



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

**CARRERA DE INGENIERÍA EN ELECTRÓNICA Y REDES DE
COMUNICACIÓN**

**“PROTOTIPO ELECTRÓNICO PROGRAMADO MEDIANTE
BLOQUES CON UNA APLICACIÓN MÓVIL PARA LA ENSEÑANZA
DE PROGRAMACIÓN A NIÑOS DE 10 A 12 AÑOS EN LA CIUDAD
DE IBARRA”**

**TRABAJO DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO
DE INGENIERO EN ELECTRÓNICA Y REDES DE
COMUNICACIÓN**

AUTOR: WILMAN ROBERTO SUÁREZ ZAMBRANO

DIRECTOR: ING. CARLOS ALBERTO VÁSQUEZ AYALA

IBARRA-ECUADOR

2016



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

La Universidad Técnica del Norte dentro del proyecto Repositorio Digital Institucional, determinó la necesidad de disponer de textos completos en formato digital con la finalidad de apoyar los procesos de investigación, docencia y extensión de la Universidad.

Por medio del presente documento dejo sentada mi voluntad de participar en este proyecto, para lo cual pongo a disposición la siguiente información.

DATOS DEL CONTACTO	
Cédula de Identidad	100371266-6
Apellidos y Nombres	Suárez Zambrano Wilman Roberto
Dirección	Av. 17 de Julio 7-201
E-mail	wrsuarez@utn.edu.ec
Teléfono móvil	999325343
DATOS DE LA OBRA	
Título	Prototipo electrónico programado mediante bloques con una aplicación móvil para la enseñanza de programación a niños de 10 a 12 años en la ciudad de Ibarra.
Autor	Wilman Roberto Suárez Zambrano
Fecha	Julio del 2016
Programa	Pregrado
Título	Ingeniero en Electrónica y Redes de Comunicación
Director	Ing. Carlos Vásquez A.

2. AUTORIZACIÓN DE USO A FAVOR DE LA UNIVERSIDAD

Yo, Suárez Zambrano Wilman Roberto, con cédula de identidad No. 100371266-6, en calidad de autor y titular de los derechos patrimoniales de la obra o trabajo de grado descrito anteriormente, hago entrega del ejemplar respectivo en forma digital y autorizo a la Universidad Técnica del Norte, la publicación de la obra en el Repositorio Digital Institucional y del uso del archivo digital en la Biblioteca de la Universidad con fines académicos, para ampliar la disponibilidad de material y como apoyo a la educación, investigación y extensión, en concordancia con la Ley de Educación Superior Artículo 144.

Firma: 

Nombre: Suárez Z. Wilman

Cédula: 100371266-6

Ibarra, Julio 2016



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

**CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE GRADO A
FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE**

Yo, Suárez Zambrano Wilman Roberto, con cédula de identidad No. 100371266-6, manifiesto mi voluntad de ceder a la Universidad Técnica del Norte los derechos patrimoniales consagrados en la Ley de Propiedad Intelectual del Ecuador artículos 4, 5 y 6, en calidad de autor del trabajo de grado con el tema: **PROTOTIPO ELECTRÓNICO PROGRAMADO MEDIANTE BLOQUES CON UNA APLICACIÓN MÓVIL PARA LA ENSEÑANZA DE PROGRAMACIÓN A NIÑOS DE 10 A 12 AÑOS EN LA CIUDAD DE IBARRA**. Que ha sido desarrollado con propósito de obtener el título de Ingeniero en Electrónica y Redes de Comunicación de la Universidad Técnica del Norte, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En mi condición de autor me reservo los derechos morales de obra antes citada. En concordancia suscribo en el momento que hago entrega del trabajo final en formato impreso y digital a la Biblioteca de Universidad Técnica del Norte.

Suárez Zambrano Wilman Roberto

100371266-6

Ibarra, Julio 2016



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICACIÓN

INGENIERO CARLOS VÁSQUEZ, DIRECTOR DEL PRESENTE TRABAJO DE
TITULACIÓN CERTIFICA:

Que, el presente trabajo de titulación “PROTOTIPO ELECTRÓNICO PROGRAMADO
MEDIANTE BLOQUES CON UNA APLICACIÓN MÓVIL PARA LA ENSEÑANZA DE
PROGRAMACIÓN A NIÑOS DE 10 A 12 AÑOS EN LA CIUDAD DE IBARRA”. Ha sido
desarrollado Suárez Zambrano Wilman Roberto bajo mi supervisión.

Es todo cuanto puedo certificar en honor a la verdad.

Ing. Carlos Vásquez

100242498-2

DIRECTOR



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CONSTANCIAS

Yo, Suárez Zambrano Wilman Roberto declaro bajo juramento que el trabajo aquí descrito es de mi autoría, y que este no ha sido previamente presentado para ningún grado o calificación profesional y que he consultado las referencias bibliográficas que se presentan en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondiente a este trabajo, a la Universidad Técnica del Norte, según lo establecido en las leyes de propiedad intelectual, reglamentos y normativa vigente de la Universidad Técnica del Norte.

En la ciudad de Ibarra, Julio del 2016.

EL AUTOR

Suárez Zambrano Wilman Roberto

CI: 100371266-6

DEDICATORIA

Dedicado a la memoria de María...

Una guerrera que luchó y me apoyó en todo momento, que empezó este reto junto a mí pero que el tiempo no le permitió ver el resultado de su esfuerzo y el triunfo de uno de los suyos.

AGRADECIMIENTO

Agradezco a todas las personas que fueron parte de este largo camino, a las que me apoyaron de una u otra forma y a todas las que estuvieron conmigo en todo momento, familiares, compañeros, docentes y amigos.

Agradecimiento especial a Anita, Clarisse, Rafael, María Elena, Marcelo, Luis y Diego.

CONTENIDO

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE	II
CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE GRADO A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE	¡Error! Marcador no definido.
CERTIFICACIÓN	¡Error! Marcador no definido.
CONSTANCIAS	¡Error! Marcador no definido.
DEDICATORIA	VII
AGRADECIMIENTO.....	VIII
CONTENIDO	IX
ÍNDICE DE ILUSTRACIONES	XVI
ÍNDICE DE TABLAS	XXII
ÍNDICE DE ECUACIONES	XXIII
RESUMEN	XXIV
ABSTRACT	XXV
PRESENTACIÓN	XXVI
CAPÍTULO I. ANTECEDENTES.....	1
1.1. TEMA	1
1.2. PROBLEMA.....	1
1.3. OBJETIVOS.....	2
1.3.1. OBJETIVO GENERAL.....	2
1.3.2. OBJETIVOS ESPECIFICOS.....	2

1.4.	ALCANCE	3
1.5.	JUSTIFICACIÓN	5
CAPÍTULO 2. FUNDAMENTACIÓN TEÓRICA		7
2.1.	TEORÍAS DE APRENDIZAJE	7
2.1.1.	TEORÍA CONDUCTISTA	8
2.1.2.	TEORÍA COGNITIVA	11
2.1.3.	TEORÍA CONSTRUCTIVISTA	15
2.1.4.	TABLA COMPARATIVA ENTRE TEORÍAS DE APRENDIZAJE	18
2.2.	ETAPAS DEL DESARROLLO COGNITIVO SEGÚN JEAN PIAGET ..	18
2.2.1.	ETAPA DE OPERACIONES CONCRETAS 10 – 12 AÑOS	20
2.3.	ENCUESTAS	21
2.4.	TECNOLOGÍA, INCIDENCIA EN LOS NIÑOS	23
2.5.	PROGRAMACIÓN ENFOCADA A NIÑOS	25
2.5.1.	INTRODUCCIÓN	25
2.5.2.	DEFINICIONES BÁSICAS	26
2.5.3.	PROGRAMACIÓN MEDIANTE BLOQUES	28
2.5.4.	HERRAMIENTAS PARA LA ENSEÑANZA DE PROGRAMACIÓN A NIÑOS. 31	
2.6.	SISTEMAS OPERATIVOS MÓVILES MAS COMUNES QUE SOPORTAN APLICACIONES MEDIANTE BLOQUES	37

2.6.1.	SISTEMA OPERATIVO iOS	38
2.6.2.	SISTEMA OPERATIVO ANDROID.....	40
2.7.	TECNOLOGÍAS INALÁMBRICAS	42
2.7.1.	WI-FI	44
2.7.2.	BLUETOOTH.....	45
2.7.3.	ZIGBEE.....	47
2.7.4.	TABLA COMPARATIVA ENTRE TECNOLOGÍAS INALÁMBRICAS	49
2.8.	COMPONENTES ELECTRÓNICOS	50
2.8.1.	PLATAFORMA ELECTRÓNICA ARDUINO.....	50
2.8.2.	TABLA COMPARATIVA DE PLACAS ARDUINO.....	65
2.8.3.	MOTORES DC (CORRIENTE CONTÍNUA)	66
2.8.4.	MICRO MOTOR	68
2.8.5.	SENSOR ULTRASÓNICO HC-SR04.....	69
2.8.6.	MÓDULO BLUETOOTH HC-05.....	72
2.8.7.	CIRCUITO INTEGRADO L293D	73
2.8.8.	BATERÍAS.....	75
2.8.9.	LIGHT EMITTER DIODE – LED.....	77

2.9.	SOFTWARE	79
2.9.1.	ENTORNO DE DESARROLLO INTEGRADO (IDE ARDUINO)	79
2.9.2.	EAGLE	80
2.9.3.	ISIS PROTEUS	80
2.10.	POSIBLES MATERIALES A UTILIZAR EN LA CARCASA DEL PROTOTIPO	81
2.10.1.	MADERA	81
2.10.2.	PLÁSTICO	84
2.10.3.	TABLA COMPARATIVA DE MATERIALES	86
CAPÍTULO 3. DESARROLLO DEL PROTOTIPO		87
3.1.	DESCRIPCIÓN GENERAL DEL PROYECTO	87
3.2.	SELECCIÓN DEL MATERIAL A UTILIZAR EN LA CARCASA DEL PROTOTIPO	88
3.3.	SELECCIÓN DE LA PLATAFORMA PARA LA APLICACIÓN	89
3.4.	SELECCIÓN DE LA PLATAFORMA ARDUINO	91
3.5.	DISEÑO DEL PRIMER PROTOTIPO	92
3.5.1.	DIAGRAMA DE BLOQUES	93
3.5.2.	DIAGRAMA DE FLUJO	95
3.5.3.	SIMULACIÓN	96

3.5.4.	DISEÑO E IMPRESIÓN DEL CIRCUITO PARA LA BAQUELITA...	97
3.5.5.	ELECCIÓN Y CONFIGURACIÓN DE LA TECNOLOGÍA INALÁMBRICA	100
3.5.6.	DISEÑO Y PROGRAMACIÓN DE LA APLICACIÓN.....	106
3.5.7.	IMPLEMENTACIÓN DEL PRIMER PROTOTIPO	114
3.6.	DISEÑO DE LAS SECUENCIAS DE PROGRAMACIÓN	118
3.6.1.	PRIMERA SECUENCIA	118
3.6.2.	SEGUNDA SECUENCIA	120
3.6.3.	TERCERA SECUENCIA	121
3.6.4.	CUARTA SECUENCIA	123
3.6.5.	QUINTA SECUENCIA.....	124
3.6.6.	SECUENCIA ADICIONAL	126
3.7.	CONSIDERACIONES DEL PRIMER PROTOTIPO	128
3.8.	DISEÑO FINAL	129
3.7.1	MEJORAS AL DISEÑO ANTERIOR.....	130
3.9.	CÁLCULO DE CORRIENTE, VOLTAJE Y POTENCIA	133
3.9.1.	CORRIENTE	133
3.9.2.	VOLTAJE	133

3.9.3. POTENCIA	133
3.10. PRUEBAS	134
3.10.1. PRUEBA DE ENCENDIDO	134
3.10.2. PRUEBA DE SECUENCIAS LÓGICAS DE PROGRAMACIÓN	135
3.10.3. PRUEBA CON ESCOLARES DE 10 A 12 AÑOS	152
3.10.4. VERIFICACIÓN DE APRENDIZAJE	155
3.11. MANUAL DE USUARIO PARA LA CORRECTA UTILIZACIÓN Y MEJOR APROVECHAMIENTO DEL PROTOTIPO ELECTRÓNICO Y SU APLICACIÓN	156
3.11.1. GUÍA DE USO	157
3.11.2. RECOMENDACIONES GENERALES PARA EL BUEN USO Y CONSERVACIÓN DEL PROTOTIPO ELECTRÓNICO	164
CAPÍTULO 4. ANÁLISIS DE COSTO BENEFICIO	165
4.1. PRESUPUESTO DEL HARDWARE EMPLEADO	165
4.2. PRESUPUESTO DEL SOFTWARE EMPLEADO	168
4.3. ANÁLISIS COSTO BENEFICIO	168
CAPÍTULO 5. CONCLUSIONES Y RECOMENDACIONES	171
CONCLUSIONES	171
RECOMENDACIONES	173
BIBLIOGRAFÍA	175
ANEXOS	179

ANEXO 01. ENCUESTAS	180
ANEXO 02. CÓDIGO DEL PROTOTIPO	186
ANEXO 03. SECUENCIAS CONVERTIDAS EN LÍNEAS DE CÓDIGO SIMPLE.....	195
ANEXO 04. RESPALDO FOTOGRÁFICO DE RESULTADOS	197
ANEXO 05. RESPALDO DE PRECIOS.....	199

ÍNDICE DE ILUSTRACIONES

Ilustración 1. Teoría Conductista.....	8
Ilustración 2. Teoría Cognitiva.....	12
Ilustración 3. Teoría Constructivista	15
Ilustración 4. Funcionamiento cognitivo en Operaciones Concretas	20
Ilustración 5. Sintaxis de la Estructura de control IF	27
Ilustración 6. Sintaxis de la Estructura de control FOR	28
Ilustración 7. Programación mediante Bloques.....	28
Ilustración 8. Logo representativo de Scratch	31
Ilustración 9. Code.org	33
Ilustración 10. MIT App Inventor	34
Ilustración 11. Blockly	36
Ilustración 12. iOS de Apple	38
Ilustración 13. Sistema Operativo Android	40
Ilustración 14. Tecnologías Inalámbricas.....	43
Ilustración 15. Logo representativo Wi-Fi	44
Ilustración 16. Logo representativo Bluetooth	45
Ilustración 17. Logo Representativo ZigBee.....	47
Ilustración 18. Topologías de ZigBee	48
Ilustración 19. Placa Arduino DUE.....	52
Ilustración 20. Placa Arduino Leonardo.....	54
Ilustración 21. Placa Arduino Mega 2560.....	56
Ilustración 22. Placa Arduino Mini PRO	58

Ilustración 23. Placa Arduino Nano	59
Ilustración 24. Placa Arduino Uno	61
Ilustración 25. Placa Arduino Yun	62
Ilustración 26. Motor DC con Caja Reductora	66
Ilustración 27. Micro motor Solarbotics 150:1	68
Ilustración 28. Sensor Ultrasónico	69
Ilustración 29. Ángulo de Medición	71
Ilustración 30. Onda de trabajo del Sensor	71
Ilustración 31. Módulo Bluetooth HC-05	72
Ilustración 32. Distribución de pines del Circuito Integrado L293D	74
Ilustración 33. Configuración puente en H para un motor de Corriente Continua	75
Ilustración 34. Batería recargable Ion Litio 3.7 v 800 mAh	77
Ilustración 35. Diodo Emisor de Luz	77
Ilustración 36. IDE de Arduino	79
Ilustración 37. Ícono CadSoft EAGLE	80
Ilustración 38. Logo Proteus	81
Ilustración 39. Partes de la madera	82
Ilustración 40. Plástico	85
Ilustración 41. a) Vista anterior Arduino Mini PRO b) Vista posterior Arduino Mini PRO	92
Ilustración 42. Diagrama de Bloques	93
Ilustración 43. Diagrama de Flujo	95
Ilustración 44. Simulación en Proteus, Primer prototipo	97
Ilustración 45. Impresión del Diseño de Circuito Impreso	98
Ilustración 46. Baquelita en Ácido Férrico	99

Ilustración 47. Baquelita terminada.....	99
Ilustración 48. Sincronización de bluetooth 1	102
Ilustración 49. Activación de bluetooth.....	103
Ilustración 50. Dispositivos disponibles.....	104
Ilustración 51. Ingreso de PIN.....	105
Ilustración 52. Dispositivo sincronizado	106
Ilustración 53. Plataforma de diseño de MIT App Inventor.....	107
Ilustración 54. Base de diseño de la Aplicación Móvil	108
Ilustración 55. Botones y elementos requeridos.....	109
Ilustración 56. Botón Btn_conectar.....	109
Ilustración 57. Botón Btn_desconectar.....	110
Ilustración 58. Botón Btn_play.....	110
Ilustración 59. Botón Btn_adelante	111
Ilustración 60. Botón Btn_graduación.....	111
Ilustración 61. Botones: a) 0°, b) 180°	112
Ilustración 62. Botón Btn_decisión	112
Ilustración 63. Botón Btn_repetición	113
Ilustración 64. Botones: a) Btn_X2, b) Btn_X3	113
Ilustración 65. a) Btn_Abrir, b) Btn_Cerrar	114
Ilustración 66. Motor y Llanta.....	114
Ilustración 67. Componentes fijados al Chasis.....	115
Ilustración 68. a) Elementos sobre la placa b) Soldadura.....	116
Ilustración 69. Módulos instalados.....	116
Ilustración 70. Primer prototipo.....	117

Ilustración 71. Primer prototipo Completo.....	117
Ilustración 72. Instrucciones de la Aplicación	118
Ilustración 73. Indicaciones Secuencia 1.....	119
Ilustración 74. Secuencia Avanzar.	119
Ilustración 75. Indicaciones secuencia 2	120
Ilustración 76. Secuencia Giro.....	121
Ilustración 77. Indicaciones secuencia 3	122
Ilustración 78. Secuencia Repetición.....	122
Ilustración 79. Instrucciones Secuencia 4	123
Ilustración 80. Secuencia mixta Giro - Repetición.....	124
Ilustración 81. Instrucciones secuencia 5	125
Ilustración 82. Secuencia Decisión.....	125
Ilustración 83. Instrucciones secuencia Adicional	127
Ilustración 84. Secuencia Combinada	127
Ilustración 85. Primer Prototipo Implementado	128
Ilustración 86. Nuevos elementos considerados.....	130
Ilustración 87. Chasis, Llantas y micro motores	131
Ilustración 88. Placa asegurada al nuevo chasis	131
Ilustración 89. Diseño del segundo prototipo.....	132
Ilustración 90. Prueba de encendido del prototipo	135
Ilustración 91. 1a. Partida de la secuencia.....	136
Ilustración 92. 1b. Desarrollo de la secuencia.....	136
Ilustración 93. 1c. fin de la secuencia.....	136
Ilustración 94. 1d. Instrucciones seleccionadas.....	137

Ilustración 95. 2a. Partida de la secuencia.....	137
Ilustración 96. 2b. Desarrollo de la secuencia.....	138
Ilustración 97. 2c. giro del prototipo electrónico.....	138
Ilustración 98. 2d. Fin de la secuencia.....	138
Ilustración 99. 2e. Instrucciones seleccionadas.....	139
Ilustración 100. 3a. Partida de la secuencia.....	140
Ilustración 101. 3b. desarrollo de la secuencia.....	140
Ilustración 102. 3c. Fin de la secuencia.....	140
Ilustración 103. 3d. Instrucciones Seleccionadas.....	140
Ilustración 104. 4a. Partida de la secuencia.....	141
Ilustración 105. 4b. Giro hacia la derecha.....	141
Ilustración 106. 4c. Desarrollo de la secuencia.....	142
Ilustración 107. 4d. Desarrollo de la secuencia.....	142
Ilustración 108. 4e. Fin de la secuencia.....	142
Ilustración 109. 4f. Instrucciones Seleccionadas.....	143
Ilustración 110. 5a. Partida de la secuencia.....	144
Ilustración 111. 5b. Desarrollo de la secuencia.....	144
Ilustración 112. 5c. Giro a la izquierda luego de un obstáculo.....	145
Ilustración 113. 5d. Desarrollo de la secuencia.....	145
Ilustración 114. 5e. Desarrollo de la secuencia.....	146
Ilustración 115. 5f. Giro a la derecha luego de un obstáculo.....	146
Ilustración 116. 5g. Fin de la secuencia.....	147
Ilustración 117. 5h. Instrucciones Seleccionadas.....	147
Ilustración 118. 6a. Partida de la Secuencia.....	148

Ilustración 119. 6b. Giro hacia la izquierda.....	149
Ilustración 120. 6c. Giro a la derecha.....	149
Ilustración 121. 6d. Giro a la izquierda tercera repetición	150
Ilustración 122. 6e. Giro a la derecha tercera repetición	150
Ilustración 123. 6f. Fin de la Secuencia.....	151
Ilustración 124. 6g. Instrucciones Seleccionadas	151
Ilustración 125. Pruebas con niños y niñas escolares de 10 a 12 años	152
Ilustración 126. Portada del Manual.....	156
Ilustración 127. Encendido del Prototipo	157
Ilustración 128. Aplicación dentro del menú.....	158
Ilustración 129. Pantalla Principal de la Aplicación.....	159
Ilustración 130. Uso de Instrucción Adelante	160
Ilustración 131. Uso de la Instrucción Graduación	161
Ilustración 132. Uso de la Instrucción Repetición.....	162
Ilustración 133. Uso de la Instrucción Decisión.....	162
Ilustración 134. Íconos de conjunto.....	163
Ilustración 135. Ejemplo de una Secuencia.....	164

ÍNDICE DE TABLAS

Tabla 1. Comparativa entre Teorías de Aprendizaje	18
Tabla 2. Etapas del desarrollo cognitivo	19
Tabla 3. Tabulación de las encuestas.....	22
Tabla 4. Información general de iOS.....	39
Tabla 5. Información general de Android	41
Tabla 6. Comparativa entre tecnologías inalámbricas.....	49
Tabla 7. Especificaciones Técnicas del Microcontrolador AVR	53
Tabla 8. Especificaciones Técnicas del Arduino Leonardo.	55
Tabla 9. Especificaciones Técnicas del Arduino MEGA 2560.....	57
Tabla 10. Especificaciones Técnicas del Arduino Mini	58
Tabla 11. Especificaciones técnicas del Arduino Nano	60
Tabla 12. Especificaciones Técnicas de la Placa Arduino Uno	61
Tabla 13. Especificaciones Técnicas del Arduino Yun.....	63
Tabla 14. Comparativa de Placas Arduino	65
Tabla 15. Valores de rendimiento del micro motor.....	69
Tabla 16. Consumo de corriente de los dispositivos electrónicos dentro del prototipo	75
Tabla 17. Comparativa de Materiales.....	86
Tabla 18. Comparativa entre prototipos	132
Tabla 19. Resultados de las pruebas con niños de 10 a 12 años	153
Tabla 20. Tiempos de Perfeccionamiento del Prototipo	154
Tabla 21. Presupuesto del Prototipo.....	165
Tabla 22. Presupuesto de elementos electrónicos usados.....	166

Tabla 23. Presupuesto del Smartphone.....	167
Tabla 24. Presupuesto de Software.....	168
Tabla 25. Costos totales del proyecto.....	169

ÍNDICE DE ECUACIONES

(1)	70
(2)	76
(3)	76
(4)	134

RESUMEN

El presente proyecto consiste en la implementación de un prototipo electrónico programado mediante bloques que tiene como objeto la enseñanza de bases de programación hacia los niños y niñas escolares comprendidas entre las edades de 10 a 12 años; el mismo que permitirá involucrar a los escolares en la tecnología a través del uso de las herramientas tecnológicas que tienen a disposición conjuntamente con el prototipo implementado propiamente para este propósito y a la vez que se fundamentan las bases de programación orientándose para su vida académica y tecnológica.

El Prototipo Electrónico implementado para este trabajo de titulación pretende dar a conocer a los niños los elementos básicos de la lógica de programación de manera sencilla, apoyarlos en su aprendizaje de manera interactiva usando un Smartphone y el prototipo, estos les dará una experiencia nueva en su proceso de razonamiento puesto que descubren y desarrollan nuevas formas de crear, asimilar un conocimiento y resolver problemas por su propia cuenta. El niño no está sujeto a una forma de pensar y razonar sino que está en libertad de resolver sus problemas de varias formas.

Este trabajo se desarrolló basado en plataformas Open Source de hardware y Software que integra dos partes, la parte de la Aplicación para la programación en bloques soportada por App Inventor y la parte de ejecución de la programación soportada por la plataforma Arduino; estas dos plataformas se comunican de manera inalámbrica a través de la tecnología Bluetooth que se encuentra integrada en casi la totalidad de los Smartphone por lo que es fácil acceder a uno y aplicar la programación desde cualquier dispositivo.

ABSTRACT

The current project involves the implementation of an electronic prototype programmed through blocks that has as purpose teaching programming basis to children and understood schoolgirls aged 10 to 12 years; the same which will involve schoolchildren in technology through the use of technological tools at their disposal together with the prototype implemented properly for this purpose and while programming basis oriented for academic and technological life are based.

Electronic Prototype implemented for this work degree aims to show children the basics of programming logic simply, support them in their learning different interactive when using the Smartphone and the prototype way and give them an experience new in its reasoning process since discovered and developed a way to create knowledge and solve problems on their own. The child is not subject to a way of thinking and reasoning but is free to solve their problems in several ways.

This work was developed based on Open Source platforms hardware and software that integrates two parts, the part of the Application programming supported by App Inventor blocks and the execution part of the program supported by the Arduino platform; these two platforms communicate wirelessly via Bluetooth technology that is integrated in almost all smartphones so it is easy to access and apply one programming from any device.

PRESENTACIÓN

En este mundo cambiante, en una época donde predomina el avance tecnológico y la presentación a diario de nueva tecnología tanto en hardware como en software, las aplicaciones y sistemas electrónicos cada vez más avanzados hacen que nuestro país una vez más se vea relegado por las potencias en la creación y desarrollo de tecnología; los estudiantes primarios y secundarios deben adaptarse a ese cambio acelerado, siendo conscientes que son los llamados a cambiar el panorama actual en nuestro medio para fomentar el desarrollo tecnológico.

El fomento de plataformas libres no licenciadas les da la pauta para involucrarse en el tema tecnológico e informático haciendo de lado el tema económico para generar conocimiento y que mejore la capacidad de rendimiento académico que existe en nuestros jóvenes para llevar al país al desarrollo y generación de tecnología. Este prototipo electrónico presentado en este proyecto se implementó con el objetivo de cambiar la situación actual de aprendizaje en estos campos, siendo una herramienta didáctica que les permitirá conocer y explorar el mundo de la tecnología desde el punto de vista del desarrollador.

En este proyecto encontrará una revisión de las teorías de aprendizaje usadas en el ámbito educativo, conceptos básicos de programación orientada a las edades propuestas, herramientas de hardware y software como de tecnologías de comunicación para la implementación de este prototipo. Además un análisis costo beneficio acerca de la viabilidad del prototipo y las conclusiones y recomendaciones generadas a los largo del desarrollo del prototipo.

CAPÍTULO I. ANTECEDENTES

Antes de realizar el proyecto de titulación es necesario presentar los puntos que se consideran en el presente capítulo, como son: tema, problema, objetivos, alcance y justificación; con el fin de justificar la importancia del proyecto al identificar la necesidad que se resuelve con la implementación de un prototipo electrónico programado mediante bloques para la enseñanza de programación a niños de 10 a 12 años de Ibarra.

1.1. TEMA

PROTOTIPO ELECTRÓNICO PROGRAMADO MEDIANTE BLOQUES CON UNA APLICACIÓN MÓVIL PARA LA ENSEÑANZA DE PROGRAMACIÓN A NIÑOS DE 10 A 12 AÑOS EN LA CIUDAD DE IBARRA.

1.2. PROBLEMA

Las escuelas fiscales de la ciudad de Ibarra no han puesto atención en desarrollar las capacidades y destrezas propias de los niños ya sean motrices, sociales, menos aún incentivar tendencias de innovación y desarrollo tecnológico, especialmente en tempranas edades de entre los 10 y los 12 años edad que es donde los niños pueden desarrollar y comprender de forma natural y rápida estas capacidades, para esto se requiere metodologías y elementos adecuados enfocados en la educación propios para el desarrollo intelectual en el área tecnológica.

Actualmente las escuelas fiscales de Ibarra, sus autoridades y directamente los docentes no usan herramientas didácticas enfocadas en software que combinen el aprendizaje

con la recreación y simultáneamente permita a los escolares robustecer e incrementar el intelecto investigativo desde sus inicios, por lo que desfavorablemente siguen usando herramientas y materiales que no se encaminan en las directrices nuevas de la educación en el Ecuador y principalmente en Ibarra.

En consecuencia, la falta de elementos y herramientas didácticas de software que estimulen en los escolares la innovación y el desarrollo tecnológico aprovechando su etapa de curiosidad cognitiva y su interés apasionado por percibir conocimiento, reduce ampliamente la posibilidad que cuando estén en su juventud se interesen por el desarrollo de tecnología, de aplicaciones a través de fundamentos básicos de programación que le permitan desarrollar productos para el mejoramiento y desarrollo del entorno social en el que se desenvuelve.

1.3. OBJETIVOS

1.3.1. OBJETIVO GENERAL

Desarrollar un prototipo electrónico a través de plataformas libres para la enseñanza básica de programación a escolares de 10 a 12 años de edad que ayude al niño en el aprendizaje de la base de la programación y razonamiento lógico.

1.3.2. OBJETIVOS ESPECIFICOS

- ✓ Investigar acerca de la incidencia de la lógica de programación sobre el aprendizaje en niños de 10 a 12 años de edad para determinar la posibilidad del diseño de secuencias básicas de programación para mencionadas edades.

- ✓ Diseñar secuencias lógicas de programación por bloques en una aplicación móvil que le permita al niño utilizar su razonamiento para aplicarlo al prototipo electrónico por medio de la comunicación con la aplicación.

- ✓ Implementar el prototipo electrónico con sus pruebas respectivas de funcionamiento usando una selección de niños con edades comprendidas entre 10 y 12 años y documentar los resultados obtenidos.

- ✓ Realizar un análisis del sistema que garantice su factibilidad, un análisis costo beneficio y un manual de usuario como complemento al proyecto a desarrollarse.

1.4. ALCANCE

Dentro de los límites del trabajo de grado se llevará a cabo el desarrollo de una aplicación móvil destinada a niños de 10 a 12 años que les permita impulsar el desarrollo tecnológico y la programación mediante el juego a la vez que fortalecen su educación. La aplicación se fundamenta en programación por bloques la cual permitirá controlar un prototipo electrónico de forma inalámbrica, para su implementación y aceptación se respaldará en un estudio de teorías de aprendizaje respecto a los niños, con encuestas a maestros que se incluirán en la parte de Anexos, en las rutinas básicas que se presente de acuerdo al estudio y en plataformas libres tanto de software como de hardware para su soporte.

El proyecto de tesis empezará con un estudio conjunto entre las teorías de aprendizaje, herramientas didácticas y encuestas a maestros orientado a los niños con edades comprendidas entre los 10 y los 12 años que atraviesan una etapa escolar, acerca de sus habilidades y destrezas naturales, capacidades que han podido desarrollar en el transcurso de su vida escolar, y cómo fortalecerlas para su correcta formación escolar y orientarlas tanto al desarrollo tecnológico como a la programación usando nuevas herramientas didácticas desarrolladas para este efecto.

Tomando como base los resultados arrojados por el estudio y las capacidades propias de estas edades se procederá a diseñar una aplicación móvil de programación por bloques a través de una plataforma libre que contenga cinco (5) rutinas simples predefinidas que contemplan las siguientes ordenes: avanzar, retroceder, giro a la izquierda, giro a la derecha, giro con graduación ya sea a la izquierda o a la derecha, tomar una decisión (si retroceder, si girar a la izquierda noventa grados, si girar a la derecha 90 grados o girar con graduación a la izquierda o la derecha), que serán aplicadas en el prototipo electrónico y que combinadas reflejarán la forma en como el niño puede resolver un problema de lógica de programación usando bloques.

Seguido del estudio y del diseño de la aplicación móvil de programación por bloques se procederá con una comparación de las tecnologías inalámbricas fundamentado en las normas establecidas para este caso y elegir la mejor para la comunicación entre el sistema. Y finalmente se implementará un Prototipo Electrónico basado en plataformas libres de hardware con el tamaño acorde a los niños, con el cual se procederá a su validación a través de pruebas de campo con una muestra de cinco (5) niños de una escuela fiscal de la ciudad de

Ibarra que permitirá evaluar el proyecto, documentar los resultados obtenidos, además como complemento se realizará un manual de usuario para su presentación final, un análisis costo beneficio, conclusiones y recomendaciones.

1.5. JUSTIFICACIÓN

En Ecuador las nuevas propuestas de aprendizaje dadas por el Gobierno Nacional como parte de la revolución educativa están dirigidas en proporcionar a los estudiantes habilidades que promuevan el desarrollo e investigación desde edades escolares, por tal razón y cumpliendo con la misión de la Universidad Técnica del Norte, que fomenta e impulsa el desarrollo tecnológico se ha planteado este proyecto de innovación tecnológica, desarrollando e implementando un prototipo electrónico programado mediante bloques que está orientado hacia el ámbito de la enseñanza de lógica de programación a niños escolares, que conjuntamente con el diseño de aplicaciones son las bases para el progreso tecnológico en la ciudad y el país en esta época. (PNBV, 2009-2013)

Según Steve Jobs cofundador y ex presidente ejecutivo de Apple Inc. *“Everybody in this country should learn how to program a computer... because it teaches you how to think.”* Con el desarrollo de este proyecto se busca impulsar la enseñanza de la lógica de programación en los niños en edades comprendidas entre 10 y 12 años, que son las edades en la que los niños poseen la gran capacidad de captar conocimientos de forma rápida con una gran habilidad de resolución de problemas. Según Code.org que es una fundación sin fin de lucro dedicada a la difusión de la enseñanza de programación de ordenadores, con el objetivo de concienciar a la sociedad de que hay escasez mundial de programadores informáticos, se

contribuye con el proyecto de manera directa a desarrollar las capacidades de resolución de problemas en los niños y mejorar su pensamiento lógico. (ATAREAO, 2013),

La iniciativa de este proyecto persigue aportar a la revolución educativa contemplada en el Plan Nacional del Buen Vivir que busca disminuir la brecha tecnológica desarrollando un prototipo electrónico fundamentado en hardware y software libre para imponer una enseñanza diferente con metodologías nuevas que vayan acorde al desarrollo de la tecnología e involucrando a los niños de la ciudad de Ibarra y del país como futuros profesionales en el área tecnológica que se persigue juntamente con el cambio de la matriz productiva. (Desarrollo, 2013)

En lo personal la motivación para desarrollar este proyecto de tesis es aportar al desarrollo tecnológico del Ecuador enfocándome en un grupo considerado como el futuro del país que son los niños a través del mejoramiento de sus prácticas de estudio y aprendizaje tomando en cuenta las épocas anteriores donde la educación escolar no usaba materiales didácticos para impulsar el aprendizaje tecnológico e innovador para contribuir a mejorar los niveles de innovación y desarrollo tecnológico para que el Ecuador avance hacia el primer mundo en lo referente a ciencia y tecnología. (Rodríguez, 2012)

CAPÍTULO 2. FUNDAMENTACIÓN TEÓRICA

El capítulo dos empieza citando aspectos relevantes en el campo teórico que fundamentarán y respaldarán el desarrollo de este proyecto, se inicia con las diferentes teorías de aprendizaje donde se detallan las más importantes, seguido se detallan las etapas del desarrollo cognitivo que es la base para el desarrollo e implementación de este proyecto de tesis, luego se da una introducción de la programación mediante bloques que es donde se detalla en que consiste y porque se toma en cuenta esta programación, además se citan los diferentes sistemas operativos que soportan la aplicación móvil, las tecnologías inalámbricas que hacen posible la comunicación entre la aplicación móvil y el prototipo y finaliza con una descripción de los componentes electrónicos a usar en este proyecto de tesis conjuntamente con los materiales que se usan en el diseño y construcción de la carcasa del prototipo.

2.1. TEORÍAS DE APRENDIZAJE

Las teorías de aprendizaje tratan de describir algunos procesos por los cuales tanto los seres humanos como los animales aprenden, desde que se implantaron estas teorías numerosos profesionales en el tema como psicólogos y pedagogos han aportado diversas teorías respecto a este tema.

La variedad de teorías intentan ayudar a comprender, predecir y controlar el comportamiento humano, derivando en la elaboración de estrategias de aprendizaje e intentando explicar cómo los humanos acceden al conocimiento. Estas a su vez se enfocan en la adquisición de destrezas y habilidades en el razonamiento y en la adquisición de conceptos claros y precisos.

Las teorías de aprendizaje contemplan un diverso conjunto de marcos teóricos o conceptos que por lo general comparten aspectos, como también establecen postulados totalmente contradictorios. A continuación se mencionan y describen tres teorías relevantes:

2.1.1. TEORÍA CONDUCTISTA

La teoría conductista o el conductismo como se lo suele llamar aparece como una teoría psicológica y posteriormente se aplicó en la educación formal. Esta es una teoría pionera que vino a influir con gran peso la forma como se comprende el aprendizaje humano. Anterior a la aparición del conductismo el aprendizaje era admitido como un proceso interno y se lo investigaba por medio del método llamado “introspección” en el cual se les pedía a las personas que relataran y describieran que es lo que estaban pensando, ver *Ilustración 1*.



Ilustración 1. Teoría Conductista

Fuente: Recuperado de <http://teoriasimportantes.blogspot.com/2013/03/teoria-conductista.html>

A partir de este método nace el conductismo como un rechazo al “método de introspección” y con una propuesta más orientada a un enfoque externo, en la que las

mediciones y los resultados se realizan a través de fenómenos observables. El conductismo data sus inicios a las tempranas décadas del siglo XX y cuyo fundador y defensor fue J.B. Watson. De acuerdo con Watson “para que la psicología lograra un estatus verdaderamente científico, tenía que olvidarse del estudio de la conciencia y los procesos mentales (procesos inobservables) y, en consecuencia, nombrar a la conducta (los procesos observables) su objeto de estudio". (Hernández, 2002)

En los amaneceres del siglo XX el conductismo watsoniano tuvo una gran acogida entre los estudiosos de este tema y ligeramente se asoció a otras escuelas con nociones similares, tal fue el caso de B.F. Skinner y su conductismo operante, que sus ideas asociadas con las ideas de Watson llegaron a plasmarse en la principal corriente del conductismo.

Desde la perspectiva conductista el aprendizaje es definido como un cambio observable y medible en el comportamiento, a diferencia de los procesos internos como procesos mentales que son considerados irrelevantes para el estudio del aprendizaje humano debido a que no pueden ser medidos ni observados de forma directa.

2.1.1.1. IDEAS IMPORTANTES

El estudio del aprendizaje se ha enfocado en fenómenos observables y medibles que se fundamentan en el aprendizaje producto de una relación estímulo – respuesta, por ende los procesos o fenómenos internos como el pensamiento y la motivación no pueden ser tomados en cuenta en la investigación científica del aprendizaje.

El aprendizaje exclusivamente ocurre cuando se identifica un fenómeno o cambio en el comportamiento. Si no hay cambio o fenómeno observable y medible no hay aprendizaje.

El conductismo tuvo significativas aportaciones científicas acerca del comportamiento humano, en sus esfuerzos por resolver problemas concernientes con la conducta humana y el modelamiento de conductas que no pueden enmendarse completamente a base de premio – castigo, llega a descubrir que el uso de refuerzos pueden fortalecer conductas apropiadas y su desuso disminuir las no deseadas, además la asignación de calificaciones, recompensa y castigo son aportaciones importantes en esta teoría.

Las nociones de las ideas conductistas pueden emplearse con éxito en la adquisición de conocimientos memorísticos, que admiten niveles primarios o básicos de comprensión, tal es el caso del aprendizaje de capitales del mundo o las tablas de multiplicar. No obstante esto presenta una limitación significativa: que la repetición no garantiza asimilación de la nueva conducta, sino solo su ejecución (sabe multiplicar pero no sabe cuando debe hacerlo, se sabe las tablas de multiplicar pero no sabe resolver un problema en el que tiene que usar multiplicación), esto muestra que la situación aprendida no es transmisible a otra o a otras situaciones. (UOCTIC, 2005)

2.1.1.2. CONCEPCIÓN DEL ALUMNO

El alumno figura como un sujeto cuyo desempeño y aprendizaje escolar pueden ser arreglados o configurados desde el exterior, la instrucción formal, los métodos, los contenidos,

donde solo se necesita programar adecuadamente el material educativo para que se logre el aprendizaje de conductas escolares deseables.

2.1.1.3. CONCEPCIÓN DEL MAESTRO

La labor del maestro consiste en fomentar una apropiada serie de arreglos de contingencia de reforzamiento y revisión de estímulos para enseñar.

El conductismo es una de las teorías que se ha mantenido por más tiempo y con una mayor consolidación, a pesar de que no encaja totalmente en las nuevas teorías educativas y ha sido continuamente puesto a revisión, debido a que percibe al aprendizaje como algo mecánico, deshumano y reduccionista, pero a pesar de ello tiene vigencia en nuestra cultura y abarca muchas prácticas que aún se usan en el sistema escolar vigente.

