



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**
FACULTAD DE EDUCACIÓN TÉCNICA
PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

TEMA:

Diseño e implementación de un algoritmo con control PID y comunicación Bluetooth utilizando un Arduino Mega para el funcionamiento de un robot balance a distancia.

AUTOR:

Hormaza Rodríguez, Génesis Del Rocío.

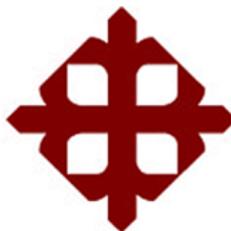
Componente práctico del examen complejo previo a la
obtención del título de **INGENIERO EN
TELECOMUNICACIONES**

REVISOR:

Palacios Meléndez, Edwin Fernando

Guayaquil, Ecuador

8 de marzo del 2018



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**
FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

CERTIFICACIÓN

Certificamos que el presente **componente práctico del examen complejo**, fue realizado en su totalidad por **Hormaza Rodríguez, Génesis del Rocío** como requerimiento para la obtención del título de **INGENIERO EN TELECOMUNICACIONES**.

REVISOR

Córdova Rivadeneira, Luis Silvio

DIRECTOR DE CARRERA

Heras Sánchez, Miguel Armando

Guayaquil, a los 8 del mes de marzo del año 2018.



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**
FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

DECLARACIÓN DE RESPONSABILIDAD

Yo Hormaza Rodríguez, Génesis Del Rocío

DECLARO QUE:

El componente práctico del examen complejo, Diseño e implementación de un algoritmo con control PID y comunicación Bluetooth utilizando un Arduino Mega para el funcionamiento de un robot balance a distancia, previo a la obtención del Título de Ingeniero en Telecomunicaciones, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de mi total autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Titulación referido.

Guayaquil, a los 8 del mes de Marzo del año 2018

LA AUTORA

HORMAZA RODRIGUEZ, GENESIS DEL ROCIO



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**
FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

AUTORIZACIÓN

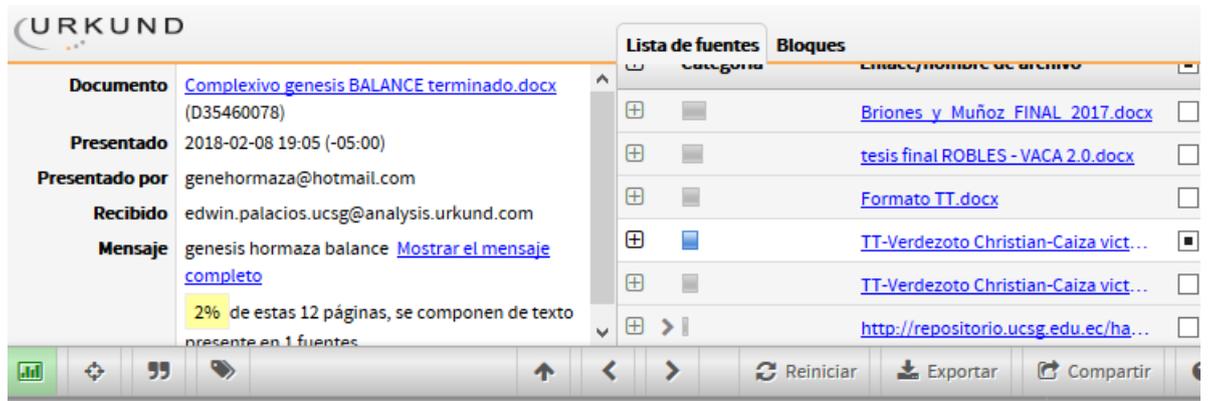
Yo, **Hormaza Rodríguez, Génesis Del Rocío** Autorizó a la Universidad Católica de Santiago de Guayaquil a la **publicación** en la biblioteca de la institución del **componente práctico del examen complejo, Diseño e implementación de un algoritmo con control PID y comunicación Bluetooth utilizando un Arduino Mega para el funcionamiento de un robot balance a distancia**, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y total autoría.

Guayaquil, a los 10 del mes de Febrero del año 2018

LA AUTORA

HORMAZA RODRIGUEZ, GENESIS DEL ROCIO

REPORTE DE URKUND



The screenshot shows the URKUND interface. On the left, a document summary is displayed:

- Documento:** [Complexivo genesis BALANCE terminado.docx](#) (D35460078)
- Presentado:** 2018-02-08 19:05 (-05:00)
- Presentado por:** genehormaza@hotmail.com
- Recibido:** edwin.palacios.ucsg@analysis.orkund.com
- Mensaje:** genesis hormaza balance [Mostrar el mensaje completo](#)

A yellow highlight indicates: "2% de estas 12 páginas, se componen de texto presente en 1 fuentes."

On the right, a table titled "Lista de fuentes" shows the following sources:

Lista de fuentes	Bloques
Icono	Enlace/nombre de archivo
+	Briones y Muñoz FINAL 2017.docx <input type="checkbox"/>
+	tesis final ROBLES - VACA 2.0.docx <input type="checkbox"/>
+	Formato TT.docx <input type="checkbox"/>
+	TT-Verdezoto Christian-Caiza vict... <input checked="" type="checkbox"/>
+	TT-Verdezoto Christian-Caiza vict... <input type="checkbox"/>
+	http://repositorio.ucsg.edu.ec/ha... <input type="checkbox"/>

At the bottom of the interface, there are navigation icons and buttons: "Reiniciar", "Exportar", and "Compartir". A notification in the bottom right corner says "0 Advertencia".



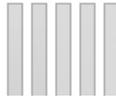
UNIVERSIDAD CATÓLICA DE SANTIAGO DE GUAYAQUIL FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO CARRERA DE INGENIERÍA EN TELECOMUNICACIONES TEMA:

Diseño e implementación de un

algoritmo con control PID y comunicación Bluetooth utilizando un Arduino Mega para el funcionamiento de un robot balance a distancia.

AUTOR: Hormaza Rodríguez, Génesis Del Rocio.

Componente práctico del examen complejo



previo a la obtención del grado de INGENIERO EN TELECOMUNICACIONES

DEDICATORIA

Dedico este proyecto de titulación con todo el corazón a mis padres y hermano que en situaciones difíciles y delicadas me apoyaron incondicionalmente siempre económica y moralmente, mi gratitud hacia ellos.

En especial a mi querido y amado abuelito José Rodríguez que ha luchado por su vida y me ha enseñado que, con amor, perseverancia y el respaldo del Dios todo se puede lograr, gracias al cariño y comprensión puedo decir que por el he logrado esto.

LA AUTORA

HORMAZA RODRIGUEZ, GENESIS DEL ROCIO.

AGRADECIMIENTO

Deseo agradecer a Dios en primer lugar por brindarme la oportunidad de vivir y haber alcanzado esta nueva meta en mi vida. Siendo el pilar fundamental para poder hacer todo lo que me proponga y por permitirme compartir este momento en especial para mi familia.

También a la Universidad Católica de Santiago de Guayaquil y a la Facultad de Educación Técnica para el Desarrollo y a cada uno de los maestros que en el transcurso de este tiempo formaron mi carrera impartiendo conocimientos y experiencias que de seguro será de vital importancia en mi carrera profesional.

Y a una de las personas más importantes en mi vida, Robert Vega, que fue participe en la parte más importante de mi trayecto académico, a mis compañeros del Club de Robótica y al coordinador del mismo, ya que compartimos experiencia, anécdotas, y apoyo, ya que no ha sido fácil llegar hasta aquí, tuvimos que dar nuestra máxima dedicación incluso haciendo a un lado nuestras actividades familiares y personales.

LA AUTORA

HORMAZA RODRIGUEZ, GENESIS DEL ROCIO.



