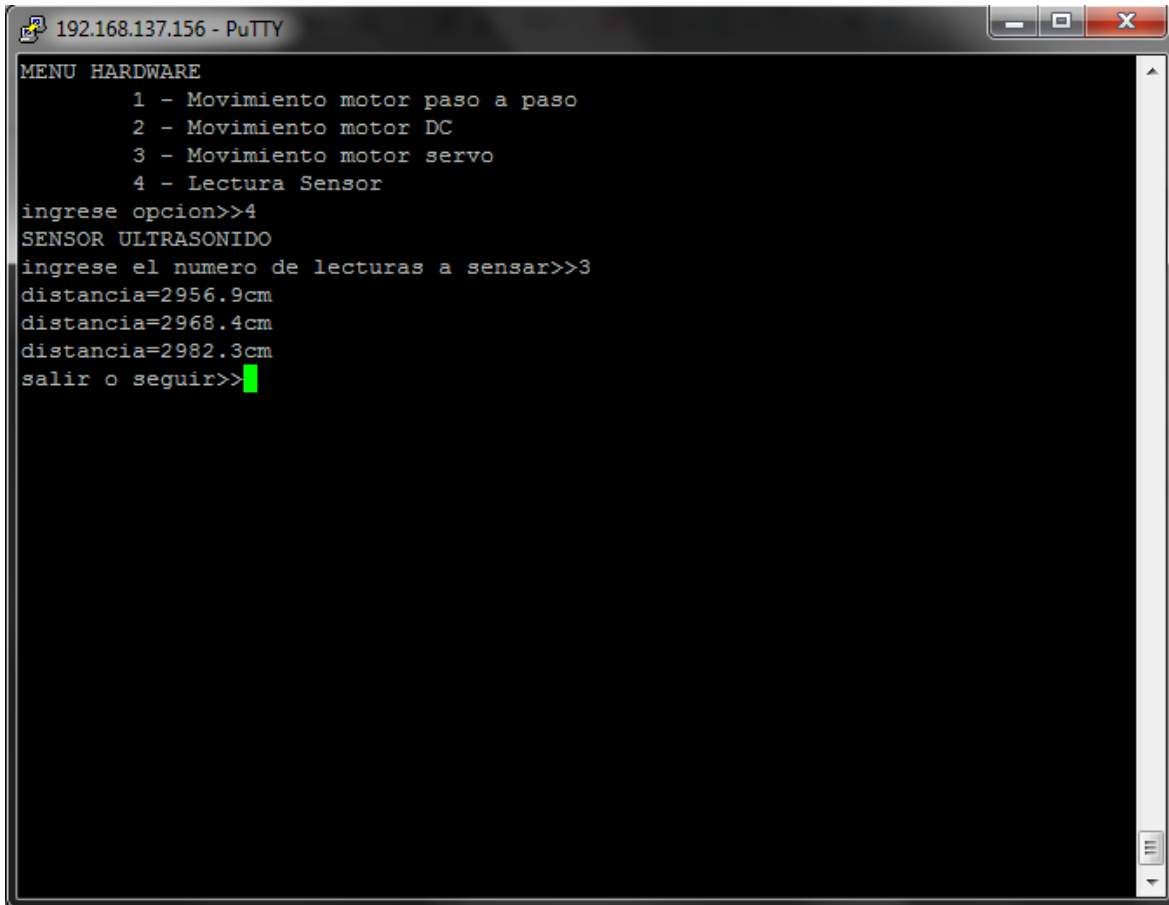
La Opción 4, Lectura del Sensor:



```
192.168.137.156 - PuTTY
MENU HARDWARE
  1 - Movimiento motor paso a paso
  2 - Movimiento motor DC
  3 - Movimiento motor servo
  4 - Lectura Sensor
ingrese opcion>>4
SENSOR ULTRASONIDO
ingrese el numero de lecturas a sensar>>
```

