



**UNIVERSIDAD CENTRAL DE VENEZUELA
FACULTAD DE CIENCIAS
ESCUELA DE COMPUTACIÓN**

Escáner 3D Móvil utilizando Lego Mindstorms y Dispositivos Android en red

Seminario

**Bachilleres: Iñaki Xabier De Tejada Plessmann
Y
Fernando De Freitas**

**Tutores: Profesor Eugenio Scalise
Y
Profesor Esmitt Ramírez**

Caracas, 14 de Noviembre de 2013

Índice

Introducción	iv
Capítulo 1 - Lego Mindstorms	1
1.1 Reseña Historica	1
1.1.1 Cooperación Lego- MIT Media Lab - Resnick and Papert's Epistemology and Learning Research Group (1987)	3
1.1.2 El primer Mindstorms: el bloque RCX (1998)	6
1.1.4. El Cybermaster	6
1.1.5. NXT	9
1.1.6. EV3	10
1.2 Especificaciones del hardware	10
1.2.1. RCX	11
1.2.2. NXT	14
1.2.3. EV3	15
1.3 Alternativas al Lego Mindstorms	15
1.3.1 Raspberry Pi	17
1.3.2 Arduino	19
Capítulo 2 - Herramientas de Desarrollo	19
2.1. Herramientas para Diseño y Ensamblaje	19
2.1.1 Lego Digital Designer (LDD) (NXT y Lego en general)	22
2.1.2 Autodesk Inventor Publisher (EV3)	23
2.2. Herramientas de Programación	23
2.2.1 LogoBlocks (RCX)	24
2.2.2 NXT-G Programming Environment (NXT)	25
2.2.3 AForge.NET (C# - NXT RCX)	26
2.2.4 - nxtOSEK (ANSI C/C++ - NXT)	26
2.2.5 nxt-python (Python - NXT)	27
2.2.6 hackage-NXT (Haskell- NXT)	27
2.2.7 leJOS NXJ (Java - NXT RCX)	28
2.2.8 Android SDK (Dispositivos Android)	30
2.2.9 OpenCV	30
2.2.10 VTK	31
2.2.11 CGAL	32
Capítulo 3 - Propuesta de desarrollo	32
3.1 Motivación	33
3.2 Objetivo General	33
3.3 Objetivos Específicos	34
3.4 Propuesta de Solución	34
3.4.1 Esquema propuesto	34
3.4.2 Hardware	36
3.4.3 Software	36
Controlador del NXT	36

Controlador de la Cámara	36
Controlador General	37
3.5.4 Herramientas a Utilizar	37
3.5.5 Prototipos	38
Prototipo Robot Mindstorms	38
Aplicación del Controlador General	39
Aplicación del Controlador de las Cámaras	41
Guía	42
Referencias	43

Introducción

Con el avance de la tecnología y la popularización de las ciencias de la computación, la cantidad de personas interesadas en inteligencia artificial, robótica y demás disciplinas similares, así como el mercado para los productos derivados de éstas, se encuentra en crecimiento.

Actualmente el momento en que se puedan emplear robots para encargarse de las tareas cotidianas, como en las obras de Ciencia Ficción es una imagen muy distante, pero desde hace muchos años se utilizan robots en la industria, las líneas de producción de las principales compañías automotrices los usan para ensamblaje, para la elaboración de piezas que necesiten de una precisión milimétrica y para hacer mediciones de control de calidad.

Existen otros campos en los que también son usados actualmente los robots, en la medicina se utilizan para realizar intervenciones quirúrgicas de manera remota. En el ejército y la policía se emplean para misiones riesgosas como desactivar bombas o para combate y reconocimiento, como es el caso de los *Drones* o UAVs (*Unmanned Aerial Vehicle*).

Hasta ahora se enumeran casos donde son usados para tareas de alto impacto y notoriedad, pero su alcance también llega a tareas más simples y poco críticas. Esto genera la necesidad de robots más baratos y accesibles al público en general.

A partir de esta necesidad varias compañías dirigieron su atención a ese mercado, con un gran déficit de oferta y un alto potencial de rentabilidad, y decididas a sacarle provecho comenzaron a buscar formas de satisfacer esta demanda. Para esto era necesario hacerlos más simples de diseñar y programar, ya que hasta hace pocos años sólo aquellos grupos con conocimientos avanzados en electrónica y mecánica podían incursionar en el campo de la creación de robots.

Es así como nacen los Lego Mindstorms, los Arduino y los Raspberry Pi, entre otras alternativas de hardware a bajo costo. De éstos, los Mindstorms son unos de los más populares gracias a su enfoque didáctico, a la fama de la marca Lego y al trabajo de la comunidad para explotar el potencial del bloque.

En este trabajo se recopila la información general necesaria para diseñar y construir un robot, así como implementar su software, teniendo una idea de las funciones que se desea que desempeñe, enfocándose principalmente en el bloque Mindstorms de Lego.

En el capítulo 1 se hace una reseña histórica al bloque Mindstorms así como las especificaciones de sus diferentes versiones y algunas de sus principales alternativas en el mercado actual.

El capítulo 2 enumera y describe brevemente las principales herramientas de diseño y desarrollo existentes para dichos bloques, entre ellos frameworks, ambientes de desarrollo integrados y herramientas para ensamblado.

Finalmente, en el capítulo 3 se plantea la propuesta de desarrollo de un robot móvil con funciones de escáner 3D utilizando dispositivos Android y un bloque Lego Mindstorms.

Capítulo 1 - Lego Mindstorms

En los últimos 60 años Lego ha sido la compañía líder en entretenimiento infantil, manteniéndose a la vanguardia de la tecnología para conservar este puesto. Y con el pasar de los años la robótica ha sido una de esas tecnologías vanguardistas que se ha convertido en el nuevo campo a explorar por la empresa para satisfacer a las nuevas generaciones. En esta sección se hace una reseña del origen del Lego Mindstorms y sus primeros prototipos no comerciales, se describen las diferentes versiones que han salido al mercado, junto con sus especificaciones técnicas. Y las alternativas más populares a éstos que se encuentran actualmente en el mercado.

1.1 Reseña Historica

El Lego Mindstorms Robotic Invention Kit, conocido también como “set Mindstorms” es el resultado de la cooperación entre un conjunto de institutos y empresas para proporcionarle al mundo académico, y el público en general, una línea de productos de robots programables que disminuyeran las barreras de entrada al mundo de la electrónica y la programación así como impulsar dichos campos[1]. En esta sección se muestra el origen de esta cooperación y los resultados que han logrado que el set Mindstorms se haya hecho tan popular. Así como también una breve reseña de las versiones más importantes de esta línea de productos de Lego, así como algunas las alternativas existentes en el mercado.