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**
FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

TRIBUNAL DE SUSTENTACIÓN

f. _____

MANUEL DE JESUS ROMERO PAZ
DECANO

f. _____

MIGUEL ARMANDO HERAS SÁNCHEZ
DIRECTOR DE CARRERA

f. _____

EDWIN FERNANDO PALACIOS MELÉNDEZ
OPONENTE

ÍNDICE GENERAL

ÍNDICE DE FIGURAS.....	XI
ÍNDICE DE TABLAS	XII
Resumen	XIII
CAPÍTULO 1: DESCRIPCIÓN DEL COMPONENTE PRÁCTICO	14
1.1. Introducción.....	14
1.2. Objetivo General.	14
1.3. Objetivos Específicos.	14
CAPÍTULO 2: FUNDAMENTACIÓN TEÓRICA.	16
2.1. Controlador de derivación integral (PID).	16
2.2. Las características del controlador PID.....	18
2.3. Comunicación Bluetooth.....	18
2.3.1. Protocolos.....	19
2.3.2. Especificaciones.....	19
2.3.3. Ventajas y Desventajas de Bluetooth.	20
2.3.4. La Seguridad en Bluetooth.	21
2.4. Sistema de microcontroladores.	21
2.4.1. Memoria RAM.....	22
2.4.2. Memoria ROM.	22
2.4.3. Memoria EEPROM	23
2.4.4. Memoria EEPROM Flash	23
2.4.5. Arquitectura de los microcontroladores.	23
2.5. Microcontroladores PIC.....	23
2.5.1. Propiedades habituales de un microcontrolador PIC.....	24
2.5.2. Conceptos básicos.	24
2.6. Robótica Móvil.....	25

CAPÍTULO 3: DESARROLLO DEL COMPONENTE PRÁCTICO.....	26
3.1. Diseño y ejecución del robot balance.....	26
3.2. Tarjeta controladora Arduino Mega.....	27
Conclusiones	38
Recomendaciones.....	39
Referencias bibliográficas	40

ÍNDICE DE FIGURAS

Capítulo 2:

Figura 2. 1: Punto de ajuste y ángulo de inclinación real del robot.	16
Figura 2. 2: Representación de la tecnología Bluetooth.	19
Figura 2. 3: Conjunto y miembros de protocolos en bluetooth.	19
Figura 2. 4: Diagrama de segmentación de un microcontrolador.....	21
Figura 2. 5: Cuadro de bloques de un microcontrolador.	22
Figura 2. 6: Demostración de una palabra de 8 bits.	24

Capítulo 3:

Figura 3. 1: Diseño y chasis del prototipo.	26
Figura 3. 2: Motores y llantas sujetos a plataforma.....	27
Figura 3. 3: Plataforma que soporta la batería Lipo	27
Figura 3. 4: Batería de lipo (Grafeno).	27
Figura 3. 5: Tarjeta Arduino Mega.	28
Figura 3. 6. Visualización de Acelerómetro.....	30
Figura 3. 7: Giroscopio MPU 6050.....	31
Figura 3. 8: Diagrama del Giroscopio	32

ÍNDICE DE TABLAS

Capítulo 2:

Tabla 2. 1: Efecto de la sintonización PID.....	18
---	----

Capítulo 3:

Tabla 3. 1 Resumen del microcontrolador AT MEGA.....	28
--	----

Resumen

El objetivo principal del presente componente práctico del examen complejo fue centralizar la exploración de la tecnología de la robótica móvil, incitando a la investigación en el área de las telecomunicaciones. También, se realizó la búsqueda bibliográfica del segmento físico como programable (hardware y software) para la ejecución del autómata de la categoría robot balance o equilibrio. Se describieron los fundamentos teóricos de los microcontroladores y la huella que han dejado, así como los estándares y procesos que son importantes para la implementación del robot balance y de cómo este tipo de robot se incluyen en competencias de robótica. Posteriormente, se realizó el desarrollo estructural del robot y de sus componentes electrónicos, en especial de la tarjeta controladora y bluetooth. Después, se realizó el algoritmo del robot balance, así como programación en lenguaje de alto nivel como IDE Arduino que permitió mantener el equilibrio mediante el uso de controladores PID.

Palabras claves: PID, BLUETOOTH, ARDUINO, MICROCONTROLADORES, ROBÓTICA MÓVIL, AUTONOMO, CONTROL INALÁMBRICO.

CAPÍTULO 1: DESCRIPCIÓN DEL COMPONENTE PRÁCTICO

1.1. Introducción.

En este capítulo se va a conocer una breve reseña sobre el robot balance. El robot auto-equilibrante de dos ruedas se basa en un sistema de péndulo invertido, es dinámicamente estable pero estáticamente inestable. El robot contiene varias teorías de física y control. Este proyecto describe el modelado del robot de dos ruedas con auto equilibrado, diseña el controlador de robot usando PID e implementa el controlador en el robot.

En este proyecto, se utiliza una unidad de medida inercial (IMU), que combina la medición del acelerómetro y del giroscopio para estimar y obtener el ángulo de inclinación del robot. El controlador PID se aplica para corregir el error entre el punto de ajuste deseado y el ángulo de inclinación real y ajustar la velocidad del motor de cc en consecuencia para equilibrar el robot.

El resultado obtenido muestra que el controlador PID puede equilibrar el robot de manera aceptable, pero con algunas limitaciones. El resultado de la simulación del modelo se compara con el hardware desarrollado y se analiza el rendimiento del controlador. Además, se realiza el ajuste de PID mediante el método heurístico y se puede ver claramente una mejora en términos de equilibrio del robot.

1.2. Objetivo General.

Desarrollar el robot de dos ruedas basado en el concepto de péndulo invertido mediante el uso de un controlador PID y comunicación bluetooth.

1.3. Objetivos Específicos.

- ✓ Realizar la simulación de un robot de equilibrio de dos ruedas basado en su modelo matemático existente con los parámetros reales del robot.
- ✓ Realizar el diseño e implementación el prototipo de robot de equilibrado de dos ruedas con controlador PID.

- ✓ Evaluar el rendimiento del robot autoequilibrado desarrollado utilizando un enfoque estándar.

CAPÍTULO 2: FUNDAMENTACIÓN TEÓRICA.

2.1. Controlador de derivación integral (PID).

El algoritmo de control que se usa para mantener su posición de equilibrio en el robot de dos ruedas autoequilibrante fue el controlador PID. El controlador proporcional, integral y derivativo (PID) es bien conocido como un controlador de tres términos. El controlador de Derivación integral proporcional (PID) es un mecanismo de retroalimentación de bucle de control ampliamente utilizado en la industria. El controlador intenta ajustar y corregir el error entre el proceso medido y el proceso deseado y las medidas correctivas de salida para ajustar el proceso en consecuencia. Este controlador debe ejecutarse con la frecuencia suficiente y al mismo tiempo dentro del rango controlable del sistema. (Rahmawaty, 2015)

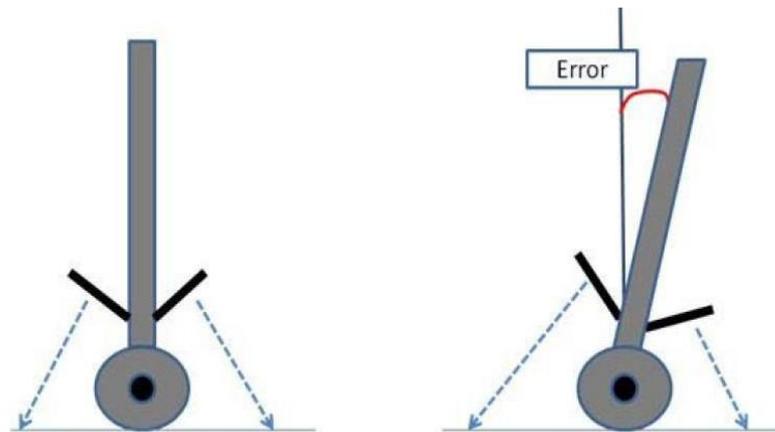


Figura 2. 1: Punto de ajuste y ángulo de inclinación real del robot.
Fuente:(Rahmawaty, 2015)

La figura 2.1 muestra el punto de ajuste y el ángulo de inclinación real del robot de dos ruedas. El error es la diferencia entre el ángulo de inclinación real y el ángulo de inclinación deseado (punto de ajuste). Como su nombre lo sugiere, el controlador PID contiene tres partes, que son el término proporcional, el término integral y el término derivado. Estos términos tienen un efecto diferente en la respuesta del motor de CC. Para equilibrar el robot, el punto de ajuste del robot debe ser de 0° .