2.1.2. TEORÍA COGNITIVA

Los estudios enfocados en la teoría cognitiva desde sus inicios se presentan como la teoría que ha de relevar a las perspectivas conductistas que habían regido hasta ese entonces la psicología, ver *Ilustración 2*, todas las ideas tuvieron aportes de diferentes investigadores y teóricos que influyeron positivamente para que esta teoría se conforme, tales como: Jean Piaget y la psicología genética, David Ausubel y el aprendizaje significativo, Jerome Bruner y el aprendizaje por descubrimiento, por citar a los más relevantes. (Sincero, 2011)



Ilustración 2. Teoría Cognitiva

Fuente: Recuperado de <http://psicopedagogia19.blogspot.com/2011/11/teoria-cognoscitiva.html>

Las ideas de los autores mencionados en el párrafo anterior tienen en común el haberse enfocado en varias de las dimensiones de lo cognitivo tales como la atención, percepción, memoria, inteligencia, lenguaje, pensamiento y que además recalcan que existen diferencias significativas entre estas.

Desde la década de los cincuenta hasta la década de los ochenta, se llevaron a cabo muchas líneas de investigación sobre las bases de la teoría cognitiva y modelos teóricos sobre las diferentes facetas de la cognición. Por ende se llega a afirmar, que en la actualidad ya no es una teoría con una aproximación monolítica, sino que existen varias corrientes desarrolladas dentro de este enfoque, por dar unos ejemplos: el constructivismo, la propuesta para teoría socio cultural entre otras.

Actualmente resulta complejo distinguir claramente debido a múltiples influencias de otras disciplinas, donde termina la teoría cognitiva y donde empieza otra teoría, debido a que pueden encontrarse líneas e investigadores con ideas de distinto orden teórico, metodológico que integran ideas de varias tradiciones e incluso ideas de teorías alternativas.

2.1.2.1. IDEAS IMPORTANTES

La teoría cognitiva proporciona significativas aportaciones al estudio de los procesos de enseñanza y aprendizaje, tales como la contribución al conocimiento preciso de algunas capacidades esenciales para el aprendizaje, como la atención, la memoria y el razonamiento.

Identifica una nueva visión del ser humano considerándolo como un ente que realiza una actividad guiada fundamentalmente en el procesamiento de la información, muy distante a la visión simplista que se conocía y que había defendido y divulgado el conductismo.

Reconoce la importancia de como este ente organiza, filtra, codifica, categoriza y evalúa la información y la manera en que estas herramientas, estructuras o esquemas mentales son empleadas para interpretar la realidad.

Considera que cada ente o individuo tiene diferentes representaciones del mundo, las que dependerán de sus propios esquemas y de su interacción con su entorno, e irán cambiando y serán cada vez más sofisticadas.

La teoría cognitiva establece que: aprender constituye un resumen de la forma y contenido recibido por las percepciones, las cuales actúan en forma relativa y particular en cada individuo, y que a su vez se encuentran influenciadas por sus antecedentes, actitudes y motivaciones individuales.

Basados en los estudios de esta teoría los psicólogos educativos se han interesado en resaltar dos cuestiones centrales, que son las que dicen que la educación debería orientarse al logro del aprendizaje significativo con sentido y al desarrollo de habilidades estratégicas generales y específicas del aprendizaje. (Bolaños, 2003)

2.1.2.2. *CONCEPCIÓN DEL ALUMNO*

El alumno es un sujeto activo procesador de información, que posee competencia cognitiva para aprender y solucionar problemas; por ende esta competencia debe ser considerada y desarrollada utilizando nuevos aprendizajes y habilidades estratégicas.

2.1.2.3. *CONCEPCIÓN DEL MAESTRO*

Este parte de la idea de que un alumno activo que aprende significativamente puede aprender a aprender y a pensar, el maestro se centra detalladamente en la elaboración y la organización de experiencias didácticas para lograr los cometidos y no debe desempeñar el papel principal en el daño de la participación cognitiva de los alumnos.

2.1.3. TEORÍA CONSTRUCTIVISTA

Esta teoría de constructivismo es compartida por diferentes tendencias de la investigación psicológica y educativa. Entre estas se puede encontrar las teorías de Jean Piaget (1952), David Ausubel (1963) y Jerome Bruner (1960), a pesar de que ninguno de estos teoristas se denominó como constructivista sus ideas y propuestas ciertamente ilustran las ideas en este sentido.

El constructivismo es principalmente una epistemología, es decir, una teoría que intenta explicar cuál es la naturaleza del conocimiento humano. El constructivismo aclara que nada viene de nada, que el conocimiento previo da nacimiento a conocimiento nuevo, ver

Ilustración 3.



Ilustración 3. Teoría Constructivista

Fuente: Recuperado de <http://uoc1112-2->

[grupo1.wikispaces.com/3.+TEOR%C3%8DA+CONSTRUTIVISTA](http://uoc1112-2-grupo1.wikispaces.com/3.+TEOR%C3%8DA+CONSTRUTIVISTA)

2.1.3.1. IDEAS IMPORTANTES

El constructivismo mantiene que el aprendizaje es básicamente activo, una persona que aprende algo nuevo, lo incorpora a sus experiencias previas y a sus propias estructuras mentales. Cada información nueva es asimilada y depositada en una red de conocimientos y experiencias que existen previamente en el individuo. Como resultado concluye que el aprendizaje no es pasivo ni objetivo, por el contrario es un proceso subjetivo que cada persona va modificando continuamente a la luz de sus experiencias. (Abbott, 1999)

El aprendizaje no es un simple ejercicio de transmisión y acumulación de conocimientos, sino más bien un “proceso activo” por parte del alumno que ensambla, extiende, restaura e interpreta y por ende genera o construye conocimientos partiendo de sus experiencia e integrándolas con la información que recibe.

La teoría constructivista pretende ayudar a los estudiantes a internalizar, reacomodar y transformar la información nueva, esta transformación ocurre a través de la creación de nuevos aprendizajes y esto resulta del surgimiento de nuevas estructuras cognitivas, que permiten enfrentarse a situaciones iguales o parecidas en la realidad. (Brooks, 1999)

El constructivismo distingue el aprendizaje como actividad personal enmarcada en contextos funcionales, significativos y auténticos.

2.1.3.2. CONCEPCIÓN DEL ALUMNO Y EL MAESTRO

En este proceso de aprendizaje en particular como lo es el constructivo, el profesor cede su protagonismo al alumno quien es el que asume el papel principal en su propio proceso de formación y educación.

Es el alumno quien se convierte en el responsable de su propio aprendizaje, a través de la participación y la colaboración con sus compañeros. Para lo cual habrá de automatizar nuevas, novedosas y útiles estructuras intelectuales que le llevarán a desempeñarse con capacidad tanto en su entorno social próximo como en su futuro profesional.

En esta teoría constructivista el propio alumno es quien logrará la transferencia de la parte teórica hacia áreas prácticas, ubicados en contextos reales. Prácticamente la posición del maestro se debilita debido a la independencia del individuo, en este caso el alumno en aprender y desarrollar y adquirir propio y nuevo conocimiento.

El alumno tiene un nuevo papel, un rol indispensable para su propia formación, un protagonismo que es imposible ceder y que le proporcionará un sin número de herramientas significativas que habrán de ponerse a prueba en el pasar de su propio futuro. Dentro del constructivismo podemos destacar a dos de los autores más importantes que han aportado con sus ideas: Jean Piaget con el constructivismo psicológico y Lev Vigotsky con el constructivismo social. (Andujar, 2013)

2.1.4. TABLA COMPARATIVA ENTRE TEORÍAS DE APRENDIZAJE

Los autores que se han citado resumen cada una de estas teorías y las implicaciones que tiene cada una de ellas al momento de comprender la naturaleza del aprendizaje. En la *Tabla 1* se realiza una comparación de estas teorías.

Tabla 1. Comparativa entre Teorías de Aprendizaje

Aprendizaje como	Enseñanza	Foco Instruccional	Resultados
Adquisición de respuestas	Suministro de retroalimentación	Centrado en el currículo (Conductas correctas)	Cuantitativos (Fuerza de las Asociaciones)
Adquisición de conocimiento	Transmisión de Información	Centrado en el currículo (Información Apropiaada)	Cuantitativos (Cantidad de información)
Construcción de Significado	Orientación del proceso cognitivo	Centrado en el alumno (Procesamiento significativo)	Cualitativos (Estructura del conocimiento)

Fuente: Adaptado de [http://eprints.rclis.org/17463/1/bases teoricas.pdf](http://eprints.rclis.org/17463/1/bases_teoricas.pdf)

Donde la primera corresponde a la teoría conductista y las dos segundas a la teoría cognitiva. La segunda se puede considerar una etapa de transición que ha estado enraizada en los lineamientos educativos y que con más seguidores cuenta en la práctica.

2.2. ETAPAS DEL DESARROLLO COGNITIVO SEGÚN JEAN

PIAGET

Dentro de las etapas del desarrollo cognitivo planteadas, estudiadas y presentadas por Jean Piaget, uno de los principales psicólogos que se dedicó al estudio del desarrollo cognitivo

de los niños desde el nacimiento hasta su juventud, planteó una composición de cuatro etapas: Sensorio motora, Pre operacional, Operaciones Concretas y Operaciones Formales. En la *Tabla 2* se cita cada una de las etapas con sus edades y características.

A breve descripción la primera etapa o Sensorio motora comprende desde el nacimiento hasta los dos años de edad, y como principal característica indica que los lactantes aprenden por medio de sus sentidos y actividad motora, la segunda etapa o Pre operacional comprende desde los dos años hasta los 7 años de edad, como principal característica indica que los niños se sofistican más en su uso de pensamiento simbólico, pero sin poder utilizar aun la lógica, la tercera etapa u Operaciones Concretas comprende desde los siete años hasta los doce años de edad, como principal característica los niños desarrollan el pensamiento lógico, pero no abstracto y por último la etapa de Operaciones Formales que comprende a partir de los doce años hasta aproximadamente los 19 años de edad, su principal característica indica que los adolescentes adquieren capacidad para pensar de manera abstracta. (Maldonado, 2013)

Tabla 2. Etapas del desarrollo cognitivo

Etapa	Edad	Características
Sensorio motora	Nacimiento - 2 años	Los lactantes aprenden por medio de sus sentidos y actividad motora
Pre operacional	2 - 7 años	Los niños se sofistican más en su uso de pensamiento simbólico, pero sin poder utilizar aun la lógica
Operaciones Concretas	7 - 12 años	los niños desarrollan el pensamiento lógico, pero no el abstracto
Operaciones Formales	12 años en adelante	Los adolescentes adquieren capacidad para pensar de manera abstracta

Fuente: Adaptado de <http://es.slideshare.net/kiiiikkaa/operaciones-concretas-piaget-28761707>

2.2.1. ETAPA DE OPERACIONES CONCRETAS 10 – 12 AÑOS

Después de revisar a breve rasgo las diferentes etapas del desarrollo cognitivo en los niños, se toma la tercera etapa que comprende las Operaciones Concretas de siete a doce años que es la que corresponde para fines de este trabajo de titulación con el fin de dar una descripción más profunda y detallada del proceso cognitivo en niños de estas edades.

El desarrollo de las operaciones concretas dura desde los siete hasta los doce años, durante este periodo el niño desarrolla esquemas cognitivos coherentes que, en un inicio son secuencias de acciones. Uno de los aspectos más importantes del pensar operativo es que es reversible y que el niño, al ser más metódico o sistemático, es menos propenso a ser adherido por el error. En esta etapa Jean Piaget muestra al funcionamiento cognitivo en función de estructura lógico – matemática. (Gonzalez, 2011)

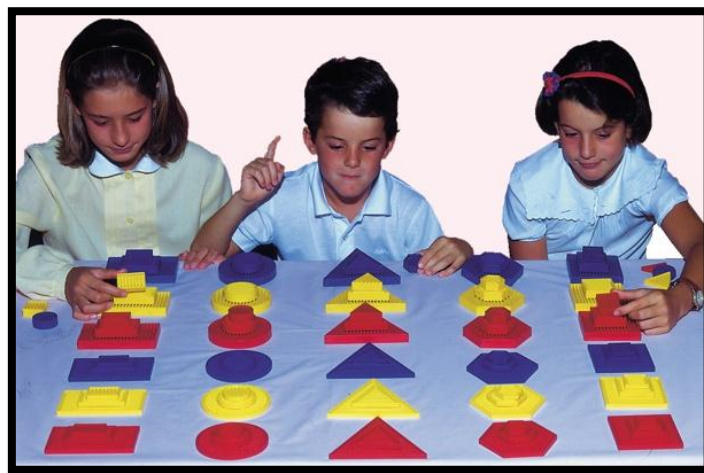


Ilustración 4. Funcionamiento cognitivo en Operaciones Concretas

Fuente: Recuperado de <http://psicopedagogia19.blogspot.com/2011/11/teoria-cognoscitiva.html>

La etapa de Operaciones Concretas se identifica por la utilización de varias comparaciones lógicas para responder a un estímulo y no reciben influencia por la apariencia, ver *Ilustración 4*. Según Piaget es el mejoramiento de la capacidad para pensar de manera lógica en un ambiente de cambio constante.

En esta etapa el escolar maneja claramente el pensamiento lógico, puede adaptarse ambientes que le proporcionen retos de esta naturaleza, estará en la capacidad de asimilar y resolver problemas o retos que se le presente ayudado no solo de su experiencia previa sino que además el escolar construye secuencias lógicas que le permitirán desenvolverse en cualquier dificultad propia a su edad.

Esta etapa facilita para que el maestro que disponga de las herramientas de aprendizaje y herramientas didácticas acordes a la edad y acordes al reforzamiento lógico le permita al alumno solidificar su pensamiento, su lógica, mejorar sus conocimientos e incluso fomentaría la atención que le preste a su entorno y se enfoque de mejor forma en su aprendizaje formal que le permitirá adquirir conocimiento formal.

2.3. ENCUESTAS

Para complementar el respaldo del prototipo inalámbrico se ha levantado información a través de encuestas a docentes de Sexto, Séptimo y Octavo de educación básica

preguntándoles sobre aspectos relacionados con la enseñanza de programación en sus grados, sus respuestas se encuentran respaldadas en el ANEXO 01 y la tabulación de sus respuestas se especifica en la *Tabla 3*.

Tabla 3. Tabulación de las encuestas

Preguntas	Sexto	Séptimo	Octavo
¿Con cuántos niños trabaja en su grado?	33	32	29
¿De qué edad en promedio son los niños de su grado?	10	11	12
En su escuela ¿se les imparte la materia de Computación o materias relacionadas con Programación?	no	no	si
De ser positiva la respuesta, ¿qué temas o actividades se imparte en esas clases?	-	-	Reconocimiento de las partes del computador y herramientas de Word
¿Disponen de un laboratorio de computación?	si	si	si
¿Cree que se debería enseñarles a los niños fundamentos de programación?	si	si	si
¿Cree que se debe enseñar desde la escuela a programar o fundamentos de programación?	si	si	si
¿Cómo cree que le ayudaría a un niño en el futuro académico si se le enseña lógica de programación?	Desenvolvimiento o académico en el colegio	En las carreras universitarias nuevas	En el colegio y la universidad en carreras tecnológicas
¿Usa herramientas didácticas para ayudar a los niños en su aprendizaje?, de ser positivo ¿qué tipo de herramientas?, de ser negativo ¿Por qué no las usan?	No, los libros tienen sus actividades	No, el distrito de educación no provee	no
¿Cree que las herramientas didácticas ayudan en el desarrollo cognitivo de los niños?	si	si	si
¿Utilizan plataformas que tengan como objetivo enseñar fundamentos de programación a niños en etapa escolar como Code.org, Blockly, Scratch u otras?	no	no	no

¿Los alumnos tienen acceso a herramientas didácticas que les permita una forma diferente y entretenida de aprender??	no	no	Si, en las clases de computación
¿Cree que con el pensum actual el niño está preparado para inclinarse por una vida profesional en el área de ingeniería o desarrollo tecnológico?	no	no	no

Fuente: Levantamiento de información por medio de encuestas

En la presentación de los resultados de la *Tabla 3* se muestra que no se imparte materias relacionadas con la programación y desarrollo tecnológico en edades donde se puede cimentar correctamente el aprendizaje.

Los maestros de estos grados no utilizan ninguna herramienta didáctica adicional a los libros en ninguna materia para enseñar a sus estudiantes, tampoco las conocen, los maestros son conscientes de que el pensum académico es deficiente frente a las exigencias actuales en la preparación futura de los estudiantes en el colegio y la universidad.

2.4. TECNOLOGÍA, INCIDENCIA EN LOS NIÑOS

La tecnología en la última década ha visto un abrumante crecimiento y desarrollo en todas las gamas que la conforman, desde la internet como expansión para el soporte de todas las tecnologías y dispositivos como de todos los productos, programas y aplicaciones que han sido desarrollados enfocándose en las personas como directos beneficiarios y principales consumidores.

Los niños están creciendo en un mundo digital, por lo que la tecnología está integrada en sus vidas. Laptops de juguete, teléfonos celulares y teclados, están disponibles para que los niños exploren el mundo y los incorporen a sus juegos. Ellos podrían percibir que hay cámaras, DVDs¹, teclados electrónicos, discos compactos, computadoras, tabletas, TV y teléfonos celulares en uso a su alrededor, y como los niños pequeños son curiosos y quieren darle sentido a su mundo, están dispuestos a involucrarse con ellos. (KELLY, 2014)

Este contacto de los niños con los dispositivos electrónicos y en especial con los Smartphone conlleva una incidencia tanto positiva como negativa; si el niño o niña usa la tecnología de manera moderada bajo supervisión de sus padres o familiares y con actividades y contenido adecuado a su edad sin crear una dependencia hacia el dispositivo puede ser de enorme utilidad y beneficio para el niño o niña, mas no será de ayuda y beneficio alguno si el niño o niña tienen acceso a estos dispositivos sin supervisión y restricción del contenido al que acceden generándole un posible daño y dependencia hacia estos.

Los niños al tener acceso a los dispositivos móviles con gran facilidad y al estar expuestos a toda la información y aplicaciones que se encuentran disponibles no es factible restringirles el uso de estos recursos, la actualidad se basa en el manejo de herramientas tecnológicas por lo que mal se haría en restringirle los dispositivos móviles a los niños, por el contrario se debe dar una guía para un manejo correcto de los dispositivos y empezar a desarrollar aplicaciones orientadas a los niños de etapa escolar como la que en este proyecto desarrolla ayudando en gran manera a que el tiempo que los niños usan un dispositivo electrónico lo usen en beneficio de su aprendizaje.

¹ Disco Versátil Digital

Existen muchos proyectos en diferentes países que se han enfocado en este propósito, con el uso de herramientas informáticas para el desarrollo de software que permite a los niños en etapa escolar resolver situaciones para llegar a cumplir el objetivo trazado por el creador del software haciendo que el tiempo empleado por el escolar en la tecnología sea aprovechado para su beneficio en el aprendizaje, la tecnología puede ser una herramienta útil si se utiliza en el lugar adecuado para ayudarnos a aprender y no todo el tiempo no como reemplazo de otras cosas. (BBC MUNDO, 2013)

2.5. PROGRAMACIÓN ENFOCADA A NIÑOS

2.5.1. INTRODUCCIÓN

Aprender a programar desde temprana edad es como aprender un nuevo idioma, entre más temprano se haga, más sencillo será para el pequeño adquirir las destrezas y habilidades del objeto a comprender. Se requiere de paciencia, compromiso y práctica para dominar tales habilidades a largo plazo. No solo incrementa la resolución de la adquisición del lenguaje y significado, además incentiva la habilidad de comprensión, utilizando la analogía del discurso como herramienta principal. Sin duda son nuevas formas de expresión a través del lenguaje cifrado en nuevas tecnologías. Hoy en día, en países de Europa, como Estonia, se fomenta a los niños a aprender código desde primer año de educación elemental. Los pequeños que a temprana edad aprenden a codificar tienden a desarrollar nuevas capacidades de resolución de problemas y nuevas aptitudes de aprendizaje lógico-verbal. (TECNOLÓGICA, 2015)

Programar es establecer instrucciones lógicas y ordenadas para ser ejecutadas por un ordenador o sistema programable. El resultado de esta tarea es un paquete de instrucciones al que se lo llama programa. El objetivo de programar es enfocarse en resolver un problema y establecer como un dispositivo u ordenador debe trabajar. (Rancel, 2014)

2.5.2. DEFINICIONES BÁSICAS

2.5.2.1. Lenguaje De Programación

Un lenguaje de programación es una herramienta que permite crear programas y software que controle el comportamiento físico y lógico de una máquina. Como ejemplo se tiene C++, Delphi, Visual Basic, Pascal, Java entre otros. (Medrano, 2015)

2.5.2.2. Programa

Un programa es una secuencia de instrucciones escritas en un lenguaje de programación que orientan al computador o al dispositivo programable a realizar una actividad o a la resolución de un problema en específico. (CCM, 2014)

2.5.2.3. Algoritmo

Un algoritmo es una secuencia ordenada de pasos u operaciones que derivan en la resolución de un problema o la realización de una tarea específica, siendo estos claros, lógicos y tienen un principio y final. (Luna, 2012)

2.5.2.4. Estructuras de Control

Las estructuras de control son pieza fundamental en los lenguajes de programación, permitiendo que las instrucciones se ejecuten indistintamente del orden como fueron concebidas.

2.5.2.4.1. Condicional

Las estructuras de control condicional permiten la ejecución de secuencias en relación a una condición establecida, permitiendo la ejecución de las instrucciones si y solo si se cumple la condición.

Dentro de la estructura de control condicional se tiene la condición IF donde permite la ejecución de las secuencias de código solo si cumple la condición establecida dentro de su sintaxis como se ve en la *Ilustración 5*.

```
If (condición) {  
    Instrucciones;  
}  
Else {  
    Instrucciones;  
}
```

Ilustración 5. Sintaxis de la Estructura de control IF

Fuente: Software de programación de Android.

2.5.2.4.2. Bucles (Repeticiones)

Las estructuras de control de bucles o repeticiones permiten ejecutar repetidamente un conjunto de instrucciones la cantidad de veces que se establezca en la condición, por tanto ejecutará las repeticiones que sean necesarias hasta cumplir con la condición establecida.

Dentro de la estructura de control por bucles o repeticiones se tiene la condición FOR donde permite la ejecución de las secuencias de código hasta que se cumpla la condición establecida dentro de su sintaxis como se ve en la *Ilustración 6*.

```
For (int i = 0; i <= x; i++) {  
    Instrucciones;  
}
```

Ilustración 6. Sintaxis de la Estructura de control FOR

Fuente: Software de programación de Android.

2.5.3. PROGRAMACIÓN MEDIANTE BLOQUES

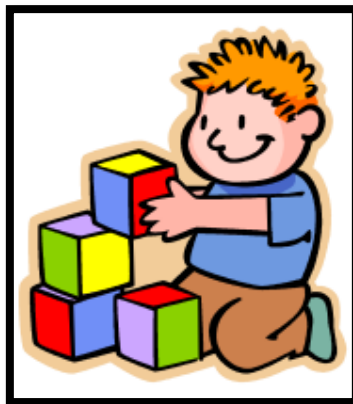


Ilustración 7. Programación mediante Bloques

Fuente: Recuperado de <http://macprogramadores.org/documentacion/bloques.pdf>

Los bloques son una sección de código con una o más declaraciones y sentencias. Un lenguaje de programación que admite bloques, encerrando bloques anidados dentro de otros bloques o sección de código es denominado lenguaje de programación estructurado mediante bloques, ver la *Ilustración 7*.

La función de los bloques de programación es permitir que ciertos grupos de sentencias sean tratados como si fueran una sola sentencia o un solo conjunto, y restringir el ámbito léxico de las variables, los procedimientos y funciones declaradas en un bloque para que no entre en conflicto con variables con el mismo nombre utilizadas para diferentes propósitos en otras partes de un programa.

Tradicionalmente los bloques han formado parte del lenguaje de scripts como Ruby o LISP² bajo nombres como *closures* o *lambdas*, posteriormente Apple introdujo esta idea en las GCC³ bajo el nombre de bloques (Blocks). Desde Mac OS X 10.6 y de iOS 4 el runtime⁴ que permite su uso se encuentra en la librería *libSystem* con la que normalmente enlazan todas las aplicaciones con lo que para usarlos no se necesita aumentar nada a nuestro entorno de programación. Los frameworks Cocoa y Cocoa Touch⁵ empezaron a hacer uso de esta construcción sintáctica en los casos en los que su uso resulta conveniente y útil.

² List Processing (Proceso de Listas)

³ GNU Compiler Collection

⁴ Tiempo de Ejecución

⁵ Cocoa y Cocoa Touch son ambientes de desarrollo de aplicaciones para OS X y iOS respectivamente.

Los bloques son similares a las funciones, en el sentido de que encapsulan una unidad de trabajo o código, es decir, un conjunto de sentencias. Poseen diferencias de las funciones en dos aspectos:

- Actúan como funciones anónimas que pueden pasar como argumentos a métodos o viceversa. Al igual que pasa con los punteros a funciones, se puede usar variables para apuntar a los bloques.
- Además de tener variables locales declaradas en su ámbito, los bloques pueden mantener el estado de variables locales declaradas fuera de su ámbito.

Los bloques son especialmente útiles como alternativa a las funciones callback porque permiten encapsular tanto el código como el estado de la operación que representan.

Específicamente:

- Los bloques se pueden escribir en un punto del programa para indicar operaciones diferidas, es decir, operaciones que se ejecutan más tarde como consecuencias de llamadas callback. Esto permite generar código fuente más claro porque se puede agrupar en un mismo punto del programa la asignación de la operación a ejecutar y la operación en sí.
- Los bloques permiten acceder a las variables locales del ámbito donde se declaran, lo cual facilita la transferencia de la información del contexto al bloque.

- Los bloques se pueden pasar a otros hilos para una ejecución diferida. El compilador se encarga de que el bloque pueda acceder a las variables locales del hilo que creó en el bloque cuando el bloque sea ejecutado por otro hilo distinto. (Hernández F. L., 2010)

2.5.4. HERRAMIENTAS PARA LA ENSEÑANZA DE PROGRAMACIÓN A NIÑOS.

A continuación se detalla cuatro plataformas que permiten el uso y la interacción con la programación mediante bloques y se enfocan en cumplir los objetivos de dar a conocer a los escolares la programación o el fundamento de la codificación con sencillas sentencias y secuencias lógicas acorde a su edad:

2.5.4.1. SCRATCH

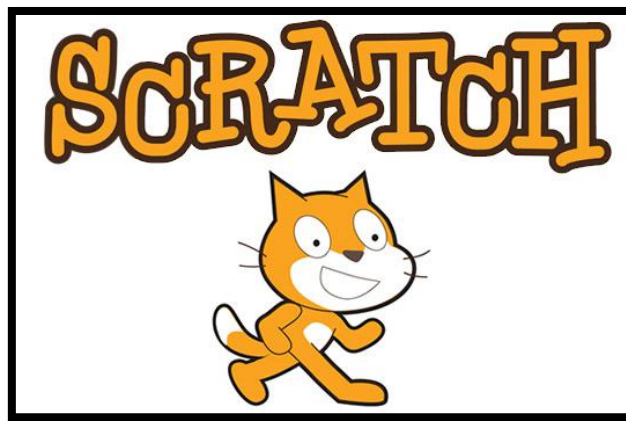


Ilustración 8. Logo representativo de Scratch

Fuente: Recuperado de

<http://www.catedu.es/javierquintana/TIC/TEMATICOS/Arduino/scratch.html>

Scratch es un programa propuesto directamente a los niños en etapa escolar que les permite conocer, explorar y experimentar con los conceptos básicos de programación de computadores a través del uso de una amigable interfaz gráfica, ver logo en la *Ilustración 8*. Scratch está escrito en Squeak, una implementación libre de Smalltalk-80. Es un entorno de programación que facilita el aprendizaje autónomo en los niños despertando su curiosidad e inventiva. Además forma parte de una comunidad de aprendizaje creativo. Fue desarrollado por el "The Lifelong Kindergarten group" en el Media Lab del MIT (Massachusetts Institute of Technology) por un equipo dirigido por Mitchel Resnick y apareció por primera vez en el verano de 2007. Scratch se puede instalar y redistribuir libremente en cualquier ordenador con Windows, Mac OS X o Linux. (Amado, 2012)

Scratch es un software libre de pago que permite, a través de bloques de programación pre escritos, crear programas o secuencias que controlan y combinan imágenes, animaciones, sonidos, e interactuar con el escolar. Puede ser aplicado por alumnos a partir de los 8 años de edad y en entornos educativos.

Scratch se enfoca principalmente en cumplir ciertos objetivos que aportan con la educación de los escolares como también con la familiarización con las herramientas tecnológicas en su vida académica, estos objetivos son: Conocer las funciones básicas del programa Scratch, su entorno gráfico y herramientas, Adquirir la habilidad de organizar bloques de programación, Identificar conceptos básicos de programación, Realizar actividades y pequeñas animaciones o secuencias y Compartir las creaciones en la web.

2.5.4.2. *CODE.ORG*

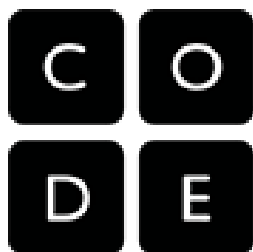


Ilustración 9. Code.org

Fuente: Recuperado de <https://code.org/>

Code.org, una organización sin fines de lucro con un meritorio objetivo: “difundir la programación como parte de la educación básica de los jóvenes”, ver logo en la *Ilustración 9*.

Code.org apareció debido a que en Estados Unidos existía y aún existe una gran demanda de programadores: según datos de la organización, esta oportunidad de empleo es tan grande que para 2020 habrá 1,4 millones de empleos relacionados con la informática y la programación frente a tan solo 400.000 estudiantes. La oferta crece a un ritmo acelerado que es el doble que en otros campos laborales, y a su vez, nueve de cada diez escuelas no ofrecen clases de programación e informática. (Marín, 2013)

Code.org es una vasta plataforma orientada a dar la oportunidad de aprender informática y programación orientada en primera instancia a niños de cualquier escuela y a cualquier público que desee aprender a codificar, para ello dispone de innumerables ejemplos, ejercicios de codificación y con un vasto ambiente en cuanto a interfaz gráfica se refiere, y logrando la atención de los escolares con personajes animados más conocidos del cine animado y la televisión.

Otro de los objetivos que persigue Code.org es el de llegar hacia los escolares para que en su futuro puedan interesarse y seguir una carrera tecnológica, tanto en hombres como en mujeres, que descubran un nuevo mundo que les espera por descubrir, y nuevos mundos que esperan ser creados y que será posible si los niños deciden estudiar carreras tecnológicas.

2.5.4.3. MIT App Inventor



Ilustración 10. MIT App Inventor

Fuente: Recuperado de <http://gamancos.com/gamancos/te-enseamos-como-crear-tu-primera-app-para-android-con-app-inventor-2>

App Inventor es un entorno de desarrollo de aplicaciones para dispositivos electrónicos Android basado en bloques, desarrollado inicialmente por el MIT (Massachusetts Institute of Technology) para cualquier persona con interés de crearse su propia aplicación móvil para cualquier campo, ver logo en la *Ilustración 10*.

Posteriormente toma las riendas el gigante Google en su división Google Labs del proyecto “Google for Education” para continuar desarrollarlo enfocado en el aprendizaje escolar en beneficio de escuelas e institutos interesados en formar parte de este proyecto.

Para crear una aplicación con App Inventor se requiere realizar dos pasos:

- El diseño de la Aplicación en donde se seleccionan los componentes necesarios para la aplicación.
- El editor de bloques en donde se va escogiendo los bloques necesarios en base a la aplicación que se pretenda realizar.

El entorno de App Inventor permite visualizar en pantalla en el dispositivo móvil paso a paso el desarrollo cuando se habilita la conexión con el objetivo de probar el trabajo conforme se va desarrollando, App Inventor incluye un emulador que permite visualizar en este en caso de no tener a la mano un dispositivo Android.

Una vez desarrollada y probada la aplicación en los emuladores de App Inventor o en el dispositivo permite empaquetar los componentes de la aplicación y producir una aplicación independiente para instalar.

Este entorno de desarrollo de App Inventor es compatible con los sistemas operativos Mac OS X, GNU / LINUX y sistemas operativos de Windows. (TuAppInventor, 2012)

2.5.4.4. **BLOCKLY**



Ilustración 11. Blockly

Fuente: Recuperado de <https://code.google.com/p/blockly/>

Es una plataforma con un entorno para la programación orientado a los niños a la codificación, a las nociones y aspectos fundamentales. Respaldada por Google Developers⁶, se asemeja a las piezas de un rompecabezas y que se puede programar de una forma intuitiva y simple enfocándose plenamente en temas didácticos, ver logo en la *Ilustración 11*.

Blockly introduce el concepto de programación visual, donde no se requiere el uso de comandos tipados por el escolar, y posee la función de poder convertir una aplicación hecha en Blockly a otros lenguajes como JavaScript, Python o XML⁷ para alguien que conozca de estos lenguajes y quiera estudiar el proceso de traducción de código.

Esta plataforma se enfoca en la lógica brindando una serie de comandos simples, permite arrastrar los distintos componentes de control, lógica, operaciones matemáticas, texto, listados y procesos para crear sencillos scripts que posteriormente se podrá **exportar a lenguajes como JavaScript, Dart, Python o XML**. Esto último es un punto interesante a tener en cuenta. Al momento está disponible para aplicaciones simples, las aplicaciones complejas es un tema pendiente y con una prestación a un público juvenil, como ellos mismos

⁶ Desarrolladores de Google

⁷ Lenguaje de Marcas Extensible

explican en la documentación “no intente usarlo para mantener el Kernel⁸ de Linux”, aunque puede ser usado por programadores experimentados para escribir una línea de código rápido para luego exportarlo a código. (Rodriguez, 2012)

2.6. SISTEMAS OPERATIVOS MÓVILES MAS COMUNES QUE SOPORTAN APLICACIONES MEDIANTE BLOQUES

Existen dos sistemas operativos dominantes dentro del mercado de los teléfonos móviles, el primero es un sistema operativo propietario perteneciente al gigante tecnológico Apple Inc, tanto en el desarrollo, diseño como en las patentes, el segundo es un sistema operativo no propietario de distribución libre y que es usado por la mayoría de marcas que dominan el mercado de los teléfonos móviles con marcas reconocidas como Samsung, Sony, Huawei entre otras y que está respaldada bajo una licencia GPL⁹.

A continuación se detallan estos dos sistemas operativos:

⁸ Núcleo de linux

⁹ General Public License (Licencia Pública General)

2.6.1. SISTEMA OPERATIVO iOS



Ilustración 12. iOS de Apple

Fuente: Recuperado de <http://tecnologia4all.com/analisis-ios-8-problemas-y-novedades-del-sistema-operativo-de-apple/2633/>

El sistema operativo móvil más avanzado del mundo, como lo denomina su creador, **iOS** es un sistema operativo móvil de la multinacional Apple Inc. Originalmente desarrollado para el iPhone (iPhone OS), después se ha usado en dispositivos como el iPod touch y el iPad, ver logo en la *Ilustración 12*. No permite la instalación de iOS en hardware de terceros. Tenía el 26% de cuota de mercado de sistemas operativos móviles vendidos en el último cuatrimestre de 2010, detrás de Android y Windows Phone. Actualmente su sistema operativo se encuentra en la novena versión, mejor conocida como iOS 9. (Apple Inc., 2015)

iOS 9 es la esencia del iPhone, el iPad y el iPod touch por muchos motivos. Es eficaz y elegante como nunca. Más seguro y fácil de usar que cualquier versión anterior. Y tiene unas prestaciones con las que da gusto hacer hasta las tareas más cotidianas, sus características se

ven en la *Tabla 4*. Además, está diseñado para aprovechar al máximo la avanzada tecnología del hardware de Apple. Por eso tus dispositivos siempre estarán un paso por delante. O dos. (Apple Inc., 2015)

Tabla 4. Información general de iOS

Desarrollador	Apple Inc.
Modelo de Desarrollo	Software Propietario
Lanzamiento Inicial	26 de Junio del 2007
Última versión estable	iOS 9.1
Escrito en	C, C++, Objective-C, Swift
Núcleo	XNU
Tipo de núcleo	Núcleo híbrido (XNU)
Interfaz Gráfica por defecto	Cocoa Touch (Multi táctil GUI)
Plataformas soportadas	ARM (iPad, iPhone y iPod Touch)
Licencia	APSL y Apple EUA
Idiomas	34

Fuente: Recuperado de <http://www.apple.com/es/ios/>

En general este sistema operativo es muy aprovechable en sus capacidades y prestaciones debido a que tanto el hardware como el software provienen de un mismo desarrollador, se aprovecha el 100% del potencial del software como del hardware.

El sistema operativo iOS es una derivación de OS¹⁰ X que es un sistema operativo tipo Unix, iOS cuenta con cuatro niveles de abstracción: el nivel del núcleo del sistema operativo, el nivel de Servicios Principales, el nivel de Medios y el nivel de Cocoa Touch y en la actualidad cuenta con varias versiones.

2.6.2. SISTEMA OPERATIVO ANDROID



Ilustración 13. Sistema Operativo Android

Fuente: Recuperado de <http://peru.com/epic/epic-users/android-9-razones-elegir-este-sistema-operativo-noticia-311282>

Android es un sistema operativo basado en el núcleo Linux, ver logo en la *Ilustración 13*. Fue diseñado principalmente para dispositivos móviles con pantalla táctil, como teléfonos inteligentes, tabletas; y también para relojes inteligentes, televisores y automóviles. Inicialmente fue desarrollado por Android Inc, empresa que Google respaldó económicamente y más tarde, en 2005, compró. Android fue presentado en 2007 junto la fundación del Open Handset Alliance (un consorcio de compañías de hardware, software y telecomunicaciones) para avanzar en los estándares abiertos de los dispositivos móviles. (Android, 2015)

¹⁰ Sistema Operativo

La versión básica de Android es conocida como Android Open Source Project (AOSP), además es el sistema operativo para móviles más vendidos y más usados del mundo, el éxito se basa en las continuas y óptimas actualizaciones, es un sistema libre que puede ser modificado por el usuario sin tener problemas legales, puede ser compartido y se crean nuevas versiones, es muy personalizable con los temas, la interfaz de usuario, posee un regente que crea un sin fin de aplicaciones que derivan en calidad en estas para la competencia y distribución, los móviles que usan este sistema operativo son mucho más baratos respecto al otro sistema operativo competidor y con una mayor gama de oferta en dispositivos y finalmente está respaldado por el Gigante Google que le ha llevado a Android a otro nivel dentro del mundo del internet y la tecnología, donde Google genera muchos servicios y son soportados y compatibles con Android, en la *Tabla 5* se citan sus características.

Tabla 5. Información general de Android

Desarrollador	Google, Open Handset Alliance
Modelo de Desarrollo	Código Abierto
Lanzamiento Inicial	23 de Septiembre del 2008
Última versión estable	Android 6.0
Escrito en	C, C++, Java
Núcleo	Linux
Tipo de núcleo	Monolítico
Interfaz Gráfica por defecto	Material Design
Plataformas soportadas	ARM, x86, MIPS, IBM

Sistema de gestión de paquetes	Google Play, APK
Licencia	Apache 2.0 y GNU GPL
Idiomas	Multilingüe

Fuente: Recuperado de <https://www.android.com/>

2.7. TECNOLOGÍAS INALÁMBRICAS

Generalmente las tecnologías inalámbricas usan ondas de radiofrecuencia de muy baja potencia y en una banda específica, son de uso libre para transmitir y comunicarse, entre dispositivos, véase la *Ilustración 14*. Estos parámetros de libertad de transmisión, sin necesidad de licencia, han beneficiado que el número de equipos, especialmente los ordenadores, que usan las ondas para comunicarse, a través de redes inalámbricas haya incrementado significativamente. La tendencia a la movilidad y la necesidad de estar en varios lugares al mismo tiempo ha permitido desarrollos significativos en las tecnologías y los sistemas inalámbricos, y el objetivo es ir reemplazando los cables en todo tipo de comunicación que desde el punto de vista técnico sea factible, no solo en el campo de las telecomunicaciones sino también en televisión, telefonía, seguridad, domótica entre otros. Un fenómeno social que ha adquirido gran importancia en todo el mundo como consecuencia del uso de las tecnologías inalámbricas son las comunidades wireless¹¹ que buscan la difusión de redes alternativas a las establecidas y comerciales. (Olmos, 2013)

¹¹ Sin cables

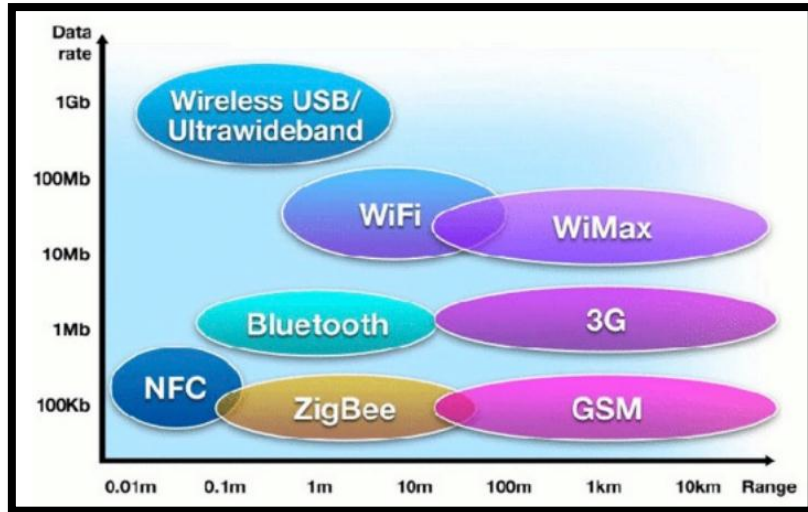


Ilustración 14. Tecnologías Inalámbricas

Fuente: Recuperado de http://www.fut-electronics.com/wp-content/plugins/fe_downloads/Uploads/Comparison%20of%20Wireless%20Technologies.pdf

El término "inalámbrico" hace referencia a la no utilización de cables que permitan conectar varios dispositivos entre sí. Las conexiones inalámbricas que se establecen entre los clientes remotos y un proveedor confieren a las empresas flexibilidad y prestaciones muy diversas especialmente para las operadoras celulares y empresas que se encuentren en el ámbito tecnológico e inalámbrico.