1.1.1 Cooperación Lego- MIT Media Lab - Resnick and Papert's Epistemology and Learning Research Group (1987)

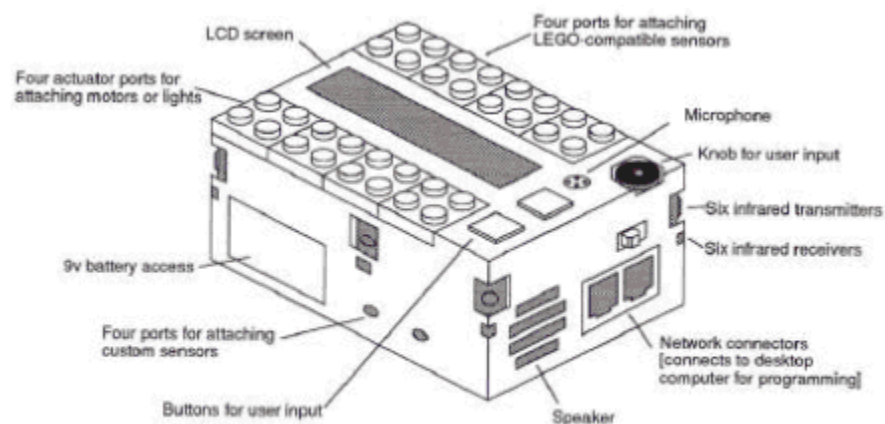
El set Mindstorms es el resultado de un esfuerzo conjunto entre Lego, el MIT Media Lab (MIT-ML) y el Resnick and Papert's Epistemology and Learning research group (RAPEALG) que en el año 1987 unieron fuerzas para desarrollar un bloque programable que se ajustara a las necesidades que tenía cada uno de los involucrados.

Lego quería recuperar parte del mercado que había perdido frente a los juguetes electrónicos y consolidarse como la marca líder en el mercado infantil, el RAPEALG quería plantear nuevos modelos de aprendizaje y el MIT-ML quería crear un modelo de investigación que fuera atractivo al público y sirviera de trampolín al desarrollo de la robótica.

Esta no fue una tarea sencilla, para desarrollar un bloque programable se deben tomar en cuenta muchos aspectos técnicos en el diseño del hardware y software:

En cuanto al software, se decidió utilizar el lenguaje de programación Logo[2], desarrollado por Daniel Bobrow, Wallace Feurzeig y Seymour Papert del grupo RAPEALG en el año 1960, como un lenguaje de programación para niños, orientado al aprendizaje y con una sintaxis simple pero poderosa que sigue la filosofía “*low-threshold high-ceiling*” de Papert [3], el compilador de Logo, para la versión RCX, fue desarrollado por Brian Silverman.

Es importante destacar que el bloque “gris” o *Pocket Programmable Brick* (PPB) (Ver Figura 1.1), uno de los prototipos intermedios, poseía su propio intérprete de Logo. También se desarrolló un prototipo de LegoBlocks un Entorno de Desarrollo Integrado (o IDE por sus siglas en inglés) para Logo, a cargo de Andy Begel, un estudiante del MIT-ML.



Source: Randy Sargent's MAS Master's Thesis, 1995

Figura 1.1. Bloque “Gris”, creado por Martin en 1995.

Para el hardware, Fred G. Martin fue el principal encargado del diseño junto con otros estudiantes del MIT-ML, quien presentó muchas versiones del bloque hasta que en 1996 surgió la versión final, un bloque rojo con cuatro puertos de salida y seis puertos de entrada conocido simplemente como el bloque “rojo” (Ver Figura 1.2). Este bloque contaba con seis puertos de entrada, cuatro motores, una entrada infrarroja y una salida de audio. Además se podían agregar dispositivos externos como un micrófono o una salida infrarroja.



Figura 1.2. Bloque “Rojo”, creado por Martin en 1996.

1.1.2 El primer Mindstorms: el bloque RCX (1998)

El bloque desarrollado por Martin fue la base para el RCX, este bloque está provisto de un microcontrolador Hitachi H8/3292 y 512Kb de memoria RAM. En comparación con el bloque rojo y sus prototipos previos, gran parte de las modificaciones hechas al RCX fueron hechas por razones de marketing, debido al público para el que estaba enfocado su lanzamiento inicialmente, niños con edades de entre 10 y 14 años.

La cantidad de puertos fue reducida a 6 (3 de entrada y 3 de salida) para hacer un bloque que fuera más económico de fabricar. Las conexiones para dichos puertos ya habían sido adaptadas para asemejar bloques de Lego para el bloque rojo, y se eligieron los colores amarillo y negro ya que recuerdan a una zona de construcción (Ver Figura 1.3). Salió al mercado en un kit llamado *Robotics Invention System* (RIS), el cual contenía inicialmente dos motores, 2 sensores de contacto, 1 sensor de luz y alrededor de 700 partes *Lego Technic*[4].

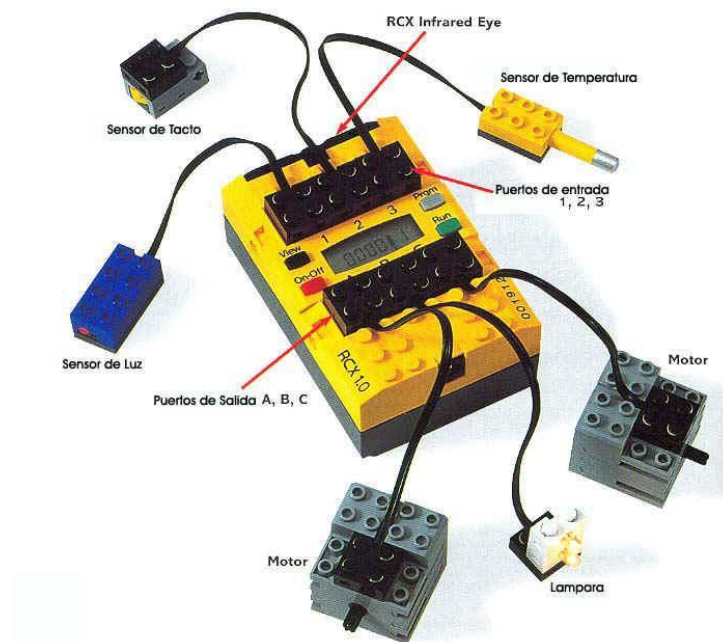


Figura 1.3. Bloque RCX con sensores y accesorios.

Con la salida del RCX al mercado se llenó un vacío que existía en muchas personas interesadas en robótica pero que no tenían acceso a un robot de precio razonable y que fuese relativamente simple de programar, a tal punto que más de un millón de personas aseguraron que el RCX era justo lo que buscaban[5]. Tal popularidad hizo que el intérprete de Logo que traía el RCX no fuese suficiente para la comunidad y comenzó un proceso para descubrir todo el potencial del bloque.

Lego quedó sorprendida al ver que al poco tiempo de estar en el mercado el RCX fue desbloqueado y lenguajes como Java tomaron el lugar de Logo. Hoy en día se pueden conseguir frameworks para desarrollar en Java, C++, Haskell, C#, Python y muchos más.

En sus primeros tres meses en el mercado las ventas del RCX sobrepasaron las expectativas de Lego, con más de 80.000 unidades vendidas rápidamente empezaron a llegar al mercado una gran cantidad de accesorios para el RCX, y se formó rápidamente una comunidad de adultos y hackers alrededor de este dispositivo, lo que hizo a Lego cambiar sus estrategias de mercadeo para la siguiente iteración de los Lego Mindstorms.