De la figura 2.1 se deduce la ecuación cuando el robot se inclina hacia el frente:

$$\text{Error} = \text{Lectura actual del sensor frontal} \\ - \text{punto de referencia del sensor frontal}$$

La ecuación del error cuando el robot se inclina hacia el lado posterior es:

$$\text{Error} = \text{Punto de referencia del sensor de retroceso} - \\ \text{lectura del sensor de retroceso de corriente}$$

A continuación, se encuentran las ecuaciones involucradas en el cálculo del PID de salida:

$$\text{Termino de salida proporcional} = K_p * \text{Error}$$

$$\text{Termino integral de salida} = k_{pi} * \text{Suma de error} * T \\ = k_{pi} * \text{Suma de error} * T$$

$$\text{Termino integral de salida} = k_p * k_d * (\text{Error} - \text{Error anterior})/T \\ = \left(K_p * \frac{K_d}{T} \right) * (\text{Error} - \text{Error anterior})$$

La simplificación de la fórmula es la siguiente.

$$\text{Termino proporcional de salida} = K_p * \text{Error}$$

$$\text{Termino integral de salida} = K_i * (\text{Suma de error})$$

$$\text{Termino integral de salida} = K_d * (\text{Error} - \text{Error anterior})$$

En general, el controlador PID de salida para el sistema de control de equilibrio será:

$$\text{Controlador PID de salida} \\ = \text{Termino proporcional de salida} \\ + \text{Termino intergral de salida} \\ + \text{Termino diferencial de salida}$$

El ángulo real es el ángulo instantáneo del robot de vez en cuando. Este ángulo real se mide mediante la Unidad de medición inercial (IMU), que produce señales de salida digitales. Al comparar con el punto de ajuste deseado, se obtuvo el error, se obtiene la diferencia entre el punto de ajuste

deseado y el ángulo real. El error será alimentado al controlador PID. El controlador PID procesará, calculará y generará la salida de velocidad correspondiente para controlar el motor de CC, a fin de lograr el equilibrio de la manera correcta.

2.2. Las características del controlador PID.

Un controlador proporcional (K_p) tendrá el efecto de reducir el tiempo de subida que se reducirá, pero nunca se eliminará, el error de estado estacionario. Un control integral (K_i) tendrá el efecto de eliminar el error de estado estable para una entrada constante o por pasos, pero puede hacer que la respuesta transitoria sea más lenta. Un control derivado (K_d) tendrá el efecto de aumentar la estabilidad del sistema, reduciendo el sobre impulso y mejorando la respuesta transitoria. Los efectos de cada uno de los parámetros del controlador, K_p , K_d y K_i en un sistema de circuito cerrado se resumen en la tabla 2.1.

Tabla 2. 1: Efecto de la sintonización PID

Respuesta	Incremento de tiempo	Incremento	Tiempo de estabilizacion	Estado estable error
Kp	Disminuye	Incrementa	Cambio menor	Disminuye
Ki	Disminuye	Incrementa	Disminuye	Eliminar
Kd	Cambio Menor	Disminuye	Disminuye	Sin cambio

Fuente: (Junoh, 2015)

Realizado por: Autor

Hay que tener en cuenta que estas correlaciones pueden no ser exactamente precisas porque K_p , K_i y K_d dependen uno del otro. De hecho, cambiar una de estas variables puede cambiar el efecto de las otras dos.

2.3. Comunicación Bluetooth.

El "Bluetooth" es un protocolo muy importante en estos tiempos. Se tienen varias aplicaciones que utilizan este tipo de estándar de comunicación. Su origen viene de Suecia, unos de los países escandinavos, que fue implementado por una de las empresas más importantes como lo es

“Ericsson”. En las siguientes subsecciones se describen sus conceptos básicos, su funcionamiento y aplicaciones que se presentan en la actualidad. (García, Daniel, Cedeño, & Enrique, 2016)

En la figura 2.2 se visualiza la simbología de la técnica Bluetooth.



Figura 2. 2: Representación de la tecnología Bluetooth.
Fuente: (Cedeño, Verdezoto, 2016)

2.3.1. Protocolos.

Esta técnica constituye de cuatro capas que encierran protocolos, que son muy necesarios para los terminales Bluetooth. En la figura 2.3. se simbolizan los distintos protocolos de la tecnología Bluetooth. (Cedeño, Verdezoto, 2016)

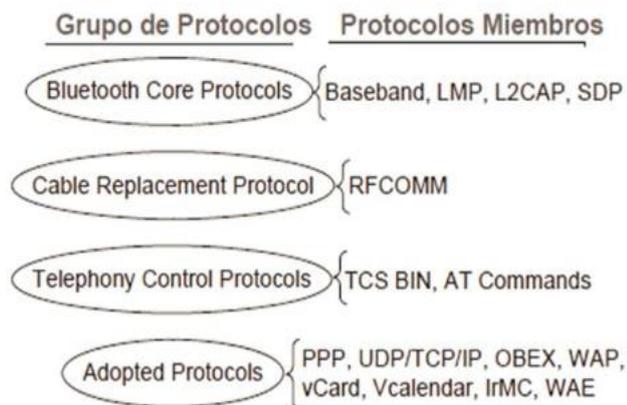


Figura 2. 3: Conjunto y miembros de protocolos en bluetooth.
Fuente:(Rahmawaty, 2015)

2.3.2. Especificaciones.

Esta técnica forma parte de las comunicaciones móviles, posee una corta distancia y su tarea primordial es proveer la comunicación de datos entre módulos inalámbricos y nuevos dispositivos de mano. Cedeño & Verdezoto, (2016) indican que Bluetooth se clasifica en 3 tipos.

- Tipo 1: Su cobertura alcanza aproximadamente 100m.
- Tipo 2: Su cobertura alcanza aproximadamente 10m.
- Tipo 3: Su alcance reducido es de 1m.

Huang & Rudolph, (2007) exponen las determinaciones sistemáticas, que son:

- Canales máximos de voz: 3 por piconet (64 kbps bidireccional)
- Banda de Frecuencia: 2.4 GHz
- Canales máximos de datos: 7 por piconet
- Interferencia: Bluetooth minimiza la interferencia potencial al emplear saltos rápidos en frecuencia (1600 veces por segundo).
- Velocidades de datos: hasta 721 kbps asimétrico o 433.9 kbps simétrico.
- Seguridad: Si, en la capa de enlace.

