

**DISEÑO DE UN MODELO FUNCIONAL DE UN ROBOT MÓVIL QUE  
FAVOREZCA EL APRENDIZAJE DE CONCEPTOS BÁSICOS DE  
PROGRAMACIÓN EN LOS NIÑOS**

**JAMES SEBASTIÁN MURILLO ESTEBAN**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FÍSICO – MECÁNICAS  
ESCUELA DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA Y DE  
TELECOMUNICACIONES  
BUCARAMANGA**

**2018**

**DISEÑO DE UN MODELO FUNCIONAL DE UN ROBOT MÓVIL QUE  
FAVOREZCA EL APRENDIZAJE DE CONCEPTOS BÁSICOS DE  
PROGRAMACIÓN EN LOS NIÑOS**

**JAMES SEBASTIÁN MURILLO ESTEBAN**

**Proyecto de grado como requisito para optar el título de Ingeniero  
Electrónico**

**Director**

**CARLOS AUGUSTO FAJARDO ARIZA**

**Ingeniero Electrónico**

**Escuela de ingeniería Eléctrica y Electrónica**

**Codirector**

**FERNANDO ANTONIO ROJAS MORALES**

**Ingeniero de Sistemas**

**Escuela de ingeniería de sistemas e informática**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FÍSICO MECÁNICAS  
ESCUELA DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA Y DE  
TELECOMUNICACIONES  
BUCARAMANGA**

**2018**

## AGRADECIMIENTOS

*Agradezco a la vida por haberme brindado la oportunidad de lograr tan anhelado logro.*

*A mi madre por creer y seguir cada uno de mis caprichos.*

*A mi nona, que siempre tuvo fe y esperanza y con mucho orgullo esperaba este momento.*

*A mis familiares y amigos que creyeron en mí.*

*A el que considero mi mejor amigo, Andrés, que me brindo una gran ayuda cuando más lo necesitaba, que me enseñó que la humildad es más importante que el dinero.*

*A el grupo ERA, donde inicie mis primeros pasos y mis más grandes caídas, pero que gracias a esas enseñanzas fue posible cumplir este hito, al ingeniero y amigo Raúl, que gracias a su temple me hicieron de un mejor profesional.*

*A mi novia Andrea, una mujer con carácter santandereano que forjó una nueva y mejor forma de vivir en mí, que, con sus concejos, paciencia y amor, fue posible terminar este último tramo que tanto esfuerzo me tomó. Es a ti 'Ma' a la que realmente te debo las gracias por ser más que una compañera, una amiga incondicional.*

*James Sebastián Murillo Esteban*

## CONTENIDO

	<b>Pág.</b>
INTRODUCCIÓN .....	13
1. GENERALIDADES.....	15
2. ANTECEDENTES .....	17
3. DESARROLLO A NIVEL DE SISTEMA .....	23
4. IMPLEMENTACIÓN Y RESULTADOS .....	59
5. CONCLUSIONES.....	68
REFERENCIAS BIBLIOGRÁFICAS .....	69
BIBLIOGRAFÍA .....	71
ANEXOS .....	73

## LISTA DE FIGURAS

	<b>Pág.</b>
Figura 1. Diagrama desarrollo de proyecto .....	24
Figura 2. Diagrama Especificaciones Generales .....	25
Figura 3. Representación de un sistema RFID .....	26
Figura 4. Tag 16mm (125Khz) .....	27
Figura 5. Funcionamiento TCS3200 .....	28
Figura 6. Sensor TCS3200.....	28
Figura 7. Divisor de tensión.....	29
Figura 8. Módulo ESP8266 .....	31
Figura 9. Módulo bluetooth HC-05 .....	32
Figura 10. Módulo Xbee 1 mW.....	33
Figura 11. Encoder infrarrojo MiniQ robot .....	34
Figura 12. Encoder efecto hall con acople al eje.....	35
Figura 13. Señal de salida.....	35
Figura 14. Señal de salida.....	36
Figura 15. Raspberry pi 3 model B.....	37
Figura 16. Atmega328p.....	38
Figura 17. Movimientos unidireccionales .....	39
Figura 18. Ficha plana. Vista superior y perspectiva caballera .....	40
Figura 19. Ficha triangular. Vista superior y perspectiva caballera .....	40
Figura 20. Ficha hexagonal. Vista superior y perspectiva caballera .....	41
Figura 21. Ficha cubo. Vista superior y perspectiva caballera .....	41
Figura 22. Tablero de mando .....	43
Figura 23. Tablero de mando y niveles integrados .....	43
Figura 24. Tablero de mando y niveles separados .....	44
Figura 25. Tablero de mando individual, orientación horizontal .....	44
Figura 26. Tablero de mando individual, orientación vertical .....	45

Figura 27. Tracción por orugas .....	47
Figura 28. Robot tipo triciclo.....	47
Figura 29. Robot rover .....	48
Figura 30. Tapete cuadrado .....	49
Figura 31. Tapete circular .....	50
Figura 32. Tapete rectangular .....	50
Figura 33. Tapete triangular .....	51
Figura 34. Dimensiones ficha .....	53
Figura 35. Set de instrucciones.....	54
Figura 36. Tamaños de imán .....	55
Figura 37. Ubicación resistencia. Vista posterior .....	55
Figura 38. Ficha. Vista anterior .....	56
Figura 39. Propuesta color fichas.....	56
Figura 40. Tablero, Vista superior y perspectiva .....	57
Figura 41. Robot. Vista superior.....	57
Figura 42. Niveles 1,2,3 .....	58
Figura 43. Niveles 4,5,6 .....	58
Figura 44. Robot móvil, vista superior.....	59
Figura 45. PCB Tablero comandos, vista superior .....	60
Figura 46. Diseño Tablero de Comandos.....	61
Figura 47. Diseño Ficha .....	61
Figura 48. Tablero y PCB Integrada.....	62
Figura 49. Función lectura datos.....	63
Figura 50. Función identificación datos .....	64
Figura 51. Función transmission datos.....	65
Figura 52. Función principal.....	66
Figura 53. Recepción datos .....	67

## LISTA DE TABLAS

	<b>Pág.</b>
Tabla 1. Revisión del estado del arte juegos de programación.....	18
Tabla 2. Revisión del estado del arte software de programación.....	19
Tabla 3. Criterio de selección, concepto fichas .....	42
Tabla 4. Criterio de selección, concepto tablero .....	45
Tabla 5. Criterio de selección, robot.....	48
Tabla 6. Criterio de selección, tapete .....	51
Tabla 7. Características fichas .....	52
Tabla 8. Características tablero .....	52
Tabla 9. Características robot .....	53
Tabla 10. Características tapete .....	53

## LISTA DE ANEXOS

	<b>Pág.</b>
ANEXO A. Tarjeta interfaz electrónica .....	73
ANEXO B. Diseño 3D Fichas .....	77
ANEXO C. Manual de usuario.....	79

## RESUMEN

**TITULO:** Diseño de un modelo funcional de un robot móvil que favorezca el aprendizaje de conceptos básicos de programación en los niños.\*

**AUTOR:** James Sebastián Murillo Esteban\*\*

**PALABRAS CLAVE:** Programación, smartphones, consolas, videojuegos, robot móvil, neodimio, microcontroladores, bluetooth, divisor de tensión.

### DESCRIPCIÓN

En este proyecto se desarrolló un modelo funcional de un robot móvil con el propósito de favorecer el aprendizaje de conceptos básicos de programación en los niños. Se busca presentar una alternativa para que los niños desarrollen habilidades para la programación. Lo anterior teniendo en cuenta que el fácil acceso de los niños a dispositivos como smartphones, computadoras y consolas de videojuegos, y su difícil supervisión, está causando consecuencias negativas como el trastorno por déficit de atención, alteraciones del sueño, sobreexposición, entre otros, de manera que esta propuesta rompe con el paradigma del aprendizaje de la lógica computacional a través de una computadora y lo pone en manos de un dispositivo físico fuera de línea.

El robot móvil es programado a través una serie de fichas que le indican al robot cómo debe moverse. Cada ficha representa una instrucción puntual y la forma en la que se dispongan en el panel, repercutirá en la trayectoria que realizará el robot. Para el acople entre las fichas y el tablero se usaron imanes de neodimio, de manera que las piezas no sólo tengan una unión consistente, sino que sean fácilmente reemplazables y re-agrupables. El tablero y el robot cuentan con dos microcontroladores comunicados a través de Bluetooth en donde el primero, de acuerdo a la lectura que haga de cada ficha, codificará y enviará los datos al microcontrolador del robot, y este ejecutará las acciones correspondientes. Cada ficha tiene un divisor de tensión en particular que la diferencia de otras fichas y de esta manera, se obtiene un conjunto de piezas completo y variado. De esta forma se ofrece una alternativa de aprendizaje diferente a los niños.

---

\* Trabajo de grado

\*\* Facultad de Ingenierías Físicomecánicas. Escuela de Ingeniería Eléctrica, Electrónica y de Telecomunicaciones. Director Carlos Augusto Fajardo Ariza., Ingeniero Electrónico. Profesor Asociado, Escuela de ingeniería Eléctrica y Electrónica. Codirector Fernando Antonio Rojas Morales., Ingeniero de Sistemas. Profesor Asociado, Escuela de ingeniería de sistemas e informática.

## ABSTRACT

**TITLE:** Design of a functional model of a mobile robot that favors the learning of basic programming concepts in children.\*

**AUTHOR:** James Sebastián Murillo Esteban \*\*

**KEY WORDS:** Programming, tablets, consoles, videogames, mobile robot, neodymium, microcontrollers, bluetooth, voltage divider.

### DESCRIPTION

In this project a functional model of a mobile robot was developed with the purpose of favoring the learning of basic programming concepts in children. It seeks to present an alternative for children to develop skills for programming. The foregoing considering that the easy access of children to devices such as tablets, smartphones, computers and video game consoles, and their difficult supervision, is causing negative consequences such as attention deficit disorder, sleep disturbances, overexposure, others, so that this proposal breaks with the paradigm of learning the computational logic through a computer and puts it in the hands of a physical device offline.

The mobile robot is programmed through a series of chips that tell the robot how to move. Each card represents a specific instruction and the way in which they are arranged in the panel, will affect the trajectory that the robot will perform. Neodymium magnets were used for the coupling between the tiles and the board, so that the pieces not only have a consistent union, but are easily replaceable and regroupable. The board and the robot have two microcontrollers communicated via Bluetooth, where the first, according to the reading of each card, will encode and send the data to the microcontroller of the robot, and the robot will execute the corresponding actions. Each tile has a particular tension divider that differentiates it from other tiles and in this way, a complete and varied set of pieces is obtained. In this way, a different learning alternative is offered to children.

---

\*Degree work

\*\* Faculty of Physicomechanical Engineering. School of Electrical, Electronics and Telecommunications Engineering. Director Carlos Augusto Fajardo Ariza., Electronic Engineer. Associate Professor, School of Electrical and Electronics Engineering. Co-director Fernando Antonio Rojas Morales., Systems Engineer. Associate Professor, School of Systems Engineering and Information Technology.

## INTRODUCCIÓN

En la última década, el mundo ha cambiado constantemente gracias al avance de la tecnología; donde se han desarrollado diferentes procesos, productos, métodos y servicios los cuales generan un impacto ya sea positivo o negativo en las personas [1]. La intervención de la tecnología en la educación en Colombia ha sido influyente para la estructuración de nuevos proyectos orientados al mejoramiento de la calidad de vida, generando posibilidades de crear personas incluyentes donde participen como emprendedores teniendo como base fundamental la innovación[2]. En Bucaramanga se han realizado diferentes eventos donde la población infantil ha vivido un proceso que busca facilitar el aprendizaje de diferentes tecnologías [3].

De otro lado, se ha demostrado la importancia del juego en los procesos de aprendizaje, ya que el juego es una forma de enfrentarse al mundo constantemente, probando y arriesgándose a conocer cuáles son los límites; esa es una forma lúdica de relacionarse con el mundo [4]. De otro lado, se ha observado que los niños que usan un lenguaje de programación a edades tempranas aprenden algunas ideas de ciencia y matemáticas, pero aún más importante es una nueva mentalidad, que hace que ellos tengan la capacidad de resolver problemas de una manera integral y eficaz fortaleciendo la idea de que ellos pueden marcar la diferencia en sus vidas y en las personas que los rodean [5].

En este proyecto se propone un dispositivo que facilite el aprendizaje de habilidades de programación a niños entre 7 y 12 años. Se pretende que sea una alternativa diferente a la que proponen los dispositivos móviles.

Se logró implementar un robot móvil, el cual puede ser programado mediante una estructura de bloques que el niño configura de acuerdo a el nivel planteado.

Este documento está organizado de la siguiente manera: el capítulo 2 describe los diferentes juegos disponibles en el mercado para que los niños aprendan a programar. El capítulo 3 describe el desarrollo a nivel de sistema del robot móvil y del tablero de comandos. El capítulo 4 el diseño del hardware y resultados. Las conclusiones cierran este informe final.

# 1. GENERALIDADES

## 1.1. PLANTEAMIENTO DEL PROBLEMA

Internet es una puerta abierta a un sinnúmero de información de todo tipo, contando con contenidos de noticias, deportes, videojuegos, educación, entretenimiento entre otros. Pero surge un inconveniente con el uso excesivo de estas tecnologías ya que éstas están aislando a los niños y los está “desconectando” de la realidad [6]. Algunos padres ven una ventaja en tecnologías off-line, ya que los niños pueden adquirir habilidades sin tener que estar conectados a la web.

La implementación de un sistema de programación físico hace que se tenga un nuevo concepto de los contenidos y de cómo se interactúa con ellos, haciendo al usuario parte del mundo exterior, creando una conexión entre él y su entorno. Esto conlleva a salir un poco de lo moderno y mirar hacia lo tradicional dejando de un lado dispositivos electrónicos que han generado la aparición de síndromes como el autismo tecnológico [6], en el cual se empieza a perder el sentido del entorno, es decir, que se pierde la comunicación verbal y gestual, que son remplazadas con horas interminables de navegación, muchas veces olvidando los alimentos, las necesidades y las obligaciones, que es el caso de personas, que se ausentan de la realidad, para vivir una realidad paralela. En este sentido, la programación implementada en un elemento físico ofrece a los niños la oportunidad de adaptar el medio e incluso de ser el propio creador, esto les ofrece poder para aprender y obtener beneficios a nivel cognitivo desarrollando su creatividad, pensamiento abstracto, fomentando autonomía y capacidad de análisis y decisión. Por esto es importante inculcar la programación en niños desde temprana edad, ya que los conceptos son adquiridos con mayor facilidad, rapidez y pueden generar un modelo mental autónomo.

## 1.2. JUSTIFICACIÓN

Codificación significa literalmente la transformación de datos en una forma que sea inteligible para un ordenador. En otras palabras, se le dice a un computador qué hacer. Codificar es interpretar una serie de instrucciones en un determinado lenguaje. Ese proceso se asemeja a seguir un número de pasos en un manual de LEGO o generar una rutina de quehaceres diarios. Expertos como Bill Gates y Mark Zuckerberg recomiendan que la introducción temprana a los conceptos básicos de programación puede ayudar a los niños a crear destrezas de resolución de problemas, pensamiento crítico entre otros [7]. Con base en estos argumentos, surge la oportunidad de crear un sistema, interactivo, amigable y divertido donde los niños a temprana edad tengan un acercamiento a la programación y se convierta en una herramienta de pensamiento de fácil acceso.

## 2. ANTECEDENTES

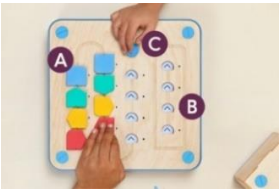




Uno de los primeros juegos diseñados para el aprendizaje de habilidades de programación fue el lenguaje LOGO, el cual nació en los años 70, creado por Seymour Papert (conocido como el lenguaje de la tortuga) [8]; una potente herramienta para la enseñanza de la programación de computadores, que ayuda al pensamiento lógico-matemático. Así, Logo funciona como un instrumento didáctico que permite a los alumnos, sobre todo a los más pequeños, construir sus conocimientos.