A continuación se detalla tres tecnologías inalámbricas que se encuentran en los quehaceres diarios y que además se han posicionado como grandes soluciones a la comunicación, con una gran variedad de aplicaciones y una muy buena aceptación por parte de los consumidores de estas tecnologías inalámbricas.

2.7.1. WI-FI



Ilustración 15. Logo representativo Wi-Fi

Fuente: Recuperado de <https://blog.ecoventsystems.com/2015/03/wirelessprotocols/>

Wi-Fi está diseñado para la comunicación de dispositivo a dispositivo, en reemplazo de las redes cableadas, ver el logo en la *Ilustración 15*. Permite navegar por Internet a velocidades de banda ancha cuando se enlaza a un punto de acceso. Wi-Fi ofrece mucha mayor velocidad de datos, mientras que Bluetooth y Zigbee ofrecen cantidades inferiores. Wi-Fi funciona tanto en 2,4 Ghz o 5 Ghz como banda de frecuencia, y es compatible con dispositivos con una fuente de alimentación constante.

Wi-fi se encuentra certificado y avalado bajo la Alianza Wi-Fi, que es una organización sin ánimo de lucro que promueve la tecnología Wi-Fi y certifica los productos Wi-Fi, además en su desarrollo ha fomentado varias versiones, basado cada una de estas en un estándar IEEE 802.11 aprobado. Se nombran a continuación:

- Los estándares IEEE 802.11b, IEEE 802.11g e IEEE 802.11n comparten de una aprobación internacional debido a que la banda de 2.4 GHz está aprovechable universalmente, con una velocidad de hasta 11 Mbit/s, 54 Mbit/s y 300 Mbit/s, respectivamente.

- En la actualidad se maneja también el estándar mejorado IEEE 802.11ac, conocido como WIFI 5, que trabaja en la banda libre de 5 Ghz y que goza de una operatividad con canales relativamente limpios (sin interferencia). La banda de 5 Ghz ha sido recientemente habilitada y, además, no existen otras tecnologías (Bluetooth, microondas, ZigBee, WUSB¹²) que la estén utilizando, por lo tanto existen muy poca interferencia. Su alcance es algo menor que el de los estándares que trabajan a 2.4 Ghz (aproximadamente un 10 %), debido a que la frecuencia es mayor (a mayor frecuencia, menor alcance). (Clic, 2015)

Existen más tecnologías inalámbricas como el Bluetooth que también trabajan a una frecuencia de 2.4 Ghz, por lo tanto puede presentar interferencias con la tecnología wi-fi. WiFi es una eficiente solución para reemplazar un cable Ethernet evitando enredarse con cables por todas partes. El beneficio de WiFi es que puede conectarse a un Punto de Acceso o enrutador existente, permitiendo a dispositivos de usuario final tener conectividad y salida hacia la Internet a través de Wi-Fi. (Kanda, 2012)

2.7.2. BLUETOOTH



Ilustración 16. Logo representativo Bluetooth

Fuente: Recuperado de <https://blog.ecoventsystems.com/2015/03/wirelessprotocols/>

¹² Wireless USB

Bluetooth está diseñado para permitir a los módulos periféricos de corto alcance de las computadoras, tales como ratones de ordenador, teclados, impresoras, entre otros, puedan “hablar” con el equipo, véase el logo en la *Ilustración 16*. Esta gama de aplicaciones se conoce como red de área personal inalámbrica (WPAN). Bluetooth opera con una banda de frecuencia de 2,4 Ghz, y realmente sólo funciona bien dentro de una habitación, no a través de una casa entera, su alcance es limitado.

Bluetooth es una frecuencia de radio de disponibilidad universal que conecta entre sí los dispositivos habilitados para Bluetooth ubicados dentro de un área de cobertura de hasta 10 metros. Permite conectar un computador portátil o un dispositivo de bolsillo con otros computadores portátiles, teléfonos móviles, cámaras, impresoras, teclados, parlantes o un mouse.

El uso de una red Bluetooth admite intercambiar archivos en reuniones improvisadas con mucha facilidad y ahorrar tiempo imprimiendo documentos sin necesidad de conectarse a una red fija.

Bluetooth desde su más antigua versión ha mostrado una eficiencia tanto en cobertura como en tasa de transmisión, de ahí su aceptación, en su versión 1.2 manejó un ancho de banda de 1 Mbit/s, para ser mejorado y sustituido por la versión 2.0 + EDR¹³ que trabajó con un ancho de banda de 3 Mbit/s, luego en la siguiente versión, la 3.0 +HS¹⁴ se logró un ancho

¹³ Tasa de Datos Mejorada

¹⁴ Alta Velocidad

de banda altamente significativo de 24 Mbit/s y finalmente la versión 4.0 se impuso en los dispositivos móviles principalmente donde se generó el mayor auge manteniendo el ancho de banda de su versión anterior.

Opera precisamente en la frecuencia de 2,4 a 2,48 Ghz con amplio espectro y saltos de frecuencia para transmitir en Full Duplex con 1600 saltos por segundo. Los saltos de frecuencia se logran entre 79 frecuencias con intervalos de 1 MHz; esto permite dar seguridad y robustez con una distancia máxima de 10 metros y con una potencia de 0 dBm (1 mW).

2.7.3. ZIGBEE



Ilustración 17. Logo Representativo ZigBee

Fuente: Recuperado de <https://blog.ecoventsystems.com/2015/03/wirelessprotocols/>

Este es un protocolo inalámbrico que también opera en la banda de 2,4 Ghz, como Wi-Fi y Bluetooth, pero manipula velocidades de datos más bajas, es una solución ideal para crear redes flexibles y robustas para aplicaciones de control y monitoreo, véase el logo en la

Ilustración 17. Las principales ventajas de ZigBee son:

Bajo consumo de energía

Red muy robusta

Hasta 65,645 nodos

Muy fácil de agregar o quitar nodos de la red

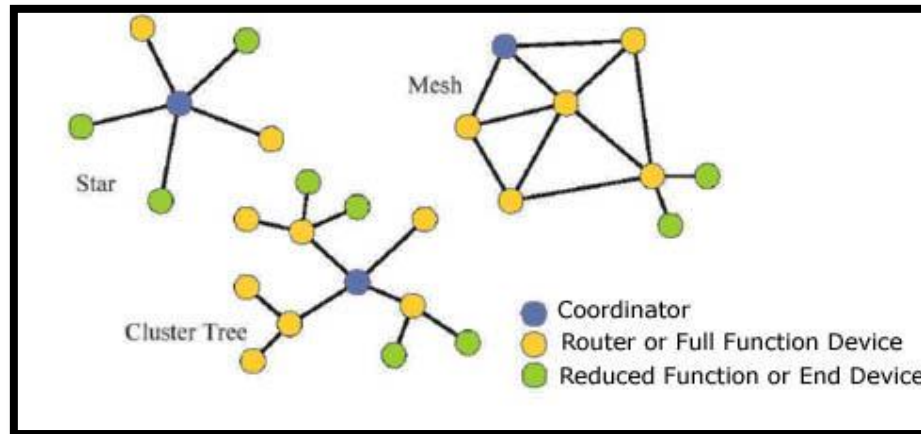


Ilustración 18. Topologías de ZigBee

Fuente: Recuperado de <https://www.kanda.com/zigbee-wireless.html>

Las redes ZigBee pueden ser clúster, estrella o redes de malla como se muestra en la *Ilustración 18*. Cada red ZigBee tiene un dispositivo que se configura como un coordinador que controla e inicializa la red. Otros dispositivos son tanto de configuración como routers que pasan los datos o como puntos finales, que sólo logra tener una conexión. Los dispositivos finales pueden configurarse para el modo sleep¹⁵, por tanto utilizan muy poca energía y trabajan con batería.

Debido a este requisito de energía muy bajo, ZigBee es ideal para redes de sensores u otras aplicaciones de control, que requieren:

¹⁵ Reposo

- Tasas de datos bajos o medianos
- Muchos nodos (hasta 65.565)
- Fácil adición o remoción de nodos de la red
- Red de malla robusta que no se cae si un nodo falla
- Muy baja potencia, el equipo funciona con baterías
- De largo alcance

Zigbee está respaldado por el estándar IEEE 802.15.4, utiliza 900 Mhz en los EE.UU., 868 Mhz en Europa y 2,4 Ghz en todo el mundo.

2.7.4. TABLA COMPARATIVA ENTRE TECNOLOGÍAS INALÁMBRICAS

En la *Tabla 6* se presenta una comparación de las características de tres protocolos o tecnologías inalámbricas, ZigBee, Wi-Fi y Bluetooth.

Tabla 6. Comparativa entre tecnologías inalámbricas

	ZigBee	Wi-Fi (802.11)	Bluetooth
Aplicación	Monitoreo y control	Email, web, video	reemplazo de cable
capa Física/MAC	IEEE 802.15.4	IEEE 802.11	IEEE 802.15.1
Velocidad de datos	20, 40 y 250 Kbits/s	11 - 54 Mbits /s	1 Mbit/s
Rango	10 - 100 metros	50 - 100 metros	10 metros
Topología de red	Ad-hoc, estrella, malla	Punto a hub	Ad-hoc, redes muy pequeñas
Frecuencia de operación	868 Mhz EU, 928 Mhz EUA, 2.4 Ghz	2.4 y 5 Ghz	2.4 Ghz

Complejidad	Baja	alta	alta
Consumo de energía	muy baja	alta	medio
Numero de dispositivos por red	64 K	32 por punto de acceso	7
Latencia de red	< 30 ms	Conexión 3 - 5 sec	hasta 10 sec
Aplicaciones típicas	Monitoreo y control industrial, sensores	Conectividad LAN inalámbrica	Conectividad inalámbrica entre dispositivos

Fuente: Adaptado de <https://www.kanda.com/zigbee-wireless.html>

2.8. COMPONENTES ELECTRÓNICOS

Dentro de este tema se consideran todos los componentes electrónicos que se encuentran involucrados o intervienen en el desarrollo del prototipo electrónico, como también partes y piezas adicionales que se usan para la implementación completa del prototipo.

2.8.1. PLATAFORMA ELECTRÓNICA ARDUINO

Arduino es una plataforma electrónica de código abierto basada en hardware y software fácil de usar, está dirigido a personas interesadas a realizar cualquier tipo de proyecto interactivo. (Arduino, Arduino, 2013)

Las placas Arduino son capaces de leer las variables o la luz en un sensor, un dedo en un botón, o un mensaje de Twitter y transformarla en una salida para la activación de un motor, encender un diodo LED¹⁶ o publicar algo en línea. Se puede ordenarle a la placa qué

¹⁶ Light Emitter Diode (Diodo Emisor de Luz)

hacer mediante el envío de un conjunto de instrucciones al Microcontrolador ubicado en la placa. Para esto se utiliza el lenguaje de programación de Arduino basado en Wiring¹⁷, y el software de Arduino (IDE), basado en Processing¹⁸.

A través de los años Arduino ha sido el cerebro de miles de proyectos, a partir de objetos cotidianos a los instrumentos científicos complejos. Una comunidad mundial de fabricantes, estudiantes, aficionados, programadores y profesionales que ha reunido alrededor de esta plataforma de código abierto, sus contribuciones han añadido hasta una increíble cantidad de conocimiento accesible que puede ser de gran ayuda para los principiantes como para expertos.

Arduino nació del Ivrea Interaction Design Institute como una herramienta fácil para prototipos rápidos, dirigido a estudiantes sin experiencia en electrónica y programación. Tan pronto como llegó a una comunidad más amplia, la placa Arduino comenzó a cambiar para adaptarse a las nuevas necesidades y desafíos, diferenciar su oferta de tablas simples de 8 bits a los productos para aplicaciones de IoT¹⁹, impresión 3D portátil y entornos integrados. Todas las placas Arduino son completamente de código abierto, permitiendo a los usuarios crear de forma independiente y, finalmente, adaptarlos a sus necesidades particulares. El software también es de código abierto, y está creciendo a través de las aportaciones de los usuarios en todo el mundo. (Arduino, What is Arduino?, 2013)

¹⁷ Plataforma de Desarrollo

¹⁸ Librería de Software Flexible

¹⁹ Internet of Things (Internet de las cosas)

A continuación se enumera las placas Arduino más reconocidas y utilizadas en el desarrollo de prototipos electrónicos que se pueden usar dentro de este trabajo de titulación, las características, procesador, número de pines entre otros y que además se encuentren disponibles en el mercado.

2.8.1.1. ARDUINO DUE

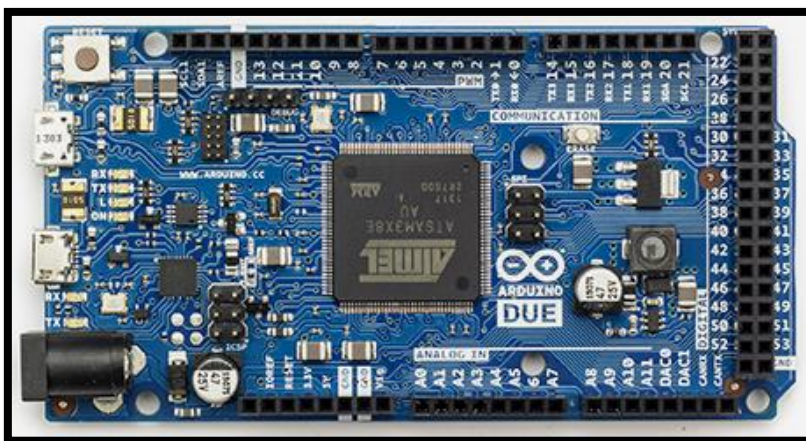


Ilustración 19. Placa Arduino DUE

Fuente: Recuperado de <https://www.arduino.cc/en/Main/ArduinoBoardDue>

El Arduino Due, ver la *Ilustración 19*, es la primera placa Arduino basada en un Microcontrolador de núcleo ARM²⁰ de 32 bits. Con 54 pines digitales de entrada - salida, 12 entradas analógicas (PWM²¹), es una placa electrónica basada en la CPU Atmel SAM3X8E ARM Cortex-M3, 4 UART²²s, un reloj de 84 MHz, 2 convertidores Digital Analógico, un conector de alimentación, una cabecera de SPI²³, un botón de reinicio y un botón de borrado, la placa contiene todo lo necesario para apoyar el Microcontrolador; simplemente se conecta a

²⁰ Advanced RISC Machine (RISC = Reduced Instruction Set Computer)

²¹ Pulse Width Modulation (Modulación por ancho de pulsos)

²² Universal Asynchronous Receiver – Transmitter (Transmisión- Recepción asíncrona universal)

²³ Serial Peripheral Interface (Interfaz periférica serial)

un computador con un cable micro-USB o de poder con un adaptador de corriente alterna o la batería a corriente continua para iniciar, funciona a 3.3 V. (Arduino, Arduino Due, 2013)

En la *Tabla 7* se detalla las especificaciones técnicas del Microcontrolador AVR de Arduino DUE:

Tabla 7. Especificaciones Técnicas del Microcontrolador AVR

Especificaciones Técnicas	
Microcontrolador	AT91SAM3X8E
Voltaje de Operación	3.3V
Voltaje de Entrada (Recomendado)	7-12V
Voltaje de Entrada (Límites)	6-16V
Pines Digitales I/O	54 (of which 12 provide PWM output)
Pines Análogos de Entrada	12
Pines Análogos de Salida	2 (DAC)
Total DC Salida sobre todas las líneas I/O	130 mA
Corriente directa para pines 3.3V	800 mA
Corriente directa para pines 5V	800 mA
memoria Flash	512 KB disponible para aplicaciones
SRAM	96 KB (two banks: 64KB and 32KB)
Velocidad de reloj	84 MHz
Largo	101.52 mm
Ancho	53.3 mm
Peso	36 g

Fuente: Adaptado de <https://www.arduino.cc/en/Main/ArduinoBoardDue>

2.8.1.2. ARDUINO LEONARDO

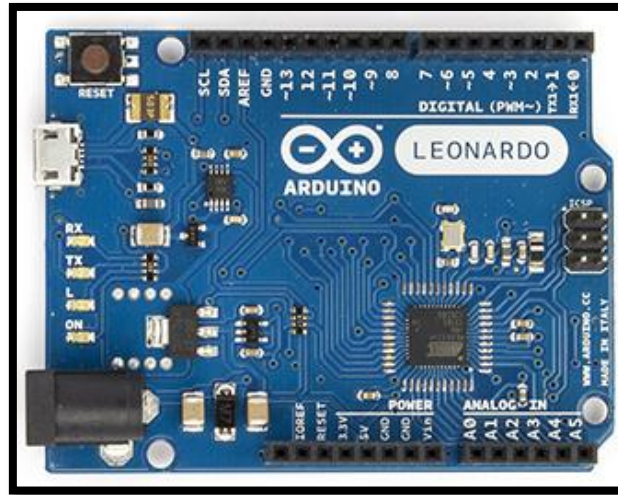


Ilustración 20. Placa Arduino Leonardo

Fuente: Recuperado de <https://www.arduino.cc/en/Main/ArduinoBoardLeonardo>

El Arduino Leonardo, ver la *Ilustración 20*, es una placa electrónica basada en el ATmega32u4, cuenta con 20 pines digitales de entrada / salida, de estos 7 se pueden utilizar como salidas PWM y 12 como entradas analógicas, un oscilador de cristal de 16 MHz, una conexión micro USB, un conector de alimentación y un botón de reinicio. Posee todo lo necesario para apoyar al Microcontrolador; simplemente se conecta a un computador con un cable USB o de poder con un adaptador de corriente alterna o la batería a corriente continua para iniciar.

Esta placa difiere de otras placas debido a que el ATmega32u4 ha incorporado la comunicación USB, eliminando la necesidad de un procesador secundario. Esto permite que el Leonardo aparezca a un ordenador conectado como un ratón y el teclado, además de un puerto serie / COM. (Arduino, Arduino Leonardo, 2013)

En la *Tabla 8* se muestra un resumen de especificaciones técnicas del Arduino Leonardo:

Tabla 8. Especificaciones Técnicas del Arduino Leonardo.

Especificaciones Técnicas	
Microcontrolador	ATmega32u4
Voltaje de Operación	5V
Voltaje de Entrada (Recomendado)	7-12V
Voltaje de Entrada (Límites)	6-20V
Pines Digitales I/O	20
Pines Análogos de Entrada	12
Canales PWM	7
canales Análogos de Entrada	12
Corriente directa por pin I/O	40 mA
Corriente directa por pin 3.3V	50 mA
memoria Flash	32 KB (ATmega32u4) de estos 4 KB usados para gestor de arranque.
SRAM	2.5 KB
Velocidad de reloj	16 Mhz
Largo	68.6 mm
Ancho	53.3 mm
Peso	20 g

Fuente: Adaptado de <https://www.arduino.cc/en/Main/ArduinoBoardLeonardo>

2.8.1.3. ARDUINO MEGA 2560

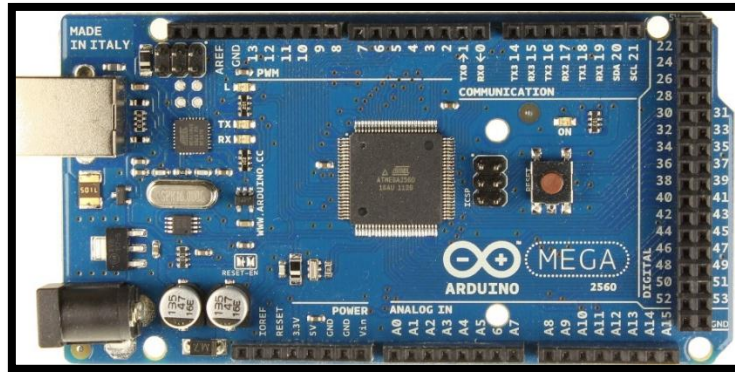


Ilustración 21. Placa Arduino Mega 2560

Fuente: Recuperado de <https://www.arduino.cc/en/Main/ArduinoBoardMega2560>

El Arduino Mega 2560, ver *Ilustración 21*, es una placa electrónica basada en el Atmega2560, cuenta con 54 pines digitales de entrada / salida de los cuales 15 se pueden utilizar como salidas PWM, 16 entradas analógicas, 4 UARTs, un oscilador de cristal de 16 MHz, una conexión USB, un conector de alimentación y un botón de reinicio. Posee todo lo necesario para apoyar al Microcontrolador; simplemente se conecta a un computador con un cable USB o de poder con un adaptador de corriente alterna o la batería a corriente continua para iniciar. La placa Mega 2560 es compatible con la mayoría de shields diseñados para el Uno y las antiguas placas Duemilanove o Diecimila. (Arduino, Arduino MEGA 2560, 2013)

En la *Tabla 9* se muestra un resumen de especificaciones técnicas del Arduino MEGA 2560:

Tabla 9. Especificaciones Técnicas del Arduino MEGA 2560

Especificaciones Técnicas	
Microcontrolador	ATmega2560
Voltaje de Operación	5V
Voltaje de Entrada (Recomendado)	7-12V
Voltaje de Entrada (Límites)	6-20V
Pines Digitales I/O	54 (15 proveen salidas PWM)
Pines Análogos de Entrada	16
EEPROM	4 KB
Corriente directa por pin I/O	20 mA
Corriente directa por pin 3.3V	50 mA
memoria Flash	256 KB. 4 KB usados para gestor de arranque.
SRAM	8 KB
Velocidad de reloj	16 Mhz
Largo	101.52 mm
Ancho	53.3 mm
Peso	37 g

Fuente: Adaptado de <https://www.arduino.cc/en/Main/ArduinoBoardMega2560>

2.8.1.4. ARDUINO MINI PRO

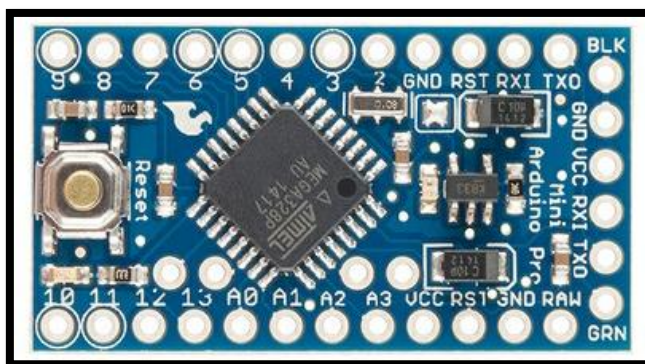


Ilustración 22. Placa Arduino Mini PRO

Fuente: Recuperado de <https://www.arduino.cc/en/Main/ArduinoBoardProMini>

El Arduino Mini PRO, ver la *Ilustración 22*, es una placa electrónica pequeña originalmente basada en el ATmega168, pero ahora suministra con el ATmega328, destinados a ser utilizados en protoboards y cuando el espacio es un bien reducido, cuenta con 14 pines digitales de entrada / salida, de estos 6 se pueden utilizar como salidas PWM y 6 como entradas analógicas además un oscilador de cristal de 8 MHz. Se puede programar con el adaptador USB o con un adaptador serial RS232 (TTL), permite tener los componentes sobre el protoboard, tiene un botón de reinicio. (Arduino, Arduino Mini PRO, 2013)

En la *Tabla 10* se muestra un resumen de especificaciones técnicas del Arduino Mini PRO:

Tabla 10. Especificaciones Técnicas del Arduino Mini

Especificaciones Técnicas	
Microcontrolador	ATmega328

Voltaje de Operación	3.3 - 5V
Voltaje de Entrada	3.35 – 12 V
Pines Digitales I/O	14 (6 proveen salidas PWM)
Pines Análogos de Entrada	6
EEPROM	1 KB
Corriente directa por pin I/O	40 mA
memoria Flash	32 KB. 0.5 KB usados para gestor de arranque.
SRAM	2 KB
Velocidad de reloj	8 Mhz
Largo	33 mm
Ancho	18 mm

Fuente: Adaptado de <https://www.arduino.cc/en/Main/ArduinoBoardProMini>

2.8.1.5. ARDUINO NANO



Ilustración 23. Placa Arduino Nano

Fuente: Recuperado de <https://www.arduino.cc/en/Main/ArduinoBoardNano>

El Arduino Nano, ver *Ilustración 23*, es una placa pequeña, completa y amigable basada en el ATmega328 (Arduino Nano 3.x) o ATmega168 (Arduino Nano 2.x). Tiene más o menos la misma funcionalidad del Arduino Duemilanove, pero en un paquete diferente. Le falta solamente una toma de alimentación de corriente continua, y trabaja con un cable USB Mini-B

en vez de una normal. El Arduino Nano fue diseñado y producido por Gravitech. (Arduino, Arduino Nano, 2013)

En la *Tabla 11* se muestra un resumen de especificaciones técnicas del Arduino Nano:

Tabla 11. Especificaciones técnicas del Arduino Nano

Especificaciones Técnicas	
Microcontrolador	ATmega328
Voltaje de Operación (Nivel Lógico)	5V
Voltaje de Entrada (Recomendado)	7-12V
Voltaje de Entrada (Límites)	6-20V
Pines Digitales I/O	14 (6 proveen salidas PWM)
Pines Análogos de Entrada	8
Corriente directa por pin I/O	40 mA
Memoria Flash	32 KB. 2 KB usados para gestor de arranque.
SRAM	2 KB
EEPROM	1 KB
Velocidad de reloj	16 Mhz
Largo	45 mm
Ancho	18 mm
Peso	5 g

Fuente: Adaptado de <https://www.arduino.cc/en/Main/ArduinoBoardNano>

2.8.1.6. ARDUINO UNO

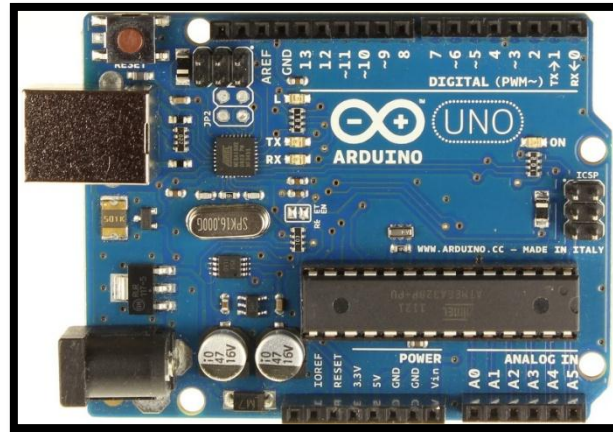


Ilustración 24. Placa Arduino Uno

Fuente: Recuperado de <https://www.arduino.cc/en/Main/ArduinoBoardUno>

El Arduino Uno, ver la *Ilustración 24*, es una placa electrónica basada en el ATmega328P, cuenta con 14 pines digitales de entrada / salida, de estos 6 se pueden utilizar como salidas PWM, 6 entradas analógicas, un cristal de cuarzo de 16 MHz, una conexión USB, un conector de alimentación y un botón de reinicio. Contiene todo lo necesario para apoyar al Microcontrolador; simplemente se conecta a un computador con un cable USB o de poder con un adaptador de corriente alterna o la batería a corriente continua para iniciar. Se puede probar con una placa UNO sin preocuparse demasiado por hacer algo mal, es muy económica y fácil de programar.

En la *Tabla 12* se muestra un resumen de especificaciones técnicas del Arduino Uno:

Tabla 12. Especificaciones Técnicas de la Placa Arduino Uno

Especificaciones Técnicas	
Microcontrolador	ATmega328P

Voltaje de Operación (Nivel Lógico)	5V
Voltaje de Entrada (Recomendado)	7-12V
Voltaje de Entrada (Límites)	6-20V
Pines Digitales I/O	14 (6 proveen salidas PWM)
Pines Digitales I/O PWM	6
Pines Análogos de Entrada	6
Corriente directa por pin I/O	20 mA
Corriente directa por pin 3.3V	50 mA
Memoria Flash	32 KB. 0.5 KB usados para gestor de arranque.
SRAM	2 KB
EEPROM	1 KB
Velocidad de reloj	16 Mhz
Largo	68.6 mm
Ancho	53.4 mm
Peso	25 g

Fuente: Adaptado de <https://www.arduino.cc/en/Main/ArduinoBoardUno>

2.8.1.7. ARDUINO YUN

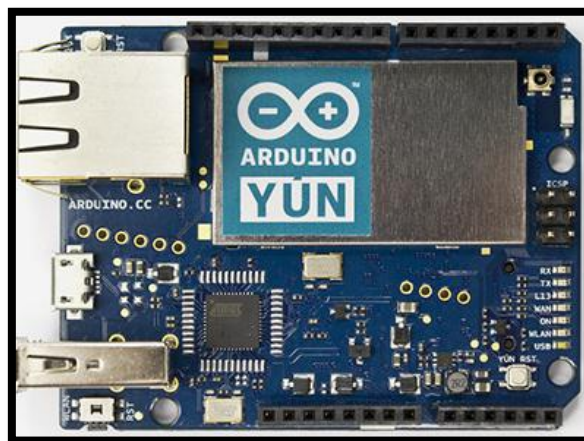


Ilustración 25. Placa Arduino Yun

Fuente: Recuperado de <https://www.arduino.cc/en/Main/ArduinoBoardYun>

El Arduino Yun, ver la *Ilustración 25*, es una placa electrónica basada en el ATmega32u4 y Atheros AR9331, el procesador Atheros es compatible con una distribución Linux basada en OpenWrt llamado OpenWrt-Yun, la placa está incorporada con Ethernet y soporte Wi - Fi, un puerto USB-A, una ranura para tarjeta micro-SD, 20 pines digitales de entradas/salida, de estos 7 se pueden utilizar como salidas PWM y 12 como entradas analógicas, un oscilador de cristal de 16 MHz, una conexión micro USB, y 3 botones de reinicio.

El Yun se distingue de otras placas Arduino debido a que se puede comunicar con la distribución de Linux integrada, ofreciendo un equipo en red de gran alcance con la facilidad de Arduino.

Arduino Yun es similar a la placa Leonardo en que el ATmega32u4 se ha incorporado en la comunicación USB, eliminando la necesidad de un procesador secundario. Esto permite que el Yun aparezca a un ordenador conectado como un ratón y el teclado, además de un puerto serie COM. (Arduino, Arduino Yun, 2013)

En la *Tabla 13* se muestra un resumen de especificaciones técnicas del Arduino Yun:

Tabla 13. Especificaciones Técnicas del Arduino Yun

Especificaciones Técnicas	
Microcontrolador	ATmega32u4
Voltaje de Operación	5V

Voltaje de Entrada	5V
Pines Digitales I/O	20
Canales PWM	7
Pines Análogos de Entrada	12
Corriente directa por pin I/O	40 mA
Corriente directa por pin 3.3V	50 mA
Memoria Flash	32 KB. 4 KB usados para gestor de arranque.
SRAM	2.5 KB
EEPROM	1 KB
Velocidad de reloj	16 Mhz

Fuente: Adaptado de <https://www.arduino.cc/en/Main/ArduinoBoardYun>

En el capítulo siguiente concerniente al desarrollo del prototipo electrónico se seleccionará la mejor placa Arduino que vaya de acuerdo con este proyecto de acuerdo a las especificaciones técnicas.

2.8.2. TABLA COMPARATIVA DE PLACAS ARDUINO

En la *Tabla 14* se presenta una comparativa de las placas Arduino con las características principales de cada una de las plataformas.

Tabla 14. Comparativa de Placas Arduino

Características / Placas	ARDUINO UNO	ARDUINO MEGA2560	ARDUINO MINI-PRO	ARDUINO DUE	ARDUINO NANO	ARDUINO YUN
Microcontrolador	Atmega328	Atmega2560	Atmega328	ARM AT91SAM3X8E	Atmega2560	Atmega32u4
Voltaje de operación	5 voltios	5 voltios	5 voltios	3.3 voltios	5 voltios	5 voltios
Rango de voltaje	6-20 voltios	6-20 voltios	5-12 voltios	6-16 voltios	6-20 voltios	5 voltios
Pines de entrada / salida digitales	14. 6 con salida PWM	54. 15 con salida PWM	14. 6 con salida PWM	54. 12 con salida PWM	54. 15 con salida PWM	20. 7 con salida PWM
Pines de entrada analógicos	6	16	6	12	16	12
Corriente DC por pin de entrada/salida	40mA	40mA	40mA	800mA	40mA	40mA
SRAM	2 KB de memoria flash	128 KB de memoria flash	2KB	96KB	256 KB de memoria flash	2,5 KB
EEPROM	1 KB	4 KB	1 KB	8 KB	4 KB	1 KB
Oscilación	16 MHz	16MHz	16 MHz	84 MHz	16MHz	16 MHz

Fuente: Elaboración Propia

2.8.3. MOTORES DC (CORRIENTE CONTÍNUA)



Ilustración 26. Motor DC con Caja Reductora

Fuente: Recuperado de <http://solorobotica.blogspot.com/2011/08/motores-de-corriente-continua.html>

Los motores DC o también denominados motores de corriente continua trabajan o se alimentan de corriente directa como la que proveen las pilas o las baterías de 9V, estos motores generalmente se usan para aplicaciones de velocidad variable donde el motor se alimenta con una fuente de voltaje variable, o para aplicaciones que requieran un control de posición, véase la *Ilustración 26*.

Estos motores DC cuentan con especificaciones y características que deben tomarse en cuenta al momento de adquirirlos, dentro de las características más importantes se tiene: voltaje de alimentación, corriente, torque y velocidad.

- **Voltaje:** es la energía eléctrica necesaria para mover el motor. Los motores DC trabajan generalmente dentro de los 1.6 y 12 voltios. Es importante conocer el voltaje de operación del motor DC debido a que al aplicar un voltaje demasiado bajo el motor DC no se moverá, por el contrario si se aplica un voltaje demasiado alto al especificado se generará calentamientos en el motor DC que ocasionaran daños permanentes.

- **Corriente:** especifica la cantidad de corriente que necesita el motor DC de la fuente de alimentación para su óptimo funcionamiento, la corriente que requiere el motor DC depende de la carga que tenga aplicada.

- **Velocidad de Giro:** se refiere al número de revoluciones que da el eje del motor DC en determinado intervalo de tiempo, se mide en rpm (revoluciones por minuto), la velocidad de giro depende la carga que tenga aplicada el motor.

- **Torque:** se refiere a la fuerza que puede ejercer un motor para mover o levantar un objeto a una distancia determinada, mientras mayor sea el torque del motor DC mayor será la carga que pueda mover.

La desventaja de los motores DC, es su alta velocidad de giro que oscila entre 2000 y 7000 rpm y su bajo torque. Este problema se soluciona se soluciona empleando cajas reductoras, que consiste en un sistema de engranajes conectados al eje del motor y que permiten disminuir la velocidad de giro y aumentar el torque del motor. Muchos motores ya vienen con su caja reductora y son conocidos como moto reductores. (Bravo, 2011)

2.8.4. MICRO MOTOR



Ilustración 27. Micro motor Solarbotics 150:1

Fuente: Recuperado de <https://solarbotics.com/product/gm13a/>

El micro motor Solarbotics GM13a es un motor DC miniatura con engranes metálicos con un eje de salida de 3 mm de diámetro, ideal para engranajes o llantas de 1/8 pulgadas, son motores específicos para reducción y con un gran torque, su relación de reducción es de 150:1 lo que lo hace ideal para movimiento de precisión y fuerza con un mínimo consumo de corriente. Su trabajo empieza con una alimentación de 3V y funciona hasta los 9V con una media de trabajo ideal de 6V, véase la *Ilustración 27*.

Sus principales características son:

Ratio 150:1

Dimensiones 34.0 x 12.0 x 10.1 mm

Weight 9.5 g

En la *Tabla 15* se muestra los valores de rendimiento del micro motor Solarbotics.

Tabla 15. Valores de rendimiento del micro motor

Voltaje (V)	RPM ²⁴ sin carga	Corriente sin carga (mA)
3	58	32
6	109	38

Fuente: Adaptado de <https://solarbotics.com/download.php?file=144>

2.8.5. SENSOR ULTRASÓNICO HC-SR04



Ilustración 28. Sensor Ultrasónico

Fuente: Recuperado de <http://elcajondeardu.blogspot.com/2014/03/tutorial-sensor-ultrasonidos-hc-sr04.html>

El sensor ultrasónico HC-SR04, ver la *Ilustración 28*, es un sensor de distancias por ultrasonidos capaz de detectar objetos y calcular la distancia a la que se encuentra en un rango de 2 a 450 cm. Su funcionamiento es tan simple como transmisión recepción, envía una ráfaga de pulsos por el transmisor y recibe la onda de retorno o eco por el receptor, enviar un pulso de arranque y mide la anchura del pulso que recibe. Su tamaño es pequeño y compacto,

²⁴ Revoluciones por minuto

ideal para cualquier aplicación y se complementa por su bajo consumo y muy alta precisión.
(ELECTRONILAB, 2013)

La *Ecuación (1)* describe como calcular la distancia del echo en centímetros usando el sensor ultrasónico HC-SR04.

$$distancia[cm] = (tiempo ECHO[micro segundos]) * \frac{velocidad\ del\ sonido\ [\frac{cm}{us}]}{2} \quad (1)$$

Donde la velocidad del sonido en el medio Aire es 343 m/s (0.034 cm/us)

Se detalla las características referentes al sensor ultrasónico tanto físico como funcionamiento:

- Dimensiones del módulos: 43 x 20 x 17 mm
- Voltaje de alimentación: 5 Vcc
- Frecuencia de trabajo: 40 KHz
- Rango máximo: 4.5 m
- Rango mínimo: 1.7 cm
- Duración mínima del pulso de disparo (nivel TTL): 10 μ s.
- Duración del pulso eco de salida (nivel TTL): 100-25000 μ s.
- Tiempo mínimo de espera entre una medida y el inicio de otra 20 ms.
- Pines de conexión: Vcc, Trig (Disparo de ultrasonido), Echo (Recepción de Ultrasonido), GND.
- Señal de salida: 5V Uno Lógico, 0V cero lógico.
- Medición de ángulo: 30°

En la *Ilustración 29* y la *Ilustración 30* se muestra el ángulo de medición del sensor ultrasónico y la onda de trabajo del sensor respectivamente.

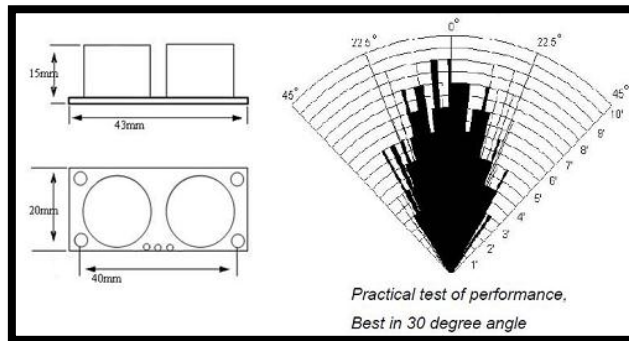


Ilustración 29. Ángulo de Medición

Fuente: Recuperado de <http://www.techmake.com/00029.html>

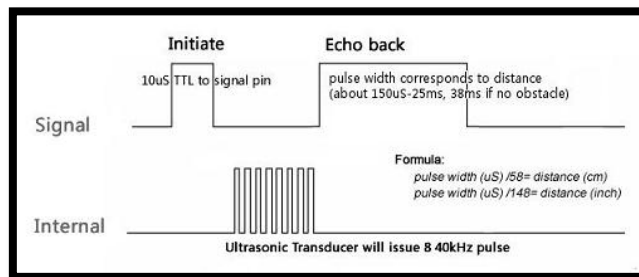


Ilustración 30. Onda de trabajo del Sensor

Fuente: Recuperado de <http://www.techmake.com/00029.html>

2.8.6. MÓDULO BLUETOOTH HC-05

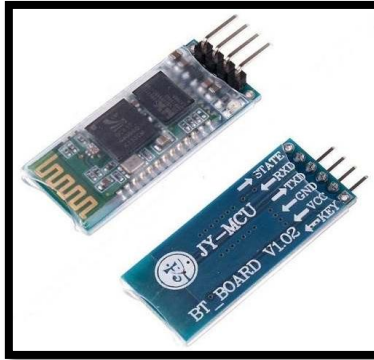


Ilustración 31. Módulo Bluetooth HC-05

Fuente: Recuperado de <http://blog.theinventorhouse.org/programando-el-bluetooth-hc-06-y-hc-05/>

El módulo Bluetooth HC-05 SPP (Protocolo de puerto serial) ver *Ilustración 31*, fue diseñado para la configuración de la conexión serial inalámbrica transparente y es una herramienta fácil de usar y configurar, se encuentra dentro de la versión 2 como Bluetooth V2.0 + EDR (Enhanced Data Rate) a una velocidad de 3 Mbps a una frecuencia de 2,4 Ghz y compatible con la versión 4.0. Utiliza un sistema Bluetooth RSE Bluecore 04-externa con la tecnología CMOS y con AFH (Adaptive Frequency Hopping Feature), su tamaño es muy pequeño y compacto con tan solo 12.7mm x 27mm. (Sabas, 2015)

Al ser un dispositivo configurable posee características de software como de hardware las cuales se detallan a continuación:

Características de Hardware

- Sensibilidad de -80 dBm

- Potencia de transmisión en Radio Frecuencia hasta 4dBm
- Rango de Operación de 1.8V a 3.6V
- Control de programación Entrada Salida
- Interface UART con velocidad de transmisión programable
- Con antena integrada
- Con conector en el borde

Características de Software

- Velocidad de Transferencia: 38400, Bits de Datos: 8, Bit de Parada: 1, Soporta Velocidades de transferencia de: 9600, 19200, 38400, 57600, 115200, 230400.
- Dado un pulso ascendente PIO0, se desconectará
- El puerto de instrucción de estado PIO1: bajo-desconectado, alto-conectado
- PIO10 y PIO11 pueden ser conectados al led rojo y azul por separado
- Auto conexión al último dispositivo encendido por defecto
- Auto pareo por defecto con el código pin 0000
- Auto reconexión en 30 segundos cuando se desconecta como resultado de una conexión fuera de rango.