Esta industria facilita establecer comunicaciones habiendo impedimentos en presencia, en longitudes de 100m por ejemplo. Hay diferentes tipos de aplicaciones y éstas podrían ser para reproducir archivos de música desde un teléfono móvil y de modo inalámbrico en un estéreo para carro. También sirve para conectar distintos tipos de dispositivos hacia máquinas de escritorio.(Huang & Rudolph, 2007)

2.3.3. Ventajas y Desventajas de Bluetooth.

Este estándar de comunicación fue hecho para hacer más fácil la vida del hombre, es así como manifestaron muchas más virtudes como los enlaces entre terminales con computadoras y dispositivos con dispositivos, sí estuvieron creados con este conjunto de técnicas. La conexión se puede visualizar cuando se enlazan los dispositivos a una cámara sin conexión física alguna. (Huang; Rudolph, 2007).

Sin embargo, este estándar de comunicación contaba con una desventaja la cual con el pasar del tiempo podía ser reemplazada. La distancia de conectividad no era la más competente, ya que estos dispositivos tenían la necesidad estar parcialmente cerca. Y si había algún tipo de impedimento dependiendo del material, este se cambiaba en un problema para la conexión.(Cedeño; Verdezoto, 2016).

2.3.4. La Seguridad en Bluetooth.

Huang & Rudolph, (2007). Revelaron que, para conservar una reserva de privacidad en la recepción y envío de investigación, esta técnica brinda mecanismos de encriptación. El cifrado sobrelleva extensiones de 64 bits o menos, por si alguien no autorizado pretende acceder a la indagación y cambiarla, entonces se tienen 3 niveles existentes, que son:

- ✓ Nivel de seguridad A: No seguro.
- ✓ Nivel de seguridad B: Asistencia de seguridad designada.
- ✓ Nivel de seguridad C: Enlace de seguridad designada.

2.4. Sistema de microcontroladores.

Hoy en la actualidad las necesidades que tiene el ser humano en su búsqueda por hacer la vida más fácil y sencilla dieron lugar a mejoras de estas técnicas digitales. Se tiene que un ordenador es un ejemplo básico de un microprocesador, con su procedimiento de mandos, software y hardware tomando en cuenta las instrucciones que demande el usuario. El microcontrolador, que no es más que un conjunto de sujetos dentro de un circuito integrado, tal como se muestra en la figura 2. 4.

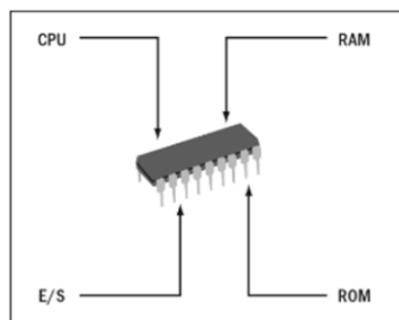


Figura 2. 4: Diagrama de segmentación de un microcontrolador
Fuente: (Cedeño; Verdezoto, 2016).

Los microcontroladores poseen un cargo primordial, el cual dirigir sus tareas. Así mismo, dentro de los circuitos se encuentran sus mismas memorias. En el ámbito económico los microcontroladores son mucho más abordables que un microprocesador. Debido a su arquitectura presenta bajos niveles de interferencia. Por consiguiente, a la hora de su repartición será más vertiginoso. Sin aludir que la conservación de energía es de mucho más trascendencia.

Para poder utilizar un microcontrolador se debe tener bases del conocimiento de lenguajes de programación de alto nivel. Dichos lenguajes son materiales muy ventajosos para el consumidor. Sujeta a varios usos, los dispositivos desde un ordenador de electrodomésticos hasta los frenos de un auto (Rahmawaty, 2015). En la figura 2.5 se ilustra un cuadro de bloques básico de un microcontrolador.

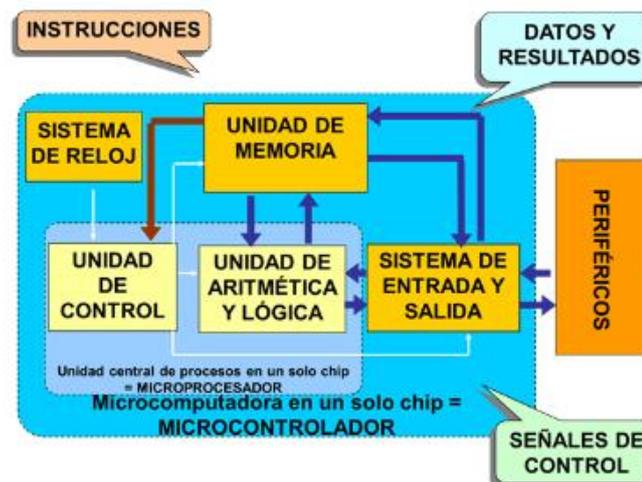


Figura 2. 5: Cuadro de bloques de un microcontrolador.
Fuente: (Cedeño; Verdezoto, 2016)

Hay una técnica de microcontroladores que se divide en dos conjuntos las cuales son memorias internas y externas. En la interna se acumula información que no se pierden al alejar la fuente; en la memoria externa se recopilan la información que pueden ser borrados al conectarse a la fuente.(Cedeño; Verdezoto, 2016).

2.4.1. Memoria RAM.

RAM (Random Access Memory), logra un alcance de 256 bytes, podría cambiar dependiendo mucho el aparato o el dispositivo, es considerada una memoria ligera por su propiedad de perdida de información cuando se la aísla de la fuente.(Cedeño; Verdezoto, 2016)

2.4.2. Memoria ROM.

ROM (Read Only Memory), ya que es programada desde que se fabrica, los datos que se guardan no se pueden alterar y son constantes.(Cedeño; Verdezoto, 2016)

2.4.3. Memoria EEPROM

EEPROM (Electrically Erasable Programmable Read Only Memory), esta clase de memoria es más amable con el consumidor programar y grabar información. Y por sus particulares características dificultosas EEPROM tiende a ser muy lenta normalmente. (Cedeño; Verdezoto, 2016)

2.4.4. Memoria EEPROM Flash

Es una versión mejorada de la memoria EEPROM con una alta velocidad ya que tiene más el poder programar y guardar información; conserva la capacidad de reprogramar información y datos. (Cedeño; Verdezoto, 2016)

2.4.5. Arquitectura de los microcontroladores.

Se operan dos tipos: arquitectura Von Neuman y Harvard.

Arquitectura Harvard.

Este ejemplo de arquitectura generalmente se encuentra en los PIC, tiene por separado las instrucciones de los datos, pero cada una conectada a un sistema de buses para tener fácil acceso a las diferentes operaciones que se tengan que realizar al mismo tiempo. (Cedeño; Verdezoto, 2016)

Arquitectura Von Neumann.

En los inicios era un diseño muy usado en las técnicas computacionales, y pasando el tiempo la tecnología tuvo que ser desplazada por otra arquitectura.

2.5. Microcontroladores PIC.

El microcontrolador PIC concierne en los modelos fabricados especialmente por la compañía de MICROCHIP TECHNOLOGY INC. La cual abarca una gran parte del mercado. Si se habla de los microcontroladores PIC, se debe saber que tienen características generales como: su valor accesible, mínimo consumo de potencia, mayor velocidad y más fácil de programar. (Pérez; Areny, 2007).

2.5.1. Propiedades habituales de un microcontrolador PIC.

Como se ha visto, la mayoría de microcontroladores se hicieron con una mayor capacidad en su memoria para que sus datos tengan mejor desarrollo. “La memoria de presentación está constituida en palabras de 12,14 o 16 bits mientras que la memoria de información está formada por registro de 8 bits”.(Pérez; Areny, 2007)

En los microcontroladores PIC se adiciona una fuente con una capacidad restringida que depende del modelo del PIC. En cuanto a la ubicación con la memoria no tiene relación con el microcontrolador.(Cedeño; Verdezoto, 2016).

Todos los microcontroladores PIC cuentan con muchas clases, que se dividen en gamas altas y bajas. Los microcontroladores de gama alta en su memoria de datos tienen acceso a información sin embargo en la gama baja no sufren ningún tipo de interrupción.(Pérez; Areny, 2007). Estos, también son llamados microcontroladores RISC.

