



**PONTIFICIA
UNIVERSIDAD
CATOLICA
DEL ECUADOR**

SEDE AMBATO

ESCUELA DE INGENIERÍA DE SISTEMAS

TEMA:

**“DESARROLLO Y CONSTRUCCIÓN DE UN HUMANOIDE UTILIZANDO LAS
HERRAMIENTAS PROPORCIONADAS POR EL EQUIPO DE ROBÓTICA
LEGO MINDSTORM NXT”**

**DISERTACIÓN DE GRADO PREVIA LA OBTENCIÓN DEL TÍTULO DE
INGENIERÍA DE SISTEMAS**

AUTOR:

EDMUNDO JOSÉ SANTACRUZ LÓPEZ

ASESOR:

ING. M.Sc JANIO JADÁN



AMBATO – ECUADOR

OCTUBRE 2008

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

SEDE AMBATO

ESCUELA DE INGENIERÍA DE SISTEMAS

HOJA DE APROBACION

TEMA:

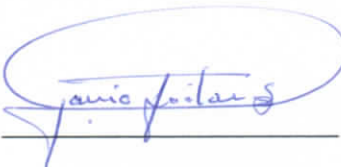
“DESARROLLO Y CONSTRUCCIÓN DE UN HUMANOIDE UTILIZANDO
LAS HERRAMIENTAS PROPORCIONADAS POR EL EQUIPO DE ROBÓTICA
LEGO MINDSTORM NXT”

AUTOR:

EDMUNDO JOSÉ SANTACRUZ LÓPEZ

Janio Lincon Jadán Guerrero, Ing. Msc.

DIRECTOR DE TESIS / DISERTACIÓN / EXAM.

f. 

Marco Polo Silva Segovia, Ing. Msc.

CALIFICADOR

f. 

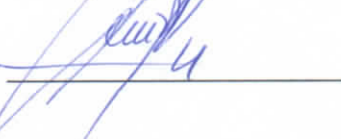
Pablo Ernesto Montalvo Jaramillo, Ing. Msc.

CALIFICADOR

f. 

Santiago Alejandro Acurio Maldonado, Ing.

DIRECTOR UNIDAD ACADÉMICA

f. 

Pablo Poveda Mora, Ab.

SECRETARIO GENERAL PUCESA

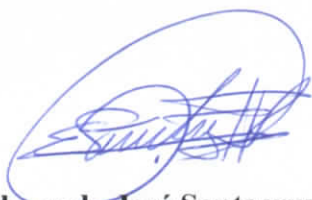
f. 

SECRETARIA GENERAL
PROCURADURIA

DECLARACIÓN DE AUTENTICIDAD Y RESPONSABILIDAD

Yo, Edmundo José Santacruz López portador de la cédula de ciudadanía No. 180344792-7 declaro que los resultados obtenidos en la investigación que presento como informe final, previo la obtención del título de Ingeniería de Sistemas son absolutamente originales, auténticos y personales.

En tal virtud, declaro que el contenido, las conclusiones y los efectos legales y académicos que se desprenden del trabajo propuesto de investigación y luego de la redacción de este documento son y serán de mi sola y exclusiva responsabilidad legal y académica.



Edmundo José Santacruz López

CI. 180344792-7

DEDICATORIA

Esta Disertación va dedicada a mis padres por darme su apoyo incondicional y un gran ejemplo de lucha y responsabilidad, a mis hermanas que con su apoyo, comprensión y aliento han sido el pilar fundamental en mi existir, a mi enamorada y amistades los cuales me dieron su ayuda en todo momento para estar más cerca de mis metas profesionales. Y a la PUCESA por permitirme forjar mi conocimiento dentro de sus aulas y hacer de mí un gran profesional.

“El éxito siempre ha sido fácil
de medir: es la distancia entre
el punto de partida de una
persona y su mayor logro”

Michael Korda

AGRADECIMIENTO

A mis padres por haberme brindado su apoyo y los recursos necesarios para obtener este Título Superior y haberme guiado en el camino.

A mis hermanas por ser mis mejores amigas y compañeras y alentarme durante todo el transcurso de mi carrera.

A Janio Jadán, director de esta tesis y amigo gracias por ayudarme con su conocimiento y voluntad para la culminación de este proyecto.

A todos los profesores que durante estos cuatro años compartieron conmigo sus conocimientos y experiencias.

A mis compañeros y amigos que durante la vida universitaria fueron como mis hermanos.

Finalmente un agradecimiento a la Pontificia Universidad Católica del Ecuador sede Ambato, en especial a la Escuela de Ingeniería de Sistemas, por permitirme forjarme como ser humano y profesional.

RESUMEN

Hoy en día los robots son ideales para trabajos que requieren movimientos repetitivos y precisos. Una desventaja para las empresas es que los humanos necesitan descansos, salarios, comida, dormir, y una área segura para trabajar, los robots no. La fatiga y aburrimiento de los humanos afectan directamente a la producción de una compañía, los robots nunca se aburren por lo tanto su trabajo va a ser el mismo desde que abra la compañía a las 8:00 AM hasta las 6:00 PM. Por eso la tendencia es que en un futuro la mayoría del trabajo pesado sea realizado por un robot ahorrando tiempo y dinero.

En mis años de estudio dentro de la universidad noté la falta de equipo actual para la enseñanza del área de la robótica haciendo difícil su aprendizaje por esa razón el desarrollo de este proyecto ayudará a tener un material de apoyo en cuanto a la teoría y una gran aporte práctico por el equipo que se utilizará durante la realización de esta tesis. El cual es el equipo de robótica Lego Mindstorm NXT tiene mucha flexibilidad lo cual hace que este equipo no se centre en la construcción de algo fijo sino que se puede obtener infinidad de modelos de robots y a cada modelo se le puede implementar igualmente varios tipos de comportamientos.

Finalmente el lenguaje de programación que este equipo utiliza es muy dinámico y didáctico, permite configurar varias opciones dentro de cada uno de los componentes, posee herramientas de programación como ciclos, condiciones, variables, etc. El conjunto de todas estas características nos brinda un lenguaje muy fácil de entender pero a la vez con una amplia gama de opciones. En la tesis se incluye detalladamente la construcción y programación de uno de los modelos más

complejos (androide), este material podrá ser usado para la enseñanza y el uso del equipo de robótica.

ABSTRACT

Nowadays, robots are ideal for works that require repetitive and precise movements. A disadvantage for the companies is that humans need rest, wages, food, sleep, and a secure area to work, robots do not. Human's fatigue and boredom affect the production of a company directly, but robots never get bored. Therefore, their work will be the same from 8:00 AM to 6:00 PM. For that reason the tendency is that in the future most of the heavy work will be carried out by a robot by saving time and money.

In my years of study in this university, I have noticed the lack of up to date tools for the teaching of robotics, making learning difficult, for that reason the development of this project will help to have support material for the theory and a great practical contribution for the tools that will be used during the research of this thesis.

Robotics tool Lego Mindstorm NXT has a lot of flexibility which makes that this tool not be centered in the construction of something fixed, but rather give the possibility of obtaining an infinite of robot models and each pattern can be implemented at the same time with several types of behaviors.

Finally, the programming language that this tool uses is very dynamic and didactic. It allows configuring several options inside each one of the components. It possesses programming tools like cycles, conditions, variables, etc. The group of all these characteristics offers us a very easy language to understand but at the same time with a wide range of options. In detailed construction and programming detailed of one of the models most complex (android), are included in this thesis. This material will be available for teaching and the use of the robotics team.

TABLA DE CONTENIDOS

DECLARACIÓN DE AUTENTICIDAD Y RESPONSABILIDAD	iii
DEDICATORIA	iv
AGRADECIMIENTO	v
RESUMEN.....	vi
ABSTRACT	viii
TABLA DE CONTENIDOS.....	ix
TABLA DE FIGURAS.....	xiii
TABLAS	xxiii
INTRODUCCIÓN	1
1. PLAN DE TESIS	2
1.1 Problema de la investigación.	2
1.1.1 Antecedentes.....	2
1.1.2 Significado del problema.....	2
1.1.3 Definición del problema.	2
1.1.4 Planteamiento del tema.....	3
1.1.5 Delimitación del tema.....	3
1.2 Marco teórico.....	4
1.3 Objetivos.....	6
1.3.1 General.....	6
1.3.2 Específicos.....	6
1.4 Metodología de trabajo.....	6
1.5 Justificación.....	6
1.6 Bibliografía preliminar.....	7
1.7 Recursos.....	8
1.7.1 Recursos humanos.....	8
1.7.2 Recursos materiales.....	8
1.7.3 Recursos económicos.....	8
2. MARCO TEÓRICO.....	9
2.1 Inteligencia Artificial.....	9
2.1.1 Robótica.....	10
2.1.2 Historia de la Robótica.....	11
2.1.2.1 Actualidad de la Robótica.....	14
2.2 Electrónica Digital.....	14

2.2.1 Herramientas.....	15
Los sistemas digitales tienen las siguientes herramientas:.....	15
2.3 PIC (Programable Integrated Circuit).....	17
2.3.1 Estructura.....	18
2.3.1.1 Tipos.....	19
2.3.2 Programación de PIC.....	20
2.4 Sensores.....	21
2.4.1 Sensor de Luz.....	22
2.4.2 Sensor de Tacto o Pulsador.....	25
2.4.3 Sensor de Sonido.....	26
2.4.4 Sensor de Ultrasonido.....	28
2.4.5 Sensor de Color.....	30
2.5 Servomotores.....	33
2.6 Movimiento del cuerpo humano.....	34
2.6.1 Caminar.....	35
2.7 Humanoides.....	36
2.7.1 Características.....	36
3. DESARROLLO Y CONSTRUCCIÓN DE UN HUMANOIDE.....	37
3.1 LEGO Mindstorm NXT.....	37
3.1.1 Elementos.....	38
3.1.2 Programación.....	41
3.1.2.1 Calibración de Sensores.....	42
3.1.2.1.1 Función de Calibración de Sensores.....	43
3.1.2.2 Cable de datos.....	44
3.1.2.2.1 Dibujar un cable de datos.....	45
3.1.2.2.2 Borrar un cable de datos.....	45
3.1.2.2.3 Entrada y salida.....	45
3.1.2.2.4 Tipos de datos.....	46
3.1.2.2.5 Cable de datos rotos.....	46
3.1.2.2.6 Rango de datos de un punto.....	47
3.1.2.3 Errores de cables de datos rotos.....	47
3.1.2.3.1 Tipo de dato incompatible.....	47
3.1.2.3.2 Ciclo.....	48
3.1.2.3.3 Entrada perdida.....	49

3.1.2.3.4 Exceso de entradas.....	50
3.1.2.4 Administración de archivos y memoria del Bloque NXT.	51
3.1.2.5 Secuencia Beam.....	54
3.1.2.6 Punto de Inicio.....	55
3.1.2.7 Bloque de Pantalla.....	56
3.1.2.8 Bloque de Repetición.....	58
3.1.2.9 Bloque de Movimiento.....	58
3.1.2.10 Bloque de Sonido.....	61
3.1.2.11 Bloque Switch.....	61
3.1.2.12 Bloque de Espera.....	66
3.1.2.13 Bloque de Interrupción.....	68
3.1.2.14 Bloque de Motor.....	69
3.1.2.15 Bloque del Sensor de Luz.....	70
3.1.2.16 Bloque del Sensor de Sonido.....	72
3.1.2.17 Bloque del Sensor de Tacto.....	73
3.1.2.18 Bloque del Sensor de Ultrasonido.....	74
3.1.2.19 Bloque Timer.....	75
3.2 Humanoide.....	76
3.2.1 Caminar.....	76
3.2.1.1 Construcción.....	76
3.2.1.2 Programación.....	99
3.2.2 Cuerpo.....	104
3.2.2.1 Construcción.....	104
3.2.2.2 Programación.....	125
3.2.3 Vista.....	127
3.2.3.1 Construcción.....	127
3.2.3.2 Programación.....	131
3.2.4 Oído y Tacto.....	136
3.2.4.1 Construcción.....	137
3.2.4.2 Programación.....	143
3.2.5 Luz.....	150
3.2.5.1 Construcción.....	150
3.2.5.2 Programación.....	152

3.2.6 Color.....	159
3.2.6.1 Construcción.....	159
3.2.6.2 Programación.....	160
4. VALIDACIÓN Y VERIFICACIÓN DE RESULTADOS.....	177
4.1 Validación.....	177
5. CONCLUSIONES Y RECOMENDACIONES.....	178
5.1 Conclusiones.....	178
5.2 Recomendaciones.....	178
BIBLIOGRAFÍA.....	180
ACRÓNIMOS.....	181
ANEXOS.....	183
OTROS MODELOS.....	184

TABLA DE FIGURAS

Figura 1. Evolución de la Robótica.....	12
Figura 2. Brazo Robótico	13
Figura 3. Estructura de un PIC	18
Figura 4. Entorno de Picaxe	21
Figura 5. Entorno de Cocodrile.....	21
Figura 6. Sensor de Luz del kit Lego Mindstorm NXT	22
Figura 7. Estructura del Sensor de Luz.....	24
Figura 8. Sensor de Tacto del kit Lego Mindstorm NXT	25
Figura 9. Estados del Sensor de Tacto	25
Figura 10. Estructura del Sensor de Tacto	25
Figura 11. Sensor de Sonido del kit Lego Mindstorm NXT.....	26
Figura 12. Estructura del Sensor de Sonido	27
Figura 13. Sensor de Ultrasonido del kit Lego Mindstorm NXT	28
Figura 14. Estructura del Sensor de Ultrasonido	29
Figura 15. Sensor de Color del kit Lego Mindstorm NXT	30
Figura 16. Estructura del Sensor de Color	32
Figura 17. Servomotor del kit Lego Mindstorm NXT	33
Figura 18. Estructura de un Servomotor	34
Figura 19. Humanoide.....	36
Figura 20. Bloque NXT	37
Figura 21. Componentes NXT	41
Figura 22. Aplicación incluida en el kit.....	42
Figura 23. Calibración de Sensores.....	43
Figura 24. Creación de un cable de datos	45
Figura 25. Entradas y salidas	46
Figura 26. Venta de ayuda	46
Figura 27. Error - Tipo de dato incompatible	48
Figura 28. Corrección - Tipo de dato incompatible	48
Figura 29. Error - Ciclo.....	49
Figura 30. Corrección - Ciclo	49
Figura 31. Error – Entrada perdida	50
Figura 32. Corrección – Entrada perdida	50
Figura 33. Error – Exceso de entradas	51
Figura 34. Corrección – Exceso de entradas.....	51
Figura 35. Acceso al bloque NXT.....	51

Figura 36. Ventana NXT.....	52
Figura 37. Secuencia Beam.....	54
Figura 38. Varias Secuencias Beam.....	54
Figura 39. Secuencia Beam Paralela.....	55
Figura 40. Secuencia Beam desde el centro del programa.....	55
Figura 41. Punto de Inicio.....	56
Figura 42. Bloque de Pantalla.....	56
Figura 43. Menú de Configuración Bloque de Pantalla.....	57
Figura 44. Bloque de Repetición.....	58
Figura 45. Bloque de Movimiento.....	59
Figura 46. Menú de Configuración del Bloque de Movimiento.....	60
Figura 47. Bloque de Sonido.....	61
Figura 48. Bloque Switch.....	62
Figura 49. Apariencia.....	62
Figura 50. Configuración Bloque Switch (Value).....	63
Figura 51. Configuración Bloque Switch (Sensor de Tacto).....	64
Figura 52. Configuración Bloque Switch (Sensor de Sonido).....	64
Figura 53. Configuración Bloque Switch (Sensor de Luz).....	65
Figura 54. Configuración Bloque Switch (Sensor de Ultrasonido).....	65
Figura 55. Configuración Bloque Switch (Bloque NXT).....	66
Figura 56. Bloque de Espera.....	66
Figura 57. Configuración Bloque de Espera (Sensor de Tacto).....	66
Figura 58. Configuración Bloque de Espera (Sensor de Sonido).....	67
Figura 59. Configuración Bloque de Espera (Sensor de Luz).....	67
Figura 60. Configuración Bloque de Espera (Sensor de Ultrasonido).....	68
Figura 61. Configuración Bloque de Espera (Bloque NXT).....	68
Figura 62. Bloque de Interrupción.....	68
Figura 63. Bloque de Motor.....	69
Figura 64. Descripción Bloque de Motor.....	69
Figura 65. Configuración Bloque de Motor.....	70
Figura 66. Descripción Bloque del Sensor de Luz.....	71
Figura 67. Configuración Bloque del Sensor de Luz.....	71
Figura 68. Descripción Bloque del Sensor de Sonido.....	72
Figura 69. Configuración Bloque del Sensor de Sonido.....	73
Figura 70. Descripción Bloque del Sensor de Tacto.....	73
Figura 71. Configuración Bloque del Sensor de Tacto.....	74

Figura 72. Descripción Bloque del Sensor de Ultrasonido	74
Figura 73. Configuración Bloque del Sensor de Ultrasonido	75
Figura 74. Descripción Bloque Timer.....	75
Figura 75. Configuración Bloque Timer.....	76
Figura 76. Elementos Paso 1 (Caminar)	77
Figura 77. Ensamblaje Paso 1 (Caminar).....	77
Figura 78. Elementos Paso 2 (Caminar)	77
Figura 79. Ensamblaje Paso 2 (Caminar).....	78
Figura 80. Elementos Paso 3 (Caminar)	78
Figura 81. Ensamblaje Paso 3 (Caminar).....	78
Figura 82. Elementos Paso 4 (Caminar)	79
Figura 83. Ensamblaje Paso 4 (Caminar).....	79
Figura 84. Elementos Paso 5 (Caminar)	79
Figura 85. Ensamblaje Paso 5 (Caminar).....	80
Figura 86. Elementos Paso 6 (Caminar)	80
Figura 87. Ensamblaje Paso 6 (Caminar).....	81
Figura 88. Elementos Paso 7 (Caminar)	81
Figura 89. Ensamblaje Paso 7 (Caminar).....	82
Figura 90. Ensamblaje Paso 8 (Caminar).....	82
Figura 91. Ensamblaje Paso 8 (Caminar).....	83
Figura 92. Elementos Paso 9 (Caminar)	83
Figura 93. Ensamblaje Paso 9 (Caminar).....	83
Figura 94. Elementos Paso 10 (Caminar)	84
Figura 95. Ensamblaje Paso 10 (Caminar).....	84
Figura 96. Elementos Paso 11 (Caminar)	85
Figura 97. Ensamblaje Paso 11 (Caminar).....	85
Figura 98. Elementos Paso 12 (Caminar)	85
Figura 99. Ensamblaje Paso 12 (Caminar).....	86
Figura 100. Elementos Paso 13 (Caminar)	86
Figura 101. Ensamblaje Paso 13 (Caminar).....	86
Figura 102. Elementos Paso 14 (Caminar)	87
Figura 103. Ensamblaje Paso 14 (Caminar).....	87
Figura 104. Elementos Paso 15 (Caminar)	88
Figura 105. Ensamblaje Paso 15 (Caminar).....	88
Figura 106. Elementos Paso 16 (Caminar)	88
Figura 107. Ensamblaje Paso 16 (Caminar).....	89

Figura 108. Estructura Parte 1 (Caminar)	89
Figura 109. Estructura Parte 2 (Caminar)	90
Figura 110. Elementos Paso 18 (Caminar)	90
Figura 111. Ensamblaje Paso 18 (Caminar).....	91
Figura 112. Elementos Paso 19 (Caminar)	91
Figura 113. Ensamblaje Paso 19 (Caminar).....	91
Figura 114. Elementos Paso 20 (Caminar)	92
Figura 115. Ensamblaje Paso 20 (Caminar).....	92
Figura 116. Elementos Paso 21 (Caminar)	93
Figura 117. Ensamblaje Paso 21 (Caminar).....	93
Figura 118. Elementos Paso 22 (Caminar)	93
Figura 119. Ensamblaje Paso 22 (Caminar).....	94
Figura 120. Elementos Paso 23 (Caminar)	94
Figura 121. Ensamblaje Paso 23 (Caminar).....	95
Figura 122. Elementos Paso 24 (Caminar)	95
Figura 123. Ensamblaje Paso 24 (Caminar).....	95
Figura 124. Elementos Paso 25 (Caminar)	96
Figura 125. Ensamblaje Paso 25 (Caminar).....	96
Figura 126. Elementos Paso 26 (Caminar)	96
Figura 127. Ensamblaje Paso 27 (Caminar).....	97
Figura 128. Elementos Paso 28 (Caminar)	97
Figura 129. Ensamblaje Paso 28 (Caminar).....	98
Figura 129. Elementos Paso 29 (Caminar)	98
Figura 130. Ensamblaje Paso 29 (Caminar).....	98
Figura 131. Elementos Paso 30 (Caminar)	99
Figura 132. Ensamblaje Paso 30 (Caminar).....	99
Figura 133. Paso 1 Programación (Caminar).....	100
Figura 134. Paso 2 Programación (Caminar).....	100
Figura 135. Paso 3 Programación (Caminar).....	100
Figura 136. Paso 4 Programación (Caminar).....	101
Figura 137. Paso 5 Programación (Caminar).....	101
Figura 138. Paso 6 Programación (Caminar).....	101
Figura 139. Paso 7 Programación (Caminar).....	102
Figura 140. Paso 8 Programación (Caminar).....	102
Figura 141. Paso 9 Programación (Caminar).....	102
Figura 142. Paso 10 Programación (Caminar).....	103

Figura 143. Paso 11 Programación (Caminar).....	103
Figura 144. Paso 12 Programación (Caminar).....	103
Figura 145. Paso 13 Programación (Caminar).....	104
Figura 146. Paso 14 Programación (Caminar).....	104
Figura 147. Elementos Paso 1 (Cuerpo)	105
Figura 148. Ensamblaje Paso 1 (Cuerpo).....	105
Figura 149. Elementos Paso 2 (Cuerpo)	105
Figura 150. Ensamblaje Paso 2 (Cuerpo).....	106
Figura 151. Elementos Paso 3 (Cuerpo)	106
Figura 152. Ensamblaje Paso 3 (Cuerpo).....	106
Figura 153. Elementos Paso 4 (Cuerpo)	107
Figura 154. Ensamblaje Paso 4 (Cuerpo).....	107
Figura 155. Elementos Paso 5 (Cuerpo)	107
Figura 156. Ensamblaje Paso 5 (Cuerpo).....	107
Figura 157. Elementos Paso 6 (Cuerpo)	108
Figura 158. Ensamblaje Paso 6 (Cuerpo).....	108
Figura 159. Elementos Paso 7 (Cuerpo)	109
Figura 160. Ensamblaje Paso 7 (Cuerpo).....	109
Figura 161. Elementos Paso 8 (Cuerpo)	109
Figura 162. Ensamblaje Paso 8 (Cuerpo).....	109
Figura 163. Elementos Paso 9 (Cuerpo)	110
Figura 164. Ensamblaje Paso 9 (Cuerpo).....	110
Figura 165. Elementos Paso 10 (Cuerpo)	110
Figura 166. Ensamblaje Paso 10 (Cuerpo).....	111
Figura 167. Elementos Paso 11 (Cuerpo)	111
Figura 168. Ensamblaje Paso 11 (Cuerpo).....	111
Figura 169. Elementos Paso 12 (Cuerpo)	112
Figura 170. Ensamblaje Paso 12 (Cuerpo).....	112
Figura 171. Elementos Paso 13 (Cuerpo)	113
Figura 172. Ensamblaje Paso 13 (Cuerpo).....	113
Figura 173. Elementos Paso 14 (Cuerpo)	113
Figura 174. Ensamblaje Paso 14 (Cuerpo).....	114
Figura 175. Elementos Paso 15 (Cuerpo)	114
Figura 176. Ensamblaje Paso 15 (Cuerpo).....	114
Figura 177. Elementos Paso 16 (Cuerpo)	115
Figura 178. Ensamblaje Paso 16 (Cuerpo).....	115

Figura 179. Elementos Paso 17 (Cuerpo)	115
Figura 180. Ensamblaje Paso 17 (Cuerpo).....	116
Figura 181. Elementos Paso 18 (Cuerpo)	116
Figura 182. Ensamblaje Paso 18 (Cuerpo).....	116
Figura 183. Elementos Paso 19 (Cuerpo)	117
Figura 184. Ensamblaje Paso 19 (Cuerpo).....	117
Figura 185. Elementos Paso 20 (Cuerpo)	117
Figura 186. Ensamblaje Paso 20 (Cuerpo).....	118
Figura 187. Elementos Paso 21 (Cuerpo)	118
Figura 188. Ensamblaje Paso 21 (Cuerpo).....	118
Figura 189. Elementos Paso 22 (Cuerpo)	119
Figura 190. Ensamblaje Paso 22 (Cuerpo).....	119
Figura 191. Elementos Paso 23 (Cuerpo)	119
Figura 192. Ensamblaje Paso 23 (Cuerpo).....	120
Figura 193. Elementos Paso 24 (Cuerpo)	120
Figura 194. Ensamblaje Paso 24 (Cuerpo).....	120
Figura 195. Elementos Paso 25 (Cuerpo)	121
Figura 196. Ensamblaje Paso 25 (Cuerpo).....	121
Figura 197. Elementos Paso 26 (Cuerpo)	121
Figura 198. Ensamblaje Paso 26 (Cuerpo).....	122
Figura 199. Elementos Paso 27 (Cuerpo)	122
Figura 200. Ensamblaje Paso 27 (Cuerpo).....	122
Figura 201. Elementos Paso 28 (Cuerpo)	123
Figura 202. Ensamblaje Paso 28 (Cuerpo).....	123
Figura 203. Elementos Paso 29 (Cuerpo)	123
Figura 204. Ensamblaje Paso 29 (Cuerpo).....	124
Figura 205. Elementos Paso 30 (Cuerpo)	124
Figura 206. Ensamblaje Paso 30 (Cuerpo).....	125
Figura 207. Paso 1 Programación (Cuerpo).....	125
Figura 208. Paso 2 Programación (Cuerpo).....	126
Figura 209. Paso 3 Programación (Cuerpo).....	126
Figura 210. Paso 4 Programación (Cuerpo).....	126
Figura 211. Paso 5 Programación (Cuerpo).....	127
Figura 212. Elementos (Vista)	128
Figura 213. Ensamblaje Paso 1 (Vista).....	128
Figura 214. Ensamblaje Paso 2 (Vista).....	128

Figura 215. Ensamblaje Paso 3 (Vista).....	129
Figura 216. Ensamblaje Paso 4 (Vista).....	129
Figura 217. Ensamblaje Paso 5 (Vista).....	130
Figura 218. Elementos Paso 6 (Vista).....	130
Figura 219. Ensamblaje Paso 6 (Vista).....	131
Figura 220. Paso 1 Programación (Vista).....	131
Figura 221. Paso 2 Programación (Vista).....	132
Figura 222. Paso 3 Programación (Vista).....	132
Figura 223. Paso 4 Programación (Vista).....	132
Figura 224. Paso 5 Programación (Vista).....	133
Figura 225. Paso 6 Programación (Vista).....	133
Figura 226. Paso 7 Programación (Vista).....	133
Figura 227. Paso 8 Programación (Vista).....	134
Figura 228. Paso 9 Programación (Vista).....	134
Figura 229. Paso 10 Programación (Vista).....	134
Figura 230. Paso 11 Programación (Vista).....	135
Figura 231. Paso 12 Programación (Vista).....	135
Figura 232. Paso 13 Programación (Vista).....	135
Figura 233. Paso 14 Programación (Vista).....	136
Figura 234. Paso 15 Programación (Vista).....	136
Figura 235. Paso 16 Programación (Vista).....	136
Figura 236. Elementos Primer Brazo (Oído y Tacto).....	137
Figura 237. Ensamblaje Paso 1 (Oído y Tacto).....	138
Figura 238. Ensamblaje Paso 2 (Oído y Tacto).....	138
Figura 239. Ensamblaje Paso 3 (Oído y Tacto).....	138
Figura 240. Ensamblaje Paso 4 (Oído y Tacto).....	139
Figura 241. Ensamblaje Paso 5 (Oído y Tacto).....	139
Figura 242. Ensamblaje Paso 6 (Oído y Tacto).....	139
Figura 243. Ensamblaje Paso 7 (Oído y Tacto).....	140
Figura 244. Elementos Paso 8 (Oído y Tacto).....	140
Figura 245. Ensamblaje Paso 8 (Oído y Tacto).....	141
Figura 246. Elementos Segundo Brazo (Oído y Tacto).....	141
Figura 247. Segundo Brazo.....	142
Figura 248. Elementos Paso 10 (Oído y Tacto).....	142
Figura 249. Ensamblaje Paso 10 (Oído y Tacto).....	143
Figura 250. Paso 1 Programación (Oído y Tacto).....	143

Figura 251. Paso 2 Programación (Oído y Tacto).....	143
Figura 252. Paso 3 Programación (Oído y Tacto).....	144
Figura 253. Paso 4 Programación (Oído y Tacto).....	144
Figura 254. Paso 5 Programación (Oído y Tacto).....	144
Figura 255. Paso 6 Programación (Oído y Tacto).....	145
Figura 256. Paso 7 Programación (Oído y Tacto).....	145
Figura 257. Paso 8 Programación (Oído y Tacto).....	145
Figura 258. Paso 9 Programación (Oído y Tacto).....	146
Figura 259. Paso 10 Programación (Oído y Tacto).....	146
Figura 260. Paso 11 Programación (Oído y Tacto).....	146
Figura 261. Paso 12 Programación (Oído y Tacto).....	147
Figura 262. Paso 13 Programación (Oído y Tacto).....	147
Figura 263. Paso 14 Programación (Oído y Tacto).....	147
Figura 264. Paso 15 Programación (Oído y Tacto).....	148
Figura 265. Paso 16 Programación (Oído y Tacto).....	148
Figura 266. Paso 17 Programación (Oído y Tacto).....	148
Figura 267. Paso 18 Programación (Oído y Tacto).....	149
Figura 268. Paso 19 Programación (Oído y Tacto).....	149
Figura 269. Paso 20 Programación (Oído y Tacto).....	149
Figura 270. Paso 21 Programación (Oído y Tacto).....	150
Figura 271. Elementos Paso 1 (Luz).....	150
Figura 272. Ensamblaje Paso 1 (Luz).....	151
Figura 273. Elementos Paso 2 (Luz).....	151
Figura 274. Ensamblaje Paso 2 (Luz).....	151
Figura 275. Elementos Paso 3 (Luz).....	152
Figura 276. Ensamblaje Paso 3 (Luz).....	152
Figura 277. Paso 1 Programación (Luz).....	153
Figura 278. Paso 2 Programación (Luz).....	153
Figura 279. Paso 3 Programación (Luz).....	153
Figura 280. Paso 4 Programación (Luz).....	154
Figura 281. Paso 5 Programación (Luz).....	154
Figura 282. Paso 6 Programación (Luz).....	154
Figura 283. Paso 7 Programación (Luz).....	154
Figura 284. Paso 8 Programación (Luz).....	155
Figura 285. Paso 9 Programación (Luz).....	155
Figura 286. Paso 10 Programación (Luz).....	155

Figura 288. Paso 11 Programación (Luz)	156
Figura 289. Paso 12 Programación (Luz)	156
Figura 290. Paso 13 Programación (Luz)	156
Figura 291. Paso 14 Programación (Luz)	157
Figura 292. Paso 15 Programación (Luz)	157
Figura 293. Paso 16 Programación (Luz)	157
Figura 294. Paso 17 Programación (Luz)	158
Figura 295. Paso 18 Programación (Luz)	158
Figura 296. Paso 19 Programación (Luz)	158
Figura 297. Paso 20 Programación (Luz)	159
Figura 298. Elementos Paso 1 (Color)	159
Figura 299. Ensamblaje Paso 1 (Color)	160
Figura 300. Paso 1 Programación (Color).....	160
Figura 301. Paso 2 Programación (Color).....	160
Figura 302. Paso 3 Programación (Color).....	161
Figura 303. Paso 4 Programación (Color).....	161
Figura 304. Paso 5 Programación (Color).....	161
Figura 305. Paso 6 Programación (Color).....	162
Figura 306. Paso 7 Programación (Color).....	162
Figura 307. Paso 8 Programación (Color).....	162
Figura 308. Paso 9 Programación (Color).....	162
Figura 309. Paso 10 Programación (Color).....	163
Figura 310. Paso 11 Programación (Color).....	163
Figura 311. Paso 12 Programación (Color).....	163
Figura 312. Paso 13 Programación (Color).....	163
Figura 313. Paso 14 Programación (Color).....	164
Figura 314. Paso 15 Programación (Color).....	164
Figura 315. Paso 16 Programación (Color).....	164
Figura 316. Paso 17 Programación (Color).....	164
Figura 317. Paso 18 Programación (Color).....	165
Figura 318. Paso 19 Programación (Color).....	165
Figura 319. Paso 20 Programación (Color).....	165
Figura 320. Paso 21 Programación (Color).....	165
Figura 321. Paso 22 Programación (Color).....	166
Figura 322. Paso 23 Programación (Color).....	166
Figura 323. Paso 24 Programación (Color).....	166

Figura 324. Primera parte Programa (Color)	167
Figura 325. Paso 26 Programación (Color).....	167
Figura 326. Paso 27 Programación (Color).....	167
Figura 327. Paso 28 Programación (Color).....	168
Figura 328. Paso 29 Programación (Color).....	168
Figura 329. Paso 30 Programación (Color).....	168
Figura 330. Paso 31 Programación (Color).....	168
Figura 331. Paso 32 Programación (Color).....	169
Figura 332. Paso 33 Programación (Color).....	169
Figura 333. Paso 34 Programación (Color).....	169
Figura 334. Paso 35 Programación (Color).....	170
Figura 335. Paso 36 Programación (Color).....	170
Figura 336. Paso 37 Programación (Color).....	170
Figura 337. Paso 38 Programación (Color).....	170
Figura 338. Paso 39 Programación (Color).....	171
Figura 339. Paso 40 Programación (Color).....	171
Figura 340. Paso 41 Programación (Color).....	171
Figura 341. Paso 42 Programación (Color).....	172
Figura 342. Paso 43 Programación (Color).....	172
Figura 343. Paso 44 Programación (Color).....	172
Figura 344. Paso 45 Programación (Color).....	172
Figura 345. Paso 46 Programación (Color).....	173
Figura 346. Paso 47 Programación (Color).....	173
Figura 347. Paso 48 Programación (Color).....	173
Figura 348. Paso 49 Programación (Color).....	174
Figura 349. Paso 50 Programación (Color).....	174
Figura 350. Paso 51 Programación (Color).....	174
Figura 351. Paso 52 Programación (Color).....	174
Figura 352. Paso 53 Programación (Color).....	175
Figura 353. Paso 54 Programación (Color).....	175
Figura 354. Paso 55 Programación (Color).....	175
Figura 355. Paso 55 Programación (Color).....	175
Figura 356. Botón de Descarga.....	176

TABLAS

Tabla 1. Lectura de Colores	31
Tabla 2. Valores de rendimiento de Servomotores	33
Tabla 3. Partes Técnicas de LEGO	40
Tabla 4. Archivos de Sistema	54
Tabla 5. Características centro de datos (bloque de pantalla).....	58
Tabla 6. Características centro de datos (bloque de movimiento)	61
Tabla 7. Características centro de datos (bloque de interrupción).....	69

INTRODUCCIÓN

Hoy en día los robots son ideales para trabajos que requieren movimientos repetitivos y precisos. Una desventaja para las empresas es que los humanos necesitan descansos, salarios, comida, dormir, y un área segura para trabajar, los robots no. La fatiga y aburrimiento de los humanos afectan directamente a la producción de una compañía, los robots nunca se aburren por lo tanto su trabajo va a ser el mismo desde que abra la compañía a las 8:00 AM hasta las 6:00 PM. Por eso la tendencia es que en un futuro la mayoría del trabajo pesado sea realizado por un robot ahorrando tiempo y dinero.

En mis años de estudio dentro de la universidad note la falta de equipo actual para la enseñanza del área de la robótica haciendo difícil su aprendizaje por esa razón el desarrollo de este proyecto ayudará a tener un material de apoyo en cuanto a la teoría y un gran aporte práctico por el equipo que se utilizará durante la realización de esta tesis.

El cual es el equipo de robótica Lego Mindstorm NXT tiene mucha flexibilidad lo cual hace que este equipo no se centre en la construcción de algo fijo sino que se puede obtener infinidad de modelos de robots y a cada modelo se le puede implementar igualmente varios tipos de comportamientos.

Finalmente el lenguaje de programación que este equipo utiliza es muy dinámico y didáctico, permite configurar varias opciones dentro de cada uno de los componentes, posee herramientas de programación como ciclos, condiciones, variables, etc. El conjunto de todas estas características nos brinda un lenguaje muy fácil de entender pero a la vez con una amplia gama de opciones.

1. PLAN DE TESIS

1.1 Problema de la investigación.

1.1.1 Antecedentes.

Hoy en día se observa el gran crecimiento del campo de la robótica y la electrónica. La empresa LEGO conocida por generar herramientas que motivan el desarrollo de la creatividad y las habilidades técnicas, recientemente creó un kit con el cual se puede desarrollar casi toda clase de robots con sensores básicos de sonido, ultrasonido, tacto y de luz. Mediante éste se desarrolla el interés sobre la robótica a la vez que se adquiere conocimiento sobre esta área.

En la universidad todavía no existen muchos proyectos que incentive a la investigación y estudio del área de robótica. Para lo cual se podría adquirir un kit mediante el cual el alumno adquiriera el interés y las habilidades sobre esta área.

1.1.2 Significado del problema.

La inexistencia de un proyecto que incentive el estudio del área de la robótica en la inteligencia artificial. Creó la necesidad de que con esta investigación se busque lograr utilizar a su máximo potencial el kit LEGO Mindstorm NXT, desarrollando con este un humanoide que a través de los sensores que este kit trae, sea capaz de oír, caminar, detectar distancia y que tenga un sentido de tacto básico.

1.1.3 Definición del problema.

¿Mediante qué proyecto se podría ayudar a la carencia de herramientas dentro del área de robótica?

Con proyectos que implementen equipos para el aprendizaje de la robótica

¿Qué tipo de herramientas ayudarían mejor a incentivar el estudio por parte de los alumnos dentro del área de robótica?

Con equipos que permitan realizar varios diseños y que aumenten la creatividad del estudiante alentándolo a inventar nuevos diseños.

¿Cuál sería el impacto de uso de este kit dentro de la universidad?

Conocer el movimiento humano, el funcionamiento de motores y poder imitar de cierta forma a través de los sensores algunos de los cinco sentidos.

¿Cuál es el funcionamiento del kit de robótica LEGO Mindstorm NXT y cómo se lo puede utilizar para construir un humanoide?

Este kit está diseñado para que el que lo use pueda a través de sus elementos diseñar y armar diferentes tipos de robots los cuáles funcionarán a través de los que se le programe en el CPU (Unidad Central de Procesamiento) central que el kit trae. Utilizando todos los elementos que el kit incluido su CPU, sensores y motores se puede generar un humanoide el cual camine y maneje de manera básica los sentidos del oído, vista y tacto.