Este lenguaje no está limitado a un tópico en particular o a una materia específica. Normalmente se puede realizar programas con el lenguaje LOGO sobre: matemáticas, lenguaje, música, robótica, telecomunicaciones y ciencias. También se usa para simulaciones, presentaciones multimedia, etc. Sin embargo, es más comúnmente utilizado para la exploración de las Matemáticas ya que los gráficos de tortuga de LOGO proveen un ambiente naturalmente matemático [9]. Debido a que la tortuga se mueve una determinada distancia y gira un número dado de grados, el estudio de geometría mediante la construcción e investigación de polígonos y figuras hace de LOGO una herramienta de aprendizaje poderosa. Fundamentalmente consiste en presentar a los niños retos intelectuales que puedan ser resueltos mediante el desarrollo de programas en LOGO. El proceso de revisión manual de los errores contribuye a que el niño desarrolle habilidades metacognitivas al poner en práctica procesos de autocorrección. Otra gran cualidad del LOGO es que a este lenguaje se le pueden enseñar nuevos comandos y a su vez crear otros a partir de estos nuevos. Por esta razón se puede decir que LOGO tiene la cualidad de ser extensible. Es un lenguaje poderoso en el sentido que es capaz de crear programas desde los más simples a programas complejos.

## 2.1 ESTADO DEL ARTE


Se realizó la búsqueda de los juegos y programas de programación para niños existentes, los cuales fueron analizados y tomados como referencia para los requerimientos presentes en el diseño.

**Tabla 1. Revisión del estado del arte juegos de programación**






NOMBRE	DESCRIPCIÓN	IMAGEN
<p><b>Cubetto</b></p> <p>Fuente: Cubetto. Disponible en: <a href="https://blogthinkbig.com/juguetes-para-aprender-a-programar-desde-nino-en-casa-o-en-la-escuela">https://blogthinkbig.com/juguetes-para-aprender-a-programar-desde-nino-en-casa-o-en-la-escuela</a></p>	<p>Consta de una consola de programación física (tablero), que cuenta con 16 comandos de colores para programar las funciones como adelante, izquierda y derecha. El tablero para programar las acciones, siguiendo un camino, presionando un botón de inicio y representando su movimiento mediante un robot.</p>	
<p><b>Code and go</b></p> <p>Fuente: Code and go. Disponible en: <a href="https://www.robotsparaninos.com/code-go-robot-mouse/">https://www.robotsparaninos.com/code-go-robot-mouse/</a></p>	<p>Colby es una herramienta para la introducción a la programación. Lleva un teclado de comandos sobre el mismo, programando cada instrucción sucesivamente, para luego representar los movimientos previamente “codificados”</p>	
<p><b>Legó mindstorms</b></p> <p>Fuente: Legó mindstorms. Disponible en: <a href="https://en.wikipedia.org/wiki/Lego_Mindstorms">https://en.wikipedia.org/wiki/Lego_Mindstorms</a></p>	<p>Es usado para construir un modelo de sistema integrado con partes electromecánicas controladas por computador. Todo puede ser representado con las piezas tal como un elevador o robots industriales</p>	
<p><b>Legó wedo 2.0</b></p> <p>Fuente: Legó wedo. Disponible en: <a href="https://www.robotsparaninos.com/code-go-robot-mouse/">https://www.robotsparaninos.com/code-go-robot-mouse/</a></p>	<p>Es usado para montar diferentes objetos con las piezas disponibles y darles vida con un pequeño motor y la app (Android/iOS) para programar acciones.</p>	
<p><b>Kibo</b></p> <p>Fuente: Kibo. Disponible en: <a href="https://blogthinkbig.com/juguetes-para-aprender-a-programar-desde-nino-en-casa-o-en-la-escuela">https://blogthinkbig.com/juguetes-para-aprender-a-programar-desde-nino-en-casa-o-en-la-escuela</a></p>	<p>Consiste en controlar un robot combinando cubos y fichas de colores, cada uno con un movimiento o función. Se ordenan los cubos siguiendo una secuencia de instrucciones</p>	


<p><b>Osmo coding</b></p> <p>Fuente: Osmo coding. Disponible en: <a href="https://blogthinkbig.com/juguetes-para-aprender-a-programar-desde-nino-en-casa-o-en-la-escuela">https://blogthinkbig.com/juguetes-para-aprender-a-programar-desde-nino-en-casa-o-en-la-escuela</a></p>	<p>Sirve para aprender nociones básicas de programación resolviendo problemas a partir de acciones predefinidas diseñando secuencias de instrucciones para ayudar a (Awbie) el personaje protagonista del juego.</p>	
<p><b>Dash and dot</b></p> <p>Fuente: Dash and dot. Disponible en: <a href="https://blogthinkbig.com/juguetes-para-aprender-a-programar-desde-nino-en-casa-o-en-la-escuela">https://blogthinkbig.com/juguetes-para-aprender-a-programar-desde-nino-en-casa-o-en-la-escuela</a></p>	<p>A partir de una app gratuita, se programan sus acciones y movimientos con un lenguaje de programación muy gráfico.</p>	
<p><b>Zowi</b></p> <p>Fuente: Zowit. Disponible en: <a href="https://blogthinkbig.com/juguetes-para-aprender-a-programar-desde-nino-en-casa-o-en-la-escuela">https://blogthinkbig.com/juguetes-para-aprender-a-programar-desde-nino-en-casa-o-en-la-escuela</a></p>	<p>Un robot que camina, baila y canta a partir de órdenes a través de su app para Android o vía web desde cualquier ordenador, Zowi es configurable con un lenguaje de programación modular muy similar a Scratch.</p>	
<p><b>Think &amp; Learn Code-a-Pillar</b></p> <p>Fuente: Think &amp; Learn code a pillar. Disponible en: <a href="https://www.whatsnew.com/2016/02/17/5-interesantes-juguetes-ensenar-ninos-programar/">https://www.whatsnew.com/2016/02/17/5-interesantes-juguetes-ensenar-ninos-programar/</a></p>	<p>Es una oruga de colores cuyos módulos independientes e intercambiables pueden programarse para girar, sonar y desplazarse en varios sentidos.</p>	
<p><b>Codie</b></p> <p>Fuente: Codie. Disponible en: <a href="https://www.whatsnew.com/2016/02/17/5-interesantes-juguetes-ensenar-ninos-programar/">https://www.whatsnew.com/2016/02/17/5-interesantes-juguetes-ensenar-ninos-programar/</a></p>	<p>Se trata de un tanque de madera donde se desarrolla el pensamiento lógico y resolución de problemas de forma estructurada. Se controla desde una aplicación para móviles y el lenguaje de programación se realiza a partir de coloridos e intuitivos bloques</p>	

**Tabla 2. Revisión del estado del arte software de programación**

NOMBRE	DESCRIPCIÓN	IMAGEN
<p><b>Kodable</b></p> <p>Fuente: kodable Disponible en: <a href="http://www.educaciontrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html">http://www.educaciontrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html</a></p>	<p>Es una plataforma que se presenta para enseñar programación a partir de movimientos unidireccionales. Está disponible en múltiples plataformas (Windows, Mac, iPad, Android).</p>	
<p><b>SpaceChem</b></p>	<p>Se deben realizar programación para resolver puzzles y problemas lógicos de</p>	

<p>Fuente: SpaceChen Disponible en:  <a href="http://www.educacionrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html">http://www.educacionrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html</a></p>	<p>todo tipo creando máquinas, organizando o mezclando de un modo determinado para poder dar con la solución final en cada fase. descargarse para plataformas Windows, Mac y Linux.</p>	
<p><b>Code Monkey</b></p> <p>Fuente: Code Monkey Disponible en:  <a href="http://www.educacionrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html">http://www.educacionrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html</a></p>	<p>Es un juego donde se resuelven diferentes problemas a través de la creación de pequeños programas con código y de una interfaz intuitiva. La dificultad va aumentando progresivamente desde un nivel de iniciación hasta los modos más avanzados</p>	
<p><b>Code Combat</b></p> <p>Fuente: Code Combat Disponible en:  <a href="http://www.educacionrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html">http://www.educacionrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html</a></p>	<p>Sirve para aprender a programar por medio de mandos de un personaje medieval que debería ir avanzando por los diferentes niveles del juego, solucionando los problemas planteados. Con él se aprenden lenguajes avanzados como Python, Lua, Javascript entre otros.</p>	
<p><b>Lightbot</b></p> <p>Fuente: Lightbot Disponible en:  <a href="http://www.educacionrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html">http://www.educacionrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html</a></p>	<p>Sirve para programar avanzando los diferentes niveles, en los que deberás ordenar a los robots ciertos movimientos y que cumplan algunas órdenes.</p>	
<p><b>Robocode</b></p> <p>Fuente: Robocode Disponible en:  <a href="http://www.educacionrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html">http://www.educacionrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html</a></p>	<p>Sirve para programar con personajes de 'guerra' donde se controlan robots por medio de movimientos y órdenes para derrotar al enemigo.</p>	
<p><b>Machineers</b></p> <p>Fuente: Machineers Disponible en:  <a href="http://www.educacionrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html">http://www.educacionrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html</a></p>	<p>Se construyen robots personalizados y con las piezas, controles y funciones variadas. Cada uno de ellos programable, permitiendo poner en práctica los conocimientos de programación o aprender desde cero.</p>	
<p><b>Daisy the Dinosaur</b></p> <p>Fuente: Daisy the dinosaur Disponible en:  <a href="http://www.educacionrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html">http://www.educacionrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html</a></p>	<p>Sirve para aprender fundamentos de programación a través de retos y dando movimiento a Daisy un dinosaurio animado.</p>	

<p><b>Codea</b></p> <p>Fuente: Codea Disponible en: <a href="http://www.educaciontrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html">http://www.educaciontrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html</a></p>	<p>Es un entorno de programación, que se utiliza para aprender poco a poco un lenguaje de programación y sus posibilidades a partir de un robot moviendo diferentes bloques.</p>	
<p><b>Tynker</b></p> <p>Fuente: Tynker Disponible en: <a href="http://www.educaciontrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html">http://www.educaciontrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html</a></p>	<p>Proporciona un entorno de programación por bloques que también permite programar Minecraft, robots como LEGO WeDo o incluso apps para móvil</p>	
<p><b>Codeable Crafts</b></p> <p>Fuente: Codeable Crafts Disponible en: <a href="http://www.educaciontrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html">http://www.educaciontrespuntocero.com/recursos/apps-de-juegos-para-aprender-a-programar/30909.html</a></p>	<p>Mezcla la programación junto con el arte, el dibujo y la creatividad, permitiendo elaborar y diseñar historias a través de la programación de bloques.</p>	
<p><b>The Foos: Code for an Hour FREE</b></p> <p>Fuente: The foos: Code for an Hour free Disponible en: <a href="https://www.whatsnew.com/2016/02/17/5-interesantes-juguetes-ensinar-ninos-programar/">https://www.whatsnew.com/2016/02/17/5-interesantes-juguetes-ensinar-ninos-programar/</a></p>	<p>Juego para aprender a programar a través de una historia que incluye niveles cada vez más difíciles.</p>	
<p><b>Kodable Pro</b></p> <p>Fuente: Kodable Pro: Code for an Hour free Disponible en: <a href="https://www.whatsnew.com/2016/02/17/5-interesantes-juguetes-ensinar-ninos-programar/">https://www.whatsnew.com/2016/02/17/5-interesantes-juguetes-ensinar-ninos-programar/</a></p>	<p>Introduce a los niños a los conceptos de programación o <i>coding</i>.</p>	
<p><b>ScratchJr.</b></p> <p>Fuente: ScratchJr. Disponible en: <a href="https://www.whatsnew.com/2016/02/17/5-interesantes-juguetes-ensinar-ninos-programar/">https://www.whatsnew.com/2016/02/17/5-interesantes-juguetes-ensinar-ninos-programar/</a></p>	<p>Consiste en programación con bloques donde se pueden arrastrar conectar y hacer play para ver el movimiento del personaje.</p>	
<p><b>Hopscotch</b></p> <p>Fuente: Hopscotch. Disponible en: <a href="https://www.whatsnew.com/2016/02/17/5-interesantes-juguetes-ensinar-ninos-programar/">https://www.whatsnew.com/2016/02/17/5-interesantes-juguetes-ensinar-ninos-programar/</a></p>	<p>Consiste en conceptos básicos de programación, donde dan movimiento a objetos siguiendo una serie de instrucciones.</p>	
<p><b>Mozilla Thimble</b></p> <p>Fuente: Mozilla Thimble. Disponible en: <a href="https://www.whatsnew.com/2016/02/17/5-interesantes-juguetes-ensinar-ninos-programar/">https://www.whatsnew.com/2016/02/17/5-interesantes-juguetes-ensinar-ninos-programar/</a></p>	<p>De nivel avanzado, los niños pueden programar y ver sus creaciones cobrar vida.</p>	

<p><b>CodeHS</b></p> <p>Fuente: CodeHS. Disponible en: <a href="https://www.whatsnew.com/2016/02/17/5-interesantes-juguetes-ensenar-ninos-programar/">https://www.whatsnew.com/2016/02/17/5-interesantes-juguetes-ensenar-ninos-programar/</a></p>	<p>Es un servicio de suscripción con instrucciones y ejemplos para programadores más avanzados.</p>	
--	---	---

### 3. DESARROLLO A NIVEL DE SISTEMA

#### 3.1 DESCRIPCIÓN JUEGO LIGHBOT

Lighbot es un juego de lógica y programación, donde un robot debe cumplir el objetivo de activar las baldosas azules de cada nivel. El robot generara los movimientos programado por medio de un set de instrucciones, tales como:

Adelante.

Giro derecha.

Giro izquierdo.

Salto.

Checking luz(baldosa).

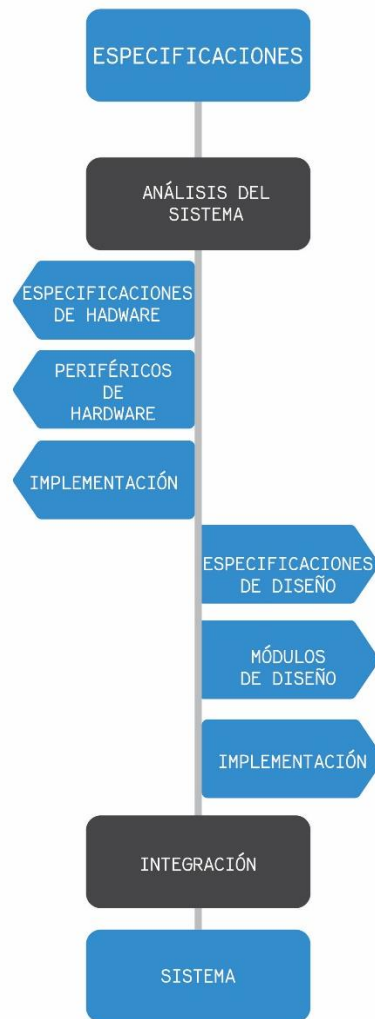
Proc1.

Proc2.

Cada instrucción será “programada” secuencialmente, ejecutando el movimiento del robot una después de otra, hasta cumplir con cabalidad el objetivo del nivel. En caso de tener un error en el programa, el usuario tiene la oportunidad de corregir su error y repetir de nuevo la acción hasta dar con la solución.

En la figura 1, se permite ver el proceso de desarrollo del proyecto, iniciando desde las especificaciones generales del proyecto hasta el sistema desarrollado.

**Figura 1. Diagrama desarrollo de proyecto**



### **3.2 ESPECIFICACIONES GENERALES**

Como objetivo general, se debe realizar un montaje físico del juego Lighbot, para cumplir las especificaciones, se debe combinar el trabajo desarrollado de hardware y diseño. El sistema físico para implementar debe contar con los siguientes requerimientos:

Fichas: representa el comando a ejecutar

Tablero comandos: interpreta el comando de las fichas.

Robot móvil: ejecuta los comandos interpretados por el tablero.

Tapete: superficie sobre la cual el robot ejecutara sus movimientos.

En la figura 2, se representa el diagrama general, entre el tablero de comandos y el robot con sus respectivos módulos.