2.5.2. Conceptos básicos.

“La memoria es la unión de celdas o emplazamientos las cuales asemejan por su dirección. En cada celda se almacena una palabra. Una palabra es la unidad lógica de datos recopilada en una celda de la memoria” (Cedeño; Verdezoto, 2016).

La figura 2.6 muestra la formación de palabras de 8 bits y de manera análoga con palabras de 12, 14 y 16 bits.

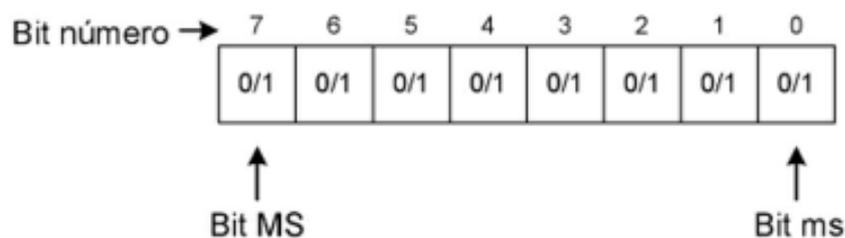


Figura 2. 6: Demostración de una palabra de 8 bits.
Fuente:(Cedeño; Verdezoto, 2016).

Esta memoria tiene que ser organizada linealmente ya que tiene un papel muy importante. La misma que es llamada también “paginas” (Cedeño ; Verdezoto, 2016).

2.6. Robótica Móvil.

Los androides portátiles son instrumentos de movilización automática, es decir, un programa mecánico que cuenta de una manera de translación que es capaz de navegar por medio de un ambiente de trabajo, con cierto grado de importancia para su descentralización de cargas. También son usados en diversas aplicaciones que están ligadas con actividades riesgosas o nocivas para la humanidad.

Tradicionalmente las aplicaciones de la robótica están en los sectores manufactureros más desarrollados de la producción masiva: industria del automóvil, transformaciones metálicas, industria química, etc. Es una disciplina que está más especializada en realizar sistemas que le permitan al hombre realizar tareas de todo tipo, desde las más fáciles hasta las más complicadas

CAPÍTULO 3: DESARROLLO DEL COMPONENTE PRÁCTICO.

3.1. Diseño y ejecución del robot balance.

Al implementar el robot balance de dos ruedas usa, motores As long Dc a 12 V con 126 RPM, una tarjeta Arduino MEGA basado en Shield balance, , driver L298P, un giroscopio MPU6050, Puente H 1AM, batería de lipo 3 celdas (grafeno), y Bluetooth low energy modelo 4.0. Cabe recalcar que el robot será manipulado vía celular inteligente, PC. En la figura 3.1 se muestra el diseño en AutoCAD del chasis del robot de balance.

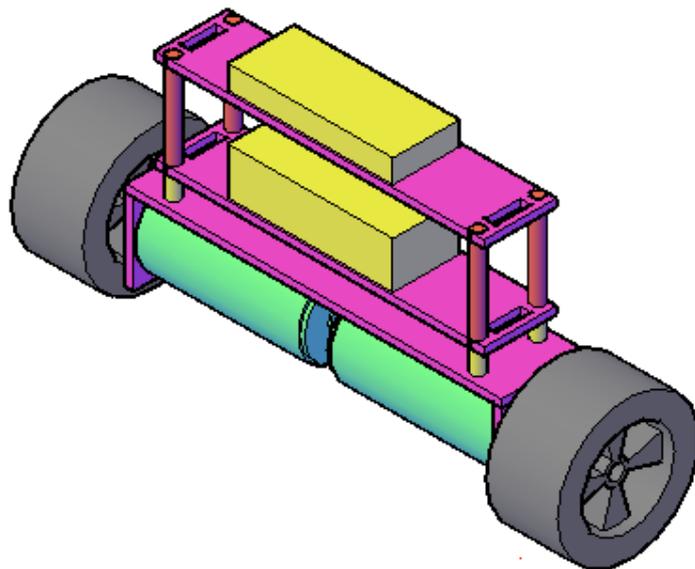


Figura 3. 1: Diseño y chasis del prototipo.
Elaborado por: Autor.

Ahora se detalla paso a paso el ensamblaje del robot balance. Está compuesto de soportes hechos con acrílico, por su facilidad y precio fue el material designado, asimismo se pudo realizar los cortes en una maquina CNC según las especificaciones correspondientes. Se proyectó una plataforma como parte principal y fundamental del robot. La primera plataforma soporta una batería de lipo 3 celdas 11.1 V, 1000 MAh, los motores aslong DC de 12V, y la última plataforma es para la parte electrónica de control.

En la figura 3.2, se visualiza los motores sujetos a la plataforma. Estos motores fueron elegidos porque incluyen encoders en sus ejes, el cual nos

aprueba ejecutar el algoritmo de programación, es decir, que corrige y mantiene el robot en equilibrio.

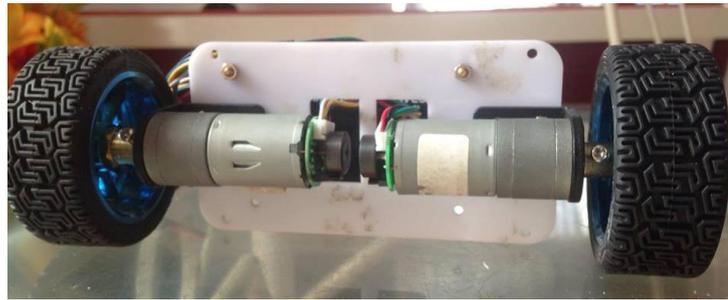


Figura 3. 2: Motores y llantas sujetos a plataforma
Elaborado por: Autor.

La siguiente plataforma soporta la batería de lipo con tornillos y pequeños soportes, tal como se muestra en la figura 3.3. Es importante que pueda estar sujeto a la plataforma para que la estructura no tenga ningún tipo de daño cuando ejecute ciertos movimientos

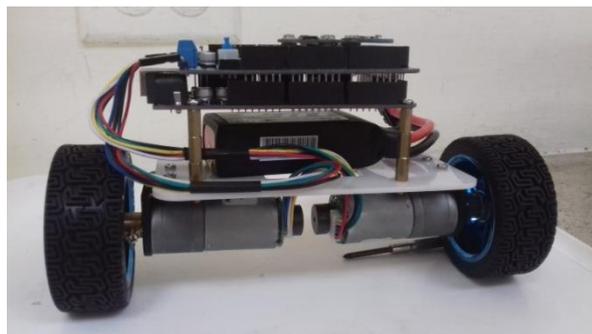


Figura 3. 3: Plataforma que soporta la batería Lipo
Elaborado por: Autor.



Figura 3. 4: Batería de lipo (Grafeno).
Elaborado por: Autor

3.2. Tarjeta controladora Arduino Mega.

Arduino Mega es una placa de microcontrolador basada en ATmega1280 (hoja de datos). Tiene 54 pines digitales de entrada / salida (de los cuales 14 se pueden usar como salidas PWM), 16 entradas analógicas, 4 UART (puertos serie de hardware), un oscilador de cristal de 16 MHz, una conexión USB, un conector de alimentación, un encabezado ICSP, y un botón

de reinicio. Contiene todo lo necesario para soportar el microcontrolador; simplemente conéctelo a una computadora con un cable USB o con un adaptador de CA a CC o batería para comenzar. El Mega es compatible con la mayoría de los escudos diseñados para el Arduino Duemilanove o Diecimila. (GOILAV & LOI, 2016).