1.1.4 Planteamiento del tema.

Desarrollo y Construcción de un Humanoide utilizando las herramientas proporcionadas por el Equipo de Robótica Lego Mindstorm NXT.

1.1.5 Delimitación del tema.

El desarrollo y construcción de un humanoide utilizando el equipo de robótica Lego Mindstorm NXT va dirigido para ayudar al área de robótica en la Escuela de Ingeniería de Sistemas de la Pontificia Universidad Católica Sede Ambato.

El desarrollo y construcción de un humanoide a través del kit de robótica Lego Mindstorm NXT se llevará a cabo durante nueve meses a partir de la aprobación de este plan de tesis.

Para el desarrollo del humanoide se van a utilizar las herramientas prestadas por el equipo de robótica Lego Mindstorm NXT junto con el software que este mismo trae para la parte de programación del humanoide

Finalmente la investigación busca conocer el funcionamiento del kit de robótica LEGO Mindstorm NXT, y utilizando éste desarrollar un humanoide el cual va a ser capaz de caminar, de utilizar los sensores que trae el kit entre los que tenemos: sensor ultrasonido (para ver distancia), sensor de sonido (detecta presencia de sonido), sensor de luz (indica presencia de luz) y un sensor de tacto (pulsador) y también la adaptación extra al humanoide de un sensor de color (distingue colores).

1.2 Marco teórico.

○ Inteligencia Artificial

Según el diccionario se denomina inteligencia artificial a la ciencia que intenta la creación de programas para máquinas que imiten el comportamiento y la comprensión humana.

▪ Robótica.

La robótica es una rama de la tecnología, que estudia el diseño y construcción de máquinas capaces de desempeñar tareas repetitivas o peligrosas para el ser humano.

○ Electrónica Digital.

La electrónica digital es una parte de la electrónica que se encarga del estudio de sistemas electrónicos en los cuales la información está codificada en dos únicos estados.

○ PIC.

Los "PIC" son una familia de microcontroladores tipo RISC fabricados por Microchip Technology Inc. y derivados del PIC1650, originalmente desarrollado por la división de microelectrónica de General Instruments.

▪ Programación de PIC.

Para transferir el código de un ordenador al PIC normalmente se usa un dispositivo llamado programador.

○ Sensores.

Un elemento imprescindible para la toma de medidas es el sensor que se encarga de transformar la variación de la magnitud a medir en una señal eléctrica. Los sensores pueden ser:

- Luz.
- Pulsador.
- Sonido.
- Ultrasonido.
- Color.

- Servomotores.

El servomotor permite dar movimiento al robot, tiene sensores de rotación. Permite realizar movimiento de alta precisión. Los tres servomotores dan al robot la capacidad de moverse. De los cuales vamos a tratar lo siguiente:

- Funcionamiento.
- Uso.

- Movimiento del cuerpo humano.

Los principios que rigen el movimiento humano derivan de los principios de la locomoción generales.

- Caminar.

La biomecánica entendida como un conjunto de conocimientos obtenidos a través del estudio de los sistemas biológicos, centrado en nuestro caso en el cuerpo humano, como un sistema de naturaleza físico- química, está sometido a la gravedad.

- Humanoides.

El término "humanoide" se refiere a cualquier ser cuya estructura corporal se asemeja a la de un humano.

- Características.

Usualmente una especie de humanoide ficticio posee el mismo aspecto exterior que un humano.

- Funcionamiento y manejo de LEGO Mindstorm NXT.

- Elementos.

El kit consta de sensores, servomotores, piezas para el cuerpo y las articulaciones.

- Funcionamiento.

Como es característico de LEGO tiene un funcionamiento bastante flexible y se adapta fácilmente al objeto que se arme.

- Programación.

El kit trae una aplicación de alto nivel pero con el cual se puede obtener control de varios aspectos de cada elemento es de fácil manejo debido a que es gráfico.

1.3 Objetivos.

1.3.1 General.

Desarrollar y construir un humanoide utilizando las herramientas proporcionadas por el Equipo de Robótica Lego Mindstorm NXT.

1.3.2 Específicos.

- Utilizar el kit de robótica para ayudar al aprendizaje de la Robótica en la Inteligencia Artificial dentro de la Escuela de Sistemas de Pontificia Universidad Católica del Ecuador Sede Ambato.
- Dar un incentivo a los alumnos a investigar más sobre el área de robótica dejando un precedente y a la vez una ayuda, ya que con este equipo no solo se puede realizar un humanoide sino que se le deja al estudiante la posibilidad de crear diferentes tipos de robots y aprender el funcionamiento de sus dispositivos.
- Investigar sobre el manejo y funcionamiento del kit de robótica Lego Mindstorm NXT, aplicando el conocimiento adquirido en el desarrollo y construcción de un humanoide.

1.4 Metodología de trabajo.

Este proyecto es una investigación descriptiva la cual se va a realizar utilizando la información que se encuentre sobre el tema y aprendiendo a través de la práctica el manejo y funcionamiento del kit de robótica Lego Mindstorm NXT.

1.5 Justificación.

Actualmente una de las áreas que está creciendo en el mundo es el área de la robótica, por esa razón y por la falta de equipos dentro de la universidad que ayuden al estudiante a mejorar su aprendizaje dentro de ésta área, se ve la necesidad de que este proyecto ayude a mejorar la enseñanza de la inteligencia artificial y que incentiven al estudiante a investigar y conocer la robótica.

1.6 Bibliografía preliminar.

- <http://mindstorms.lego.com/>
- <http://www.lego.com/dacta/robolab>
- <http://pagina.de/telec>
- <http://www.anser.com.ar/robotica.htm>
- <http://www.sadio.org.ar/modules.php>

1.7 Recursos.

1.7.1 Recursos humanos.

Este proyecto lo va a realizar una persona en un promedio de 6 horas diarias durante nueve meses dando un total de 16200 a \$0,03 la hora los recursos humanos quedarían en \$486.

1.7.2 Recursos materiales.

Para la realización de este proyecto se va a usar:

- Un Kit de Robótica LEGO Mindstorm NXT valorado en \$350.
- Horas de máquina valorada alrededor de \$150.
- Hojas impresas con un costo aproximado de \$10.

1.7.3 Recursos económicos.

En la siguiente tabla se resumen los gastos totales de la realización del proyecto:

Descripción	Costo
Recursos Humanos	\$486
Recursos Materiales	\$510
Sub-Total	\$996
Imprevistos (5%)	\$49,80
Transporte (1%)	\$9,96
TOTAL	\$1055,76

2. MARCO TEÓRICO.

2.1 Inteligencia Artificial.

Según el diccionario se denomina inteligencia artificial a la ciencia que intenta la creación de programas para máquinas que imiten el comportamiento y la comprensión humana. La investigación en el campo de la IA se caracteriza por la producción de máquinas para la automatización de tareas que requieran un comportamiento inteligente.

Algunos ejemplos se encuentran en el área de control de sistemas, planificación automática, la habilidad de responder a diagnósticos y a consultas de los consumidores, reconocimiento de escritura, reconocimiento del habla y reconocimiento de patrones. De este modo, se ha convertido en una disciplina científica, enfocada en proveer soluciones a problemas de la vida diaria. Los sistemas de IA actualmente son parte de la rutina en campos como economía, medicina, ingeniería y la milicia, y se ha usado en gran variedad de aplicaciones de software, juegos de estrategia como ajedrez de computador y otros videojuegos (MINSKY, Marvin: "Robótica La última frontera de la alta tecnología", Pág. 33).

El término "inteligencia artificial" fue acuñado formalmente en 1956 durante la conferencia de Dartmouth, más para entonces ya se había estado trabajando en ello durante cinco años en los cuales se había propuesto muchas definiciones distintas que en ningún caso habían logrado ser aceptadas totalmente por la comunidad investigadora. La IA es una de las disciplinas más nuevas que junto con la genética moderna es el campo en que la mayoría de los científicos " más les gustaría trabajar".

Una de las grandes razones por la cuales se realiza el estudio de la IA es el poder aprender más acerca de nosotros mismos y a diferencia de la psicología y de la filosofía que también centran su estudio de la inteligencia, IA y sus esfuerzos por comprender este fenómeno están encaminados tanto a la construcción de entidades de inteligentes como su comprensión.

El estudio de la inteligencia es una de las disciplinas más antiguas, por más de 2000 años los filósofos no han escatimado esfuerzos por comprender como se ve, recuerda y razona junto con la forma en que estas actividades deberían realizarse. Según John

McCarthy la inteligencia es la "capacidad que tiene el ser humano de adaptarse eficazmente al cambio de circunstancias mediante el uso de información sobre esos cambios", pero esta definición resulta muy amplia ya que de acuerdo con esta, el sistema inmunológico del cuerpo humano resultaría inteligente ya que también mediante el uso de información este logra adaptarse al cambio. Otra interesante manera de ilustrar la inteligencia sería recurrir a la teoría societal de la mente de Marvin Minsky donde cada mente humana es el resultado del accionar de un comité de mentes de menor poder que conversan entre sí y combinan sus respectivas habilidades con el fin de resolver problemas.

La llegada de las computadoras a principios de los 50, permitió el abordaje sin especulación de estas facultades mentales mediante una auténtica disciplina teórica experimental. Es a partir de esto que se encontró que la IA constituye algo mucho más complejo de lo que se pudo llegar a imaginar en principio ya que las ideas modernas que constituyen esta disciplina se caracterizan por su gran riqueza, sutileza e interés; en la actualidad la IA abarca una enorme cantidad de subcampos que van desde áreas de propósito general hasta tareas específicas (RUSSELL, Stuart y otros "Inteligencia Artificial: Un enfoque moderno").

2.1.1 Robótica.

Según la enciclopedia Wikipedia la robótica es una rama de la tecnología, que estudia el diseño y construcción de máquinas capaces de desempeñar tareas repetitivas o peligrosas para el ser humano. Las ciencias y tecnologías de las que deriva podrían ser: el álgebra, los autómatas programables, las máquinas de estados, la mecánica, la electrónica y la informática.

La robótica es un concepto de dominio público. La mayor parte de la gente tiene una idea de lo que es la robótica, sabe sus aplicaciones y el potencial que tiene; sin embargo, no conocen el origen de la palabra robot, ni tienen idea del origen de las aplicaciones útiles de la robótica como ciencia.

La robótica como hoy en día la conocemos, tiene sus orígenes hace miles de años. Nos basaremos en hechos registrados a través de la historia, y comenzaremos aclarando que antiguamente los robots eran conocidos con el nombre de autómatas, y

la robótica no era reconocida como ciencia, es más, la palabra robot surgió hace mucho después del origen de los autómatas.

Desde el principio de los tiempos, el hombre ha deseado crear vida artificial. Se ha empeñado en dar vida a seres artificiales que le acompañen en su morada, seres que realicen sus tareas repetitivas, tareas pesadas o difíciles de realizar por un ser humano. De acuerdo a algunos autores, como J. J. C. Smart y Jasia Reichardt, consideran que el primer autómata en toda la historia fue Adán creado por Dios. De acuerdo a esto, Adán y Eva son los primeros autómatas inteligentes creados, y Dios fue quien los programó y les dio sus primeras instrucciones que debieran de seguir. Dentro de la mitología griega se puede encontrar varios relatos sobre la creación de vida artificial, por ejemplo, Prometeo creó el primer hombre y la primer mujer con barro y animados con el fuego de los cielos. De esta manera nos damos cuenta de que la humanidad tiene la obsesión de crear vida artificial desde el principio de los tiempos. Muchos han sido los intentos por lograrlo.

Los hombres creaban autómatas como un pasatiempo, eran creados con el fin de entretener a su dueño. Los materiales que se utilizaban se encontraban al alcance de todo el mundo, esto es, utilizaban maderas resistentes, metales como el cobre y cualquier otro material moldeable, esto es, que no necesitara o requiriera de algún tipo de transformación para poder ser utilizado en la creación de los autómatas.

Estos primeros autómatas utilizaban, principalmente, la fuerza bruta para poder realizar sus movimientos. A las primeras máquinas herramientas que ayudaron al hombre a facilitarle su trabajo no se les daba el nombre de autómata, sino más bien se les reconocía como artefactos o simples máquinas.

2.1.2 Historia de la Robótica.

La historia de la robótica ha estado unida a la construcción de "artefactos", que trataban de materializar el deseo humano de crear seres a su semejanza y que lo descargasen del trabajo *Figura 1*. El ingeniero español Leonardo Torres Quevedo (que construyó el primer mando a distancia para su torpedo automóvil mediante telegrafía sin hilos, el primer transbordador aéreo y otros muchos

ingenios) acuñó el término "automática" en relación con la teoría de la automatización de tareas tradicionalmente asociadas a los humanos.



Figura 1. Evolución de la Robótica

Karel Capek, un escritor checo, acuñó en 1921 el término "Robot" en su obra dramática "Rossum's Universal Robots / R.U.R.", a partir de la palabra checa *Robbota*, que significa servidumbre o trabajo forzado. El término robótica es acuñado por Isaac Asimov, definiendo a la ciencia que estudia a los robots. Asimov creó también las Tres Leyes de la Robótica. En la ciencia ficción el hombre ha imaginado a los robots visitando nuevos mundos, haciéndose con el poder, o simplemente aliviando de las labores caseras.

Román Gubern analiza en su libro "El simio informatizado" los motivos del ser humano para crear seres artificiales *a su imagen y semejanza*. Algunos robots están diseñados hoy en día para parecerse a los humanos.

Por siglos el ser humano ha construido máquinas que imiten las partes del cuerpo humano. Los antiguos egipcios unieron brazos mecánicos a las estatuas de sus dioses. Estos brazos fueron operados por sacerdotes, quienes clamaban que el movimiento de estos era inspiración de sus dioses. Los griegos construyeron estatuas que operaban con sistemas hidráulicos, los cuales se utilizaban para fascinar a los adoradores de los templos.



Figura 2. Brazo Robótico

Durante los siglos XVII y XVIII en Europa fueron construidos muñecos mecánicos muy ingeniosos que tenían algunas características de robots.

Jacques de Vaucansos construyó varios músicos de tamaño humano a mediados del siglo XVIII. Esencialmente se trataba de robots mecánicos diseñados para un propósito específico: la diversión.

En 1805, Henri Maillardert construyó una muñeca mecánica que era capaz de hacer dibujos. Una serie de levas se utilizaban como "el programa" para el dispositivo en el proceso de escribir y dibujar. Estas creaciones mecánicas de forma humana deben considerarse como inversiones aisladas que reflejan el genio de hombres que se anticiparon a su época. Hubo otras invenciones mecánicas durante la revolución industrial, creadas por mentes de igual genio, muchas de las cuales estaban dirigidas al sector de la producción textil. Entre ellas se puede citar la hiladora giratoria de Hargreaves (1770), la hiladora mecánica de Crompton (1779), el telar mecánico de Cartwright (1785), el telar de Jacquard (1801), y otros.

El desarrollo en la tecnología, donde se incluyen las poderosas computadoras electrónicas, los actuadores de control retroalimentados, transmisión de potencia a través de engranes, y la tecnología en sensores han contribuido a flexibilizar los mecanismos autómatas para desempeñar tareas dentro de la industria. Son varios los factores que intervienen para que se desarrollaran los primeros robots en la década de los 50's. La investigación en inteligencia artificial desarrolló maneras de emular el procesamiento de información humana con computadoras electrónicas e inventó una variedad de mecanismos para probar sus teorías (BATURONE, Aníbal Ollero; "Robótica: Manipuladores y Robots Móviles", Pág. 21).

No obstante las limitaciones de las máquinas robóticas actuales, el concepto popular de un robot es que tiene una apariencia humana y que actúa como tal. Este concepto humanoide ha sido inspirado y estimulado por varias narraciones de ciencia ficción.

Estos principios fueron denominados por Asimov las Tres Leyes de la Robótica, y son:

- Un robot no puede actuar contra un ser humano o, mediante la inacción, que un ser humano sufra daños.
- Un robot debe de obedecer las órdenes dadas por los seres humanos, salvo que estén en conflictos con la primera ley.
- Un robot debe proteger su propia existencia, a no ser que esté en conflicto con las dos primeras leyes.

Consecuentemente todos los robots de Asimov son fieles sirvientes del ser humano, de ésta forma su actitud contraviene a la de Kapek.

2.1.2.1 Actualidad de la Robótica.

La robótica ha alcanzado un nivel de madurez bastante elevado en los últimos tiempos, y cuenta con un correcto aparato teórico. Sin embargo, al intentar reproducir algunas tareas que para los humanos son muy sencillas, como andar, correr o coger un objeto sin romperlo, no se ha obtenido resultados satisfactorios, especialmente en el campo de la robótica autónoma. A pesar de ello se espera que el continuo aumento de la potencia de los ordenadores y las investigaciones en inteligencia artificial, visión artificial, la robótica autónoma y otras ciencias paralelas permitan acercarse un poco más cada vez a los milagros soñados por los primeros ingenieros y también a los peligros que adelanta la ciencia ficción.

2.2 Electrónica Digital.

Según la enciclopedia Encarta la electrónica digital es una parte de la electrónica que se encarga del estudio de sistemas electrónicos en los cuales la información está codificada en dos únicos estados. A dichos estados se les puede llamar "verdadero" o "falso", o más comúnmente 1 y 0. Electrónicamente se les asigna a cada uno un

voltaje o rango de voltaje determinado, a los que se les denomina niveles lógicos, típicos en toda señal digital.

Se diferencia de la electrónica analógica en que, para la electrónica digital un valor de voltaje codifica uno de estos dos estados, mientras que para la electrónica analógica hay una infinidad de estados de información que codificar según el valor del voltaje.

Esta particularidad permite que, usando Álgebra Booleana y un sistema de numeración binario, se puedan realizar complejas operaciones lógicas o aritméticas sobre las señales de entrada, muy costosas de hacer empleando métodos analógicos.

La electrónica digital ha alcanzado una gran importancia debido a que es utilizada para realizar autómatas y por ser la piedra angular de los sistemas microprogramados como son los ordenadores o computadoras.

2.2.1 Herramientas.

Los sistemas digitales tienen las siguientes herramientas:

- Sistemas cableados
 - Eventos: Se denomina todo sistema digital en el que sus salidas son función exclusiva del valor de sus entradas en un momento dado (al ejecutarse un evento o acción determinada), sin que intervengan en ningún caso estados anteriores de las entradas o de las salidas.
 - Secuenciales: A diferencia de los sistemas basados en eventos, en los secuenciales, los valores de las salidas, en un momento dado, no dependen exclusivamente de los valores de las entradas en dicho momento, sino también de los valores anteriores.
 - Memorias: Se trata de una memoria de semiconductor en la que se puede tanto leer como escribir información.
 - Convertidores: Dispositivo que sirve para pasar una señal análoga a digital y viceversa.

- **Sistemas programados**
 - **Microprocesadores:** El microprocesador, micro o "unidad central de procesamiento", CPU, es un circuito integrado que sirve como cerebro de la computadora. En el interior de este componente electrónico existen millones de transistores integrados.

 - **Microcontroladores:** Un microcontrolador es un circuito integrado o chip que incluye en su interior las tres unidades funcionales de una computadora: CPU, Memoria y Unidades de E/S, es decir, se trata de un computador completo en un solo circuito integrado.

- **Tablas de la verdad:**

Representan los estados lógicos que toman las señales de entrada y salida. Estas se hacen a partir de las condiciones a cumplir o resolver. Para hacerla, se escriben en la primera fila, todas las variables de entrada y salida y en las siguientes todas las combinaciones de ceros y unos. Para saber cuántas filas deben ser, se halla 2 elevado a la n, siendo n el número de variables que intervienen. Las variables son las cosas que cambian en un circuito, por ejemplo en un circuito eléctrico, los interruptores serán las variables. Para escribir los ceros y unos, se siguen las alternancias 1, 2, 4, 8.

- **Funciones lógicas:**

Representan lo mismo que antes, pero de forma matemática. Generalmente se implementan las funciones lógicas como una suma de productos, partiendo de las salidas "1" que se obtuvieron en las tablas de la verdad, si una de las variables es "0" esta será negada y se representa con una especie de guión encima de la letra que la identifica, si la variable es "1", se representa "normal" sin negar, estas variables se multiplican y luego se suman los productos. Existen varias reglas para simplificar estas funciones, pero se dejan para el estudio en niveles posteriores.

- **Puertas Lógicas:**

Con ellas se hace realidad lo que se representaba con las tablas y funciones. Son circuitos integrados digitales que están constituidas principalmente por transistores que no vemos al estar integrados en una cápsula. Para elegir y conectar integrados de

puertas lógicas, se debe conocer, la denominación comercial, el patillaje y la tensión de alimentación. Las puertas lógicas básicas son:

OR (O), que es una función suma, equivalente a interruptores en paralelo, por lo que la salida será 1 cuando al menos una entrada sea 1. Por ejemplo el chip 7432 el cual es del tipo TTL que trabaja a 5 voltios, tiene 4 puertas OR, de doble entrada.

AND (Y), que es una función producto, equivalente a interruptores en serie, por lo que la salida será 1 cuando ambas entradas sean 1. Por ejemplo el chip 7408, tiene 4 puertas AND, de doble entrada.

NOT (Inversora), que invierte o niega una señal. Por ejemplo el chip 7404, tiene 6 puertas NOT.

2.3 PIC (Programmable Integrated Circuit).

Los 'PIC' son una familia de microcontroladores tipo RISC fabricados por Microchip Technology Inc. y derivados del PIC1650, originalmente desarrollado por la división de microelectrónica de General Instruments. El nombre actual no es un acrónimo. En realidad, el nombre completo es PICmicro, aunque generalmente se utiliza como Programmable Integrated Circuit (Circuito Integrado Programable).

Este fue desarrollado en 1980. Contiene circuitos que funcionan de acuerdo a la lógica del programa que el usuario creó. El PIC usa un juego de instrucciones tipo RISC, cuyo número puede variar desde 35 para PIC's de gama baja a 70 para los de gama alta. Las instrucciones se clasifican entre las que realizan operaciones entre el acumulador y una constante, entre el acumulador y una posición de memoria, instrucciones de condicionamiento y de salto/retorno, implementación de interrupciones y una para pasar a modo de bajo consumo llamada sleep.

Microchip proporciona un entorno de desarrollo freeware llamado MPLAB que incluye un simulador software y un ensamblador. Otras empresas desarrollan compiladores C y BASIC. Microchip también vende compiladores para los PIC's de gama alta ("C18" para la serie F18 y "C30" para los dsPIC's) y se puede descargar una edición para estudiantes del C18 que inhabilita algunas opciones después de un tiempo de evaluación.

Para Pascal existe un compilador de código abierto, JAL, lo mismo que PicForth para el lenguaje Forth. GPUTILS es una colección de herramientas distribuidas bajo licencia GNU que incluye ensamblador y enlazador, y funciona en Linux, MacOS y Microsoft Windows. GPSIM es otra herramienta libre que permite simular diversos dispositivos hardware conectados al PIC.

2.3.1 Estructura.

Los PIC's actuales vienen con una amplia gama de mejoras hardware incorporadas:

Figura 3



Figura 3. Estructura de un PIC

- Núcleos de UCP de 8/16 bits con Arquitectura Harvard modificada
- Memoria Flash y ROM disponible desde 256 bytes a 256 kilobytes
- Puertos de E/S (típicamente 0 a 5,5 voltios)
- Temporizadores de 8/16 bits
- Tecnología Nanowatt para modos de control de energía
- Periféricos serie síncronos y asíncronos: USART, AUSART, EUSART
- Conversores analógico/digital de 10-12 bits
- Comparadores de tensión
- Módulos de captura y comparación PWM
- Controladores LCD
- Periférico MSSP para comunicaciones I²C, SPI, y I²S
- Memoria EEPROM interna con duración de hasta un millón de ciclos de lectura/escritura
- Periféricos de control de motores
- Soporte de interfaz USB
- Soporte de controlador Ethernet

- Soporte de controlador CAN
- Soporte de controlador LIN
- Soporte de controlador Irda

2.3.1.1 Tipos.

PIC's modernos

Los viejos PIC's con memoria PROM (Programmable Read-Only Memory) o EPROM (Erasable Programmable Read-Only Memory) se están renovando gradualmente por chips con memoria Flash. Así mismo, el juego de instrucciones original de 12 bits del PIC1650 y sus descendientes directos ha sido suplantado por juegos de instrucciones de 14 y 16 bits. Microchip todavía vende versiones PROM y EPROM de la mayoría de los PIC's para soporte de aplicaciones antiguas o grandes pedidos.

Se pueden considerar tres grandes gamas de PIC en la actualidad: Los básicos (Linebase), los de medio rango (Mid Range) y los de alta performance (high performance). Los PIC18 son considerados de alta performance y tienen entre sus miembros a PIC's con módulos de comunicación y protocolos avanzados (USB, Ethernet, Zigbee por ejemplo).

Clones del PIC

Por todos lados surgen compañías que ofrecen versiones del PIC más baratas o mejoradas. La mayoría suelen desaparecer rápidamente. Una de ellas que va perdurando es Ubicorn (antiguamente Scenix) que vende clones del PIC que funcionan mucho más rápido que el original.

PIC's wireless

El microcontrolador rfPIC integra todas las prestaciones del PICmicro de Microchip con la capacidad de comunicación wireless. Estos dispositivos ofrecen un diseño muy comprimido para ajustarse a los cada vez más demandado requerimientos de miniaturización en aparatos electrónicos.

PIC's para procesado de señal (dsPIC's)

Los dsPIC's son el último lanzamiento de Microchip, comenzando a producirlos a gran escala a finales de 2004. Son los primeros PIC's con bus de datos inherente de

16 bits. Incorporan todas las posibilidades de los anteriores PIC's y añaden varias operaciones en hardware. Los PIC más usados son:

- PIC12C508/509 (encapsulamiento reducido de 8 pines, oscilador interno, popular en pequeños diseños como el iPod remote)
- PIC16F84 (Considerado obsoleto, pero imposible de descartar y muy popular)
- PIC16F84A (Buena actualización del anterior, algunas versiones funcionan a 20 MHz, compatible 1:1)
- PIC12F629/675
- PIC16F628
- La familia PIC16F87X (los hermanos mayores del PIC16F84, con cantidad de mejoras incluidas en hardware. Bastante común en proyectos de aficionados)
- PIC18F452

2.3.2 Programación de PIC.

Para transferir el código de un ordenador al PIC normalmente se usa un dispositivo llamado programador. La mayoría de PIC's que Microchip distribuye hoy en día incorporan ICSP (In Circuit Serial Programming, programación serie incorporada) o LVP (Low Voltage Programming, programación a bajo voltaje), lo que permite programar el PIC directamente en el circuito. Existen muchos programadores de PIC's, desde los más simples que dejan al software los detalles de comunicaciones, a los más complejos, que pueden verificar el dispositivo a diversas tensiones de alimentación e implementan en hardware casi todas las funcionalidades. Muchos de estos programadores complejos incluyen ellos mismos PIC's preprogramados como interfaz para enviar las órdenes al PIC que se desea programar. El software de programación puede ser el ICprog, muy común entre la gente que utiliza este tipo de microcontroladores. También actualmente existen programas en los cuales se trabaja a través de diagramas de flujos e interfaces gráficas con lo cual se logra una programación más sencilla y de alto nivel para el usuario, ejemplos de este tipo de software están el que ofrece PICAXE (*Figura 4*) y el que fue desarrollado por Cocodrile (*Figura 5*).

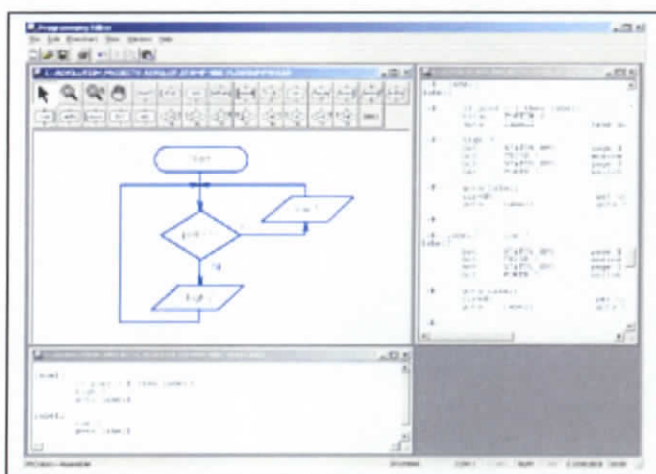


Figura 4. Entorno de Picaxe

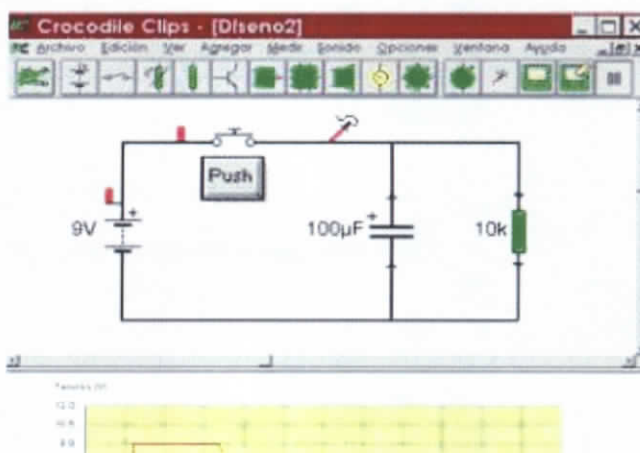


Figura 5. Entorno de Cocodrile

2.4 Sensores.

Un elemento imprescindible para la toma de medidas es el sensor que se encarga de transformar la variación de la magnitud a medir en una señal eléctrica. Los sensores se pueden dividir en:

- **Pasivos:** los que necesitan un aporte de energía externa.
 - **Resistivos:** son los que transforman la variación de la magnitud a medir en una variación de su resistencia eléctrica. Un ejemplo puede ser un termistor, que sirve para medir temperaturas.
 - **Capacitivos:** son los que transforman la variación de la magnitud a medir en una variación de la capacidad de un condensador. Un ejemplo es un condensador con un

material en el dieléctrico que cambie su conductividad ante la presencia de ciertas sustancias.

- **Inductivos:** son los que transforman la variación de la magnitud a medir en una variación de la inductancia de una bobina. Un ejemplo puede ser una bobina con el núcleo móvil, que puede servir para medir desplazamientos.
- **Activos:** los que son capaces de generar su propia energía. A veces también se les llama *sensores generadores*. Un ejemplo puede ser un transistor en el que la puerta se sustituye por una membrana permeable sólo a algunas sustancias (IsFET), que puede servir para medir concentraciones (TORRES, Fernando; "Robots y Sistemas Sensoriales", Pág. 105).

2.4.1 Sensor de Luz.

El sensor de luz (*Figura 6*) es uno de los dos sensores que permite la visión del robot. El sensor permite que tu robot distinga entre luz y oscuridad. Este puede detectar la intensidad de luz en un cuarto y medir la intensidad de luz de las superficies de colores.



Figura 6. Sensor de Luz del kit Lego Mindstorm NXT

Esto es lo que nuestros ojos ven:



Esto es lo que tu robot vería utilizando el sensor de luz:



Se puede usar el sensor de luz para realizar un robot alarma que cuando un intruso encienda la luz en un cuarto el robot pueda reaccionar y defender su propiedad. Se puede también usar para hacer un robot que siga líneas o robot que pueda ordenar cosas por color. En la *Figura 7* se muestra la estructura del Sensor de Luz.

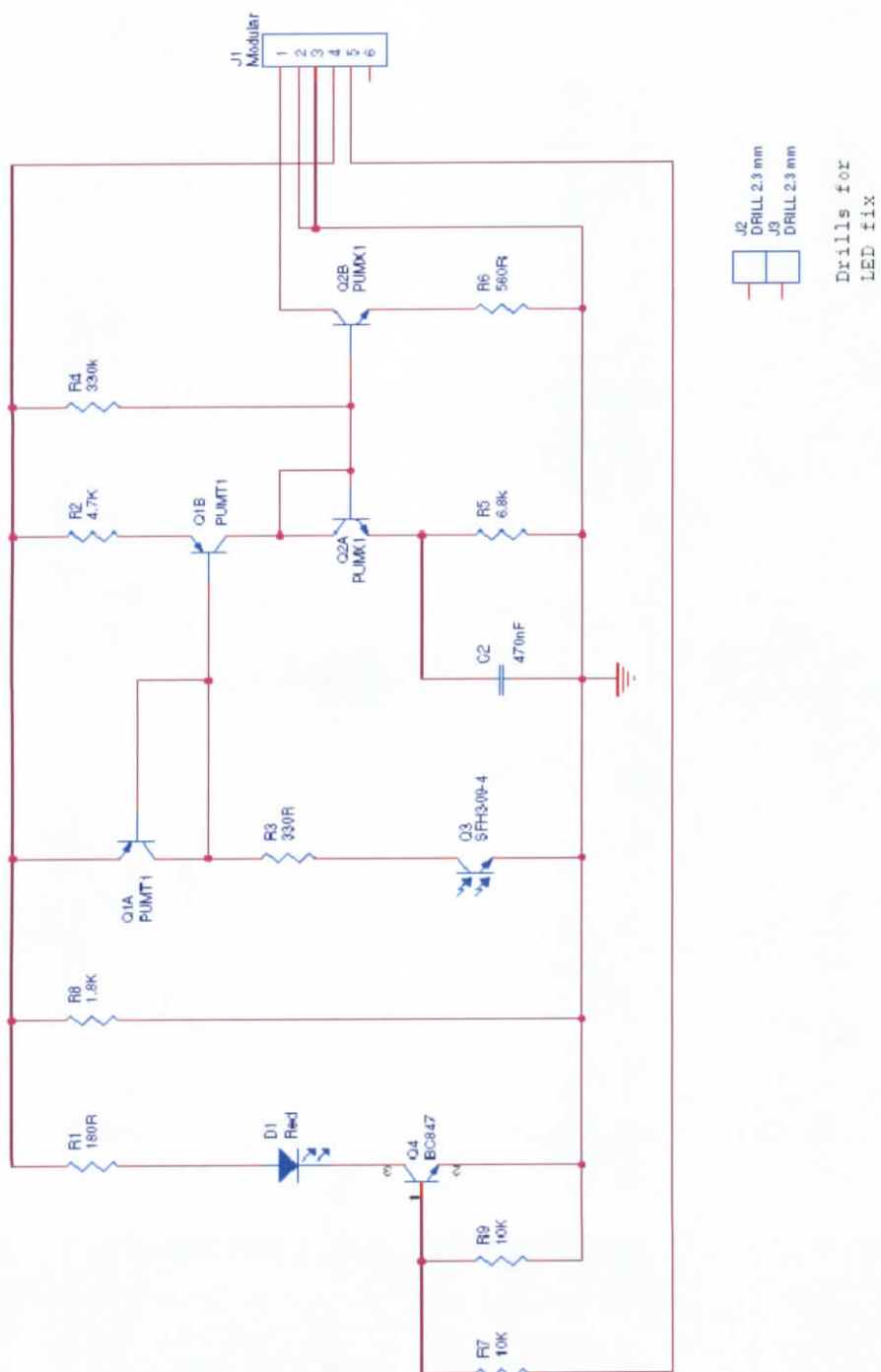


Figura 7. Estructura del Sensor de Luz (<http://www.lego.com/dacta/robolab>)

2.4.2 Sensor de Tacto o Pulsador.

El pulsador (*Figura 8*) le da al robot el sentido de tacto. El pulsador detecta cuando fue presionado por algo y cuando dejó de ser presionado (*Figura 9*).



Figura 8. Sensor de Tacto del kit Lego Mindstorm NXT

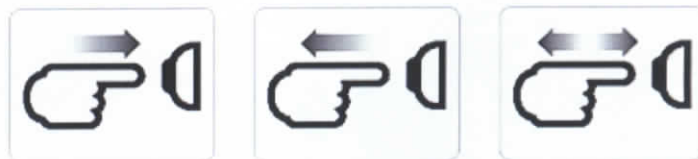


Figura 9. Estados del Sensor de Tacto

Se puede utilizar el pulsador para realizar un robot levanta cosas que consista en un brazo robótico equipado con el pulsador permitiendo que el robot detecte cuando haya algo que el brazo pueda sujetar. También se puede realizar un robot que reaccione a una orden por ejemplo que al presionar el pulsador el robot camine, hable cierre la puerta o encienda el televisor. En la *Figura 10* se muestra la estructura del Sensor de Tacto.

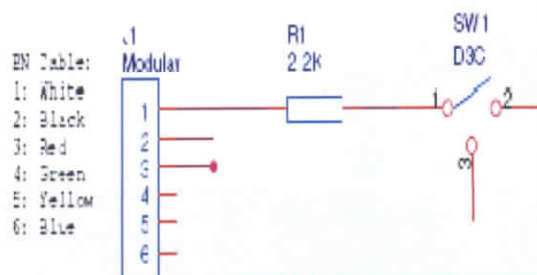


Figura 10. Estructura del Sensor de Tacto (<http://www.lego.com/dacta/robolab>)

2.4.3 Sensor de Sonido.

El sensor de sonido (*Figura 11*) puede detectar tanto decibeles dB y decibeles ajustados dBA. Un decibel es una medida de la presión del sonido.



Figura 11. Sensor de Sonido del kit Lego Mindstorm NXT

dBA: al detectar dBA la sensibilidad del sensor es adaptada a la sensibilidad del oído humano. En otras palabras, hay sonidos que sólo nuestros oídos son capaces de oír.

dB: al detectar dB, todos los sonidos son medidos con la misma sensibilidad. Ya que hay sonidos que son muy altos o muy bajos para que el ser humano los pueda escuchar.

El sensor de sonido puede medir hasta 90 dB. El sensor muestra los resultados en porcentaje para sonidos con niveles extremadamente complicados por ejemplo:

- 4.5% es para un cuarto silencioso.
- 5-10% podría ser alguien hablando a una distancia lejana.
- 10-30% es para una conversación normal cerca del sensor o música con un volumen normal.
- 30-100% son personas gritando o música a alto volumen.

En la *Figura 12* se muestra la estructura del Sensor de Sonido.