**Figura 2. Diagrama Especificaciones Generales**



### 3.3 ESPECIFICACIONES DE HARDWARE

Para la implementación de hardware se deben plantear los requerimientos necesarios:

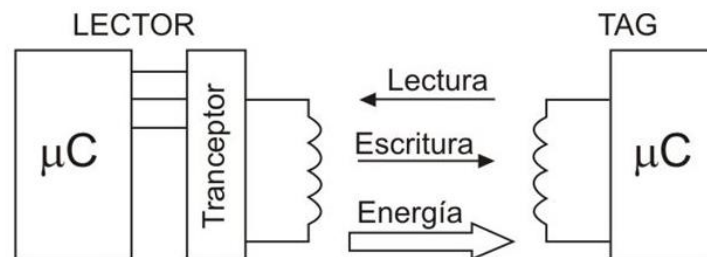
- El tablero debe identificar cada una de las distintas instrucciones de las fichas.
- El tablero y el robot se deberán comunicar inalámbricamente.
- El robot móvil deberá ejecutar sus acciones sobre una matriz de cuadrados sin desviarse o salirse de ellos. El tablero de comandos y el robot deben implementar una unidad de procesamiento.

**3.3.1 Identificación instrucción fichas.** Se tiene como reto leer cada instrucción o comando de cada ficha, teniendo una variedad mínima de 3 comandos (adelante, giro izquierda y giro derecha), por lo cual se hace necesario implementar un hardware de sensado que permita la identificación. Se evaluaron 3 alternativas presentadas a continuación:

**3.3.1.1 Modulo RFID.** El principio fundamental de un sistema RFID consiste en un transponder y un lector de RFID. El lector interroga al transponder utilizando cierta frecuencia y el transponder contesta a distancia con la información que contiene, que puede ser un numero identificador de producto. El lector recoge esta información y la envía a una unidad de cómputo para su procesamiento. El funcionamiento general se puede apreciar en la siguiente figura.

En la figura 3, se representa la transmisión y recepción de datos entre el módulo lector RFID y el TAG.

**Figura 3. Representación de un sistema RFID**



Los TAG, se pueden encontrar en gran variedad de tamaños, para aplicar en este proyecto se encuentra en el mercado del tamaño de una moneda, ideal para la aplicación que queremos implementar.

La figura 4, representa un tag circular de 1.6 cm de diámetro. Demostrando que tan pequeño pueden resultar este módulo de transmisión.

**Figura 4. Tag 16mm (125Khz)**



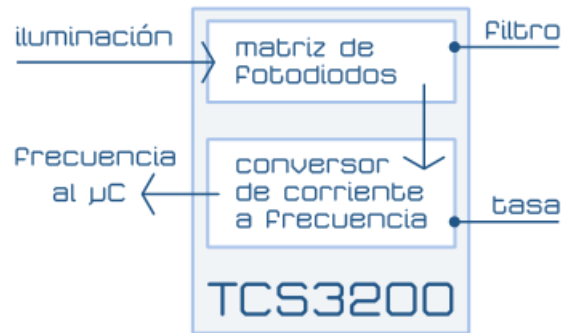
Fuente: <https://www.sparkfun.com/products/9417>

**3.3.1.2 Sensor color.** La superficie de la ficha puede ser identificada por un determinado color. Esto se puede realizar mediante el integrado TCS3200.

Este sensor convierte en frecuencia la intensidad de luz medida por una matriz de fotodiodos. La frecuencia entregada por el sensor TC3200 es mayor cuanto mayor luminosidad se detecte. La matriz de fotodiodos permite promedia el valor medido para compensar diferencias de color en la superficie muestreada, así como disponer de filtros de color que, al ir alternando su estado, sirven para distinguir cada superficie.

En la figura 5, se representa el funcionamiento entre los módulos internos del sensor TCS3200.

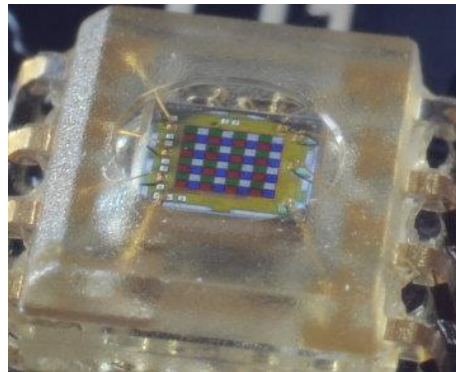
**Figura 5. Funcionamiento TCS3200**



Fuente: <https://polaridad.es/sensor-color-tcs3200-frecuencia-arduino/>

La figura 6, se puede apreciar la matriz de fotodiodos presentes en el sensor.

**Figura 6. Sensor TCS3200**

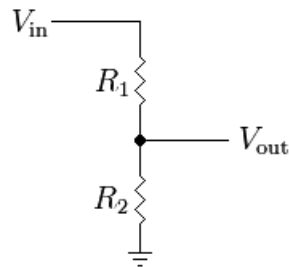


Fuente: <https://polaridad.es/sensor-color-tcs3200-frecuencia-arduino/>

**3.3.1.3 Divisor de tensión.** El divisor de tensión permite tener una respuesta en voltaje dependiendo de los valores de R2 y R1. El propósito es dejar fijo un valor de estos y variar el otro. Con esta configuración podremos identificar la ficha.

La figura 7, representa el esquemático de un divisor de tensión con sus respectivos parámetros.

**Figura 7. Divisor de tensión**



La ecuación 1, resuelve el circuito de la figura 7, teniendo como respuesta  $V_{out}$ .

Ecuación 1: Divisor de tensión

$$V_{out} = \frac{R_2 * V_{in}}{R_2 + R_1}$$

Nota: Donde  $V_{in}$  es un voltaje fijo en su defecto de 5V. Sea  $R_1$  o  $R_2$  la que identifica la ficha debe ir acoplada a ella

**3.3.1.4 Análisis.** Para la selección de la mejor alternativa, se tuvo en cuenta el costo, ya que, si se tiene pensado reproducir en cantidad, se debe reducir en lo más mínimo posible.

**RFID:** Es la opción menos viable, ya que para leer cada ficha se requiere un lector, y como mínimo se necesitan leer 20 instrucciones de 20 fichas, estableciendo 20 lectores. El precio de cada lector está por encima de los 10 USD. Un valor no muy favorable para su ejecución como prototipo.

**SENSOR COLOR:** de igual forma que en el RFID, se necesitarían como mínimo 20 sensores de color para leer cada ficha, pero teniendo un precio por sensor de 2 USD.

**DIVISOR DE TENSION:** Es la opción más factible, ya que el precio de cada resistencia puede costar céntimos de dólar. El único problema es la integración de la resistencia y la ficha, tema tratado en los requerimientos de diseño.

**3.3.1.5 Resultados.** Optando por el divisor de tensión, se prosigue a seleccionar los valores de resistencia, para un numero de 6 instrucciones. Se optó por dejar fija R2 con un valor de 10 [k $\Omega$ ]. Este valor de resistencia va en el tablero de comandos.

Los 6 valores de R1 se escogieron con una separación de 0.8V, para optar en un buen margen a la hora de hacer la conversión análoga-digital (ADC).

$$R1= 220 [\Omega] \quad V_{out}= 4.89 [V]$$

$$R2= 2.2 [k\Omega] \quad V_{out}= 4.1 [V]$$

$$R3= 4.7 [k\Omega] \quad V_{out}= 3.4 [V]$$

$$R4= 10 [k\Omega] \quad V_{out}= 2.5 [V]$$

$$R5= 22 [k\Omega] \quad V_{out}= 1.56 [V]$$

$$R6= 47 [k\Omega] \quad V_{out}=0.89 [V]$$

$$R= \infty \quad V_{out}= 5 [V]$$

Dejando fijo  $V_{in} =5 [V]$ .

Teniendo en cuenta que el sensado de cada ficha se va hacer obteniendo una respuesta en voltaje, el ADC se debe encargar de hacer su debida conversión. Pero para poder aplicar esta solución es necesario agregar un hardware adicional para el funcionamiento.

El tablero tiene como mínimo 20 casillas. En estas casillas se “acoplaran” las fichas, por lo cual, se quiere tener el sensado cuando las 20 casillas estén ocupadas por fichas, se necesitan 20 ADC, presentando un inconveniente a la hora de usar un MCU o un procesador. Por lo cual se debe usar un multiplexor analógico. Para este caso se decidió usar el CD4067BE de 16 canales de la empresa Texas Instrument.

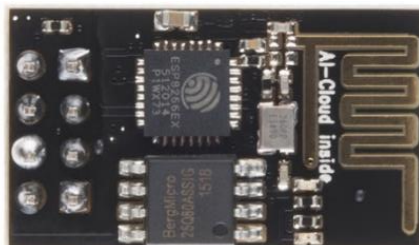
### 3.4 COMUNICACIÓN INALAMBRICA

Se debe optar por un sistema de comunicación inalámbrica, donde el tablero de comandos le enviara la información para que posteriormente sean ejecutadas por el robot.

**3.4.1 Wifi.** Wi-Fi es un sistema de envío de datos sobre redes computacionales que utiliza ondas de radio en lugar de cables. El módulo ESP8266, permite manejar protocolos TCP/IP, el cual puede dar acceso a cualquier MCU a su red Wi-Fi.

En la figura 8, se puede apreciar el módulo ESP8266, siendo el más usado en proyectos de IoT.

**Figura 8. Módulo ESP8266**



Fuente: <https://www.sparkfun.com/products/13678>

**3.4.2 Bluetooth.** Bluetooth es una especificación industrial para Redes Inalámbricas de Área Personal (WPAN) que posibilita la transmisión de voz y datos entre diferentes dispositivos mediante un enlace por radiofrecuencia en la banda ISM de los 2.4 GHz. Ideal para la comunicación entre periféricos.

La figura 9, representa el módulo bluetooth hc-05. Configurable mediante protocolo AT.

**Figura 9. Módulo bluetooth HC-05**



Fuente: <http://www.electronicaestudio.com/docs/istd016A.pdf>

**3.4.3 Modulo XBEE.** El protocolo Zig-bee usa la banda de los 2.4 GHz, siendo esta banda definida por el ministerio de tecnologías de la información y las comunicaciones de Colombia como banda libre.

El alcance depende de la potencia de transmisión del dispositivo, así como también del tipo de antenas utilizadas. Un ejemplo es que a 1mW de potencia tiene una cobertura de 100 metros libres y en interiores unos 30m. La velocidad de transmisión de datos de una red Zigbee es de hasta 256 Kbps.

La figura 10, se evidencia el módulo Xbee, siendo el más básico de su familia.

**Figura 10. Módulo Xbee 1 mW**



Fuente <https://www.sparkfun.com/products/8665>

**3.4.4 Análisis.** Para la selección del modo inalámbrico, se tuvo en cuenta su costo, y el propósito específico para el cual se va a usar. Se necesita que el módulo sea de fácil implementación en una unidad de procesamiento, además no se requiere una cobertura mayor a 3m y la transmisión de datos no exigen una alta velocidad de transferencia.

**3.4.5 Resultados.** Se decidió usar el módulo bluetooth, inicialmente por su bajo costo, además es ideal para conectar periféricos entre sí, objetivo necesario, ya que deseamos interconectar el tablero de mandos con el robot.

### **3.5 MOVIMIENTO GUIADO DEL ROBOT**

El robot ejecutara sus movimientos sobre una matriz de cuadrados, siendo el tamaño de cada cuadro un 10% mayor a la dimensión real del robot. El robot deberá estar en cada momento sobre los cuadrados, sin salirse moderadamente de ellos, por lo cual se hace necesario usar hardware, para hacer el control de giro en cada rueda, por lo cual se plantean las siguientes soluciones:

**3.5.1 Encoder infrarrojo.** El encoder consta de dos sensores infrarrojos que están separados a una diferencia de fase de 90 grados, permitiendo saber cuándo el robot gira su rueda hacia adelante o hacia atrás. Cabe resaltar que para usar este sensor es necesario emplear ruedas especiales, que cuentan con unos dientes el cual es detectado por los sensores.

En la figura 11, se dispone de los sensores infrarrojos que vienen soldados en una PCB, y de las ruedas especiales para este tipo de encoder.

**Figura 11. Encoder infrarrojo MiniQ robot**



Fuente: <https://www.dfrobot.com/product-823.html>



Fuente <https://www.dfrobot.com/product-353.html>

**3.5.2 Encoder magnético.** Este encoder, tiene el mismo principio de funcionamiento que el encoder por infrarrojo, contando con dos sensores de efecto

hall separados a una fase de 90 grados. Además, cuenta con un disco magnético de 6 polos que se acopla al eje del motor.

En la figura 12, se aprecia los micro motores, que son de uso especial para este tipo de encoder, ya que cuentan con un eje largo el cual permite acoplar la PCB y el disco magnético.

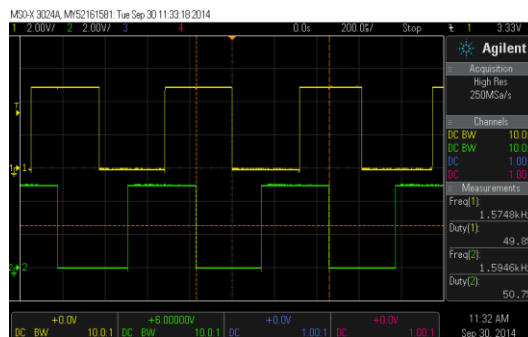
**Figura 12. Encoder efecto hall con acople al eje**



Fuente: <https://www.pololu.com/product/2598>

La figura 13, representa la señal de desfase de 90° entre las dos salidas del encoder.

**Figura 13. Señal de salida**



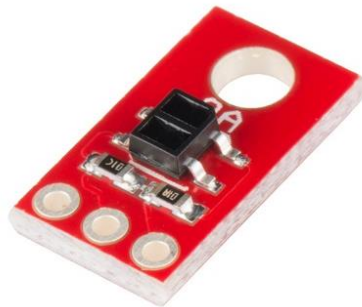
Fuente: <https://www.pololu.com/product/2598>

**3.5.3 Seguimiento de línea.** En esta solución, se requiere agregar una línea negra de ancho no mayor a 2cm, donde el carro podrá seguir esta línea sin dificultad de salirse de los cuadros.

Usando sensores infrarrojos como el QRE1113, que consta de un led emisor IR y un fototransistor sensible IR. Dependiendo de la intensidad de luz reflejada por la línea el fototransistor representara su salida en voltaje

En la figura 14, se observa el sensor QRE1113 montado sobre la PCB, listo para obtener su salida representada en voltaje.

**Figura 14. Señal de salida**



Fuente: <https://www.sparkfun.com/products/9453>

**3.5.4 Análisis y resultados.** Para escoger la mejor alternativa, se debe tener en cuenta la opción que asegure un control de movimiento sobre los recuadros. Se pensó inicialmente en usar sensores infrarrojos para seguir una línea negra, pero al tener estas marcas sobre el tapete donde el robot se va a desplazar, genera confusión y los niveles se hacen menos entendibles para el usuario, por lo cual se optó por la solución del encoder magnético. Se selecciono este sensor por encima del encoder infrarrojo, ya que la PCB se acopla sobre su propio eje, evitando futuros inconvenientes a la hora de diseñar la PCB del robot. Además, la resolución de los encoder magnéticos es mayor a la de los infrarrojos, teniendo un control más exacto en el giro del robot.

### 3.6 SELECCIÓN SoC

La SoC seleccionada, tendrá como características módulos PWM, ADC, GPIO y comunicación serial. Programación en alto nivel como C/C++ o Python. De bajo consumo y fácil implementación en hardware.

**3.6.1 Sistema embebido.** Se selecciona una raspberry pi 3 modelo B, que cuenta con un procesador ARM Quad Core 1.2Ghz Broadcom BCM2837 de 64 bit, 1GB RAM, con módulos integrados de Wifi y bluetooth. 40 pines GPIO, con los cuales se pueden configurar para señales PWM, y comunicación serial. Puertos USB y HDMI. Sistema operativo raspbian, siendo una extensión más de Linux.

La figura 15, se observa la última versión de raspberry pi, siendo esta versión con las mejores características del mercado.

**Figura 15. Raspberry pi 3 model B**



Fuente: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>

**3.6.2 Microcontrolador.** Se selecciona el microcontrolador atmega328p de la empresa ATMEL CORP, actualmente comprada por MICROCHIP. Es un MCU de 8 bits de arquitectura AVR RISC. Cuenta con pines de propósito general ideal para

proyectos de robótica, domótica, automatización, etc. Compatible con el entorno IDE de Arduino. Voltaje de operación 1.8V a 5.5V.