Figura 42. Menú de Lectura Sensor *(Elaborado por el autor)*

Cuando se ingresa la opción para leer el sensor, el código pide que ingrese el número de lecturas, pueden ser 10, 20, 30 los que el usuario quiera, cuando el sensor termine las lecturas aparece una opción diciendo salir o seguir, si se escoge salir regresa al menú principal del Hardware, si por el contrario selecciona seguir toma el mismo número de lecturas a sensar.



```
192.168.137.156 - PuTTY
MENU HARDWARE
  1 - Movimiento motor paso a paso
  2 - Movimiento motor DC
  3 - Movimiento motor servo
  4 - Lectura Sensor
ingrese opcion>>4
SENSOR ULTRASONIDO
ingrese el numero de lecturas a sensar>>3
distancia=2956.9cm
distancia=2968.4cm
distancia=2982.3cm
salir o seguir>>
```

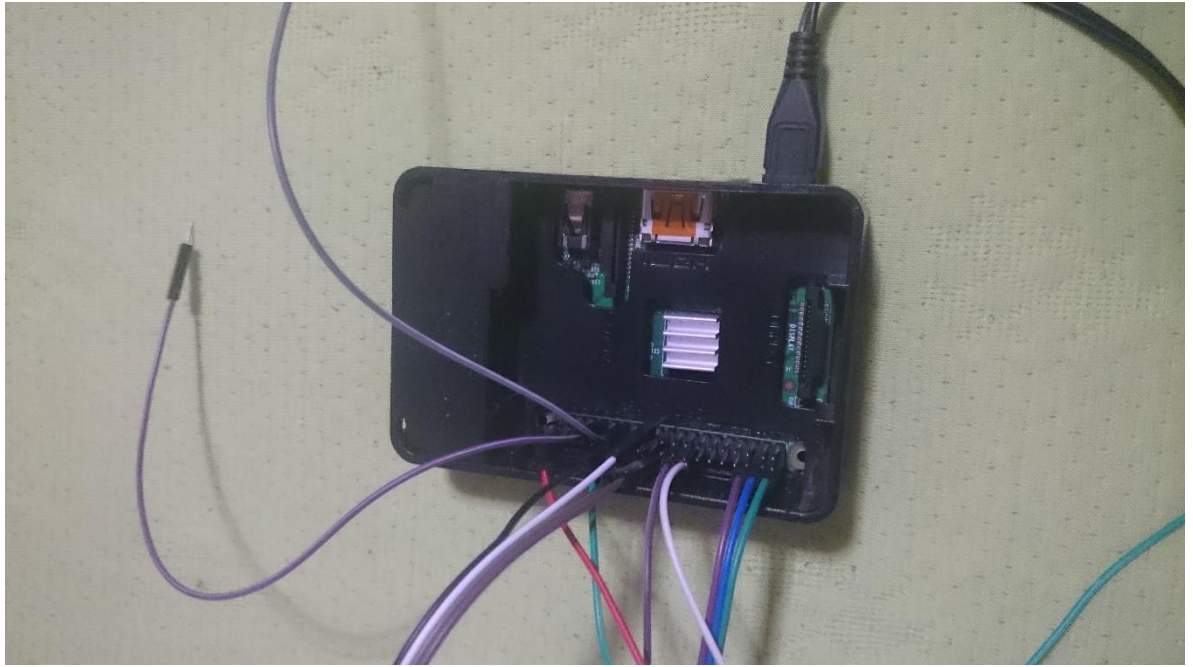
Figura 43. Lecturas hechas por el sensor *(Elaborado por el autor)*

9. CONCLUSIONES

- Este hardware proporciona al estudiante una herramienta para el control de los motores (DC, SERVO, PASO A PASO y SENSOR ULTRASONIDO) que son algunos de los elementos más usados en la licenciatura en electrónica, brindando al profesor una manera diferente e innovadora de realizar laboratorios para las diferentes asignaturas.
- Python es un lenguaje de programación muy compacto, generalmente es mucho más corto que sus similares lenguajes como C, es considerado un lenguaje de programación de muy alto nivel, además es muy versátil, puede ser orientado a objetos y logra interactuar con HTML para visualizar bases de datos, etc.
- El protocolo de comunicación SSH facilita las comunicaciones seguras entre cliente/servidor, además permite al usuario conectarse a un host remotamente, SSH encripta todo lo que envía y recibe proporcionando un nivel alto de seguridad para los hackers.
- La implementación de este hardware incide de una manera favorable el interés por los laboratorios remotos, ya que algunas veces no se puede desarrollar la práctica por diversos motivos ya sean de orden público, falta de materiales o no se encuentra disponible el espacio.
- El raspberry pi como arquitectura hardware por su reducido tamaño, su alta capacidad de proceso, su bajo costo, sus pines GPIO y su lenguaje de programación la convierten en un elemento de suma importancia para la realización de este proyecto, además raspberry pi es prácticamente una computadora funcional, lo cual la hacen el cerebro ideal para el hardware que se diseñó.