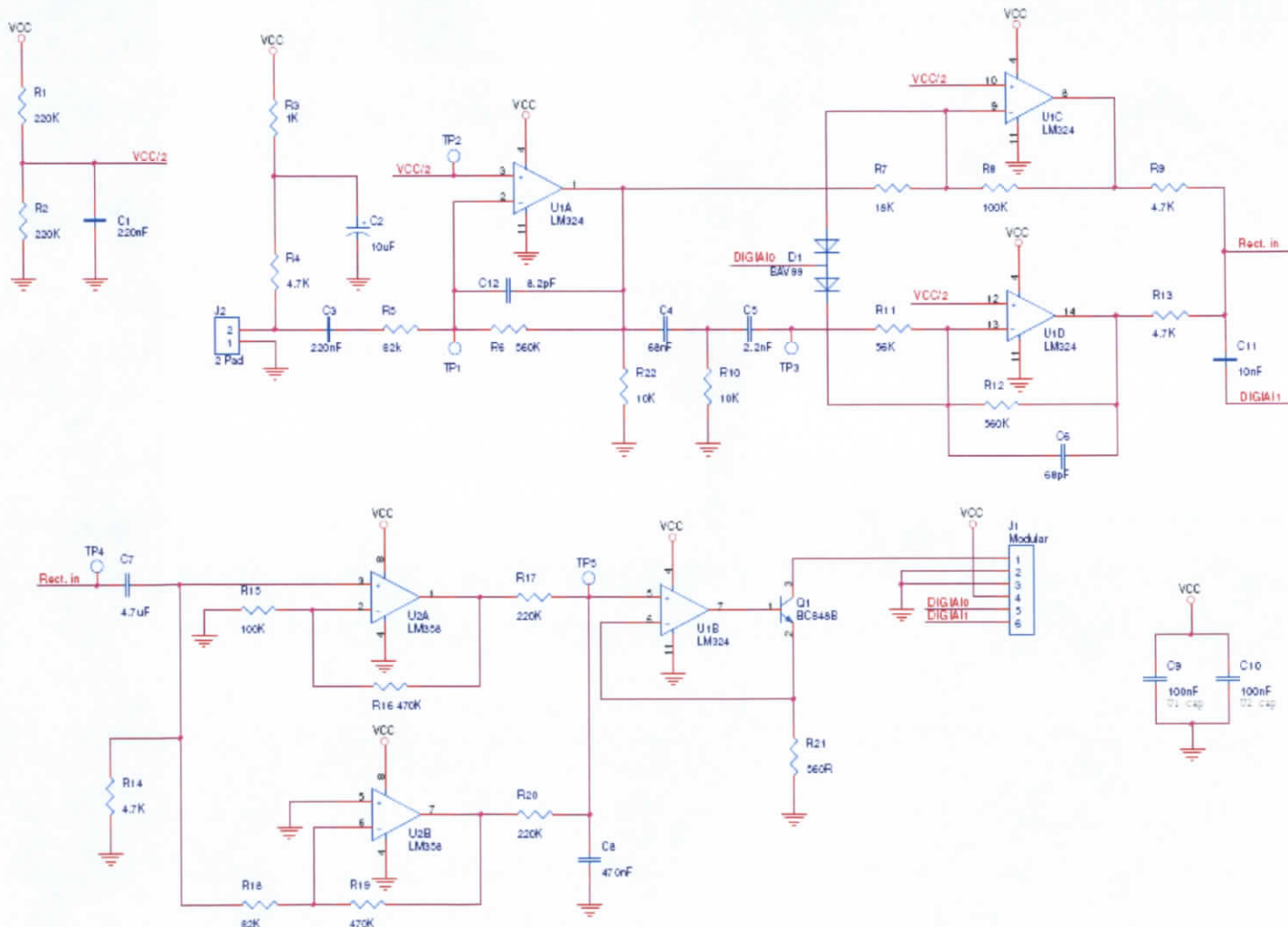


Figura 12. Estructura del Sensor de Sonido (<http://www.lego.com/dacta/robolab>)

2.4.4 Sensor de Ultrasonido.

El sensor de ultrasonido (*Figura 13*) es uno de los sensores que permite al robot tener visión. El sensor de ultrasonido permite al robot ver y detectar objetos. Se lo puede usar para realizar un robot evite obstáculos, sienta y mida distancia, y detecte movimiento.



Figura 13. Sensor de Ultrasonido del kit Lego Mindstorm NXT

El sensor de ultrasonido de Lego Mindstorm mide la distancia en centímetros y en pulgadas. Es capaz de medir distancias desde 0 a 255 centímetros con una precisión de +/- 3cm.

El sensor de ultrasonido usa el mismo principio científico que los murciélagos, estos miden la distancia calculando el tiempo que le toma a la onda de sonido en golpear un objeto y regresar.

Objetos de gran tamaño con superficies duras retornan las mejores lecturas. Objetos hechos de telas suaves, curvos o muy pequeños son difíciles de detectar por el sensor. También si hay 2 o más sensores en el mismo cuarto pueden causar interferencias en las lecturas. En la *Figura 14* se muestra la estructura del Sensor de Ultrasonido.

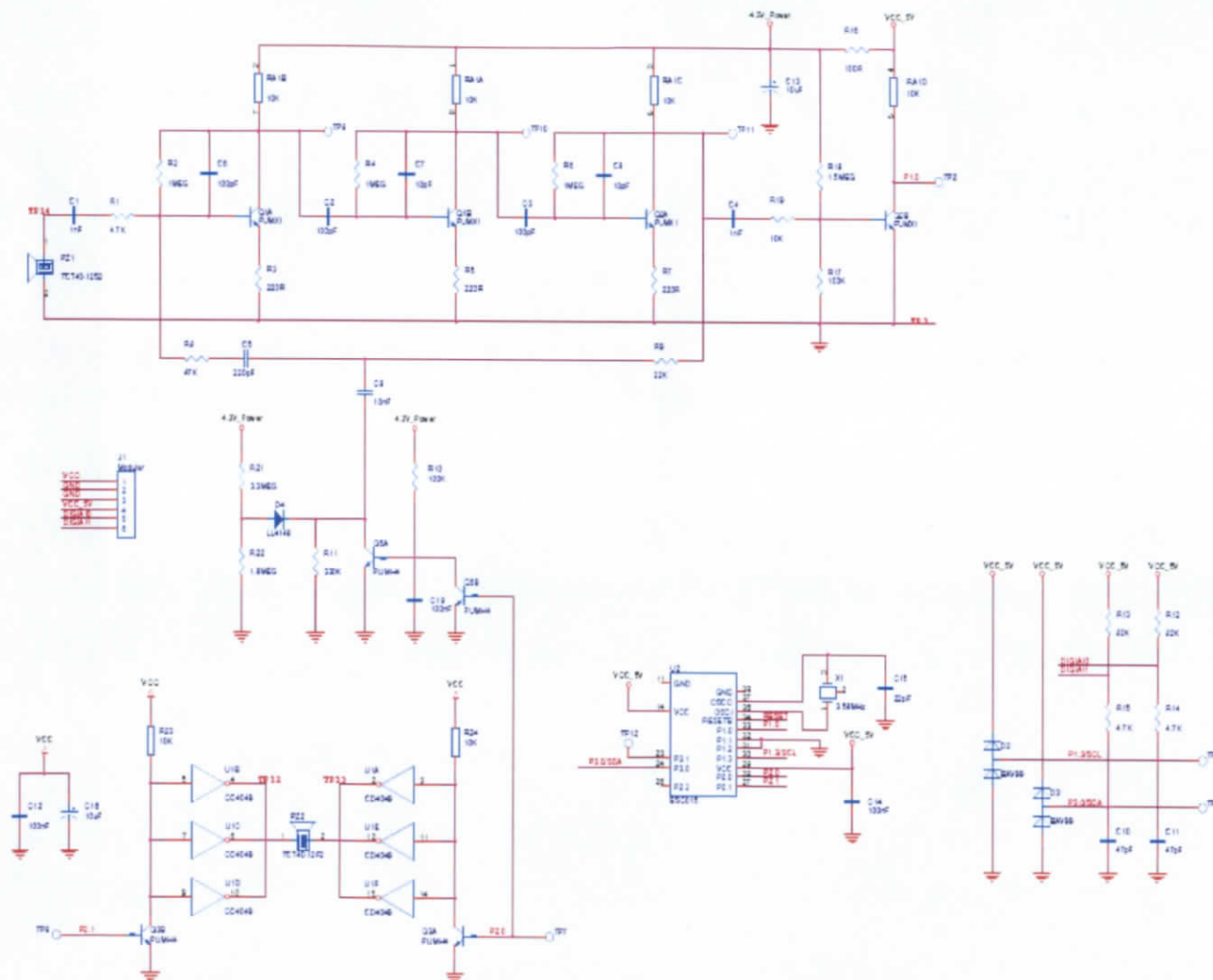


Figura 14. Estructura del Sensor de Ultrasonido (<http://www.lego.com/dacta/robofab>)

2.4.5 Sensor de Color.

El sensor de color (*Figura 15*) está diseñado para detectar con precisión el color de un objeto. El sensor de color es perfecto para construir un organizador de ladrillos o cualquier otro diseño que requiera detección de color.

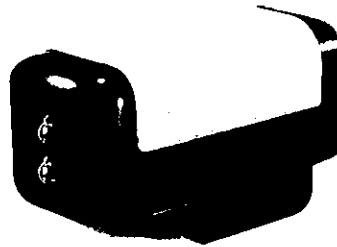


Figura 15. Sensor de Color del kit Lego Mindstorm NXT

El sensor de color funciona iluminando la superficie del objetivo con tres fuentes de luz coloreadas (Diodos emisores de luz o LED's): uno rojo, otro verde y el tercero azul. La diferencia entre la luz ambiente y el aumento debido a las fuentes de luz del sensor es utilizado para medir la luz de cada color absorbida por la superficie del objeto. Los tres valores de color se procesarán para corregir la dispersión en el espectro de emisión de cada uno de los LED.

El sensor recibe tres valores: el nivel de ROJO, el nivel de VERDE y el nivel de AZUL. El valor correspondiente a cada color está comprendido entre 0 y 255. por ejemplo, si el valor devuelto es Rojo =255, Verde =255 y Azul =255 el color leído es el Blanco.

En la *Tabla 1* se muestra las lecturas que devolverán algunos colores.

Color	Rojo	Verde	Azul
Negro	0	0	0
Blanco	255	255	255
Rojo	255	0	0
Verde	0	255	0
Azul	0	0	255
Amarillo	255	255	0

Tabla 1. Lectura de Colores (<http://www.anser.com.ar/robotica.htm>)

El sensor actualiza las lecturas a razón de 100 muestras por segundo. Las características del modo de control del sensor de color permiten dos tipos de calibración.

La calibración por nivel de negro puede utilizarse para eliminar reflejos no deseados provenientes de la estructura en la que el sensor está colocado. La calibración por balance de blancos puede utilizarse para adecuar la sensibilidad del sensor para cada uno de los tres LED's cuando ilumina una superficie blanca a una distancia determinada.

El modo de control por defecto es el 0, modo normal de medición. Si el modo de control se establece en 1, el sensor operará en modo cal. de balance de blancos y esperará ser dirigido a una superficie blanca difusa una distancia de unos 15 mm. Cuando la función cal. finaliza, los LED's destellarán y el modo de funcionamiento pasará automáticamente a 0, modo normal de medida. El proceso cal. dura alrededor de $\frac{1}{4}$ de segundo. El valor de calibración se almacena en memoria no-volátil y será recuperado cada vez que se aplica tensión al sensor. Si el modo de control se establece en 2, el sensor operará en modo cal. de nivel de negro y esperará ser situado en espacio vacío sin objetos en un cono de 90° en una distancia de al menos 0.5m. La función cal. nivel de negro mide la señal de cada color y crea un desplazamiento (offset) para anular el nivel de luz ambiente en futuras mediciones. Cuando la función cal. finaliza, los LED's destellará y el modo de funcionamiento pasará automáticamente a 0, modo normal de medida. El proceso cal. dura alrededor de 1 segundo. Esta función puede ser utilizada para contrarrestar señales devueltas por partes de la estructura circundante. El valor de calibración se almacena en memoria no-volátil y será recuperado cada vez que se aplica tensión al sensor. En la *Figura 16* se muestra la estructura del Sensor de Color.

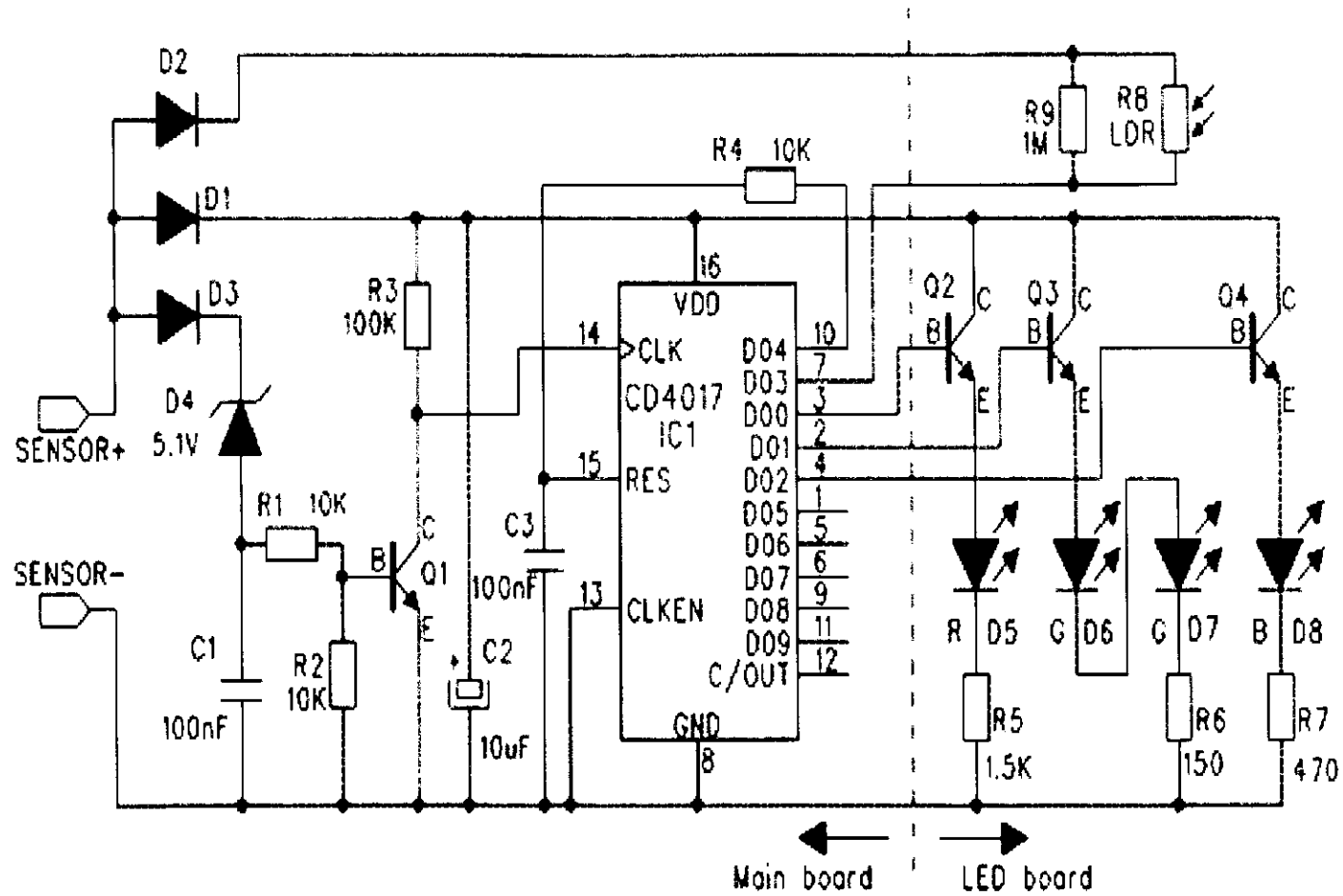


Figura 16. Estructura del Sensor de Color (<http://www.lego.com/dacta/roboLab>)

2.5 Servomotores.

El servomotor (*Figura 17*) permite dar movimiento al robot, tiene sensores de rotación. Permite realizar movimiento de alta precisión. Los tres servomotores dan al robot la capacidad de moverse.



Figura 17. Servomotor del kit Lego Mindstorm NXT

Cada motor tiene un sensor incorporado de la rotación. Esto deja los movimientos con un control exacto. El sensor de la rotación mide rotaciones del motor en grados o las rotaciones completas [exactitud de +/- un grado]. Una rotación es igual a 360 grados, así que si fijas un motor para dar vuelta a 180 grados, su eje de salida hará mitad de una vuelta. El sensor incorporado de la rotación en cada motor también deja fijar diversas velocidades para los motores.

Estos servomotores son mucho más potentes que los motores antiguos aunque también son más voluminosos y con una velocidad de rotación del eje del orden de la mitad (170 rpm frente a las 340 del antiguo). En la *Tabla 2* se muestran valores del rendimiento de servomotores.

Voltaje	Torque	Velocidad	Potencia	Poder Mecánico	Poder Eléctrico	Eficiencia
4,5v	16,7 Ncm	33rpm	0,6A	0,58W	2,7W	21,4%
7v	16,7 Ncm	82rpm	0,55A	1,44W	3,85W	37,3%
9v	16,7 Ncm	117rpm	0,55A	2,03W	4,95W	41%
12v	16,7 Ncm	177rpm	0,58A	3,10W	6,96W	44,5%

Tabla 2. Valores de rendimiento de Servomotores

(<http://www.philohome.com/nxtmotor/nxtmotor.htm>)

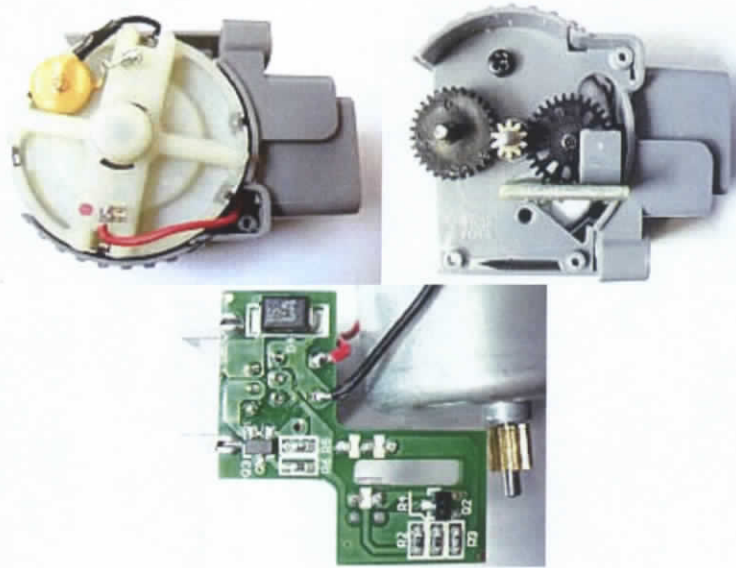


Figura 18. Estructura de un Servomotor

El servomotor tiene una estructura (*Figura 18*) que consta de principalmente una carcasa de PVC y un motor que soporta hasta máximo 15 voltios.

Los tres servomotores del juego de Lego Mindstorm tienen varios usos principalmente se los utiliza para poder controlar el movimiento del robot o también se los puede juntar con algún sensor para que en un momento dado ejecuten alguna acción.

2.6 Movimiento del cuerpo humano.

Los principios que rigen el movimiento humano derivan de los principios de la locomoción generales.

Estos principios, enunciados por el Dr. Václav Vöjta establecen que cada especie animal tiene su propia forma de locomoción, siendo la bipedestación la correspondiente al hombre.

Todas las formas de locomoción que aparecen durante el primer año de vida, no se entrenan, no se enseñan, no son producto del aprendizaje, sino que aparece por un deseo o necesidad de comunicarse con el entorno y explorarlo.

La locomoción es una forma de expresión, teniendo cada tipo de locomoción:

- Sus propios patrones de movimiento.
- Un enderezamiento del tronco contra la gravedad, y el desplazamiento del centro de gravedad. Este fenómeno se produce de manera cefalocaudal y próximodistal, es decir, de las partes superiores y centrales del cuerpo hacia las partes inferiores y las extremidades distales. La evolución es que el niño comienza a levantar la cabeza para establecer contacto con el entorno, posteriormente va levantando el tronco y los brazos, comienza a sentarse, hasta que se acaba poniendo de pie. Durante el gateo, siempre hay una mano apoyada, de manera que se establece como punto de apoyo, y la otra mano avanza en el paso. Esto sucede cuando en la marcha, siempre hay un pie apoyado y otro dando el paso. Este patrón progresa desde los miembros superiores hasta los inferiores.

Es un mecanismo innato que está impreso genéticamente en el sistema nervioso central, para ser usado desde el nacimiento.

Todos los movimientos que puede realizar el cuerpo humano han sido adquiridos y perfeccionados a lo largo de la evolución Ontogenética y filogenética.

2.6.1 Caminar.

La biomecánica entendida como un conjunto de conocimientos obtenidos a través del estudio de los sistemas biológicos, centrado en nuestro caso en el cuerpo humano, como un sistema de naturaleza físico- química, está sometido a la gravedad. Desde un punto de vista muy simplista a la biomecánica le interesa el movimiento del cuerpo humano y las cargas mecánicas y energías que se producen en ese movimiento.

El cuerpo humano ha sido construido para moverse mediante la utilización y acción de ciertas estructuras de sostén como huesos, articulaciones y músculos, y este movimiento puede tomar muy variadas y complicadas formas. Debido a esto se ha desarrollado una nueva disciplina, la biomecánica, que estudia la mecánica y los rangos del movimiento humano.

Las acciones que interesan son fundamentalmente las de caminar y levantar. Los rangos de movimiento de las articulaciones varían de persona a persona, debido a las diferencias antropométricas y al resultado de otros factores, como la edad, el sexo, la

raza, la estructura del cuerpo, el ejercicio, la ocupación, la fatiga, la enfermedad, la posición del cuerpo y la presencia o ausencia de ropa.

2.7 Humanoides.

El término "humanoide" se refiere a cualquier ser cuya estructura corporal se asemeja a la de un humano. En este sentido, el término describe homínidos no-humanos y desde luego la mayoría de los primates así como diversas criaturas mitológicas y organismos artificiales (robots), especialmente en el contexto de la ciencia ficción y la ficción fantástica. Un androide es un robot humanoide (*Figura 19*).



Figura 19. Humanoide

2.7.1 Características.

Usualmente una especie de humanoide ficticio posee el mismo aspecto exterior que un humano, siendo bípedo, pero difiere en detalles tales como los colores, forma y cualidades del rostro, presencia de cabello, estatura y peso, tipo de piel, "añadidos" como cuernos, garras, colas u otros, estructura de extremidades y linaje taxonómico (descendiente de reptiles, peces, roedores, marsupiales o vegetales en vez de primates). Los reptiles humanoides son una concepción común.

3. DESARROLLO Y CONSTRUCCIÓN DE UN HUMANOIDE

3.1 LEGO Mindstorm NXT.

Legó Mindstorm NXT según wikipedia es un juego de robótica fabricado por la empresa Legó, el cual posee elementos básicos de las teorías robóticas, como la unión de piezas y la programación de acciones, en forma interactiva. Fue puesto a la venta a partir de julio del 2006. Reemplazando al primer kit de Legó Mindstorm.

El principal componente del kit es la computadora el cerebro llamado bloque NXT (*Figura 20*). Este puede recibir datos de 4 sensores y controlar 3 motores a través de conexión por cable. El bloque tiene una pantalla de 100x64 pixels en escala de grises y 4 botones que pueden ser usados para navegar a través de la interfaz de usuario que el bloque trae. También tiene un parlante que puede reproducir archivos de sonido de hasta 16 kHz. Utiliza 6 baterías AA.



Figura 20. Bloque NXT

El bloque NXT tiene las siguientes características técnicas:

- Procesador principal ARM7 de 32-bit con 256 KB de memoria flash y 64 KB de RAM.
- Microcontrolador Atmel AVR de 8 bit y 4 MHz con 4 KB de memoria flash y 512 Bytes de RAM
- LCD de 100×64 pixel.
- Puede ser programado usando Windows o Mac OS.
- Un puerto USB 2.0

- Bluetooth (Clase II), para transferir programas al NXT de manera inalámbrica o manejo de manera remota a través de algún dispositivo móvil.
- 4 puertos de entrada.
- 3 puertos de salida.

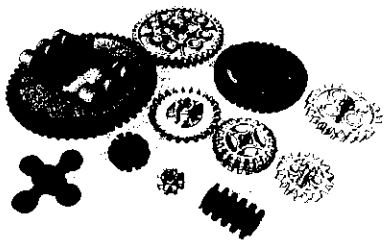

El firmware con el que trabaja el bloque NXT es de código abierto por lo cual LEGO pone a disposición mucha información sobre este, podemos encontrar la siguiente documentación:





- Kit de Desarrollo para Software (SDK), incluye información de los controladores de USB, formato del archive ejecutable y código de referencia.
- Kit de Desarrollo para Hardware (HDK), incluye documentación y esquemas para el bloque NXT y los sensores.
- Kit de Desarrollo para Bluetooth (BDK), documentos de los protocolos usados para la comunicación por Bluetooth.

3.1.1 Elementos.

El kit de robótica Lego Mindstorm NXT incluye:

- 519 partes técnicas de LEGO. En la *Tabla 3* se muestra la clasificación de las partes técnicas:

Imagen	Nombre
	Engranajes
	Vigas Anguladas

	Bloques Conectores
	Ejes
	Ruedas
	Conectores



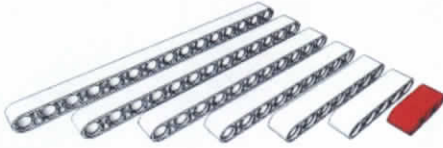

	Anillos
	Vigas Reductoras
	Vigas
	Ejes Extensores

Tabla 3. Partes Técnicas de LEGO

(<http://www.edured.gob.sv/Comunidad/blogs/carmelita/archive/2007/03/12/936.asp>)

- Componentes NXT (Figura 21). Los cuales estan compuestos por:
 - Bloque NXT (Figura 20).

- Tres servomotores (*Figura 17*).
- Sensor ultrasonido, de distancia y movimiento (*Figura 13*).
- Sensor de sonido (*Figura 11*).
- Sensor de luz (*Figura 6*).
- Sensor de tacto (*Figura 8*).

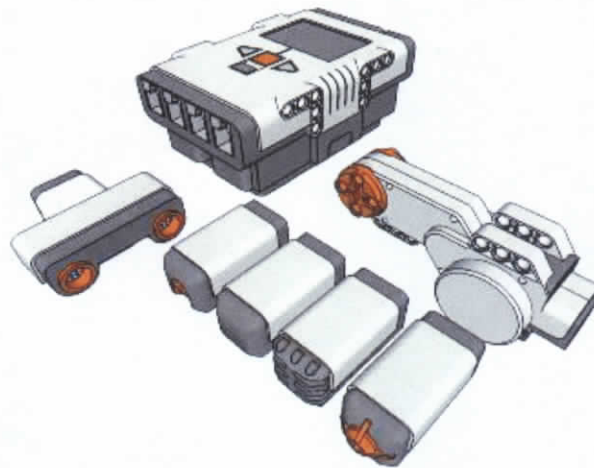


Figura 21. Componentes NXT

3.1.2 Programación.

Se pueden realizar muy simples programas directamente desde el bloque NXT. Programas más complicados y archivos de sonido pueden ser descargados usando el Puerto USB o inalámbricamente a través del Bluetooth. También se pueden copiar archivos entre 2 bloques NXT inalámbricamente, y algunos teléfonos móviles pueden ser usados como control remoto. Hasta 3 bloques NXT pueden comunicarse simultáneamente vía Bluetooth cuando un programa creado por el usuario está en ejecución.

El kit trae una aplicación predefinida (*Figura 22*) con la cual seremos capaces de programar nuestro robot y subir los programas al NXT vía USB o por Bluetooth. Es compatible tanto con Mac como con Windows. Es una aplicación Drag and Drop, viene con instrucciones de construcción y guías de programación fáciles de entender. La aplicación básicamente utiliza como herramienta principal diferentes tipos de

bloques los cuales controlan diferentes partes del kit. (MINDSOTRM;”Ayuda y Soporte para LEGO Mindstorm NXT”, 1º Edición).

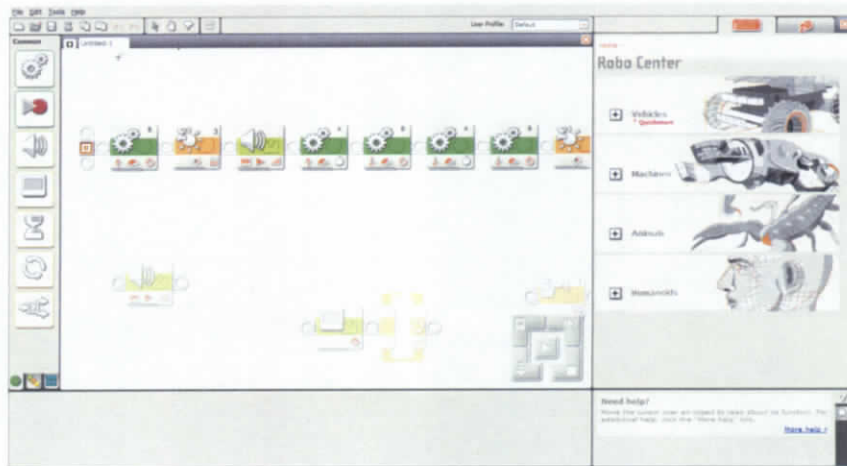


Figura 22. Aplicación incluida en el kit

3.1.2.1 Calibración de Sensores.

Las condiciones ambientales pueden afectar el rendimiento de los sensores de luz y sonido. Por eso es importante calibrar estos sensores para un ambiente en particular y así lograr un rendimiento óptimo.

Por ejemplo, en un cuarto muy brillante, el sensor de luz sin calibración sólo mostraría lecturas muy altas reduciendo así su rendimiento. Si se creó el programa en un cuarto oscuro, el robot podría reaccionar de manera inusual cuando es puesto bajo una luz intensa. Calibrando el sensor de luz se puede obtener valores adecuados bajo diferentes condiciones.

Hay dos maneras de calibrar los sensores de luz y sonido utilizando el software propio de Lego Mindstorm NXT. La primera es usando la función de calibración de sensores desde el menú Herramientas. Usando esta función, se calibra el sensor de luz o sonido solo una vez (no cada vez que el programa se ejecute).

La segunda opción para calibración incluye el uso de varios bloques de calibración en el programa. Cada vez que el programa se ejecute, estos bloques calibrarán los sensores para las condiciones actuales.

3.1.2.1.1 Función de Calibración de Sensores.

Para usar esta función, primero se debe asegurar de que el bloque NXT está conectado al computador y que esta encendido. También, asegúrese que el sensor que se desea calibrar esté conectado al bloque NXT. Se debe prestar atención al puerto al cual está conectado este.

Cuando este todo listo, se selecciona la función de calibración de sensores desde el menú herramientas. Deberá aparecer una ventana (*Figura 23*). Si el bloque NXT no está conectado la ventana aparecerá atenuada.

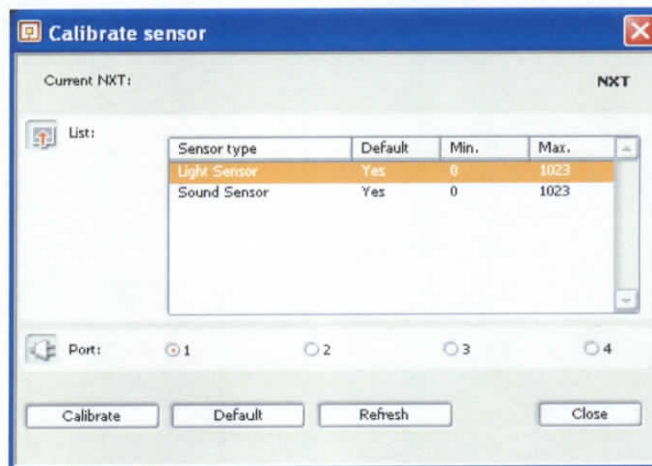


Figura 23. Calibración de Sensores

Para calibrar el sensor de luz se selecciona éste de la lista junto con el puerto al cual está conectado. Entonces se da click al botón Calibrar. Esto descargará un pequeño programa al bloque NXT y lo correrá automáticamente.

En la pantalla del bloque NXT se observará un texto que dice “Min Value:”. Entonces se debe apuntar el sensor de luz hacia un material que represente oscuridad y se presiona el botón naranja del bloque NXT.

A continuación se verá un texto que dice “Max Value:”. Entonces se debe apuntar el sensor de luz hacia un material que represente luz y se presiona el botón naranja del bloque NXT. Con eso la calibración estará completa.

Para calibrar el sensor de sonido se selecciona éste de la lista junto con el puerto al cual está conectado. Entonces se da click al botón Calibrar. Esto descargará un pequeño programa al bloque NXT y lo correrá automáticamente.

En la pantalla del bloque NXT se observará un texto que dice "Min Value:". Entonces se debe posicionar el sensor de sonido en un ambiente ruidoso y se presiona el botón naranja del bloque NXT.

A continuación se verá un texto que dice "Max Value:". Entonces se debe posicionar el sensor de sonido en un ambiente silencioso y se presiona el botón naranja del bloque NXT. Con eso la calibración estará completa.

Si se desea regresar a la calibración de fábrica se debe primero prender el bloque NXT y conectarlo al computador. Seleccionar la función de calibración de sensores desde el menú herramientas. En la ventana que aparece se selecciona el sensor al cual se quiere regresar a la calibración de fábrica y damos click en Default.

3.1.2.2 Cable de datos.

Los cables de datos llevan la información entre los bloques de programación. Varios bloques requieren de un cable de datos para funcionar. Por ejemplo, la salida de un bloque Random puede sólo ser enviada a través de un cable de datos.

Se puede crear un cable de datos dibujándolo desde el centro de datos del bloque (*Figura 24*). Casi todos los bloques de programación tienen centro de datos y pueden soportar cables de datos.

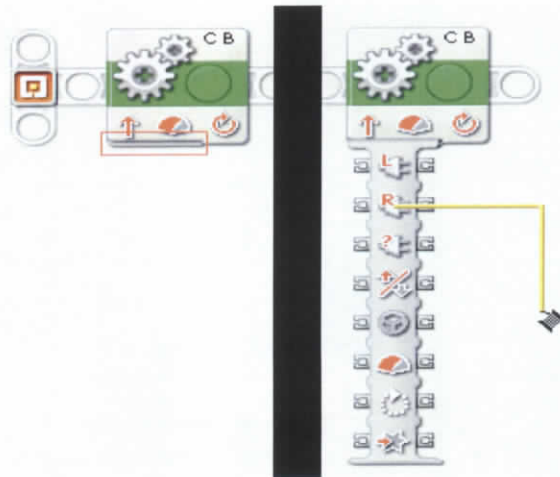


Figura 24. Creación de un cable de datos

Para abrir el centro de datos de un bloque se da click en la pestaña que se encuentra en la esquina inferior izquierda del bloque.

3.1.2.2.1 Dibujar un cable de datos.

El cursor cambiará de forma cuando esté cerca o sobre un punto de dato. Si se presiona el botón del mouse y se lo arrastra, el cable de datos crecerá y podrá ser conectado a un punto en otro centro de datos de un bloque.

3.1.2.2.2 Borrar un cable de datos.

Para borrar un cable de datos que se extiende de un punto a otro se debe dar click derecho en cualquiera de los extremos del cable de datos.

3.1.2.2.3 Entrada y salida.

Los cables de datos llevan información hacia un bloque (entrada) que tiene conectado los puntos en el lado izquierdo de su centro de datos. También los cables de datos llevan información de un bloque a otro (salida) que tienen conectados los puntos en el lado derecho (Figura 25).

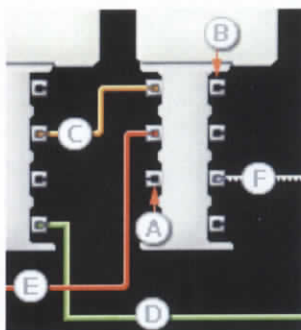


Figura 25. Entradas y salidas

[A] Punto de entrada

[B] Punto de salida

[C] Cable de datos numérico (amarillo)

[D] Cable de datos lógico (verde)

[E] Cable de datos de texto (anaranjado)

[F] Cable de datos roto (gris)

3.1.2.2.4 Tipos de datos.

Cada cable de datos lleva un tipo de dato específico entre los bloques. Por ejemplo, si un cable de datos es arrastrado desde un punto lógico en el centro de datos de un bloque, este solo puede ser conectado a un punto lógico en el centro de datos de otro bloque. Cada cable de datos tiene un color que va de acuerdo al tipo (*Figura 25*).

3.1.2.2.5 Cable de datos rotos.

Si se intenta conectar un cable de datos en un punto de un tipo de dato erróneo, el cable de datos se romperá y se pondrá de color gris. El programa no podrá ser descargado al bloque NXT si este contiene cables de datos rotos.

Si se da click en un cable roto se puede leer por qué está roto en una pequeña ventana de ayuda en la esquina inferior derecha del área de trabajo (*Figura 26*).

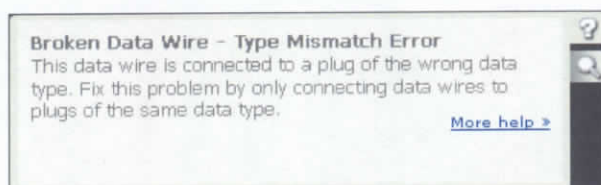


Figura 26. Venta de ayuda

La solución es remover uno o más cables del ciclo.

Ejemplo: Supongamos que se quiere crear un programa que duplica la última duración leída de un motor y que vuelva a ejecutarla.

El programa (*Figura 29*) no trabajará porque el cable de datos creó un ciclo. Se puede ver que el camino del cable pasa dos veces por los bloques.

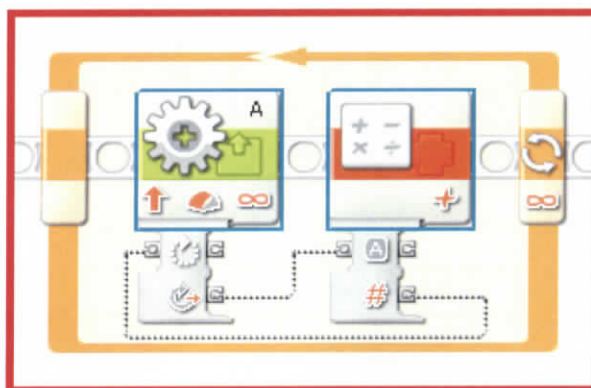


Figura 29. Error - Ciclo

El programa (*Figura 30*) resuelve el problema utilizando una variable numérica la cual es la que indique el valor duplicado y trabaja dentro de la repetición.

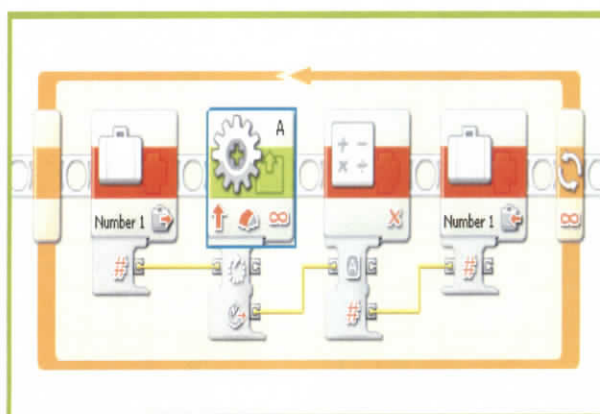


Figura 30. Corrección - Ciclo

3.1.2.3.3 Entrada perdida.

Un cable de datos sin un dato fuente no puede funcionar; Por tanto si no lleva ningún dato será marcado como roto.