La figura 16, se observa el microcontrolador atmega328p de 8 bits empaquetado SMD de 32 pines.

**Figura 16. Atmega328p**



Fuente: <http://www.microchip.com/wwwproducts/en/ATmega328P>

**3.6.3 Análisis y resultados.** Para la selección de la unidad de procesamiento del tablero de comandos y el robot, es importante tener en cuenta, cual es la alternativa que ofrece una mejor solución sin la necesidad de desaprovechar los recursos de cada SoC presentadas anteriormente. La mejor alternativa para implementar es el MCU atmega328p, ya que este presenta la mayoría de características, que se necesitan para completar el desarrollo del proyecto. Además, de su precio, es muy económico comparada con la raspberry pi 3.

La raspberry pi 3 sería una buena alternativa al contar con los módulos de Wifi y bluetooth integrados, pero toda la capacidad de computo se desperdiciaría, ya que el proyecto no está enfocado a realizar cálculos a altas velocidades, los 16 MHz del atmega328p son más que suficiente para suplir la necesidad.

El atmega328p se escogió por encima de la raspberry pi, ya que se puede usar el IDE de Arduino, esto ahorra tiempo a la hora de implementar librerías como son las de serial, vitales para el manejo del bluetooth.

Teniendo en cuenta los requerimientos de hardware es hora de evaluar las especificaciones principales de diseño que dependían en primera instancia de algunos requerimientos de esta primera fase y se deben evaluar los distintos conceptos planteados en cuanto a forma, tamaño, usabilidad entre otras.

### **3.7 GENERACIÓN CONCEPTO: FICHAS**

Para el desarrollo de las fichas se debe tener en cuenta que las n superficies que pueda tener la ficha, debe llevar impreso los movimientos unidireccionales mostrados en la figura X.

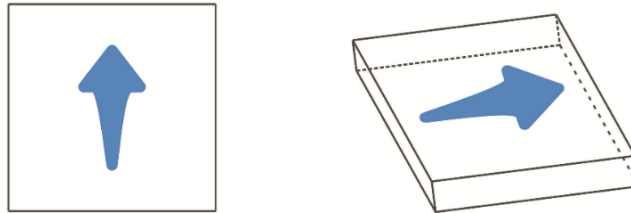
La figura 17, se observa los movimientos unidireccionales que serían impresos sobre las fichas.

**Figura 17. Movimientos unidireccionales**



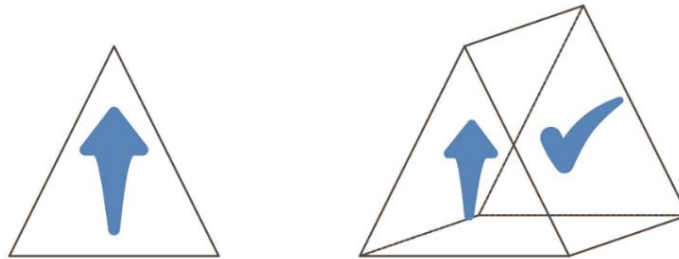
**3.7.1 Concepto 1: Ficha plana.** La figura 18, se observa el diseño de ficha plana en vistas superior y perspectiva caballera, con su respectivo movimiento unidireccional.

**Figura 18. Ficha plana. Vista superior y perspectiva caballera**



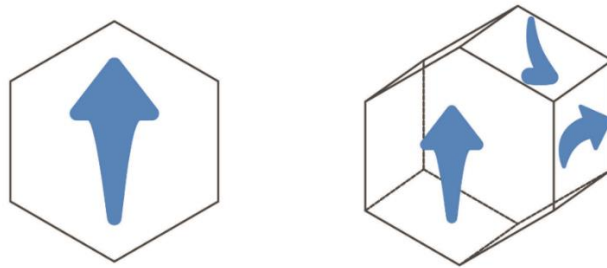
**3.7.2 Concepto 2: Ficha triangular.** La figura 19, se observa el diseño de ficha triangular en vistas superior y perspectiva caballera, con su respectivo movimiento unidireccional.

**Figura 19. Ficha triangular. Vista superior y perspectiva caballera**



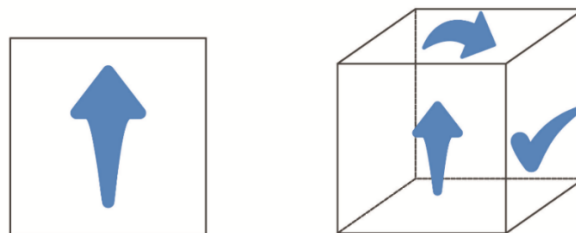
**3.7.3 Concepto 3: Ficha hexagonal.** La figura 20, se observa el diseño de ficha hexagonal en vistas superior y perspectiva caballera, con sus respectivos movimientos unidireccionales.

**Figura 20. Ficha hexagonal. Vista superior y perspectiva caballera**



**3.7.4 Concepto 4: Ficha cubo.** La figura 21, se observa el diseño de ficha cubo en vistas superior y perspectiva caballera, con sus respectivos movimientos unidireccionales.

**Figura 21. Ficha cubo. Vista superior y perspectiva caballera**



**3.7.5 Selección concepto.** Para la selección de la ficha se tiene en cuenta el juego Lightbot, cuenta con un total de 7 instrucciones, pero omitiendo la instrucción de salto serian 6 en total. Se busca una ficha que pueda integrar el mayor número de instrucciones, además de ser de fácil manipulación, por parte del usuario final.

La tabla 3, representa los criterios de selección de los 4 conceptos presentados anteriormente. Al final, el concepto que presente la mejor propuesta a desarrollar

continuará el proceso a la siguiente fase.

**Tabla 3. Criterio de selección, concepto fichas**

CRITERIOS DE SELECCIÓN	CONCEPTO FICHAS			
	1	2	3	4
Fácil de almacenar en conjunto, apilable	+	+	+	+
Fácil mantenimiento: Formas que permitan el fácil acceso, material lavable, fácil secado, cambio de piezas sencillo	-	+	+	+
Permite libre posición y acoplamiento	-	+	+	+
Bordes redondeados y superficies suaves	+	+	+	+
Debe permitir la unificación de todos los movimientos unidireccionales propuestos	-	-	-	+
Suma +	2	4	4	5
Suma -	3	2	1	0
Suma 0	0	0	0	0
Evaluación neta	-1	2	3	5
Posición	4	3	2	1
Continúa?	NO	NO	NO	SI

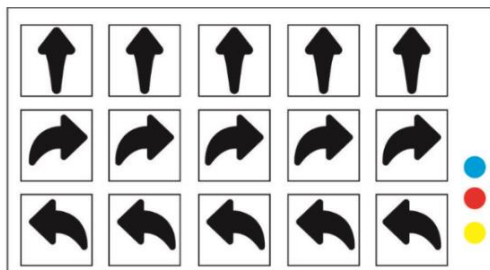
**3.7.6 Resultados.** Se selecciona el concepto de ficha en forma de cubo, debido a que puede integrar el mayor número de instrucciones y sus caras son todas simétricas a diferencia del hexágono que sus caras no presentan la misma medida. Además, permite una óptima manipulación por parte del usuario.

### 3.8 GENERACIÓN CONCEPTO: TABLERO MANDOS

Teniendo en cuenta que el juego Lighbot cuenta con 28 casillas. 12 de ellas son del MAIN, 8 de Proc1 y 8 del Proc2. Para el desarrollo del proyecto se omite el uso del Proc2, reduciendo el número de casillas a un total de 20. Por lo cual se busca generar un concepto que favorezca una buena distribución de las fichas sobre el tablero de mandos. Para el desarrollo de esta etapa se seleccionó la ficha en forma de cubo.

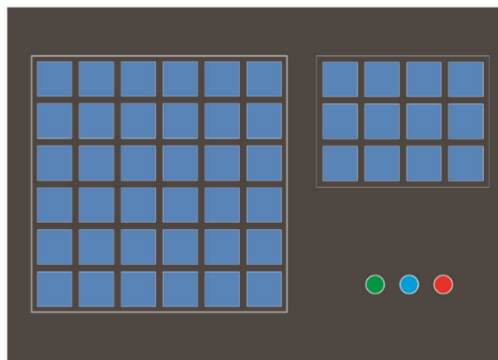
La figura 22, se observa una posible configuración de las fichas sobre el tablero, teniendo en cuenta las fichas en forma de cubo.

**Figura 22. Tablero de mando**



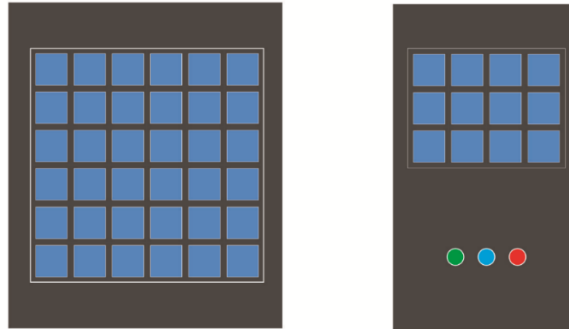
**3.8.1 Concepto 1: Tablero de mando y noveles integrado.** La figura 23, se observa la distribución del tablero de mandos ubicado en la derecha, donde se introducen las respectivas fichas en formas de cubo y a su izquierda la matriz de cuadrados donde se desplazará el robot.

**Figura 23. Tablero de mando y niveles integrados**



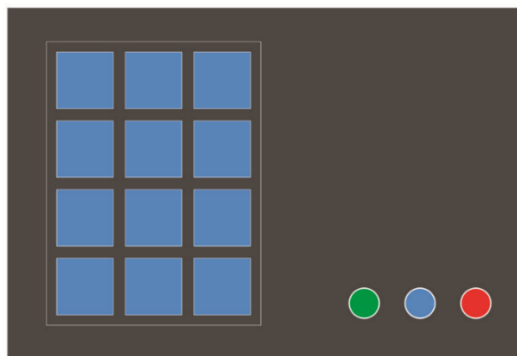
**3.8.2 Concepto 2: Tablero de mando y niveles separados.** La figura 24, se observa al lado derecho el tablero de comandos y a su izquierda de forma independiente la matriz de cuadrados en orientación vertical.

**Figura 24. Tablero de mando y niveles separados**



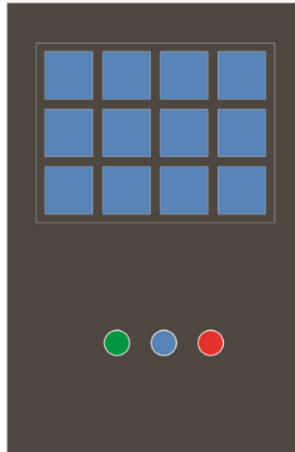
**3.8.3 Concepto 3: Tablero de mando individual, orientación horizontal.** La figura 25, se observa de modo independiente el tablero de comandos en orientación horizontal.

**Figura 25. Tablero de mando individual, orientación horizontal**



**3.8.4 Concepto 4: Tablero de mando individual, orientación vertical.** La figura 26, se observa de modo independiente el tablero de comandos en orientación vertical.

**Figura 26. Tablero de mando individual, orientación vertical**



**3.8.5 Selección concepto.** Como parámetro de selección se busca que el tablero sea lo más pequeños posible, para que el usuario lo pueda manipular en cualquier posición (sentado, acostado, de pie, etc.)

La tabla 4, representa los criterios de selección de los 4 conceptos presentados anteriormente. Al final, el concepto que presente la mejor propuesta a desarrollar continuara el proceso a la siguiente fase.

**Tabla 4. Criterio de selección, concepto tablero**

CRITERIOS DE SELECCIÓN	CONCEPTO TABLERO			
	1	2	3	4
Fácil acoplamiento de las fichas	+	+	+	+
Debe permitir la buena distribución de los componentes	+	+	+	+
Sus dimensiones deben ser menores a 120cm <sup>2</sup>	-	-	+	+
Su peso debe ser menor a 1 kg	-	-	+	+
Debe permitir la unificación de todos los movimientos unidireccionales propuestos	+	+	+	+
Suma +	3	3	5	5
Suma -	2	2	0	0
Suma 0	0	0	0	0
Evaluación neta	1	1	5	5
Posición	4	3	2	1
Continúa?	NO	NO	NO	SI

**3.8.6 Resultados.** Para la selección de la mejor propuesta se requiere el diseño de un tablero ligero y entendible para el usuario. El concepto 1 propone integrar el tablero de comandos y los niveles en un solo dispositivo, presentando un contra debido que, el robot a desarrollar debe ser lo suficientemente pequeño para que el sistema total no resulte muy grande o estorboso. Desarrollar un robot de tales especificaciones requiere un diseño detallado, objetivo no competente en el desarrollo del proyecto.

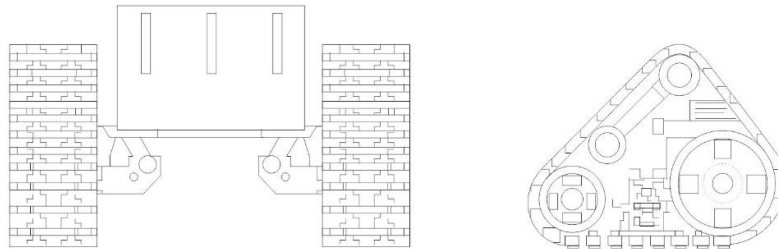
Se opta por implementar un tablero de comandos independiente, tales como los conceptos 3 y 4, eligiendo el concepto 4 de orientación vertical, preferible para una óptima distribución de las cacillas donde se ubicarán cada una de las fichas.

### 3.9 GENERACIÓN CONCEPTO: ROBOT

El diseño del robot tendrá como propósito desplazarse sobre una superficie plana.

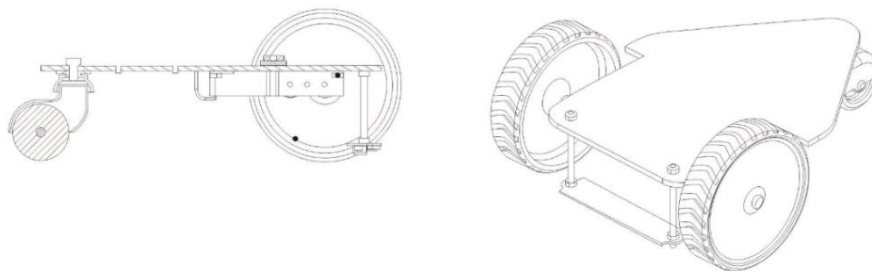
**3.9.1 Concepto 1: Robot con orugas.** La figura 27, Se observa un diseño preliminar de un robot con tracción por orugas.

**Figura 27. Tracción por orugas**



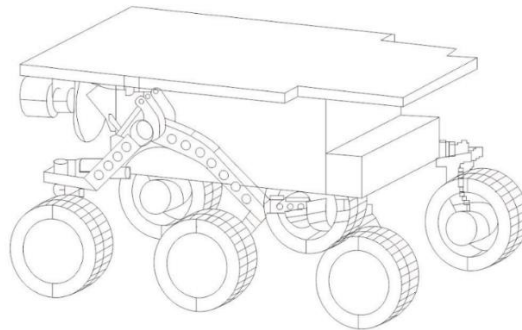
**3.9.2 Concepto 2: Robot triciclo.** La figura 28, se observa un robot triciclo que consta de dos ruedas accionadas por motores DC, y una “rueda loca” o ballcaster que permite ser un tercer punto de contacto, esto le da al robot un mayor grado de movimiento en giros.

**Figura 28. Robot tipo triciclo**



**3.9.3 Concepto 3: Robot rover.** La figura 29, se observa un robot rover, presenta un número significativo de ruedas, siendo de preferencia para ser robots exploradores en terrenos adversos.

**Figura 29. Robot rover**



**3.9.4 Selección concepto.** En la selección del robot se busca que los actuadores con sus respectivas “ruedas” puedan hacer giros sobre su propio eje, evitando al máximo la fricción entre las ruedas y el suelo, además de ser lo más pequeño posible.

Tabla 5, representa los criterios de selección de los 4 conceptos presentados anteriormente. Al final, el concepto que presente la mejor propuesta a desarrollar continuara el proceso a la siguiente fase.