2.8.7. CIRCUITO INTEGRADO L293D

El circuito integrado L293D de Texas Instruments permite controlar motores de corriente continua o bipolares de pasos (Bipolar stepping motors), capaz de conducir corrientes bidireccionales de hasta 1 amperio en el modelo L293 y hasta 600 mA en el modelo L293D con tensiones desde los 4.5V hasta los 36V, Las salidas tienen un diseño que permite

el manejo directo de cargas inductivas tales como relés, solenoides, motores de corriente continua y motores por pasos, ya que incorpora internamente los diodos de protección de contracorriente para cargas inductivas. (Manuel, 2012)

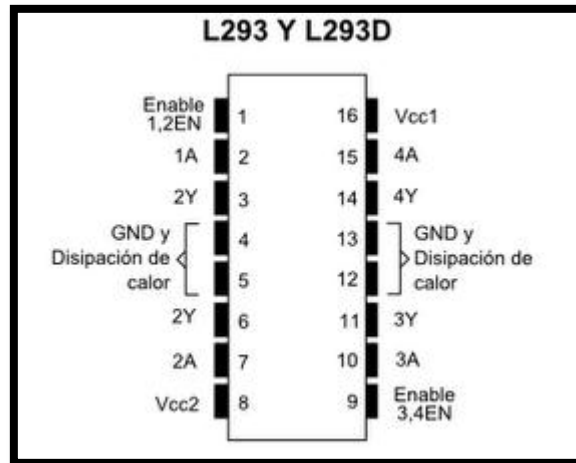


Ilustración 32. Distribución de pines del Circuito Integrado L293D

Fuente: Recuperado de <http://www.manuelvillasur.com/2012/10/driver-l293d-de-texas-instruments.html>

En la *Ilustración 32* se muestra la distribución de pines, este circuito integrado es ideal para trabajar con múltiples componentes electrónicos de carga inductiva como son los motores de corriente continua, ideal para una configuración puente en H como la requerida para este proyecto, se reduce el uso de pines al tener esta configuración como se ve en la *Ilustración 33*.

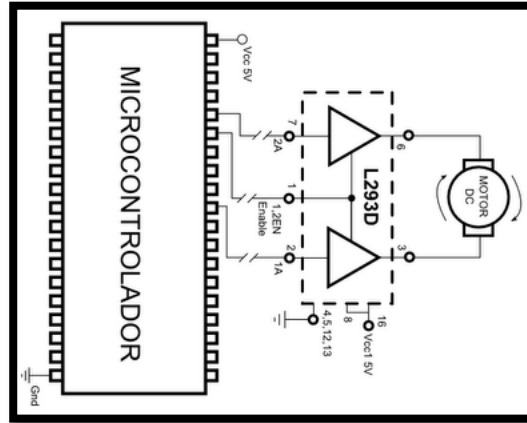


Ilustración 33. Configuración puente en H para un motor de Corriente Continua

Fuente: Recuperado de <http://www.manuelvillasur.com/2012/10/driver-l293d-de-texas-instruments.html>

2.8.8. BATERÍAS

El prototipo debe poseer autonomía energética para su funcionamiento constante por un considerable periodo de tiempo sin la necesidad de recargar las baterías. Para esto es necesario considerar el consumo de energía de los componentes electrónicos mencionados en el diagrama de bloques y que están incluidos en el prototipo.

En la *Tabla 16* se detalla el consumo de los componentes electrónicos dentro del prototipo.

Tabla 16. Consumo de corriente de los dispositivos electrónicos dentro del prototipo

COMPONENTE ELECTRÓNICO	CONSUMO MÁXIMO (mA)
Arduino Mini Pro	40
Micro Motor X2	76
Módulo Bluetooth	40
Sensor Ultrasónico	15
Led x3	60
Integrado L293D	17
TOTAL	248

Fuente: Elaboración Propia

Obtenido este dato se decide utilizar dos baterías de Ion de Litio debido a que son muy comunes en el mercado, el tiempo de recarga no es muy extenso, el costo es relativamente económico y nos permitirá una mayor autonomía, ver *Ilustración 34*.

En la *Ecuación (2)* se determina con valores obtenidos el tiempo de suministro de la batería hacia el prototipo en el caso de máximo consumo.

$$\text{duración de la batería} = \frac{\text{Capacidad de la batería [mAh]}}{\text{Corriente de carga [mA]}} * 0.7 \quad (2)$$

Donde 0.7 es una constante de factor de tolerancia a factores externos que pueden afectar el rendimiento de la batería.

$$\text{duración de la batería} = \frac{1600 \text{ [mAh]}}{248 \text{ [mA]}} * 0.7 = 4.5 \text{ [h]} \quad (3)$$

Tomando en cuenta el consumo máximo en 248 mA, las baterías permitirán una autonomía de 4.5 horas como se muestra en la *Ecuación (3)*, donde se puede considerar como tiempo mínimo debido a que el prototipo no estará trabajando siempre a máxima carga, la batería es recargable.



Ilustración 34. Batería recargable Ion Litio 3.7 v 800 mAh

Fuente: Recuperado de <http://spanish.phonebatteryreplacement.com/sale-1748181-3-7v-800mah-blackberry-battery-replacement-c-m2-li-ion-battery-for-blackberry-pearl-8100-8110-8120-8.html>

2.8.9. LIGHT EMITTER DIODE – LED

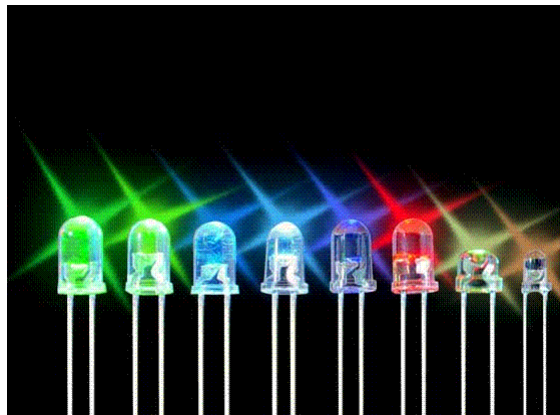


Ilustración 35. Diodo Emisor de Luz

Fuente: Recuperado de <http://www.vekell.com/>

Un diodo emisor de luz (LED), ver *Ilustración 35*, es un dispositivo electrónico semiconductor que emite luz visible cuando una corriente eléctrica pasa a través de este. La

luz no es prácticamente brillante, pero en la mayoría de los LED es monocromática, que se generan en una sola longitud de onda. La salida de un LED puede variar de rojo a una longitud de onda de aproximadamente 700 nanómetros a azul-violeta alrededor de 400 nanómetros. Algunos LEDs emiten energía infrarroja (IR) alrededor de 830 nanómetros o más; un dispositivo de estas características se conoce como un diodo de infrarrojos emisor de luz (IRED).

Un LED o IRED consta de dos elementos de material mesclado llamados semiconductores tipo P y semiconductores tipo N. Estos dos elementos se colocan en contacto directo, formando una región llamada la unión PN. En este sentido, el LED o IRED se asemeja a la mayoría de los otros tipos de diodos, pero hay diferencias importantes. El LED o IRED tiene un envase transparente, permitiendo luz visible o energía IR para pasar a través. Además, el LED o IRED tiene una amplia zona de unión PN cuya forma está adaptada a cada aplicación, además necesita de 0.7V para su operación. (Whatls, 2014)

Las características y beneficios positivos de los LEDs comparados con dispositivos de iluminación fluorescente son:

- Bajo consumo
- Alta eficiencia
- Larga vida

Algunas aplicaciones importantes:

- Luz indicadora o informativa
- Paneles LCD con retroalimentación
- Transmisión de datos por fibra óptica
- Control remoto

2.9. SOFTWARE

2.9.1. ENTORNO DE DESARROLLO INTEGRADO (IDE ARDUINO)

Arduino incluye un entorno de desarrollo integrado, ver *Ilustración 36*, que permite la programación fácil de la placa, el lenguaje de programación de Arduino se basa en C/C++, soporta los sistemas operativos Windows, Linux y Mac OS X los cuales se interconectan con la interfaz USB para subir el código del IDE a la placa Arduino. (Angela, 2012)

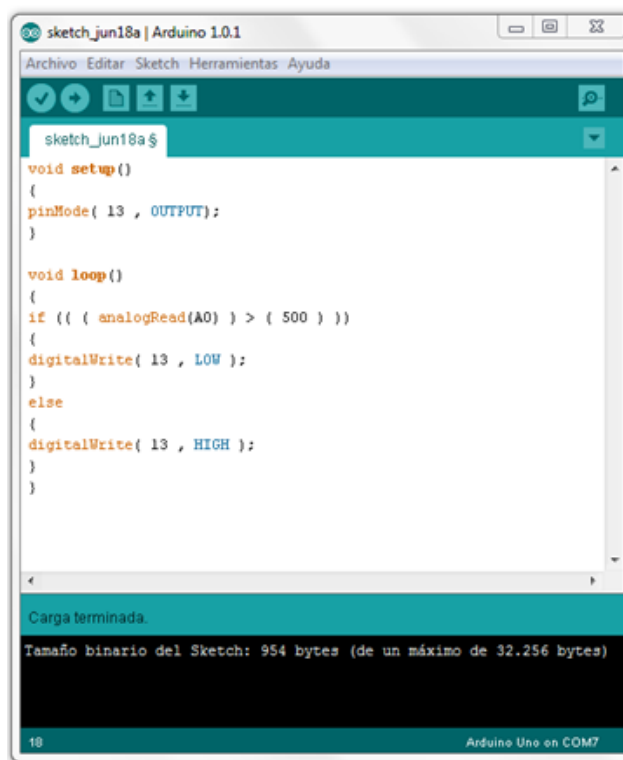


Ilustración 36. IDE de Arduino

Fuente: Recuperado de <http://solorobotica.blogspot.com/2012/07/programacion-de-arduino-elide-de.html>

2.9.2. EAGLE

Es un software editor de diseño de circuitos y placas electrónicas fácil de usar, dispone de una gran vitrina de componentes electrónicos que le permite al usuario desarrollar todos sus proyectos relacionados con la electrónica. (CadSoft EAGLE, 2015), ver logo en la *Ilustración 37*.



Ilustración 37. Ícono CadSoft EAGLE

Fuente: Recuperado de <http://www.cadsoftusa.com/eagle-pcb-design-software/about-eagle/>

2.9.3. ISIS PROTEUS

Es un entorno integrado diseñado para la implementación integral de proyectos de electrónica en todas sus etapas que son diseño, simulación, depuración y construcción. Proteus permite una gran ventaja para un diseñador de placas y componentes electrónicos puesto que permite las pruebas y corrección de errores antes de la implementación física de cualquier proyecto. Durante el desarrollo permite realizar simulaciones avanzadas acompañado de herramientas avanzadas. (HUBOR, 2015), ver logo en la *Ilustración 38*.



Ilustración 38. Logo Proteus

Fuente: Recuperado de <http://www.identi.li/index.php?topic=102287>

2.10. POSIBLES MATERIALES A UTILIZAR EN LA CARCASA DEL PROTOTIPO

Existen diversos materiales que servirán como carcasa o fachada del prototipo electrónico a implementar en este proyecto de tesis, se tomará en cuenta dos tipos de materiales para su evaluación, se detallan sus características, propiedades, durabilidad y sobre todo que no sean perjudiciales para los niños que son los beneficiarios que van a manipular el prototipo electrónico.

2.10.1. MADERA

La madera es la materia prima de origen vegetal más explotada por el hombre. Se encuentra en los árboles de tallo leñoso encontrando su parte más sólida debajo de la corteza del árbol, se utiliza para fabricar productos de gran utilidad como mesas, sillas y camas, muebles en general y en tecnología se usa para realizar muchos proyectos, la madera es un recurso renovable, abundante, orgánico, económico y con el cual es muy fácil de trabajar y moldear. (TecnologíaArea, 2013), ver las partes de la madera en la *Ilustración 39*.

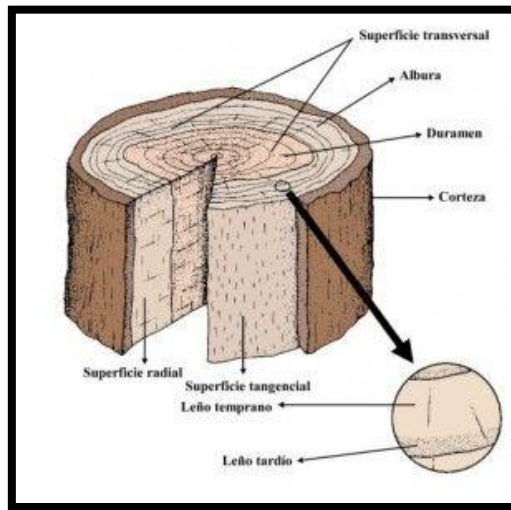


Ilustración 39. Partes de la madera

Fuente: Recuperado de <http://www.arqhys.com/contenidos/madera-propiedades.html>

2.10.1.1. COMPOSICIÓN DE LA MADERA

Está formada por fibras de celulosa, sustancia que conforma el esqueleto de los vegetales, y lignina, que le proporciona rigidez y dureza, por las fibras circulan y se adhieren sustancias como agua, resinas, aceites, sales y que en su composición están en mayoría elementos como el hidrógeno, el oxígeno, el carbono y el nitrógeno con cantidades menores de potasio, sodio, calcio, silicio y otros elementos, la madera se descompone gracias a microorganismos tales como bacterias, hongos o insectos, por tal razón es importante darle un tratamiento que evite su deterioro.

2.10.1.2. PROPIEDADES DE LA MADERA

Anisotropía. Es un material anisótropo, es decir no se comporta igual en todas las direcciones de las fibras. Es más fácil cepillar longitudinalmente al sentido de las fibras que transversalmente, y ocurre a la inversa con el aserrar.

Resistencia. La madera es uno de los materiales más idóneos para su trabajo a tracción, por su especial estructura direccional, su resistencia será máxima cuando la sollicitación sea paralela a la fibra y cuando sea perpendicular su resistencia disminuirá.

Flexibilidad. La madera puede ser curvada o doblada por medio de calor, humedad, o presión. Se dobla con más facilidad la madera joven que la vieja, la madera verde que la seca. Las maderas duras son menos flexibles que las blandas.

Dureza. Está relacionada directamente con la densidad, a mayor densidad mayor dureza. Si la humedad es elevada la dureza disminuye significativamente. Por el contrario si la madera se reseca, carece de humedad y se vuelve muy frágil.

Peso específico o densidad. Depende de su contenido de agua, la densidad absoluta viene determinada por la celulosa y sus derivados, su valor oscila alrededor de 1550 kg/m³, apenas varía de unas maderas a otras. La densidad aparente viene determinada por los poros que tiene la madera, que dependiendo de si están más o menos carentes de agua crece o disminuye la densidad.

Conductividad térmica. La madera seca contiene células diminutas de burbujas de aire, por lo que se comporta como aislante calorífico; el coeficiente λ vale 0,03 en sentido perpendicular a la fibra y vale 0.01 en sentido paralelo a la fibra. Lo cual quiere decir que su capacidad aislante es mayor en este último sentido. (ARQUITECTURA, 2013)

2.10.1.3. VENTAJAS DE LA MADERA

- Docilidad de Moldear o tallar
- Escasa densidad
- Belleza
- Calidad
- Resistencia mecánica
- Propiedades térmicas
- Propiedades acústicas

2.10.2. PLÁSTICO

Los plásticos son sustancias químicas pastas sintéticas denominados polímeros, de estructuras macro molecular que puede ser moldeada mediante calor o presión y cuyo componente principal es el carbono. Estos polímeros son grandes agrupaciones de monómeros unidos mediante un proceso químico llamado polimerización. Los plásticos proporcionan el balance necesario de propiedades que no pueden lograrse con otros materiales por ejemplo: color, poco peso, tacto agradable y resistencia a la degradación ambiental y biológica. (Technologie, 2012), ver *Ilustración 40*.



Ilustración 40. Plástico

Fuente: Recuperado de <http://www.spesa.com/productos/masterbatches/>

Los plásticos pueden adoptar múltiples formas y sus aplicaciones son casi infinitas. Para conseguir plásticos con aplicaciones específicas se le añaden aditivos como pigmentos, modificadores de impacto, agentes antiestáticos, etc. Por sus características, los plásticos han permitido innovar mejorando productos existentes y creando otros nuevos que optimizan nuestra calidad de vida y que minimizan el impacto ambiental. (REPSOL, 2014)

2.10.2.1. PROPIEDADES DEL PLÁSTICO

- Ligereza de peso
- Resistencia a la rotura
- Capacidad de aislamiento (eléctrico, térmico y acústico)
- Manejabilidad y seguridad
- Versatilidad
- Reciclabilidad
- Utilidad
- Sencillez y economía en su fabricación
- Impermeabilidad (humedad, luz, gases)

- No son conductores eléctricos

2.10.3. TABLA COMPARATIVA DE MATERIALES

En la *Tabla 17* se presenta una comparativa con las diferentes características principales de los materiales que pueden usarse para la carcasa del prototipo.

Tabla 17. Comparativa de Materiales

Características / material	Madera	Plástico
Origen	Natural	Químico
Conductividad eléctrica	Baja	Baja
Interferencia	Baja	Baja
Resistencia	Alta	Media
Flexibilidad	Media	Alta
Durabilidad	Alta	Media
Costo de tratamiento	Bajo	Medio
Peso	Alto	Ligero
Impermeabilidad	Media	Alta

Fuente: Elaboración Propia

CAPÍTULO 3. DESARROLLO DEL PROTOTIPO

El capítulo III empieza con una descripción general de todo el proyecto, enfocándose en las partes que lo conforman, como son: el prototipo electrónico, la aplicación móvil y las secuencias lógicas de programación. En lo que se refiere al prototipo electrónico se tomará a consideración todos los elementos que forman parte de este; en la parte física tangible y la programación en la parte lógica, para la aplicación móvil se definirá el software de respaldo, se explicará el diseño y se incluirá el código de programación empleado y en el caso de las secuencias lógicas de programación se explicará cada una de ellas enfatizando en el uso de las instrucciones que intervienen.

3.1. DESCRIPCIÓN GENERAL DEL PROYECTO

El proyecto consta básicamente de tres fases o etapas que se irán citando y detallando en el desarrollo del diseño del proyecto, la aplicación móvil que abarcará las secuencias básicas de programación en bloques para la enseñanza de los fundamentos de programación a los niños escolares de diez a doce años de edad, la comunicación entre la aplicación móvil y el prototipo electrónico y la implementación final del prototipo electrónico que aplicará las secuencias básicas de programación que son enviadas desde el celular y más específicamente de la aplicación móvil.

La aplicación móvil forma parte de lo intangible o software que se la realiza enfocándose en la programación en bloques, consta de seis instrucciones que cada una representa un bloque; avanzar, repetición, giro con graduación (90, 180 grados), y una toma de decisión, estas órdenes a su vez generan un carácter que se envía a través del módulo

bluetooth integrado en el teléfono celular usando la comunicación inalámbrica hacia el prototipo electrónico.

Las secuencias básicas de programación se diseñan tomando en cuenta los bloques que se dispone en la aplicación móvil, se diseñarán cinco secuencias enfocadas en la resolución de un problema o reto, y cada secuencia varia en dificultad acorde al reto que se le presente al escolar que se disponga a resolver.

El prototipo electrónico forma parte del hardware, recibe los caracteres enviados por la aplicación móvil que representan una orden o un bloque y los almacena en un vector para que ejecute todas y cada una de las órdenes que haya recibido, el prototipo electrónico se lo diseñará acorde a las necesidades de los niños en cuanto a tamaño, diseño vistoso y llamativo que cautive al escolar.

3.2. SELECCIÓN DEL MATERIAL A UTILIZAR EN LA CARCASA DEL PROTOTIPO

En el capítulo 2 se plantearon dos posibles materiales que se puede utilizar para la carcasa del prototipo electrónico para darle un acabado final y una adecuada presentación y que además se encuentran fácilmente en el mercado local; a continuación se presenta un análisis cualitativo resaltando sus principales propiedades y poder determinar cuál material es el más idóneo.

El plástico es un material de origen sintético que hace posible combinar elementos con el propósito de moldear de acuerdo a un requerimiento específico, tales como color deseado, densidad o si se requiere de alta durabilidad, aporta fácil manipulación con el fin de adoptar múltiples formas. Sus ventajas como son ligereza de peso, resistencia, impermeabilidad a la humedad y sus características dieléctricas lo hace un candidato para ser elegido para la carcasa del prototipo. Un aspecto negativo que no permite ser tomado en cuenta es que no se cuenta con maquinaria especializada para adecuar, mezclar y moldear los polímeros que se requiere para llegar al producto final.

La madera es un material de origen natural, utilizada en una gran variedad de artículos principalmente del hogar, es un recurso renovable, abundante, orgánico, económico y con el cual es muy fácil de trabajar y moldear. Posee múltiples ventajas como docilidad al manejo, flexibilidad, resistente, calidad y belleza propias del material obtenidas en el tallado, es un material dieléctrico.

Una vez hecho un breve análisis de los materiales se elige a la madera como material a usar en la carcasa de nuestro prototipo electrónico, sus propiedades, su disponibilidad en el mercado local y su fácil manejabilidad ha hecho que sea elegida ante el plástico.

3.3. SELECCIÓN DE LA PLATAFORMA PARA LA APLICACIÓN

En el anterior capítulo se detalló cuatro plataformas y entornos de desarrollo destinados al fomento del desarrollo de la lógica de programación en niños de diferentes edades, enfocados en la programación en bloques y que cada plataforma contiene un fuerte dentro de

su entorno, ya sea enfocándose en una edad específica, usando gráficos llamativos acorde a la edad a la que está destinado o con una sencilla interfaz de programación.

Scratch está dirigido a niños que deseen conocer, explorar y experimentar con los conceptos básicos de programación de computadores a través del uso de una interfaz gráfica amigable, facilita el aprendizaje autónomo en los niños despertando su curiosidad e inventiva, crea secuencias que controlan y combinan imágenes, animaciones, sonidos, e interactuar con el niño, identifica conceptos básicos de programación permitiendo secuencias pequeñas y compartirlas en la web. Code.org, con el objetivo de “difundir la programación como parte de la educación básica de los jóvenes”, enfoca sus recursos al aprendizaje de la informática y programación orientada a niños y a cualquier público que desee aprender a codificar, dispone de innumerables ejemplos, ejercicios de codificación y usa personajes animados conocidos del cine animado y la televisión. App Inventor desarrolla aplicaciones para dispositivos electrónicos Android basado en bloques, se enfoca en el aprendizaje escolar en beneficio de escuelas e institutos, es fácil de programar, posee una interfaz muy amigable y permite visualizar el desarrollo paso a paso y permite generar una aplicación para instalación. Blockly se orienta a la codificación, a las nociones fundamentales, se asemeja a las piezas de un rompecabezas y que se puede programar de una forma intuitiva y simple enfocándose en temas didácticos y edades tempranas introduciendo el concepto de programación visual con comandos muy simples.

Basado en los objetivos del trabajo de grado, en lo que se pretende desarrollar y en la comparación anterior se escoge a App Inventor como la plataforma para este proyecto, esta plataforma nos permite crear y diseñar aplicaciones móviles basadas en la programación por

bloques, es fácil de comprender para los niños con las edades a las que se ha enfocado y permite empaquetar todo el código en una aplicación instalable en dispositivos móviles.

3.4. SELECCIÓN DE LA PLATAFORMA ARDUINO

En el Capítulo II se dan a conocer las diferentes plataformas de Arduino que se encuentran disponibles en el mercado local, de las cuales se detalla las principales características que estas plataformas ofrecen para su uso e implementación en los diferentes proyectos dependiendo de la necesidad de cada uno de estos.

Para la implementación del Prototipo Electrónico programado mediante bloques se necesita una plataforma de Arduino relativamente básica, no se requiere demasiados pines digitales para las conexiones, además pequeña debido a la dimensión del prototipo y que cumpla con las funciones de recibir datos a través de sus pines de comunicación serial que previamente se habilitan para recibir los datos desde el módulo bluetooth.

La placa Arduino Mini Pro, ver *Ilustración 41*, es seleccionada para la implementación del Prototipo Electrónico por cumplir características básicas, pines limitados, tamaño muy pequeño que en conjunto la hace idónea para este proyecto.

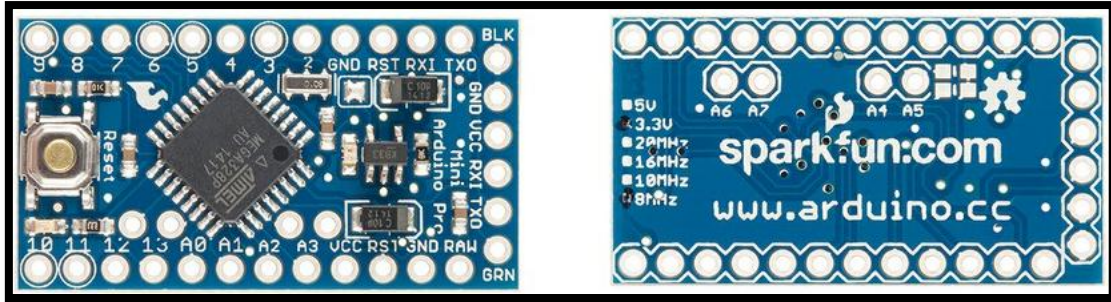


Ilustración 41. a) Vista anterior Arduino Mini PRO b) Vista posterior Arduino Mini PRO

Fuente: Recuperado de <https://www.arduino.cc/en/Main/ArduinoBoardProMini>

3.5. DISEÑO DEL PRIMER PROTOTIPO

Como parte del proyecto de titulación se procede a implementar un prototipo electrónico programado mediante bloques con una aplicación móvil para la enseñanza de secuencias básicas de programación a niños en etapa escolar comprendida entre los 10 y los 12 años, se empleará los conocimientos adquiridos en las aulas de clase a lo largo de nuestra carrera como también del material investigado a profesionales en este tema de la enseñanza y el aprendizaje. Se inicia con un detalle rápido en diagrama de bloques para concebir la idea del proyecto, luego más detalladamente se usará un diagrama de flujo para observar el funcionamiento de nuestro prototipo electrónico. Se diseñará las secuencias básicas de programación que serán empleadas por la aplicación móvil basada en Arquitectura Open Source (Código abierto), se procederá al armado del prototipo, se buscará mejoras a ese prototipo y se realizará las pruebas respectivas para dar respaldo y validez al trabajo de titulación.

3.5.1. DIAGRAMA DE BLOQUES

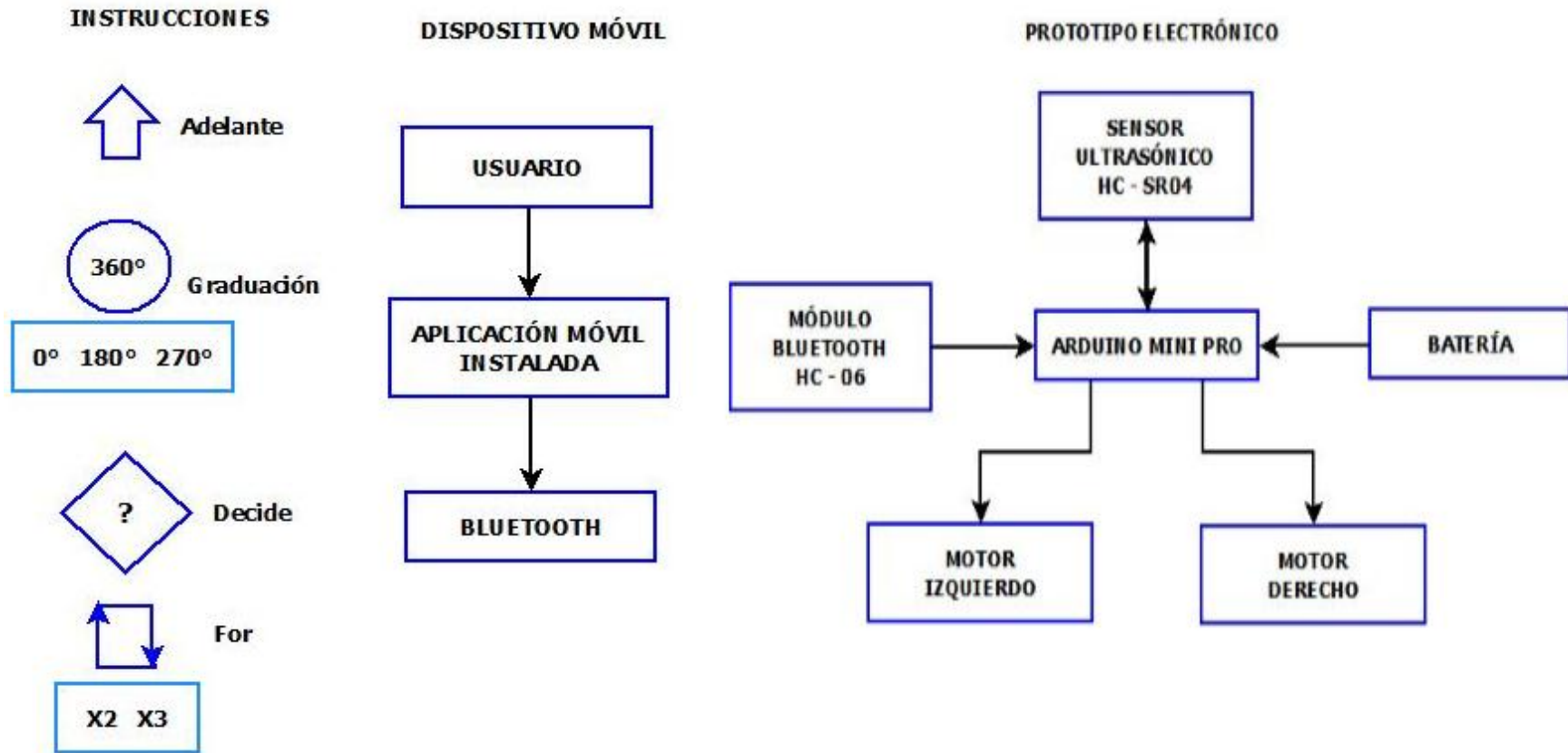


Ilustración 42. Diagrama de Bloques

Fuente: Elaboración Propia

En el diagrama de bloques de la *Ilustración 42* se observa tres grupos que son:

- **Instrucciones**, cuenta con cuatro tipos de instrucciones principales, Adelante, Graduación, Decide y For (Repetición). Del tipo graduación se derivan dos sub tipos que son 0° y 180°, del tipo For se derivan dos sub tipos de repetición X2 y X3.
- **Dispositivo Móvil**, cuenta con tres etapas, la etapa usuario es la que genera el requerimiento a través de las instrucciones soportadas por la aplicación móvil que es la segunda etapa y la tercera etapa donde los requerimientos del usuario se transmiten por Bluetooth hacia el prototipo electrónico.
- **Prototipo Electrónico**, funciona de manera centralizada, utiliza la plataforma Arduino como gestora principal del prototipo, inicia recibiendo requerimientos en forma de caracteres a través del módulo bluetooth integrado, esos caracteres pasan hacia el gestor central o Arduino Mini PRO y este a su vez usando recursos de Batería puede satisfacer los requerimientos recibidos por el módulo bluetooth HC-05, si el requerimiento es de tipo Decidir entonces usa el sensor ultrasónico para obtener más información y dar solución a los requerimientos recibidos por el módulo bluetooth y que a su vez son del usuario.

3.5.2. DIAGRAMA DE FLUJO

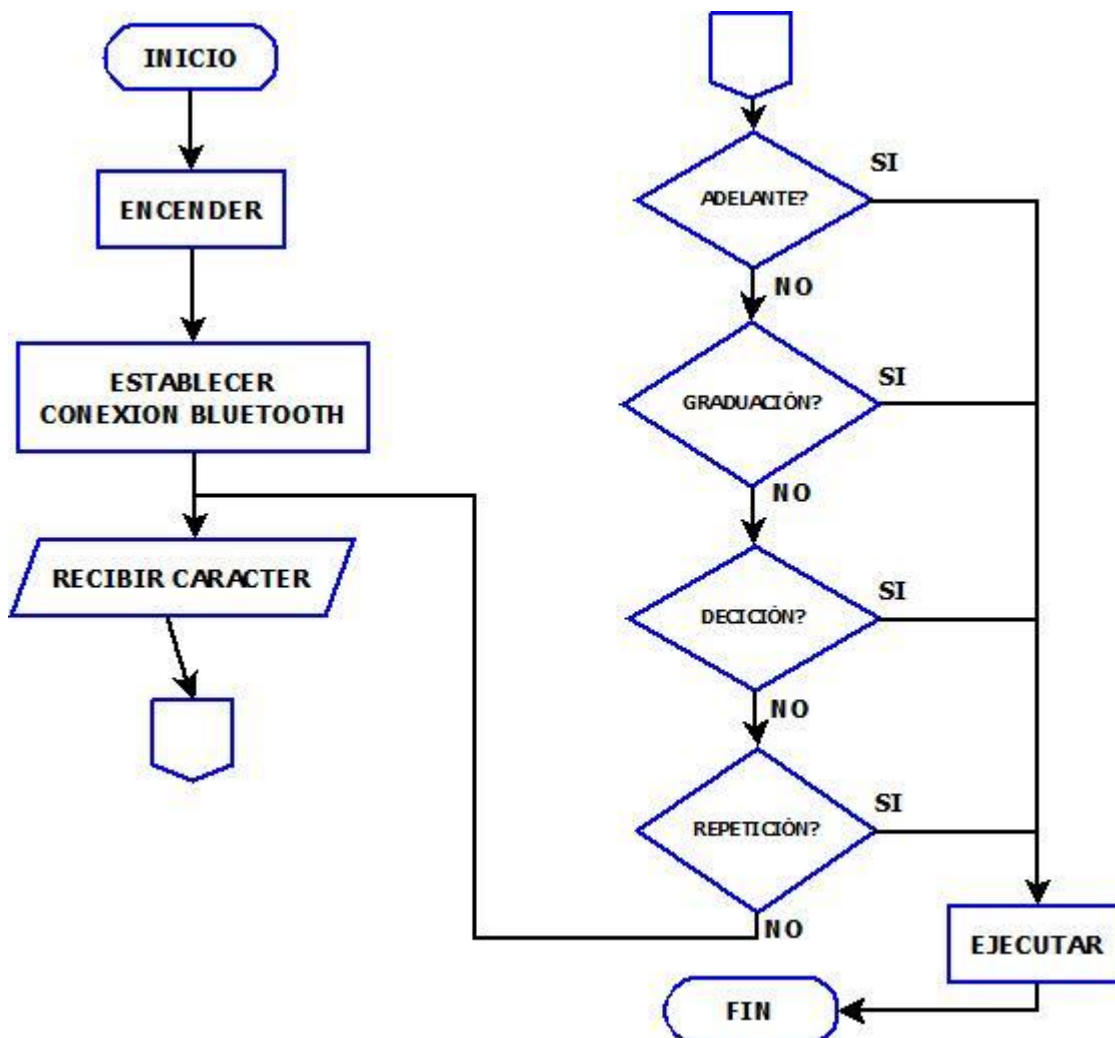


Ilustración 43. Diagrama de Flujo

Fuente: Elaboración Propia.

En el diagrama de flujo de la *Ilustración 43* se inicia encendiendo el sistema, una vez encendido y energizado todos los componentes, se procede a establecer la conexión inalámbrica a través de Bluetooth, una vez realizada la conexión se espera a recibir caracteres desde el dispositivo móvil, recibe, los compara y ejecuta la instrucción dependiendo de cuál sea.

Si recibe un carácter correspondiente a la instrucción Adelante compara en la condición correspondiente y simplemente ejecuta y termina el programa, si recibe caracteres correspondientes a Graduación compara en la condición de graduación y procede a ejecutar y termina el programa, si recibe un carácter correspondiente a decisión compara en la condición correspondiente y procede a usar recursos del sensor ultrasónico, calcula, ejecuta y termina el programa, si recibe caracteres de repetición compara en la condición de repetición y ejecuta las veces que sea según la condición y termina el programa.

3.5.3. SIMULACIÓN

Para la simulación del proyecto se emplea el Software Proteus versión 7.9, estas versiones actualizadas disponen de las librerías de las placas Arduino ya integradas, haciendo que se facilite la simulación con la mayoría de placas disponibles de Arduino.

En la *Ilustración 44* se observa que se ha simulado una placa Simulino Uno de Arduino, cuatro motores conectados a los pines digitales, que nos permitirá controlar el funcionamiento de los micro motores según las especificaciones que se reciben por los pines 0 y 1 de Recepción y Transmisión que en la simulación se lo hace con Comunicación Serial y en la realidad se conectará con el Módulo Bluetooth para la comunicación.

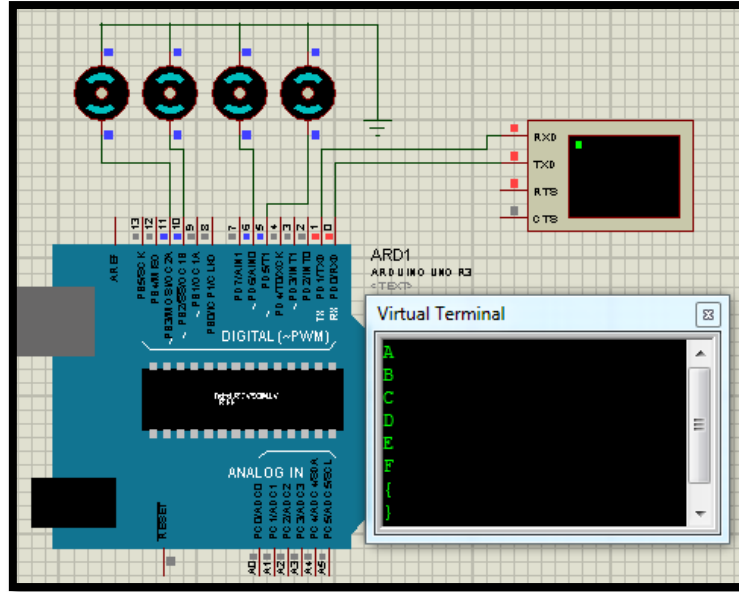


Ilustración 44. Simulación en Proteus, Primer prototipo

Fuente: Software de Simulación Proteus 7.9

3.5.4. DISEÑO E IMPRESIÓN DEL CIRCUITO PARA LA BAQUELITA

Realizada la simulación del circuito y verificado su funcionamiento se procede a realizar un esquema para el circuito impreso en el Software de diseño de diagramas y PCB²⁵s llamado Eagle (Easily Applicable Graphical Layout Editor), es donde se genera un primer plano de la pista para ser plasmada en la baquelita, este programa permite dibujar el esquema de pista, editar en caso de cambios y eliminar pistas en caso de requerirse, este programa brinda un apoyo integral en el campo de la electrónica debido a que integra todos una gran mayoría de componentes electrónicos digitalizados. Se selecciona los componentes del circuito que se deseen y sus correctas conexiones entre pines para que no se tenga problemas en el momento del soldado de los componentes y su funcionamiento.

²⁵ Printed Circuit Board – Placa de Circuito Impreso

Realizado esto se obtienen las pistas impresas en modo Mirror o espejo con el objetivo de que no se invierta el sentido de la pista, se imprime en papel transfer (termo transferible) que es óptimo para este tipo de trabajos con baquelitas, en la *Ilustración 45* se muestra el resultado de este proceso.