Por ejemplo, en un centro de datos que tenga tanto entrada y salida para una determinada propiedad, la salida solo trabajará si otro cable de datos es también conectado a la entrada proporcionando una salida con un dato fuente.

Ejemplo: Supongamos que se quiere pasar un dato a través del centro de datos de otro bloque.

En el programa (*Figura 31*), el cable de datos roto no lleva un dato porque nada está pasando a través de éste desde la entrada. Este programa es inválido y no podrá ser descargado.

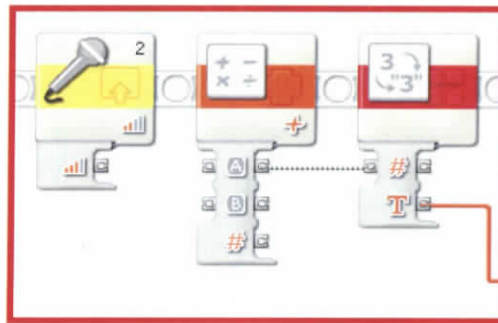


Figura 31. Error – Entrada perdida

En el programa corregido (*Figura 32*), una entrada del cable de datos fue conectada a un punto de entrada. El dato numérico del bloque del sensor de sonido puede pasar a través del bloque matemático al bloque convertidor de texto.

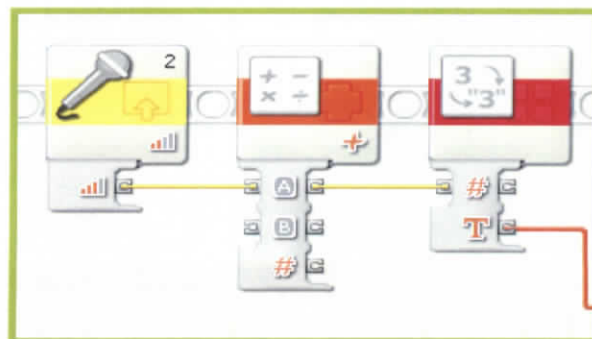


Figura 32. Corrección – Entrada perdida

3.1.2.3.4 Exceso de entradas.

Cuando más de un cable de datos es conectado a la misma entrada causará un cable de datos roto y por tanto un programa inválido (*Figura 33*).

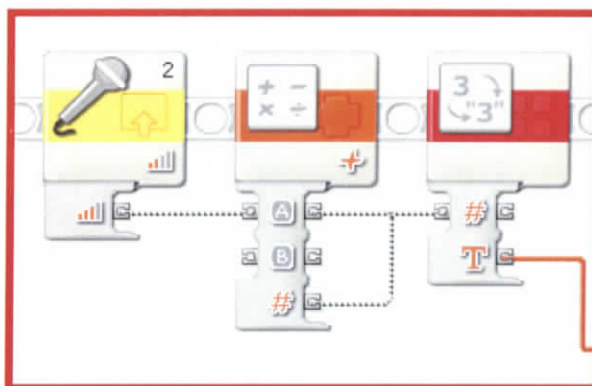


Figura 33. Error – Exceso de entradas

La solución a este problema es borrar o mover el cable de datos roto para que cada entrada esté conectada sólo a una salida (*Figura 34*).

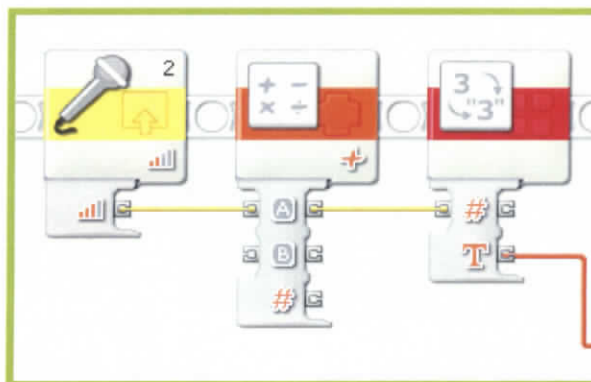


Figura 34. Corrección – Exceso de entradas

3.1.2.4 Administración de archivos y memoria del Bloque NXT.

Para acceder a los archivos y memoria del bloque NXT se utiliza el botón de la esquina superior izquierda del panel de control (*Figura 35*).



Figura 35. Acceso al bloque NXT

En la pestaña memoria de la ventana NXT nos muestra de manera gráfica la memoria usada y la memoria libre del bloque NXT. Desde esta ventana también se puede

enviar programas, sonidos, gráficos y otros archivos al NXT desde el PC, y copiar y borrar archivos que se encuentren en el bloque NXT (Figura 36).

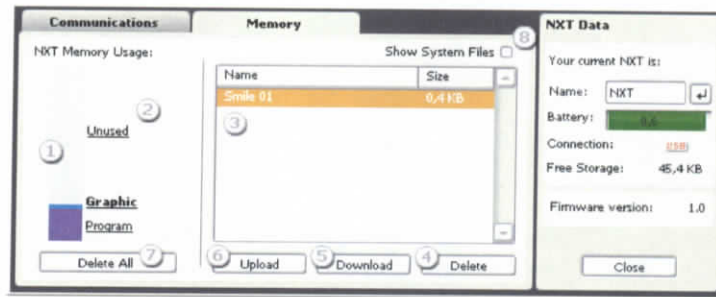


Figura 36. Ventana NXT

Cada bloque NXT trae una memoria de 118 Kb. Parte de esta memoria es usada para guardar programas ejemplos, gráficos y archivos de sonido que son pre-cargados en el bloque NXT. Esto nos deja unos 56 Kb para los archivos creados por nosotros.

En la *Tabla 4* se muestran los archivos de sistema del bloque NXT con su respectiva descripción.

Categoría	Nombre de archivo	Descripción
Otros	NVConfig.sys	Configuraciones como tiempo de espera y el volumen son guardadas en este archivo. Si se lo borra uno nuevo se crea automáticamente.
	Try-*.rtm	Cualquier archivo de sistema que comienza con "Try-" es un programa Try Me para un específico sensor o salida. Al borrar cada uno de estos archivos se removerá individualmente el programa Try Me del menú Try Me del NXT.

	RPGreader.sys	Este programa nos permite correr los programas creados desde el bloque NXT. Si se borra este archivo se perderá la habilidad de correr estos programas. Los programas descargados desde el PC si se podrán ejecutar.
	*.tmp	Cualquier archivo que termine en “.tmp” es un archivo temporal. Los archivos temporales se pueden borrar sin peligro.
	*.bak	Cualquier archivo que termine en “.bak” es un archivo temporal. Los archivos temporales se pueden borrar sin peligro.
Gráficos	faceopen	Icono usado para los programas Try Me
	faceclosed	Icono usado para los programas Try Me
Sonidos	Woops	Archivo de sonido usado en los programas Try Me. Si se borra este archivo se removerá el sonido de los programas cuando estén en uso. El programa continuará trabajando pero sin sonido.
	! Startup	Archivo de sonido que se reproduce cuando el bloque NXT arranca. Si se borra este archivo se removerá el sonido al momento del arranque.
	! Click	Archivo de sonido que se reproduce cuando es presionado un botón en el bloque NXT. Si se borra este archivo se removerá el sonido al momento de presionar un botón.
	! Attention	Archivo de sonido que es reproducido cuando ocurre un error en el bloque NXT. Si se borra este archivo se removerá el sonido cuando ocurra un error.

Programas	Demo	Programa demo para el modelo de inicio rápido. Si se borra este archivo se removerá el programa demo del bloque NXT.
-----------	------	--

Tabla 4. Archivos de Sistema (MINDSOTRM: "Ayuda y Soporte para LEGO Mindstorm NXT", 1º Edición)

Se puede restaurar la memoria del bloque NXT a las condiciones de fábrica reinstalando el firmware. Asegúrese de realizar un back-up de los archivos personales antes de realizar la restauración.

3.1.2.5 Secuencia Beam.

La secuencia beam controla (Figura 37) el flujo del programa. Esta indica la secuencia en la cual los bloques programados se ejecutarán. Los bloques conectados a la secuencia beam podrán ser descargados al bloque NXT. Otros bloques en el área de trabajo que no estén conectados a la secuencia no se podrán descargar.



Figura 37. Secuencia Beam

Se puede usar el punto de inicio para crear una secuencia beam adicional que permitirá al programa ejecutar tareas simultáneas. Por ejemplo, la secuencia beam principal controlará el movimiento del robot pero la segunda secuencia beam controlará el brazo robótico (Figura 38).

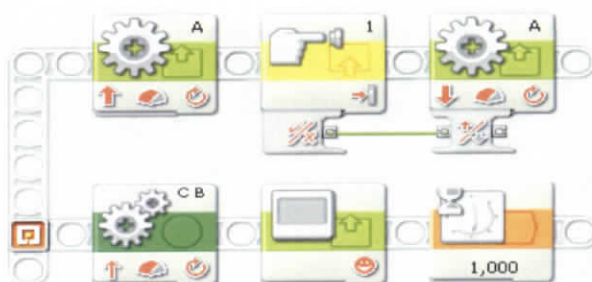


Figura 38. Varias Secuencias Beam

Se puede crear una secuencia beam paralela (*Figura 39*) arrastrando con el ratón desde el punto de inicio. A esta secuencia beam se le podrán conectar bloques. Cuando se descargue y ejecute el programa, los bloques de ambas secuencias beam se ejecutarán en paralelo. Se pueden utilizar cables de datos para comunicar los bloques de ambas secuencias.



Figura 39. Secuencia Beam Paralela

También se puede iniciar una nueva secuencia beam en medio del flujo del programa y no desde el punto de inicio (*Figura 40*). Para realizar esto se debe mantener presionado shift y arrastrar el ratón desde cualquier punto de la secuencia beam principal.



Figura 40. Secuencia Beam desde el centro del programa

3.1.2.6 Punto de Inicio.

El punto de inicio (*Figura 41*) es el inicio para todos los programas. Los bloques conectados al punto de inicio podrán ser descargados al bloque NXT.



Figura 41. Punto de Inicio

3.1.2.7 Bloque de Pantalla.

El bloque de pantalla (*Figura 42*) sirve para mostrar una imagen o un texto en la pantalla del bloque NXT. Colocando varios de estos bloques en una fila se puede crear imágenes más complejas. La pantalla tiene una resolución de 100x64 pixeles.



Figura 42. Bloque de Pantalla

CONFIGURACIÓN

Toda configuración se la realiza desde el menú en la parte inferior de la pantalla (*Figura 43*), el menú consta de las siguientes propiedades

1. Menú desplegable para escoger entre imagen, texto, un dibujo propio o reset la pantalla al ícono original.
2. Para borrar todo el contenido de la pantalla del bloque NXT marcamos la casilla "Clear".

Si se escoge mostrar una imagen se habilitarán las siguientes propiedades:

3. Lista de imágenes que se pueden mostrar, estas imágenes se encuentran en la carpeta "Image" de la aplicación, seleccione cada una para ver una pre-visualización a la derecha.
4. Usando el mouse o las casillas X Y se puede ubicar la imagen de la pre-visualización en la posición que se quiera que aparezca en la pantalla.

Si se escoge mostrar texto se habilitarán las siguientes propiedades:

5. En este campo se debe ingresar el texto que se quiere mostrar.
6. Menú desplegable en el cual se indicarán el número de líneas en las que se quiere mostrar el texto.

Si se escoge mostrar un dibujo se habilitarán las siguientes propiedades:

7. Menú desplegable que permite escoger la forma a dibujar, se puede escoger entre punto, línea o círculo.

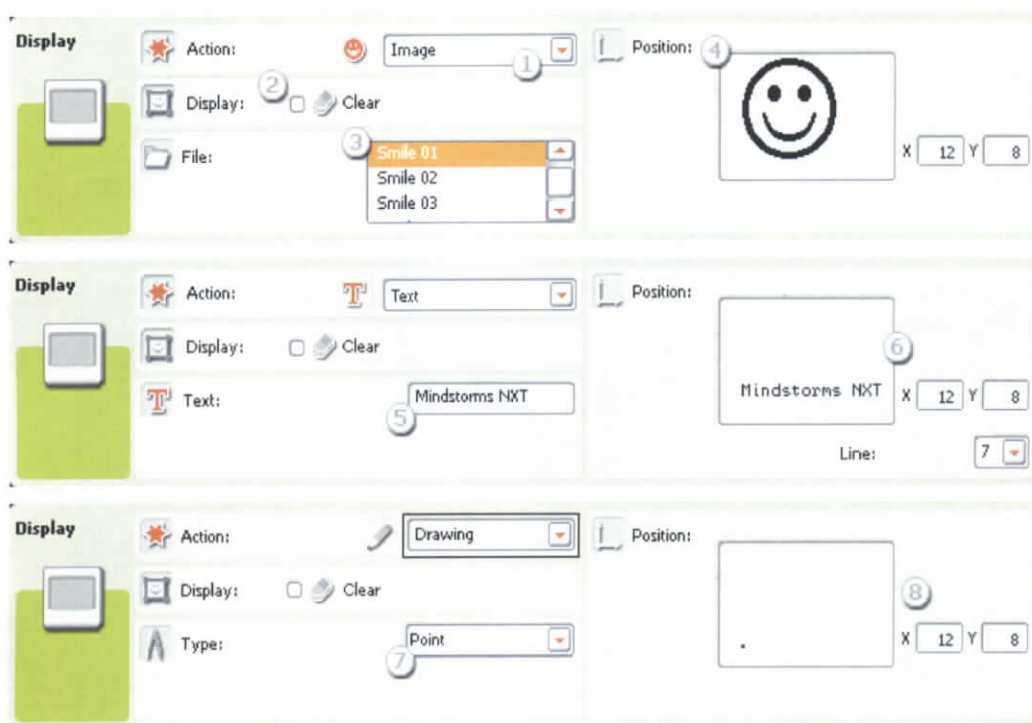


Figura 43. Menú de Configuración Bloque de Pantalla

CENTRO DE DATOS

La Tabla 5 muestra las características de los diferentes puntos del centro de datos del bloque de pantalla.

Img.	Punto	Tipo	Rango	Significado	Excep.
	Acción	Número	0 - 5	0 = Imagen, 1 = Texto, 2 = Punto, 3 = Línea, 4 = Círculo, 5 = Reset	









	Clear	Lógico	True/False	True = Limpiar pantalla False = No limpiar pantalla	
	Archivo	Texto	15 caracteres máximo	Nombre del archivo de la imagen	Archivo no es imagen
	Texto	Texto		Texto	No es texto
	X	Número	0 - 99	Coordenada X	
	Y	Número	0 - 63	Coordenada Y	
	Fin punto X	Número	0 - 99	Fin coordenada X (solo en línea)	No es línea
	Fin punto Y	Número	0 - 63	Fin coordenada Y (solo en línea)	No es línea
	Radio	Número	0 - 120	Rado (solo en círculo)	No es círculo

Tabla 5. Características centro de datos (bloque de pantalla)

(MINDSOTRM; "Ayuda y Soporte para LEGO Mindstorm NXT", 1ª Edición)

3.1.2.8 Bloque de Repetición.

El bloque de repetición (*Figura 44*) se utiliza para repetir secuencias de código. Trabaja con condiciones de repetición: tiempo, número de repeticiones, señal lógica o un sensor. También se puede realizar una repetición indefinida.



Figura 44. Bloque de Repetición

3.1.2.9 Bloque de Movimiento.

El bloque de movimiento (*Figura 45*) permite al robot ir hacia adelante o hacia atrás en línea recta o girar. Y también define el tiempo de duración del movimiento.

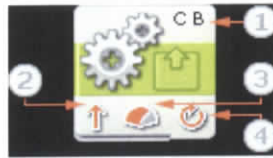


Figura 45. Bloque de Movimiento

1. La letra en la esquina superior derecha del bloque muestra cual puerto del bloque NXT está siendo controlado.
2. Este ícono muestra la dirección en la que el robot se moverá.
3. Este ícono muestra el nivel de potencia. La velocidad del robot puede ser afectada por otras condiciones, como la superficie en la que se va a mover.
4. Este ícono muestra la opción de propiedad de duración sea ilimitada, grados, rotación, parar (esta opción detiene todos los motores) o segundos.

CONFIGURACIÓN

Toda configuración se la realiza desde el menú en la parte inferior de la pantalla (*Figura 46*), el menú consta de las siguientes propiedades

1. Selecciona el motor que se desea controlar. Si se escoge controlar 2 motores (ejemplo: B y C), los motores funcionarán de manera sincronizada y con la misma potencia.
2. Controla la dirección del motor hacia delante, hacia atrás o detenerse.
3. Si se usa 2 motores con esto se puede controlar para girar o ir en línea recta.
4. Controla la potencia del motor que de 0 a 100%.
5. Controla las opciones de duración: ilimitada, segundos, rotaciones o grados.
6. Permite escoger entre las opciones brake o coast. Si se desea que el robot se detenga en una exacta posición se debe escoger brake.
7. Si el bloque NXT está conectado al computador estos campos mostrarán cuantos grados los motores giraron, tiene un botón reset para ponerlos en cero.

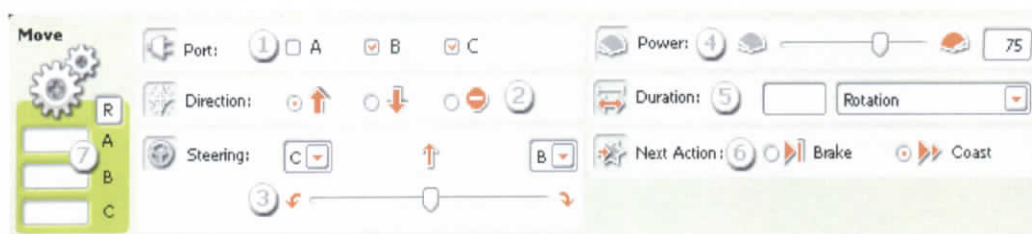


Figura 46. Menú de Configuración del Bloque de Movimiento

CENTRO DE DATOS

En la *Tabla 6* se muestran las características de los diferentes puntos del centro de datos del bloque de movimiento.









Img.	Punto	Tipo	Rango	Significado	Excep.
	Motor Izquierdo	Número	1 - 3	1 = A, 2 = B, 3 = C	
	Motor Derecho	Número	1 - 3	1 = A, 2 = B, 3 = C	
	Otro Motor	Número	1 - 3	1 = A, 2 = B, 3 = C	
	Dirección	Lógico	True/False	True = Adelante, False = Atrás	
	Giro	Número	-100 - 100	< 0 = motor izquierdo, > 0 = motor derecho	
	Potencia	Número	0 - 100		
	Duración	Número	0 - 2147483647	Depende del tipo de duración: Grados/Rotaciones = Grados, Segundos = Segundos	Tipo de Duración = Ilimitado
	Siguiente Acción	Lógico	True/False	True = Brake, False = Coast	Tipo de Duración = Ilimitado. Giro no es igual a cero

Tabla 6. Características centro de datos (bloque de movimiento)
(MINDSOTRM; "Ayuda y Soporte para LEGO Mindstorm NXT", 1ª Edición)

3.1.2.10 Bloque de Sonido

El bloque de sonido (*Figura 47*) se utiliza para reproducir un sonido o un tono. Si se quiere componer una melodía de tonos se debe colocar varios bloques de sonido en una fila y en cada uno poner un tono diferente.

Si se configura la opción “Wait for completion”, el sonido o tono se terminará de reproducir antes de continuar con la acción del siguiente bloque. Sin esta opción el sonido o tono se reproducirá mientras se ejecutan los bloques siguientes. Si se escoge “Repeat” el sonido o tono se reproducirá una y otra vez.



Figura 47. Bloque de Sonido

1. Este ícono muestra cuando el sonido es reproducido.
2. Este ícono muestra si el bloque inicia o detiene la reproducción del sonido.
3. Este ícono muestra el volumen.

3.1.2.11 Bloque Switch

El bloque switch (*Figura 48*) se utiliza cuando se quiere escoger entre 2 secuencias de código. Por ejemplo, con el sensor de tacto, el bloque switch puede ejecutar una secuencia cuando el sensor está presionado y otra cuando no lo está.

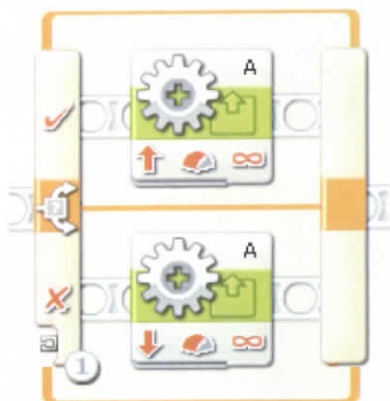


Figura 48. Bloque Switch

APARIENCIA

La apariencia (Figura 49) consta de:

1. Este ícono indica el sensor u otra condición que controlará la secuencia que el bloque escoja.
2. Secuencia verdadera o primera secuencia dependiendo del control.
3. Secuencia falsa o segunda secuencia dependiendo del control.

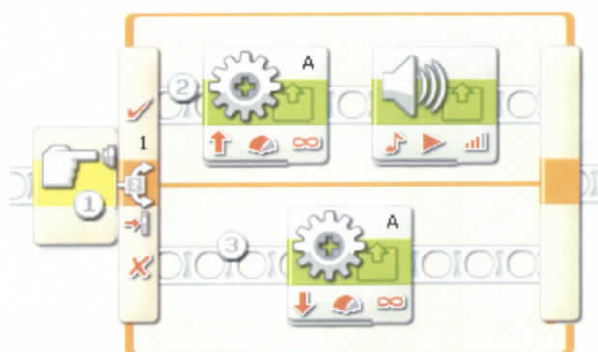


Figura 49. Apariencia

CONFIGURACIÓN

Si el control del bloque es "Value" (Figura 50) este aceptará tanto un valor lógico o uno numérico.

Si se decide que sea un valor lógico, el bloque funcionará ejecutando la primera secuencia si el valor es verdadero (True) y ejecutando la segunda secuencia si el valor es falso (False).

Si se decide que sea un valor numérico, se puede especificar un valor exacto que produzca la ejecución de las diferentes secuencias, con el tipo de valor numérico puede haber más de 2 secuencias.

1. En este campo se mostrará el tipo de entrada detectada por el bloque.
2. Si se deselecciona “Display Flat View”, el bloque usará una interfaz de pestañas para mostrar las diferentes secuencias.
3. Lista de valores para controlar cada una de las secuencias.
4. Este campo sirve para establecer el valor de acuerdo a la secuencia que se seleccione en la lista.
5. Estos botones permiten agregar o eliminar secuencias.
6. Este botón establece la secuencia “Default” que se ejecutará en caso de que el valor que llegue no sea igual a ninguno de los valores que controlan las secuencias.



Figura 50. Configuración Bloque Switch (Value)

Si el control del bloque depende del sensor de tacto (*Figura 51*) se tendrá la siguiente configuración.

1. Si se deselecciona “Display Flat View”, el bloque usará una interfaz de pestañas para mostrar las diferentes secuencias.
2. Selección del puerto donde está conectado el sensor.
3. Opciones de las diferentes condiciones del sensor, establece en cual condición el bloque ejecutará la primera secuencia. La segunda secuencia se ejecutará cuando no ocurra ninguna acción.

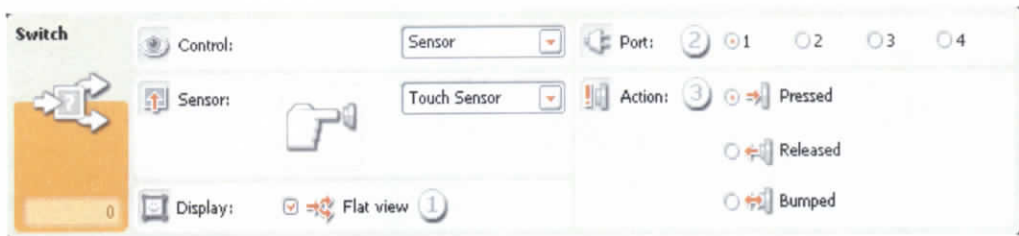


Figura 51. Configuración Bloque Switch (Sensor de Tacto)

Si el control del bloque depende del sensor de sonido (Figura 52) se tendrá la siguiente configuración.

1. Si se deselecciona “Display Flat View”, el bloque usará una interfaz de pestañas para mostrar las diferentes secuencias.
2. Selección del puerto donde está conectado el sensor.
3. Establece la condición o el valor del nivel de sonido en el cual se ejecutará la primera secuencia caso contrario se ejecutará la segunda secuencia.



Figura 52. Configuración Bloque Switch (Sensor de Sonido)

Si el control del bloque depende del sensor de luz (Figura 53) se tendrá la siguiente configuración.

1. Si se deselecciona “Display Flat View”, el bloque usara una interfaz de pestañas para mostrar las diferentes secuencias.
2. Selección del puerto donde está conectado el sensor.
3. Establece la condición o el valor del nivel de luz en el cual se ejecutará la primera secuencia caso contrario se ejecutará la segunda secuencia.
4. Si se selecciona la opción “Generated Light”, el sensor generará una pequeña luz y detectará si esta luz es reflejada.
5. Este campo muestra la actual lectura del nivel de luz (0-100%).



Figura 53. Configuración Bloque Switch (Sensor de Luz)

Si el control del bloque depende del sensor de ultrasonido (Figura 54) se tendrá la siguiente configuración.

1. Si se deselecciona “Display Flat View”, el bloque usará una interfaz de pestañas para mostrar las diferentes secuencias.
2. Selección del puerto donde está conectado el sensor.
3. Establece la condición o el valor de la distancia en la cual se ejecutará la primera secuencia caso contrario se ejecutará la segunda secuencia.
4. Establece si la distancia es en pulgadas o centímetros.
5. Este campo muestra la actual lectura del sensor.

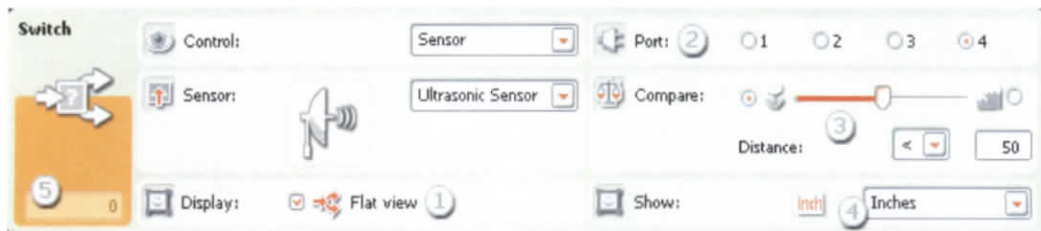


Figura 54. Configuración Bloque Switch (Sensor de Ultrasonido)

Si el control del bloque depende de los botones del bloque NXT (Figura 55) se tendrá la siguiente configuración.

1. Establece cual botón del bloque NXT controlará la selección de la secuencia.
2. Establece en qué condición se ejecutará la primera secuencia, la segunda secuencia se ejecutará cuando no se realice ninguna acción.

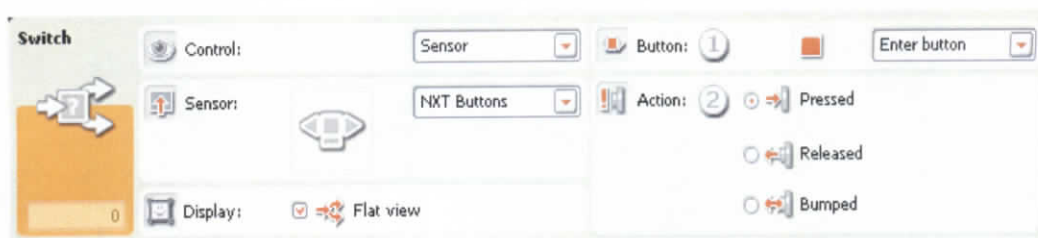


Figura 55. Configuración Bloque Switch (Bloque NXT)

3.1.2.12 Bloque de Espera

El bloque de espera (Figura 56) sirve para que el robot se mantenga en espera hasta que ocurra alguna determinada acción en el control que se establezca, y pueda continuar.



Figura 56. Bloque de Espera

CONFIGURACIÓN

Si el control del bloque depende del sensor de tacto (Figura 57) se tendrá la siguiente configuración.

1. Selección del puerto donde está conectado el sensor.
2. Opciones de las diferentes condiciones del sensor, establece en cual condición el bloque dejará de esperar.
3. Muestra la condición del sensor si aparece el valor de 1 es porque el sensor ha sido activado.

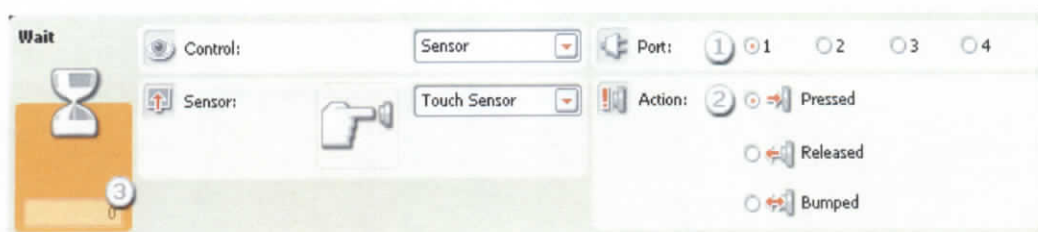


Figura 57. Configuración Bloque de Espera (Sensor de Tacto)

Si el control del bloque depende del sensor de sonido (Figura 58) se tendrá la siguiente configuración.

1. Selección del puerto donde está conectado el sensor.
2. Establece la condición o el valor del nivel de sonido en el cual se el bloque dejará de esperar.
3. Este campo muestra la actual lectura del nivel de sonido (0-100%).



Figura 58. Configuración Bloque de Espera (Sensor de Sonido)

Si el control del bloque depende del sensor de luz (Figura 59) se tendrá la siguiente configuración.

1. Selección del puerto donde está conectado el sensor.
2. Establece la condición o el valor del nivel de luz en el cual se el bloque dejará de esperar.
3. Si se selecciona la opción “Generated Light”, el sensor generará una pequeña luz y detectará si esta luz es reflejada.
4. Este campo muestra la actual lectura del nivel de luz (0-100%).



Figura 59. Configuración Bloque de Espera (Sensor de Luz)

Si el control del bloque depende del sensor de ultrasonido (Figura 60) se tendrá la siguiente configuración.

1. Selección del puerto donde está conectado el sensor.

2. Establece la condición o el valor de la distancia en la cual el bloque dejará de esperar.
3. Establece si la distancia es en pulgadas o centímetros.
4. Este campo muestra la actual lectura del sensor.

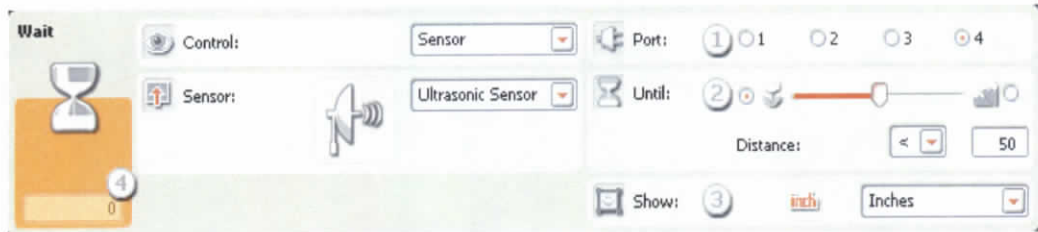


Figura 60. Configuración Bloque de Espera (Sensor de Ultrasonido)

Si el control del bloque depende de los botones del bloque NXT (Figura 61) se tendrá la siguiente configuración.

1. Establece cual botón del bloque NXT controlará la interrupción de la espera.
2. Establece en que condición se terminará la espera.



Figura 61. Configuración Bloque de Espera (Bloque NXT)

3.1.2.13 Bloque de Interrupción

El bloque de interrupción (Figura 62) detiene el programa, todos los motores, lámparas y sonidos.




Figura 62. Bloque de Interrupción

Este bloque no contiene ningún parámetro de configuración.

CENTRO DE DATOS

La *Tabla 7* muestra las características de los diferentes puntos del centro de datos del bloque de interrupción.

Img.	Punto	Tipo	Rango	Significado	Excep.
	Stop	Lógico	True/False	“True” si se quiere que el programa se detenga	

*Tabla 7. Características centro de datos (bloque de interrupción)(
(MINDSOTRM; "Ayuda y Soporte para LEGO Mindstorm NXT", 1º Edición))*

3.1.2.14 Bloque de Motor

Este bloque (*Figura 63*) permite un control preciso de la velocidad de un motor. Se puede subir o detener la velocidad del motor. La opción “Wait for Completion” el programa ejecutará el siguiente bloque cuando el bloque de motor haya comenzado.



Figura 63. Bloque de Motor

En la (*Figura 64*) se describe el ícono del bloque de motor:

1. La letra en la esquina superior derecha del bloque muestra en cual puerto del bloque NXT está puesto el control de motor.
2. Este ícono muestra la dirección del motor.
3. Este ícono muestra el nivel de potencia.
4. Este ícono muestra cual sea la propiedad de duración ilimitada, grados, rotación, o segundos.
5. Centro de datos.

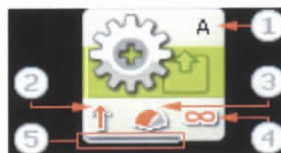


Figura 64. Descripción Bloque de Motor

CONFIGURACIÓN

El bloque de motor tiene la siguiente configuración (Figura 65):

1. Selector del motor que se desea controlar.
2. Selector de la dirección del motor adelante, atrás o detenerse.
3. Control de la aceleración del motor.
4. Control del nivel de potencia (0 a 100%).
5. Configuración de encendido o apagado del control de potencia, este compensa la potencia en caso de que esta baje y puede mantenerla al 100% por unos pocos segundos.
6. Control de duración del funcionamiento del motor que puede ser ilimitado, o por un número de rotaciones, grados o segundos.
7. Con “Wait for completion” accionado, el motor finalizará la acción completamente antes de continuar con el programa.
8. Control de la acción del motor al finalizar su acción, si selecciona “break” el motor se detendrá y mantendrá su posición, se debe seleccionar “coast” si a este bloque le siguen otros que controlen el mismo motor.
9. En este cuadro se muestra cuántos grados o rotaciones dio el motor, para recibir esta información el bloque NXT debe estar conectado al computador.

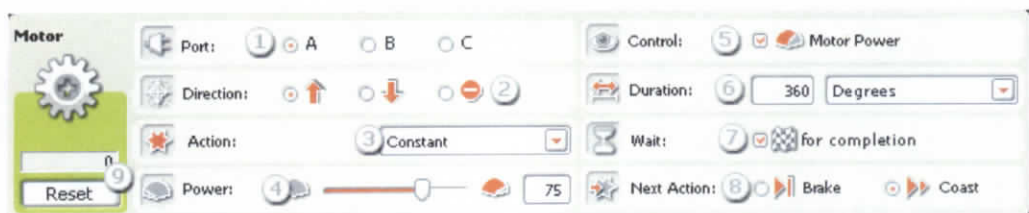


Figura 65. Configuración Bloque de Motor

3.1.2.15 Bloque del Sensor de Luz

Este sensor detecta la luz ambiental. En este sensor se puede establecer un punto de control. El punto de control es un valor específico en un rango numérico. La configuración predeterminada del bloque es para el nivel de luz sobre el 50% para generar una señal verdadera.

Si la función “Generated Light” está activa, el sensor de luz encenderá una pequeña luz y detectará esta luz si es reflejada. Esta función es especialmente usada en

condiciones de baja de difícil luminosidad como cuartos con luz muy brillante. La función también permite al sensor de luz mantener una corta distancia y podría usarse para esquivar obstáculos.

En la (Figura 66) se describe el ícono del bloque del sensor de luz:

1. El número muestra en que puerto del bloque NXT está conectado el sensor.
2. Este ícono indica a que nivel el punto de control fue configurado.
3. Este ícono indica si la función “Generated Light” está activada.
4. El centro de datos del bloque se abre automáticamente cuando el bloque es colocado en el área de trabajo.

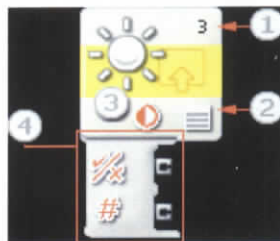


Figura 66. Descripción Bloque del Sensor de Luz

CONFIGURACIÓN

El bloque del sensor de luz tiene la siguiente configuración(Figura 67):

1. Indica el puerto en el que el sensor está conectado. Por predeterminedo, el bloque pone el puerto 3.
2. Determina el valor de control y la condición de este para el valor verdadero y el falso.
3. Determina la activación o no de la función “Generated Light”.
4. Muestra la lectura del valor de luz actual para lo cual el bloque NXT debe estar conectado al computador.



Figura 67. Configuración Bloque del Sensor de Luz

3.1.2.16 Bloque del Sensor de Sonido

Es un bloque detector de sonido. En este sensor se puede establecer un punto de control. El punto de control es un valor específico en un rango numérico. La configuración predeterminada del bloque es para el nivel de sonido sobre el 50% para generar una señal verdadera.

En la (Figura 68) se describe el ícono del bloque del sensor de sonido:

1. El número muestra en cual puerto del bloque NXT está conectado el sensor de sonido.
2. Este ícono indica en nivel del punto de control.
3. El centro de datos del bloque se abre automáticamente cuando el bloque es colocado en el área de trabajo.

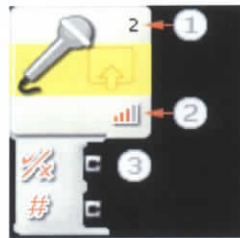


Figura 68. Descripción Bloque del Sensor de Sonido

CONFIGURACIÓN

El bloque del sensor de sonido tiene la siguiente configuración (Figura 69):

1. Indica el puerto en el que el sensor está conectado. Por predeterminado, el bloque pone el puerto 2.
2. Determina el valor de control y la condición de este para el valor verdadero y el falso.
3. Muestra la lectura del valor de luz actual para lo cual el bloque NXT debe estar conectado al computador.

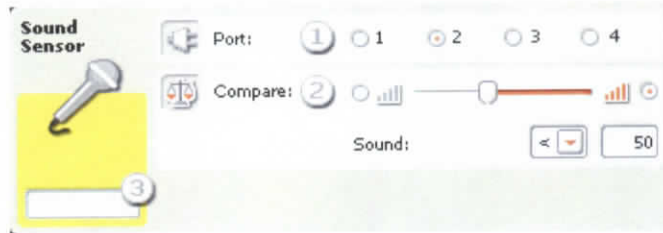


Figura 69. Configuración Bloque del Sensor de Sonido

3.1.2.17 Bloque del Sensor de Tacto

Es un bloque que verifica la condición del sensor en un punto específico en el programa. Si el sensor fue activado el bloque enviará un valor de verdadero; y si no es disparado el bloque enviará un valor de falso.

En la (Figura 70) se describe el ícono del bloque del sensor de tacto:

1. El número muestra en cual puerto del bloque NXT está conectado el sensor de sonido.
2. Este ícono indica que acción fue determinada como verdadera.
3. El centro de datos del bloque se abre automáticamente cuando el bloque es colocado en el área de trabajo.