10. ANEXOS

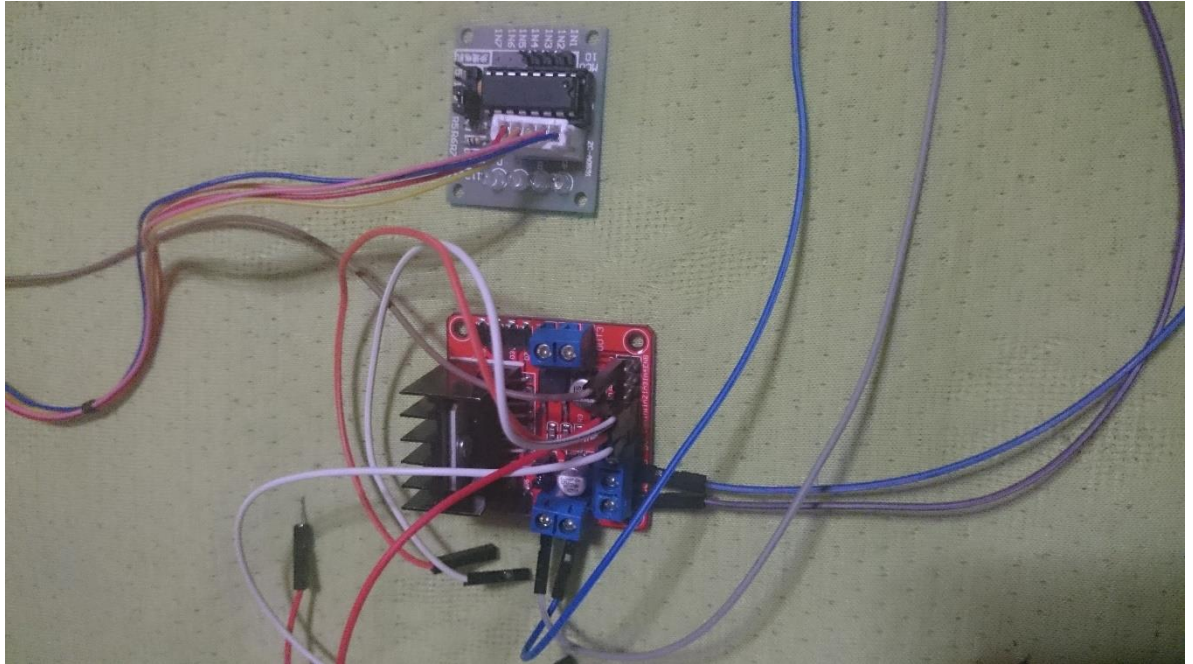
DISEÑO DE PRUEBA DEL HARDWARE



UBICACIÓN DE LAS PARTES INTERNAS DEL HARDWARE