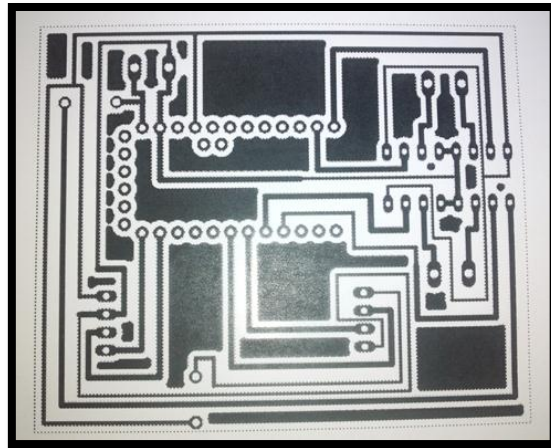


Ilustración 45. Impresión del Diseño de Circuito Impreso

Fuente: Elaboración Propia

Una vez impresa la pista en papel se procede a planchar la pista impresa sobre la baquelita, este proceso toma de entre 15 a 30 minutos dependiendo del calor al que se lo exponga, seguido se procede a bañar a la baquelita ya plasmada en una mezcla de agua y ácido férrico en un recipiente de plástico para deshacerse del residuo de cobre que no se requiere en la baquelita y que solo quedará el cobre correspondiente a la pista plasmada de la impresión. En la *Ilustración 46* se muestra este proceso.

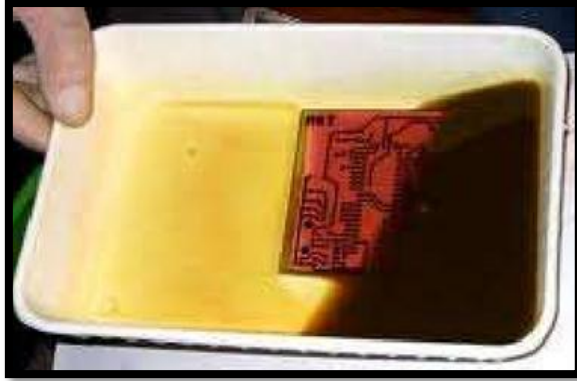


Ilustración 46. Baquelita en Ácido Férrico

Fuente: Elaboración Propia

Culminado este proceso se verifica que todo esté en orden, que todo el cobre excedente haya sido retirado, que las pistas estén correctas y finalmente se procede a un lijado con lija de agua para darle un buen acabado, se lava en agua limpia y se lo seca para obtener una baquelita lista con sus pistas impresas sobre esta, ver *Ilustración 47*.

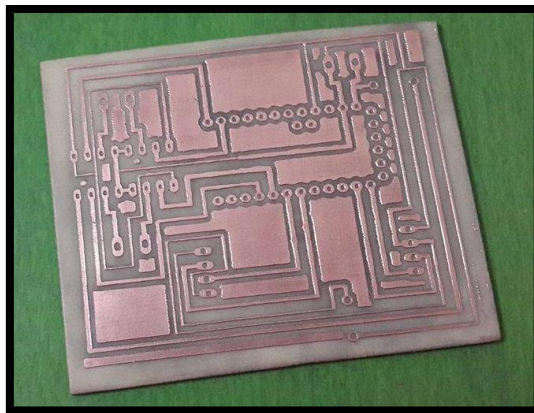


Ilustración 47. Baquelita terminada

Fuente: Elaboración Propia.

3.5.5. ELECCIÓN Y CONFIGURACIÓN DE LA TECNOLOGÍA INALÁMBRICA

3.5.5.1. *ELECCIÓN DE LA TECNOLOGÍA INALÁMBRICA*

En el capítulo anterior se detalla tres tecnologías inalámbricas que permitirán la comunicación desde la aplicación móvil hacia el Prototipo Electrónico, estas tecnologías presentadas presentan características importantes que son fuertes dentro de su campo y de su protocolo, en este proyecto se verá cual tecnología es conveniente aplicar dependiendo de las necesidades y requerimientos del proyecto, a continuación se realiza un análisis cualitativo de cada una de estas tecnologías para luego determinar cuál es la adecuada.

Wi – Fi es una tecnología de comunicación inalámbrica que permite velocidades de transmisión elevadas, trabaja en la banda 2.4 GHz y 5 GHz, requiere un punto de acceso para la conectividad, usado mayormente por computadoras personales y teléfonos de alta gama, posee una cobertura considerable y el coste de implementación es elevado comparado con otras tecnologías, el estándar que usa requiere una demanda de ancho de banda superior y por ende mayor consumo de energía, a pesar de ser una tecnología muy usada y muy eficiente se prescinde de esta en el proyecto.

ZigBee es una tecnología que también opera en la banda de 2,4 GHz, manipula volúmenes de datos bajos y medianos, es una solución ideal para crear redes flexibles y robustas para aplicaciones de control y monitoreo con sensores. Tiene un bajo consumo de energía, fácil de configurar y manipular pero que difiere de este proyecto debido a su coste, disponibilidad de módulos en el mercado y porque no es una tecnología que se haya

incorporado a los dispositivos móviles o Smartphone como lo ha hecho bluetooth y que es de gran utilidad para la aplicación móvil que interviene en el proyecto.

Bluetooth también opera en la banda de frecuencia de 2.4 Ghz que es una banda no licenciada, su operatividad se ha beneficiado por corto alcance y el volumen bajo de datos que maneja, muchos periféricos trabajan con esta tecnología para la comunicación con computadoras personales, su consumo de energía es bajo por lo que se ha integrado en laptops, Smartphone y otros dispositivos y que lo hace ideal para nuestro proyecto por su coste bajo, su disponibilidad en el mercado y principalmente porque está integrado en los teléfonos móviles que nos es de gran utilidad para la aplicación móvil.

Luego de hacer una descripción cualitativa a las tres tecnologías, mostrando sus usos más comunes, su funcionamiento y sus aplicaciones se toma en cuenta a la tecnología Bluetooth para la comunicación entre nuestro Prototipo Electrónico y la aplicación móvil instalada en el celular.

3.5.5.2. CONFIGURACIÓN DE LA TECNOLOGÍA INALÁMBRICA

Para poder conectar la aplicación móvil con el prototipo electrónico se debe sincronizar el módulo integrado en el Smartphone con el módulo bluetooth hc-05 integrado en el prototipo, a continuación se especifica paso a paso este procedimiento.

Ubicarse en la pantalla principal del Smartphone, deslizar el dedo desde la parte superior de la pantalla hacia abajo y aparecerá una pantalla en tono gris como se muestra en la *Ilustración 48*, dar clic en el icono de la parte superior derecha.

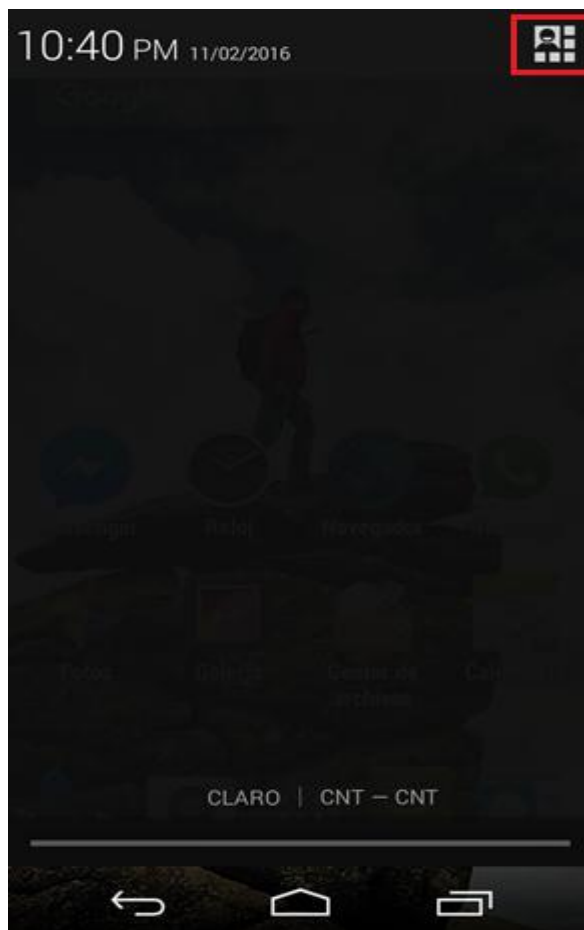


Ilustración 48. Sincronización de bluetooth 1

Fuente: Captura de Smartphone Sky 6.0Q.

En el menu que se despliega seleccionar el icono de bluetooth, notar que se encuentra en marca de agua, significa que el bluetooth está desactivado, dar clic para activarlo. Ver *Ilustración 49*.

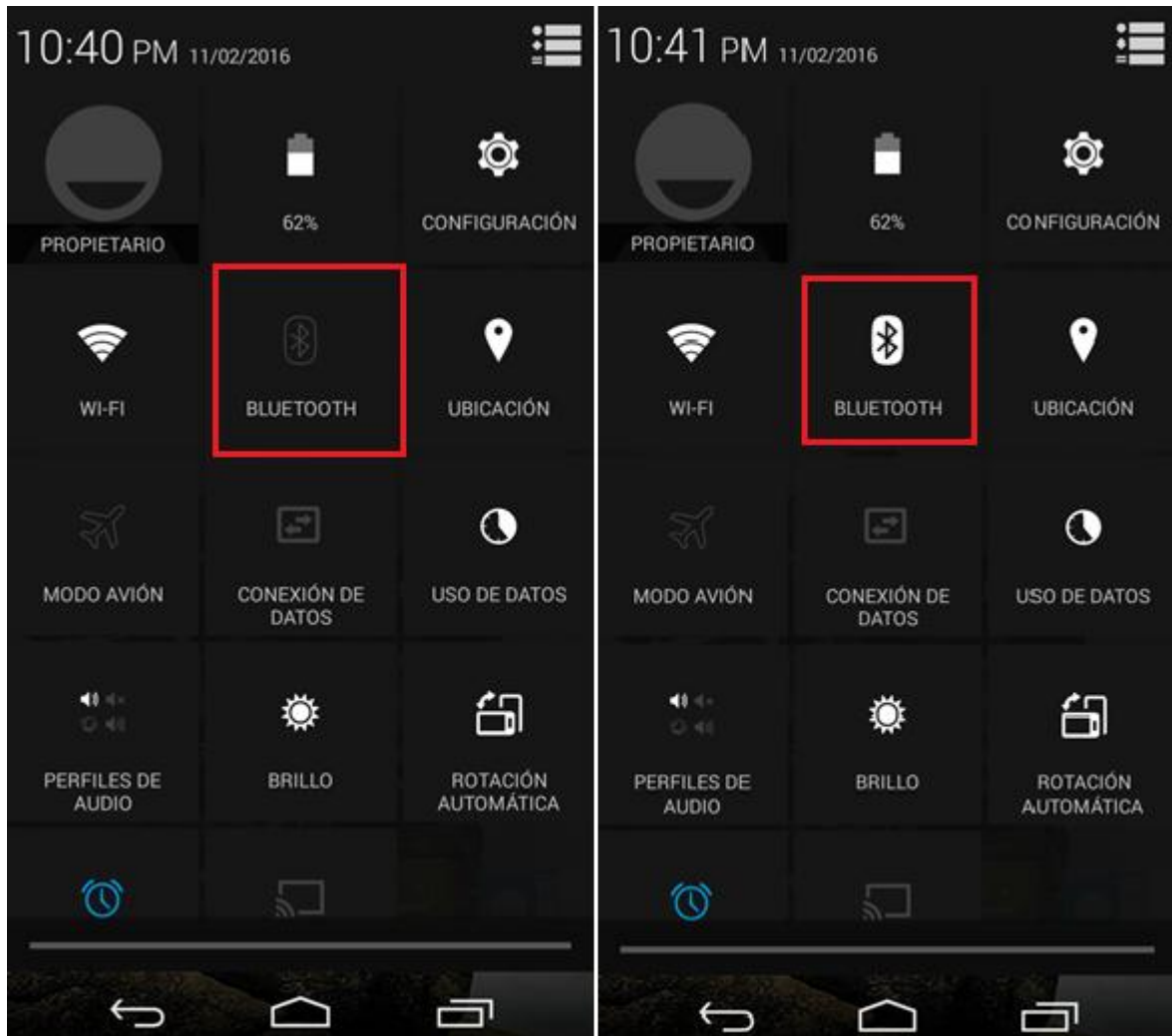


Ilustración 49. Activación de bluetooth

Fuente: Captura de Smartphone Sky 6.0Q.

Una vez activado el bluetooth mantener presionado el icono de bluetooth y se despliega una pantalla donde nos muestra los dispositivos sincronizados y sin sincronizar, nótese que el nuevo módulo bluetooth aparece visible en el listado que se muestra en la *Ilustración 50*.



Ilustración 50. Dispositivos disponibles

Fuente: Captura de Smartphone Sky 6.0Q.

Una vez en esta pantalla dar clic en el dispositivo descubierto, en este caso es HC-05 que es el nombre del bluetooth, pedirá que ingrese el código de PIN del dispositivo que se quiere sincronizar, para este efecto el pin por defecto es “0000” o “1234”, para este módulo el PIN es “1234”, digitar el código y clic en aceptar como se muestra en la *Ilustración 51*.

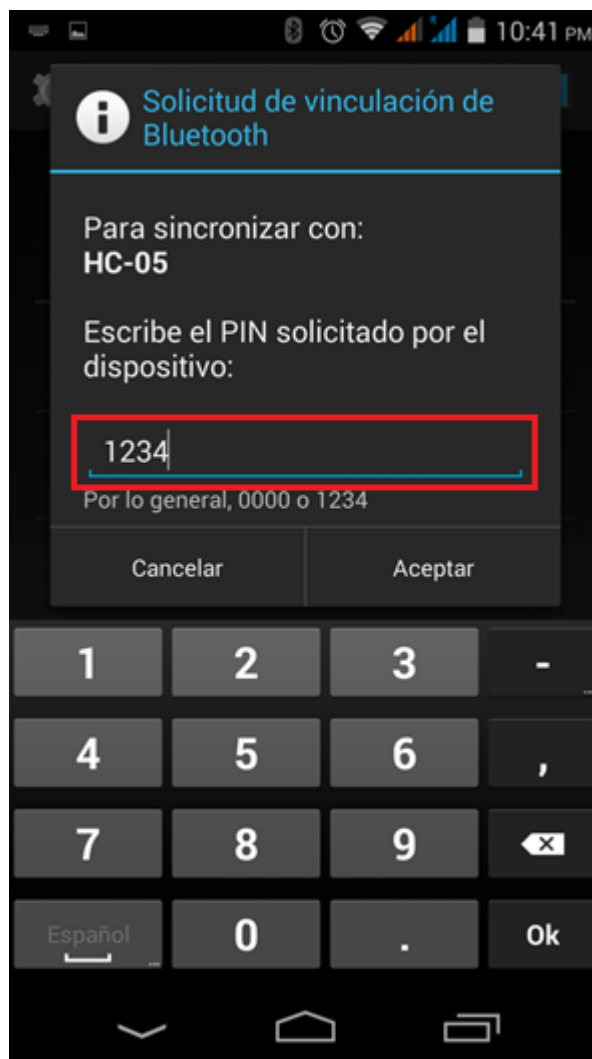


Ilustración 51. Ingreso de PIN

Fuente: Captura de Smartphone Sky 6.0Q.

Una vez validado el código PIN esta lista la sincronización, se muestra nuevamente el listado con los dispositivos compatibles con la tecnología bluetooth y el dispositivo que se acabó de sincronizar aparece como sincronizado y listo para ser conectado como se muestra en la *Ilustración 52*.



Ilustración 52. Dispositivo sincronizado

Fuente: Captura de Smartphone Sky 6.0Q.

3.5.6. DISEÑO Y PROGRAMACIÓN DE LA APLICACIÓN

En la *Ilustración 53* se muestra la base de diseño de la aplicación móvil, se encuentra estructurada por conjunto de elementos o tablas, los botones se agrupan en una tabla, las imágenes se agrupan en otra tabla diferente y los botones principales que son los comandos se agrupan en otra tabla distinta.



Ilustración 53. Plataforma de diseño de MIT App Inventor

Fuente: Recuperado de <http://ai2.appinventor.mit.edu/#4848183070425088>

- **Paleta:** Dispone de todos los elementos que pueden ser usados en la aplicación, se hallan cuadros de textos, botones, etiquetas, dibujos como también elementos especiales ocultos como son acelerómetros, bases de datos, cámara de video y Bluetooth.
- **Pantalla:** Muestra una cercana apariencia con la aplicación que se realiza, abarca todos los elementos posibles de la Paleta.
- **Componentes:** Visualiza de manera organizada todos los elementos que hayan sido arrastrados desde la paleta hacia la pantalla.
- **Propiedades:** Permite configurar las características que un elemento tiene como fuente, alto, ancho, color, porcentaje de distribución dependiendo de cada elemento.



Ilustración 54. Base de diseño de la Aplicación Móvil

Fuente: Pantalla de Diseño de App Inventor.

Dentro de cada tabla, ver *Ilustración 55*, se muestra agrupados los diferentes botones usados y demás elementos requeridos en la aplicación móvil.

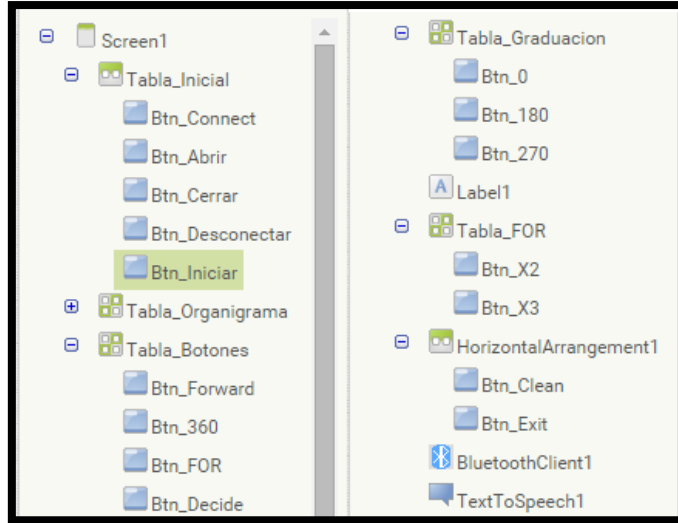


Ilustración 55. Botones y elementos requeridos

Fuente: Recuperado de <http://ai2.appinventor.mit.edu/#4848183070425088>

En las siguientes ilustraciones se muestra la programación en bloques de cada uno de los botones que se usan en esta aplicación móvil.

El botón Btn_conectar establece la conexión y comunicación Bluetooth entre el dispositivo móvil y el módulo integrado en el prototipo electrónico, ver la *Ilustración 56*.

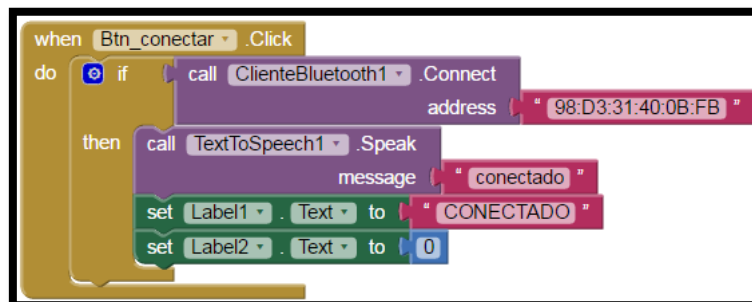


Ilustración 56. Botón Btn_conectar

Fuente: Recuperado de <http://ai2.appinventor.mit.edu/#4848183070425088>

El botón Btn_desconectar termina la conexión Bluetooth entre el dispositivo móvil y el módulo integrado en el prototipo electrónico, ver la *Ilustración 57*.

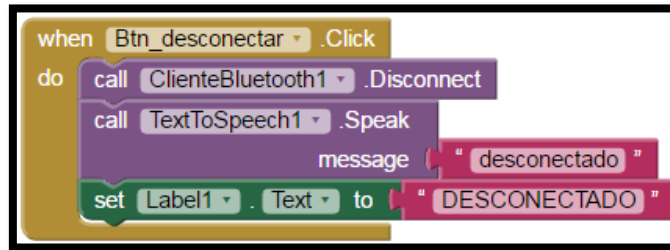


Ilustración 57. Botón Btn_desconectar

Fuente: Recuperado de <http://ai2.appinventor.mit.edu/#4848183070425088>

El botón Btn_play envía el carácter “0” hacia el bluetooth del prototipo electrónico, su función es la de iniciar la compilación o demostración en el prototipo una vez se hayan enviado todas las instrucciones desde la aplicación móvil, ver la *Ilustración 58*.

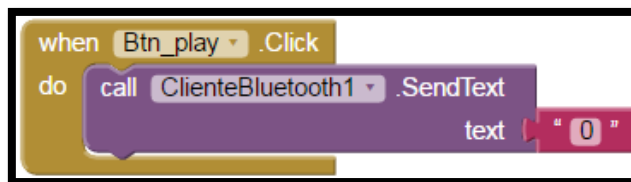


Ilustración 58. Botón Btn_play

Fuente: Recuperado de <http://ai2.appinventor.mit.edu/#4848183070425088>

El botón Btn_adelante envía el carácter “A” hacia el bluetooth del prototipo electrónico, su función es la de establecer la acción de avanzar de frente una unidad de distancia con el prototipo, ver la *Ilustración 59*.

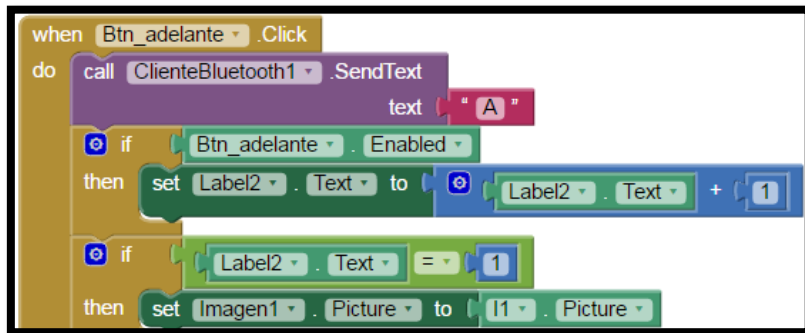


Ilustración 59. Botón Btn_adelante

Fuente: Recuperado de <http://ai2.appinventor.mit.edu/#4848183070425088>

El botón Btn_360 habilita la tabla Tabla_graduacion para permitir elegir entre dos posibles opciones de 0° y 180° para que se ejecute en el prototipo, ver la Ilustración 60.

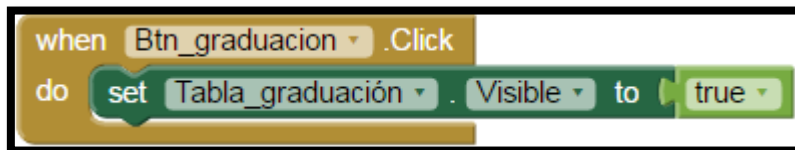


Ilustración 60. Botón Btn_graduación

Fuente: Recuperado de <http://ai2.appinventor.mit.edu/#4848183070425088>

Los botones Btn_0 y Btn_180 son consecuencia del botón Btn_graduación, se habilitan al dar clic en el botón Btn_graduación, su función es la de enviar un carácter sesteado en cada botón para establecer la acción de girar 0° o 180° con el prototipo, ver la Ilustración 61.

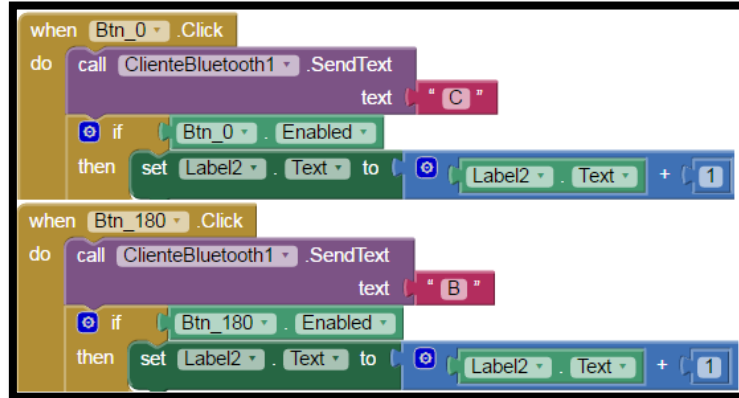


Ilustración 61. Botones: a) 0°, b) 180°

Fuente: Recuperado de <http://ai2.appinventor.mit.edu/#4848183070425088>

El botón Btn_decisión envía el carácter “F” hacia el bluetooth del prototipo electrónico, su función es la de establecer la acción de Decidir cuando el prototipo encuentre un obstáculo frente a él, si girar a la derecha, girar a la izquierda o retroceder, ver la *Ilustración 62*.

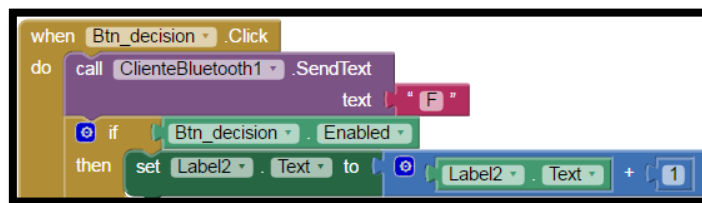


Ilustración 62. Botón Btn_decisión

Fuente: Recuperado de <http://ai2.appinventor.mit.edu/#4848183070425088>

El botón Btn_repetición habilita la tabla Tabla_repetición para permitir elegir entre dos posibles opciones de repetición, X2 o X3 para que se ejecute en el prototipo la repetición dos o tres veces, ver la *Ilustración 63*.



Ilustración 63. Botón Btn_repetición

Fuente: Recuperado de <http://ai2.appinventor.mit.edu/#4848183070425088>

Los botones Btn_X2 y Btn_X3 son consecuencia del botón Btn_repetición, se habilitan al dar clic en el botón Btn_repetición, su función es la de enviar un carácter seteado en cada botón para establecer la acción de repetir X2 o X3 las acciones deseadas, ver la *Ilustración 64*.

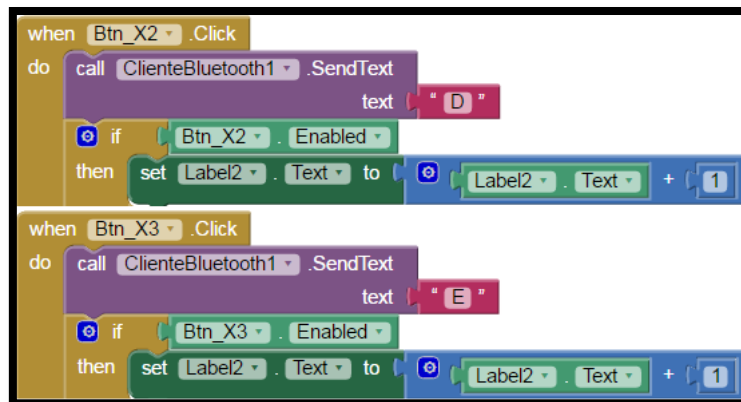


Ilustración 64. Botones: a) Btn_X2, b) Btn_X3

Fuente: Recuperado de <http://ai2.appinventor.mit.edu/#4848183070425088>

Los botones Btn_abrir y Btn_cerrar permiten enviar un carácter respectivamente que le indicará al prototipo que se abre o se cierra un conjunto de instrucciones, estos botones acompañarán a las instrucciones de Repetición y de Decisión, ver la *Ilustración 65*.

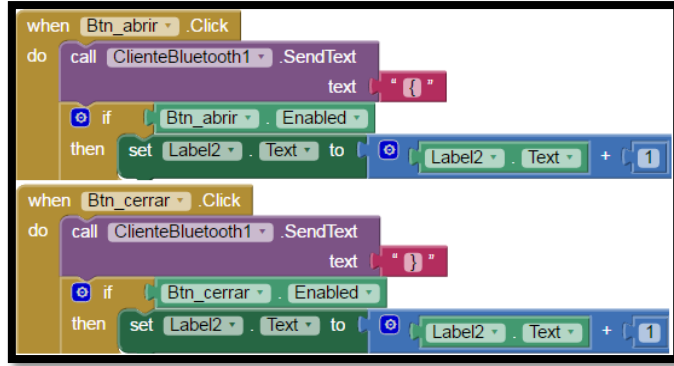


Ilustración 65. a) Btn_Abrir, b) Btn_Cerrar

Fuente: Recuperado de <http://ai2.appinventor.mit.edu/#4848183070425088>

3.5.7. IMPLEMENTACIÓN DEL PRIMER PROTOTIPO

En este ítem se muestra los pasos a seguir para el armado de los elementos que intervienen en el prototipo, la suelta y conexiones necesarias para su correcto funcionamiento.

Primero se procede a asegurar las llantas con cada uno de los motores DC, procurando no girar el motor mientras se realiza este procedimiento, se verifica que los motores queden correctamente asegurados, ver ilustración 66.



Ilustración 66. Motor y Llanta

Fuente: Elaboración Propia.

Luego se procede a asegurar la rueda loca, los motores con sus llantas al chasis, se verifica que esté correctamente ajustado al chasis para evitar inconvenientes posteriores, ver

Ilustración 67.

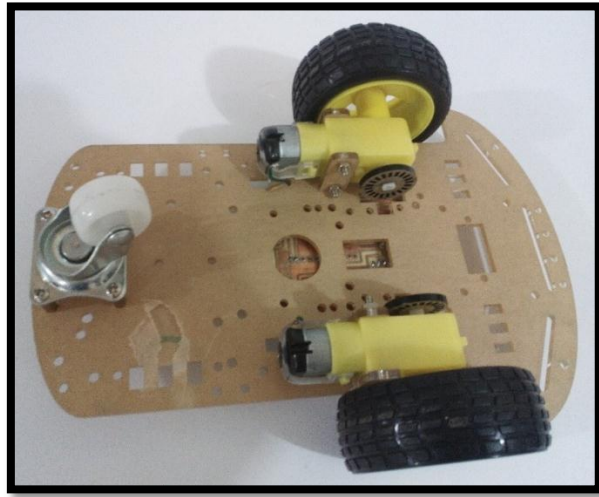


Ilustración 67. Componentes fijados al Chasis

Fuente: Elaboración Propia.

Ahora se toma la baquelita previamente preparada como se detalló anteriormente para proceder a perforar los hoyos necesarios, luego se procede a colocar espadines, borneras y el zócalo para el circuito integrado L293D, se procede a soldar cada uno de estos elementos sobre la placa, ver *Ilustración 68.*

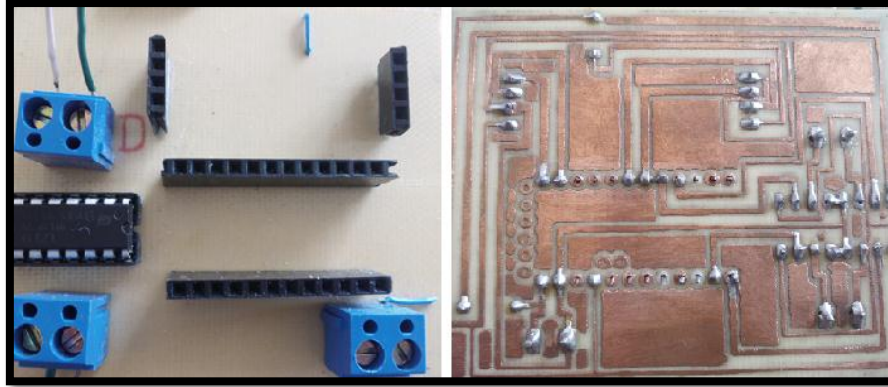


Ilustración 68. a) Elementos sobre la placa b) Soldadura

Fuente: Elaboración Propia.

Además se coloca el Arduino Mini PRO sobre su base, el circuito integrado sobre el zócalo, el módulo Bluetooth y el sensor ultrasónico sobre sus espadines, ver *Ilustración 69*.

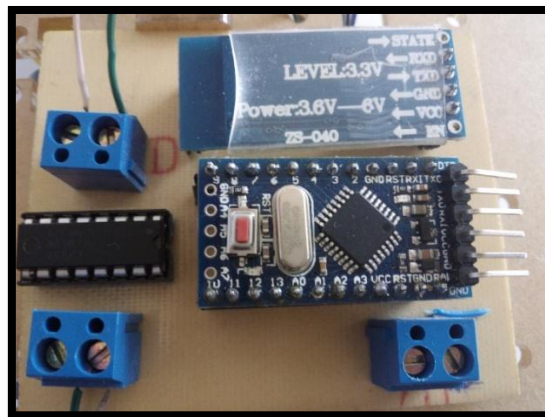


Ilustración 69. Módulos instalados

Fuente: Elaboración Propia.

Y finalmente se integran todas las partes en un solo conjunto, la placa con sus elementos se fija en el chasis, los motores se conectan a la placa y se le añade y conecta las baterías, ver *Ilustración 70*.

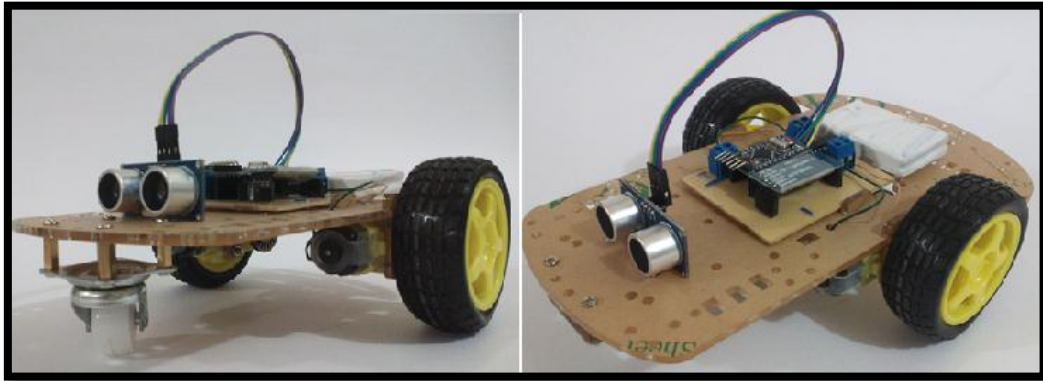


Ilustración 70. Primer prototipo

Fuente: Elaboración Propia.

En la *Ilustración 71* se muestra el primer prototipo con el complemento en la presentación final.

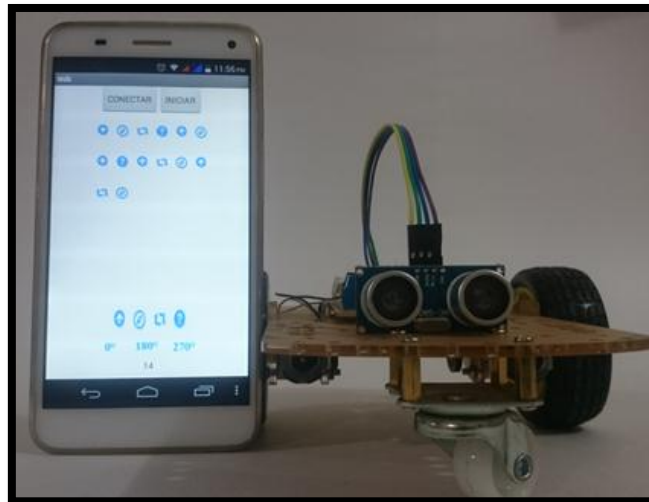


Ilustración 71. Primer prototipo Completo

Fuente: Elaboración Propia.

3.6. DISEÑO DE LAS SECUENCIAS DE PROGRAMACIÓN

Las secuencias se desarrollan de tal manera que involucren todas las instrucciones que han sido planteadas y empleadas dentro de la aplicación móvil, estas instrucciones han sido preestablecidas respondiendo a la necesidad de la lógica de programación, tales como Avanzar, Giro a la Derecha (0°), giro a la Izquierda (180°), repeticiones de las instrucciones anteriores y decisiones que pueda tomar el prototipo electrónico en caso de encontrar un obstáculo en su camino, en la *Ilustración 72* se indica las instrucciones.



Ilustración 72. Instrucciones de la Aplicación

Fuente: Software de edición de imágenes de Microsoft.

3.6.1. PRIMERA SECUENCIA

El objetivo es llegar hacia la bandera a cuadros.

Para que esta secuencia funcione correctamente lo que se realiza primero es: conectarse al bluetooth del prototipo electrónico dando clic en el ícono del logo de bluetooth. Seguido

ubicar la tabla de instrucciones y seleccionar tres veces el botón Adelante y por ultimo clic en Iniciar. En la *Ilustración 73* se ilustra la explicación.

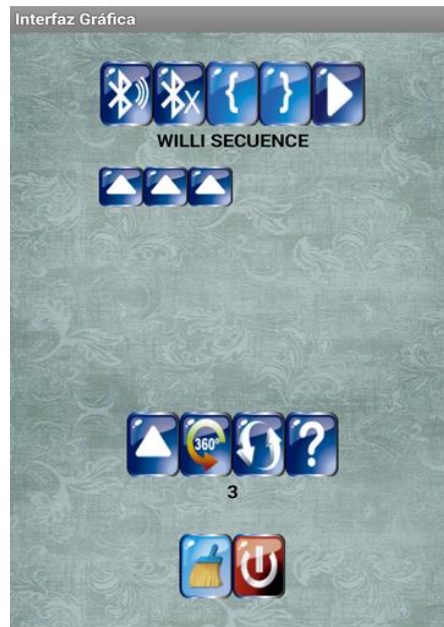


Ilustración 73. Indicaciones Secuencia 1

Fuente: <http://ai2.appinventor.mit.edu/#4848183070425088>

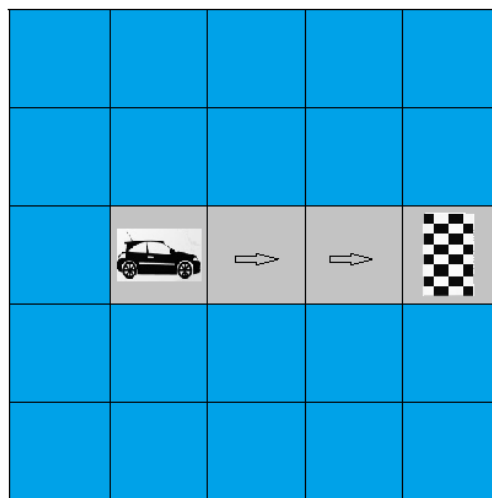


Ilustración 74. Secuencia Avanzar.

Fuente: Software de edición de imágenes de Microsoft.

Una vez dado clic en Iniciar el prototipo electrónico recorrerá tres veces una distancia hasta llegar a su destino que es la bandera a cuadros, ver la *Ilustración 74*.

3.6.2. SEGUNDA SECUENCIA

El objetivo es llegar hacia la bandera a cuadros.

Para que esta secuencia funcione correctamente lo que se realiza primero es: conectarse al bluetooth del prototipo electrónico dando clic en el ícono del logo de bluetooth. Seguido ubicar la tabla de instrucciones y seleccionar dos veces el botón Adelante, seleccionar el botón graduación y se habilitan dos opciones, tomar la opción 0°, una vez más el botón Adelante y por ultimo clic en Iniciar. En la *Ilustración 75* se indica la explicación.

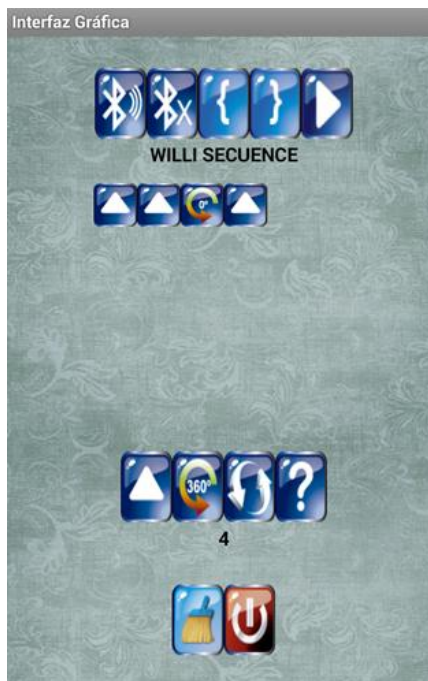


Ilustración 75. Indicaciones secuencia 2

Fuente: <http://ai2.appinventor.mit.edu/#4848183070425088>

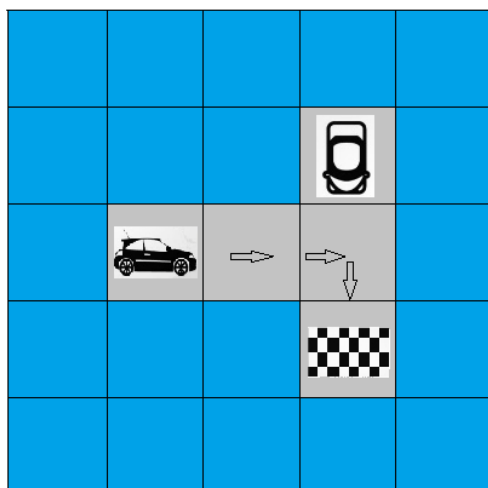


Ilustración 76. Secuencia Giro

Fuente: Software de edición de imágenes de Microsoft.

Una vez dado clic en Iniciar el prototipo electrónico recorrerá dos veces una distancia, girará hacia la derecha según la graduación seteada y avanzará una vez una distancia hasta llegar a su destino que es la bandera a cuadros, ver la *Ilustración 76*.