**Tabla 5. Criterio de selección, robot**

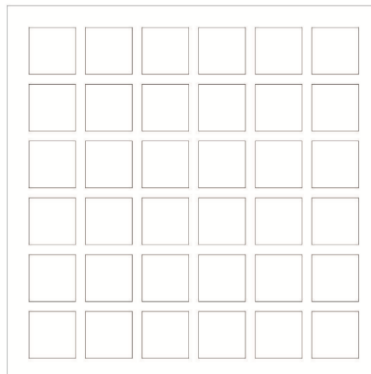
CRITERIOS DE SELECCIÓN	CONCEPTO ROBOT		
	1	2	3
Las dimensiones deben ser máximo de 100cm <sup>2</sup>	+	+	+
Su peso debe ser menor a 600 gr	-	+	+
Debe permitir un buen giro diferencial	+	+	-
Suma +	2	3	2
Suma -	1	0	1
Suma 0	0	0	0
Evaluación neta	1	3	1
Posición	3	1	2
Continúa?	NO	SI	NO

**3.9.5 Resultados.** Para la selección del concepto se usó conocimientos previos en la práctica de estos tres tipos de tracciones, escogiendo la que mayor se adapta al requerimiento solicitado. Se descarta el rover, debido a que está diseñado para terrenos pocos transitables, siendo un requerimiento no necesario en el sistema a desarrollar. Se selecciona el robot triciclo debido a su elevado grado de giro, por encima de las orugas, ya que estas presentan una considerable fricción cuando están girando sobre su propio eje. Ocasionando en la mayoría de casos no gire lo esperado.

### 3.10 GENERACIÓN CONCEPTO: TAPETE

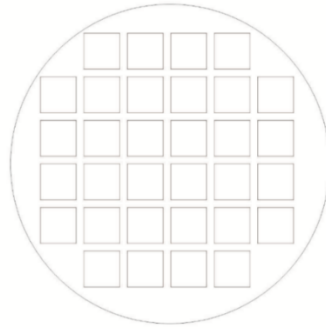
**3.10.1 Concepto 1: Tapete cuadrado.** La figura 30, se observa un conjunto de cuadrados, formando entre si un cuadrado.

**Figura 30. Tapete cuadrado**



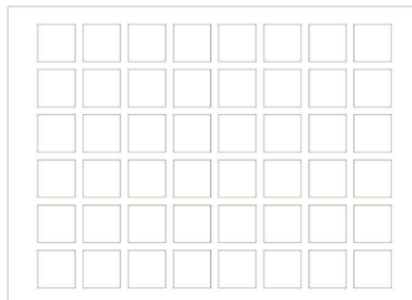
**3.10.2 Concepto 2: Tapete circular.** La figura 31, se observa un conjunto de cuadrados, circunscritos en una circunferencia.

**Figura 31. Tapete circular**



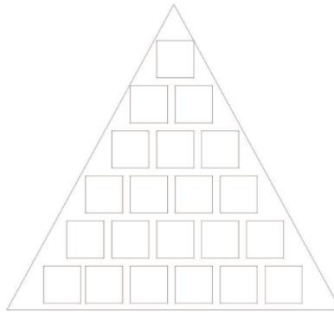
**3.10.3 Concepto 3: Tapete rectangular.** La figura 32, se observa un conjunto de cuadrados, formando entre si un rectángulo.

**Figura 32. Tapete rectangular**



**3.10.4 Concepto 4: Tapete triangular.** La figura 33, se observa un conjunto de cuadrados circunscritos en un triángulo.

**Figura 33. Tapete triangular**



**3.10.5 Selección concepto.** En el diseño del tapete se busca que presente una buena forma y distribución de los cuadrados, donde se desplazara el robot.

La tabla 6, representa los criterios de selección de los 4 conceptos presentados anteriormente. Al final, el concepto que presente la mejor propuesta a desarrollar continuara el proceso a la siguiente fase

**Tabla 6. Criterio de selección, tapete**

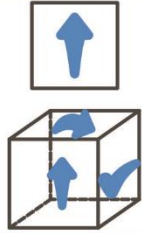
CRITERIOS DE SELECCIÓN	CONCEPTO TAPETE			
	1	2	3	4
Debe tener una superficie completamente lisa	+	+	+	+
Debe ser plegable	+	+	+	+
Sus dimensiones deben tener un mínimo de 250 cm <sup>2</sup>	+	+	+	+
Deber permitir la disposición de los niveles propuestos	+	-	-	-
El material debe soportar constante manipulación	+	+	+	+
Suma +	5	4	4	4
Suma -	0	1	1	1
Suma o	0	0	0	0
Evaluación neta	5	3	3	3
Posición	1	2	2	2
Continúa?	SI	NO	NO	NO

**3.10.6 Resultados.** Se selecciona el concepto del cuadrado por encima de los otros conceptos, debido a su simetría y permite hacer más conjugación de niveles.

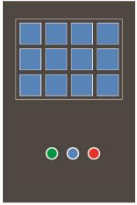
### 3.11 CARACTERÍSTICAS CONCEPTOS SELECCIONADOS

En las tablas 7,8,9 y 10, se representa un resumen de cada concepto seleccionado y sus debidas características para un posterior desarrollo de diseño.

**Tabla 7. Características fichas**

CARACTERÍSTICAS DE CONCEPTO FICHAS	CONCEPTO FICHAS
Fácil de almacenar en conjunto, apilable	
Fácil mantenimiento: Formas que permitan el fácil acceso,	
material lavable, fácil secado, cambio de piezas sencillo	
Permite libre posición y acoplamiento	
Bordes redondeados y superficies suaves	
Debe permitir la unificación de todos los movimientos unidireccionales propuestos	

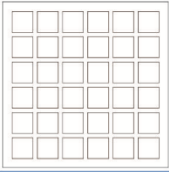
**Tabla 8. Características tablero**

CARACTERÍSTICAS DE CONCEPTO TABLERO	CONCEPTO TABLERO
Fácil acoplamiento de las fichas	
Debe permitir la buena distribución de los componentes	
Sus dimensiones deben ser menores a 120cm <sup>2</sup>	
Su peso debe ser menor a 1 kg	
Debe permitir la unificación de todos los movimientos unidireccionales propuestos	

**Tabla 9. Características robot**

CARACTERÍSTICAS DE CONCEPTO ROBOT	CONCEPTO ROBOT
Las dimensiones deben ser máximo de 100cm <sup>2</sup>	
Su peso debe ser menor a 600 gr	
Debe permitir un buen giro diferencial	

**Tabla 10. Características tapete**

CARACTERÍSTICAS DE CONCEPTO TAPETE	CONCEPTO TAPETE
Debe tener una superficie completamente lisa	
Debe ser plegable	
Sus dimensiones deben tener un mínimo de 250 cm <sup>2</sup>	
Deber permitir la disposición de los niveles propuestos	
El material debe soportar constante manipulación	

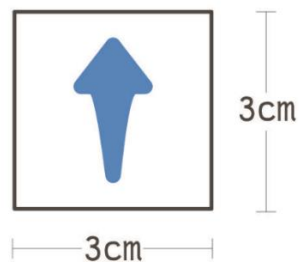
### 3.12 EVOLUCIÓN DE CONCEPTOS

#### 3.12.1 Ficha

**3.12.1.1 Tamaño cubo.** Se realizaron distintos cubos para optar por la mejor medida acorde en tamaño, seleccionando una medida por cara de **9cm<sup>2</sup>**.

En la figura 34, se observa la dimensión de la ficha.

**Figura 34. Dimensiones ficha**



**3.12.1.2 Instrucciones del cubo.** Las instrucciones presentadas a continuación, fueron escogidas para ser expuestas en cada superficie del cubo.

La figura 35, se observa cada una de las instrucciones visibles en las 6 caras del cubo.

**Figura 35. Set de instrucciones**



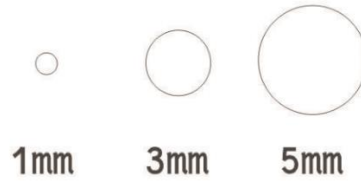
**3.12.1.3 Selección acople.** Para el acople entre la ficha y el tablero se hace necesario usar algún tipo de sistema sea mecánico, magnético o de contacto superficial.

Se seleccionaron imanes de neodimio, por las siguientes características:

- Su superficie es conductora.
- Se puede conseguir en distintos tamaños.
- Su precio en cantidad es económico.
- Se asegura una buena junción.
- Se requiere seleccionar un tamaño de imán. Y se selecciona el imán de 3mm de diámetro siendo esta medida justa para el tamaño de cada cara del cubo.

En la figura 36, se observa las 3 medidas de imán. Cada valor corresponde al diámetro del imán.

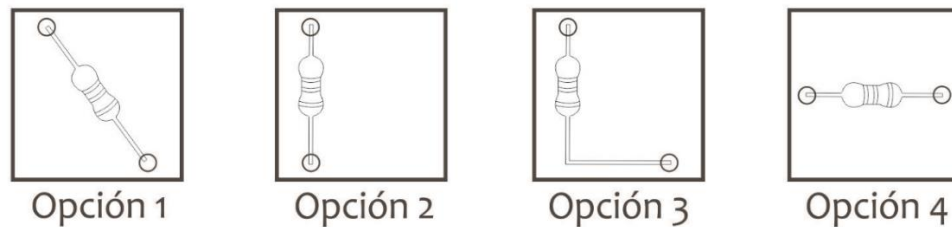
**Figura 36. Tamaños de imán**



**3.12.1.4 Ubicación resistencia.** Se busca posicionar la resistencia de tal forma que no interfiera en la disposición de los comandos impresos en cada cara. Por cuestiones de diseño y de no afectar la instrucción impresa en la cara de contacto, se elige la opción 1, como mejor alternativa.

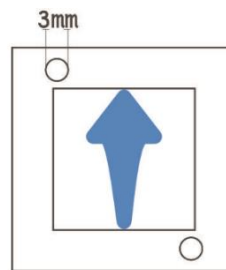
En la figura 37, se observa 4 distintas opciones de como posiblemente se puede ubicar la resistencia en el interior del cubo.

**Figura 37. Ubicación resistencia. Vista posterior**



En la figura 38, se esquematiza a escala la disposición de los imanes y el área real de trabajo de cada instrucción.

**Figura 38. Ficha. Vista anterior**



**3.12.1.5 Color fichas.** En la figura 39, se observa 5 propuestas de como posiblemente se pintaría el cubo.

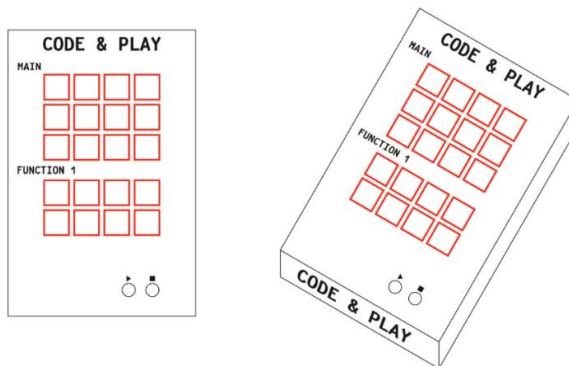
**Figura 39. Propuesta color fichas**



### **3.12.2 Tablero comandos**

**3.12.2.1 Propuesta final.** En la figura 40, se observa un diseño preliminar del tablero de comandos, teniendo seleccionado orientación vertical. El tablero cuenta con la función principal MAIN de 12 casillas y un FUNCTION 1 de 8 casillas, agregado el tablero cuenta con un botón de inicio y pausa.

**Figura 40. Tablero, Vista superior y perspectiva**

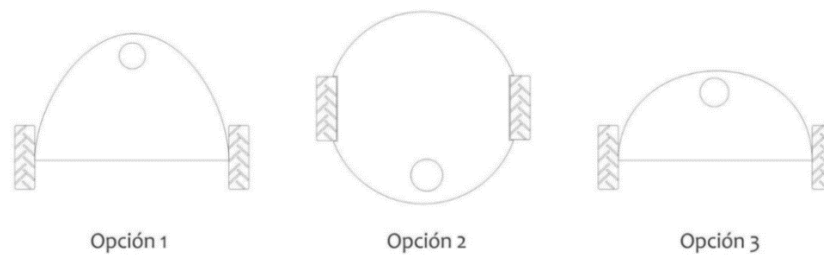


### 3.12.3 Robot

**3.12.3.1 Selección forma robot.** Se selecciona la segunda opción, ya que cuenta con mayor superficie, esto facilita la ubicación de componentes a la hora de desarrollar el circuito impreso.

En la figura 41, se observa tres opciones para desarrollar el robot triciclo.

**Figura 41. Robot. Vista superior**

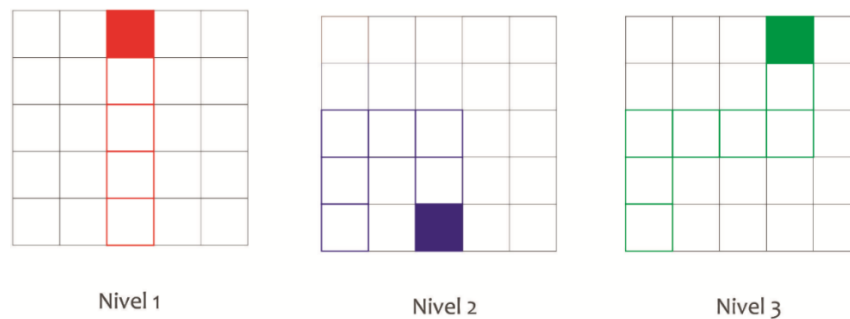


### 3.12.4 Tapete

**3.12.4.1 Selección niveles.** Se seleccionó un número de 6 niveles. Todos los niveles tienen el mismo tamaño sin importar si no abarcan todos los recuadros. La dimensión máxima del tapete es de 5x5.

En la figura 42, se observa los tres primeros niveles introductorios, donde el usuario tendrá la oportunidad de conocer la mecánica del juego planteada.

**Figura 42. Niveles 1,2,3**



En la figura 43, se observa los tres últimos niveles planteados, teniendo como inicio que, el usuario ya está familiarizado con la mecánica de juego. Estos tres últimos niveles representan una mayor carga mental, donde el usuario deberá hacer uso del FUNCTION 1.

**Figura 43. Niveles 4,5,6**



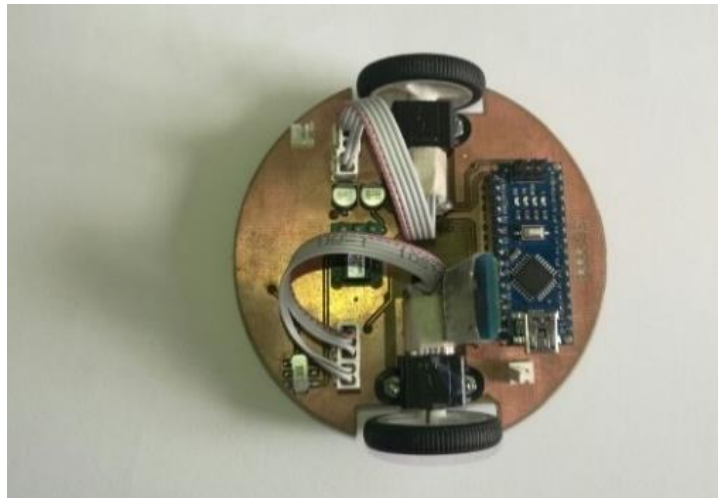
## 4. IMPLEMENTACIÓN Y RESULTADOS

### 4.1 IMPLEMENTACIÓN HARDWARE

Ya teniendo definido los componentes de hardware necesarios para el diseño del tablero y el robot se prosigue a su debido diseño y montaje.

En la figura 44, se observa el robot ensamblado con todos sus periféricos.

**Figura 44. Robot móvil, vista superior**



Para el diseño del robot se emplearon dos micro motores DC de marca Pololu. 150:1 (200 RPM), 120mA sin carga a 6V nominales, con un torque de 2.9 kg-cm.

El microcontrolador es una atmega328p, está montado sobre la tarjeta Arduino NANO, con una alimentación de entrada de 5-12V.

En la figura 45, se observa el circuito impreso del tablero de comandos, con sus respectivos periféricos, tales como: bluetooth, Arduino nano y los multiplexores analógicos.