Etapa de proteccion para la tarjeta

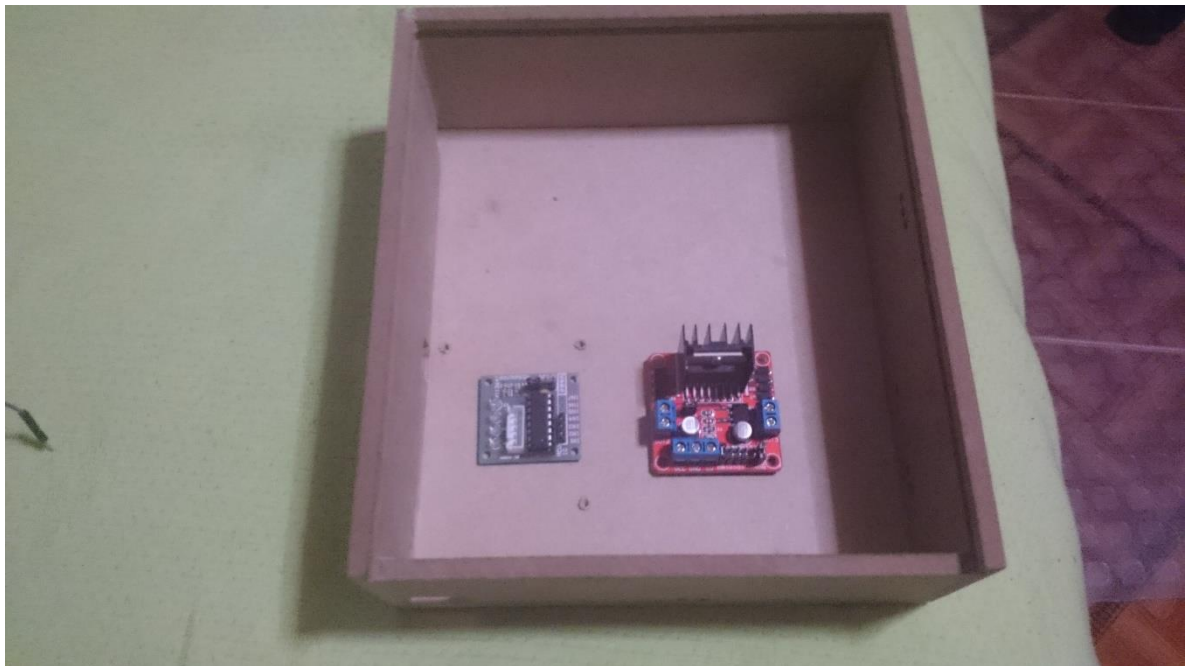


CARCASA DEL HARDWARE