3.6.3. TERCERA SECUENCIA

El objetivo es llegar hacia la bandera a cuadros.

Para que esta secuencia funcione correctamente lo que se realiza primero es: conectarse al bluetooth del prototipo electrónico dando clic en el ícono del logo de bluetooth. Seguido ubicar la tabla de instrucciones y seleccionar botón Repetición, una vez dado clic en Repetición se habilita un menú y seleccionar X3, seguido clic en la llave abierta para incluir las secuencias dentro del conjunto, para este caso se elige Adelante, cerrar el conjunto, y por ultimo clic en Iniciar. En la *Ilustración 77* se indica la explicación.

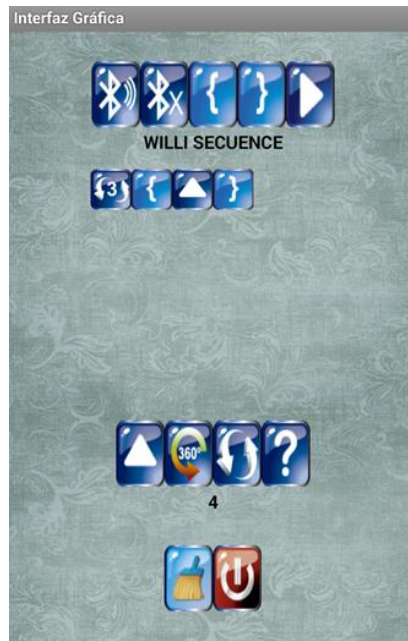


Ilustración 77. Indicaciones secuencia 3

Fuente: <http://ai2.appinventor.mit.edu/#4848183070425088>

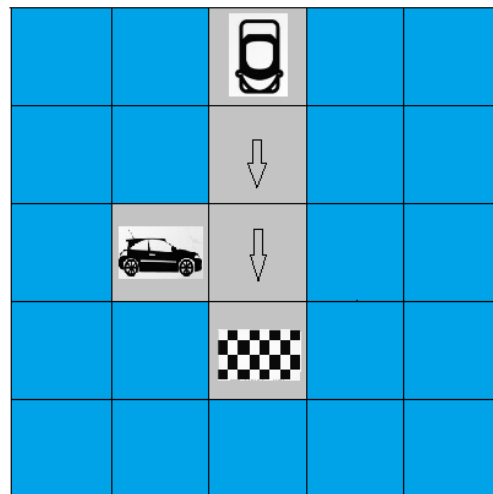


Ilustración 78. Secuencia Repetición

Fuente: Software de edición de imágenes de Microsoft.

Una vez dado clic en Iniciar el prototipo electrónico reconocerá la instrucción de repetición e identifica el contenido del conjunto y ejecutará el contenido del conjunto, en este

caso es Adelante por lo que repetirá tres veces una distancia y habrá llegado a su destino que es la bandera a cuadros, ver la *Ilustración 78*.

3.6.4. CUARTA SECUENCIA

El objetivo es llegar hacia la bandera a cuadros.

Para que esta secuencia funcione correctamente lo que se realiza primero es: conectarse al bluetooth del prototipo electrónico dando clic en el ícono del logo de bluetooth. Seguido ubicar la tabla de instrucciones y seleccionar botón Adelante, luego dar clic en Graduación y seleccionar 0° en el menú que se despliega, una vez hecho esto se usa la secuencia número tres que es repetición, y por ultimo clic en Iniciar. En la *Ilustración 79* se indica la explicación.

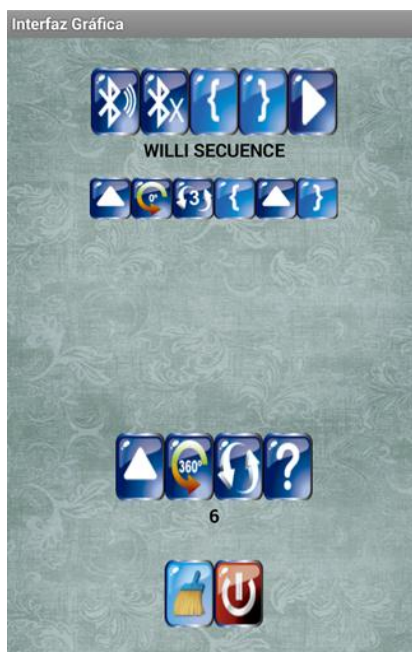


Ilustración 79. Instrucciones Secuencia 4

Fuente: <http://ai2.appinventor.mit.edu/#4848183070425088>

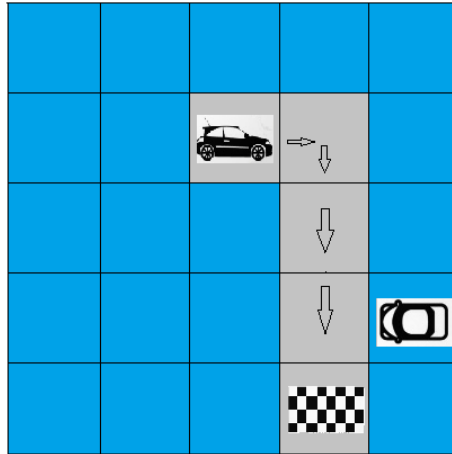


Ilustración 80. Secuencia mixta Giro - Repetición

Fuente: Software de edición de imágenes de Microsoft.

Una vez dado clic en Iniciar el prototipo electrónico avanzará una distancia, girará hacia la derecha y reconocerá la instrucción de repetición e identifica el contenido del conjunto y ejecutará el contenido del conjunto, en este caso es Adelante por lo que repetirá tres veces una distancia y habrá llegado a su destino que es la bandera a cuadros, ver la *Ilustración 80*.

3.6.5. QUINTA SECUENCIA

El objetivo es llegar hacia la bandera a cuadros.

Para que esta secuencia funcione correctamente lo que se realiza primero es: conectarse al bluetooth del prototipo electrónico dando clic en el ícono del logo de bluetooth.

Seguido ubicar la tabla de instrucciones y seleccionar botón Decisión, clic en llave abierta, clic en el botón Graduación y se selecciona 180°, seguido clic en llave cerrada. A continuación seleccionando el botón Repetición, en el submenú que se habilita escoger X3,

seleccionar abrir conjunto, seleccionar Adelante y cerrar el conjunto. Seguido clic en Graduación, del submenú se selecciona 0°, luego avanzar y por ultimo clic en Iniciar. En la *Ilustración 81* se indica la explicación.

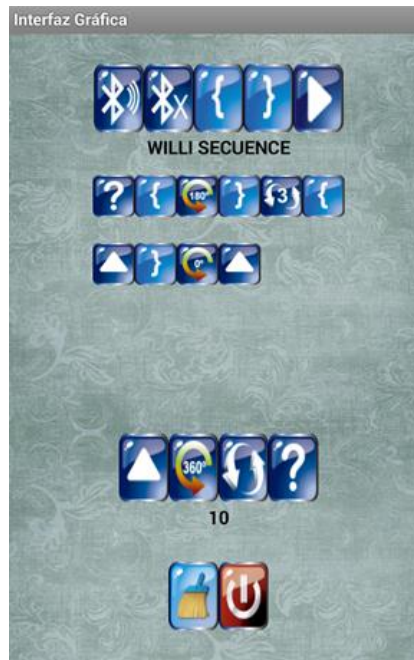


Ilustración 81. Instrucciones secuencia 5

Fuente: <http://ai2.appinventor.mit.edu/#4848183070425088>

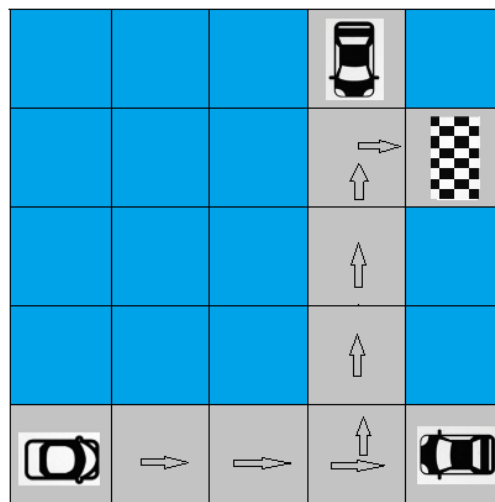


Ilustración 82. Secuencia Decisión

Fuente: Software de edición de imágenes de Microsoft.

Una vez dado clic en Iniciar el prototipo electrónico reconocerá la instrucción de repetición e identifica el contenido del conjunto y ejecutará el contenido del conjunto, en este caso es Adelante por lo que repetirá tres veces una distancia, encontrará un obstáculo por lo que decidirá ir a la izquierda, terminada esa instrucción reconocerá la instrucción de repetición e identifica el contenido del conjunto y ejecutará el contenido del conjunto, en este caso es Adelante por lo que repetirá tres veces una distancia, encontrará un obstáculo nuevamente por lo que decidirá ir a la derecha y avanzar una distancia y habrá llegado a su destino que es la bandera a cuadros, ver *Ilustración 82*.

3.6.6. SECUENCIA ADICIONAL

El objetivo es llegar hacia la bandera a cuadros.

Para que esta secuencia funcione correctamente lo que se realiza primero es: conectarse al bluetooth del prototipo electrónico dando clic en el ícono del logo de bluetooth. Seguido ubicar la tabla de instrucciones y seleccionar botón Repetición, en el submenú que se habilita escoger X3, seleccionar abrir conjunto, seleccionar Adelante, 180°, adelante, 0° y cerrar el conjunto, luego dar clic en adelante y por ultimo clic en Iniciar. En la *Ilustración 83* se indica la explicación.

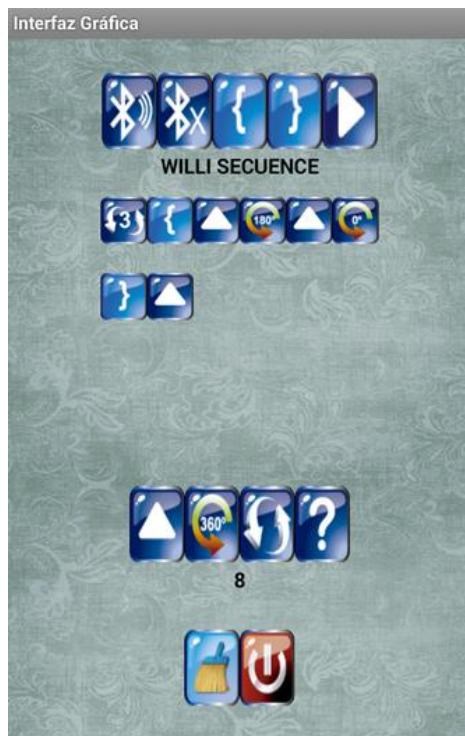


Ilustración 83. Instrucciones secuencia Adicional

Fuente: <http://ai2.appinventor.mit.edu/#4848183070425088>

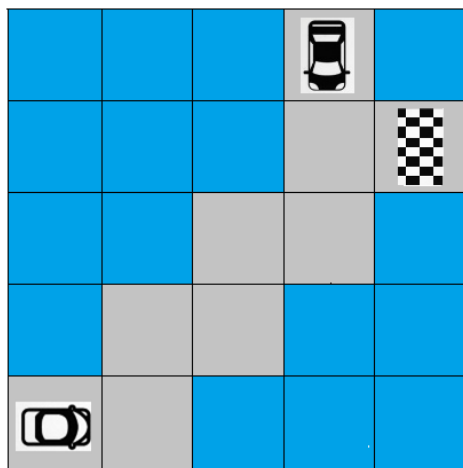


Ilustración 84. Secuencia Combinada

Fuente: Software de edición de imágenes de Microsoft.

Una vez dado clic en Iniciar el prototipo electrónico reconocerá la instrucción de repetición e identifica el contenido del conjunto y ejecutará el contenido del conjunto, en este caso es Adelante, giro a la izquierda, adelante, giro a la derecha por lo que repetirá tres veces las cuatro instrucciones dentro del conjunto repetición, terminado esta repetición avanzará una distancia hacia a delante y habrá llegado a su destino que es la bandera a cuadros, ver *Ilustración 84*.

3.7. CONSIDERACIONES DEL PRIMER PROTOTIPO



Ilustración 85. Primer Prototipo Implementado

Fuente: Elaboración Propia.

- El primer prototipo, ver la *Ilustración 85*, nace como una verificación rápida de la factibilidad del proyecto, se realiza pruebas de funcionamiento principalmente de conexión entre el bluetooth del Smartphone con el módulo Bluetooth integrado en la placa del prototipo.

- Se verifica rápidamente que los componentes funcionen tales como los motores, el sensor ultrasónico, el módulo Bluetooth, que el Arduino trabaje correctamente sin que se cuelgue o presente fallas.
- El aspecto físico como la presentación y el tamaño son no adecuados para los escolares que es donde se orienta este prototipo.
- La aplicación móvil requiere mejorar en la presentación, en el fondo, los íconos y cuestiones de diseño.
- Se ve la necesidad de construir una carcasa para el prototipo con un tema acorde a los escolares de edades de diez a doce años.

3.8. DISEÑO FINAL

A lo largo del desarrollo de este proyecto se ha visto la necesidad y la factibilidad de mejorar el diseño antes presentado, aspectos como tamaño, se puede reducir puesto que el tamaño anterior es muy grande para las manos de un niño de 10 o 12 años, llantas, si el tamaño reduce también reduce el tamaño de las llantas, en este aspecto las llantas han sido consideradas con orugas porque mejora la adherencia evitando patinar como se vio en el prototipo anterior, con las orugas el prototipo puede avanzar por caminos irregulares como el césped y arena sin problemas de que se caiga a las irregularidades del terreno, se cambia a baterías reemplazando las pilas que anteriormente energizaban el prototipo además se cambia los motores por micro motores que son más pequeños.

3.7.1 MEJORAS AL DISEÑO ANTERIOR

En la *Ilustración 86* se muestra los elementos que reemplazan a los elementos del diseño anterior, las llantas tienen un tamaño reducido, los motores reducen su tamaño y se adiciona las orugas.



Ilustración 86. Nuevos elementos considerados

Fuente: Elaboración Propia.

3.8.1.1. ARMADO DEL NUEVO PROTOTIPO

Previo al armado se ha procedido a desarmar el prototipo anterior para extraer los elementos importantes como la placa impresa y soldada, el bluetooth, el sensor ultrasónico, en general toda la parte electrónica.

Se toma los elementos del nuevo prototipo, las llantas nuevas, el chasis nuevo, los micromotores nuevos y se los arma como se ve en la *Ilustración 87*.

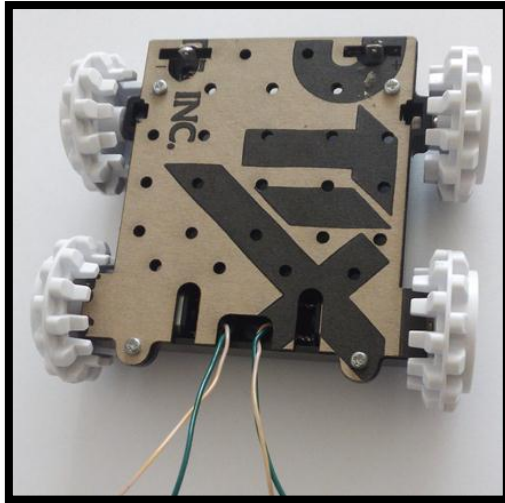


Ilustración 87. Chasis, Lantas y micro motores

Fuente: Elaboración Propia.

Seguido se toma los elementos retirados del chasis del prototipo anterior y se los asegura sobre el nuevo chasis, verificando que todo esté en orden como se ve en la *Ilustración 88*.

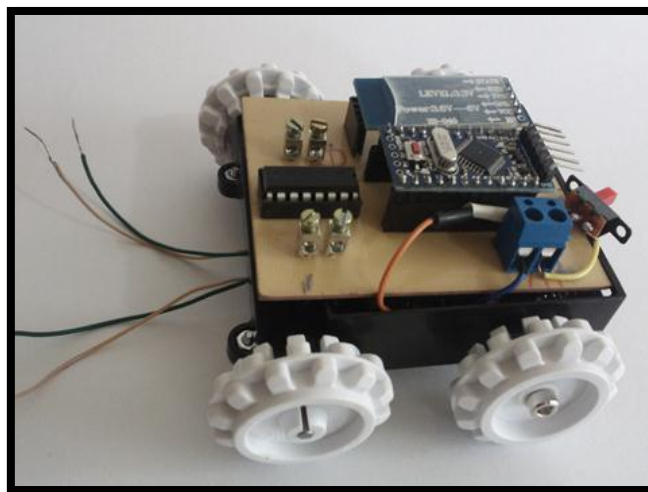


Ilustración 88. Placa asegurada al nuevo chasis

Fuente: Elaboración Propia.

Este procedimiento es más rápido que el procedimiento anterior, puesto que los componentes electrónicos ya están listos, solo se tiene que pasar de un chasis a otro.

Y finalmente se colocan las orugas y el la carcasa temporal que se la diseñado para darle personalidad al prototipo como se ve en la *Ilustración 89*.

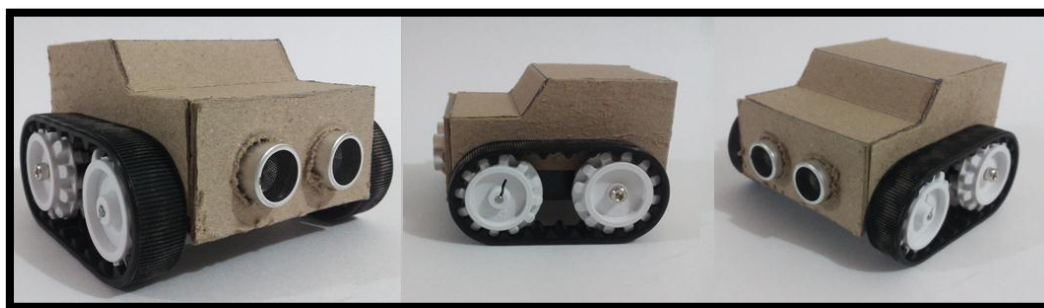


Ilustración 89. Diseño del segundo prototipo

Fuente: Elaboración Propia.

3.8.1.2. TABLA DE MEJORAS AL DISEÑO ANTERIOR

En la *Tabla 18* se presenta una comparativa de las mejoras que se ha hecho al diseño preliminar, lo que se ha considerado que se puede mejorar para prototipo final.

Tabla 18. Comparativa entre prototipos

	Diseño Preliminar	Diseño Final
Chasis	Grande	Pequeño
Ruedas	Caucho	Orugas
Fuente de Alimentación	Pilas de Carbón AA	Baterías recargables Ion de Litio
Diseño de la Aplicación	Básico	Mejorado
Tamaño	Grande	Pequeño y compacto
Carcasa	No tiene	Carcasa de madera
Motores	Motores DC	Micro Motores DC
Color	No tiene	Verde militar
Presentación	Mala	Buena

Fuente: Elaboración Propia.

3.9. CÁLCULO DE CORRIENTE, VOLTAJE Y POTENCIA

3.9.1. CORRIENTE

Para el cálculo de la corriente que se requiere tomamos como punto de partida un dato obtenido en la *Tabla 16*, donde se consideró el consumo de corriente de cada uno de los elementos electrónicos dentro del prototipo. El consumo total es de 248 mAh de corriente.

3.9.2. VOLTAJE

El voltaje a utilizar se establece luego de haber establecido los componentes q se involucran en el proyecto, sus rangos de operación y sus voltajes recomendados de trabajo, se determina trabajar con un voltaje común q no afecte el rendimiento de los diferentes dispositivos, en el caso de este proyecto debemos trabajar con un mínimo de 6 voltios que es el voltaje de operación de los micro motores y el resto de componentes operan con 5 voltios recomendados por lo que la incidencia sobre estos es mínima.

Tomando como referencia la *Ilustración 34*, se ha usado un voltaje de operación de 7.4 voltios provistos por dos baterías recargables de 3.7 voltios.

3.9.3. POTENCIA

Una vez establecidos los valores de corriente y voltaje se calcula el valor de potencia consumida, para esto nos valemos de la fórmula de la potencia mostrada en la Ecuación (4).

$$P = V * I \quad (4)$$

Donde P es potencia, V es voltaje e I es corriente.

Reemplazando los valores establecidos en los subtemas 3.9.1 y 3.9.2 en la Ecuación (4) se tiene:

$$P = 7.4v * 248mAh$$

$$P = 1.84 W$$

El prototipo requerirá de 1.84 watts de potencia para su funcionamiento.

3.10. PRUEBAS

Se desarrollarán pruebas tanto de funcionamiento del prototipo en el encendido como de las secuencias lógicas de programación visualizadas conforme se desplaza el prototipo.

3.10.1. PRUEBA DE ENCENDIDO

Como se ha explicado anteriormente en el diseño de las secuencias lógicas de programación el prototipo tiene que encenderse previamente antes de establecer la conexión con el módulo bluetooth, es por ello q es muy importante constatar que el encendido se dé adecuadamente.

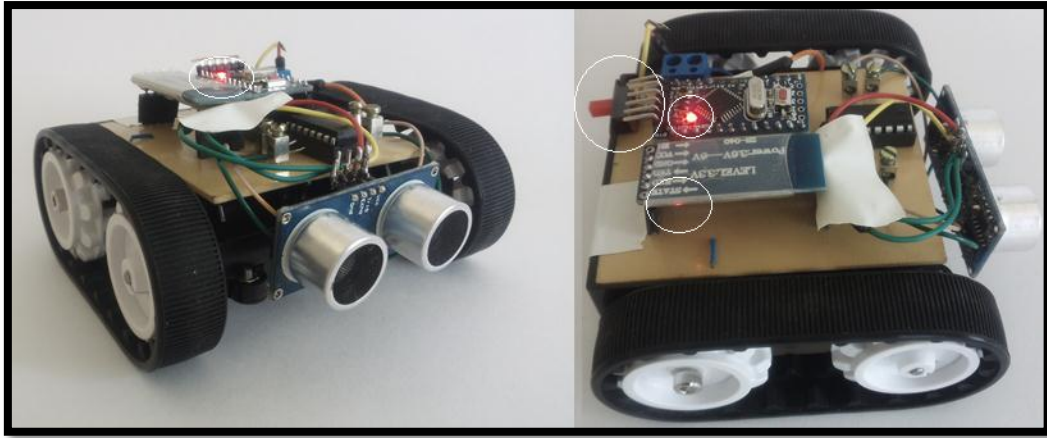


Ilustración 90. Prueba de encendido del prototipo

Fuente: Elaboración Propia.

En la *Ilustración 90* se muestra el switch de encendido, el módulo bluetooth y el Arduino Mini Pro que son los que intervienen en esta prueba. Se activa el switch y se verifica que se enciendan dos leds, El led del Arduino y el led del módulo bluetooth en intermitencia.

3.10.2. PRUEBA DE SECUENCIAS LÓGICAS DE PROGRAMACIÓN

En esta prueba se ha considerado dos tipos de pruebas, la una en laboratorio que es donde se respaldará fotográficamente las secuencias y los resultados y la otra se respalda con un video en campo en una escuela de Ibarra que será presentado el día de la defensa de este proyecto de titulación.

Para la primera prueba se toma las secuencias diseñadas en los temas anteriores para resolverlas una a una y los resultados se expondrán detalladamente a continuación.

3.10.2.1. PRUEBA DE LA SECUENCIA UNO

Esta prueba es la más sencilla, consiste en llegar a la meta después de haber recorrido tres distancias usando el botón Adelante. Los resultados se muestran a continuación.

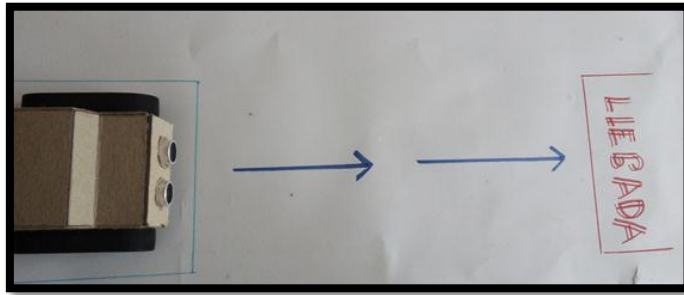


Ilustración 91. 1a. Partida de la secuencia



Ilustración 92. 1b. Desarrollo de la secuencia

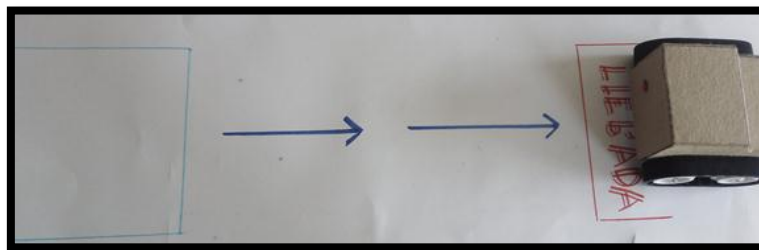


Ilustración 93. 1c. fin de la secuencia.



Ilustración 94. 1d. Instrucciones seleccionadas

Fuente: Elaboración Propia.

Como se ve en la *Ilustración 91*, *Ilustración 92* e *Ilustración 93* el prototipo electrónico cumple con la secuencia enviada desde el Smartphone como se muestra en la *Ilustración 94*.

3.10.2.2. PRUEBA DE LA SECUENCIA DOS

Esta prueba añade una instrucción más, consiste en llegar a la meta después de haber recorrido dos distancias usando el botón Adelante, un giro a la derecha y una distancia adicional al giro. Los resultados se muestran a continuación.

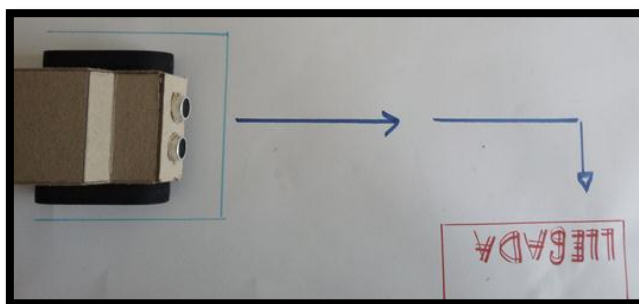


Ilustración 95. 2a. Partida de la secuencia

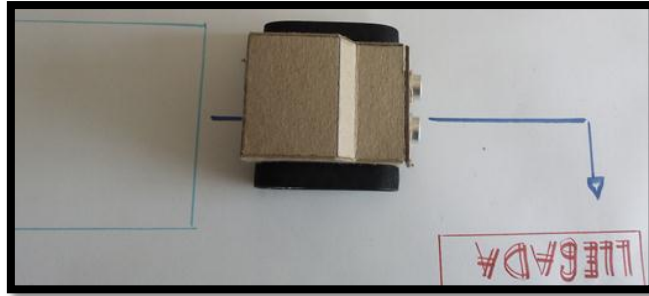


Ilustración 96. 2b. Desarrollo de la secuencia

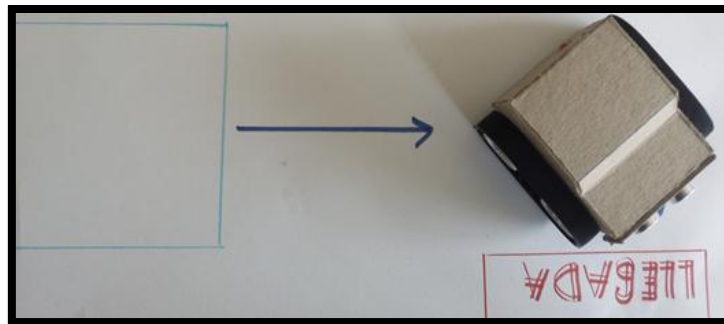


Ilustración 97. 2c. giro del prototipo electrónico

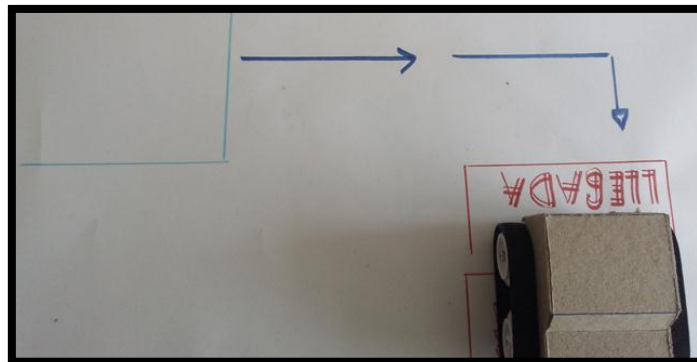


Ilustración 98. 2d. Fin de la secuencia



Ilustración 99. 2e. Instrucciones seleccionadas

Fuente: Elaboración Propia.

Como se ve en la *Ilustración 95, Ilustración 96, Ilustración 97 e Ilustración 98* el prototipo electrónico cumple con la secuencia enviada desde el Smartphone como se muestra en la *Ilustración 99*.

3.10.2.3. PRUEBA DE LA SECUENCIA TRES

Esta prueba es sencilla, consiste en llegar a la meta después de haber recorrido tres distancias usando el botón Repetición y dentro del conjunto de repetición el botón Adelante. Los resultados se muestran a continuación.

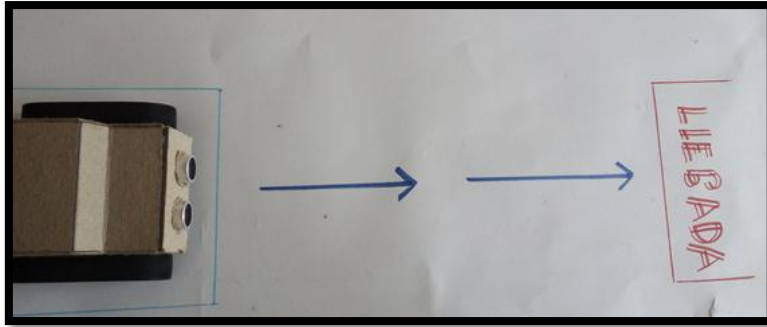


Ilustración 100. 3a. Partida de la secuencia

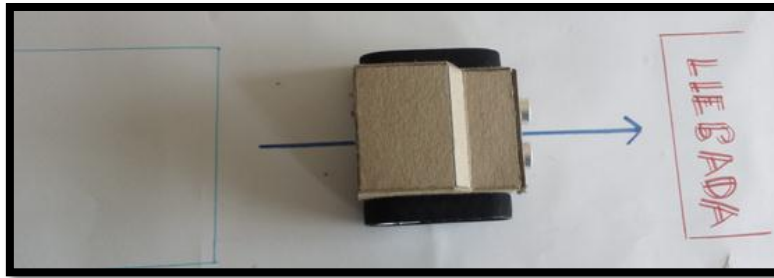


Ilustración 101. 3b. desarrollo de la secuencia

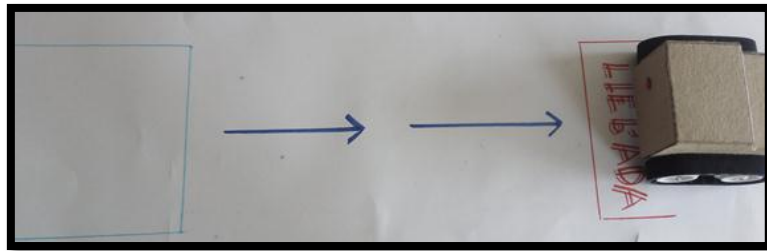


Ilustración 102. 3c. Fin de la secuencia

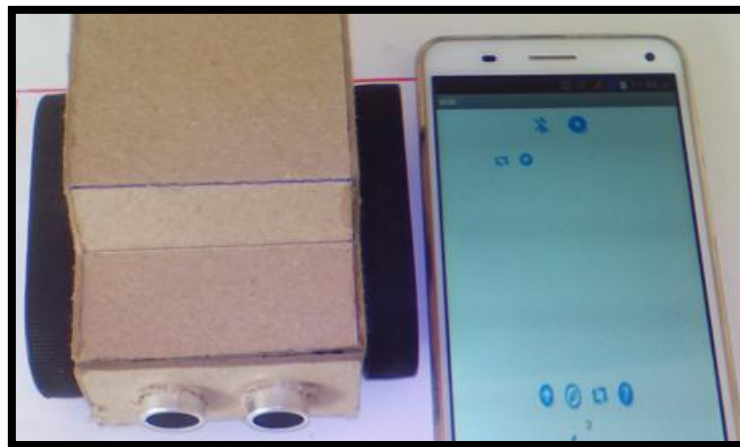


Ilustración 103. 3d. Instrucciones Seleccionadas

Fuente: Elaboración Propia.

Como se ve en la *Ilustración 100*, *Ilustración 101* e *Ilustración 102* el prototipo electrónico cumple con la secuencia enviada desde el Smartphone como se muestra en la *Ilustración 103*.

3.10.2.4. PRUEBA DE LA SECUENCIA CUATRO

Esta prueba consiste en llegar a la meta después de haber recorrido hacia Adelante, un giro a la derecha y una repetición de tres veces la instrucción Adelante. Los resultados se muestran a continuación.

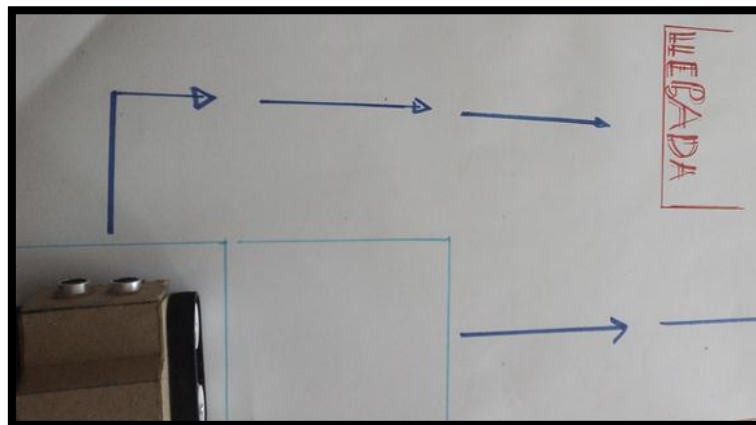


Ilustración 104. 4a. Partida de la secuencia

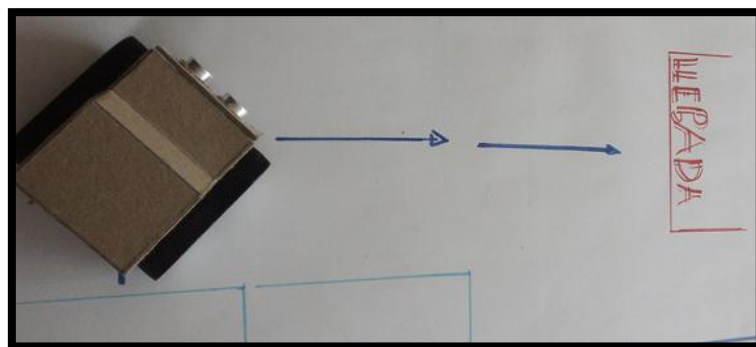


Ilustración 105. 4b. Giro hacia la derecha

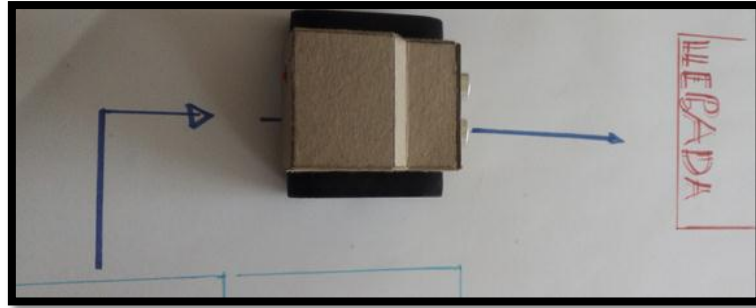


Ilustración 106. 4c. Desarrollo de la secuencia

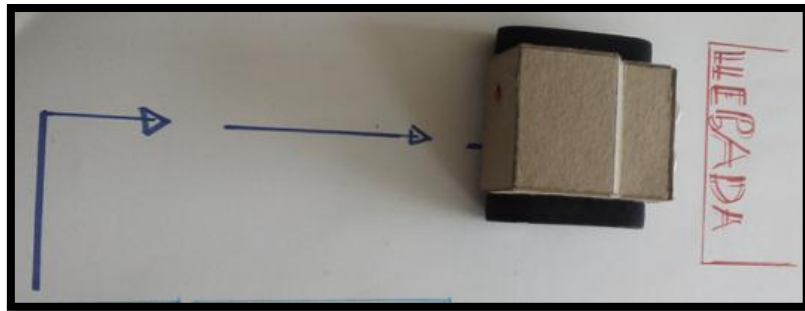


Ilustración 107. 4d. Desarrollo de la secuencia

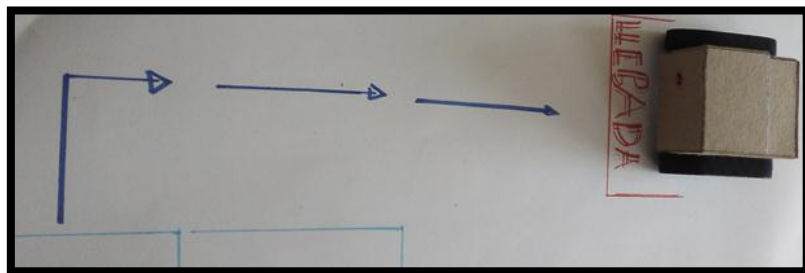


Ilustración 108. 4e. Fin de la secuencia

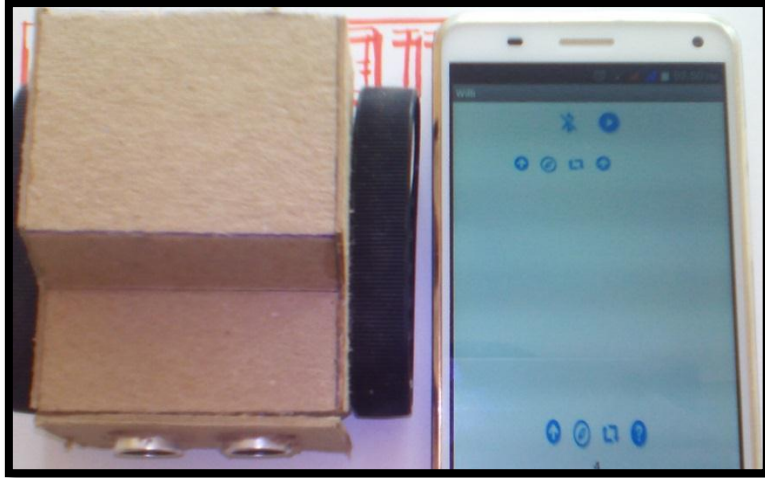


Ilustración 109. 4f. Instrucciones Seleccionadas

Fuente: Elaboración Propia.

Como se ve en la *Ilustración 104*, *Ilustración 105*, *Ilustración 106*, *Ilustración 107* e *Ilustración 108* el prototipo electrónico cumple con la secuencia enviada desde el Smartphone como se muestra en la *Ilustración 109*.

3.10.2.5. PRUEBA DE LA SECUENCIA CINCO

Esta prueba consiste en llegar a la meta después de haber recorrido una repetición de tres veces la instrucción Adelante, tomar una decisión luego de haber encontrado un obstáculo de girar a la izquierda, nuevamente recorrido una repetición de tres veces la instrucción Adelante, nuevamente tomar una decisión luego de haber encontrado un obstáculo de girar a la derecha y una medida hacia Adelante. Los resultados se muestran a continuación.