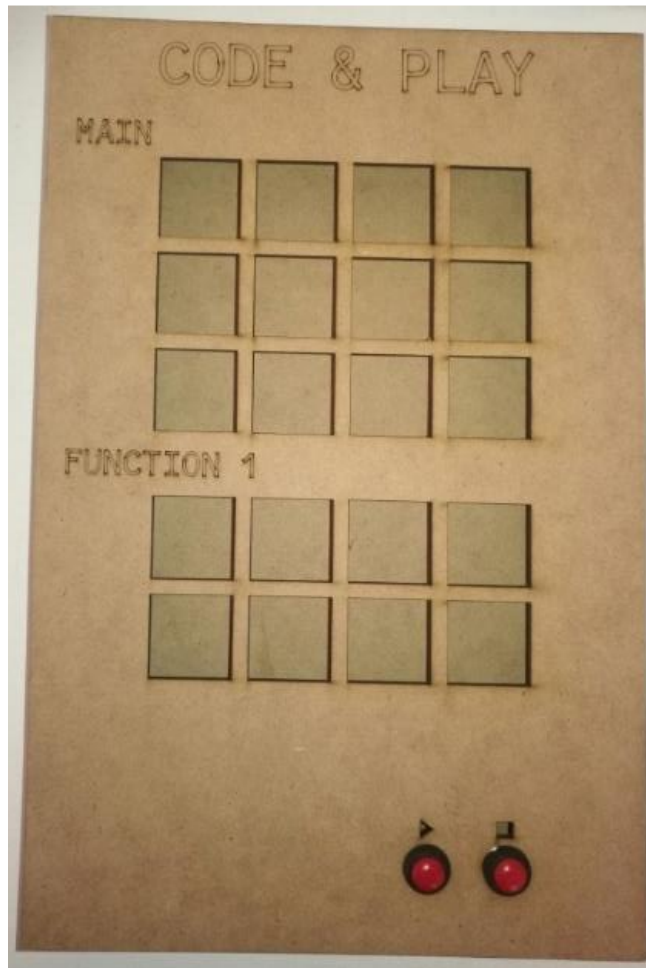
**Figura 45. PCB Tablero comandos, vista superior**



## 4.2 IMPLEMENTACIÓN DISEÑO

En la figura 46, se observa el diseño del tablero de comandos, el cual fue elaborado en cortadora laser.

**Figura 46. Diseño Tablero de Comandos**



En la figura 47, se observa el diseño final de la ficha con sus respectivos imanes. La ficha fue elaborada en impresión 3D, empleando como material ABS.

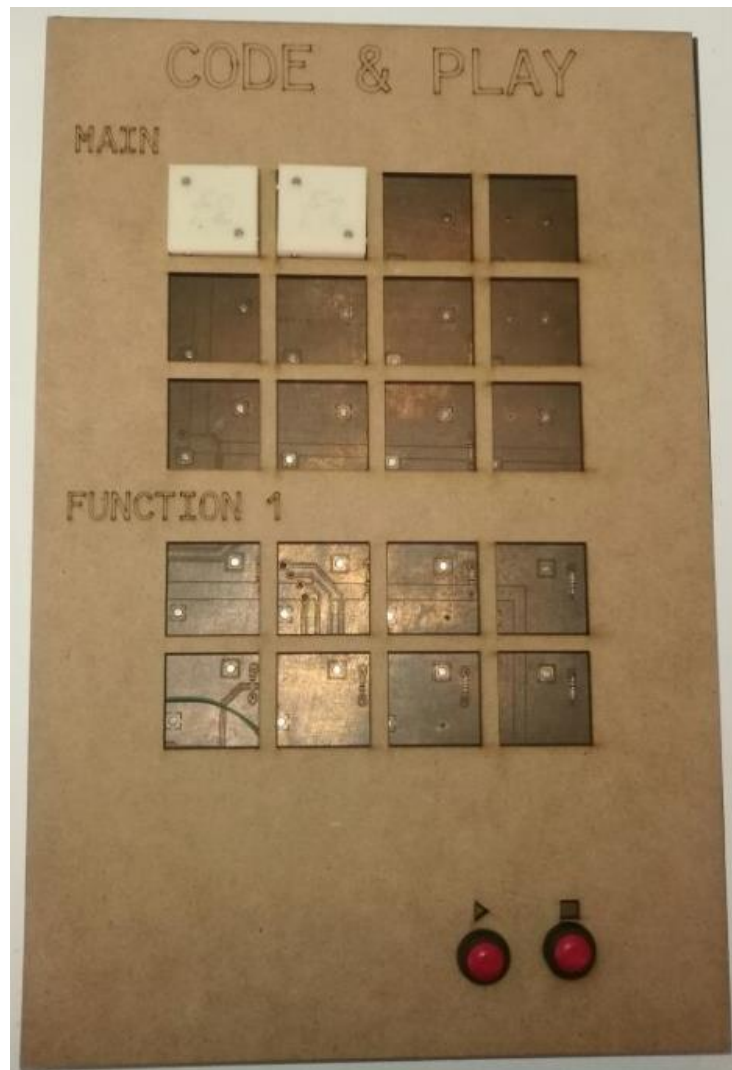
**Figura 47. Diseño Ficha**



### **4.3 INTEGRACION DISEÑO Y HARDWARE**

En la figura 48, se observa el acople de la ficha en el tablero, con todo el hardware interno.

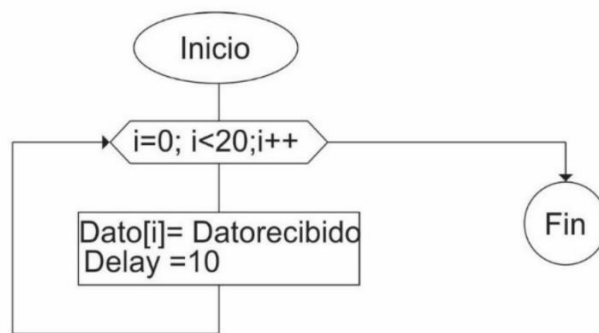
**Figura 48. Tablero y PCB Integrada**



#### 4.4 ALGORITMO TABLERO DE COMANDOS

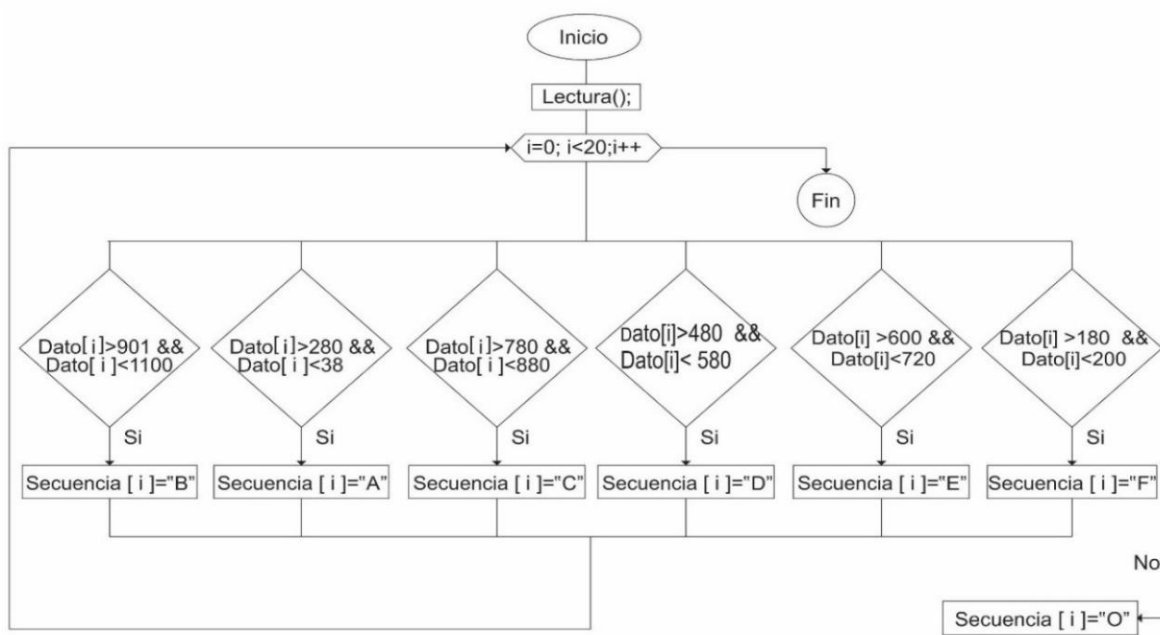
**4.4.1 Función lectura datos.** En la figura 49, se observa el diagrama de flujo de la función que se encarga de leer las fichas del tablero de comandos, siendo un total de 20 instrucciones y almacenándolas en la variable Dato.

**Figura 49. Función lectura datos**



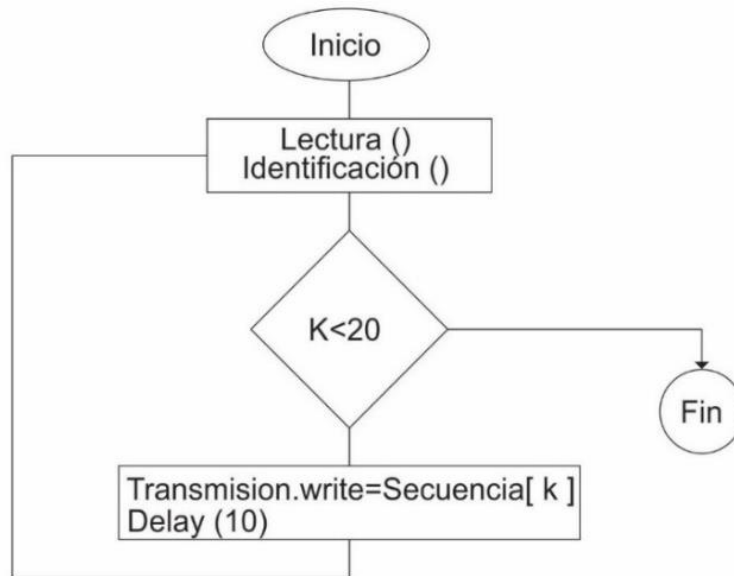
**4.4.2 Función identificación datos.** En la figura 50, se observa el diagrama de flujo de la función identificación datos, cada cara del cubo tiene un valor de resistencia distinto dependiendo de la instrucción indicada. Debido a que cada configuración es un divisor de tensión obtendremos en el MCU un valor analógico el cual dependiendo del rango identificamos ese valor y le otorgamos en este caso una letra.

**Figura 50. Función identificación datos**



**4.4.3 Función transmisión datos.** En la figura 51, se observa la función transmisión de datos, después de haber realizado la lectura e identificación, se prosigue a enviar los datos via bluetooth.

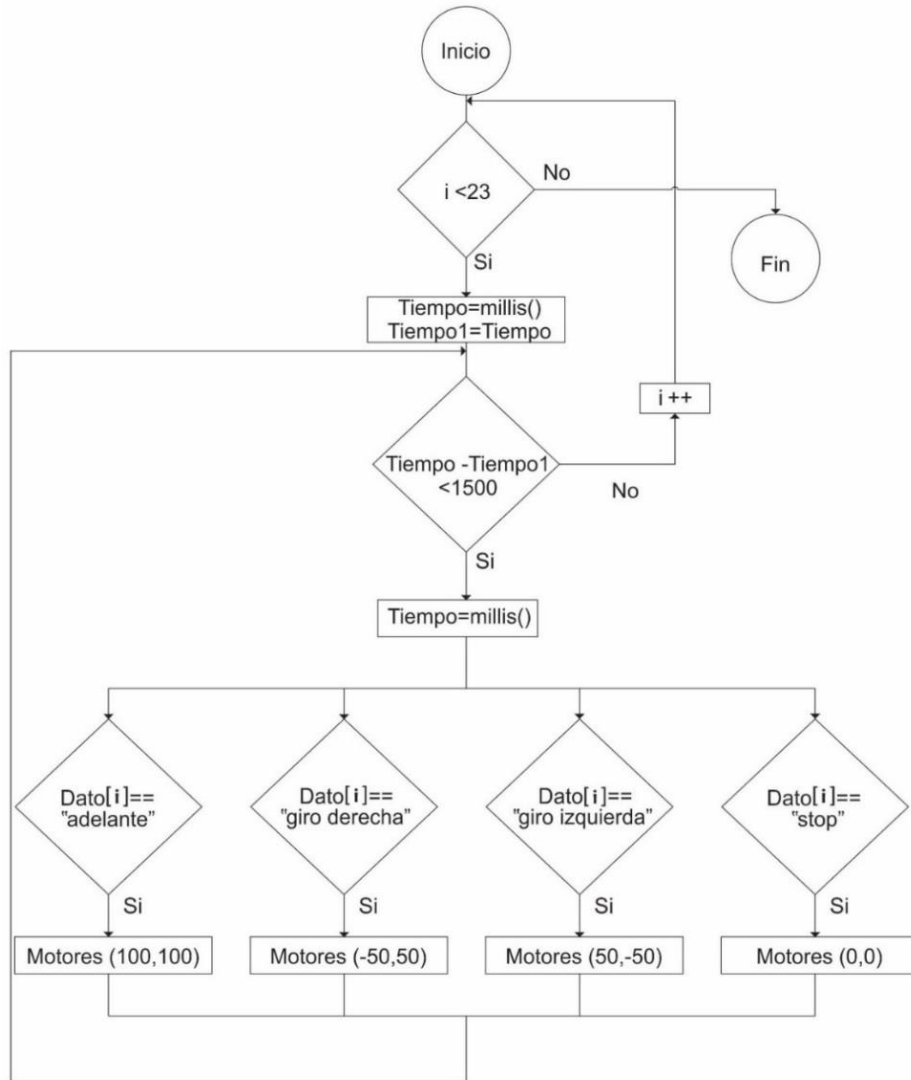
**Figura 51. Función transmision datos**



## 4.5 ALGORITMO ROBOT

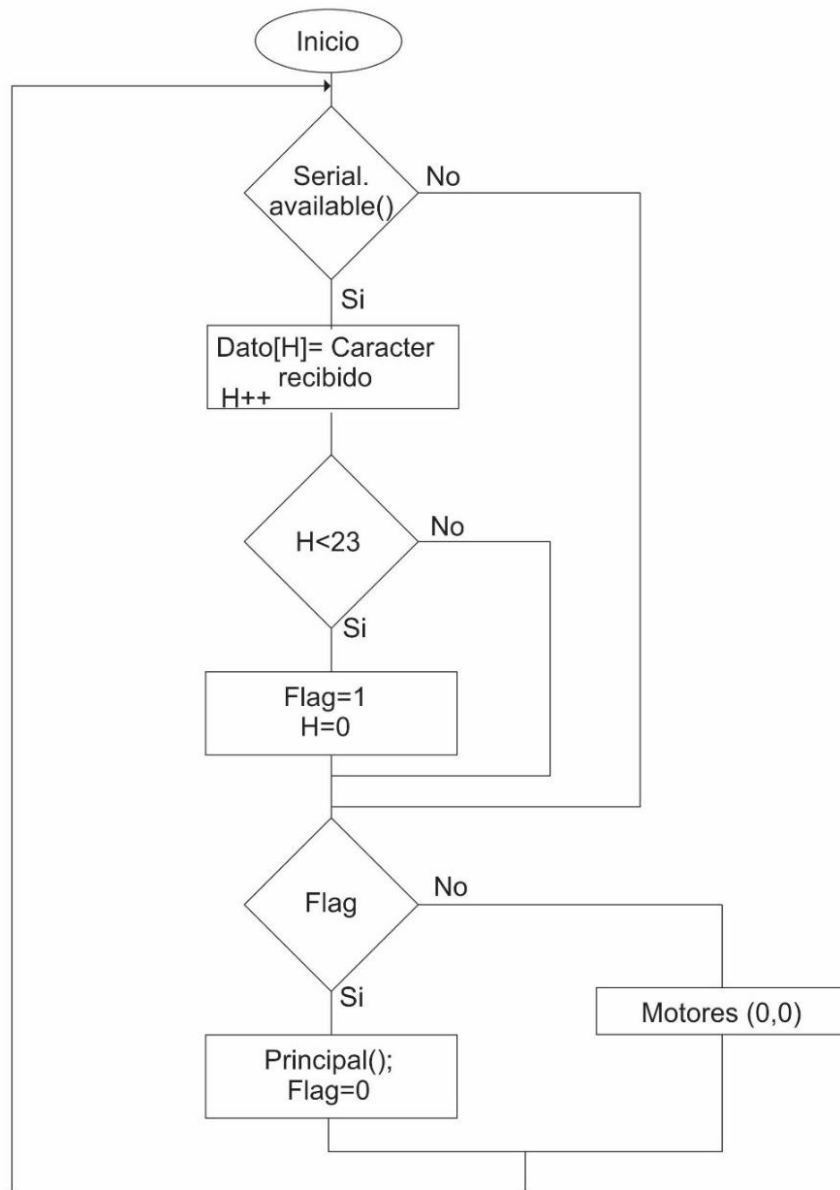
**4.5.1 Función principal.** En la figura 52, se observa el diagrama de flujo de la función principal, esta función es la que se encarga de realizar los movimientos en los motores del robot, dependiendo de cuál instrucción a ejecutar. Cabe resaltar que para realizar cada movimiento se tiene un tiempo de segundo y medio, la función millis es la encargada de llevar el tiempo real del MCU.