En las figuras 1.4 y 1.5 se muestran algunos datos de ventas del RCX[6]:

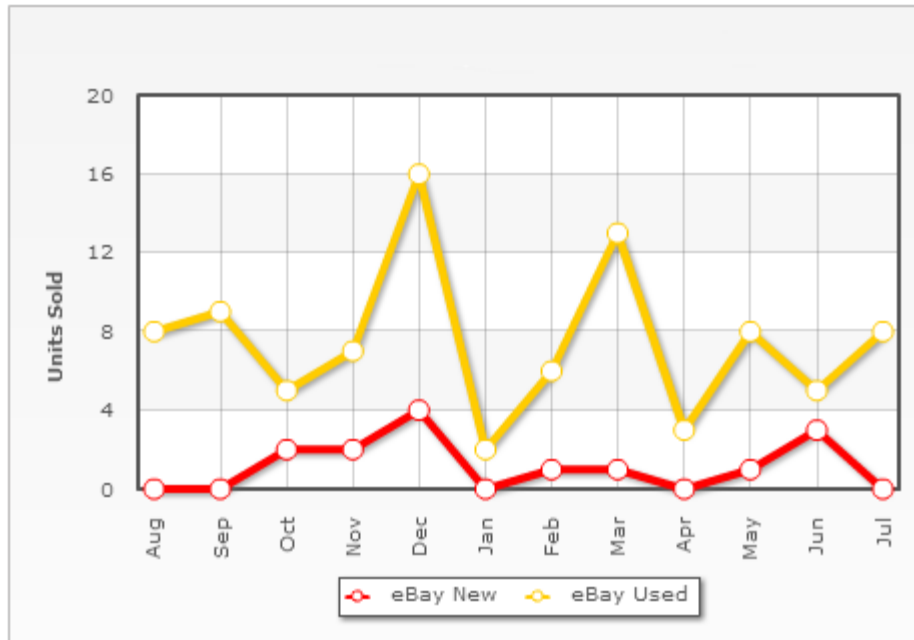


Figura 1.4. Ventas (Unidades) de LEGO Mindstorms RCX para el periodo Agosto 2012 - Julio 2013.

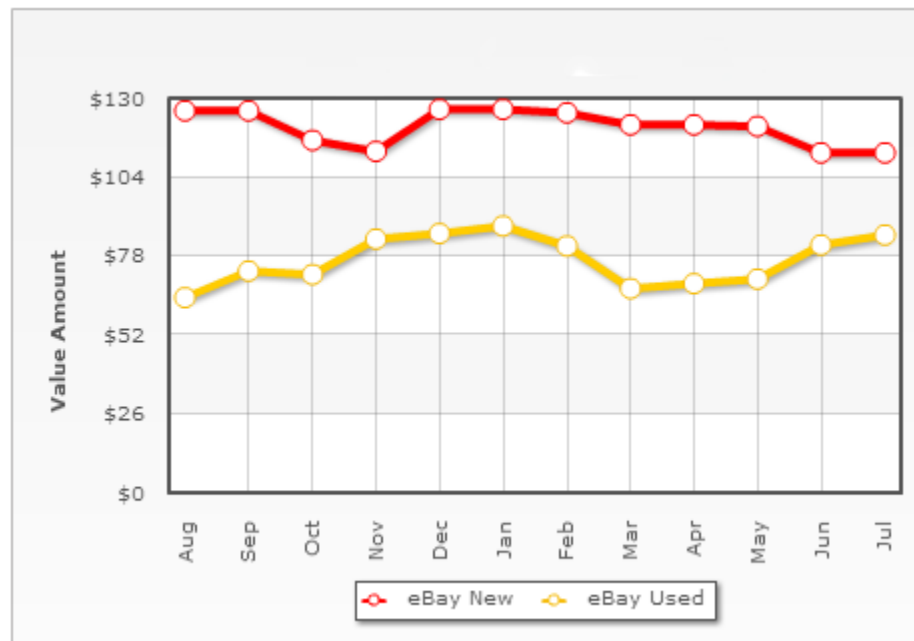


Figura 1.5. Ventas (Montos) de LEGO Mindstorms RCX para el periodo Agosto 2012 - Julio 2013.

En estas figuras se puede observar que el volumen de ventas es casi nulo fuera de épocas de alto consumo, como las festividades decembrinas, y se debe tener en consideración que para el año 2013 este es un producto que tiene 15 años en el mercado y se ha

revalorizado por su carácter de ser un artículo de colección. Aumentando a lo largo del año (2013) debido al revuelo creado por la cercanía del lanzamiento del próximo modelo (EV3) en el último trimestre de 2013.

1.1.4. El Cybermaster

Una vez que fue descubierto todo el potencial del bloque, atrajo más adultos que niños. Se estima que el 70% de los usuarios de Mindstorms son adultos cuya edad promedio es de 30 años. A pesar del éxito comercial, ésta estadística mostró que el objetivo inicial del Mindstorms no se estaba cumpliendo, es por esto que Lego decidió simplificar el RCX para enfocarlo mejor a su target de mercado original (niños de entre 10 y 14 años)[1]. De esta transformación nacieron el Scout, el Spybot y en 1998 el Cybermaster.

El Cybermaster era una versión menos poderosa de RCX, más barata, que se vendió en un kit parecido al RIS. La principal atracción del Cybermaster era la capacidad de comunicarse por radio con una PC. Este intento de atraer un grupo más joven no fue tan exitoso como fue el RCX en general.

1.1.5. NXT

En 2004, Lego designó a Søren Lund para crear la nueva generación del Mindstorms, adaptándose a nuevas tecnologías y dando así potencial para mucho más de lo que ofrecía el RCX. Se dice que en la puerta del laboratorio de Lund había un cartel que decía “Haremos por la robótica lo que hizo el iPod por la música”[1], esto pudo influir sobre el diseño del NXT que es muy parecido al de un iPod de primera generación. El 2 de Agosto de 2006 el NXT fue lanzado, manteniendo al MIT-ML y al RAPEALG como aliados.



Figura 1.6. Bloque NXT con sensores y accesorios.

El NXT representa un gran avance con respecto al RCX. En este bloque decidieron no simplificar sino agregar nuevas habilidades, por ejemplo el uso de Bluetooth y de I²C, un protocolo de bus serial utilizado para interactuar con periféricos de baja velocidad (100kbps). El bloque tiene 3 puertos de salida y 4 puertos de entrada (uno más que el RCX). Otro cambio del NXT fue en los conectores, el NXT abandonó los conectores parecidos a un bloque de Lego y pasó a usar conectores RJ11 (Ver Figura 1.6).

El NXT se comercializa en un kit llamado Lego Mindstorms NXT 2.0, el kit consiste del bloque NXT, un sensor ultrasónico, un sensor de color, 2 sensores de contacto, 3 servomotores, un cable USB, cables con conectores RJ11, un manual de instrucciones, el software básico para ensamblar programas y 619 piezas de Lego.

El NXT representó un gran avance en el mundo de los Mindstorms, el uso de RJ11 en los conectores permitió que muchas compañías externas a Lego (y miembros de la comunidad con conocimientos en electrónica) pudieran crear nuevos sensores. Esto disparó el potencial del NXT para aplicar soluciones a miles de problemas. La integración de Bluetooth también permite que se puedan utilizar varios NXT para una tarea. Las especificaciones del NXT serán detalladas en la sección 1.2 de este capítulo.