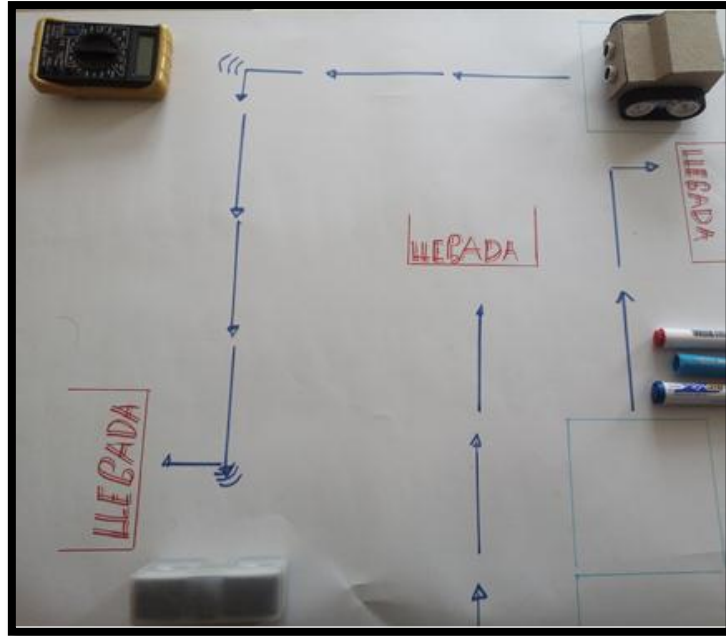


Ilustración 110. 5a. Partida de la secuencia

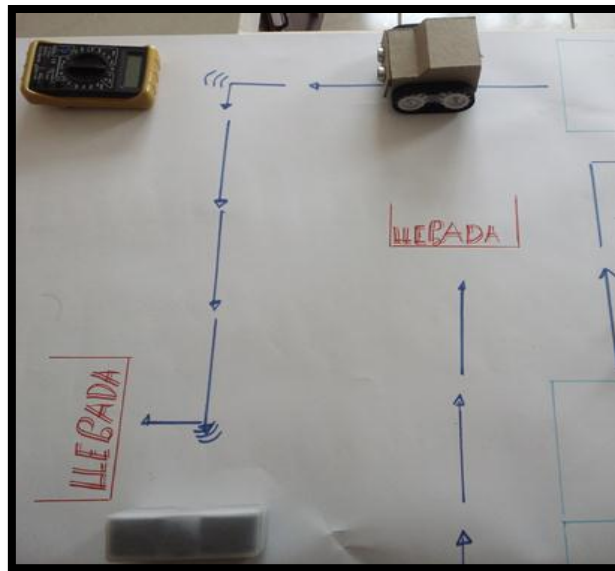


Ilustración 111. 5b. Desarrollo de la secuencia

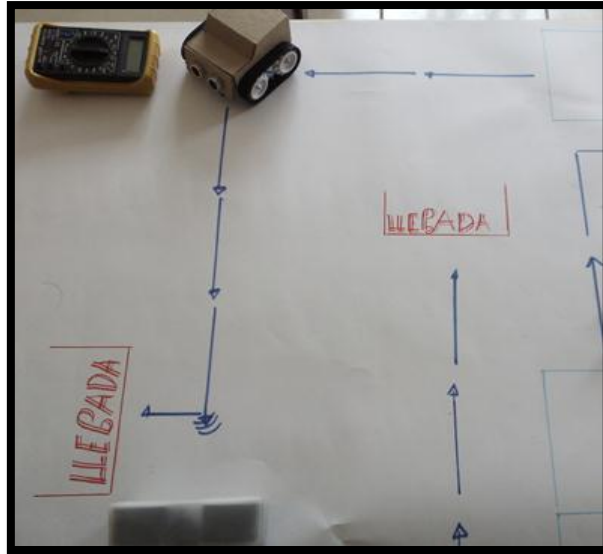


Ilustración 112. 5c. Giro a la izquierda luego de un obstáculo

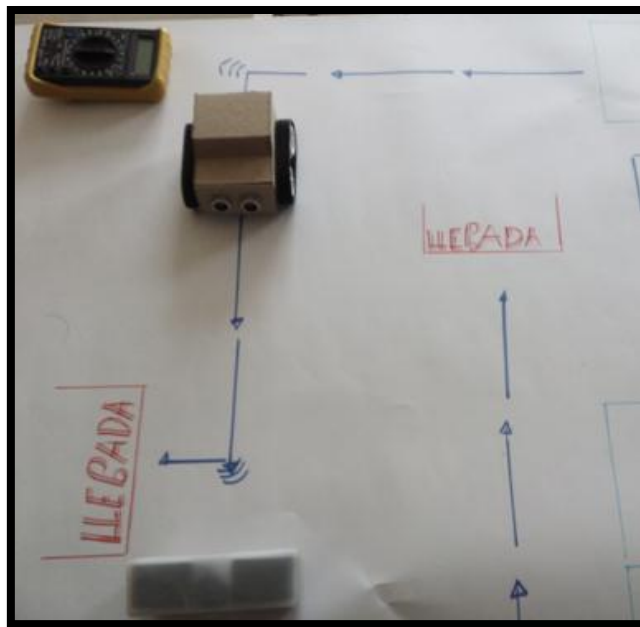


Ilustración 113. 5d. Desarrollo de la secuencia

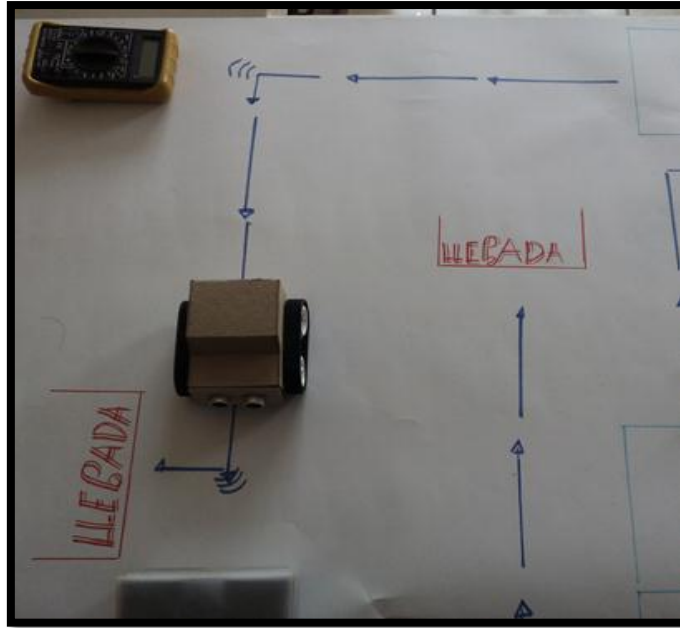


Ilustración 114. 5e. Desarrollo de la secuencia

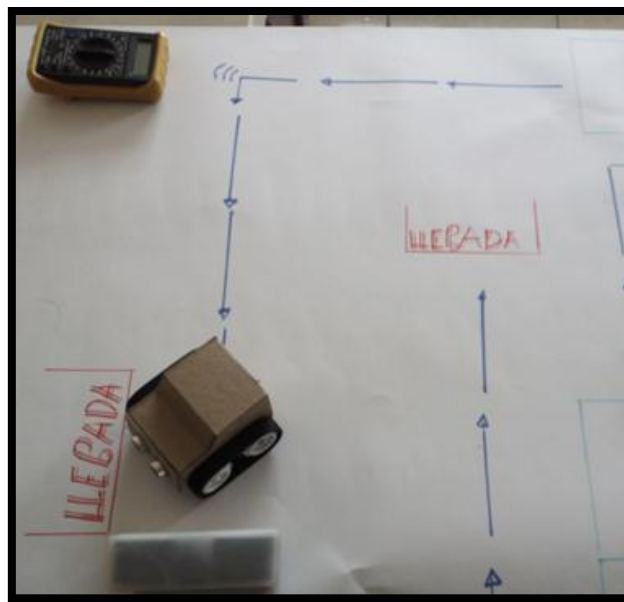


Ilustración 115. 5f. Giro a la derecha luego de un obstáculo

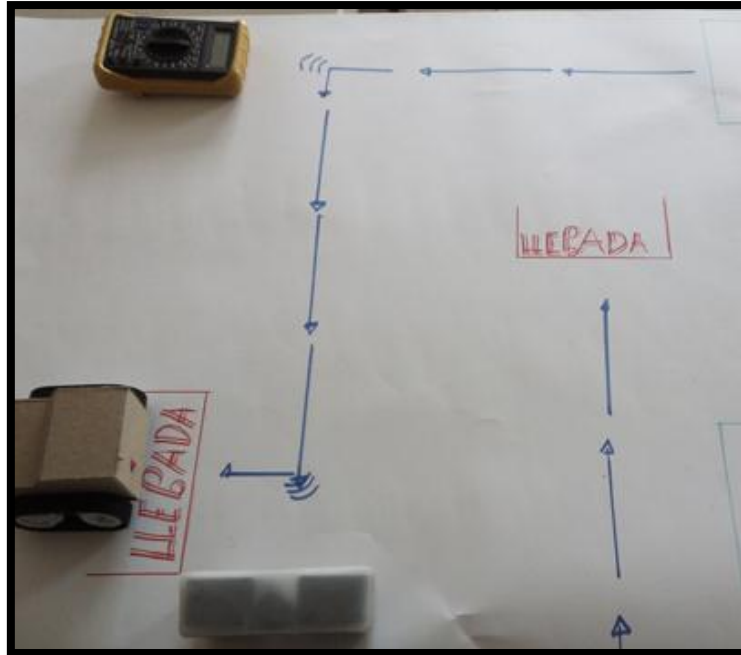


Ilustración 116. 5g. Fin de la secuencia



Ilustración 117. 5h. Instrucciones Seleccionadas

Fuente: Elaboración Propia.

Como se ve en la *Ilustración 110*, *Ilustración 111*, *Ilustración 112*, *Ilustración 113*, *Ilustración 114*, *Ilustración 115* e *Ilustración 116* el prototipo electrónico cumple con la secuencia enviada desde el Smartphone como se muestra en la *Ilustración 118*.

3.10.2.6. PRUEBA DE LA SECUENCIA ADICIONAL

Esta prueba consiste en llegar a la meta después de haber recorrido una repetición de tres veces el conjunto de instrucciones Adelante, giro a la izquierda, Adelante y giro a la derecha y una medida hacia Adelante. Los resultados se muestran a continuación.

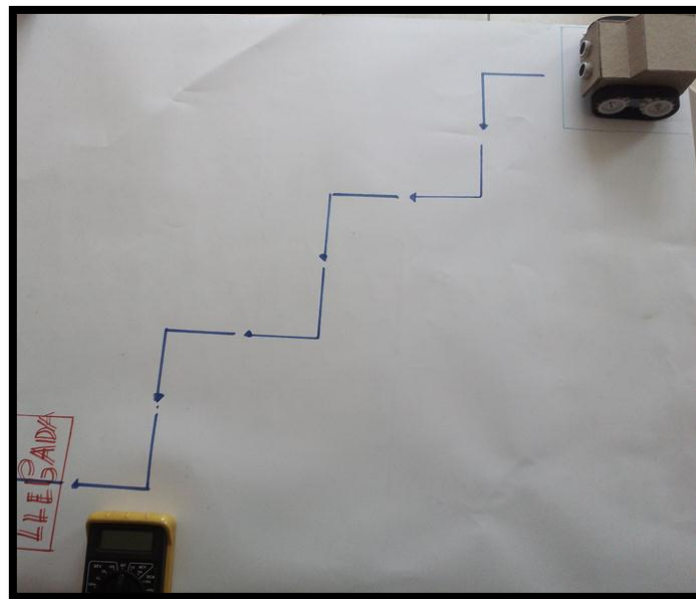


Ilustración 118. 6a. Partida de la Secuencia

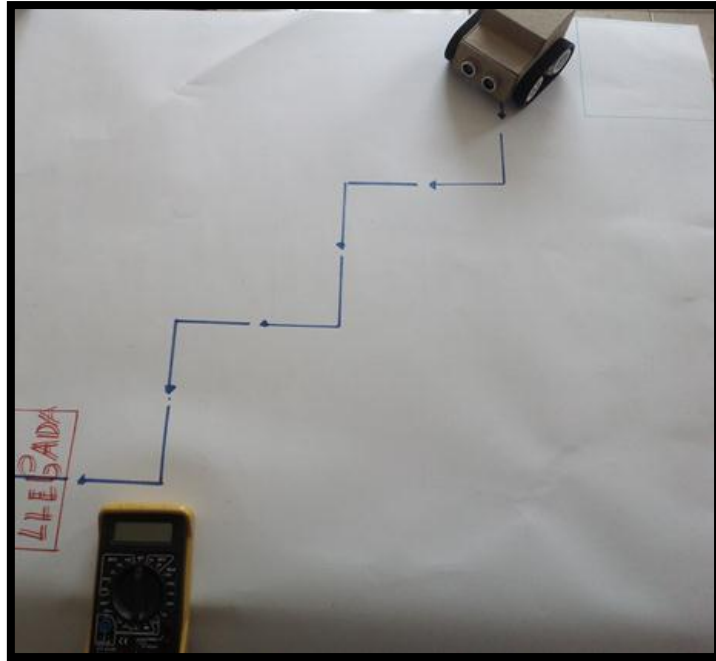


Ilustración 119. 6b. Giro hacia la izquierda

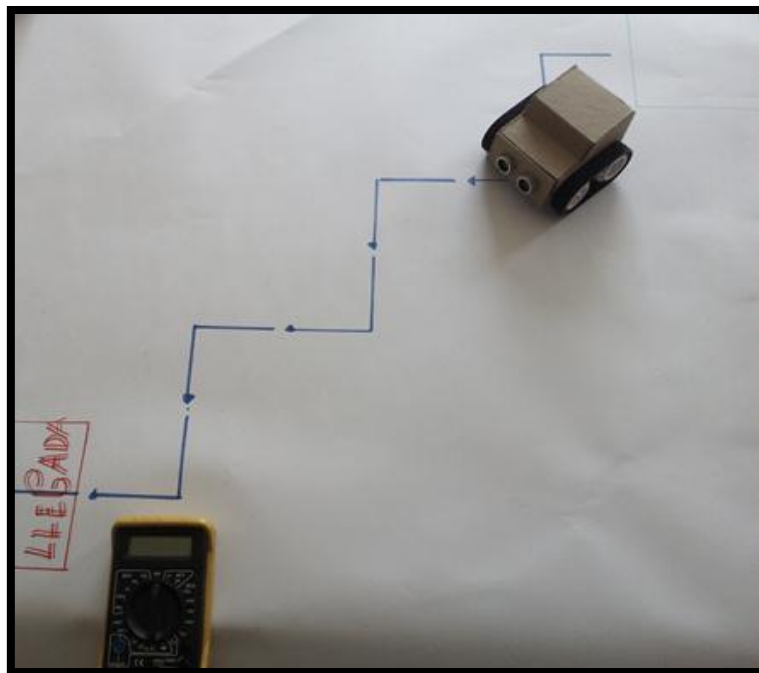


Ilustración 120. 6c. Giro a la derecha

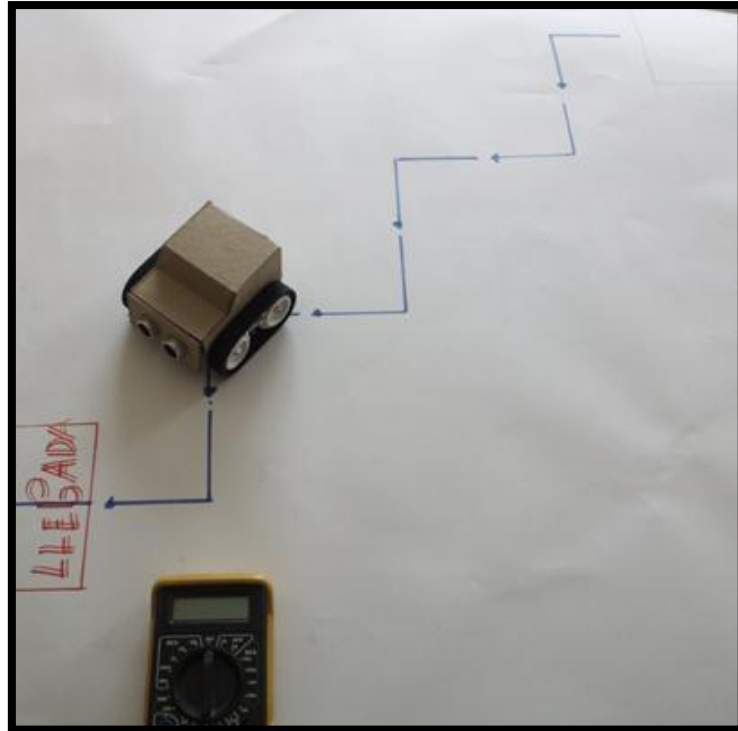


Ilustración 121. 6d. Giro a la izquierda tercera repetición

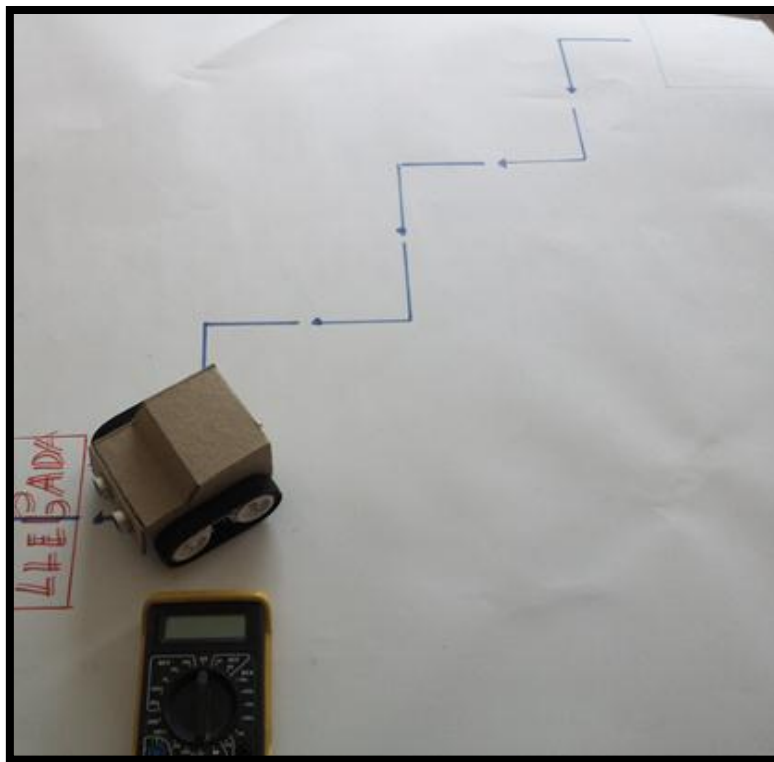


Ilustración 122. 6e. Giro a la derecha tercera repetición

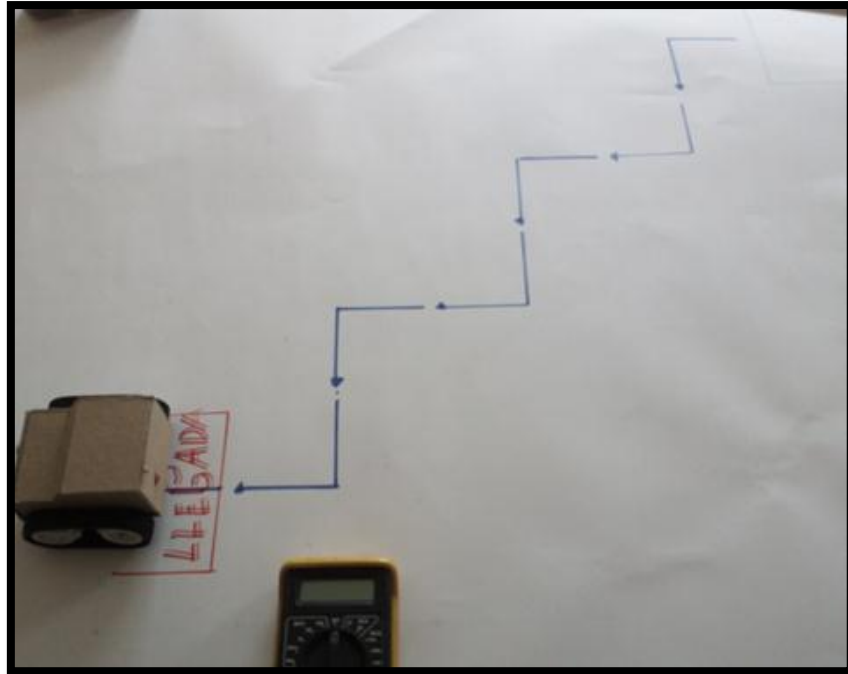


Ilustración 123. 6f. Fin de la Secuencia



Ilustración 124. 6g. Instrucciones Seleccionadas

Fuente: Elaboración Propia.

Como se ve en la *Ilustración 118, Ilustración 119, Ilustración 120, Ilustración 121, Ilustración 122 e Ilustración 123* el prototipo electrónico cumple con la secuencia enviada desde el Smartphone como se muestra en la *Ilustración 124*.

Las pruebas de secuencias lógicas de programación han tenido éxito, se ha probado todas las secuencias y a su vez se ha utilizado y combinado todas las instrucciones disponibles en la aplicación móvil basada en Arquitectura Open Source MIT App Inventor.

3.10.3. PRUEBA CON ESCOLARES DE 10 A 12 AÑOS

Las pruebas se realizaron con diez niños y niñas pertenecientes a las escuelas de la ciudad de Ibarra en las edades comprendidas de entre 10 a 12 años que son las edades a las cuales se enfocó este proyecto, en la *Ilustración 125* se muestra el desarrollo de las pruebas.



Ilustración 125. Pruebas con niños y niñas escolares de 10 a 12 años

Fuente: Elaboración Propia.

Durante la realización de las pruebas se observó el interés que los escolares tuvieron desde el momento que se les explicó a breve rasgo la prueba, se les indicó y leyeron el manual de usuario y después de haber recibido indicaciones acerca de las instrucciones se procedió a la manipulación y puesta en marcha del prototipo a través de las secuencias lógicas de programación; a simple vista se obtuvieron resultados favorables que se los detalla en la *Tabla 19*. Además en el ANEXO 04 se presenta un respaldo fotográfico y como resultados observables se presenta un Video Adjunto a este trabajo.

Tabla 19. Resultados de las pruebas con niños de 10 a 12 años

Actividades	Explicación	Resultado
Indicaciones generales del proyecto	Se les explicó a breve rasgo como se realizaría la prueba	Los escolares mostraron interés desde que se inició con la explicación
Presentación de los elementos que constituyen el proyecto	Los elementos del proyecto se les presentó de forma separada e indicándoles su función	Cada uno de los participantes observo con atención los elementos y mostraron inquietud
Indicaciones acerca de la aplicación móvil	Se les enseñó la App móvil, las instrucciones que contiene y la forma en cómo usarla	Los niños empiezan a comprender de mejor forma de lo que se trata y empiezan a comentar entre ellos según cada uno entiende
Indicaciones acerca del prototipo electrónico	Se presentó el prototipo electrónico, como se elaboró, las partes que lo conforman y como lo usaremos	Cada niño tuvo especial interés en el prototipo, todos lo manipularon con interés, lo asemejan con un personaje de la televisión
Indicaciones de las instrucciones en la App	Se explica cada una de las instrucciones que lleva la App, lo que significa y puede realizar cada una	Aquí los niños tienen dificultades en comprender inmediatamente, se solventa inquietudes, se explica nuevamente
Generación de secuencias de prueba	Después de la explicación anterior se les impulsa a ensayar en la App	Cada niño genera secuencias de programación con las indicaciones dadas

Aplicación de las secuencias de programación en el prototipo	Se les indicó que podía realizar las secuencias de programación y aplicarlo en el prototipo	Los niños realizan las secuencias de programación, el interés incrementa significativamente luego de observar que el prototipo realiza cada una de las instrucciones que programaron
---	---	--

Fuente: Resultados obtenidos según las pruebas realizadas.

Para cada actividad se empleó un tiempo determinado como se ve en la *Tabla 20* en la cual se indicó cada uno de los procedimientos a seguir para llegar a desarrollar las secuencias de programación, cada actividad estuvo documentada, explicada y verificada que cada niño comprenda las explicaciones impartidas. Cabe recalcar que los tiempos empleados para cada niño en comprender la idea y las explicaciones varían debido a la edad y a la instrucción escolar de cada niño, además son tiempos promedio entre las mismas edades, los niños de 12 años sin duda transformaron las explicaciones en construcción de secuencias en menor tiempo que los de 11 y 10 años, este resultado se lo obtiene a través de observación directa en el transcurso de la prueba.

Tabla 20. Tiempos de Perfeccionamiento del Prototipo

ACTIVIDADES	10 AÑOS	11 AÑOS	12 AÑOS
Indicaciones generales del proyecto	15 Minutos	15 Minutos	15 Minutos
Presentación de los elementos que constituyen el proyecto	30 Minutos	30 Minutos	30 Minutos
Indicaciones acerca de la aplicación móvil	15 Minutos	10 Minutos	10 Minutos
Indicaciones acerca del prototipo electrónico	10 Minutos	10 Minutos	10 Minutos
Indicaciones de las instrucciones en la App	25 Minutos	20 Minutos	20 Minutos
Generación de secuencias de prueba	10 Minutos	10 Minutos	7 Minutos

Aplicación de las secuencias de programación en el prototipo	15 Minutos	12 Minutos	10 Minutos
---	------------	------------	------------

Fuente: Elaboración Propia de acuerdo a los tiempos empleados para cada actividad en la prueba realizada con niños de 10 a 12 años.

La *Tabla 20* permitió determinar que hay niños que pueden aprender más rápido que otros pero en general los niños de mayor edad aprenden y captan las indicaciones rápidamente que los de menor edad por ende la generación de secuencias toma menos tiempo. El tiempo de perfeccionamiento del prototipo en promedio es de 110 minutos o 1 hora con 50 minutos, tiempo en el cual los niños y niñas perfeccionaron el manejo del prototipo electrónico y la aplicación móvil.

3.10.4. VERIFICACIÓN DE APRENDIZAJE

Una vez culminada la prueba con los niños de 10 a 12 años se les pidió que escribieran en orden las secuencias que habían realizado con el prototipo, se seleccionó una muestra de 5 niños para respaldar los resultados de aprendizaje obtenidos y adquiridos por los niños, la muestra de los resultados de que el objetivo fue cumplido constan en el ANEXO 03.

3.11. MANUAL DE USUARIO PARA LA CORRECTA UTILIZACIÓN Y MEJOR APROVECHAMIENTO DEL PROTOTIPO ELECTRÓNICO Y SU APLICACIÓN.



Ilustración 126. Portada del Manual

Fuente: Software de edición de imágenes de Microsoft.

Primeramente se define como usuario a la escolar niña o niño en la edad comprendida entre los diez y los doce años de edad para quien se ha propuesto este trabajo y se dirige este manual, esperando que sea de fácil comprensión y que le dé un manejo adecuado a la aplicación móvil y al prototipo electrónico que se muestran en la *Ilustración 126*.

Este manual le guiará al usuario paso a paso en el manejo de la Aplicación, le dará recomendaciones para un correcto trato al prototipo, para que obtenga el mejor provecho posible y deseándole que sea de mucha utilidad dentro de su proceso de aprendizaje en las aulas de clase que es donde se usará este proyecto.

3.11.1. GUÍA DE USO

3.11.1.1. GUIA DE ENCENDIDO DEL PROTOTIPO.

- Identifique el prototipo electrónico y proceda a encenderlo a través del switch ubicado en la parte posterior del prototipo, cambiando la posición de OFF a ON. En la *Ilustración 127* se observa este paso.

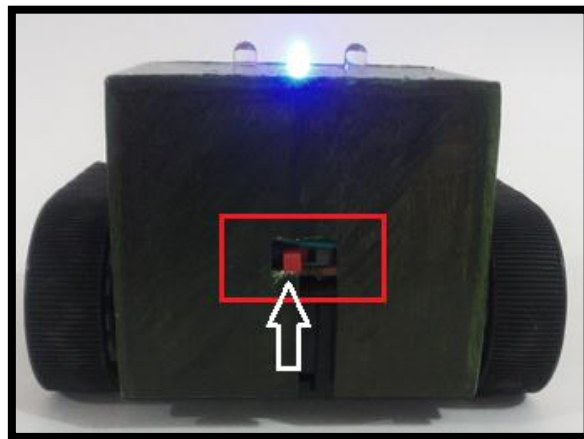


Ilustración 127. Encendido del Prototipo

Fuente: Elaboración Propia.

3.11.1.2. GUIA DE USO DE LA APLICACIÓN.

- Ubíquese en el dispositivo móvil o Smartphone desbloqueado, presione el botón Menú e identifique la aplicación de nuestro prototipo previamente instalada y presione sobre el ícono de la aplicación. Este paso se muestra en la *Ilustración 128*.

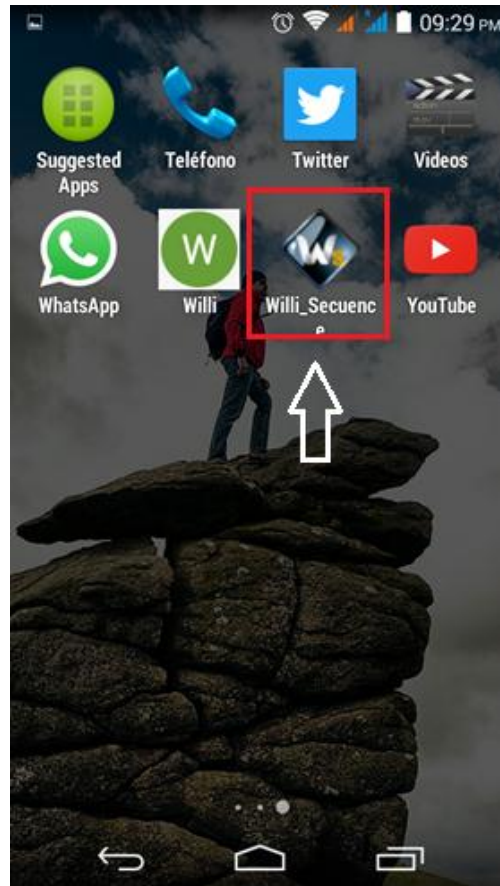


Ilustración 128. Aplicación dentro del menú

Fuente: Captura de Smartphone Sky 6.0Q.

- Una vez abierta la aplicación se despliega la portada principal de esta como se ve en la *Ilustración 129*, donde se construirá las diferentes secuencias de programación. Para

familiarizarse con la aplicación se indican cada uno de los botones que esta contiene y que le permitirán poder usarlas correctamente muy pronto.



Ilustración 129. Pantalla Principal de la Aplicación

Fuente: Captura de Smartphone Sky 6.0Q.

- 1.- Permite conectarse mediante conexión bluetooth del Smartphone hacia el módulo bluetooth integrado en el prototipo electrónico, permitiendo el envío de los caracteres hacia Arduino.
- 2.- Permite desconectarse de la conexión bluetooth una vez finalizadas las secuencias de programación. Es importante que desconecte el bluetooth antes de salir de la aplicación.
- 3.- Permite abrir un conjunto de instrucciones.
- 4.- Permite cerrar un conjunto de instrucciones.
- 5.- Permite iniciar la ejecución de la secuencia una vez que esté desarrollada.
- 6.- Instrucción Adelante, permite escoger la instrucción de un avance en sentido frontal.

7.- Instrucción Giro, permite visualizar un subconjunto de instrucciones de giro a 0° y 180° que podrán ser escogidos dependiendo de la necesidad del problema presentado.

8.- Instrucción Repetición, permite visualizar un subconjunto de instrucciones de dos repeticiones y de tres repeticiones que podrán ser escogidos dependiendo de cuantas veces se quiera repetir un conjunto de instrucciones.

9.- Instrucción Decisión, permite escoger la instrucción de decidir en caso de que el prototipo se encuentre con un obstáculo.

10.- Permite limpiar el escritorio de trabajo una vez realizado las secuencias o cuando se considere necesario.

11.- Cierra la aplicación.

- Una vez revisado los íconos principales se procede a usar las instrucciones. Primeramente se tiene la instrucción Adelante, al dar clic sobre la instrucción esta se posicionará sobre el área de trabajo en la primera posición, véase la *Ilustración 130*.

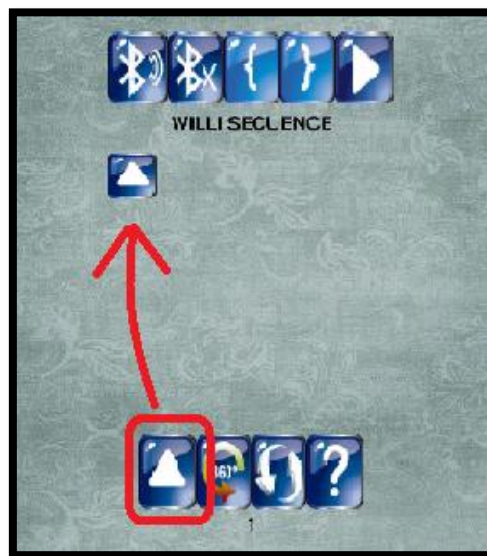


Ilustración 130. Uso de Instrucción Adelante

Fuente: Captura de Smartphone Sky 6.0Q.

- Luego se tiene la instrucción Graduación, al dar clic sobre esta se despliega un subconjunto de instrucciones que indicarán las opciones de giro en grados. Se escoge el grado que se desea girar. El ícono de graduación se ubicará en el área de trabajo correspondiente como se ve en la *Ilustración 131*.



Ilustración 131. Uso de la Instrucción Graduación

Fuente: Captura de Smartphone Sky 6.0Q.

- Luego se tiene la instrucción Repetición, al dar clic sobre esta se despliega un subconjunto de instrucciones que indicarán las veces de. Se escoge las veces que se desea repetir. El ícono de repetición se ubicará en el área de trabajo correspondiente como se ve en la *Ilustración 132*.

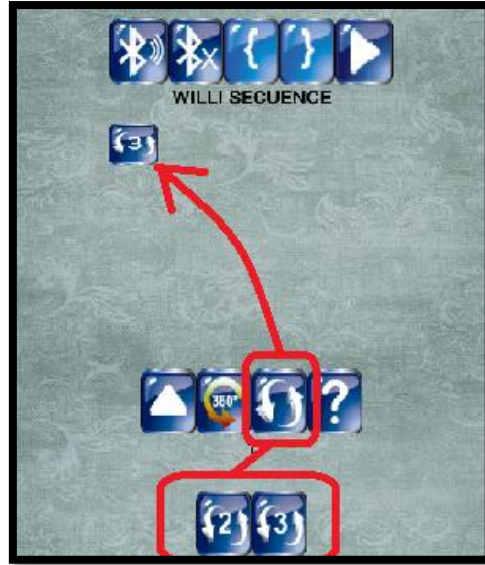


Ilustración 132. Uso de la Instrucción Repetición.

Fuente: Captura de Smartphone Sky 6.0Q.

- Y por último se tiene la instrucción Decisión. Al dar clic sobre el ícono de decisión se ubicará en el área de trabajo correspondiente como se ve en la *Ilustración 133*.

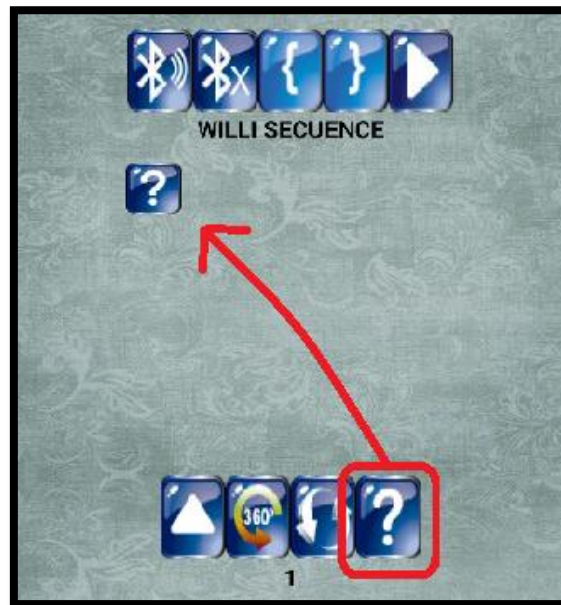


Ilustración 133. Uso de la Instrucción Decisión.

Fuente: Captura de Smartphone Sky 6.0Q.

- Se tiene dos íconos adicionales que serán usados conjuntamente con las instrucciones de Repetición y Decisión como son Abrir conjunto y Cerrar conjunto. Estos íconos permiten crear un conjunto de instrucciones que se desee repetir o en el caso de decisión que se desea realizar en caso de obstáculos. Abrir el conjunto, seleccionar las instrucciones deseadas y Cerrar el conjunto, véase la *Ilustración 134*.

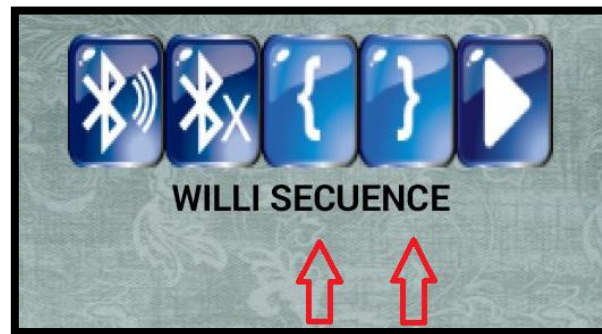


Ilustración 134. Íconos de conjunto.

Fuente: Captura de Smartphone Sky 6.0Q.

- Finalmente se muestra un ejemplo de una secuencia de instrucciones realizada y ubicada en el área de trabajo como se ve en la *Ilustración 135*.

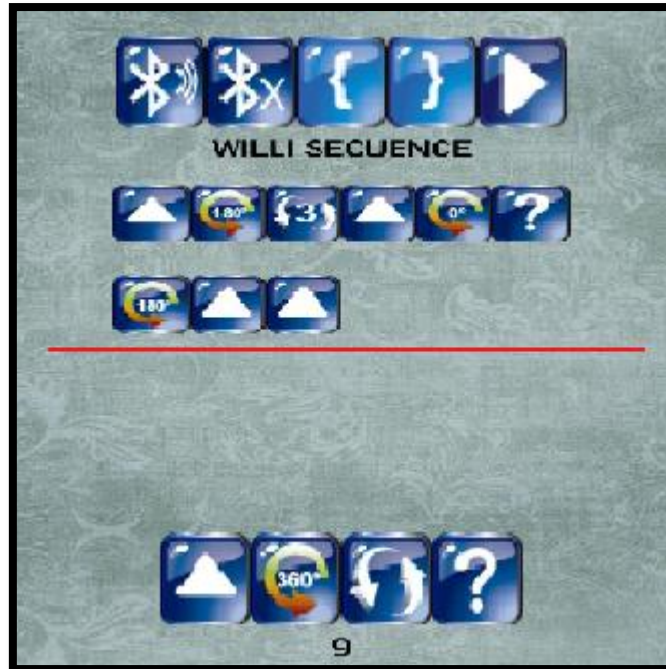


Ilustración 135. Ejemplo de una Secuencia.

Fuente: Captura de Smartphone Sky 6.0Q.

3.11.2. RECOMENDACIONES GENERALES PARA EL BUEN USO Y CONSERVACIÓN DEL PROTOTIPO ELECTRÓNICO.

- Evitar tirarlo al piso.
- No introducirlo en agua.
- No presionarlo con las manos.
- No golpearlo.
- No transportarlo fuera de su caja.
- No pisarlo.
- No manipularlo con las manos de grasa.

CAPÍTULO 4. ANÁLISIS DE COSTO BENEFICIO

En este capítulo se realiza el análisis económico de los materiales usados con respecto al proyecto el Prototipo electrónico programado mediante bloques con una aplicación móvil para la enseñanza de programación a niños de 10 a 12 años en la ciudad de Ibarra y de los elementos electrónicos utilizados en este mismo proyecto, también del dispositivo móvil o Smartphone que se usó de plataforma para el desarrollo de la aplicación móvil.

4.1. PRESUPUESTO DEL HARDWARE EMPLEADO

Para la implementación del prototipo electrónico programado mediante bloques, en la *Tabla 21* se observan los costos que incurrieron en la implementación del segundo prototipo, así como de los materiales adicionales que se usaron, los respaldos se encuentran en el ANEXO 05.

Tabla 21. Presupuesto del Prototipo

CANTIDAD	MATERIAL	COSTO UNITARIO	COSTO TOTAL
1	Kit Chasis Pololu	\$ 25.00	\$ 25.00
1	Carcasa de Madera	\$ 12.00	\$ 12.00
1	Kit Llantas	\$ 28.00	\$ 28.00
1	Mano de Obra	\$ 35.00	\$ 35.00
	TOTAL	-	\$ 100.00

Fuente: Elaboración Propia.

Como se muestra en la *Tabla 21*, el costo de los materiales empleados en el segundo prototipo, incluyendo la mano de obra, su valor se estima en 100.00 dólares americanos.

A continuación se detallan los costos generales de los elementos electrónicos que han sido empleados incluyendo los elementos del primer prototipo hasta el prototipo final, como se observa en la *Tabla 22*.

Tabla 22. Presupuesto de elementos electrónicos usados

CANTIDAD	MATERIAL	COSTO	COSTO TOTAL
		UNITARIO	
2	Micro motores	\$ 12.00	\$ 24.00
1	Módulo Bluetooth HC-05	\$ 18.00	\$ 18.00
1	Arduino Mini PRO	\$ 9.00	\$ 9.00
1	Sensor Ultrasónico HC-SR04	\$ 5.00	\$ 5.00
1	Circuito integrado L293D	\$ 4.00	\$ 4.00
2	Baterías 3.7V - 800mAh	\$ 9.00	\$ 18.00
2	Diodos Led	\$ 0.20	\$ 0.40
2	Hilo conductor (2m)	\$ 0.60	\$ 1.20
2	Resistencias 1Kohm	\$ 0.80	\$ 1.60
3	Borneras de 2 pines	\$ 0.35	\$ 1.05
1	Switch	\$ 0.75	\$ 0.75
1	Espadines tipo hembra	\$ 1.20	\$ 1.20
1	Zócalo	\$ 0.50	\$ 0.50
1	Estaño (1m)	\$ 0.40	\$ 0.40
1	Cautín	\$ 3.50	\$ 3.50
2	Acido	\$ 1.20	\$ 2.40
1	Baquelita de cobre	\$ 2.00	\$ 2.00
1	Cable Termotransferible (1m)	\$ 0.60	\$ 0.60
1	Silicona	\$ 1.50	\$ 1.50

TOTAL	-	\$	95.10
--------------	---	----	--------------

Fuente: Elaboración Propia.

En la *Tabla 22* se ha detallado todos los componentes empleados en el prototipo electrónico, dando un total de 95.10 dólares americanos.

En la *Tabla 23* se muestra el costo de la adquisición del dispositivo Smartphone con sistema Android 4.2, este dispositivo se utilizó como plataforma de soporte de la aplicación móvil de App Inventor.