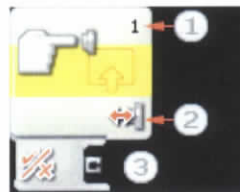


Figura 70. Descripción Bloque del Sensor de Tacto

CONFIGURACIÓN

El bloque del sensor de tacto tiene la siguiente configuración (Figura 71):

1. Indica el puerto en el que el sensor está conectado. Por predeterminado, el bloque pone el puerto 1.
2. El sensor de tacto se activa de 3 maneras. “Bumped” si se quiere que se activa en un presionado y levantado rápido (en menos de 0,5 segundos). “Pressed” si se quiere

que se activa al momento que se presiona. “Released” si se quiere que se active al momento que se levanta de lo presionado.

3. Muestra la lectura si el sensor fue activado o no si fue activado saldrá el valor de 1.

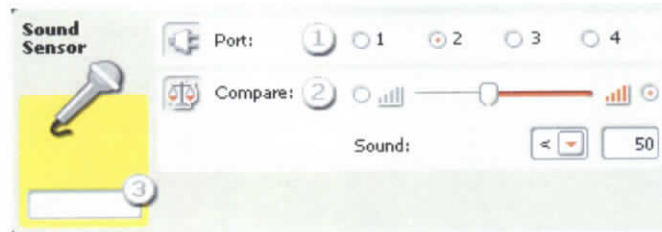


Figura 71. Configuración Bloque del Sensor de Tacto

3.1.2.18 Bloque del Sensor de Ultrasonido

Es un bloque que puede detectar objetos a un rango máximo de 250cm. En este sensor se puede establecer un punto de control. El punto de control es un valor específico en un rango numérico. La configuración predeterminada del bloque es para el nivel de sonido sobre el 50cm para generar una señal verdadera.

En la (Figura 72) se describe el ícono del bloque del sensor de ultrasonido:

1. El número muestra en cual puerto del bloque NXT está conectado el sensor de sonido.
2. Este ícono indica el nivel del valor de control.
3. El centro de datos del bloque se abre automáticamente cuando el bloque es colocado en el área de trabajo.

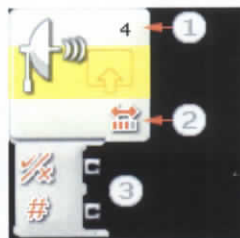


Figura 72. Descripción Bloque del Sensor de Ultrasonido

CONFIGURACIÓN

El bloque del sensor de ultrasonido tiene la siguiente configuración (Figura 73):

1. Indica el puerto en el que el sensor está conectado. Por predeterminado, el bloque pone el puerto 4.
2. Determina el valor de control y la condición de éste para el valor verdadero y el falso.
3. Determina la unidad en que se va a medir sea en centímetros o pulgadas.
4. Muestra la lectura del sensor de 0 a 255cm.

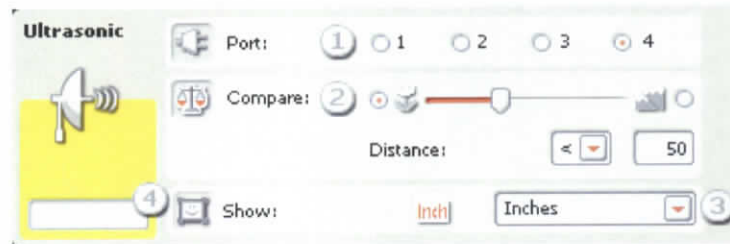


Figura 73. Configuración Bloque del Sensor de Ultrasonido

3.1.2.19 Bloque Timer

Cuando un programa inicia los 3 timers en el NXT empiezan a funcionar. Con este bloque se puede escoger cualquiera de las lecturas del actual valor de los timers o causar comience a funcionar desde 0 de nuevo. En este bloque se puede determinar un punto de control. Un punto de control es un valor específico en un rango numérico.

En la (Figura 74) se describe el icono del bloque timer:

1. Este número muestra cuál de los 3 timers está siendo monitoreado con este bloque.
2. El centro de datos del bloque se abre automáticamente cuando el bloque es colocado en el área de trabajo.

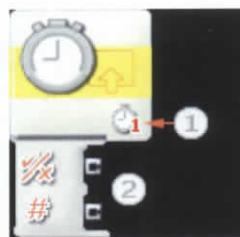


Figura 74. Descripción Bloque Timer

CONFIGURACIÓN

El bloque del sensor de ultrasonido tiene la siguiente configuración (*Figura 75*):

1. Determina cual de los 3 timers se va a controlar.
2. Determina y muestra las opciones de control. Read si solo se quiere obtener el valor del timer. Reset si se quiere que el timer comience desde 0.
3. Determina la condición en la cual se quiere que el bloque se ejecute.



Figura 75. Configuración Bloque Timer

3.2 Humanoide.

A continuación se describirá la construcción y programación de un robot humanoide de dos piernas capaz de caminar, girar y responder al habla.

3.2.1 Caminar.

Construcción y programación del caminar del humanoide.

3.2.1.1 Construcción.

Construcción de las 2 piernas del humanoide. Las siguientes imágenes muestran de manera ilustrada paso a paso la construcción de las piernas del humanoide.

PASO 1

Se necesita (*Figura 76*):

- 2 conectores azules.
- 2 conectores negros pequeños
- 1 viga angulada grande.

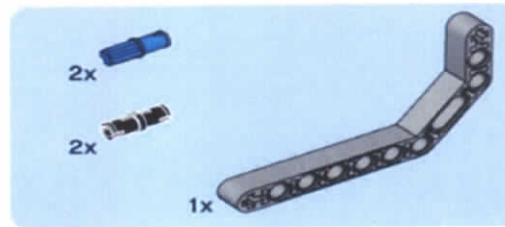


Figura 76. Elementos Paso 1 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 77):

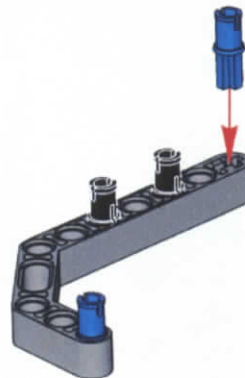


Figura 77. Ensamblaje Paso 1 (Caminar)

PASO 2

Se necesita (Figura 78):

- 2 conectores azules.
- 1 conectores negros pequeños
- 1 viga angulada grande.

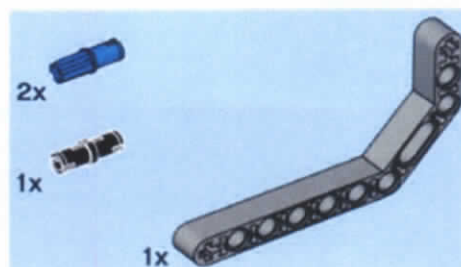


Figura 78. Elementos Paso 2 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 79):



Figura 79. Ensamblaje Paso 2 (Caminar)

PASO 3

Se necesita (Figura 80):

- 1 viga angulada pequeña.

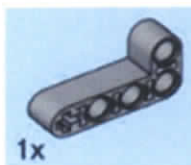


Figura 80. Elementos Paso 3 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 81):

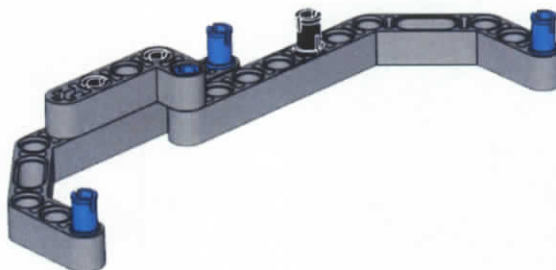


Figura 81. Ensamblaje Paso 3 (Caminar)

PASO 4

Se necesita (Figura 82):

- 2 conectores azules.
- 2 viga angulada pequeña.

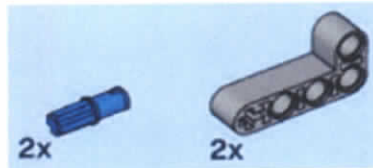


Figura 82. Elementos Paso 4 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 83):

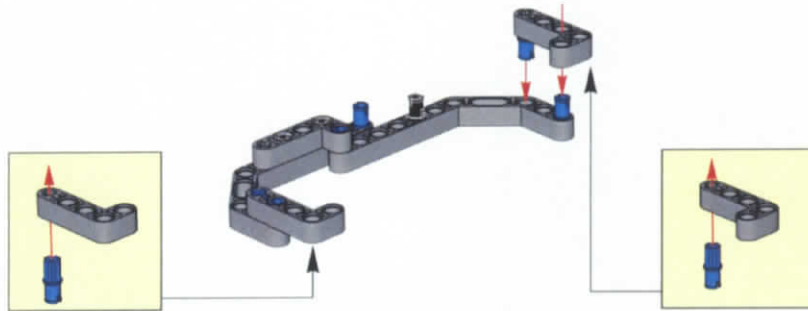


Figura 83. Ensamblaje Paso 4 (Caminar)

PASO 5

Se necesita (Figura 84):

- 7 conectores negros pequeños.
- 1 viga grande.

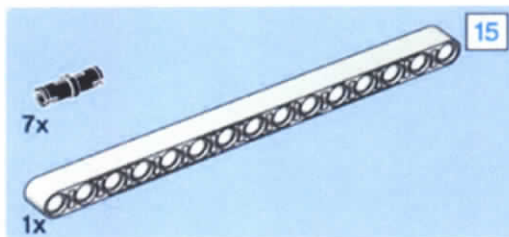


Figura 84. Elementos Paso 5 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 85):

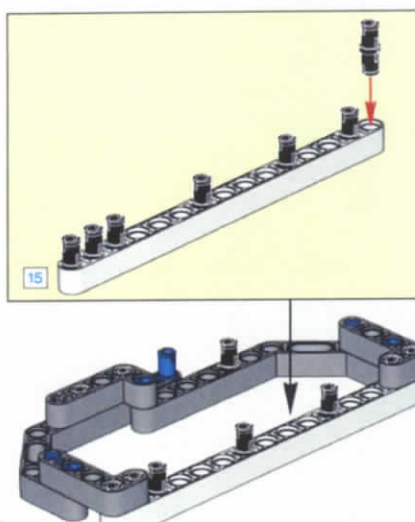


Figura 85. Ensamblaje Paso 5 (Caminar)

PASO 6

Se necesita (Figura 86):

- 4 conectores azules.
- 1 bloque conector pequeño.
- 1 eje mediano.
- 2 ejes extensores.
- 1 bloque conector mediano.
- 1 bloque conector grande.

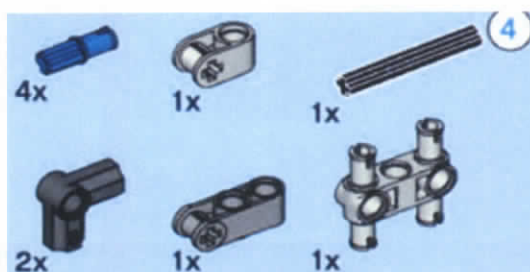


Figura 86. Elementos Paso 6 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 87):

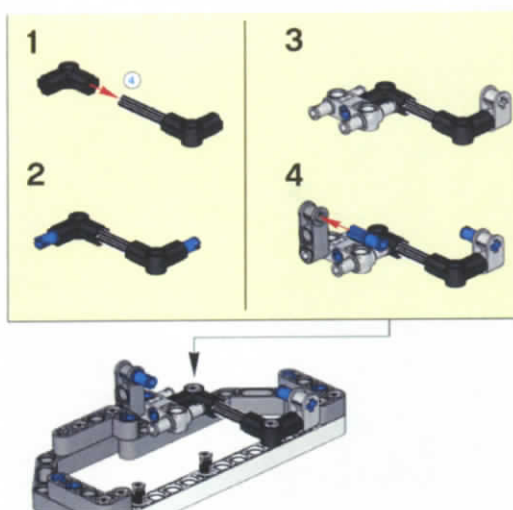


Figura 87. Ensamblaje Paso 6 (Caminar)

PASO 7

Se necesita (Figura 88):

- 1 conectores azules.
- 1 bloque conector pequeño.
- 1 eje extensor.
- 1 eje pequeño.



Figura 88. Elementos Paso 7 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 89):

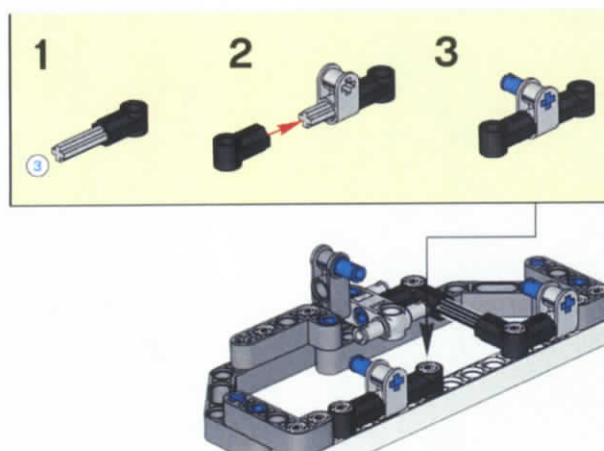


Figura 89. Ensamblaje Paso 7 (Caminar)

PASO 8

Se necesita (*Figura 90*):

- 2 conectores negros pequeños.
- 2 conectores negros medianos.
- 1 viga grande.

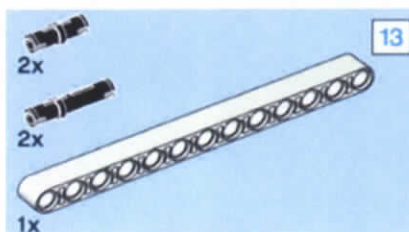


Figura 90. Ensamblaje Paso 8 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (*Figura 91*):

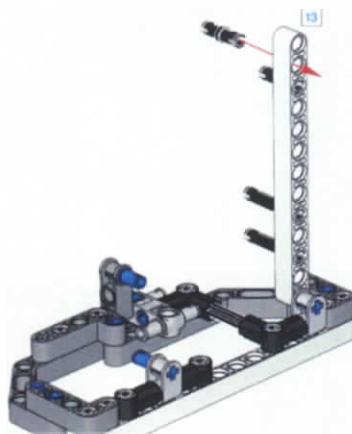


Figura 91. Ensamblaje Paso 8 (Caminar)

PASO 9

Se necesita (Figura 92):

- 1 conectores negro pequeño.
- 1 viga pequeña.

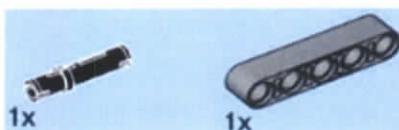


Figura 92. Elementos Paso 9 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 93):



Figura 93. Ensamblaje Paso 9 (Caminar)

PASO 10

Se necesita (*Figura 94*):

- 1 conectores negro pequeño.
- 1 bloque conector pequeño.



Figura 94. Elementos Paso 10 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (*Figura 95*):

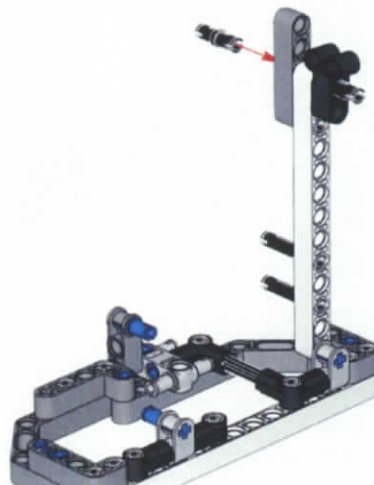


Figura 95. Ensamblaje Paso 10 (Caminar)

PASO 11

Se necesita (*Figura 96*):

- 1 conectores azul.
- 1 conector negro pequeño.
- 1 conector blanco grande.
- 1 viga angulada pequeña.
- 1 viga grande.

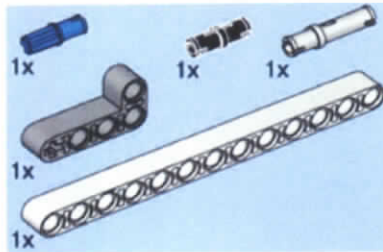


Figura 96. Elementos Paso 11 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 97):

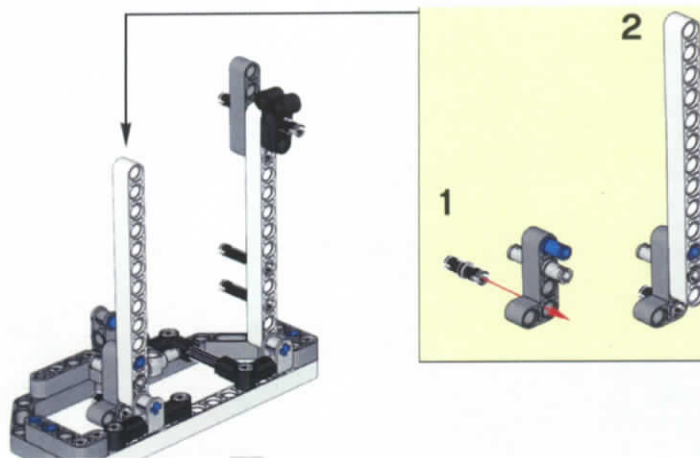


Figura 97. Ensamblaje Paso 11 (Caminar)

PASO 12

Se necesita (Figura 98):

- 2 conectores negros pequeños.
- 2 conectores negros grandes.



Figura 98. Elementos Paso 12 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 99):



Figura 99. Ensamblaje Paso 12 (Caminar)

PASO 13

Se necesita (Figura 100):

- 2 vigas reductoras negras.
- 1 viga reductora horizontal grande.

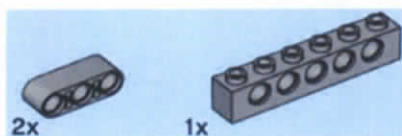


Figura 100. Elementos Paso 13 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 101):

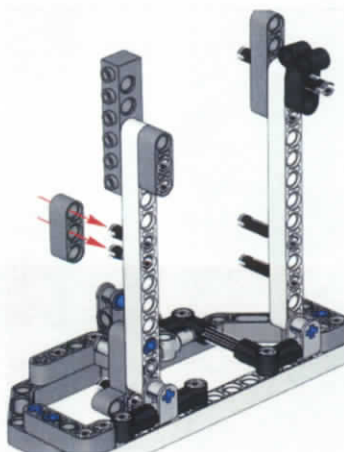


Figura 101. Ensamblaje Paso 13 (Caminar)

PASO 14

Se necesita (*Figura 102*):

- 2 conectores negros pequeños.
- 1 conector azul.
- 1 viga pequeña.



Figura 102. Elementos Paso 14 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (*Figura 103*):

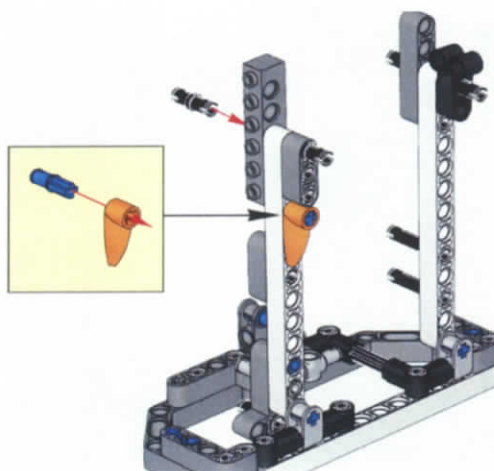


Figura 103. Ensamblaje Paso 14 (Caminar)

PASO 15

Se necesita (*Figura 104*):

- 1 conector negro pequeño.
- 1 viga mediana.

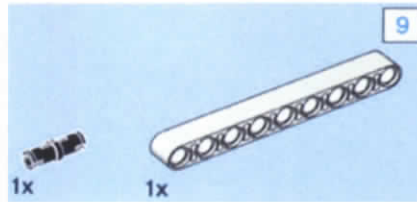


Figura 104. Elementos Paso 15 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 105):



Figura 105. Ensamblaje Paso 15 (Caminar)

PASO 16

Se necesita (Figura 106):

- 1 conector azul.
- 1 bloque conector pequeño.



Figura 106. Elementos Paso 16 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 107):

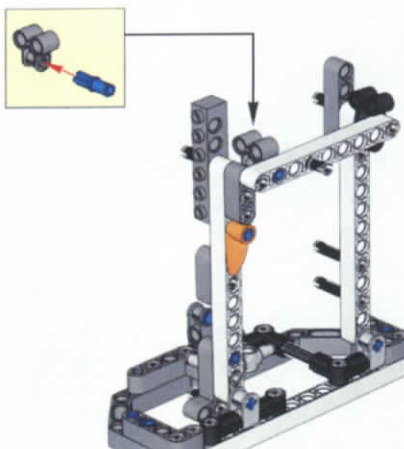


Figura 107. Ensamblaje Paso 16 (Caminar)

Con eso terminamos la primera parte de la construcción y debe quedar una estructura como la siguiente (*Figura 108*):



Figura 108. Estructura Parte 1 (Caminar)

PASO 17

Repetir desde el paso 1 hasta el paso 16 pero todo al lado opuesto deberá quedar la siguiente estructura (*Figura 109*):



Figura 109. Estructura Parte 2 (Caminar)

PASO 18

Se necesita (Figura 110):

- 1 servomotor.
- 1 viga mediana.
- 1 eje mediano.
- 1 engranaje pequeño
- 1 eje extensor simple.

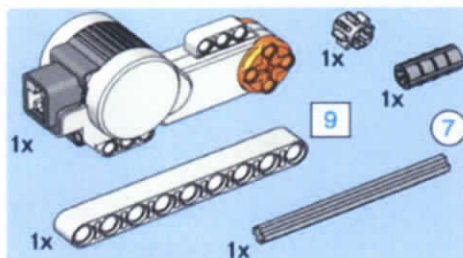


Figura 110. Elementos Paso 18 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 111):

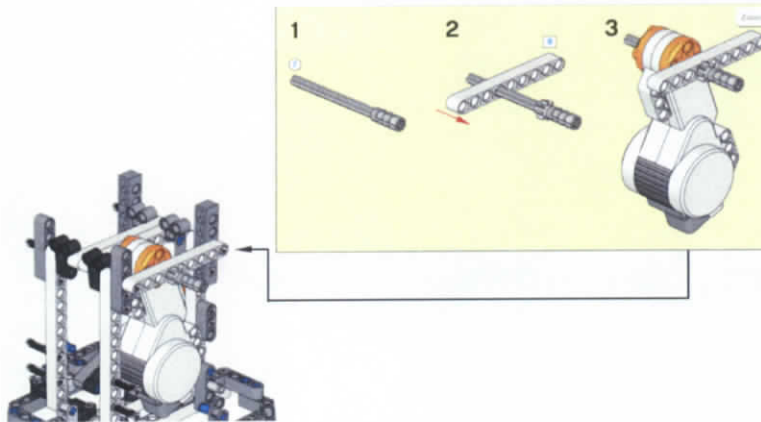


Figura 111. Ensamblaje Paso 18 (Caminar)

PASO 19

Se necesita (Figura 112):

- 1 engranaje mediano.
- 1 eje grande.

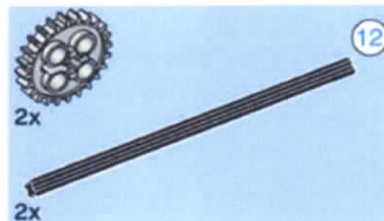


Figura 112. Elementos Paso 19 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 113):

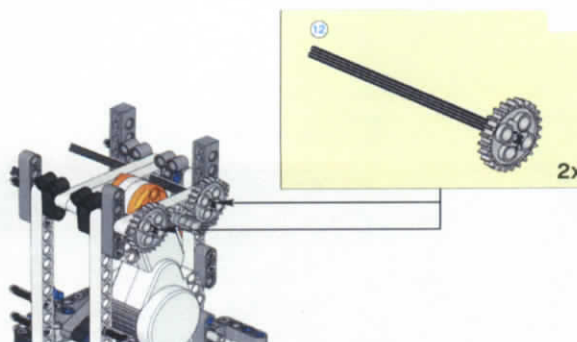


Figura 113. Ensamblaje Paso 19 (Caminar)

PASO 20

Se necesita (*Figura 114*):

- 1 conector pequeño.
- 1 bloque conector negro doble.
- 1 viga angulada mediana.
- 1 viga angulada pequeña.

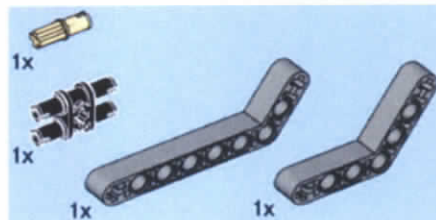


Figura 114. Elementos Paso 20 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (*Figura 115*):

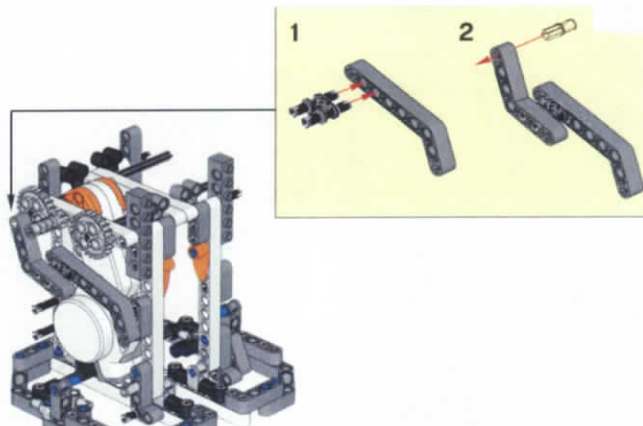


Figura 115. Ensamblaje Paso 20 (Caminar)

PASO 21

Se necesita (*Figura 116*):

- 1 anillo pequeño.
- 1 conector negro mediano.

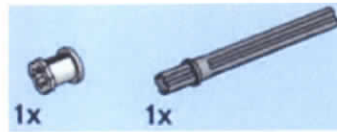


Figura 116. Elementos Paso 21 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 117):

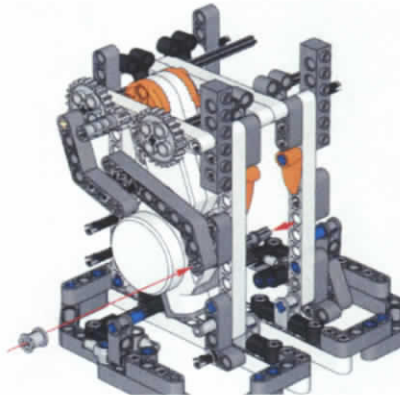


Figura 117. Ensamblaje Paso 21 (Caminar)

PASO 22

Se necesita (Figura 118):

- 1 eje extensor pequeño.
- 3 conectores azules.
- 1 conector negro pequeño.
- 1 eje extensor blanco.
- 1 viga angulada pequeña.
- 1 viga angulada grande.
- 1 viga pequeña.

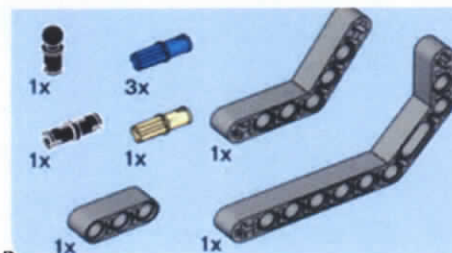


Figura 118. Elementos Paso 22 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (*Figura 119*):

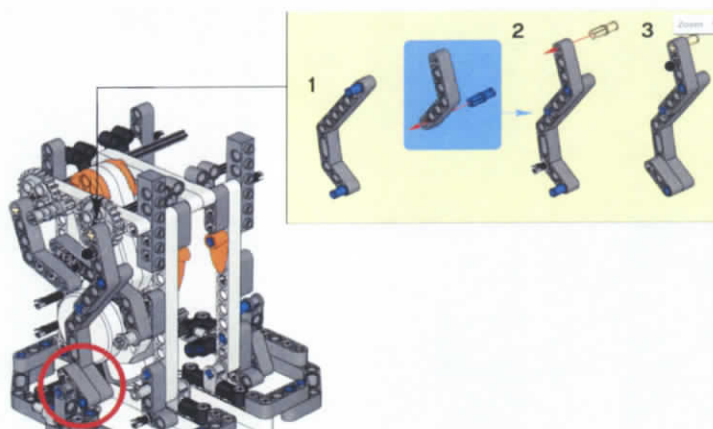


Figura 119. Ensamblaje Paso 22 (Caminar)

PASO 23

Se necesita (*Figura 120*):

- 1 servomotor.
- 1 viga mediana.
- 1 eje mediano.
- 1 engranaje pequeño
- 1 eje extensor simple.

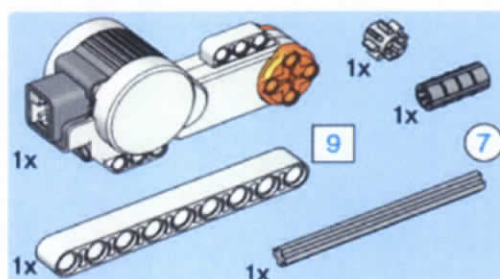


Figura 120. Elementos Paso 23 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (*Figura 121*):

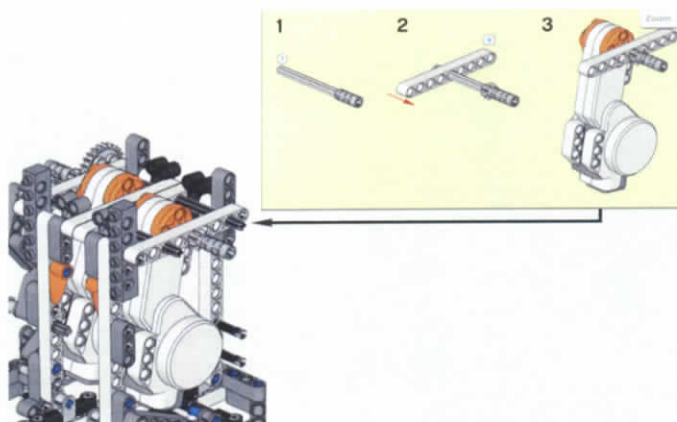


Figura 121. Ensamblaje Paso 23 (Caminar)

PASO 24

Se necesita (Figura 122):

- 2 engranajes medianos.



Figura 122. Elementos Paso 24 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 123):

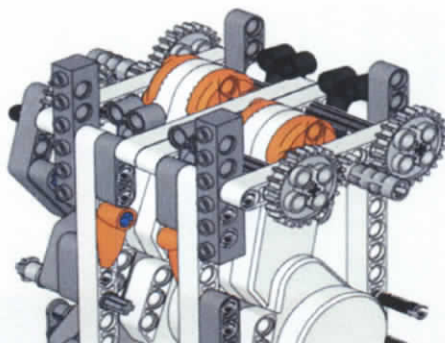


Figura 123. Ensamblaje Paso 24 (Caminar)

PASO 25

Se necesita (Figura 124):

- 1 conector pequeño.
- 1 bloque conector negro doble.
- 1 viga angulada mediana.
- 1 viga angulada pequeña.

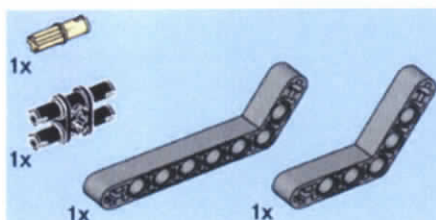


Figura 124. Elementos Paso 25 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 125):

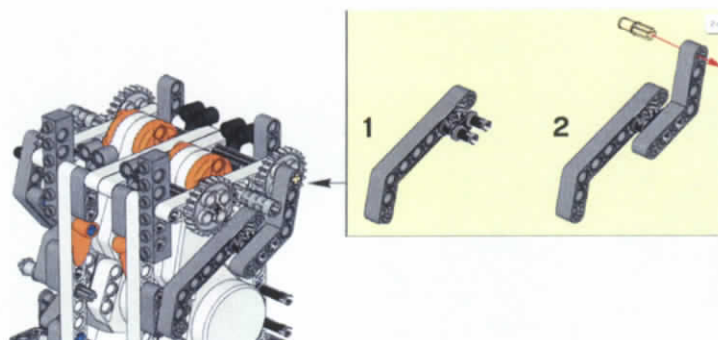


Figura 125. Ensamblaje Paso 25 (Caminar)

PASO 26

Se necesita (Figura 126):

- 1 anillo pequeño.
- 1 conector negro mediano.



Figura 126. Elementos Paso 26 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 127):

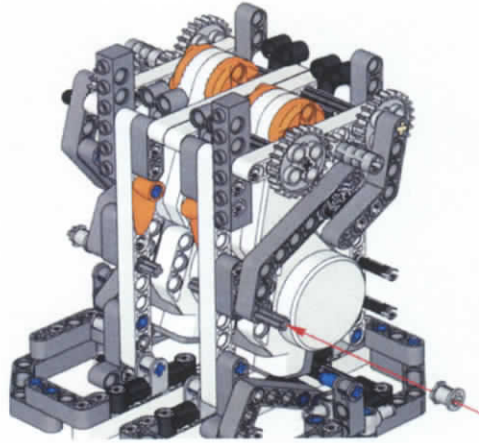


Figura 127. Ensamblaje Paso 27 (Caminar)

PASO 28

Se necesita (Figura 128):

- 1 eje extensor pequeño.
- 3 conectores azules.
- 1 conector negro pequeño.
- 1 eje extensor blanco.
- 1 viga angulada pequeña.
- 1 viga angulada grande.
- 1 viga pequeña.

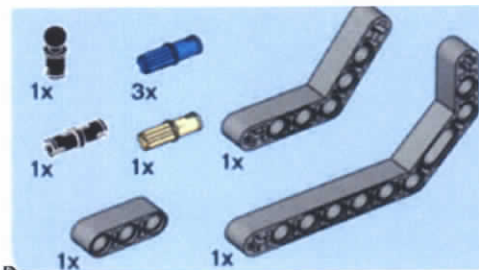


Figura 128. Elementos Paso 28 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 129):

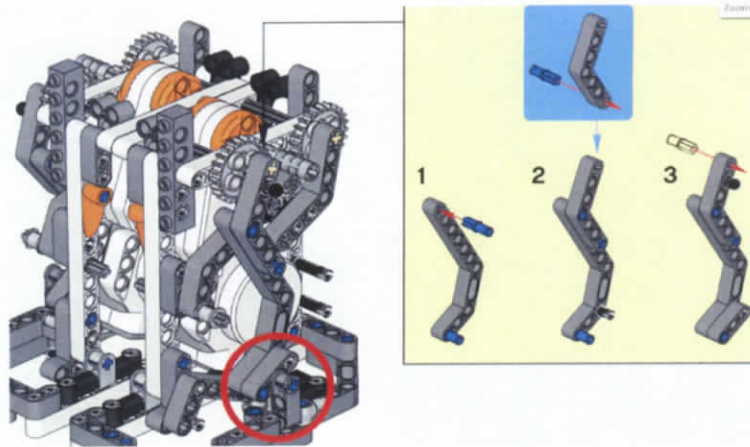


Figura 129. Ensamblaje Paso 28 (Caminar)

PASO 29

Se necesita (Figura 129):

- 4 conectores azules.
- 4 viga reductora grande.



Figura 129. Elementos Paso 29 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 130):

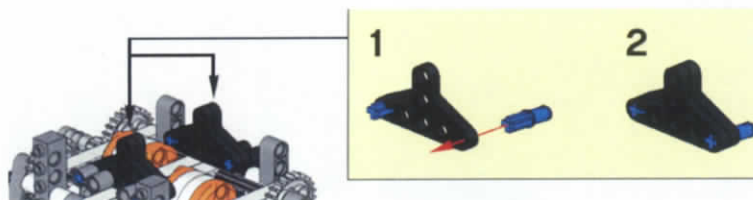


Figura 130. Ensamblaje Paso 29 (Caminar)

PASO 30

Se necesita (Figura 131):

- 2 cables de conexión de 50cm.

- Bloque NXT.

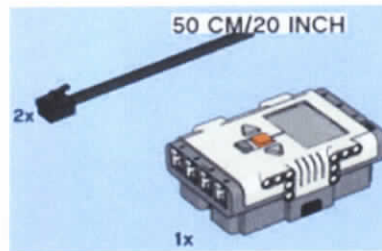


Figura 131. Elementos Paso 30 (Caminar)

Los anteriores elementos se deberán ensamblar de la siguiente manera (*Figura 132*):



Figura 132. Ensamblaje Paso 30 (Caminar)

3.2.1.2 Programación.

Programación para que las piernas caminen hacia delante mientras un corazón es desplegado en la pantalla del bloque NXT. Las siguientes imágenes muestran de manera ilustrada paso a paso la estructura y configuración de la programación.

1. Colocar un bloque de repetición en el punto de inicio (*Figura 133*).

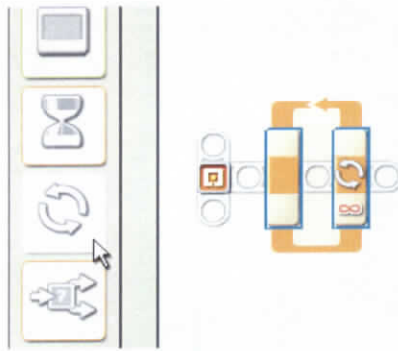


Figura 133. Paso 1 Programación (Caminar)

2. Configurar el bloque de repetición para que se mantenga activo siempre (forever) (Figura 134).

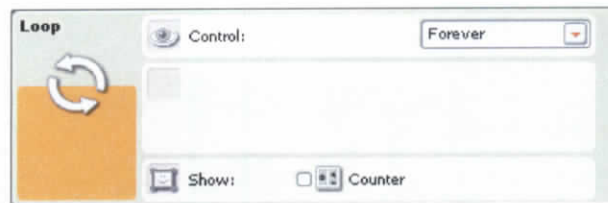


Figura 134. Paso 2 Programación (Caminar)

3. Dentro del bloque de repetición colocar un bloque de pantalla (Figura 135).



Figura 135. Paso 3 Programación (Caminar)

4. Configurar un bloque de pantalla para que muestre un corazón en una determinada posición de la pantalla del bloque NXT (Figura 136).