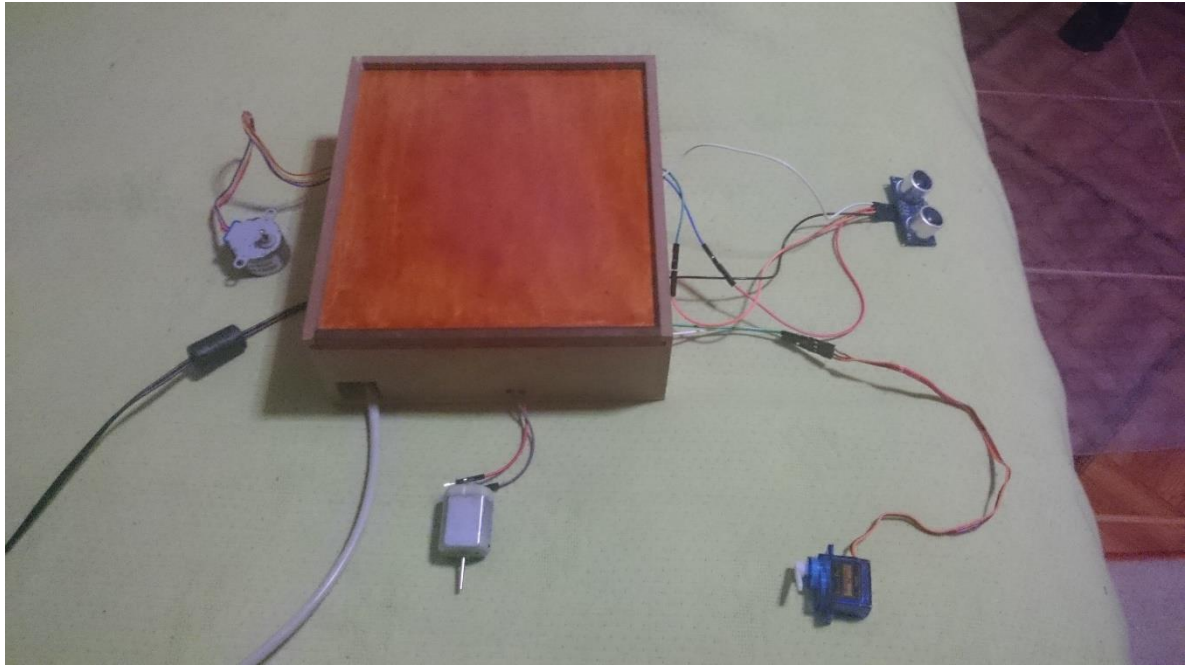


PRIMER DISEÑO DEL HARDWARE

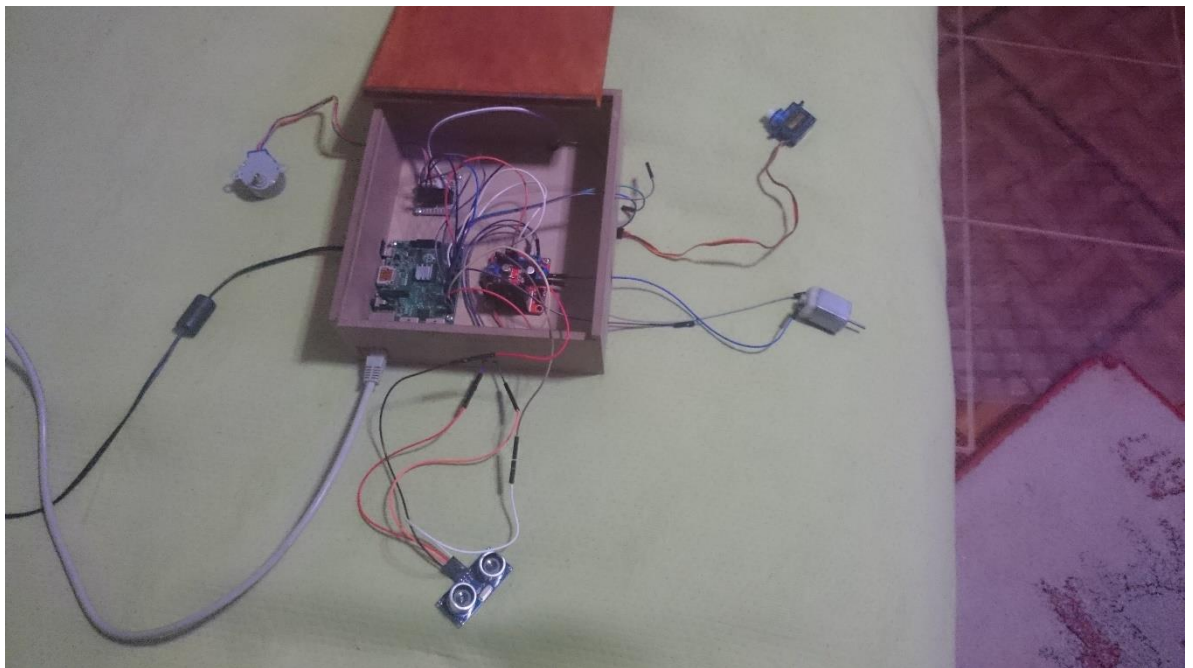


DISEÑO FINAL DEL HARDWARE