Como era de esperarse, Lego no creó métodos de seguridad para prevenir que su software interno fuese modificado por terceros lo que permitió que, al igual que con el RCX, aparecieran frameworks no oficiales para sustituir Logo y así liberar todo el potencial del NXT.

En las figuras 1.7 y 1.8 se presentan las estadísticas de ventas del NXT 2.0 en el año (2012-2013)[7]:

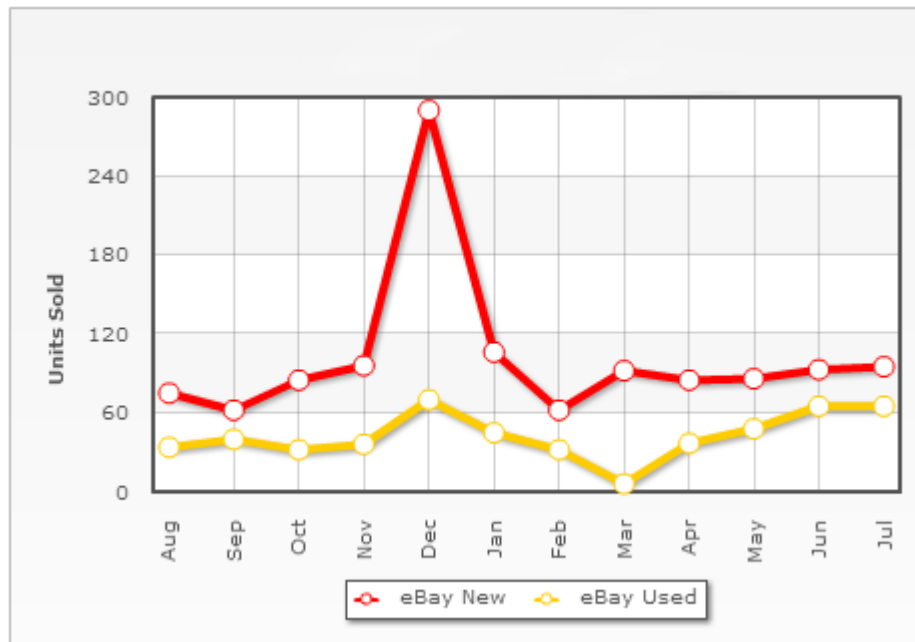


Figura 1.7. Ventas (Unidades) de LEGO Mindstorms NXT para el periodo Agosto 2012 - Julio 2013.

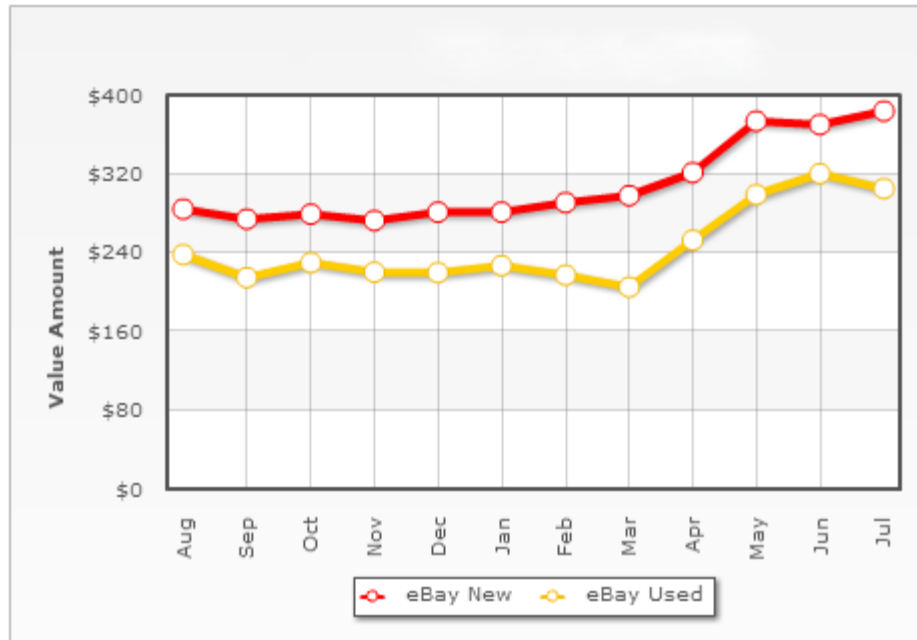


Figura 1.8. Ventas (Montos) de LEGO Mindstorms NXT para el periodo Agosto 2012 - Julio 2013.

En estas figuras se puede observar que el volumen de ventas se incrementa en la época navideña, y su precio ha ido aumentando a lo largo del año (2013) debido al revuelo creado por la cercanía del lanzamiento del próximo modelo (EV3) en el último trimestre de 2013.

1.1.6. EV3

La última generación de Lego Mindstorms, fue lanzada el 1 de Septiembre de 2013[8]. Entre sus mejoras se encuentran: hardware con mejores prestaciones que sus versiones anteriores, como memoria RAM, Procesador ARM9, memoria Flash, un SO basado en Linux, entradas USB 2.0 y un slot para memorias SD (Ver Figura 1.9). También se agregó interoperabilidad nativa con Android e iOS. Además de un conjunto nuevo de ambientes de desarrollo, se puede programar en el mismo bloque, mediante una interfaz gráfica similar a LegoBlocks o con un IDE estándar para programadores y *hackers*[9].



Figura 1.9. Bloque EV3.

1.2 Especificaciones del hardware

Como todos los dispositivos electrónicos, con el avance del tiempo es necesario lanzar nuevas versiones adaptadas a los requerimientos actuales para mantenerse a la vanguardia. Para el Mindstorms esto no ha sido distinto, necesitando cada vez más capacidad y funcionalidades para poder enfrentar los retos de los programadores, incorporar nuevas tecnologías y mantenerse competitivo en el mercado. Esta sección trata las especificaciones de las diferentes versiones del bloque, su hardware oficial y de terceros, así como accesorios y sensores para expandir el campo de acción del bloque.

1.2.1. RCX

El bloque RCX fue el primer Mindstorms comercial luego de muchos prototipos y modelos de prueba. Sus especificaciones se detallan en la Figura 1.10.

Especificaciones Técnicas	Microcontroladores		16-bit Hitachi H8/3292
	Memoria	Flash	16 KB
		RAM	512 B
	Puertos	Entrada	3 puertos RCX identificados del 1, 2 y 3
		Salida	3 puertos RCX identificados de la A a la C
	Pantalla		LCD de 43 segmentos
	Sonido		Corneta con calidad de sonido de 500Hz
	Botones		4 botones identificados como On-Off, View, Run, Prgm
	Transmisión de datos		Transmisor/Receptor Infrarrojo (IR) a 38.5Hz
	Opciones de poder		6 baterías AA para un total de 9.0 V
Accesorios	Oficiales		Motores
	No Oficiales		N/A
Sensores	Oficiales		- Sensor de contacto - Sensor de Temperatura - Sensor de Luz - Sensor de Rotación
	No Oficiales		N/A

Figura 1.10. Especificaciones del RCX.

1.2.2. NXT

Las mejoras en la versión NXT con respecto a la RCX fueron muchas en cuanto a extensibilidad y capacidad de hardware, dada la nueva interfaz y la gran popularidad que tuvo el RCX muchas compañías decidieron trabajar en crear opciones a los sensores oficiales y en crear sensores que Lego no ofrece.