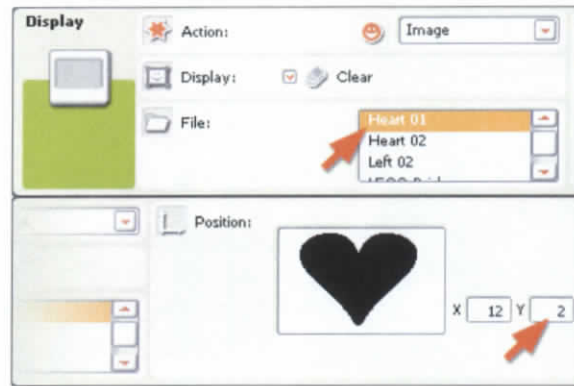


Figura 136. Paso 4 Programación (Caminar)

5. Colocar un bloque de espera después del bloque de pantalla (Figura 137).



Figura 137. Paso 5 Programación (Caminar)

6. Configurar el bloque de espera para un tiempo de 0,5 segundos (Figura 138).

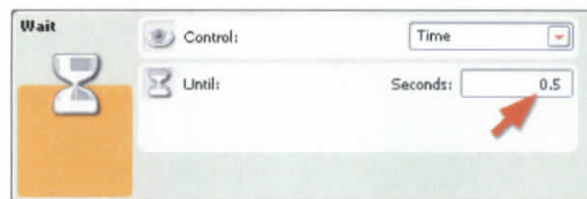


Figura 138. Paso 6 Programación (Caminar)

7. Colocar un bloque de pantalla después del bloque de espera (Figura 139).

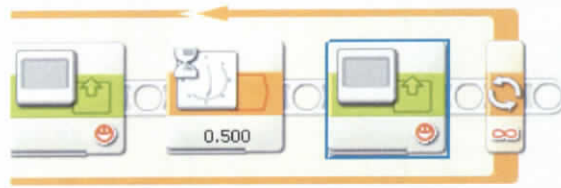


Figura 139. Paso 7 Programación (Caminar)

8. Configurar un bloque de pantalla para que muestre un corazón en una posición diferente a la determinada en el bloque de pantalla anterior de la pantalla del bloque NXT (Figura 140).

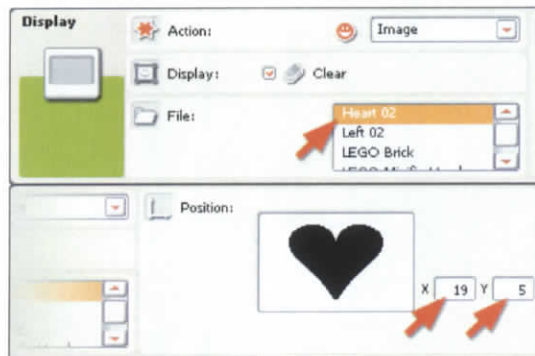


Figura 140. Paso 8 Programación (Caminar)

9. Colocar un bloque de espera después del anterior bloque de pantalla (Figura 141).



Figura 141. Paso 9 Programación (Caminar)

10. Configurar el segundo bloque de espera para un tiempo de 0,5 segundos (Figura 142).

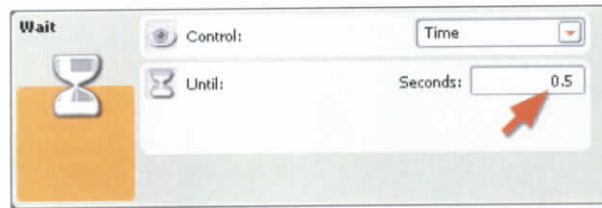


Figura 142. Paso 10 Programación (Caminar)

11. Colocar un bloque de movimiento fuera del bloque de repetición y separado de la secuencia beam del punto de inicio (Figura 143).

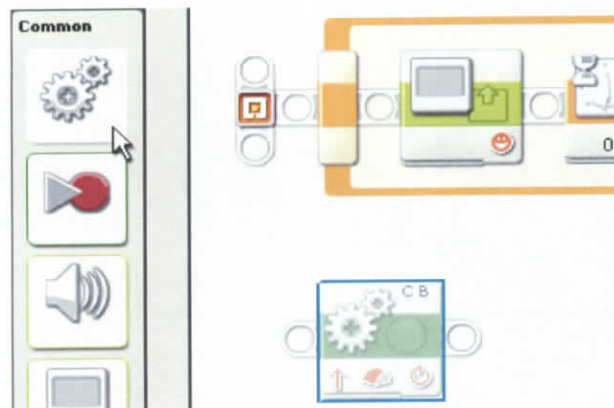


Figura 143. Paso 11 Programación (Caminar)

12. Configurar el bloque de movimiento para que tenga una duración de 28,5 rotaciones (Figura 144).

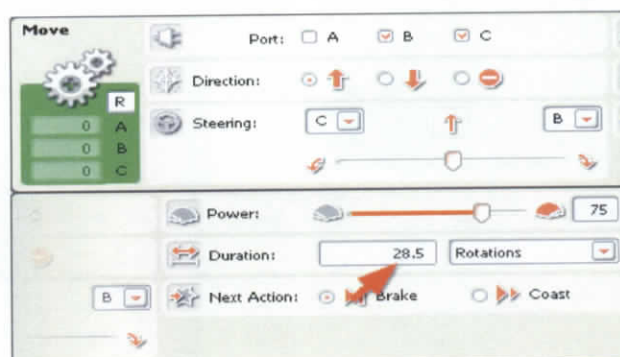


Figura 144. Paso 12 Programación (Caminar)

13. Agrandar la secuencia beam teniendo presionado la tecla “shift” mientras arrastramos con el mouse (Figura 145).

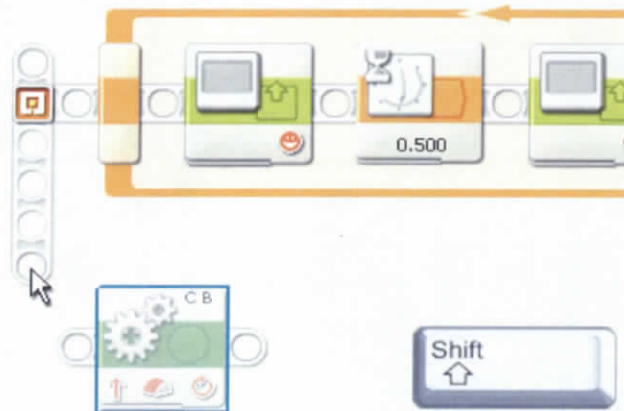


Figura 145. Paso 13 Programación (Caminar)

14. Unir la secuencia beam agrandada con el bloque de movimiento (Figura 146).

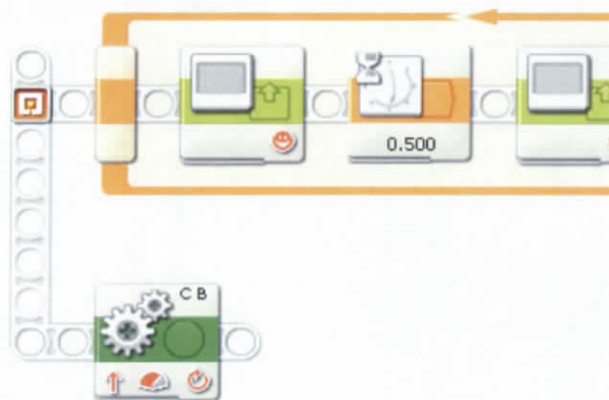


Figura 146. Paso 14 Programación (Caminar)

3.2.2 Cuerpo.

Construcción y programación del cuerpo del humanoide.

3.2.2.1 Construcción.

Construcción del cuerpo del humanoide para que sea capaz de girar, se necesita el tercer motor y las piernas anteriormente construidas. Las siguientes imágenes muestran de manera ilustrada paso a paso la construcción del cuerpo del humanoide.

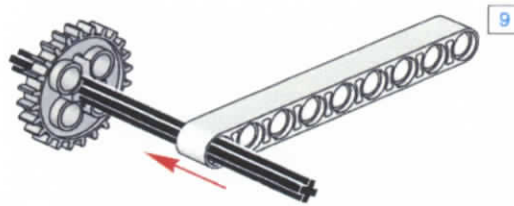


Figura 150. Ensamblaje Paso 2 (Cuerpo)

PASO 3

Se necesita (Figura 151):

- 1 anillo mediano.
- 1 viga angulada pequeña.

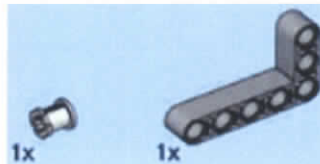


Figura 151. Elementos Paso 3 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 152):

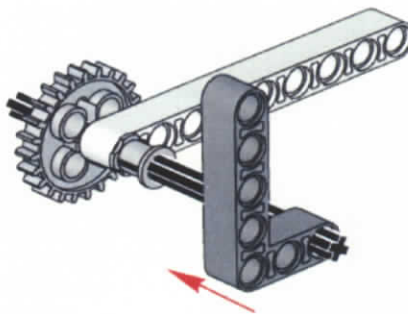


Figura 152. Ensamblaje Paso 3 (Cuerpo)

PASO 4

Se necesita (Figura 153):

- 1 anillo mediano.
- 1 conector negro grande.



Figura 153. Elementos Paso 4 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 154):

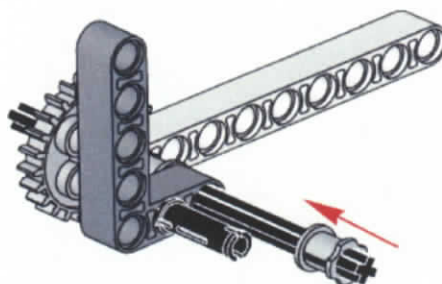


Figura 154. Ensamblaje Paso 4 (Cuerpo)

PASO 5

Se necesita (Figura 155):

- 1 eje negro mediano.
- 1 servomotor.

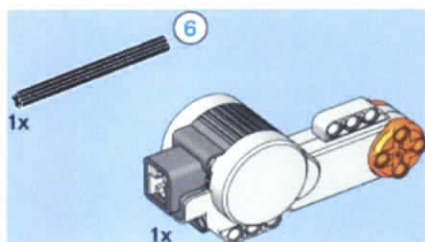


Figura 155. Elementos Paso 5 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 156):

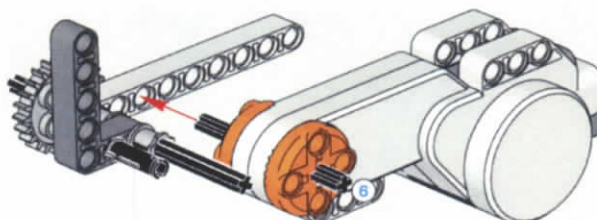


Figura 156. Ensamblaje Paso 5 (Cuerpo)

PASO 6

Se necesita (*Figura 157*):

- 2 conectores negros grandes.
- 2 bloque conector doble blanco.
- 1 viga mediana.

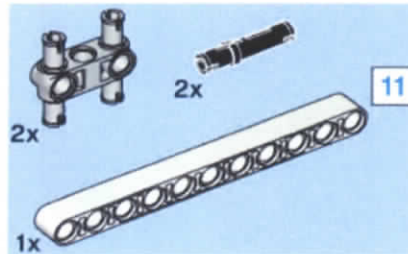


Figura 157. Elementos Paso 6 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (*Figura 158*):

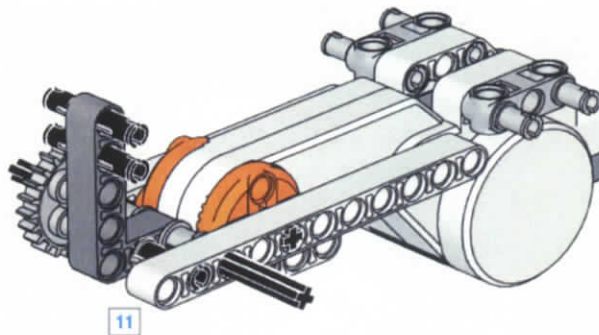


Figura 158. Ensamblaje Paso 6 (Cuerpo)

PASO 7

Se necesita (*Figura 159*):

- 1 anillo pequeño.
- 1 engranaje pequeño.
- 4 conectores negros pequeños.



Figura 159. Elementos Paso 7 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 160):

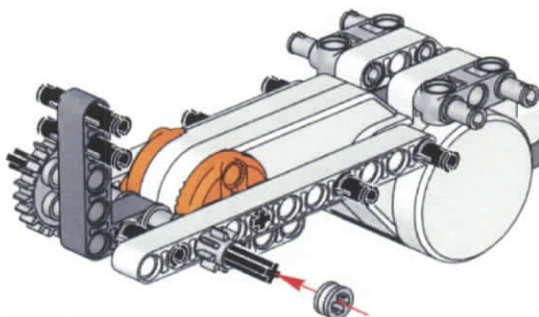


Figura 160. Ensamblaje Paso 7 (Cuerpo)

PASO 8

Se necesita (Figura 161):

- 2 ejes negros pequeños.
- 2 bloques conectores triples.
- 2 vigas anguladas pequeñas.

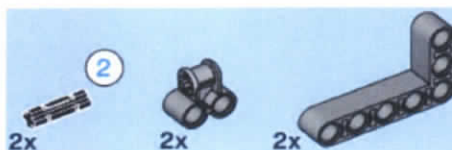


Figura 161. Elementos Paso 8 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 162):

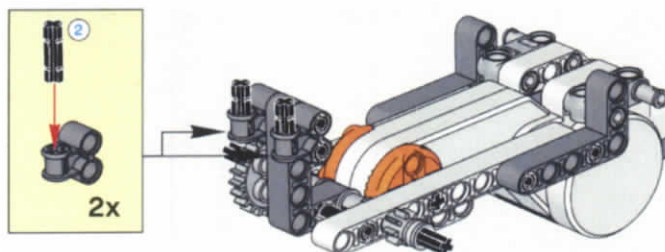


Figura 162. Ensamblaje Paso 8 (Cuerpo)

PASO 9

Se necesita (*Figura 163*):

- 1 engranaje mediano.
- 1 eje pequeño.

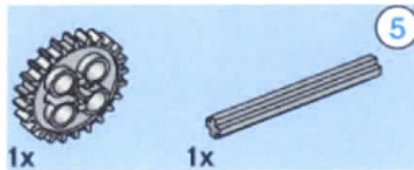


Figura 163. Elementos Paso 9 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (*Figura 164*):

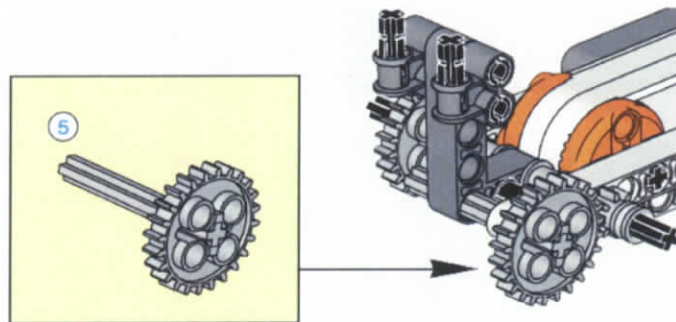


Figura 164. Ensamblaje Paso 9 (Cuerpo)

PASO 10

Se necesita (*Figura 165*):

- 1 eje extensor pequeño.
- 2 conectores negros pequeños.
- 1 conector negro grande.
- 1 engranaje mediano.

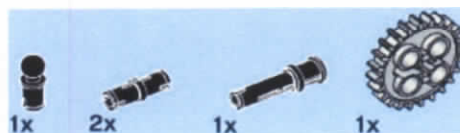


Figura 165. Elementos Paso 10 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (*Figura 166*):

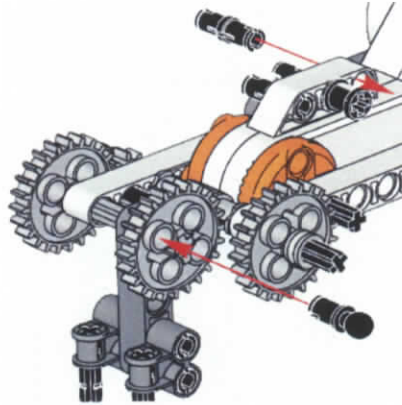


Figura 166. Ensamblaje Paso 10 (Cuerpo)

PASO 11

Se necesita (*Figura 167*):

- 2 conectores negros pequeños.
- 1 viga pequeña.

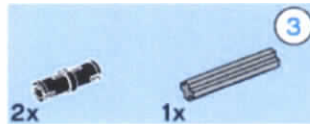


Figura 167. Elementos Paso 11 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (*Figura 168*):

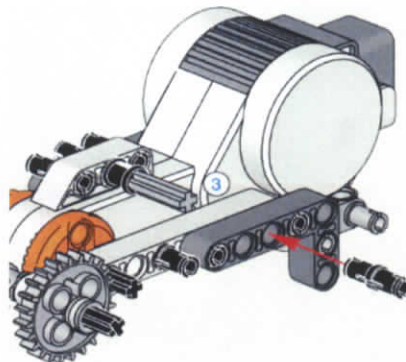


Figura 168. Ensamblaje Paso 11 (Cuerpo)

PASO 12

Se necesita (*Figura 169*):

- 1 engranaje pequeño.
- 2 conectores negros pequeños.
- 2 conectores negros grandes.



Figura 169. Elementos Paso 12 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (*Figura 170*):

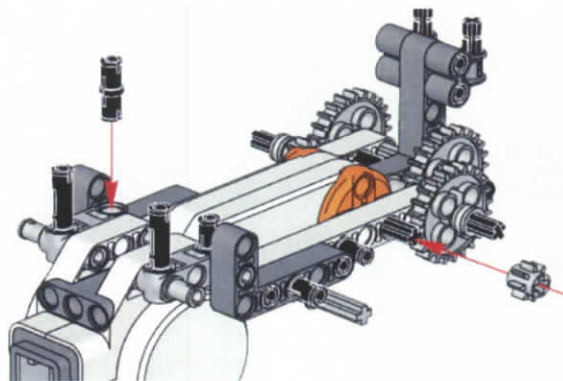


Figura 170. Ensamblaje Paso 12 (Cuerpo)

PASO 13

Se necesita (*Figura 171*):

- 4 conectores negros pequeños.
- 2 conectores negros grandes.
- 1 viga mediana.
- 1 viga grande.
- 1 engranaje mediano.

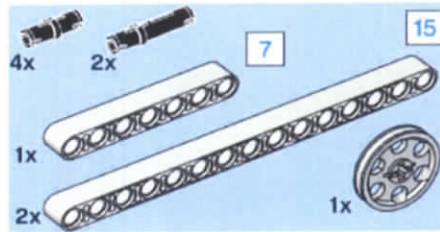


Figura 171. Elementos Paso 13 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 172):

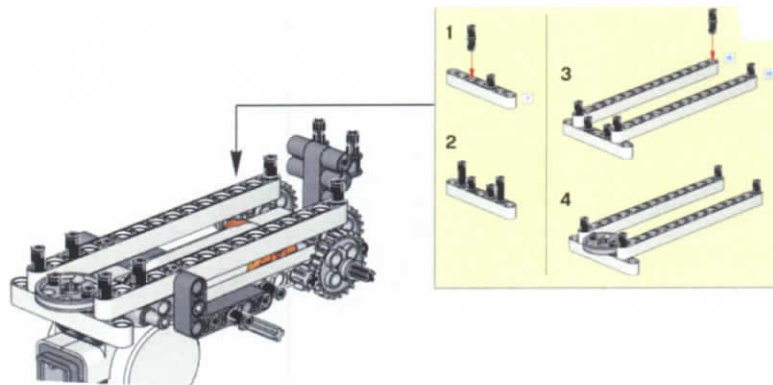


Figura 172. Ensamblaje Paso 13 (Cuerpo)

PASO 14

Se necesita (Figura 173):

- 2 conectores negros pequeños.
- 1 viga mediana.

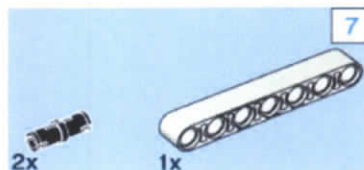


Figura 173. Elementos Paso 14 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 174):

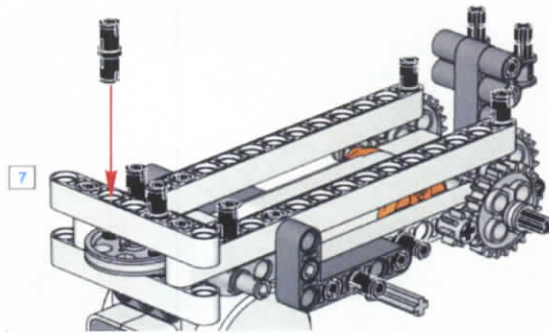


Figura 174. Ensamblaje Paso 14 (Cuerpo)

PASO 15

Se necesita (Figura 175):

- 1 viga pequeña.

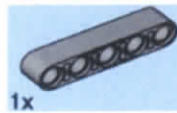


Figura 175. Elementos Paso 15 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 176):

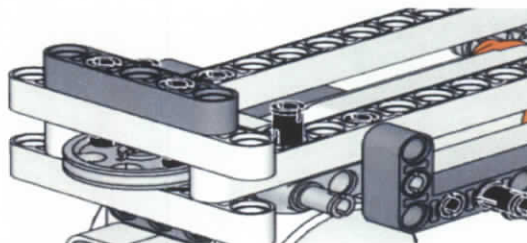


Figura 176. Ensamblaje Paso 15 (Cuerpo)

PASO 16

Se necesita (Figura 177):

- 1 bloque NXT.

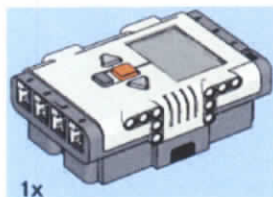


Figura 177. Elementos Paso 16 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (*Figura 178*):

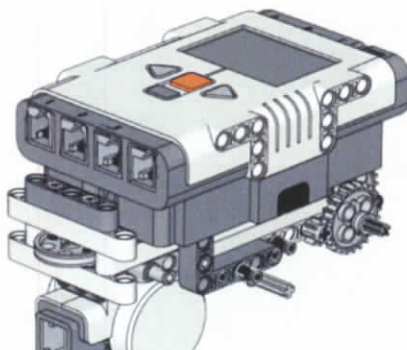


Figura 178. Ensamblaje Paso 16 (Cuerpo)

PASO 17

Se necesita (*Figura 179*):

- 2 conectores negros pequeños.
- 2 conectores negros grandes.



Figura 179. Elementos Paso 17 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (*Figura 180*):

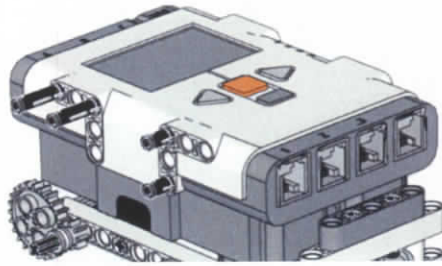


Figura 180. Ensamblaje Paso 17 (Cuerpo)

PASO 18

Se necesita (*Figura 181*):

- 3 conectores negros pequeños.
- 1 viga pequeña.
- 1 viga mediana.

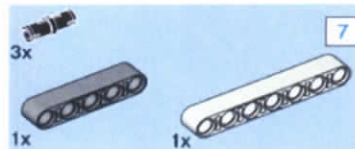


Figura 181. Elementos Paso 18 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (*Figura 182*):

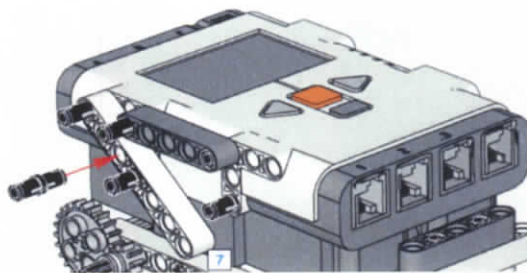


Figura 182. Ensamblaje Paso 18 (Cuerpo)

PASO 19

Se necesita (*Figura 183*):

- 2 conectores negros pequeños.
- 1 viga pequeña.



Figura 183. Elementos Paso 19 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 184):

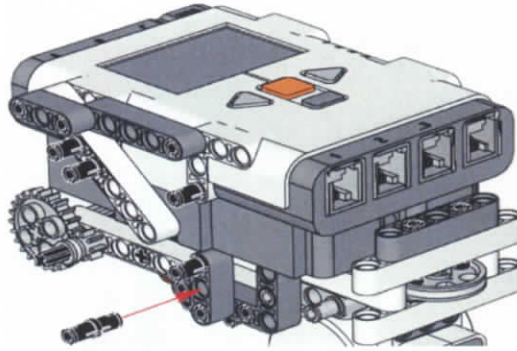


Figura 184. Ensamblaje Paso 19 (Cuerpo)

PASO 20

Se necesita (Figura 185):

- 1 bloque conector simple.
- 1 bloque conector triple.
- 1 viga angulada mediana.

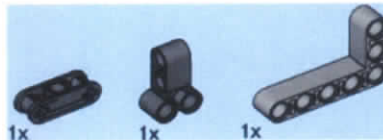


Figura 185. Elementos Paso 20 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 186):

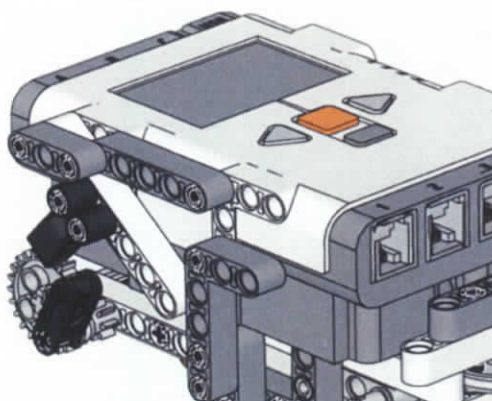


Figura 186. Ensamblaje Paso 20 (Cuerpo)

PASO 21

Se necesita (Figura 187):

- 1 eje extensor blanco pequeño.
- 1 eje extensor negro pequeño.

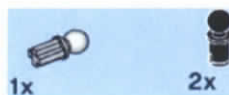


Figura 187. Elementos Paso 21 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 188):

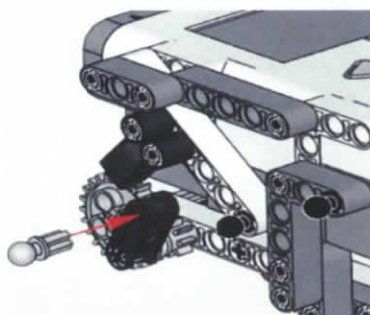


Figura 188. Ensamblaje Paso 21 (Cuerpo)

PASO 22

Se necesita (Figura 189):

- 1 eje extensor mediano.
- 1 eje extensor grande.



Figura 189. Elementos Paso 22 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (*Figura 190*):



Figura 190. Ensamblaje Paso 22 (Cuerpo)

PASO 23

Se necesita (*Figura 191*):

- 2 conectores negros pequeños.
- 2 conectores negros grandes.

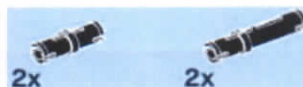


Figura 191. Elementos Paso 23 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (*Figura 192*):

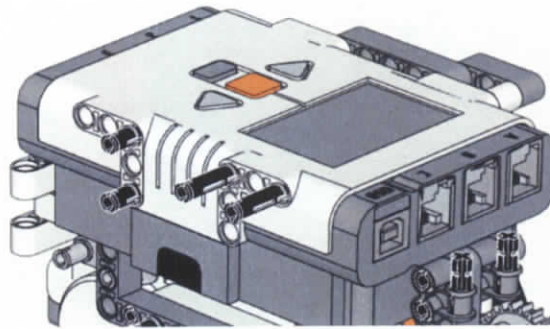


Figura 192. Ensamblaje Paso 23 (Cuerpo)

PASO 24

Se necesita (Figura 193):

- 2 conectores negros pequeños.
- 1 viga pequeña.
- 1 viga mediana.

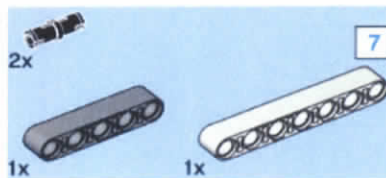


Figura 193. Elementos Paso 24 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 194):

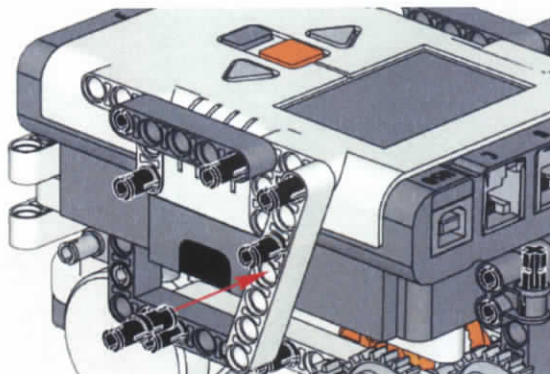


Figura 194. Ensamblaje Paso 24 (Cuerpo)

PASO 25

Se necesita (*Figura 195*):

- 2 conectores negros pequeños.
- 2 vigas pequeñas.



Figura 195. Elementos Paso 25 (Cuerpo)

Los anteriores elementos se deberán ensamblar de la siguiente manera (*Figura 196*):

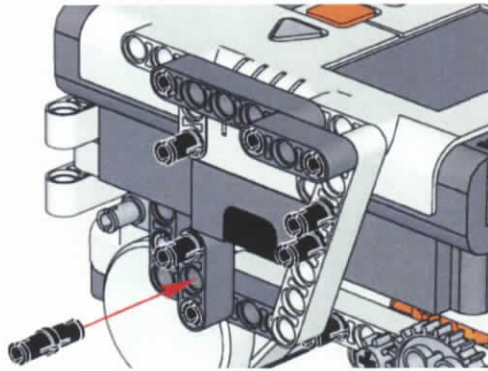


Figura 196. Ensamblaje Paso 25 (Cuerpo)

PASO 26

Se necesita (*Figura 197*):

- 1 bloque conector simple.
- 1 bloque conector triple.
- 1 viga angulada pequeña.

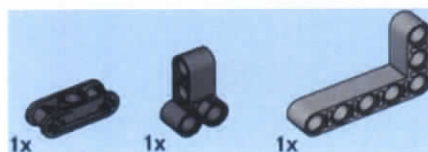


Figura 197. Elementos Paso 26 (Cuerpo)

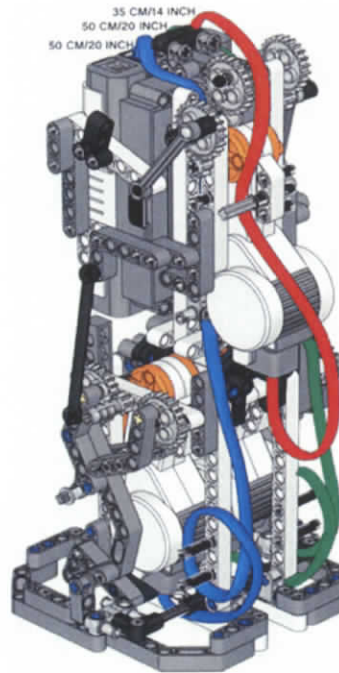


Figura 206. Ensamblaje Paso 30 (Cuerpo)

3.2.2.2 Programación.

Programación para que el robot vaya hacia delante gire y regrese al punto de inicio. Las siguientes imágenes muestran de manera ilustrada paso a paso la estructura y configuración de la programación.

1. Comenzar con el diagrama que quedo al final de la programación del caminar (*Figura 207*).

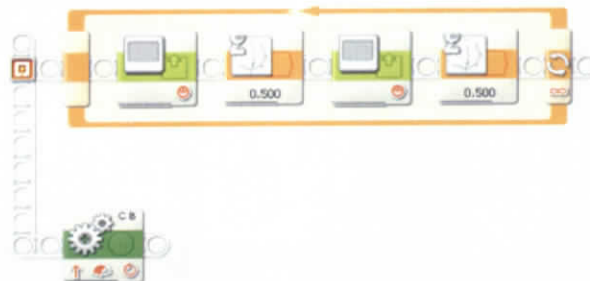


Figura 207. Paso 1 Programación (Cuerpo)

2. Colocar un segundo bloque de movimiento a lado del primero (*Figura 208*).

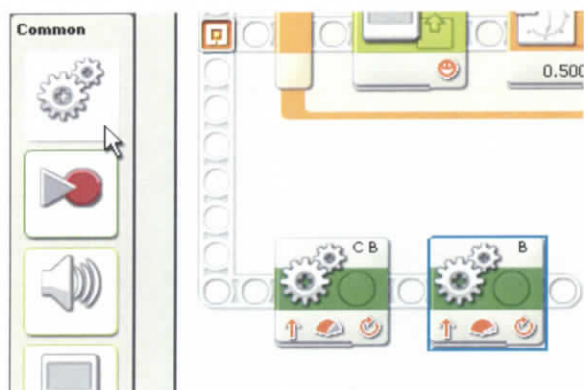


Figura 208. Paso 2 Programación (Cuerpo)

3. Configurar este bloque de movimiento para que controle el puerto B, trabaje al 100% y su duración sea de 30 rotaciones (Figura 209).

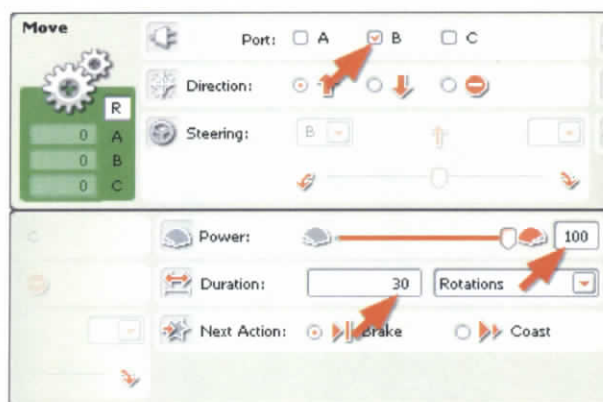


Figura 209. Paso 3 Programación (Cuerpo)

4. Colocar un tercer bloque de movimiento a lado del segundo (Figura 210).



Figura 210. Paso 4 Programación (Cuerpo)

5. Configurar este bloque de movimiento para que su duración sea de 25.5 rotaciones (*Figura 211*).

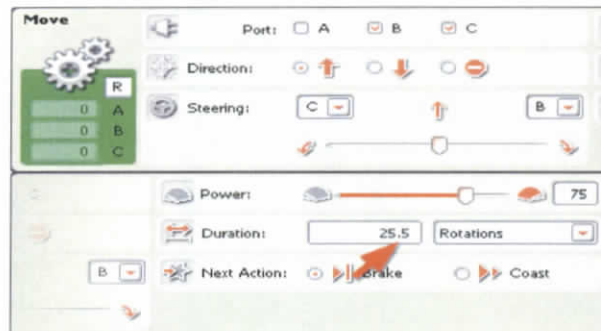


Figura 211. Paso 5 Programación (Cuerpo)

3.2.3 Vista.

Construcción y programación de los ojos del humanoide.

3.2.3.1 Construcción.

Construcción de un robot que pueda ver, para lo cual utilizaremos el sensor de ultrasonido unido a la estructura cuerpo y piernas anteriormente construida. Las siguientes imágenes muestran de manera ilustrada paso a paso la construcción de los ojos del humanoide.

Para toda la construcción se necesita (*Figura 212*):

- 3 conectores negro grandes.
- 1 eje extensor negro pequeño.
- 2 ejes pequeño.
- 1 conector azul.
- 4 bloques conectores dobles.
- 1 bloque conector simple.
- 1 sensor ultrasonido.

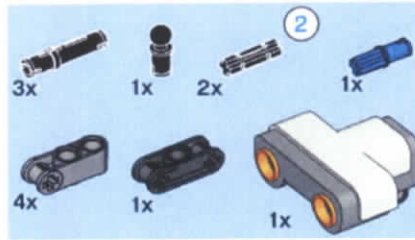


Figura 212. Elementos (Vista)

PASO 1

Para este paso los siguientes elementos se deberán ensamblar de la siguiente manera (Figura 213):



Figura 213. Ensamblaje Paso 1 (Vista)

PASO 2

Para este paso los siguientes elementos se deberán ensamblar de la siguiente manera (Figura 214):



Figura 214. Ensamblaje Paso 2 (Vista)

PASO 3

Para este paso los siguientes elementos se deberán ensamblar de la siguiente manera (Figura 215):

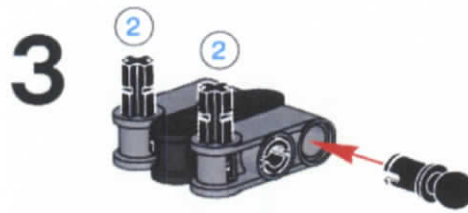


Figura 215. Ensamblaje Paso 3 (Vista)

PASO 4

Para este paso los siguientes elementos se deberán ensamblar de la siguiente manera (Figura 216):

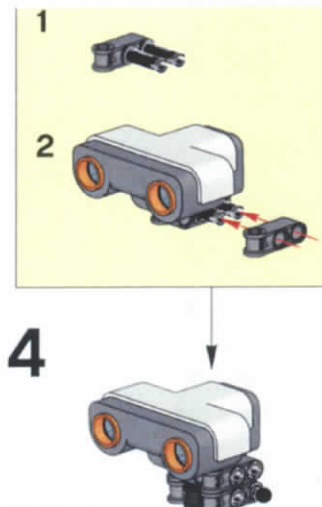


Figura 216. Ensamblaje Paso 4 (Vista)

PASO 5

Para este paso los siguientes elementos se deberán ensamblar de la siguiente manera (Figura 217):

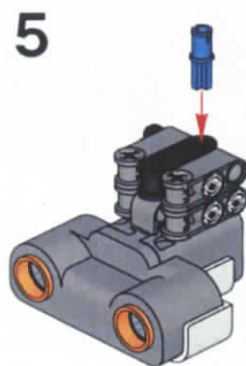


Figura 217. Ensamblaje Paso 5 (Vista)

PASO 6

Se necesita (Figura 218):

- 1 eje extensor mediano.
- 1 cable de conexión de 35cm.



Figura 218. Elementos Paso 6 (Vista)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 219):

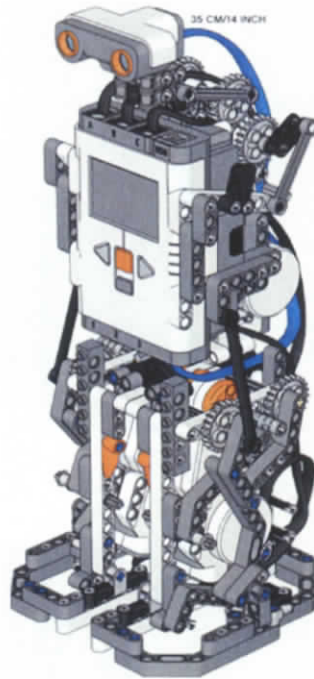


Figura 219. Ensamblaje Paso 6 (Vista)

3.2.3.2 Programación.

Programación para que el robot diga “HELLO” cuando alguien se pare frente a él y gire y diga “GOODBYE” y se aleje. Las siguientes imágenes muestran de manera ilustrada paso a paso la estructura y configuración de la programación.