Tabla 23. Presupuesto del Smartphone

CANTIDAD	MATERIAL	COSTO UNITARIO	COSTO TOTAL
1	Smartphone SKY 6.0Q	\$ 260.00	\$ 260.00
	TOTAL	-	\$ 260.00

Fuente: Elaboración Propia.

Cabe aclarar que el costo de adquisición de este dispositivo móvil no es fijo, debe cumplir con las características especificadas anteriormente del sistema operativo y recursos de memoria y procesador, existen Smartphone que pueden ser utilizados que sus precios oscilan desde los 100.00 hasta los 1200.00 dólares americanos que cumplen con los requerimientos. Dependiendo de las condiciones económicas de los usuarios que usen este prototipo, podrían adquirir un dispositivo a la medida de su bolsillo o a su vez obviar este gasto si ya tiene un Smartphone con sistema operativo Android.

4.2. PRESUPUESTO DEL SOFTWARE EMPLEADO

En el análisis del presupuesto de software se tiene como resultado 0.00 dólares debido a que los programas que se utilizan son bajo Arquitectura Open Source y acceso libre, pero esto no impide que se detallen tal como se muestra en la *Tabla 24*.

Tabla 24. Presupuesto de Software

CANTIDAD	DESCRIPCIÓN	COSTO TOTAL
1	IDE de Arduino	\$ 0.00
1	Eagle 6.5.0	\$ 0.00
1	MIT App Inventor	\$ 0.00
1	MIT Ai2 Companion	\$ 0.00
TOTAL		\$ 0.00

Fuente: Elaboración Propia.

4.3. ANÁLISIS COSTO BENEFICIO

“La característica que distingue al análisis de costo beneficio es el intento de llevar al máximo posible la cuantificación los beneficios y costos en términos monetarios. Sin embargo, el análisis muy pocas veces logra ese ideal de medir todos los beneficios y costos en términos monetarios...” (Martínez, 2014)

Este análisis obliga a detallar los elementos utilizados en el diseño e implementación del prototipo electrónico tanto lo tangible como lo intangible, tomando en cuenta que los costos deben ser reales y que se puedan medir económicamente, para este caso la unidad es el dólar americano adoptado por nuestro país. Los beneficios son determinados de manera grupal

debido a que el proyecto está enfocado a niños de la escuela que les permite beneficiarse en el desarrollo de su inteligencia y una forma diferente en la que ellos aprenden supliendo la falta de herramientas didácticas en su entorno educativo.

En la *Tabla 25* se detalla los costos que implicó poner a punto este proyecto.

Tabla 25. Costos totales del proyecto

DESCRIPCIÓN	VALOR
Prototipo Electrónico	\$ 100.00
Costo de Elementos Electrónicos	\$ 95.10
Costo del Smartphone	\$ 260.00
Costo del Software	\$ -
TOTAL	\$ 455.10

Fuente: Elaboración Propia.

El beneficio que se obtiene una vez implementado en prototipo electrónico programado mediante bloques, es el de empezar a adquirir educación de manera diferente, ya no basado en lo tradicional que el profesor enseña y los alumnos reciben esa información y todo esto en un pizarrón o una computadora, con este prototipo el alumno puede usar las herramientas tecnológicas que tiene a la mano como Tablet o Smartphone no solo para revisar contenido multimedia sino lo podrá usar para aprender lógica de programación a través de la aplicación móvil y el prototipo electrónico que es hacia donde se enfoca este proyecto.

En el Ecuador a través de las autoridades de Educación ha invertido en suministros, útiles escolares, equipos de computación, laboratorios de informática, mobiliario y demás cosas que se requiere en las escuelas todo con el fin de mejorar la educación en nuestro país, este proyecto aportará un granito de arena hacia ese fin, el alumno será el principal beneficiado porque cuenta con nueva herramienta para desarrollar sus capacidades a la vez que aumenta su aprendizaje.

CAPÍTULO 5. CONCLUSIONES Y RECOMENDACIONES

Una vez finalizado e implementado el presente proyecto de titulación denominado prototipo electrónico programado mediante bloques con una aplicación móvil para la enseñanza de programación a niños de 10 a 12 años en la ciudad de Ibarra; se presentan las siguientes conclusiones y recomendaciones.

CONCLUSIONES

- ✓ Se desarrolló un prototipo electrónico a través de plataformas Open Source para la enseñanza básica de programación a escolares de 10 a 12 años de edad el cual ayudó a los niños en el aprendizaje de la base de la programación y razonamiento lógico demostrado en las pruebas con los escolares y respaldado en la evidencia fotográfica, audiovisual y en las secuencias transformadas en líneas de código simple.

- ✓ Para conocer la realidad del aprendizaje de programación en los escolares se levantó encuestas, se hizo una investigación acerca de las metodologías de aprendizaje en niños de 10 a 12 años de edad, las mismas que permitieron conocer el nivel de desarrollo cognitivo y de aprendizaje que tienen en esas edades enfocado en el aprendizaje de programación, y donde se determina la posibilidad del diseño de secuencias básicas de programación para los escolares.

- ✓ El uso de secuencias lógicas de programación por medio de bloques integradas en la aplicación móvil utilizando las instrucciones predefinidas en la aplicación le permiten

al escolar interactuar con el prototipo utilizando su razonamiento, evaluar la situación del problema presentado, realizar una secuencia lógica y verificarla inmediatamente se procede a su ejecución.

- ✓ Al implementar el prototipo y realizar las pruebas con una selección de niños entre los 10 y los 12 años en etapa escolar se obtuvieron resultados halagadores e inesperados; en primer lugar los escolares mostraron un gran interés en el prototipo y su aplicación, cooperaron con el resto de estudiantes durante el desarrollo de las pruebas, fueron capaces de resolver sus retos planteados individualmente y posterior plasmaron los resultados de sus secuencias en papel en forma de código. Todos estos resultados se encuentran respaldados en los Anexos.

- ✓ Como instrumento de apoyo y manipulación del prototipo se elaboró un manual de usuario que permitió una mejor comprensión en el uso y manejo de la aplicación por parte del escolar permitiéndole un correcto uso sin exponer al prototipo a una incorrecta funcionalidad o que comprometa su integridad.

- ✓ El análisis costo beneficio del prototipo electrónico detalla minuciosamente el costo de todos y cada uno de los elementos y materiales electrónicos empleados en este proyecto, lo que permite determinar el costo económico del prototipo, un costo referencial para en el futuro desarrollar y comercializar más unidades.

RECOMENDACIONES

- ✓ El proyecto abre camino en el tema de herramientas didácticas para el uso en las escuelas y por ello es recomendable que los alumnos de ingeniería y principalmente de nuestra carrera observen las potencialidades que ofrece esta área de la enseñanza, se busque desarrollar prototipos para esta área sacándoles el mayor provecho posible en beneficio de los escolares.
- ✓ Se recomienda leer primero el manual de usuario incluido en este proyecto con el objetivo de evitar problemas en el uso la aplicación del dispositivo móvil y evitar cualquier amenaza a la integridad del prototipo electrónico en caso de suscitarse algún inconveniente.
- ✓ Se recomienda conocer todos y cada uno de los elementos electrónicos a utilizar en la implementación del prototipo electrónico, conocer sus dimensiones, características técnicas y de operación con el propósito de evitar fallas, quemadura de elementos y cortocircuitos durante la implementación.
- ✓ Se recomienda el uso de software libre bajo la Arquitectura Open Source debido a que son herramientas desarrolladas por equipos de trabajo e instituciones que tienen como fin mejorar procesos ya establecidos y que la comunidad tecnológica y de código sea más grande, los programas que se ha usado a lo largo de este trabajo como IDE de Arduino o MIT App Inventor responden a esos lineamientos además que son gratuitos.

- ✓ Se recomienda a los ingenieros que tomen como referencia este trabajo se enfoquen en mejorar el aspecto de diseño, mayores funcionalidades, más llamativo, más compacto entre otras cosas que se puede mejorar y perfeccionarlo.

BIBLIOGRAFÍA

- Abbott, E. (1999). *Constructing Knowledge and Shaping Brains*. Londres: Abbott.
- Amado, N. (2012, Septiembre 5). *Bloques de programacion scratch*. Retrieved from Bloques de programacion scratch: <http://es.slideshare.net/nicolosamado/bloques-de-programacion-scratch>
- Android. (2015, Noviembre 13). *android*. Retrieved from android: <https://www.android.com/>
- Andujar, O. (2013). *22 Estrategias y Técnicas de aprendizaje*. Ciudad Real: Andujar.
- Angela, B. (2012, 07 09). *Programación de Arduino: El IDE de Arduino*. Retrieved from Programación de Arduino: El IDE de Arduino: <http://solorobotica.blogspot.com/2012/07/programacion-de-arduino-elide-de.html>
- Apple Inc. (2015, Noviembre 19). *iOS*. Retrieved from iOS: <http://www.apple.com/ios/>
- Apple Inc. (2015, Agosto 31). *iOS 9*. Retrieved from iOS 9: <http://www.apple.com/es/ios/what-is/>
- Arduino. (2013, Mayo 28). *Arduino*. Retrieved from Arduino: <https://www.arduino.cc/>
- Arduino. (2013, Mayo 28). *Arduino Due*. Retrieved from Arduino Due: <https://www.arduino.cc/en/Main/ArduinoBoardDue>
- Arduino. (2013, Mayo 28). *Arduino Leonardo*. Retrieved from Arduino Leonardo: <https://www.arduino.cc/en/Main/ArduinoBoardLeonardo>
- Arduino. (2013, Mayo 28). *Arduino MEGA 2560*. Retrieved from Arduino MEGA 2560: <https://www.arduino.cc/en/Main/ArduinoBoardMega2560>
- Arduino. (2013, Mayo 30). *Arduino Mini PRO*. Retrieved from Arduino Mini PRO: <https://www.arduino.cc/en/Main/ArduinoBoardProMini>
- Arduino. (2013, Mayo 30). *Arduino Nano*. Retrieved from Arduino Nano: <https://www.arduino.cc/en/Main/ArduinoBoardNano>
- Arduino. (2013, Mayo 30). *Arduino Uno*. Retrieved from Arduino Uno: <https://www.arduino.cc/en/Main/ArduinoBoardUno>
- Arduino. (2013, Mayo 30). *Arduino Yun*. Retrieved from Arduino Yun: <https://www.arduino.cc/en/Main/ArduinoBoardYun>
- Arduino. (2013, Mayo 28). *What is Arduino?* Retrieved from What is Arduino?: <https://www.arduino.cc/en/Guide/Introduction>
- ARQUITECTURA, A. (2013, 04 24). *Propiedades de la madera*. Retrieved from Propiedades de la madera: <http://www.arqhys.com/contenidos/madera-propiedades.html>

- ATAREAO, E. (2013, 06 28). *TODO EL MUNDO DEBERÍA SABER PROGRAMAR*. Retrieved from TODO EL MUNDO DEBERÍA SABER PROGRAMAR: <http://www.atareao.es/programacion/todo-el-mundo-deberia-saber-programar/>
- BBC MUNDO. (2013, 05 01). *Tecnología, ¿beneficia o perjudica el desarrollo de los niños?* Retrieved from Tecnología, ¿beneficia o perjudica el desarrollo de los niños?: http://www.bbc.com/mundo/noticias/2013/05/130422_salud_bebe_tecnologia_desarrollo_gtg
- Bolaños, B. H. (2003, 07 21). *Paradigmas*. Retrieved from Paradigmas: <http://www.academia.edu/3670572/Paradigmas>
- Bravo, A. (2011, Agosto 3). *MOTORES DE CORRIENTE CONTINUA*. Retrieved from MOTORES DE CORRIENTE CONTINUA: <http://solorobotica.blogspot.com/2011/08/motores-de-corriente-continua.html>
- Brooks, J. G. (1999). *In Search of Understanding: The Case for Constructivist Classrooms*. Virginia, USA: Brooks.
- CadSoft EAGLE. (2015, 29 02). *What is EAGLE*. Retrieved from What is EAGLE: <http://www.cadsoftusa.com/eagle-pcb-design-software/about-eagle/>
- CCM. (2014, 05 21). *¿Qué es un programa informático?* Retrieved from ¿Qué es un programa informático?: <http://es.ccm.net/contents/305-programa-informatico>
- Clic, A. (2015, 06 30). *WIFI. La comunicación inalámbrica*. Retrieved from WIFI. La comunicación inalámbrica: <http://www.aulaclip.es/articulos/wifi.html>
- Desarrollo, S. N. (2013, 10 31). *Transformación de la Matriz Productiva*. Retrieved from Transformación de la Matriz Productiva: http://www.planificacion.gob.ec/wp-content/uploads/downloads/2013/01/matriz_productiva_WEBtodo.pdf
- Eduardo, C. (2016). *JUGUETE ELECTRÓNICO DIDÁCTICO, COMO ELEMENTO DE APOYO PARA LA ENSEÑANZA DE PROGRAMACIÓN A NIÑOS Y NIÑAS DE 4 A 7 AÑOS*. Ibarra: UTN.
- ELECTRONILAB. (2013, Octubre 15). *Sensor de Distancia de Ultrasonido HC-SR04*. Retrieved from Sensor de Distancia de Ultrasonido HC-SR04: <http://electronilab.co/tienda/sensor-de-distancia-de-ultrasonido-hc-sr04/>
- ETOOLS. (2016, 03 09). *Como funciona el Regulador de voltaje 7805*. Retrieved from Como funciona el Regulador de voltaje 7805: <http://www.electrontools.com/Home/WP/2016/03/09/regulador-de-voltaje-7805/>
- Gonzalez, K. (2011, Noviembre 18). *Teoría del Aprendizaje*. Retrieved from Teoría del Aprendizaje: <http://psicopedagogia19.blogspot.com/2011/11/teoria-cognoscitiva.html>
- Hernández. (2002, 08 30). *Paradigma Conductista*. Retrieved from Paradigma Conductista: http://www.cca.org.mx/profesores/cursos/cep21-tec/modulo_1/main0_31.htm
- Hernández, F. L. (2010). *Programación con Bloques*. Madrid: MacProgramadores.

- HUBOR. (2015, 07 14). *Proteus*. Retrieved from Proteus: <http://hubor-proteus.com/proteus-pcb/proteus-pcb/2-proteus.html>
- Kanda, A. (2012, 05 29). *WIFI, BLUETOOTH OR ZIGBEE WIRELESS*. Retrieved from WIFI, BLUETOOTH OR ZIGBEE WIRELESS: <http://www.kanda.com/blog/wireless/wifi-bluetooth-zigbee-wireless/>
- KELLY, D. C. (2014, Agosto 24). *Children and Technology*. Retrieved from Children and Technology: <http://www.cbeebies.com/lat-am/grown-ups/helpful-articles?article=los-ninos-y-la-tecnologia>
- Luna, C. (2012, 04 03). *DEFINICION DE ALGORITMO*. Retrieved from DEFINICION DE ALGORITMO: <http://candyluna.galeon.com/aficiones813476.html>
- Maldonado, K. (2013). *Operaciones Concretas (Piaget)*. Madrid, España: Paidós.
- Manuel, V. (2012, 10 24). *Driver L293D de Texas Instruments*. Retrieved from Driver L293D de Texas Instruments: <http://www.manuelvillasur.com/2012/10/driver-l293d-de-texas-instruments.html>
- Marín, A. (2013, 03 13). *¿Qué es Code.org?* Retrieved from ¿Qué es Code.org?: <https://alexismarin.wordpress.com/2013/03/13/que-es-code-org/>
- Martínez, J. A. (2014, Febrero 28). *ANÁLISIS DE COSTO BENEFICIO Ejemplos de análisis sector privado*. Retrieved from ANÁLISIS DE COSTO BENEFICIO Ejemplos de análisis sector privado: http://gis.jp.pr.gov/Externo_Econ/Talleres/PresentationCB_JP_ETI.pdf
- Medrano, A. (2015, 10 17). *Lenguajes de Programación*. Retrieved from Lenguajes de Programación: <http://www.lenguajes-de-programacion.com/lenguajes-de-programacion.shtml>
- Olmos, J. (2013, 10 20). *REDES INALAMBRICAS*. Retrieved from REDES INALAMBRICAS: <http://tecinal.blogspot.com/>
- PNBV, P. N. (2009-2013, 10 31). *Plan Nacional del Buen Vivir*. Retrieved from Plan Nacional del Buen Vivir: <http://www.buenvivir.gob.ec/34>
- Rancel, M. R. (2014, 07 23). *¿Qué es y para qué sirve programar y la programación de ordenadores?* Retrieved from ¿Qué es y para qué sirve programar y la programación de ordenadores?: http://aprenderaprogramar.com/index.php?option=com_content&view=article&id=35&Itemid=77
- REPSOL. (2014, 09 25). *Propiedades del plástico*. Retrieved from Propiedades del plástico: http://www.repsol.com/es_es/productos-servicios/quimica/quimica-sociedad/propiedades-plastico/
- Rodríguez, D. N. (2012, 04 10). *Situación de la Ciencia, la Tecnología e Innovación en el Ecuador*. Retrieved from Situación de la Ciencia, la Tecnología e Innovación en el Ecuador: http://www.reaces.ec/reaces/index.php?option=com_content&view=article&id=53:situacion-de-la-ciencia-la-tecnologia-e-innovacion-en-el-ecuador&catid=36:articulos&Itemid=14
- Rodriguez, T. (2012, junio 24). *Google Blockly, un lenguaje visual para aprender a programar*. Retrieved from Google Blockly, un lenguaje visual para aprender a programar:

<http://www.genbetadev.com/herramientas/google-blockly-un-lenguaje-visual-para-aprender-a-programar>

Sabas. (2015, Agosto 4). *Programando el bluetooth HC-06 y HC-05*. Retrieved from Programando el bluetooth HC-06 y HC-05: <http://blog.theinventorhouse.org/programando-el-bluetooth-hc-06-y-hc-05/>

Sincero, S. M. (2011, 03 11). *Teoría Cognitiva del Aprendizaje*. Retrieved from Teoría Cognitiva del Aprendizaje: <https://explorable.com/es/teoria-cognitiva-del-aprendizaje>

TecnologíaArea. (2013, 08 15). *MADERA*. Retrieved from MADERA: <http://www.areatecnologia.com/materiales/madera.html>

TECNOLÓGICA, I. (2015, 01 16). *La importancia de la programación en los pequeños*. Retrieved from La importancia de la programación en los pequeños: <http://www.schoolcontrol.com/blog/la-importancia-de-la-programacion-en-los-pequenos/>

Tecnologie, A. (2012, 04 18). *PLASTICOS*. Retrieved from PLATICOS: <https://sites.google.com/site/amandatecnologie/plasticos/propiedades-y-caracteristicas-de-los-plasticos>

TuAppInventor. (2012, 09 18). *¿QUE ES APPINVENTOR?* Retrieved from ¿QUE ES APPINVENTOR?: <http://www.tuappinvetorandroid.com/aprender/>

UOCTIC. (2005, 05 04). *Conductismo*. Retrieved from Conductismo: <http://uoctic-grupo6.spaces.com/Conductismo>

WhatIs. (2014, Septiembre 30). *light-emitting diode (LED)*. Retrieved from light-emitting diode (LED): <http://whatis.techtarget.com/definition/light-emitting-diode-LED>

ANEXOS

ANEXO 01. ENCUESTAS

ENCUESTA

Esta encuesta está dirigida a docentes de sexto, séptimo y octavo de educación básica con el propósito de recopilar información para la implementación de un prototipo electrónico que permita la enseñanza de fundamentos de lógica de programación en los niños de 10 a 12 años en la ciudad de Ibarra

Nombre: Ldo. Luis Rojas
Institución Educativa: Oriental Equatorial Tumbaco Colectiva Tumbaco
Año de Educación Básica: Séptimo Educación Básica

¿Con cuántos niños trabaja en su grado?

32

¿De qué edad en promedio son los niños de su grado?

11 años

En su escuela ¿se les imparte la materia de Computación o materias relacionadas con Programación?

En promoción No a partir de Octavo de Educ. Básica

De ser positiva la respuesta, ¿qué temas o actividades se imparte en esas clases?

—

¿Disponen de un laboratorio de computación?

Sí

¿Cree que se debería enseñarles a los niños fundamentos de programación?

Sí, complementaria a las demás materias

¿Cree que se debe enseñar desde la escuela a programar o fundamentos de programación?

Sí

¿Cómo cree que le ayudaría a un niño en el futuro académico si se le enseña lógica de programación?

Creo que si le ayudaría mucho en este mundo tecnológico en que
vive en los carreras universitarias nuevas

¿Usa herramientas didácticas para ayudar a los niños en su aprendizaje?, de ser positivo ¿qué tipo de herramientas?, de ser negativo ¿Por qué no las usan?

No, el docente no posee herramientas adicionales a las de
los laboratorios, para usarlos debemos usar nuestro bolsillo

¿Cree que las herramientas didácticas ayudan en el desarrollo cognitivo de los niños?

Si, con expectativas en niveles de edad inferiores los niños
muestran especial interés en los objetos que manipulan

¿Utilizan plataformas que tengan como objetivo enseñar fundamentos de programación a niños en etapa escolar como Code.org, Blockly, Scratch u otras?

No

¿Los alumnos tienen acceso a herramientas didácticas que les permita una forma diferente y entretenida de aprender??

No

¿Cree que con el pensum actual el niño está preparado para inclinarse por una vida profesional en el área de ingeniería o desarrollo tecnológico?

No, el nivel de abstracción en la ciencia tecnológica de computación es
bajo en nuestro medio

Gracias por su colaboración.

ENCUESTA

Esta encuesta está dirigida a docentes de sexto, séptimo y octavo de educación básica con el propósito de recopilar información para la implementación de un prototipo electrónico que permita la enseñanza de fundamentos de lógica de programación en los niños de 10 a 12 años en la ciudad de Ibarra

Nombre: *Msc. Atella Rosero*

Institución Educativa: *Unidad Educativa Teodoro Gómez de la Torre*

Año de Educación Básica: *Sexto Básico*

¿Con cuántos niños trabaja en su grado?

33

¿De qué edad en promedio son los niños de su grado?

10 Años

En su escuela ¿se les imparte la materia de Computación o materias relacionadas con Programación?

En mi grado no se les da la materia de Computación

De ser positiva la respuesta, ¿qué temas o actividades se imparte en esas clases?

¿Disponen de un laboratorio de computación?

Sí, lo usan los de años superiores

¿Cree que se debería enseñarles a los niños fundamentos de programación?

Me parece que sí

¿Cree que se debe enseñar desde la escuela a programar o fundamentos de programación?

Sería bueno que se les enseñara programación

¿Cómo cree que le ayudaría a un niño en el futuro académico si se le enseña lógica de programación?

Yo creo que le ayudaría en el desenvolvimiento académico en la Universidad o en el Colegio principalmente.

¿Usa herramientas didácticas para ayudar a los niños en su aprendizaje?, de ser positivo ¿qué tipo de herramientas?, de ser negativo ¿Por qué no las usan?

No uso ninguna herramienta a más del pizarrón y los marcadores. No los uso porque considero que no es necesario, los libros tienen actividades.

¿Cree que las herramientas didácticas ayudan en el desarrollo cognitivo de los niños?

Sí lo creo, en particular lo usaba con niños muy pequeños de jardín y aprendían manipulando los juguetes, rompecabezas.

¿Utilizan plataformas que tengan como objetivo enseñar fundamentos de programación a niños en etapa escolar como Code.org, Blockly, Scratch u otras?

No uso las herramientas o plataformas.

¿Los alumnos tienen acceso a herramientas didácticas que les permita una forma diferente y entretenida de aprender??

Los alumnos no usan herramientas didácticas, sus libros y vienen preparados para las actividades.

¿Cree que con el pensum actual el niño está preparado para inclinarse por una vida profesional en el área de ingeniería o desarrollo tecnológico?

Yo creo que en la escuela no, quizás con los conocimientos del colegio se prepare mejor y pueda afrontar una profesión tecnológica.

.....
.....

Gracias por su colaboración.

ENCUESTA

Esta encuesta está dirigida a docentes de sexto, séptimo y octavo de educación básica con el propósito de recopilar información para la implementación de un prototipo electrónico que permita la enseñanza de fundamentos de lógica de programación en los niños de 10 a 12 años en la ciudad de Ibarra

Nombre: *Lcdo. Luis Salazar*
Institución Educativa: *Unidad Ed. Pedagogos Gomez de la Torre*
Año de Educación Básica: *Octavo de Ed. Básica*

¿Con cuántos niños trabaja en su grado?

29 niños

¿De qué edad en promedio son los niños de su grado?

12 años

En su escuela ¿se les imparte la materia de Computación o materias relacionadas con Programación?

En octavo se les imparte computación

De ser positiva la respuesta, ¿qué temas o actividades se imparte en esas clases?

- Reconocimiento de los equipos partes y piezas de un computador, usar los herramientas de word, excel

¿Disponen de un laboratorio de computación?

Si

¿Cree que se debería enseñarles a los niños fundamentos de programación?

Si

¿Cree que se debe enseñar desde la escuela a programar o fundamentos de programación?

Si

¿Cómo cree que le ayudaría a un niño en el futuro académico si se le enseña lógica de programación?

creo que le ayudaría mucho en el colegio y en la universidad o carreras afines a las tecnologías

¿Usa herramientas didácticas para ayudar a los niños en su aprendizaje?, de ser positivo ¿qué tipo de herramientas?, de ser negativo ¿Por qué no las usan?

No uso

¿Cree que las herramientas didácticas ayudan en el desarrollo cognitivo de los niños?

Si lo creo

¿Utilizan plataformas que tengan como objetivo enseñar fundamentos de programación a niños en etapa escolar como Code.org, Blockly, Scratch u otras?

No se usa

¿Los alumnos tienen acceso a herramientas didácticas que les permita una forma diferente y entretenida de aprender??

En las clases de computación les enseñan a los padres ingresar a la plataforma de scratch

¿Cree que con el pensum actual el niño está preparado para inclinarse por una vida profesional en el área de ingeniería o desarrollo tecnológico?

No lo creo los alumnos universitarios actuales han tenido mucho problema en la programación por lo que en la

universidad

Gracias por su colaboración.

ANEXO 02. CÓDIGO DEL PROTOTIPO

Este código se escribió con la colaboración del Ing. Eduardo Cartagena, por lo que se lo considera como aporte y referencia.

```
// -----  
// ----- DECLARACIÓN DE VARIABLES -----  
  
const int MOTOR_IZQ_ADELANTE = 5;    // SE DECLARA AL PIN 5 COMO SALIDA PARA AVANZAR CON EL MOTOR IZQUIERDO  
const int MOTOR_IZQ_ATRAS = 6;      // SE DECLARA AL PIN 6 COMO SALIDA PARA RETROCEDER CON EL MOTOR IZQUIERDO  
const int MOTOR_DER_ADELANTE = 10;   // SE DECLARA AL PIN 10 COMO SALIDA PARA AVANZAR CON EL MOTOR DERECHO  
const int MOTOR_DER_ATRAS = 11;     // SE DECLARA AL PIN 11 COMO SALIDA PARA RETROCEDER CON EL MOTOR  
DERECHO  
  
const int echo = 4;                  // SE DECLARA AL PIN 4 COMO RECEPTOR ECO DEL SENSOR ULTRASÓNICO  
const int trigger = 3;               // SE DECLARA AL PIN 3 COMO EMISOR TRIGGER DEL SENSOR ULTRASÓNICO  
float distancia = 5;                 // SE INICIA UNA DISTANCIA FUERA DEL RANGO ESTABLECIDO PARA QUE EJECUTE EL CICLO  
  
int i, j, k, l, m, n, x, z;          // SE DECLARAN VARIABLES PARA CICLOS  
int var1=0;                           // SE DECLARA LA VARIABLE var1 INICIALIZADA EN CERO  
int var2=0;                           // SE DECLARA LA VARIABLE var2 INICIALIZADA EN CERO  
int var3=0;                           // SE DECLARA LA VARIABLE var3 INICIALIZADA EN CERO  
int pos1;                             // SE DECLARA LA VARIABLE pos1 COMO POSICIÓN DEL VECTOR funcion1  
int pos2;                             // SE DECLARA LA VARIABLE pos2 COMO POSICIÓN DEL VECTOR funcion2  
int pos3;                             // SE DECLARA LA VARIABLE pos3 COMO POSICIÓN DEL VECTOR funcion3
```

```

char vectorPrincipal[16];          // SE DECLARA PARA ALMACENAR TODOS LOS DATOS RECIBIDOS
char funcion1[16];                // SE ALMACENA LOS DATOS DEL CICLO DE 3 REPETICIONES
char funcion2[16];                // SE ALMACENA LOS DATOS DEL CICLO DE 2 REPETICIONES
char funcion3[16];                // SE ALMACENA LOS DATOS A EJECUTAR CUANDO ENCONTRAMOS UN OBSTÁCULO
char dato;                        // SE DECLARA EL DATO PARA SER EJECUTADO POR LAS INSTRUCCIONES
char iniciar;                     // SE DECLARA UNA VARIABLE PARA INICIAR EL PROCESO

// ----- //

void setup() {
  Serial.begin(9600);              // SE HABILITA LA COMUNICACIÓN SERIAL
  pinMode(trigger, OUTPUT);        // SE ESTABLECE EL PIN trigger COMO SALIDA PARA EL SENSOR ULTRASÓNICO
  pinMode(echo, INPUT);           // SE ESTABLECE EL PIN echo COMO ENTRADA PARA EL SENSOR ULTRASÓNICO
  pinMode(MOTOR_IZQ_ADELANTE, OUTPUT); // SE ESTABLECE LA SALIDA PARA EL MOTOR IZQUIERDO HACIA ADELANTE
  pinMode(MOTOR_IZQ_ATRAS, OUTPUT); // SE ESTABLECE LA SALIDA PARA EL MOTOR IZQUIERDO HACIA ATRAS
  pinMode(MOTOR_DER_ADELANTE, OUTPUT); // SE ESTABLECE LA SALIDA PARA EL MOTOR DERECHO HACIA ADELANTE
  pinMode(MOTOR_DER_ATRAS, OUTPUT); // SE ESTABLECE LA SALIDA PARA EL MOTOR DERECHO HACIA ATRAS
  delay(2000);                    // SE ESTABLECE UN RETARDO DE 2 SEGUNDOS
}

// ----- //
// ----LECTURA DE DATOS ENTRANTES EN LA COMUNICACIÓN SERIAL Y ALMACENAMIENTO EN EL VECTOR PRINCIPAL---- //

```

```

void loop() {
while(true) {
if (Serial.available() > 0) { // SE VERIFICA DATOS EN LA ENTRADA SERIAL
char letra = Serial.read(); // SE LEE EL DATO RECIBIDO Y SE ALMACENA EN LA VARIABLE letra
if (letra == 'A' || letra == 'B' || letra == 'C' || letra == 'D' || letra == 'E' || letra == 'F' || letra == '{' || letra == '}' ) {
// SE VERIFICA UNO POR UNO LOS DATOS ENTRANTES

vectorPrincipal[i] = letra; // SE ALMACENA EL DATO EN EL VECTOR PRINCIPAL
delay(100);
i++; // SE AUMENTA EN UNO LA POSICION DE ALMACENAMIENTO EN EL VECTOR PRINCIPAL
//Serial.println(i);
}
else {
if (letra == '0') // SE VERIFICA SI EL DATO ENTRANTE ES 0 (CERO)
{
iniciar = '1'; // SE ESTABLECE COMO '1' A LA VARIABLE INICIAR ('1' INICIA EL PROCESO)
}
}
}

// ----- //

```

```

if (iniciar == '1') {           // SE ESPERA QUE LA VARIABLE iniciar SEA '1'
  LeerCiclo3();                // SE LLAMA A LA SUBROUTINA LeerCiclo3()
  LeerCiclo2();                // SE LLAMA A LA SUBROUTINA LeerCiclo2()
  Obstaculo();                 // SE LLAMA A LA SUBROUTINA obstaculo()
  for (z = 0; z < i; z++) {    // SE LEE EL VECTOR PRINCIPAL POSICIÓN POR POSICIÓN
    if (vectorPrincipal[z] != 'D' && vectorPrincipal[z] != 'E' && vectorPrincipal[z] != 'F' ) {
      // SE VERIFICA QUE LOS DATOS SEAN A, B o C PARA EJECUTAR LAS INSTRUCCIONES

      dato = vectorPrincipal[z]; // SE VERIFICA LA CONDICIÓN Y SE ALMACENA LOS CARACTERES EN LA VARIABLE dato
      delay(10);                 // SE SETEA UN RETARDO ANTES DE EJECUTAR LAS INSTRUCCIONES
      instrucciones();           // SE EJECUTA LAS INSTRUCCIONES
      var1=0; var2=0;var3=0;     // SE ENCERAN LAS VARIABLES
    }

    else                         // CASO CONTRARIO
    {
      if (vectorPrincipal[z] == 'E') { // SI EL CARACTER ALMACENADO ES E
        for (l = 0; l < 3; l++) { // SE EJECUTA UN CICLO PARA LEER EL VECTOR funcion1 3 VECES
          for (x = 0; x < k; x++) {
            dato = funcion1[x]; // LEEMOS EL DATO EN LA POSICIÓN x Y LE ASIGNAMOS A dato PARA EJECUTAR LAS
            INSTRUCCIONES
            instrucciones();     // SE EJECUTA LAS INSTRUCCIONES
            var1=0; var2=0;var3=0; // SE ENCERAN LAS VARIABLES
          }
        }
      }
    }
  }
}

```

```

    delay(200);
}
}
}

else // CASO CONTRARIO
{
if (vectorPrincipal[z] == 'D') { // SI EL CARACTER ALMACENADO ES D
for (l = 0; l < 2; l++) { // SE EJECUTA UN CICLO PARA LEER EL VECTOR funcion2 2 VECES
for (x = 0; x < m; x++) {
dato = funcion2[x]; // LEEMOS EL DATO EN LA POSICIÓN x Y LE ASIGNAMOS A dato PARA EJECUTAR
LAS INSTRUCCIONES
instrucciones(); // SE EJECUTA LAS INSTRUCCIONES
var1=0; var2=0;var3=0; // SE ENCERAN LAS VARIABLES
delay(200);
}
}
}

else // CASO CONTRARIO
{
if (vectorPrincipal[z] == 'F') { // SI EL CARACTER ALMACENADO ES F

```



```

while (distancia > 0.30) { // SE EJECUTA EL CICLO HASTA QUE LA DISTANCIA CON EL OBSTÁCULO SEA MAYOR A 30
CENTIMETROS

    digitalWrite(trigger, LOW); // SE DA UN PULSO DE 5 MICROSEGUNDOS PARA EL SENSOR DE DISTANCIA
    delayMicroseconds(5);
    digitalWrite(trigger, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigger, LOW);
    distancia = pulseIn(echo, HIGH); // SE MIDE EL TIEMPO DEL PULSO ACTIVADO PARA CONOCER LA DISTANCIA
    distancia = distancia * 0.0001657; // CON ESTA OPERACIÓN SE OBTIENE LA DISTANCIA EN CENTÍMETROS
    var=0; var2=0;var3=0; // SE ENCERAN LAS VARIABLES
};
for (x = 0; x < n; x++) { // CUANDO SE DETECTA UN OBSTÁCULO SE LEE LOS DATOS ASIGNADOS
    dato = funcion3[x]; // LEEMOS EL DATO EN LA POSICIÓN x Y LE ASIGNAMOS A dato PARA EJECUTAR
LAS INSTRUCCIONES
    instrucciones(); // SE EJECUTA LAS INSTRUCCIONES
    var1=0; var2=0;var3=0; // SE ENCERAN LAS VARIABLES
    delay(200);
};
}
}
}

```

```

    }
}

i = 0;           // SE REINICIAN LOS CONTADORES
k = 0;
m = 0;
n = 0;
distancia = 5;
iniciar = '0';   // SE ASIGNA UN VALOR DIFERENTE DE '1' PARA SALIR DEL CICLO
}
}
}

// ----- //
// -----METODO PARA ALMACENAR LAS INSTRUCCIONES DE UN CICLO DE 3 REPETICIONES----- //

void LeerCiclo3 ()
{
    for (j = 0; j <= i; j++) {           // ESTE CICLO LEE EL VECTOR PRINCIPAL
        if (vectorPrincipal[j] == 'E' && vectorPrincipal[j + 1] == '{') // SE COMPARA SI EL CARACTER ES 'E' Y EL CARACTER SIGUIENTE
            ES '{'
            {
                // SE ALMACENA LOS DATOS QUE ESTÁN DENTRO DE LAS
                LLAVES EN EL VECTOR
            }
        }
    }
}

```

```

    pos1 = j + 1;           // LA VARIABLE pos1 ALMACENA LA POSICION SIGUIENTE
    while (vectorPrincipal[pos1] != '}') {           // SE EJECUTA PARA ALMACENAR LOS DATOS HASTA QUE SE CIERRE LAS
LLAVES
        funcion1[k] = vectorPrincipal[pos1];       // SE ALMACENA EN EL VECTOR funcion1 LOS DATOS DENTRO DE LAS
LLAVES
        delay(10);
        vectorPrincipal[pos1] = ' ';               // SE BORRA EL DATO DEL vectorPrincipal QUE ESTA GUARDADO EN funcion1
        delay(10);
        pos1++;                                   // AUMENTA EN 1 LA POSICIÓN DE LECTURA DEL vectorPrincipal
        k++;                                       // EL CONTADOR AUMENTA EN UNO
    }
}
}
}

case 'C':
    delay(350);
    while(var3<122){
delayMicroseconds(300);
digitalWrite(MOTOR_IZQ_ADELANTE,HIGH);
digitalWrite(MOTOR_DER_ADELANTE, HIGH);
digitalWrite(MOTOR_IZQ_ATRAS, LOW);
digitalWrite(MOTOR_DER_ATRAS,LOW);
delayMicroseconds(3000);

```

```
digitalWrite(MOTOR_IZQ_ADELANTE, LOW);
digitalWrite(MOTOR_IZQ_ATRAS, LOW);
digitalWrite(MOTOR_DER_ADELANTE, LOW);
digitalWrite(MOTOR_DER_ATRAS, LOW);
var3++;
Serial.println(var3);
delayMicroseconds(100);
}
break;

default:
digitalWrite(MOTOR_IZQ_ADELANTE, LOW);
digitalWrite(MOTOR_IZQ_ATRAS, LOW);
digitalWrite(MOTOR_DER_ADELANTE, LOW);
digitalWrite(MOTOR_DER_ATRAS, LOW);
break;
}
} (Eduardo, 2016)
```

ANEXO 03. SECUENCIAS CONVERTIDAS EN LÍNEAS DE CÓDIGO SIMPLE

Nº: Secuencia

- Avanzar
- Avanzar
- Avanzar
- Meta

- Avanzar
- Giro 180°
- Avanzar
- Giro 0°
- Avanzar
- Meta

- Avanzar
- Giro 180°
- Repetición X3
- {
- Avanzar
- }
- Giro 0°
- Avanzar
- Meta

Secuencias

Avanzar
 avanzar
 avanzar
 Meta

Avanzar
 giro 180
 repetición X3
 {
 avanzar
 }
 giro 0°
 Avanzar
 Meta

Mis Secuencias

- Avanzar
- Giro 180
- Avanzar
- Giro 0°
- Avanzar
- Meta

- Decisión
- {
- giro 0°
- avanzar
- avanzar
- giro 180°
- avanzar
- }
- Meta

- Repetición por tres
 - {
 - avanzar
 - giro 180°
 - avanzar
 - giro 0°
 - }
 - avanzar
 - meta
-
- avanzar
 - giro 180°
 - avanzar
 - avanzar
 - giro 0°
 - avanzar
 - meta

- Mis secuencias
- Decision
- {
 - giro 0°
 - avanzar
 - avanzar
 - giro 180°
 - avanzar
 - }
 - meta
- Repetición por dos
- {
 - avanzar
 - }
 - giro 180°
 - avanzar
 - meta

ANEXO 04. RESPALDO FOTOGRÁFICO DE RESULTADOS







ANEXO 05. RESPALDO DE PRECIOS

ARIAS ERAZO JONATHAN LEONARDO

Dirección: Hernán González de Saa 25-151 y Princesa Paccha
 Cel.: 095 955 5577 Ibarra - Ecuador

RUC: 1003688726001 AUT. SRI: 1117501224

SERIE 001-001-		FACTURA		0000019	
Cliente: Wilman Suarez F		Fecha: 15-10-2015			
RUC/CI: 1003412666001		Telf.: 2016321			
Dirección: Ibarra					
Cant.	DESCRIPCIÓN	V. Unit.	V. TOTAL		
1	Celular Smartphone Sky Devices G. 04 fene	228,80	228,80		
	353445660348014				
DOCUMENTO CATEGORIZADO: NO		Subtotal \$ 228,80			
		0% IVA			
Entregué Conforme		12% IVA 27,46			
		TOTAL \$ 260,00			
Recibí Conforme					

SALAS VACA PATRICIO ANDRÉS / GRÁFICAS DEL NORTE - IBARRA TELFS.: 062 611 623 / RUC: 1001774619001/AUT. 2173
 DEL 001 AL 025 FECHA DE AUT. 31-AGOSTO 2015 CADUCA 30-NOVIEMBRE 2015 ORIGINAL: ADQUIRENTE, COPIA: EMISOR

Video Willi Secuence