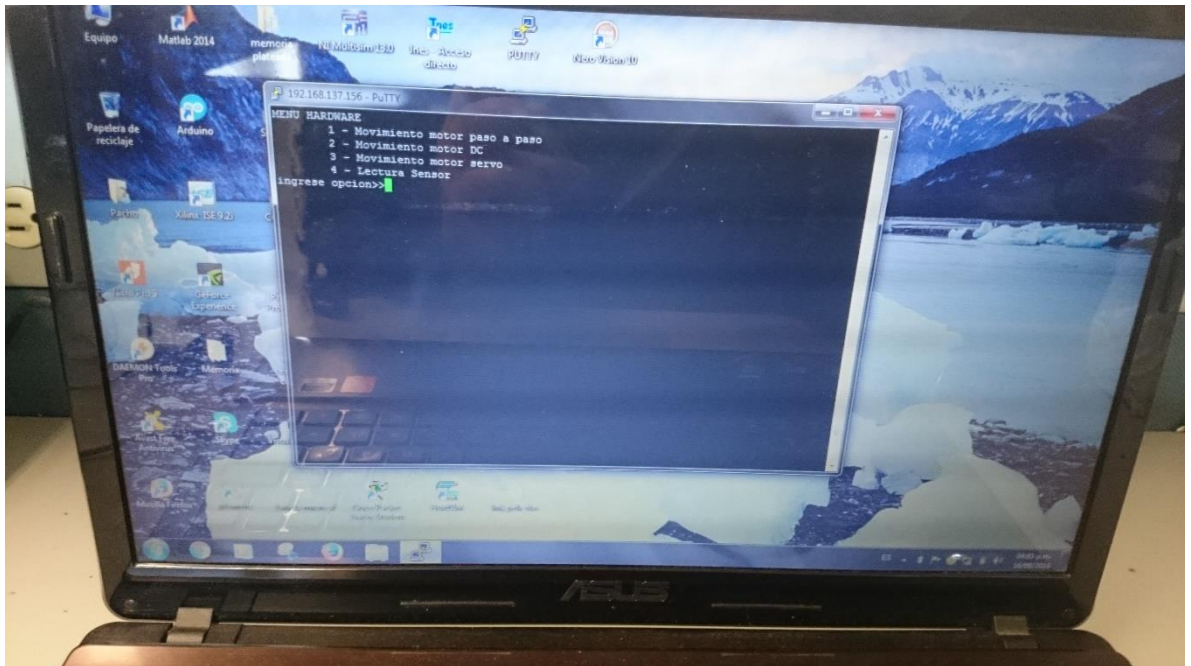
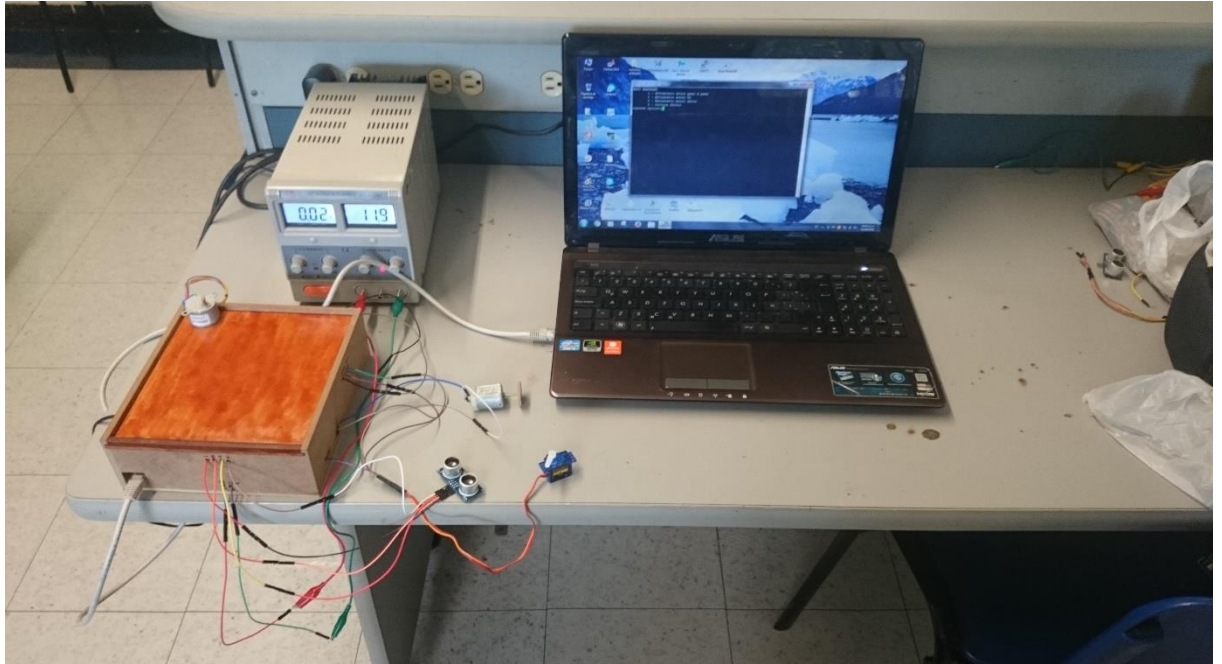
Parte superior del Hardware