1. Comenzar con el diagrama que quedó al final de la programación del cuerpo (Figura 220).



Figura 220. Paso 1 Programación (Vista)

2. Seleccionar los 2 bloques de movimiento (Figura 221).

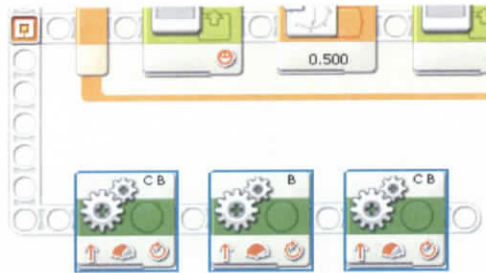


Figura 221. Paso 2 Programación (Vista)

3. Borrar los 3 bloques de movimiento seleccionados (Figura 222).



Figura 222. Paso 3 Programación (Vista)

4. Colocar un bloque de repetición en el lugar de los bloques de movimiento eliminados (Figura 223).

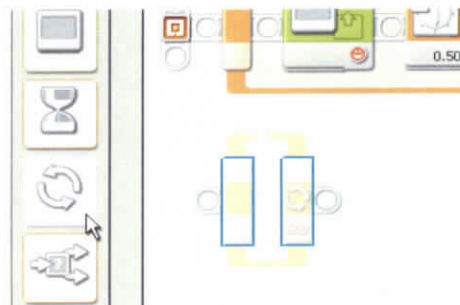


Figura 223. Paso 4 Programación (Vista)

5. Configurar el bloque de repetición para que se mantenga de manera infinita "forever" (Figura 224).



Figura 224. Paso 5 Programación (Vista)

6. Agrandar el punto de inicio hasta el bloque de repetición (Figura 225).



Figura 225. Paso 6 Programación (Vista)

7. Colocar un bloque de espera controlado por el sensor de ultrasonido dentro del bloque de repetición del paso anterior (Figura 226).

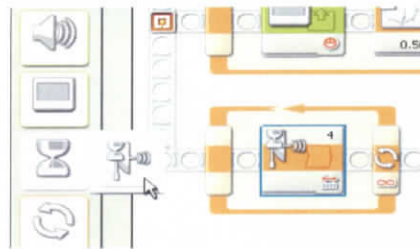


Figura 226. Paso 7 Programación (Vista)

8. Configurar el bloque de espera para que la distancia sea menor que 15 pulgadas (Figura 227).

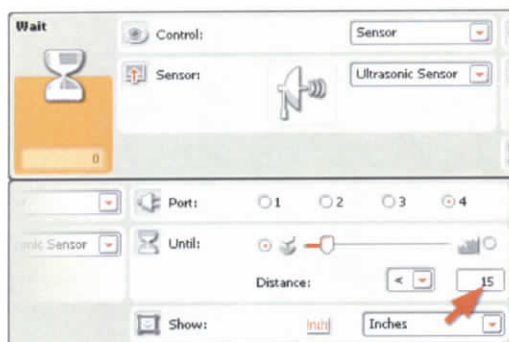


Figura 227. Paso 8 Programación (Vista)

9. Colocar un bloque de sonido después del bloque de espera (Figura 228).

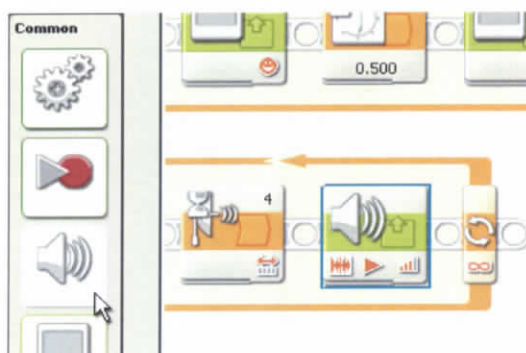


Figura 228. Paso 9 Programación (Vista)

10. Configurar el bloque de sonido para que utilice el archivo "Hello" (Figura 229).

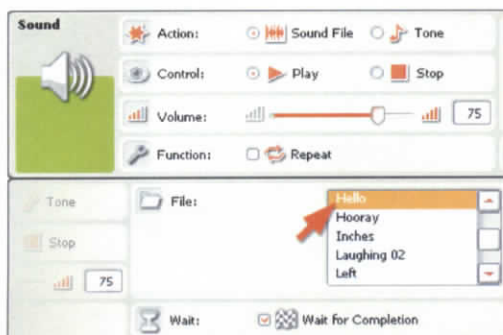


Figura 229. Paso 10 Programación (Vista)

11. Colocar un segundo bloque de espera controlado por el sensor de ultrasonido a lado del bloque de sonido (Figura 230).

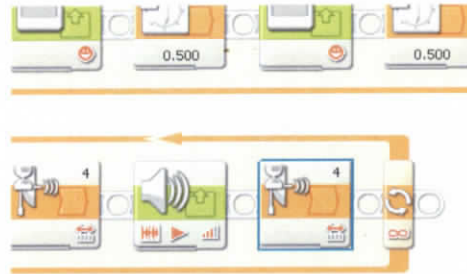


Figura 230. Paso 11 Programación (Vista)

12. Configurar el segundo bloque de espera para que la distancia sea mayor que 15 pulgadas (Figura 231).

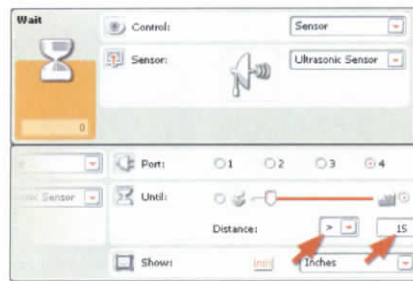


Figura 231. Paso 12 Programación (Vista)

13. Colocar un segundo bloque de sonido a lado del segundo bloque de espera (Figura 232).

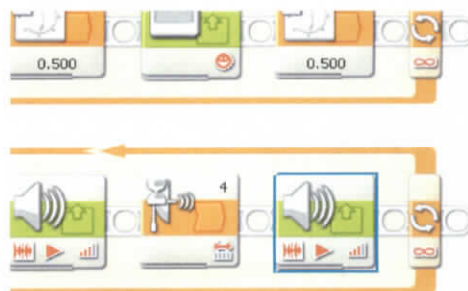


Figura 232. Paso 13 Programación (Vista)

14. Configurar el segundo bloque de sonido para que utilice el archivo "Goodbye" (Figura 233).

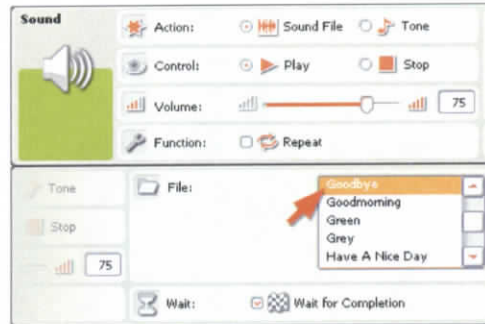


Figura 233. Paso 14 Programación (Vista)

15. Colocar un bloque de movimiento a lado del segundo bloque de sonido (Figura 234).

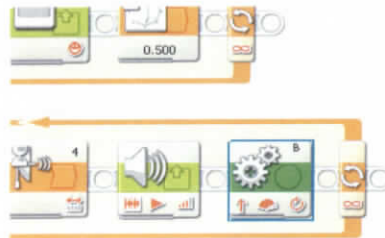


Figura 234. Paso 15 Programación (Vista)

16. Configurar el bloque de movimiento para que controle el puerto B, utilice el 100% de potencia y su duración sea de 8 rotaciones (Figura 235).

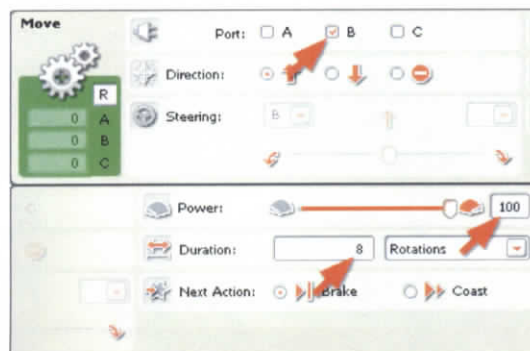


Figura 235. Paso 16 Programación (Vista)

3.2.4 Oído y Tacto.

Construcción y programación del oído y tacto del humanoide.

3.2.4.1 Construcción.

Construcción de un robot que escucha y reacciona a los sonidos, y siente cuando lo tocan; para lo cual se construirá 2 brazos uno con el sensor de sonido y el otro con el sensor de tacto. Las siguientes imágenes muestran de manera ilustrada paso a paso la construcción del oído y tacto del humanoide.

PRIMER BRAZO

Para toda la construcción se necesita (*Figura 236*):

- 5 conectores negros pequeños.
- 1 viga tomate.
- 1 sensor de sonido.
- 4 conectores negros grandes.
- 2 vigas pequeñas.
- 2 ejes pequeños.
- 1 bloque conector doble.
- 1 eje mediano.
- 1 viga mediana.
- 1 viga grande.
- 6 viga anguladas pequeñas.

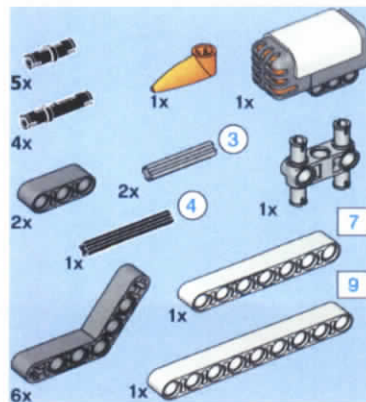


Figura 236. Elementos Primer Brazo (Oído y Tacto)

PASO 1

Para este paso los siguientes elementos se deberán ensamblar de la siguiente manera (Figura 237):



Figura 237. Ensamblaje Paso 1 (Oído y Tacto)

PASO 2

Para este paso los siguientes elementos se deberán ensamblar de la siguiente manera (Figura 238):

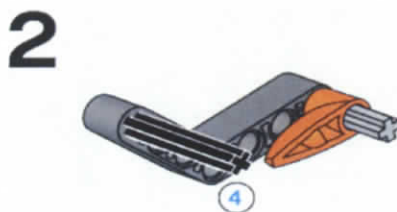


Figura 238. Ensamblaje Paso 2 (Oído y Tacto)

PASO 3

Para este paso los siguientes elementos se deberán ensamblar de la siguiente manera (Figura 239):

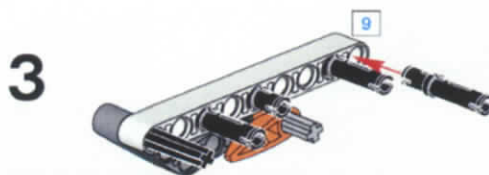


Figura 239. Ensamblaje Paso 3 (Oído y Tacto)

PASO 4

Para este paso los siguientes elementos se deberán ensamblar de la siguiente manera (Figura 240):

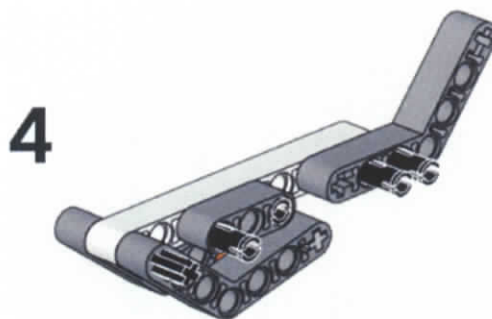


Figura 240. Ensamblaje Paso 4 (Oído y Tacto)

PASO 5

Para este paso los siguientes elementos se deberán ensamblar de la siguiente manera (Figura 241):

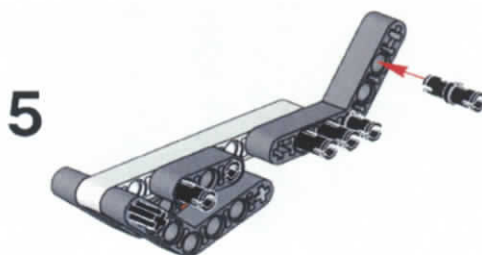


Figura 241. Ensamblaje Paso 5 (Oído y Tacto)

PASO 6

Para este paso los siguientes elementos se deberán ensamblar de la siguiente manera (Figura 242):

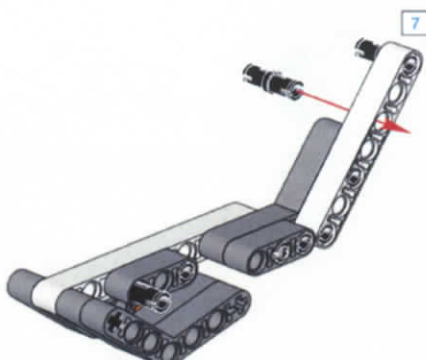


Figura 242. Ensamblaje Paso 6 (Oído y Tacto)

PASO 7

Para este paso los siguientes elementos se deberán ensamblar de la siguiente manera (Figura 243):

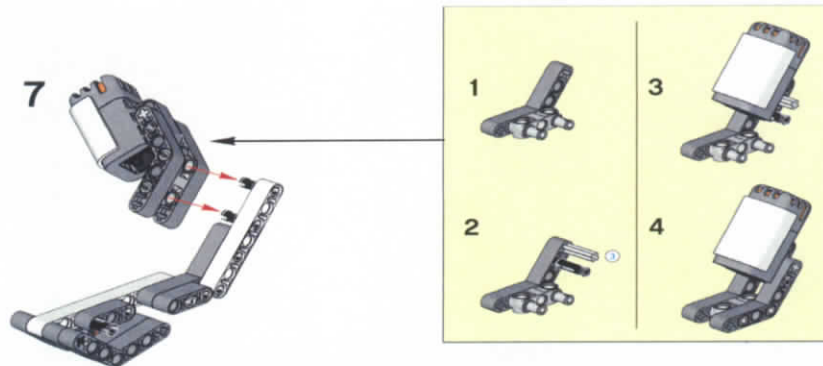


Figura 243. Ensamblaje Paso 7 (Oído y Tacto)

PASO 8

Se necesita (Figura 244):

- 1 cable de conexión de 35cm.

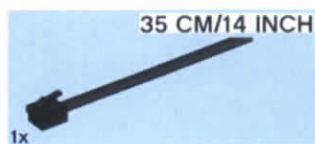


Figura 244. Elementos Paso 8 (Oído y Tacto)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 245):

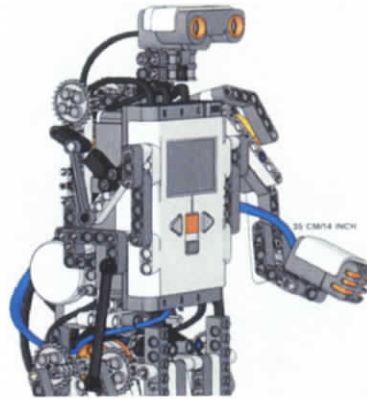


Figura 245. Ensamblaje Paso 8 (Oído y Tacto)

SEGUNDO BRAZO

Para toda la construcción se necesita (Figura 246):

- 5 conectores negros pequeños.
- 1 viga tomate.
- 1 sensor de sonido.
- 4 conectores negros grandes.
- 2 vigas pequeñas.
- 2 ejes pequeños.
- 1 bloque conector doble.
- 1 eje mediano.
- 1 viga mediana.
- 1 viga grande.
- 6 viga anguladas pequeñas.

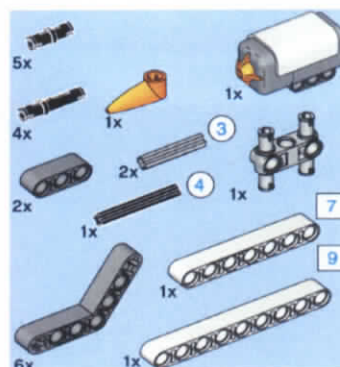


Figura 246. Elementos Segundo Brazo (Oído y Tacto)

PASO 9

Como se observa se utilizan los mismos que para el primer brazo por esa razón realizaremos los mismos pasos desde el 1 al 7 pero al lado opuesto y en vez del sensor de sonido utilizamos el sensor de tacto, al final deberá quedar el segundo brazo (*Figura 247*):



Figura 247. Segundo Brazo

PASO 10

Se necesita (*Figura 248*):

- 1 cable de conexión de 35cm.

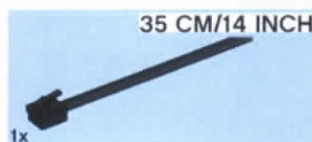


Figura 248. Elementos Paso 10 (Oído y Tacto)

Los anteriores elementos se deberán ensamblar de la siguiente manera (*Figura 249*):

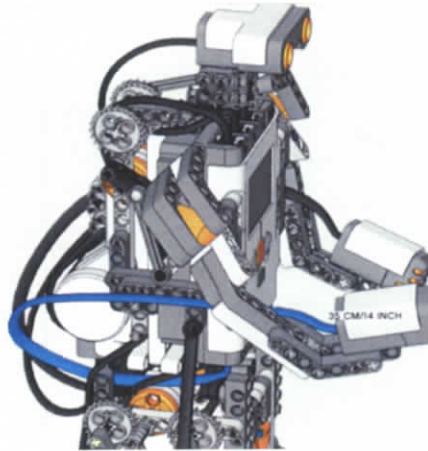


Figura 249. Ensamblaje Paso 10 (Oído y Tacto)

3.2.4.2 Programación.

Programación para que el robot diga “Play music” cuando se presiona el sensor de tacto y comience a bailar cuando escuche música o el aplaudir de las manos, y pare cuando se presione de nuevo el sensor de tacto. Las siguientes imágenes muestran de manera ilustrada paso a paso la estructura y configuración de la programación.

1. Comenzar con el diagrama que quedó al final de la programación de la vista (Figura 250).

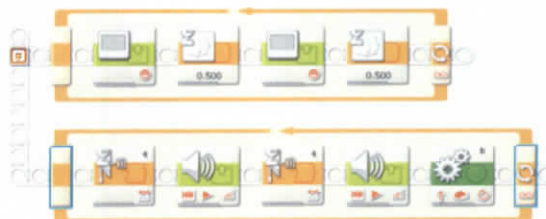


Figura 250. Paso 1 Programación (Oído y Tacto)

2. Borrar el bloque de repetición inferior con todo su contenido (Figura 251).



Figura 251. Paso 2 Programación (Oído y Tacto)

3. Seleccionar el bloque de repetición restante (Figura 252).

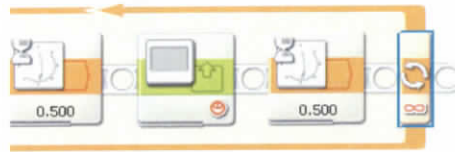


Figura 252. Paso 3 Programación (Oído y Tacto)

4. Configurar el bloque de repetición seleccionado para que sea controlado por el sensor de tacto (Figura 253).

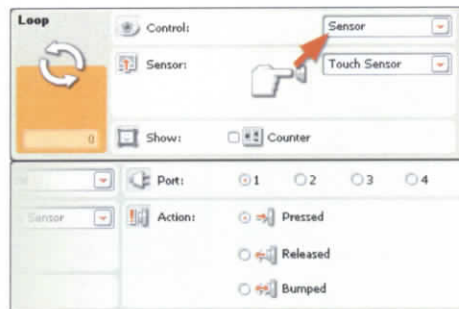


Figura 253. Paso 4 Programación (Oído y Tacto)

5. Colocar un bloque de espera controlado por el sensor de tacto en el lugar en donde estaba el bloque de repetición eliminado y no mover su configuración (Figura 254).

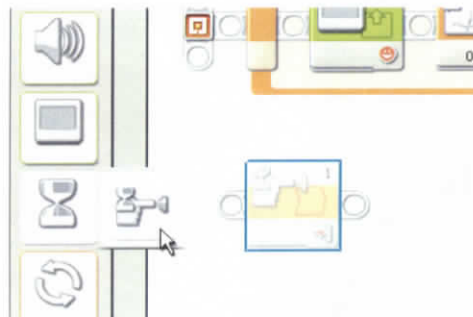


Figura 254. Paso 5 Programación (Oído y Tacto)

6. Agrandar el punto de inicio hasta el bloque de espera (Figura 255).



Figura 255. Paso 6 Programación (Oído y Tacto)

7. Colocar un bloque de sonido a lado del bloque de espera (Figura 256).

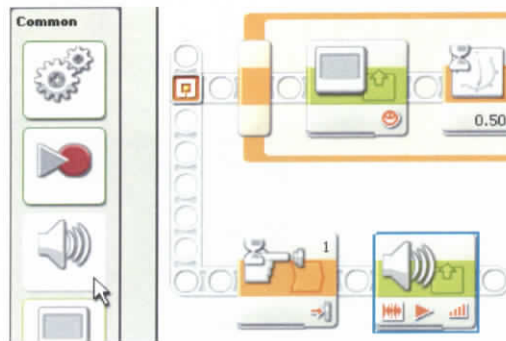


Figura 256. Paso 7 Programación (Oído y Tacto)

8. Configurar el bloque de sonido para que tenga el máximo volumen y trabaje con el archivo "Play" (Figura 257).

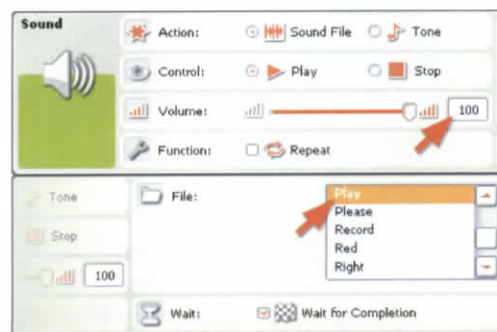


Figura 257. Paso 8 Programación (Oído y Tacto)

9. Colocar un segundo bloque de sonido a lado del primero (Figura 258).

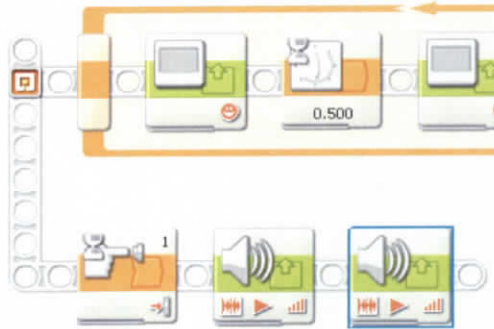


Figura 258. Paso 9 Programación (Oído y Tacto)

10. Configurar el segundo bloque de sonido para que tenga el máximo volumen y trabaje con el archivo “Music” (Figura 259).

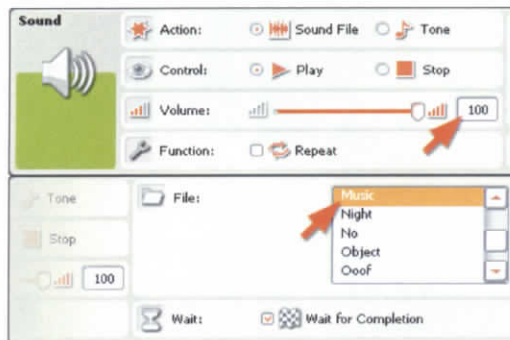


Figura 259. Paso 10 Programación (Oído y Tacto)

11. Colocar un bloque de movimiento a lado del segundo bloque de sonido (Figura 260).

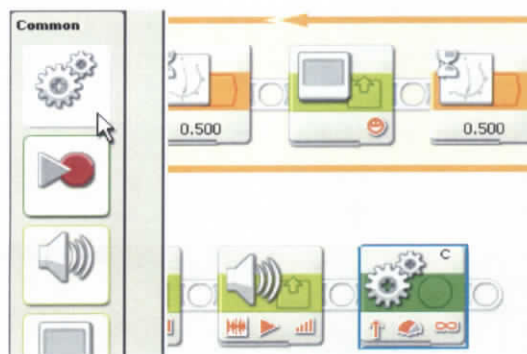


Figura 260. Paso 11 Programación (Oído y Tacto)

12. Configurar el bloque de movimiento para que trabaje con el puerto C y tenga una duración ilimitada (Figura 261).

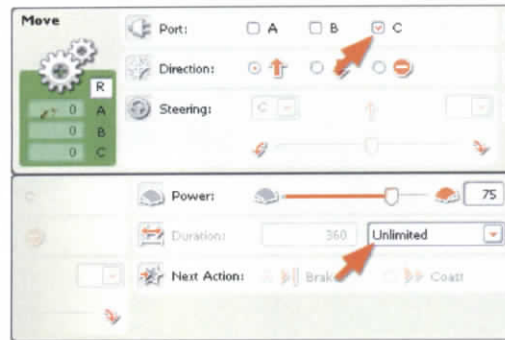


Figura 261. Paso 12 Programación (Oído y Tacto)

13. Colocar un bloque de repetición controlado por el sensor de tacto a lado del bloque de movimiento (Figura 262).

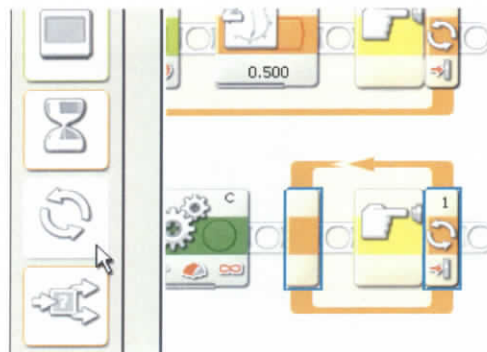


Figura 262. Paso 13 Programación (Oído y Tacto)

14. Colocar un bloque switch controlado por el sensor de sonido dentro del bloque de repetición (Figura 263).

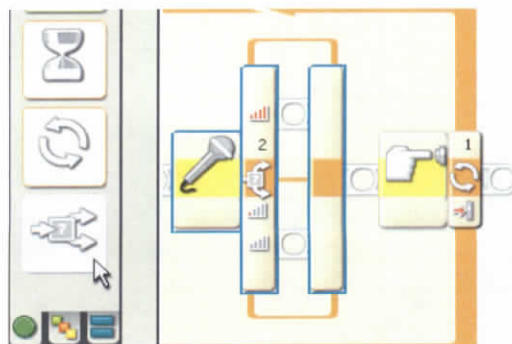


Figura 263. Paso 14 Programación (Oído y Tacto)

15. Configurar el bloque de sonido para un sonido mayor a 25 (Figura 264).

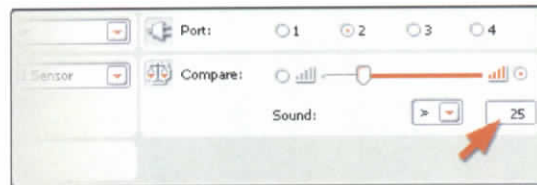


Figura 264. Paso 15 Programación (Oído y Tacto)

16. Colocar en el lado verdadero del bloque switch un bloque de movimiento (Figura 265).

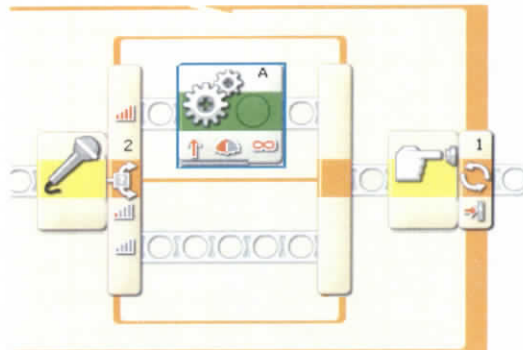


Figura 265. Paso 16 Programación (Oído y Tacto)

17. Configurar el bloque de movimiento para que trabaje con el puerto A, su potencia sea del 50% y su duración sea ilimitada (Figura 266).

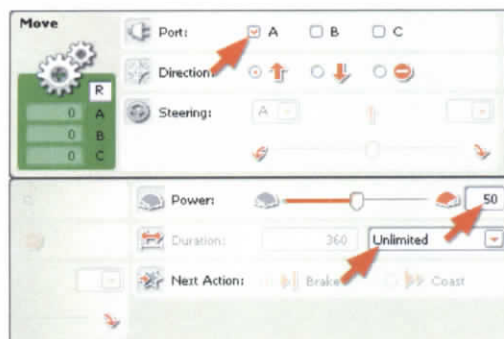


Figura 266. Paso 17 Programación (Oído y Tacto)

18. Colocar en el lado falso del bloque switch un segundo bloque de movimiento (Figura 267).

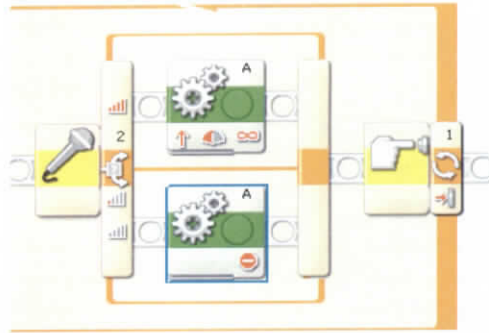


Figura 267. Paso 18 Programación (Oído y Tacto)

19. Configurar el segundo bloque de movimiento para que trabaje con el puerto A y para que se detenga (Figura 268).

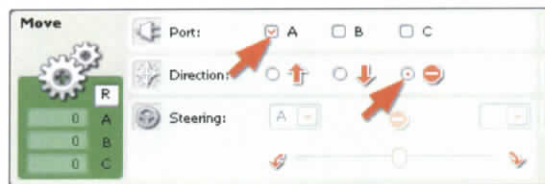


Figura 268. Paso 19 Programación (Oído y Tacto)

20. Colocar un tercer bloque de movimiento fuera del bloque de repetición (Figura 269).

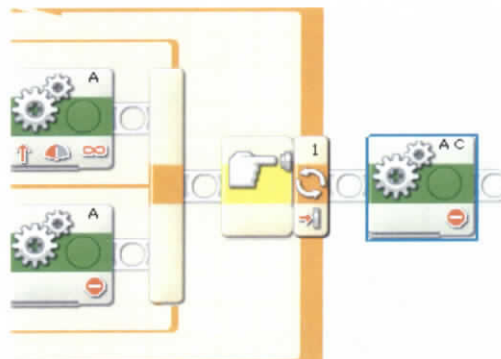


Figura 269. Paso 20 Programación (Oído y Tacto)

21. Configurar el tercer bloque de movimiento para que trabaje con el puerto A y C, y para que se detenga (Figura 270).

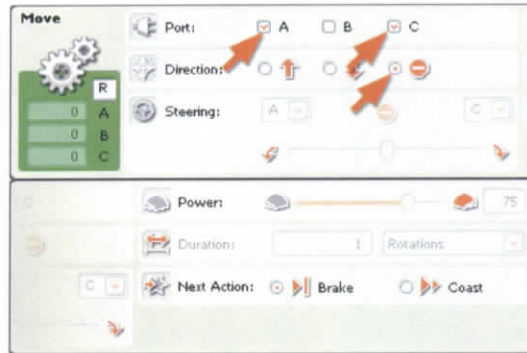


Figura 270. Paso 21 Programación (Oído y Tacto)

3.2.5 Luz.

Construcción y programación de la capacidad de detectar luz del humanoide.

3.2.5.1 Construcción.

Construcción de un robot que puede reaccionar a la luz, para lo cual se usará el sensor de luz. Las siguientes imágenes muestran de manera ilustrada paso a paso la construcción del oído y tacto del humanoide.

PASO 1

Se necesita (Figura 271):

- 1 sensor de luz.



Figura 271. Elementos Paso 1 (Luz)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 272):

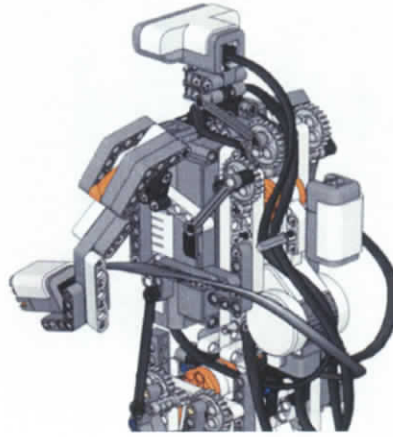


Figura 272. Ensamblaje Paso 1 (Luz)

PASO 2

Se necesita (Figura 273):

- 1 cable de conexión de 20cm.

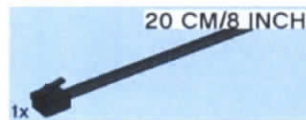


Figura 273. Elementos Paso 2 (Luz)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 274):



Figura 274. Ensamblaje Paso 2 (Luz)

PASO 3

Se necesita (*Figura 275*):

- 2 conectores negros pequeños.
- 1 viga pequeña.
- 2 vigas anguladas pequeñas.



Figura 275. Elementos Paso 3 (Luz)

Los anteriores elementos se deberán ensamblar de la siguiente manera (*Figura 276*):

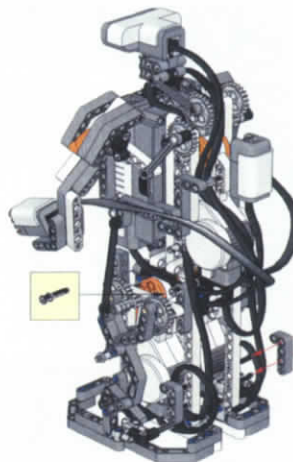


Figura 276. Ensamblaje Paso 3 (Luz)

3.2.5.2 Programación.

Programación para que el robot diga “Goodnight” y aparezca “Zzz” en la pantalla del bloque NXT cuando se apague la luz y cuando se prenda la Luz diga “Morning”. Las siguientes imágenes muestran de manera ilustrada paso a paso la estructura y configuración de la programación.

1. Colocar un bloque de repetición en el punto de inicio (*Figura 277*).

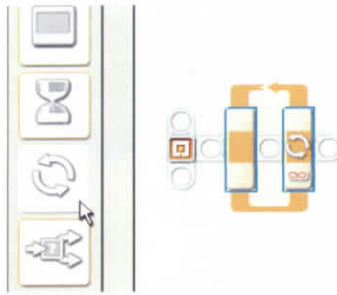


Figura 277. Paso 1 Programación (Luz)

2. Configurar el bloque de repetición para que dure de manera infinita “forever” (Figura 278).



Figura 278. Paso 2 Programación (Luz)

3. Colocar dentro del bloque de repetición un bloque switch controlado por el sensor de luz (Figura 279).

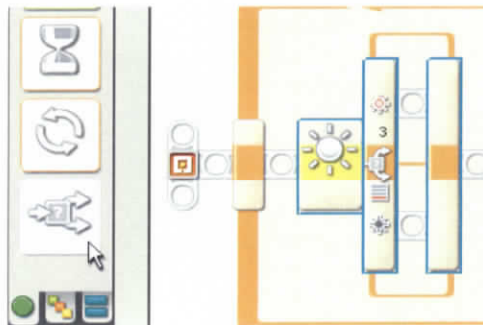


Figura 279. Paso 3 Programación (Luz)

4. Configurar para que funcione con una intensidad de luz mayor a 20 y activar la opción “Generated Light” (Figura 280).

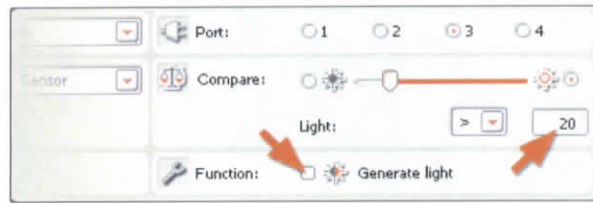


Figura 280. Paso 4 Programación (Luz)

5. Colocar un bloque de sonido en el lado verdadero del bloque switch (Figura 281).

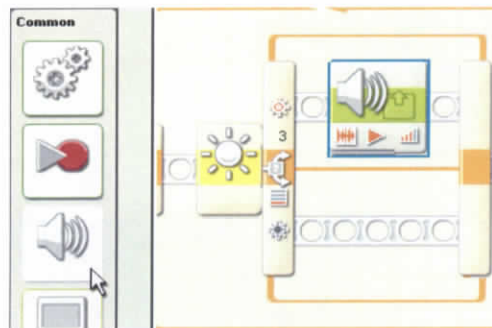


Figura 281. Paso 5 Programación (Luz)

6. Configurar el de sonido para que utilice el archivo "Goodmorning" (Figura 282).

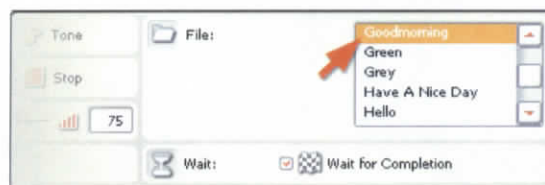


Figura 282. Paso 6 Programación (Luz)

7. Colocar un bloque de pantalla a lado del bloque de sonido (Figura 283).

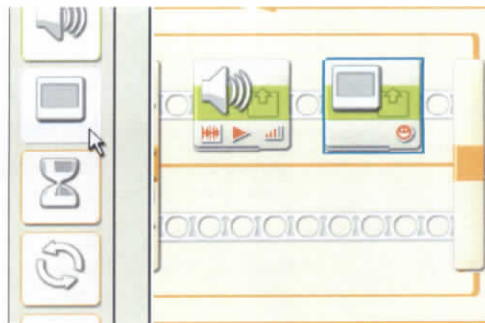


Figura 283. Paso 7 Programación (Luz)

8. Configurar el bloque de pantalla para que utilice el archivo “Smile 01” en una determinada posición de la pantalla del bloque NXT (Figura 284).

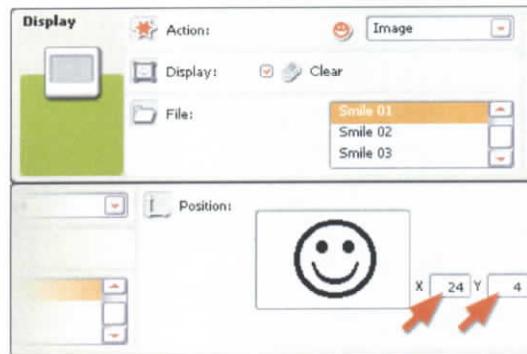


Figura 284. Paso 8 Programación (Luz)

9. Colocar un bloque de espera controlado por el sensor de luz (Figura 285).

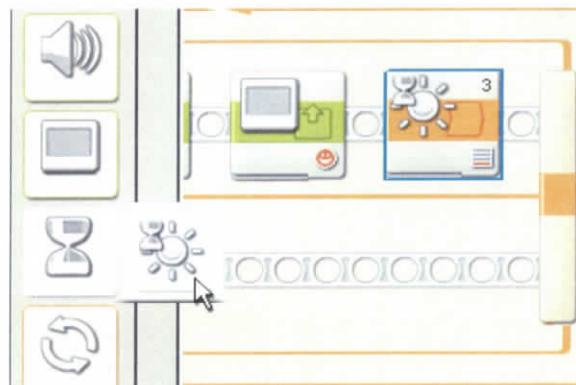


Figura 285. Paso 9 Programación (Luz)

10. Configurar el bloque de espera para que trabaje con un valor menor que 20 y activar la opción “Generated Light” (Figura 286).