En el mercado se consiguen 3 compañías principales en el desarrollo de sensores y accesorios para el NXT:

- **Hitechnic¹:** Fue la pionera en el desarrollo de sensores no oficiales, al punto de tener un trato con Lego para obtener las carcasas plásticas de los sensores oficiales para utilizarlos en sus sensores pero con colores diferentes (negro y gris) para diferenciarse. En la Figuras 2.3 y 2.4 se muestran los accesorios creados por esta marca para el NXT.
- **Mindsensors²:** Tiene más variedad de sensores a un precio más reducido, además que dan la posibilidad de comprar los conectores RJ11 que utiliza el NXT para poder tener cables que se adapten a las necesidades del ensamblador.

No tiene convenio con Lego para obtener las carcasas de plástico lo que hace que sus sensores no mantengan el estilo visual y que parezcan más como

¹ www.hitechnic.com

² www.mindsensors.com

tarjetas de circuitos conectadas a través de pines al NXT. En las Figuras 2.3 y 2.4 se muestran los accesorios creados por esta marca para el NXT.

- **Vernier³**: Es una compañía que se especializa en hacer sensores químicos para cualquier tipo de necesidad, estos sensores no están diseñados para el NXT, pero al ver el potencial de ventas decidieron desarrollar un adaptador que permitiera conectarle cualquiera de sus sensores

Las especificaciones junto con los accesorios y sensores (oficiales y no oficiales) disponibles para el bloque NXT se indican en las figuras 1.11, 1.12 y 1.13

Especificaciones técnicas	Microcontroladores (MC)		1) 32-bit ARM7 2) 8-bit AVR
	Memoria	Flash	- MC1, 256 KB - MC2, 4 KB
		RAM	- MC1, 64 KB - MC2, 512 B
	Puertos	Entrada	4 puertos RJ12. El cuarto puerto es un EIA-485 de alta velocidad (926.1 Kbps) numerados del 1 al 4
		Salida	3 puertos RJ12 identificados de la A a la C
	Pantalla		LCD de 100x64 pixels a 60 Hz
	Sonido		Corneta con calidad de sonido de 8 KHz, resolución de sonido de 8 bits y tasa de muestreo de 2 a 16 KHz
	Botones		4 botones, izquierda, derecha, seleccionar y On/Off, no etiquetados
	Transmisión de datos		- Bluetooth clase 2 V2.0 - USB 1.0 Full Speed (12 Mbps)
	Opciones de poder		6 baterías AA para proveer 7.2 V (requerimiento mínimo). Para la mayor duración se necesitan 9.0 V

Figura 1.11. Especificaciones del bloque NXT.

³ www.vernier.com/NXT/

Accesorios	Oficiales		<ul style="list-style-type: none"> - Batería recargable de Litio que provee 7.4 - 8.2 V. - Convertidor para conectar sensores del RCX. - Servo motor con puerto RJ12, ajustable en velocidad y fuerza, pesa 81g. - Set de 7 cables RJ12. Contiene uno de 20cm, 4 de 35cm y 2 de 50cm. - Antena Bluetooth para comunicarse inalámbricamente con el brick.
	No Oficiales	HiTechnic	<ul style="list-style-type: none"> - Faro Infrarrojo - Rotacaster - Bola electrónica infrarroja - Multiplexores para sensores digitales y de contacto - Cables - Tabla de diseño digital
		Mindsensors	<ul style="list-style-type: none"> - Multiplexores para Motores (para 2, 4 y 8 motores) - Multiplexores para 3 sensores de contacto o para 3 sensores digitales - Controladores de relays y de motores RF - Controlador inalámbrico para interfaz de control de Playstation 2 - Controlador inalámbrico para arbitro inalámbrico - Kit de ensamblaje de sensores Magic Wand - Extensión para cables RJ12 - Kit de ensamblaje PCF8591 IC - Kit de ensamblaje PCF8574 IC - Servo motor RC de 43 gr - Servo motor RC de rotación continua - Mini-Servo motor RC - Monturas para Servo motores Hitec, Modelcraft o Futaba - Kit de válvulas neumáticas operadas por servo motores - Kit de cables - Tester de cables RJ12 - Conectores RJ12 macho y hembra - Compartimiento para 4 y 6 baterías AA - Adaptador para batería LiPo - Conector para baterías RC
		Vernier	<ul style="list-style-type: none"> - Adaptador para la gama de sensores Vernier

Figura 1.12. Accesorios del bloque NXT.

Sensores	Oficiales		<ul style="list-style-type: none"> - Sensor de contacto - Sensor de luz - Sensor de sonidos - Sensor ultrasónico - Brújula - Sensor de color - Acelerómetro
	No Oficiales	HiTechnic	<ul style="list-style-type: none"> - Sensor de ángulo - Acelerómetro / sensor de inclinación - Sensor barométrico - Sensor de color - Detector electro-óptico de proximidad - Giroscopio - Sensor de conexión infrarrojo - Sensor de recepción infrarrojo - Sensor pasivo infrarrojo - Brújula - Sensor Magnético - Sensor de búsqueda infrarrojo
		Mindsensors	<ul style="list-style-type: none"> - Subsistema de visión - Panel táctil - Panel numérico - Detector infrarrojo de obstáculos en tres ángulos a dos rangos - Sensor digital de presión neumática - Sensor lineal - Reloj - Sensor infrarrojo de distancia de alta precisión de rango largo, medio o corto - Medidor de corriente - Medidor de voltaje - AbsoluteIMU - acelerómetro y brújula

Figura 1.13. Sensores del bloque NXT.

1.2.3. EV3

La nueva generación (2013) de Lego Mindstorms, debido a la inmensa popularidad de sus predecesores, tiene muchas mejoras en cuanto a software y hardware, las últimas reflejadas en las Figuras 1.14 y 1.15.

Especificaciones técnicas	Microcontroladores		Procesador ARM9 de 300 MHz con OS Linux integrado
	Memoria	Flash	16MB y slot de expansión miniSDHC de hasta 32GB
		RAM	64MB
	Puertos	Entrada	- 4 puertos RJ12 numerados del 1 al 4 - Puerto Análogo - Puerto Digital de 460.8 Kbps
		Salida	4 puertos RJ12 identificados de la A a la D
	Pantalla		Marix Display de 178x128
	Sonido		Corneta de Alta Calidad
	Botones		Interfaz de 6 botones con iluminación de 3 colores
	Transmisión de datos		- USB 2.0 - Bluetooth 2.1 - WiFi (mediante USB) - Hasta 4 Bricks conectados en serie funcionando como una unidad
	Opciones de poder		- 6 baterías AA - Batería DC Recargable EV3 de lithium ion con 2050 mAh
	Conectividad		- iOS - Android - Windows

Figura 1.14. Especificaciones del bloque EV3.

Accesorios	Oficiales		- Motores Grandes - Motores Pequeños - Control Remoto - Cables - Cable USB
	No Oficiales	N/A	N/A
Sensores	Oficiales		- Sensor de contacto - Sensor de color - Sensor Infrarrojo
	No Oficiales	N/A	N/A

Figura 1.15. Accesorios y Sensores del bloque EV3.