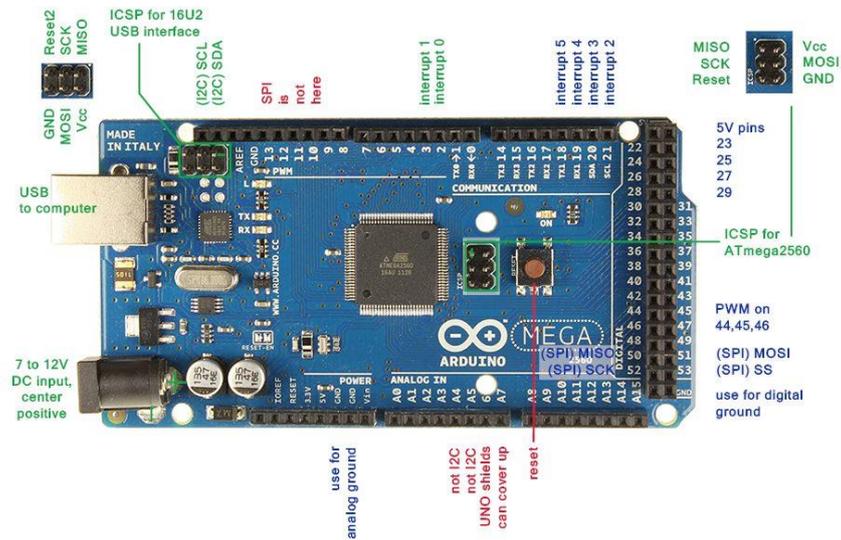


Figura 3. 5: Tarjeta Arduino Mega.
 Autor: (GOILAV & LOI, 2016)

Tabla 3. 1 Resumen del microcontrolador AT MEGA.

Microcontrolador	ATmega1280
Funcionamiento del Voltaje	5V
Voltaje de entrada (recomendado)	7-12V
Voltaje de entrada Limite	6-20V
Pines digitales I/O	54 (of which 15 provide PWM output)
Pines de entrada analógica	16
DC corriente por I/O Pines	40 mA
DC Corriente por 3.3V Pines	50 mA
Flash Memory	128 KB of which 4 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Velocidad del reloj	16 MHz

Fuente: Elaborador por Autor

El Arduino Mega se alimenta a través de un enlace USB o con una fuente de alimentación externa. La fuente de poder se selecciona automáticamente. (Goilav; Loi, 2016)

La alimentación externa (no USB) puede provenir de un adaptador de CA a CC (wall-wart) o batería. El adaptador se puede conectar al enchufar un conector positivo de 2.1 mm en el conector de alimentación de la placa. Los cables de una batería se pueden insertar en los conectores Gnd y Vin del conector POWER.

La placa puede operar con un suministro externo de 6 a 20 voltios. Sin embargo, si se suministra con menos de 7 V, el pin de 5 V puede suministrar menos de cinco voltios y la placa puede ser inestable. Si usa más de 12 V, el regulador de voltaje puede sobrecalentarse y dañar la placa. El rango recomendado es de 7 a 12 voltios. Los pines de alimentación son los siguientes:

- ✓ **VIN.** El voltaje de entrada a la placa Arduino cuando está usando una fuente de alimentación externa (a diferencia de 5 voltios de la conexión USB u otra fuente de alimentación regulada). Puede suministrar voltaje a través de este pin o, si suministra voltaje a través del conector de alimentación, acceda a través de este pin.
- ✓ **5V.** La fuente de alimentación regulada utilizada para alimentar el microcontrolador y otros componentes en el tablero. Esto puede provenir de VIN a través de un regulador de a bordo, o puede ser suministrado por USB u otro suministro regulado de 5V.
- ✓ **3V3.** Un suministro de 3.3 voltios generado por el chip FTDI a bordo. El consumo máximo de corriente es de 50 mA.
- ✓ **GND.** Pines de tierra.

3.2.1. Memoria

El ATmega1280 tiene 128 KB de memoria flash para almacenar el código (de los cuales 4 KB se utilizan para el gestor de arranque), 8 KB de SRAM y 4 KB de EEPROM (que se pueden leer y escribir con la biblioteca EEPROM).

3.3. Acelerómetro.

El acelerómetro sirve para medir la aceleración, detectar vibraciones y también para la aceleración de la gravedad. La detección de movimiento se basa en 3 ejes, es decir, de derecha a izquierda, de arriba hacia abajo y de adelante hacia atrás. La aplicación de este sensor generalmente es para la medición de la velocidad del motor, la vibración del motor, la vibración en el edificio y la disección de velocidad con la influencia de la gravedad de la tierra. Ejemplos de acelerómetro aplicativo es como un dispositivo electrónico, instalación de seguridad en el vehículo.



Figura 3. 6. Visualización de Acelerómetro.
Elaborado: Autor.

3.4. Giroscopio.

El giroscopio sirve para medir o determinar la orientación de un objeto en función de la determinación del momento angular. Desde otro sentido, el giroscopio sirve para determinar el movimiento de acuerdo con la gravedad realizada por el usuario. Este giroscopio tiene un papel muy importante en el mantenimiento del equilibrio de un objeto como su uso en un avión que puede determinar la pendiente de los ejes x, y y z . La salida producida por el giroscopio es la velocidad angular en el eje x que se convierte en phi (Φ), el eje y es $theta$ (θ), y el eje z es psi (Ψ). Antes del uso, el giroscopio generalmente se usa en el primero al usar un péndulo cuya función es determinar el valor del factor o también puede mirar la hoja de datos del sensor utilizado.

El principio de funcionamiento de este giroscopio es que cuando el giroscopio gira el giroscopio tendrá un valor de salida. Si el giroscopio gira en

el sentido de las agujas del reloj en el eje Z, el voltaje de salida resultante disminuirá, mientras que si el giroscopio gira en sentido antihorario en el eje Z, la tensión de salida resultante se agrandará. En el momento en que el giroscopio no gira o está en reposo, la tensión de salida corresponderá al valor de compensación del giro sensor.

Para ver los datos del giroscopio con un microcontrolador, puede usar el puerto *ADC*. El valor de salida del sensor se convierte en radianes / segundo (*rad / s*) y luego se convierte de nuevo a grado / segundo (*grados / s*).



Figura 3. 7: Giroscopio MPU 6050
Elaborado: Autor.

3.4.1. Características de este MPU 6050.

- ✓ Sensibilidad del acelerómetro que se puede seleccionar de 2/4/8 samapai 16 g.
- ✓ La sensibilidad de un giróscopo seleccionable de 250/500/ 1000 a 2000 *grados / s*.
- ✓ Rango de 16 bits para ambos sensores.
- ✓ La sensibilidad de aceleración lineal del giroscopio es de 0.1 *grado / s*
- ✓ Velocidad de salida de datos de hasta 1000Hz, equipada con filtro de paso bajo digital y tiene una frecuencia angular máxima de 256Hz.

En la figura 3.8 se muestra el diagrama esquemático del Giroscopio MPU6050.

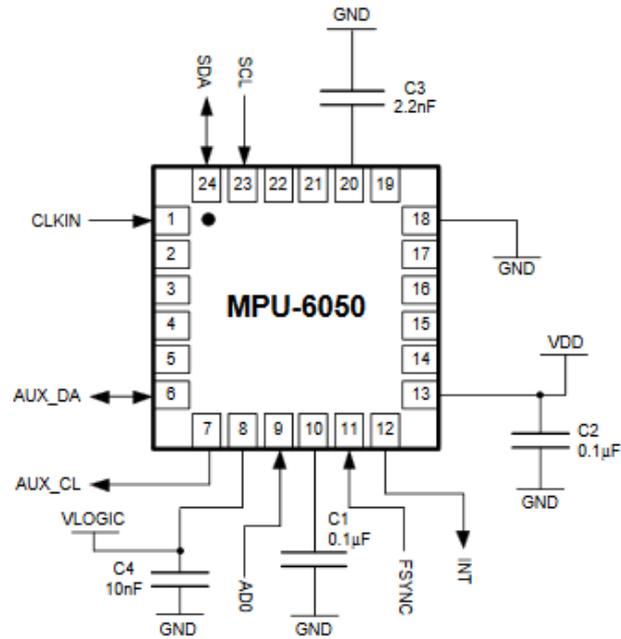


Figura 3. 8: Diagrama del Giroscopio
 Autor:(Cedeño, 2016).