Figura 52. Función principal



**4.5.2 Recepción datos.** En la figura 53, se observa el diagrama de flujo de recepción de datos, esta función se encarga de almacenar los datos enviados anteriormente vía bluetooth. Conforme reciba los datos se proseguirá seguir a la función principal, encargada de ejecutar cada movimiento.

Figura 53. Recepción datos



## 5. CONCLUSIONES

- En este trabajo se desarrolló un robot móvil que busca facilitar el aprendizaje de habilidades de programación en los niños de 7-12 años. El sistema está basado en dos plataformas Arduino interconectadas a través de una interfaz bluetooth. Los resultados evidenciaron que efectivamente el robot móvil le permite a un niño jugar a lograr retos de creciente nivel de dificultad. Los niveles de dificultad se diseñaron con el propósito de facilitar conceptos relacionados con la programación.
- El juego diseñado puede convertirse en una alternativa diferente a las que actualmente se ofrecen en el mercado y que podría ser de interés a los padres, puesto que los juegos online a veces son de difícil supervisión y pueden traer consecuencias nocivas para el desarrollo del niño.

## REFERENCIAS BIBLIOGRÁFICAS

[1] GARITAONANDIA, C. FERNÁNDEZ, C.E. OLEAGA J.A. "Las tecnologías de la información y de la comunicación y su uso por los niños y adolescentes". Doxa común, 2005, no 3. 45-64 p

[2] M, AIDÉ. VELANDIA, A. F, HUMBERTO. MORALES, F. DUARTE, E. "Utilización de material didáctico para la enseñanza de los conceptos de ciencia y tecnología en niños" Volumen II. 2011. 35-43

[3] Chiquimakers [online]. Disponible en: <https://chiquimakers.com/>. [Fecha de consulta:10-oct-2017. Hora:xxx]

[4] Cordero, C.P. 2010. "La importancia del juego y los juguetes para el desarrollo integral de los niños/as de educación infnartil. Rev. la Educ. en Extrem. 10-20 p

[5] PALMA, C.A. SARMIENTO, R.E. "Estado del arte sobre experiencias de enseñanza de programación a niños y jóvenes para el mejoramiento de las competencias matemáticas en primaria. Colombia. 2014

[6] ECHEBURÚA, E. "Factores de riesgo y factores de protección en la adicción a las nuevas tecnologías y redes sociales en jóvenes y adolescentes". Rev. Española Drog. Volumen II. No 4. 2012. 435-447p

[7] Code.org, 2013. "Todo el mundo debería saber programar". Disponible en: <https://youtu.be/X5Wkp1gsNik>. [Fecha de consulta:xxx. Hora:xxx]

[8] PAPERT, S. "Lenguaje Logo". Disponible en <http://www.larevistainformatica.com/Logo.htm>. [Fecha de consulta:xxx. Hora:xxx]

[9] RODRÍGUEZ, Y.F.B. ROZO, O.A.C. “Desarrollo de un lenguaje de programación con fines educativos basado en la filosofía de logo”. Colombia. 2008

## BIBLIOGRAFÍA

CODE.ORG, 2013. “Todo el mundo debería saber programar”. Disponible en: <https://youtu.be/X5Wkp1gsNik>. [Fecha de consulta:xxx. Hora:xxx]

CODERO, C.P. 2010. “La importancia del juego y los juguetes para el desarrollo integral de los niños/as de educación infnatil. Rev. la Educ. en Extrem. 10-20 p

CHIQUIMAKERS [online]. Disponible en: <https://chiquimakers.com/>. [Fecha de consulta:10-oct-2017. Hora:xxx]

ECHEBURÚA, E. “Factores de riesgo y factores de protección en la adicción a las nuevas tecnologías y redes sociales en jóvenes y adolescentes”. Rev. Española Drog. Volumen II. No 4. 2012. 435-447p

GARITAONANDIA, C. FERNÁNDEZ, C.E. OLEAGA J.A. ”Las tecnologías de la información y de la comunicación y su uso por los niños y adolescentes”. Doxa común, 2005, no 3. 45-64 p

M, AIDÉ. VELANDIA, A. F, HUMBERTO. MORALES, F. DUARTE, E. “Utilización de material didáctico para la enseñanza de los conceptos de ciencia y tecnología en niños” Volumen II. 2011. 35-43

PALMA, C.A. SARMIENTO, R.E. “Estado del arte sobre experiencias de enseñanza de programación a niños y jóvenes para el mejoramiento de las competencias matemáticas en primaria. Colombia. 2014

PAPERT, S. “Lenguaje Logo”. Disponible en <http://www.larevistainformatica.com/Logo.htm>. [Fecha de consulta:xxx. Hora:xxx]

RODRÍGUEZ, Y.F.B. ROZO, O.A.C. “Desarrollo de un lenguaje de programación con fines educativos basado en la filosofía de logo”. Colombia. 2008

# ANEXOS

## ANEXO A. Tarjeta interfaz electrónica

Figura A.1: diseño esquemático, tablero de comandos.

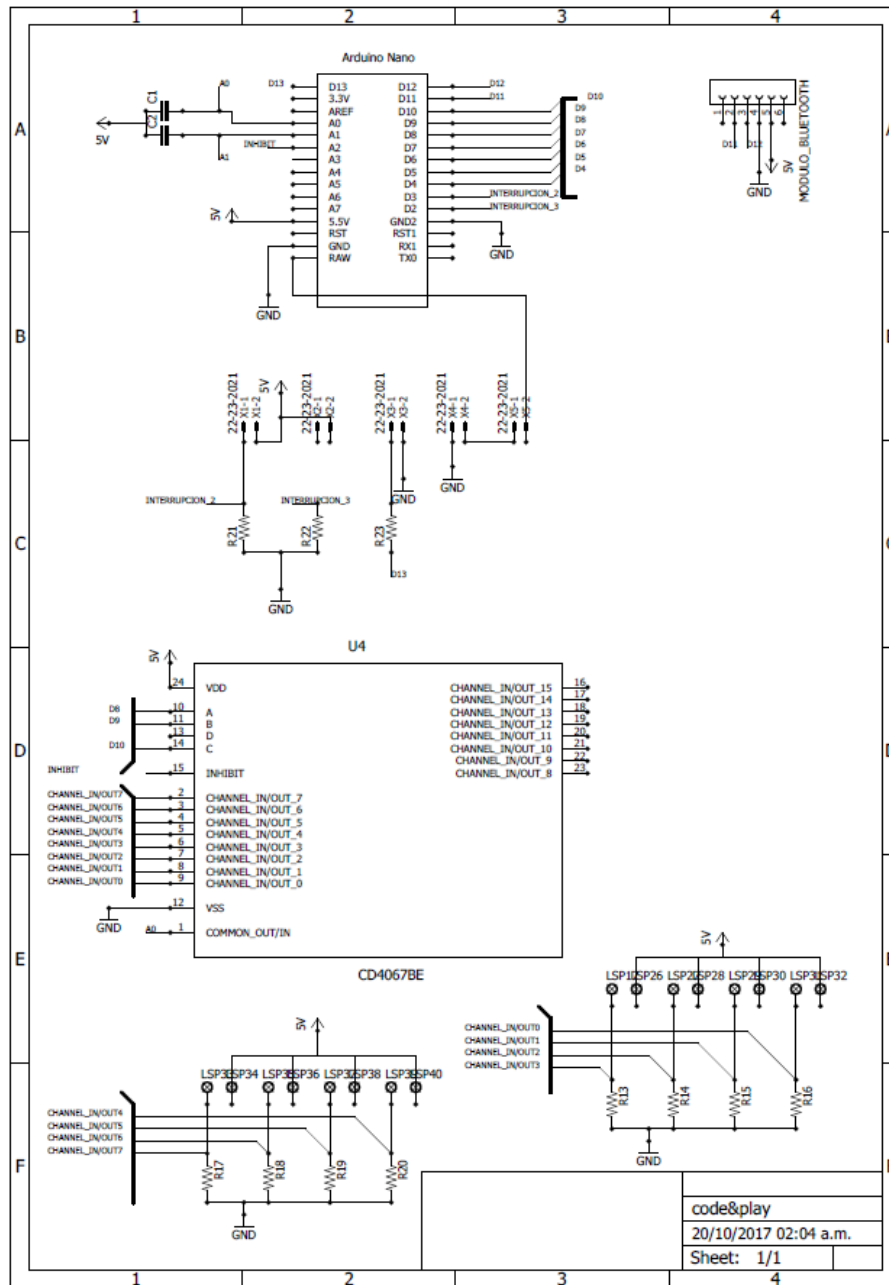


Figura A.2: diseño esquemático, tablero de comandos.

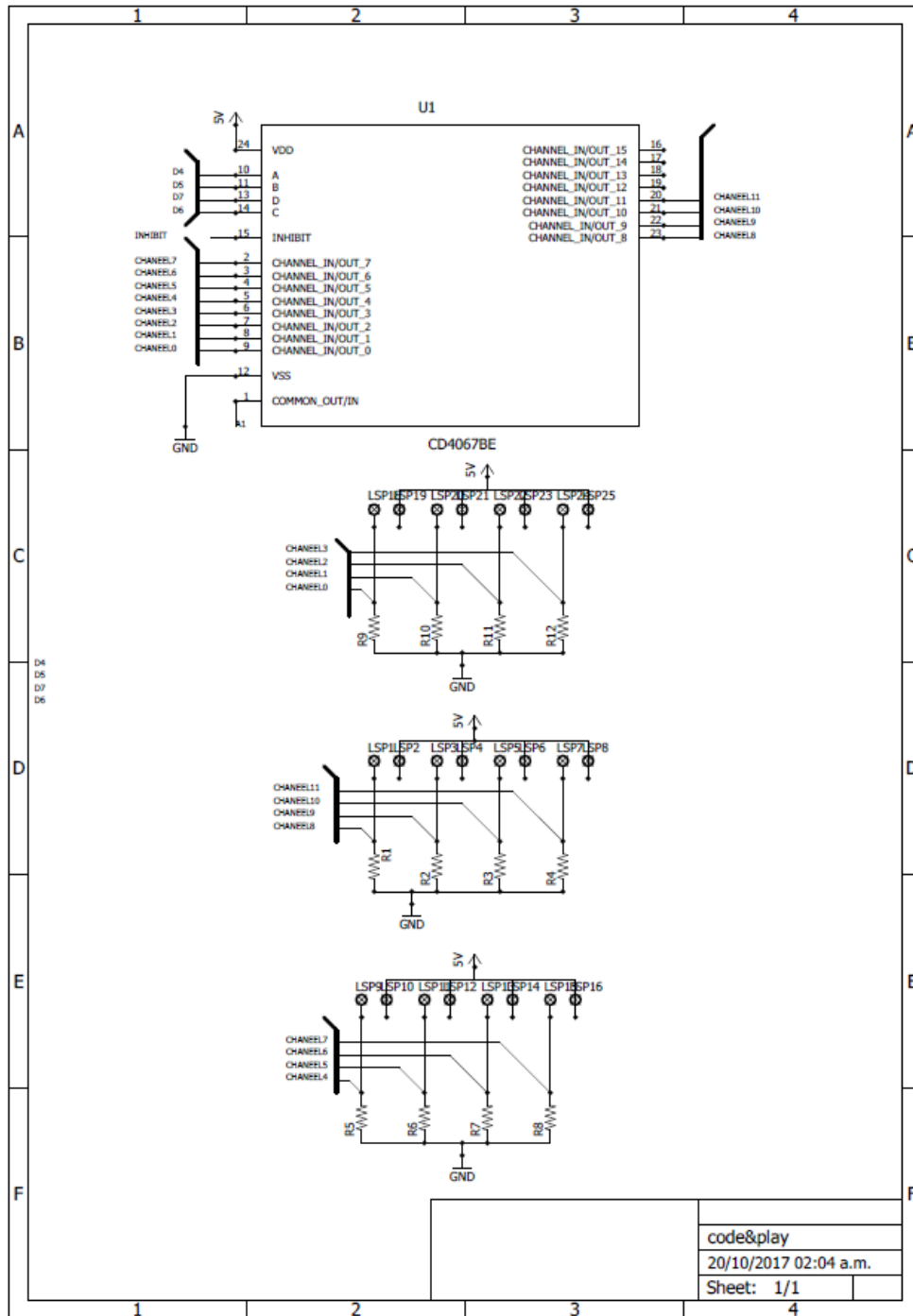


Figura A.3: Diseño circuito impreso, tablero de comandos, vista top.