Lamentablemente debido a las diferencias en la arquitectura del bloque NXT y el EV3 gran parte de los programas desarrollados para el NXT no funcionan para esta versión más reciente, aunque dado el acelerado crecimiento de la comunidad muy probablemente esto sea subsanado en poco tiempo, ya sea con adaptaciones de dichos programas para NXT o con versiones completamente nuevas para EV3.

1.3 Alternativas al Lego Mindstorms

Lego no fue la única empresa que vio el potencial de producir un robot programable para el público, también surgieron alternativas un poco más complejas para un público más especializado o que quisiera un reto más grande.

La principal diferencia, ventaja o desventaja dependiendo de la perspectiva desde la que se vea, de los Lego Mindstorms es lo específico de su naturaleza. Su ventaja es que tiene un ecosistema de accesorios, sensores y tecnologías con características de productos finales, enfocados específicamente para él, más conciso y robusto que su competencia. La desventaja es que ese mismo ecosistema lo limita, haciéndolo parecer estar destinado a permanecer en el mundo Lego y a proyectos de pequeña escala si no cuenta con la complicitad de un usuario que se atreva a pensar *“outside of the box”* y que lo haga llegar más allá de las limitaciones inherentes a su diseño.

Los dos principales competidores del Lego Mindstorms son un microcontrolador (Arduino) y una micro computadora (Raspberry Pi), ambos hechos famosos gracias a las comunidades y fundaciones *open-source* (y *open-hardware* en el caso de Arduino) detrás de ellos. Con ideologías de diseño un poco diferentes entre los tres pero que se solapan en muchas aplicaciones en común que se les pueden dar.

1.3.1 Raspberry Pi

Raspberry Pi⁴ es una mini computadora de arquitectura ARM con memoria RAM, CPU, GPU, salidas de video HDMI y RCA, salida de audio, puerto Ethernet, puerto USB y una ranura para tarjetas SD (Ver Figura 1.17). Una configuración muy parecida a la de un teléfono Android de gama baja, pero sin almacenamiento interno.

⁴ <http://www.raspberrypi.org>

RASPBERRY PI MODEL B

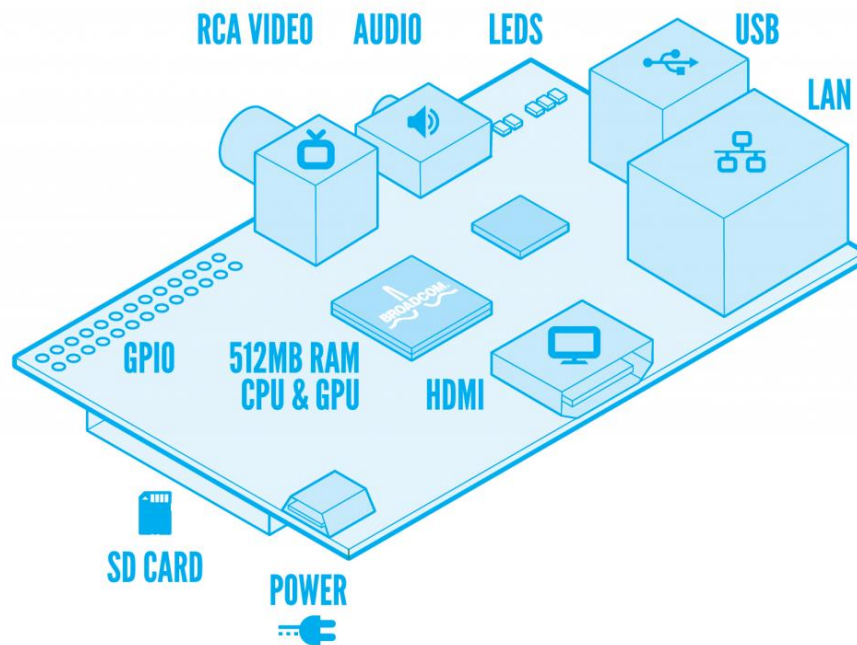


Figura 1.17 Diagrama del Raspberry Pi B.

Soporta un SO, instalado en la SD y basado en Linux, por defecto es *Raspbian*, un SO basado en Debian con el entorno de escritorio LXDE, optimizado para estos dispositivos.

Al ser una mini computadora los proyectos para los que se emplea están enfocados principalmente a software más que hardware, se utilizan como centros multimedia portables para televisores, mini servidores web para diferentes propósitos, proyectos en los que al Raspberry Pi se le conectará teclado, ratón y un dispositivo de salida de video como una computadora corriente. Aunque también permite la posibilidad de ser usado para controlar hardware mediante sus pines GPIO, a los cuales se les pueden conectar otros dispositivos como microcontroladores y demás accesorios de electrónica para ser controlados por el Raspberry Pi mediante librerías[10][11].

Ejemplos de proyectos más enfocados a hardware son adaptaciones de aspiradoras Roomba para convertirlos en robots controlados por control remoto, como el *Tiger Eye*[12], desarrollado por *doitnowlabs*. Para construir robots para ser controlarlos vía web como los *Livebots*[13]. O para construir drones como el *Ziphius*[14], un drone acuático autónomo, con

soporte para ser controlado por control remoto y aplicaciones para diferentes plataformas que permiten utilizarlo en juegos de realidad aumentada.

1.3.2 Arduino

El Arduino⁵ es un microcontrolador, que por si mismo no puede dar soporte a un SO, las funcionalidades que proporciona son solamente aquellas que el desarrollador ha programado en el software que es cargado al dispositivo mediante el gestor de arranque incorporado que posee.

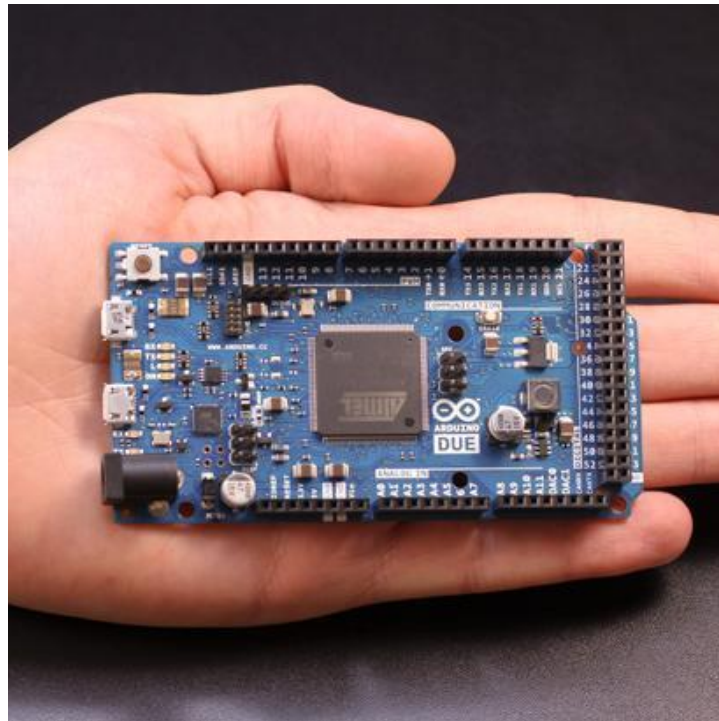


Figura 1.18 Microcontrolador Arduino Modelo Due.