Para el siguiente proyecto de titulación, el robot balance utilizará el sensor MPU6050 que nos permite controlar el equilibrio del robot balance. Para esto se debe utilizar una aplicación llamada SainSmartInstaBots que es compatible con los móviles inteligentes que utilizan sistemas Android y el mismo también debe de incluir un módulo Bluetooth para que se pueda manejar a distancia.

3.4.2. Programación androide balance.

Se declaran las librerías base a usar para el desarrollo del control PID en el balance:

```
include <FlexiTimer2.h>
#include <BalanceCar.h>
#include <KalmanFilter.h>
#include <I2Cdev.h>
#include <MPU6050_6Axis_MotionApps20.h>
#include <Wire.h>
```

Se definen las variables que se usaran para la programación del robot y los parámetros de set point del control PID:

```

//TB6612FNG
#define IN1M 6
#define IN2M 7
#define IN3M 12
#define IN4M 13
#define PWMA 4
#define PWMB 5
#define STBY 8

#define PinA_right 18 //INT05
#define PinA_left 2 //INT00

//Bluetooth
#define Left_Encoder 0x01
#define Right_Encoder 0x02
#define Left_PWM 0x03
#define Right_PWM 0x04
#define Balance_Angle 0x05
#define Upright_Kp 0x06
#define Upright_Ki 0x07
#define Upright_Kd 0x08
#define Speed_Kp 0x09

#define Speed_Ki 0x10
#define Speed_Kd 0x0A
#define Rotate_Kp 0x0B
#define Rotate_Ki 0x0C
#define Rotate_Kd 0x0D
#define Contrl_val 0x0E
#define angle_output 0x0F
#define speed_output 0x11

MPU6050 mpu;
BalanceCar balancecar;
KalmanFilter kalmanfilter;
int16_t ax, ay, az, gx, gy, gz;

double Outputs = 0;

double kp = 29.525, ki = 0.0, kd = 0.2765;
double kp_speed = 5.0225, ki_speed = 0.1528, kd_speed = 0.0;
double kp_turn = 23.1625, ki_turn = 0, kd_turn = 0.277;

double setp0 = 0;

```

A continuación, se muestran las líneas de programación para la creación de la función UART para el intercambio de información serial entre el dispositivo bluetooth y el microcontrolador Atmega 1280.

```

-
unsigned int Ctrl_val;
int Speed_val = 85;
char i=0;

void Uart_Send(unsigned char function,unsigned int value)
{
    unsigned int temp;
    i=0;
    temp=value;
    a[5]=function;
    a[6]=(value>>8) & 0xFF;
    a[7]=temp & 0xFF;
    while(i!=12)
    {
        Serial.write(a[i]);
        i++;
    }
}

void Uart_Recieve()
{
    if(Serial.available()>=12)
    {
        i=0;
    while(i!=12)
    {
        RxBuf[i]=Serial.read();
        i++;
    }
    Ctrl_val=RxBuf[6]*0x100+RxBuf[7];
    switch(RxBuf[5])
    {
        case Upright_Kp : kp=(double)Ctrl_val/1000;break;
        case Upright_Ki :break;
        case Upright_Kd : kd=(double)Ctrl_val/1000;break;
        case Speed_Kp :kp_speed=(double)Ctrl_val/1000;break;
        case Speed_Ki :ki_speed=(double)Ctrl_val/1000;break;
        case Speed_Kd :break;
        case Rotate_Kp :kp_turn=(double)Ctrl_val/1000;break;
        case Rotate_Ki :break;
        case Rotate_Kd :kd_turn=(double)Ctrl_val/100;break;
        case Contrl_val:
            switch(RxBuf[7])
            {
                case 0x01: front = Speed_val;    break;
                case 0x02: back = -Speed_val;    break;
                case 0x03: turnl = 1;front = 0;back = 0;    break;
                case 0x04: turnr = 1;front = 0;back = 0;    break;
            }
        }
    }
}

```

Después, se crea la función del ángulo de salida para realizar un control proporcional y derivativo de la posición del robot con respecto a su origen. A continuación, se especifica la configuración:

```

void angleout()
{
    balancecar.angleoutput = -kp * (kalmanfilter.angle + angle0) - kd * kalmanfilter.Gyro_x;//PD
}

```

Se establece la funcion Inter que realiza el control PID de velocidad en referencia al ángulo y el giro del robot en el instante actual para determinar postura y error.

```

void inter()
{
    sei();
    countpluse();
    mpu.getMotion6(&ax, &ay, &az, &gx, &gy, &gz); //MPU6050
    kalmanfilter.Angletest(ax, ay, az, gx, gy, gz, dt, Q_angle, Q_gyro, R_angle, C_0, KI);
    angleout();

    speedcc++;
    if (speedcc >= 8)
    {
        Outputs = balancecar.speedpiout(kp_speed, ki_speed, kd_speed, front, back, setp0);
        speedcc = 0;
    }
    turncount++;
    if (turncount > 4)
    {
        turnoutput = balancecar.turnspin(turnl, turnr, spinl, spinr, kp_turn, kd_turn, kalmanfilter.Gyro_z);
        turncount = 0;
    }
    balancecar.posture++;
    balancecar.pwm(Outputs, turnoutput, kalmanfilter.angle, kalmanfilter.angle6, turnl, turnr, spinl, spinr, front, back, kalmanfilter.accelz, IN1M, IN2M, IN3M, IN4M, PWMA, PWMB);
    timer++;
}

```

Posteriormente, se procede a realizar la programación para la creación de la función bluetooth para la comunicación entre módulo de transmisión y módulo de recepción:

```

void Bluetooth()
{
  if(timer==200)
  {
    Uart_Send(Left_Encoder, (unsigned int)lz);
    delay(2);
    Uart_Send(Right_Encoder, (unsigned int)rz);
    delay(2);
    Uart_Send(Left_PWM, abs(balancecar.pwm2));
    delay(2);
    Uart_Send(Right_PWM, abs(balancecar.pwm1));
    delay(2);
    Uart_Send(Balance_Angle, (unsigned int)kalmanfilter.angle*10);
    delay(2);
    Uart_Send(angle_output, (unsigned int)balancecar.angleoutput*10);
    delay(2);
    Uart_Send(speed_output, (unsigned int)Outputs*10);
    timer=0;
  }
}

```

Las siguientes líneas de programación se realizar la configuración de variables de entrada/salida y la definición de inicio de las mismas:

```

void setup() {
  // Establecimiento de variables
  pinMode(IN1M, OUTPUT);
  pinMode(IN2M, OUTPUT);
  pinMode(IN3M, OUTPUT);
  pinMode(IN4M, OUTPUT);
  pinMode(PWMA, OUTPUT);
  pinMode(PWMB, OUTPUT);
  pinMode(STBY, OUTPUT);

  //Inicializacion de variable
  digitalWrite(IN1M, 0);
  digitalWrite(IN2M, 1);
  digitalWrite(IN3M, 1);
  digitalWrite(IN4M, 0);
  digitalWrite(STBY, 1);
  analogWrite(PWMA, 0);
  analogWrite(PWMB, 0);

  pinMode(PinA_left, INPUT);
  pinMode(PinA_right, INPUT);
}