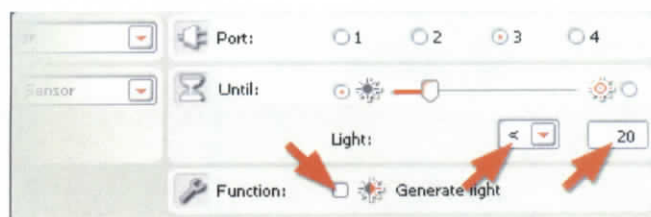


Figura 286. Paso 10 Programación (Luz)

11. Colocar un segundo bloque de sonido en lado falso del bloque switch (Figura 287).

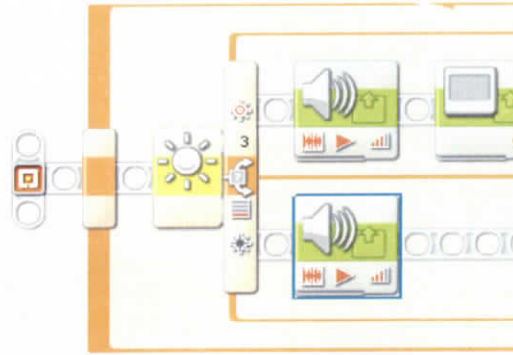


Figura 288. Paso 11 Programación (Luz)

12. Configurar el segundo bloque de sonido para que utilice el archivo “Good” (Figura 289).

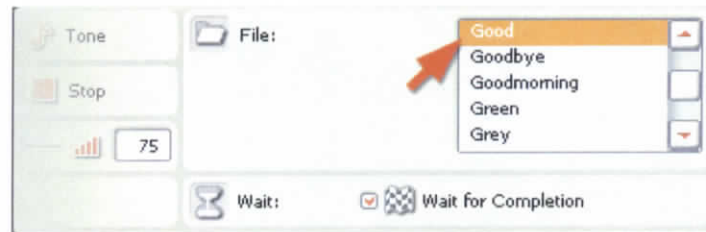


Figura 289. Paso 12 Programación (Luz)

13. Colocar un tercer bloque de sonido a lado del segundo (Figura 290).

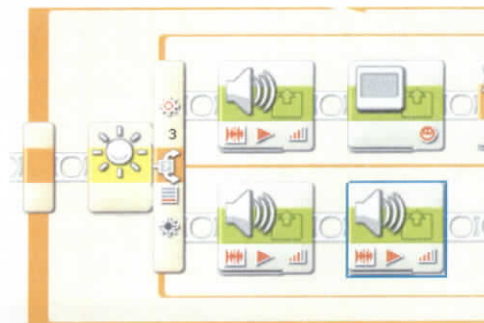


Figura 290. Paso 13 Programación (Luz)

14. Configurar el tercer bloque de sonido para que utilice el archivo “Night” (Figura 291).



Figura 291. Paso 14 Programación (Luz)

15. Colocar un segundo bloque de pantalla a lado del tercer bloque de sonido (Figura 292).

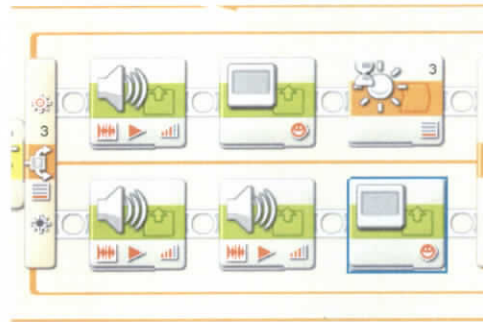


Figura 292. Paso 15 Programación (Luz)

16. Configurar el segundo bloque de pantalla para que utilice el archivo “Zzzz” en una posición determinada dentro de la pantalla del bloque NXT (Figura 293).

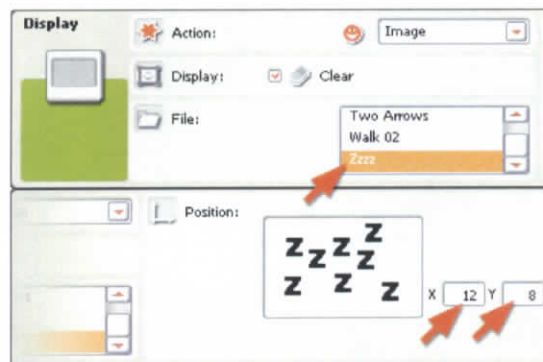


Figura 293. Paso 16 Programación (Luz)

17. Colocar un segundo bloque de repetición controlado por el sensor de luz a lado del segundo bloque de pantalla (Figura 294).

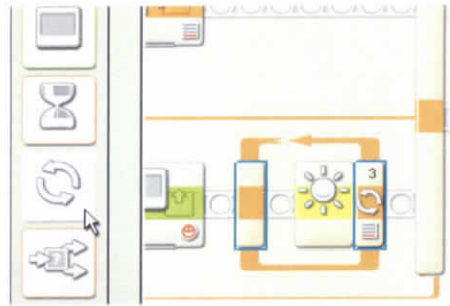


Figura 294. Paso 17 Programación (Luz)

18. Configurar el segundo bloque de repetición para que trabaje con un valor mayor que 20 y activar la opción “Generated Light” (Figura 295).

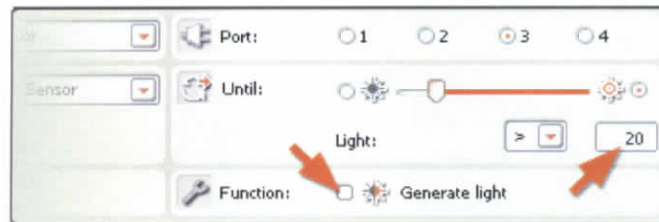


Figura 295. Paso 18 Programación (Luz)

19. Colocar un cuarto bloque de sonido dentro del segundo bloque de repetición (Figura 296).

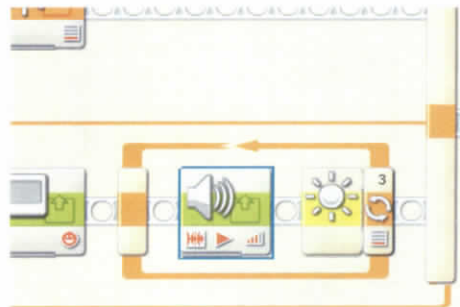


Figura 296. Paso 19 Programación (Luz)

20. Configurar el cuarto bloque de sonido para que utilice el archivo “Snore” (Figura 297).

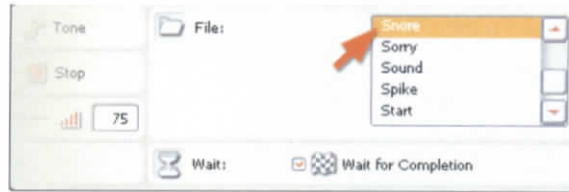


Figura 297. Paso 20 Programación (Luz)

3.2.6 Color.

Construcción y programación de la capacidad de detectar color del humanoide.

3.2.6.1 Construcción.

Construcción de un robot que seguirá un camino dividido en secciones de colores en que en cada color realizara una acción distinta. Las siguientes imágenes muestran de manera ilustrada paso a paso la construcción del color del humanoide.

PASO 1

Se necesita (Figura 298):

- 1 sensor de color.
- 1 bloque de conexión doble pequeño.



Figura 298. Elementos Paso 1 (Color)

Los anteriores elementos se deberán ensamblar de la siguiente manera (Figura 299):



Figura 299. Ensamblaje Paso 1 (Color)

3.2.6.2 Programación.

Programación para que el robot siga un camino de colores reproduciendo un sonido y realizando un acción distinta en cada color; que se detenga si hay algo en su camino y que funcione sólo cuando haya luz. Las siguientes imágenes muestran de manera ilustrada paso a paso la estructura y configuración de la programación.

1. Insertar un bloque de repetición en el punto de inicio (Figura 300).



Figura 300. Paso 1 Programación (Color)

2. Configurar para que el bloque de repetición se mantenga funcionando ilimitadamente (Figura 301).



Figura 301. Paso 2 Programación (Color)

3. Dentro del bloque de repetición colocar un bloque switch controlado por el sensor de luz (Figura 302).

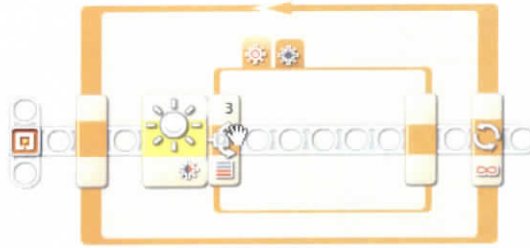


Figura 302. Paso 3 Programación (Color)

4. Configurar el bloque switch para que funcione con el puerto 2, que la condición verdadera se cumpla en valores menores que 20 niveles de luz y desactivar la opción “Generated Light” (Figura 303).

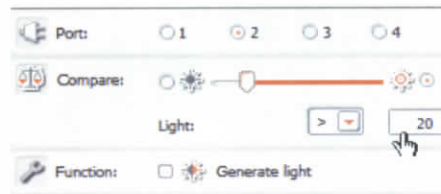


Figura 303. Paso 4 Programación (Color)

5. En el lado verdadero del bloque switch colocar un segundo bloque switch controlado por el sensor de ultrasonido (Figura 304).

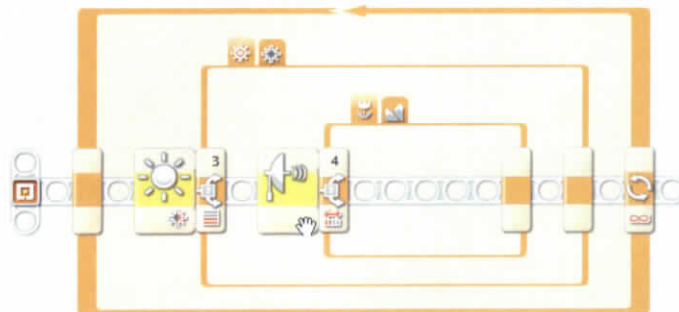


Figura 304. Paso 5 Programación (Color)

6. Configurar este bloque switch para que la condición verdadera se cumpla con valores menores que 10 centímetros de distancia (Figura 305).

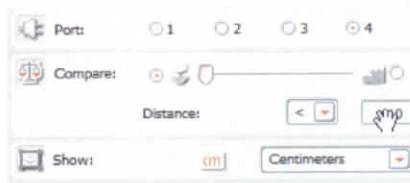


Figura 305. Paso 6 Programación (Color)

7. En el lado verdadero de este bloque switch colocar un bloque de sonido (Figura 306).



Figura 306. Paso 7 Programación (Color)

8. Configurar el bloque de sonido para que utilice el archivo “Ahnoo” (Figura 307)

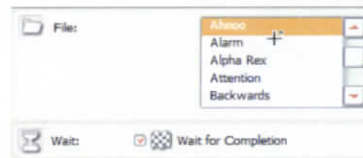


Figura 307. Paso 8 Programación (Color)

9. En el lado falso del anterior bloque switch colocar un bloque switch controlado por el sensor de colores (Figura 308).

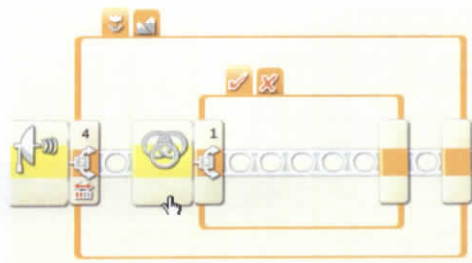


Figura 308. Paso 9 Programación (Color)

10. Configurar este bloque switch para que funcione con el puerto 1 y que la condición verdadera se cumpla en los colores 8 y 9 (Figura 309).

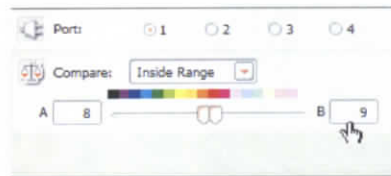


Figura 309. Paso 10 Programación (Color)

11. En el lado verdadero del anterior bloque colocar un bloque de sonido (Figura 310).



Figura 310. Paso 11 Programación (Color)

12. Configurar este bloque de sonido para que utilice el archivo “!Applause” (Figura 311).

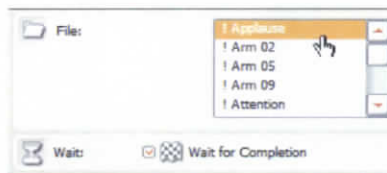


Figura 311. Paso 12 Programación (Color)

13. En el lado falso del anterior bloque switch colocar otro bloque switch controlado por el sensor de colores (Figura 312).

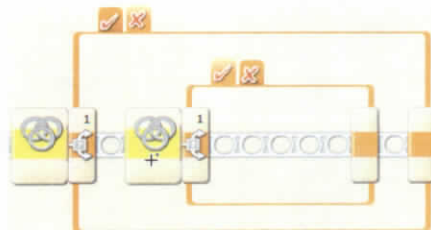


Figura 312. Paso 13 Programación (Color)

14. Configurar este bloque switch para que funciones con el puerto 1 y que el valor verdadero se cumpla entre los colores 3 y 5 (Figura 313).

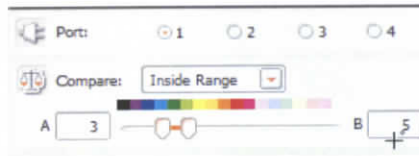


Figura 313. Paso 14 Programación (Color)

15. En el lado verdadero del bloque switch colocar un bloque de sonido (Figura 314).



Figura 314. Paso 15 Programación (Color)

16. Configurar este bloque de sonido para que utilice el archivo "Goodmorning" (Figura 315).

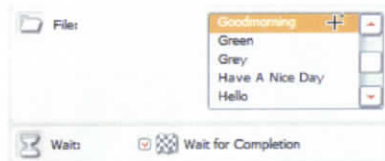


Figura 315. Paso 16 Programación (Color)

17. Colocar en lado falso del anterior bloque switch otro bloque switch controlado por el sensor de colores (Figura 316).

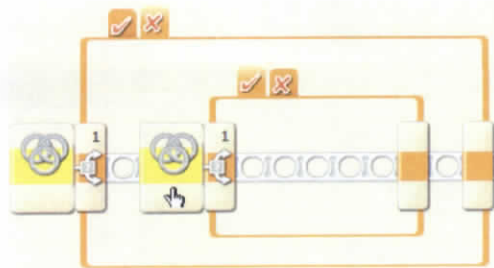


Figura 316. Paso 17 Programación (Color)

18. Configurar este bloque switch para que funciones con el puerto 1 y que el valor verdadero se cumpla entre los colores 1 y 2 (Figura 317).

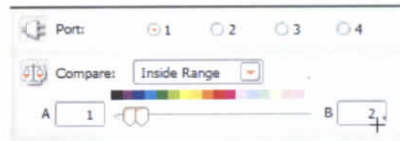


Figura 317. Paso 18 Programación (Color)

19. En el lado verdadero del bloque switch colocar un bloque de sonido (Figura 318).



Figura 318. Paso 19 Programación (Color)

20. Configurar este bloque de sonido para que utilice el archivo "Have A Nice Day" (Figura 319).

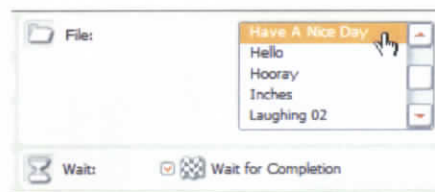


Figura 319. Paso 20 Programación (Color)

21. Colocar en lado falso del anterior bloque switch otro bloque switch controlado por el sensor de colores (Figura 320).

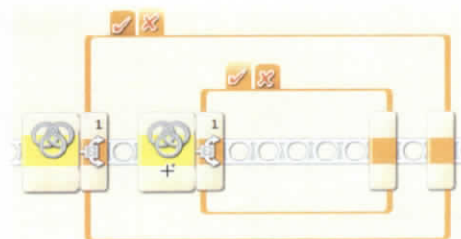


Figura 320. Paso 21 Programación (Color)

22. Configurar este bloque switch para que funcione con el puerto 1 y que el valor verdadero se cumpla entre los colores 6 y 7 (*Figura 321*).

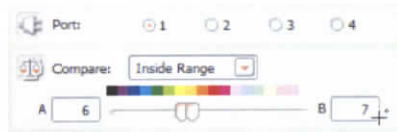


Figura 321. Paso 22 Programación (Color)

23. Finalmente para terminar la primera parte del programa colocar en el lado verdadero del bloque switch un bloque de sonido (*Figura 321*).



Figura 322. Paso 23 Programación (Color)

24. Configurar para que este bloque de sonido funcione con el archivo “Good Job” (*Figura 323*).

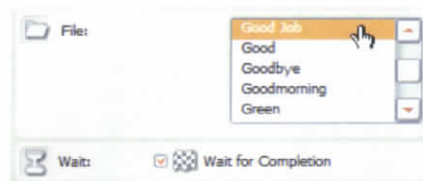


Figura 323. Paso 24 Programación (Color)

25. La primera parte del programa debería quedar así (*Figura 324*).

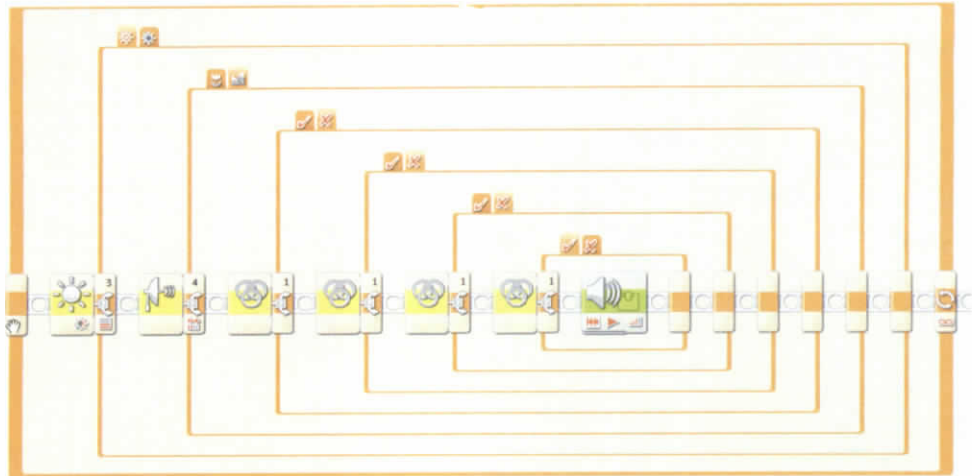


Figura 324. Primera parte Programa (Color)

26. Para la segunda parte colocar un bloque de repetición en la parte inferior y agrandar el punto de inicio para que se una con este bloque (Figura 325).



Figura 325. Paso 26 Programación (Color)

27. Configurar el bloque de repetición para que dure infinitamente (Figura 326).



Figura 326. Paso 27 Programación (Color)

28. Dentro del bloque de repetición colocar un bloque switch controlado por el sensor de luz (Figura 327).



Figura 327. Paso 28 Programación (Color)

29. Configurar este bloque switch para que funcione con el puerto 2 y que la condición sea verdadera para valores mayores que 20 niveles de luz (Figura 328).

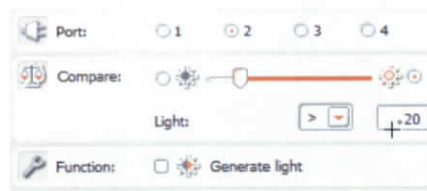


Figura 328. Paso 29 Programación (Color)

30. En el lado falso de este bloque switch colocar un bloque de movimiento (Figura 329).

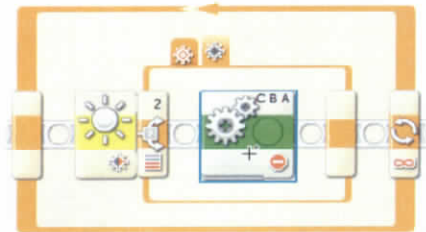


Figura 329. Paso 30 Programación (Color)

31. Configurar este bloque de movimiento para que controle los motores A, B y C y que detenga los motores (Figura 330).

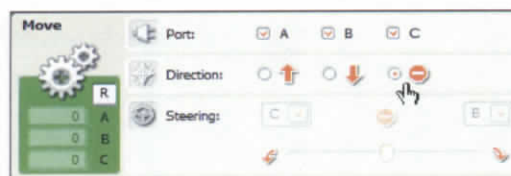


Figura 330. Paso 31 Programación (Color)

32. En el lado verdadero del anterior bloque switch colocar otro bloque switch controlado por el sensor de ultrasonido (*Figura 331*).



Figura 331. Paso 32 Programación (Color)

33. Configurar este bloque switch para que trabaje con el puerto 4 y que la condición verdadera se cumpla para valores menores a 10 centímetros (*Figura 332*).

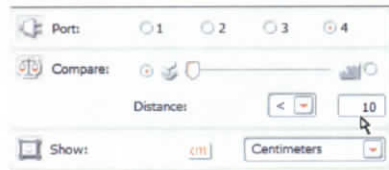


Figura 332. Paso 33 Programación (Color)

34. En el lado verdadero de este bloque switch colocar un bloque de movimiento (*Figura 333*).

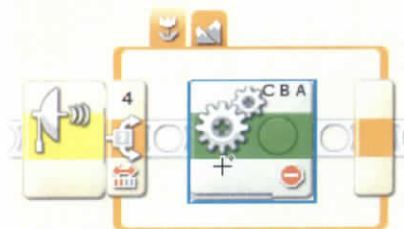


Figura 333. Paso 34 Programación (Color)

35. Configurar este bloque de movimiento para que controle los motores A, B y C y que detenga los motores (*Figura 334*).

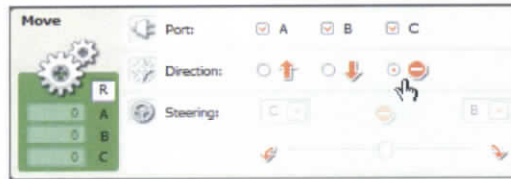


Figura 334. Paso 35 Programación (Color)

36. En el lado falso de anterior bloque switch colocar otro bloque switch controlado por el sensor de colores (Figura 335).

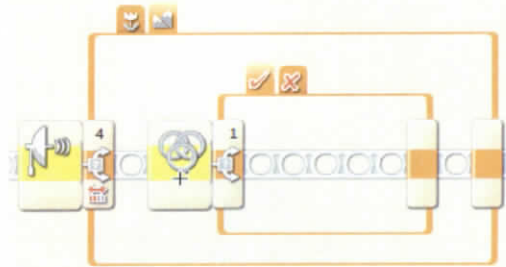


Figura 335. Paso 36 Programación (Color)

37. Configurar este bloque switch para que funcione con el puerto 1 y con los colores de 16 a 17 (Figura 336).

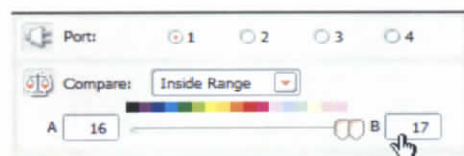


Figura 336. Paso 37 Programación (Color)

38. En lado verdadero de este bloque switch colocar 3 bloques de movimiento uno a lado de otro (Figura 337).

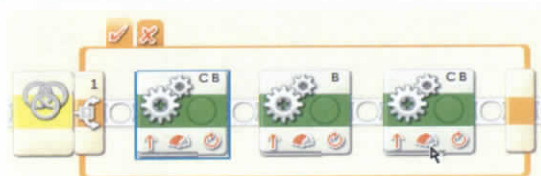


Figura 337. Paso 38 Programación (Color)

39. Configurar el primer bloque de movimiento para que su duración sea de 1 segundo (*Figura 338*).

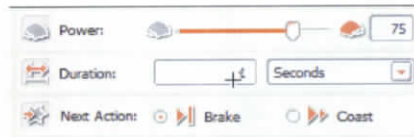


Figura 338. Paso 39 Programación (Color)

40. Configurar el segundo bloque de movimiento controle solo el motor B, su potencia sea al 100% y su duración sea de 35 rotaciones (*Figura 339*).

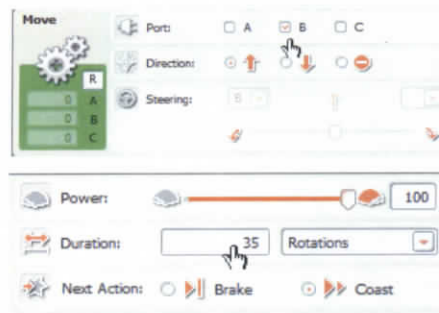


Figura 339. Paso 40 Programación (Color)

41. Configurar el tercer bloque de movimiento para que su duración sea de 1 segundo (*Figura 340*).

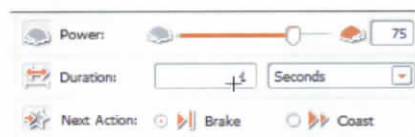


Figura 340. Paso 41 Programación (Color)

42. En el lado falso del bloque switch colocar una copia exacta del anterior bloque switch pero que trabaje con colores de 6 a 7 para su condición verdadera (*Figura 341*).



Figura 341. Paso 42 Programación (Color)

43. En la copia solo cambiar la configuración del segundo bloque de movimiento cambiando su dirección a reversa (Figura 342).

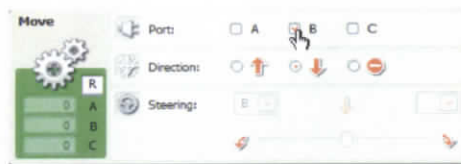


Figura 342. Paso 43 Programación (Color)

44. En el lado falso de este bloque switch colocar otro bloque switch controlado por el sensor de colores que maneje colores de 0 a 1 (Figura 343).

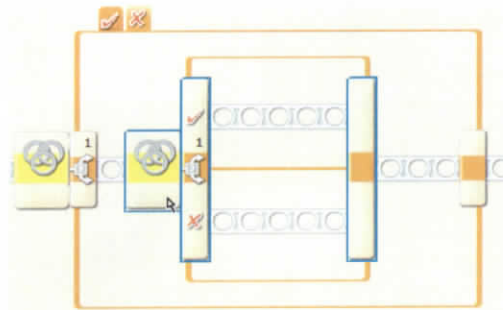


Figura 343. Paso 44 Programación (Color)

45. En el lado falso de este bloque switch colocar un bloque de movimiento que tenga una duración de 1 segundo (Figura 344).

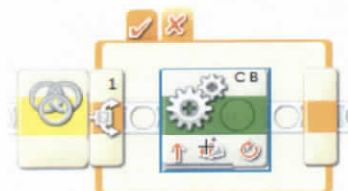


Figura 344. Paso 45 Programación (Color)

46. En el lado verdadero colocar 2 bloques de movimiento uno a lado del otro (Figura 345).



Figura 345. Paso 46 Programación (Color)

47. Configurar el primero bloque de movimiento para que tenga una duración de 5 segundos y el segundo para que detenga los motores C y B (Figura 246).

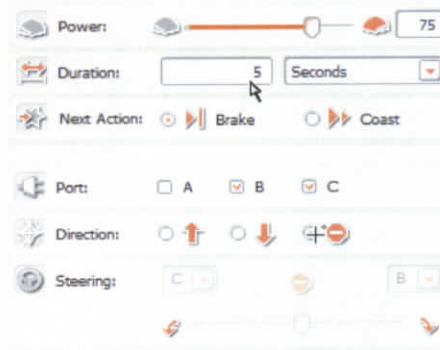


Figura 346. Paso 47 Programación (Color)

48. Colocar 2 bloques de sonidos después de los bloques de movimiento (Figura 347).

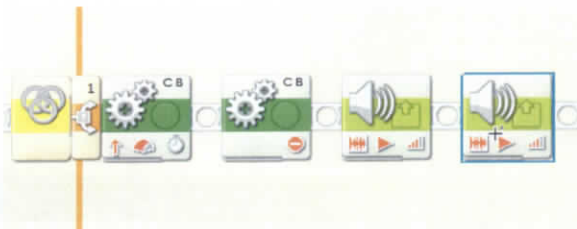


Figura 347. Paso 48 Programación (Color)

49. Configurar el primer bloque de sonido para que utilice el archivo "Play" y el segundo bloque de sonido para que utilice el archivo "Music" (Figura 348).

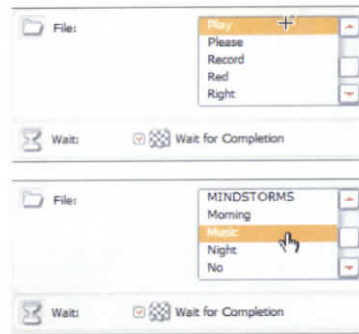


Figura 348. Paso 49 Programación (Color)

50. A lado del segundo bloque de sonido colocar un bloque de espera controlado por el sensor de sonido (Figura 349).



Figura 349. Paso 50 Programación (Color)

51. Configurar el bloque de espera para que se accione con valores mayores que 25 decibeles de sonido (Figura 350).

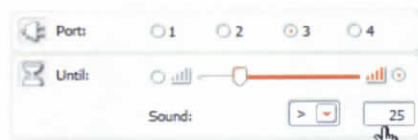


Figura 350. Paso 51 Programación (Color)

52. Colocar 2 bloques de movimiento a lado del bloque de espera (Figura 351).



Figura 351. Paso 52 Programación (Color)

53. Configurar el primero bloque de movimiento para que controle el motor A y su duración sea de 2 segundos; y configurar el segundo bloque de movimiento para que detenga el motor A (Figura 352).

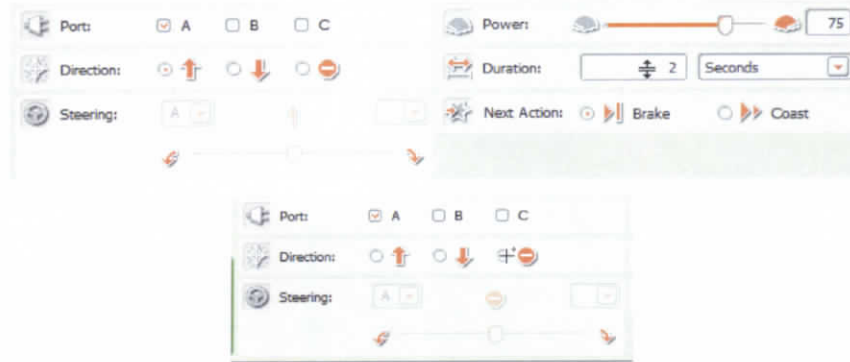


Figura 352. Paso 53 Programación (Color)

54. Después del segundo bloque de movimiento colocar un bloque de sonido y configurarlo para que utilice el archivo “Goodbye” (Figura 353).



Figura 353. Paso 54 Programación (Color)

55. Colocar un bloque de movimiento después del bloque de sonido (Figura 354).



Figura 354. Paso 55 Programación (Color)

56. Finalmente configurar este bloque de movimiento para que su duración sea de 5 segundos (Figura 355).

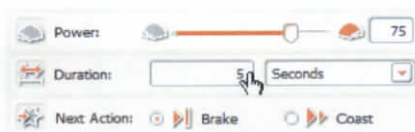


Figura 355. Paso 55 Programación (Color)

Para probar cualquier proyecto es necesario siempre bajarlo primero al bloque NXT simplemente presionando el botón de descarga (Figura 356).



Figura 356. Botón de Descarga

4. VALIDACIÓN Y VERIFICACIÓN DE RESULTADOS

4.1 Validación.

Ambato septiembre 03, 2008.

Ingeniero

Santiago Acurio Maldonado

DIRECTOR

Escuela de Ingeniería en Sistemas

PUCESA

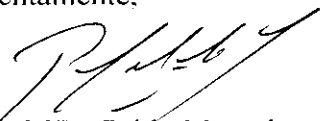
Presente

De mi consideración:

Me permito informar a usted que luego de haber realizado la validación de la Disertación de Grado titulada: **“DESARROLLO Y CONSTRUCCION DE UN HUMANOIDE UTILIZANDO LAS HERRAMIENTAS PROPORCIONADAS POR EL EQUIPO DE ROBOTICA LEGO MINDSTORM NXT”**, realizada por el señor Edmundo José Santacruz López, ha cumplido con las expectativas y sugerencias hechas para su validación.

Sin otro particular, suscribo de usted.

Atentamente,



Ing. MSc. Pablo Montalvo

PROFESOR-VALIDADOR

Ambato septiembre 16, 2008.

Ingeniero

Santiago Acurio Maldonado

DIRECTOR

Escuela de Ingeniería en Sistemas

PUCESA

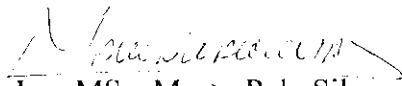
Presente

De mi consideración:

Me permito informar a usted que luego de haber realizado la validación de la Disertación de Grado titulada: **“DESARROLLO Y CONSTRUCCION DE UN HUMANOIDE UTILIZANDO LAS HERRAMIENTAS PROPORCIONADAS POR EL EQUIPO DE ROBOTICA LEGO MINDSTORM NXT”**, propuesta por el señor Edmundo José Santacruz López, ha cumplido con las expectativas y sugerencias hechas para su validación.

Sin otro particular, suscribo de usted.

Atentamente,



Ing. MSc. Marco Polo Silva

PROFESOR-VALIDADOR

5. CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones.

- El ser humano es una creación de Dios y lograr imitar esa creación a la perfección todavía no está en nuestras manos, en este proyecto se logro una imitación básica de los sentidos del hombre lo cual me hizo comprender la gran complejidad del funcionamiento del ser humano.
- Se comprueba que el equipo de robótica LEGO Mindstorm NXT tiene gran capacidad de flexibilidad e incentiva a la creación y diseño de nuevos modelos en donde sólo la imaginación es el límite
- Por el contenido de esta tesis se puede concluir que el material escrito de la investigación presente puede ser usado para la enseñanza tanto en el ámbito teórico como práctico del área de autómatas y robótica.
- La entrega del material usado en esta tesis ayudará a la realización de prácticas en el área de autómatas, robótica e inteligencia artificial.
- Se concluye que tanto el diseño de la estructura física como de la programación la estructura lógica tiene igual importancia para la construcción de un robot o algún dispositivo electrónico.

5.2 Recomendaciones.

- Se recomienda a la Escuela de Ingeniería de Sistemas poner énfasis en la automatización de procesos ya que ese es el futuro de la carrera y lo que ayudará a los futuros ingenieros a integrarse en el tan difícil mercado actual.
- También se debería incluir tanto la información escrita como el material práctico de esta tesis dentro del pensum de la asignatura referente a autómatas y robótica.
- Las prácticas dentro de la tesis sirven para comenzar a entender el funcionamiento de cada una de las partes del equipo de robótica LEGO Mindstorm NXT, por eso se las recomienda utilizar para el inicio del estudio de este equipo.

- Al realizar cada una de las prácticas que constan en el proyecto se recomienda guardarlos por separado para poder probar cada uno y a la vez poder utilizar partes de cada práctica en el caso que se quiera realizar una nueva práctica.
- Para mantener la integridad del equipo se recomienda tener sumo cuidado al momento de utilizar y guardar el equipo, se deberá mantener un orden establecido ya que el equipo consta de varias piezas y algunas de ellas son muy pequeñas.

BIBLIOGRAFÍA

Documentación:

- MINDSTOORM: "Ayuda del software LEGO Mindstorm NXT", 1º Edición, EEUU, 2006.
- MINSKY, Marvin: "Robótica La última frontera de la alta tecnología", edit. Planeta, 1º Edición, Barcelona España, 1986.
- BATURONE, Aníbal: "Robótica: Manipuladores y Robots Móviles", edit. Marcombo, 1º Edición, Madrid España, 2001.
- TORRES, Fernando: "Robots y Sistemas Sensoriales", edit, Prentice Hall, 1º Edición, Madrid España, 2002.
- RUSSELL, Stuart: "Inteligencia Artificial: Un enfoque moderno", edit. Prentice Hall, 1º Edición, 2003.

Internet:

- <http://mindstorms.lego.com/>
Página oficial del equipo de robótica Lego Mindstorm NXT.
- <http://www.lego.com/dacta/robofab>
Información técnica del equipo.
- <http://www.anser.com.ar/robotica.htm>
Robótica básica.
- http://es.wikipedia.org/wiki/Circuito_digital
Electrónica digital.
- <http://www.philohome.com/nxtmotor/nxtmotor.htm>
Descripción técnica de los servomotores.
- http://espanol.geocities.com/robbottotem/Historia_robotica.htm
Datos históricos de la robótica.
- <http://www.edured.gob.sv/Comunidad/blogs/carmelita/archive/2007/03/12/9336.aspx>
Información técnica de las piezas.

ACRÓNIMOS

AND: Comparador lógico.

AUSART: Acronyms Universal Synchronous Asynchronous Receiver Transmitter.

BDK: Bluetooth Development Kit.

CPU: Unidad Central de Procesos.

EEPROM: Electrically Erasable Programmable Read Only Memory.

EUSART: Enhanced Universal Synchronous Asynchronous Receiver Transmitter.

GNU: General Public License.

HDK: Hardware Development Kit.

IA: Inteligencia Artificial.

JAL: Just Another Language.

LCD: Liquid Crystal Display.

LED: Light Emitting Diode.

LEGO: Empresa danesa reconocida por sus bloques plásticos interconectables.

MPLAB: Herramienta para escribir y desarrollar código en lenguaje ensamblador.

MSSP: Managed Security Service Provider

NOT: Comparador lógico.

NXT: Cerebro central del equipo de robótica Lego Mindstorm NXT.

OR: Comparador lógico.

PIC: Circuito Integrado Programable.

PICAXE: Microcontroladores reprogramables.

PVC: Poly Vinyl Chloride.

RAM: Random Access Memory.

ROM: Read Only Memory.

SDK: Software Development Kit.

UCP: United Cerebral Palsy.

USART: Universal Synchronous Asynchronous Receiver Transmitter.

USB: Universal Serial Bus.

Ontogenética: Formación y desarrollo individual de un organismo, referido en especial al período embrionario.

Filogenética: Origen y desarrollo evolutivo de las especies.

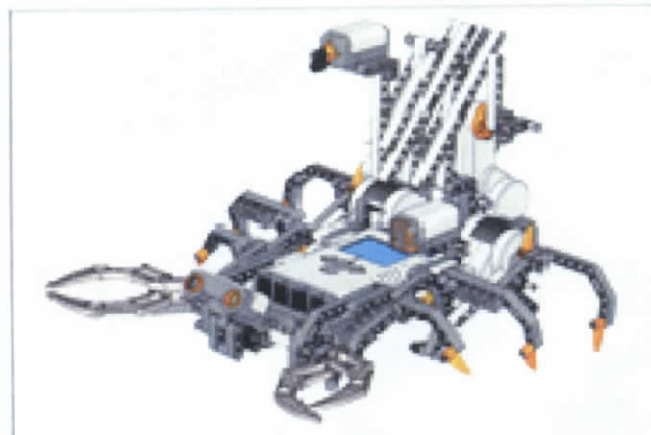
ANEXOS

OTROS MODELOS

A continuación se describirán algunos de los modelos que se podría armar con el equipo de robótica LEGO Mindstorm NXT. Cabe notar que no solo se puede construir estos modelos sólo son un ejemplo de lo que el equipo puede hacer.

Spike

Robot tipo animal, que reacciona como un verdadero escorpión. El robot puede caminar como un insecto, tiene pinzas, puede ver y oír con el sensor de sonido y el sensor de ultrasonido, y ataca con su cola al detectar algún objeto.



Su programación es simple y permite al escorpión el blanco, reproducir un sonido cuando el blanco sea atacado, retraer su cola y regresar al punto de inicio.

Robot-Arm

Robot tipo máquina, que puede alzar mover y dejar objetos con sus tenazas. Puede detectar color con el sensor de luz, sentir objetos con el sensor de tacto. Utiliza los 3 motores, 1 para manejar sus tenazas y los otros 2 permite mover el brazo arriba, abajo y girar.