El Arduino (Ver Figura 1.18) está diseñado para leer datos de sensores, procesarlos, y mostrarlos en dispositivos de salida (como pantallas LED y LCD) o enviarlos a otro dispositivo, generalmente de mayor desempeño, para hacer uso de los datos procesados por el Arduino. Según las necesidades del usuario el fabricante ofrece diferentes componentes adicionales (como cases y accesorios) que se pueden agregar para adaptarse a las necesidades del usuario.

Además de contar con una amplia disponibilidad de componentes de hardware, también dispone de su propio IDE, Arduino Software IDE, que utiliza el lenguaje de programación

⁵ www.arduino.cc

Processing⁶ similar a C++, y está basado en Wiring⁷ un IDE y plataforma de prototipado de componentes electrónicos.

A diferencia del Raspberry Pi, su enfoque está en el área de componentes mecatrónicos o de hardware, ya que su función principal es la de ser un controlador de otros dispositivos o subsistemas principalmente mecánicos o electrónicos [10] [11].

Un modelo específico del Arduino que tiene relación directa con este trabajo de investigación es el Arduino Robot⁸ compuesto básicamente por 2 Arduinos, uno para controlar los motores y otro para todo el procesamiento de la información de sensores y decisiones de operación. Para el control de este robot se desarrolló la librería RobotLibrary, incluida en el IDE oficial de Arduino, haciendo a ambos en conjunto, robot y librería, un ejemplo muy completo de desarrollo de robots utilizando software y hardware Arduino. Y también con mucho potencial para ser adaptado y extendido a otros proyectos de robots similares.

Aunque los Arduino y los Raspberry Pi están enfocados para aspectos diferentes del desarrollo de robots pueden ser utilizados en conjunto. Un Arduino se puede integrar en un Raspberry Pi conectándose a través de sus puertos USB utilizando interfaces desarrolladas en Python por la comunidad o conectando el Arduino a los pines GPIO del Raspberry Pi y utilizando la librería de GPIO del mismo. También se puede ir un paso más allá e integrarlos con los bloques Mindstorms de Lego ya que ambos poseen puerto USB, al cual se le pueden conectar dispositivos de WiFi o Bluetooth para interactuar inalámbricamente con los Mindstorms EV3 o NXT.

⁶ www.processing.org/

⁷ wiring.org.co/

⁸ arduino.cc/en/Main/Robot

Capítulo 2 - Herramientas de Desarrollo

Desde el lanzamiento de los primeros Lego Mindstorms existieron un conjunto de herramientas para diseño de robots y desarrollo de software para controlar a los mismos. Con el paso del tiempo, junto a las herramientas oficiales existentes se han sumado un gran número de herramientas desarrolladas por la comunidad. A continuación se listan las herramientas más populares para el diseño de hardware y software.

2.1. Herramientas para Diseño y Ensamblaje

Debido a la gran variedad de piezas de Lego disponibles en el mercado, especialmente las piezas Lego Technic, que hacen virtualmente posible cualquier diseño imaginable en el mundo Lego, se hacen necesarias herramientas para el diseño de los robots.

En esta sección se describen las principales de herramientas enfocadas al ensamblaje y diseño del hardware siguiendo el estilo de herramientas CAD, dentro éstas se pueden utilizar todas las piezas oficiales incluidas con el bloque Mindstorms así como a otros subconjuntos de piezas comercializadas por Lego. Entre otras de las funcionalidades que proveen se encuentran varios modos de visualización, así como también la posibilidad de generar las instrucciones de ensamblaje del robot que se diseñe.

2.1.1 Lego Digital Designer (LDD) (NXT y Lego en general)

Es una herramienta gráfica de diseño para productos de Lego en general. Con la que se pueden diseñar modelos en un ambiente virtual en 3D (Ver Figura 2.1), que luego de terminados el programa puede convertir a planos e instrucciones para su ensamblaje.[15]



Figura 2.1. Visualización del modelo en Lego Digital Designer.

Lego Digital Designer tiene 3 modos de funcionamiento básico:

1. **Modo de Diseño o Construcción (Build mode):** Se disponen de paletas con las piezas disponibles para utilizar en el ensamblado (Ver Figura 2.2), así como un conjunto de herramientas para buscar, rotar, agrupar, alinear, clonar, y cambiar color, tamaño, entre otros atributos de las piezas.

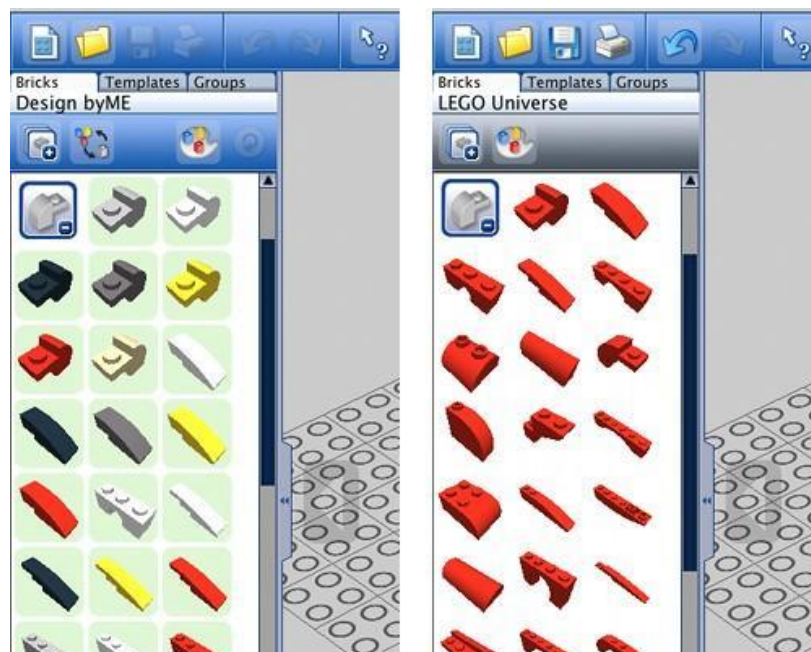


Figura 2.2. Paleta de Piezas/Partes en Lego Digital Designer.

2. **Modo de Visualización (View mode):** En este modo se puede visualizar el modelo, sacar capturas de pantalla, cambiar el fondo de la escena, así como una funcionalidad “explode” que hace que el modelo “explote” mostrando todas las piezas que componen el modelo separadas unas de otras (Ver Figura 2.3).