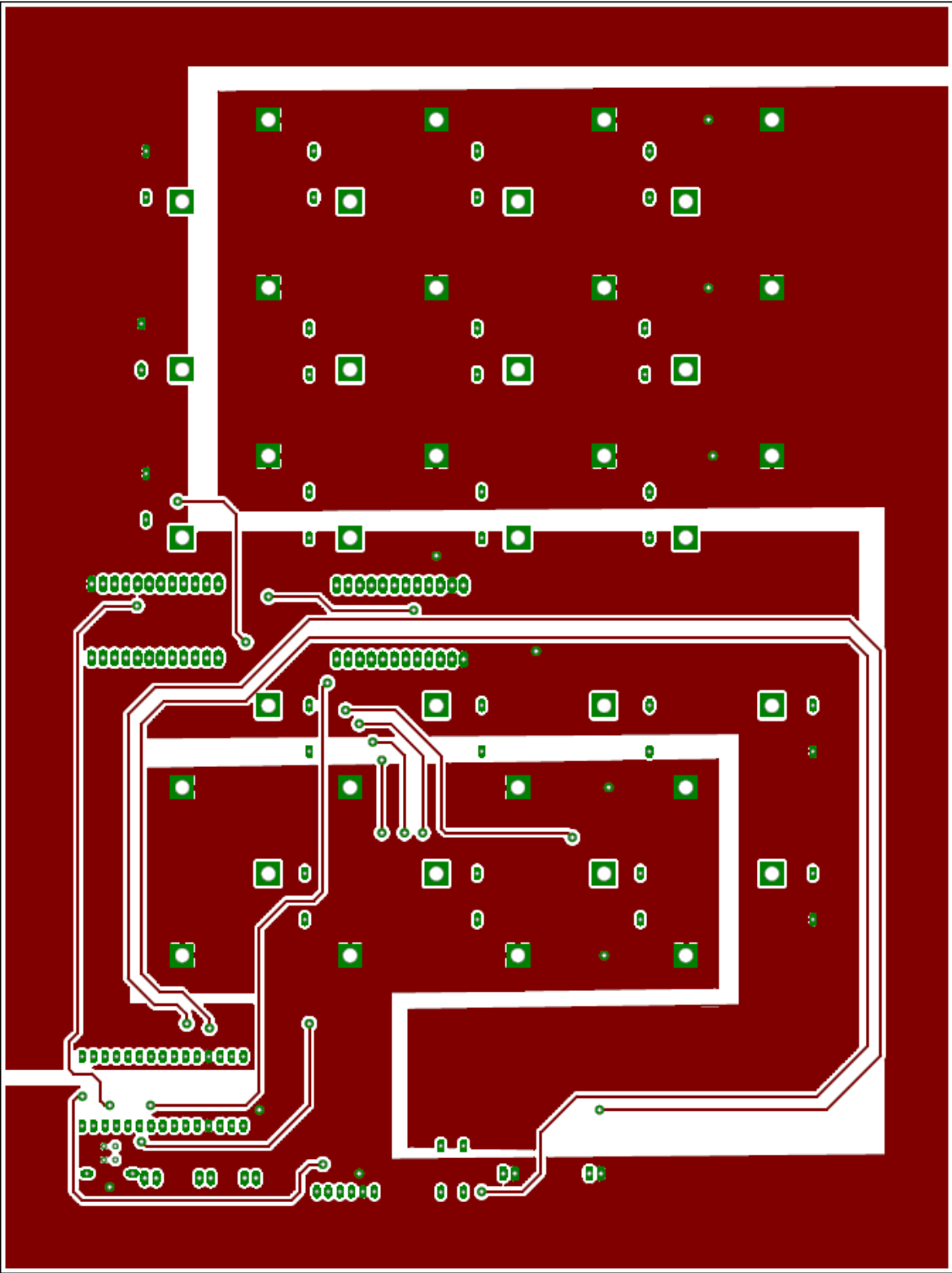
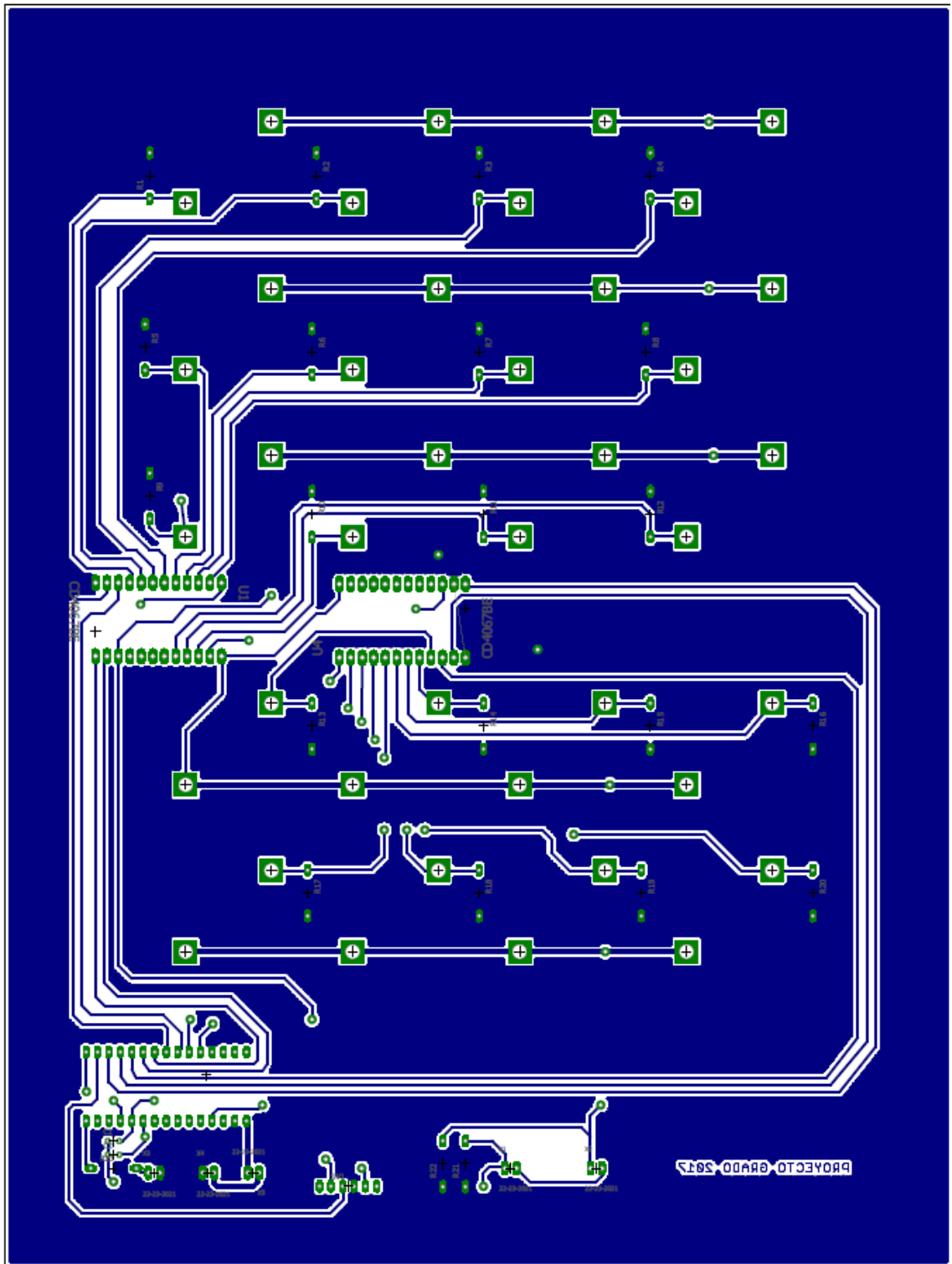
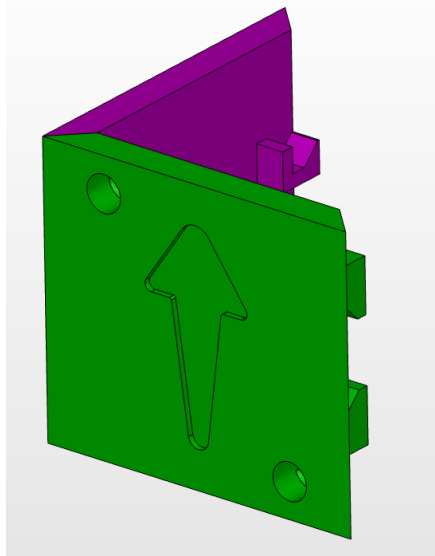


Figura A.4: Diseño circuito impreso, tablero de comandos, vista bottom.

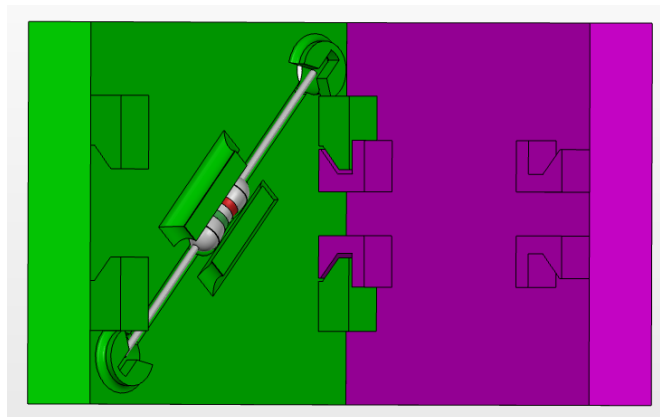


## ANEXO B. Diseño 3D Fichas

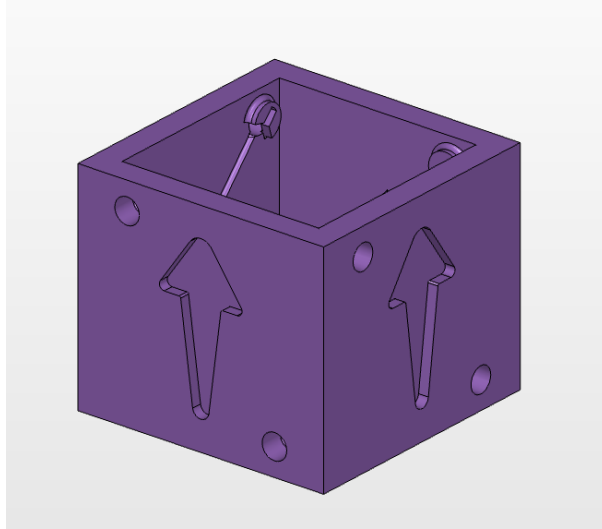
**Figura B.1: Diseño ficha. Concepto, caras por separado, vista frontal en perspectiva.**



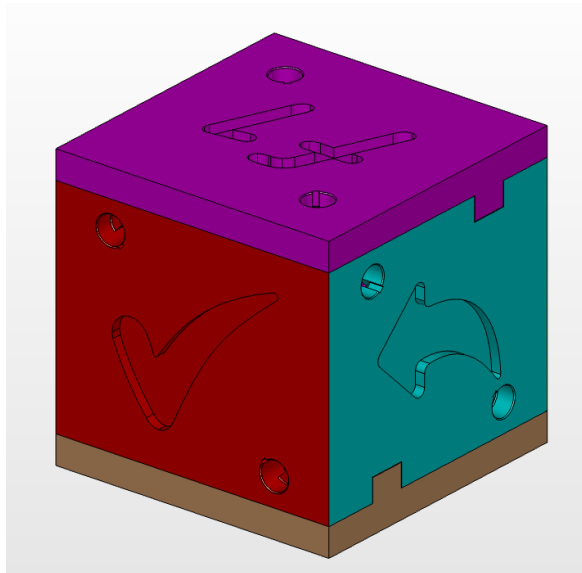
**Figura B.2: Diseño ficha. Concepto, caras por separado, vista posterior.**



**Figura B.3: Diseño ficha. Concepto, caras unidas, vista en perspectiva.**

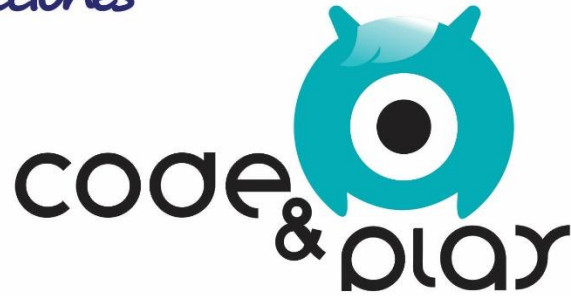


**Figura B.4: Diseño ficha. Cubo ensamblado**



## ANEXO C. Manual de usuario

### Manual de Instrucciones

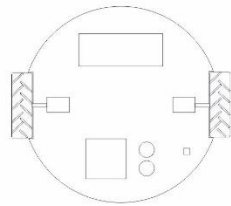


#### PRESENTACIÓN

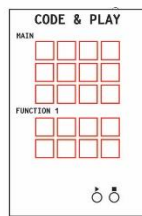
Este manual es una guía de ayuda, se recomienda que un adulto le explique el funcionamiento y la mecánica de juego y del dispositivo al niño

#### CONTENIDO DEL KIT

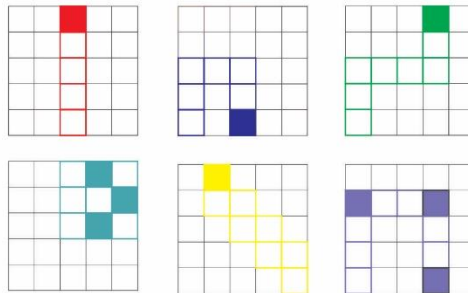
Robot x1



Tablero comando x 1



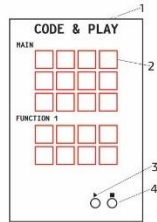
Niveles x 6



## DESCRIPCIÓN DEL PRODUCTO

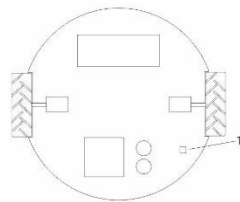
Esta sección proporciona la descripción de los aspectos básicos del dispositivo. Le ayudará a conocer más sobre la apariencia del dispositivo antes de usarlo.

## COMANDOS DE TABLERO



- 1 Encendido on/of
- 2 Casillas de fichas
- 3 Botón inicio
- 4 Botón pausa

## ROBOT

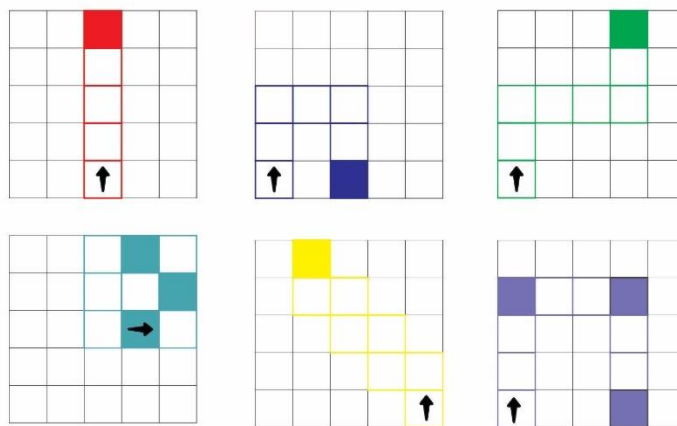


- 1 Encendido on/of

## MECÁNICA DEL JUEGO

El juego tendrá un total de 6 niveles. Para seguir avanzando de un nivel a otro es necesario culminar el nivel anterior, de lo contrario se hace difícil completar el juego.

Se debe disponer el robot en una determinada posición, mostrada a continuación

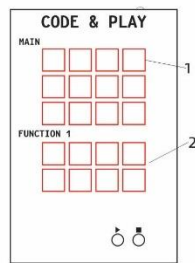


El main es la función principal y cuenta con 12 casillas. En cada casilla se introduce una ficha y se dispone a construir el algoritmo para el nivel requerido

Para los niveles 5 y 6, es necesario hacer uso de FUCTION 1



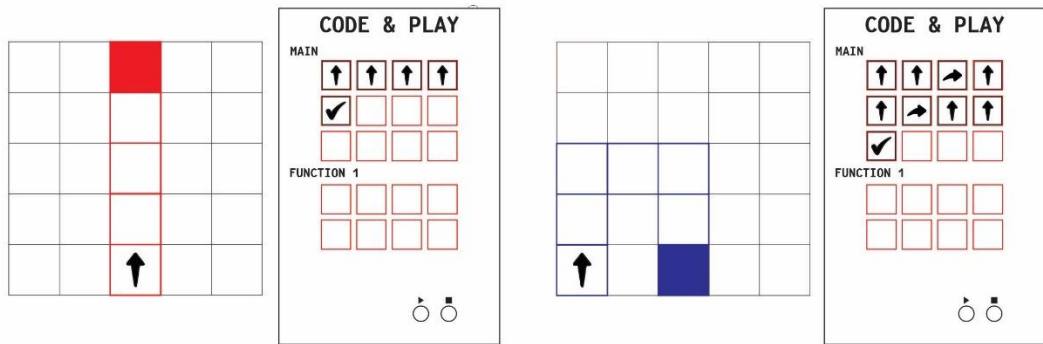
1 Ficha



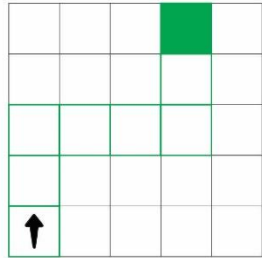
1 Main  
2 Fuction 1

## SOLUCIÓN DE NIVELES

### NIVELES 1 - 2



### NIVELES 3 - 4



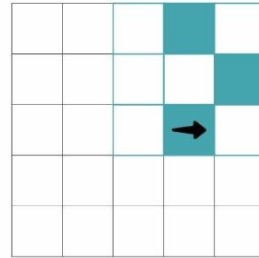
**CODE & PLAY**

MAIN

↑	↑	↘	↑
↑	↑	←	↑
↑	✓		

FUNCTION 1


○ ○



**CODE & PLAY**

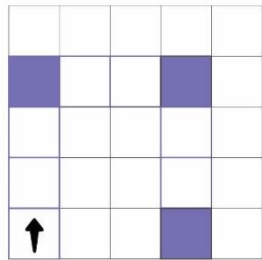
MAIN

✓	↑	↘	↑
✓	↑	↘	↑
✓			

FUNCTION 1


○ ○

### NIVELES 5 - 6



**CODE & PLAY**

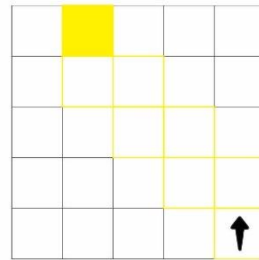
MAIN

f!	f!	f!	

FUNCTION 1

↑	↑	↑	↘
✓			

○ ○



**CODE & PLAY**

MAIN

f!	f!	f!	↑
✓			

FUNCTION 1

↑	←	↑	↘

○ ○