```

Función Loop de Arduino que inicia después de la función Setup:

```
void loop() {  
  attachInterrupt(digitalPinToInterrupt(5), Countright, CHANGE);  
  attachInterrupt(digitalPinToInterrupt(0), Countleft, CHANGE);  
  Uart_Recieve();  
  Bluetooth();  
}
```

Conclusiones

- A través de la descripción teórica de los microcontroladores se pudo fundamentar la implementación del robot balance.
- La implementación ayudó a fomentar los conocimientos adquiridos en clases, el cual será destinado para su uso en competencias de robótica nacionales e internacionales por parte del Club de Robótica “ROBOFET” de la Universidad Católica de Santiago de Guayaquil en la Carrera de Ingeniería en Telecomunicaciones.
- Se realizaron pruebas de los datos de respuesta del robot sin controlador y usando el controlador PID. También, se realizaron pruebas con PID con el parámetro K_p cambiado, y parámetros K_i y K_d fijos. Después, se realizaron con otros parámetros de K_i modificado, y parámetros K_p y K_d corregidos. Y, finalmente con parámetro K_d cambiado, y parámetros K_p y K_i fijos. Los resultados obtenidos permitieron afinar la programación del control PID para lograr el balance o equilibrio del robot.
- Las pruebas también se realizaron al proporcionar interferencia al robot, la misma que permitió la estabilización del robot balance.

Recomendaciones.

- El robot balance es muy frágil al usarlo, así que se debe manipular de la manera más cuidadosa los materiales con los que se implementaron, con esto se refiere a la electrónica que posee ya que es muy sensible.
- Utilizar el sistema de radio control (RC), en vez del aplicativo del dispositivo móvil ya que esto daría más facilidades de manejar el robot debido a que opera a mayor distancia, mejor control y más ergonomía.

Referencias bibliográficas

- Cedeño, & Verdezoto. (2016). Implementación de dos robots autónomos y uno controlado mediante tecnología bluetooth para las categorías seguidor de línea con obstáculos, laberinto y balancín. UCSG. Recuperado a partir de <http://repositorio.ucsg.edu.ec/handle/3317/6607>
- García, C., Daniel, V., Cedeño, V., & Enrique, C. (2016). Implementación de dos robots autónomos y uno controlado mediante tecnología bluetooth para las categorías seguidor de línea con obstáculos, laberinto y balancín. Recuperado a partir de <http://repositorio.ucsg.edu.ec/handle/3317/6607>
- Goilav, N., & Loi, G. (2016). Arduino: Aprender a desarrollar para crear objetos inteligentes. Ediciones ENI.
- Huang, A. S., & Rudolph, L. (2007). Bluetooth Essentials for Programmers. Cambridge University Press.
- Junoh, S. A. (2015). Two-wheeled balancing robot controller designed using pid (masters). Universiti Tun Hussein Onn Malaysia. Recuperado a partir de <http://eprints.uthm.edu.my/7850/>
- Pérez, F. E. V., & Areny, R. P. (2007). Microcontroladores: fundamentos y aplicaciones con PIC. Marcombo.
- Rahmawaty, M. (2015). Balancing Robot Beroda Dua Menggunakan Metoda Kontrol Proporsional, Integral dan Derivatif. Universitas politecnica de catalnya barcelonatech. Recuperado a partir de <https://jurnal.pcr.ac.id/index.php/elementer/article/view/48>



Presidencia
de la República
del Ecuador



Plan Nacional
de Ciencia, Tecnología,
Innovación y Saberes



SENESCYT
Secretaría Nacional de Educación Superior,
Ciencia, Tecnología e Innovación

DECLARACIÓN Y AUTORIZACIÓN

Yo, **HORMAZA RODRIGUEZ, GÉNESIS DEL ROCÍO** con C.C: # 0950504548 autor del Trabajo de Titulación: **DISEÑO E IMPLEMENTACION DE UN ALGORITMO PID Y COMUNICACIÓN BLUETOOTH UTILIZANDO UN ARDUINO MEGA PARA EL FUNCIONAMIENTO DE UN ROBOT BALANCE A DISTANCIA** previo a la obtención del título de **INGENIERO EN TELECOMUNICACIONES** en la Universidad Católica de Santiago de Guayaquil.

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 entregar a la SENESCYT en formato digital una copia del referido trabajo de titulación 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 titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Guayaquil, 08 de marzo del 2018.

f. _____

Nombre: **HORMAZA RODRIGUEZ, GENESIS DEL ROCIO**

C.C: 095050454-8

REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA

FICHA DE REGISTRO DE TESIS/TRABAJO DE TITULACIÓN

TÍTULO Y SUBTÍTULO:	DISEÑO E IMPLEMENTACION DE UN ALGORITMO CON CONTROL PID Y COMUNICACIÓN BLUETOOTH UTILIZANDO UN ARDUINO MEGA PARA EL FUNCIONAMIENTO DE UN ROBOT BALANCE A DISTANCIA.		
AUTOR(ES)	HORMAZA RODRIGUEZ, GÉNESIS DEL ROCÍO.		
REVISOR(ES)/TUTOR(ES)	M. Sc. EDWIN F. PALACIOS MELÉNDEZ.		
INSTITUCIÓN:	Universidad Católica de Santiago de Guayaquil		
FACULTAD:	Facultad de Educación Técnica para el Desarrollo		
CARRERA:	Ingeniería en Telecomunicaciones		
TITULO OBTENIDO:	Ingeniero en Telecomunicaciones		
FECHA DE PUBLICACIÓN:	8 de marzo del 2018	No. DE PÁGINAS:	40
ÁREAS TEMÁTICAS:	Electrónica, Comunicaciones Inalámbricas, Microcontroladores		
PALABRAS CLAVES/ KEYWORDS:	PID, BLUETOOTH, ARDUINO, MICROCONTROLADORES, ROBÓTICA MÓVIL, AUTONOMO, CONTROL INALÁMBRICO.		
RESUMEN/ABSTRACT (150-250 palabras):			
<p>El objetivo principal del presente componente práctico del examen complejo fue centralizar la exploración de la tecnología de la robótica móvil, incitando a la investigación en el área de las telecomunicaciones. También, se realizó la búsqueda bibliográfica del segmento físico como programable (hardware y software) para la ejecución del autómata de la categoría robot balance o equilibrio. Se describieron los fundamentos teóricos de los microcontroladores y la huella que han dejado, así como los estándares y procesos que son importantes para la implementación del robot balance y de cómo este tipo de robot se incluyen en competencias de robótica. Posteriormente, se realizó el desarrollo estructural del robot y de sus componentes electrónicos, en especial de la tarjeta controladora y bluetooth. Después, se realizó el algoritmo del robot balance, así como programación en lenguaje de alto nivel como IDE Arduino que permitió mantener el equilibrio mediante el uso de controladores PID.</p>			
ADJUNTO PDF:	<input checked="" type="checkbox"/> SI	<input type="checkbox"/> NO	
CONTACTO CON AUTOR/ES:	Teléfono: +593-9-68847988	E-mail: genehormaza@hotmail.com	
CONTACTO CON LA INSTITUCIÓN: COORDINADOR DEL PROCESO DE UTE	Nombre: Palacios Meléndez Edwin Fernando		
	Teléfono: +593-9-68366762		
	E-mail: edwin.palacios@cu.ucsg.edu.ec		
SECCIÓN PARA USO DE BIBLIOTECA			
No. DE REGISTRO (en base a datos):			
No. DE CLASIFICACIÓN:			
DIRECCIÓN URL (tesis en la web):			