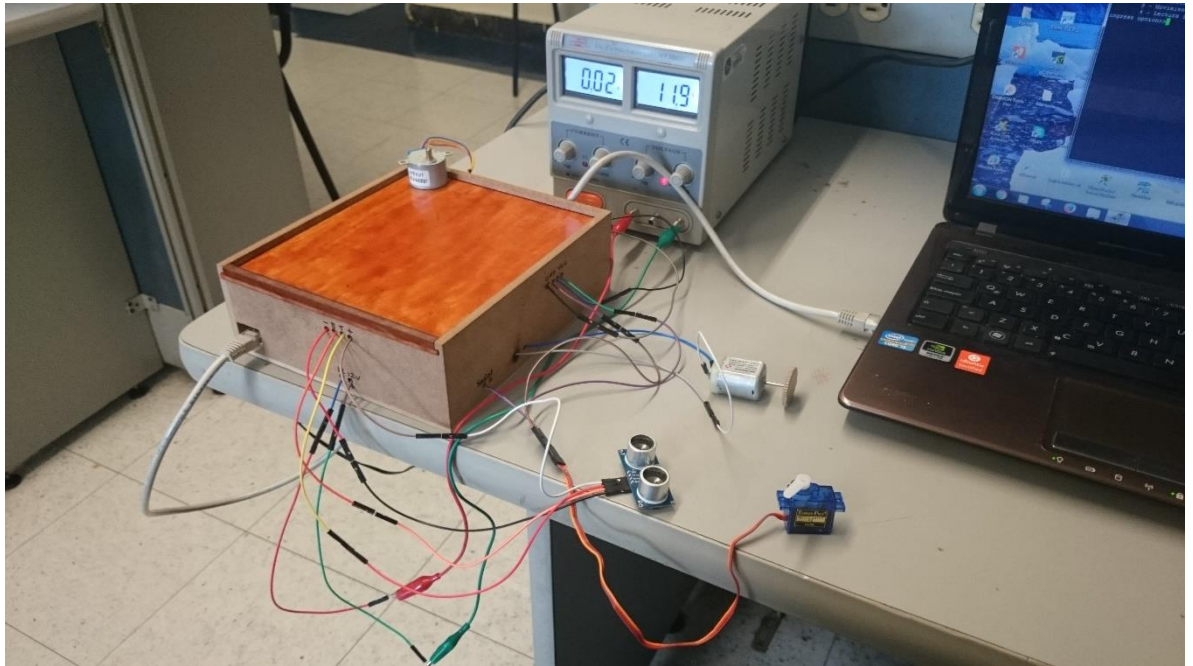


Parte interna del Hardware



Hardware implementado con la línea de comando CMD del PC





11. BIBLIOGRAFIA

Universidad de Murcia. (s.f.). ¿Qué es el aula virtual?

[Fecha de consulta: 6 de mayo de 2015]. Disponible en:

<<http://www.um.es/aulavirtual/primeros-pasos/que-es-el-aula-virtual/>>

Centro Nacional de desarrollo e investigación en tecnologías libres. (s.f.) Hacia una definición de Hardware libre. [Fecha de consulta: 28 de abril de 2015]. Disponible en:

<<http://www.cenditel.gob.ve/node/691>>

Aliane N, Fernández J, Martínez A y Ortiz J, Un laboratorio de Ingeniería de Control basado

en internet. Información Tecnológica – Vol. 18 N° 6 – 2007

Candelas F, Torres F, Gil P, Ortiz F, Puente S, Pomares J, (2004). Laboratorio virtual remoto para robótica y evaluación de su impacto en la docencia. Revista Iberoamericana de automática e informática industrial, Vol. 1, Núm. 2.

Turón R, Picos R, Roca M. Experimentos de electrónica a través de internet.

[Fecha de consulta: 21 de abril de 2015]. Disponible en:

<<http://www.upc.edu/euetib/xiicuiet/comunicaciones/din/comunicacions/113.pdf>>

Upton, C, (s.f). *Programming The Raspberry Pi.element14*. www.element14.com

Gonzales. ((2010)). *Python para todos*. España: Autoedición.

Downey, A.E. ((2002)). *Aprenda a Pensar como un Programador con Python*. Wellesley, Massachusetts: Green Tea Press

Gilfillian, I. (s.f.). *La biblia MySQL*. ANAYA MULTIMEDIA.

Kenkeiras. (s.f.) *De 0 a Python*. Hack x Crack

Wachenchauzer, R. (2011). *Algoritmos y Programación con lenguaje Python*. San Francisco, California, USA.