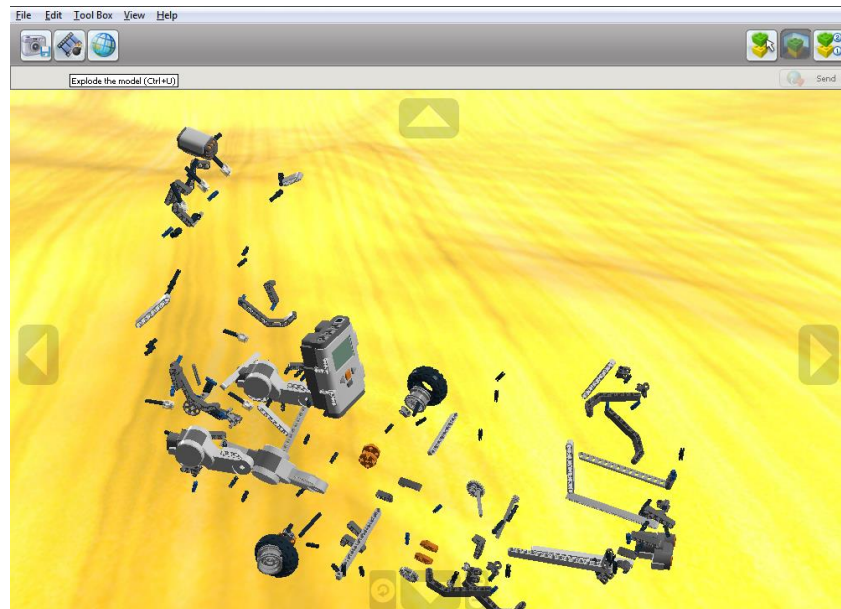


Figura 2.3. Vista *Exploded* en Lego Digital Designer.

3. **Modo de Guía de Ensamblaje (Building guide mode):** Aquí se accede al Reproductor de la Guía de Ensamblaje, el cual reproduce a modo de video los pasos para ensamblar el modelo que ha sido elaborado (Ver Figura 2.4).

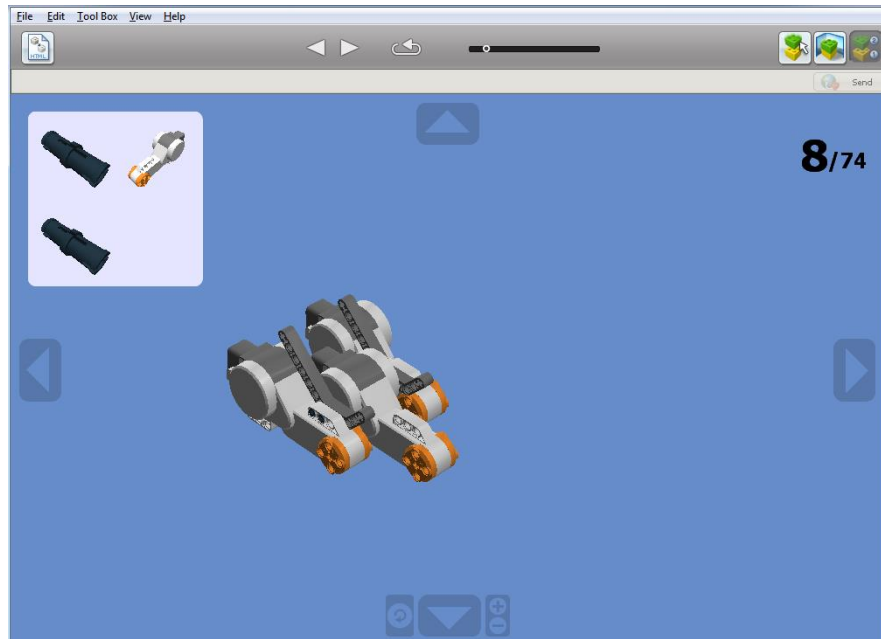


Figura 2.4. Vista *Building Guide Mode* en Lego Digital Designer.

LDD también ofrece la funcionalidad de Importar y Exportar sus modelos en distintos formatos como LXF, LXFML y LDraw.

2.1.2 Autodesk Inventor Publisher (EV3)

Para proveer una experiencia actualizada en el ensamblaje de los nuevos Mindstorms EV3, Lego se asoció con Autodesk para incluir soporte [16] para los EV3 (Ver Figura 2.5) dentro de *Autodesk Inventor Publisher* [17], herramienta de CAD famosa por proveer un ambiente virtual en 3D para la elaboración de casi cualquier tipo de maquinaria y simular su funcionamiento en el mundo real dentro de este mismo ambiente virtual.

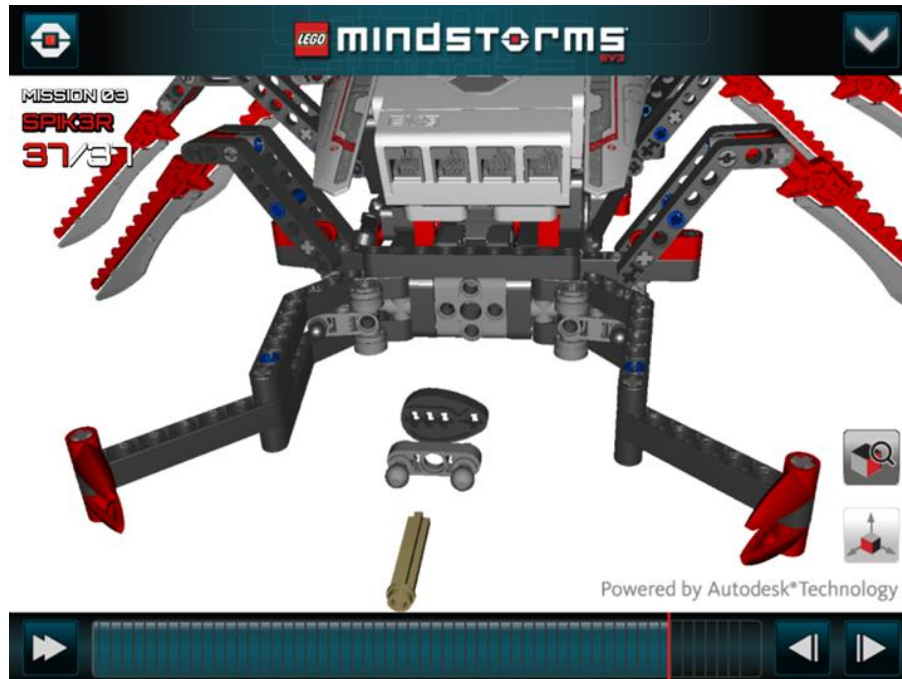


Figura 2.5. Modelo de Mindstorms EV3 en Autodesk Inventor Publisher.

2.2. Herramientas de Programación

Gracias a los diferentes frameworks que se encuentran disponibles, en el apartado de desarrollo de software existe un abanico de alternativas mucho más grande que en el de diseño. Herramientas que pueden utilizarse para interactuar con el Mindstorms y aumentar su potencial de desarrollo, permitiendo que sean incorporados en tareas y sistemas más grandes y complejas.

En esta sección se describen las principales opciones, oficiales y no oficiales, disponibles para diferentes lenguajes de programación. También se contempla la capacidad de conectar el Mindstorms con dispositivos Android mediante el SDK de Android y utilizando leJOS como el *firmware* de los bloques.

2.2.1 LogoBlocks (RCX)

Es una Interfaz Gráfica (o GUI por sus siglas en inglés) que provee los comandos disponibles en el lenguaje de programación Logo en forma de un conjunto de bloques de colores (Ver Figura 2.6), haciéndolo un lenguaje de programación ideal para niños, tanto poderoso como fácil de utilizar.

Estos bloques se pueden arrastrar y soltar dentro de la aplicación, permitiendo encajarlos, como un rompecabezas, para armar secuencias de comandos. Los bloques se encuentran disponibles para el usuario en una paleta o barra de herramientas en la interfaz de

la aplicación, donde se puede ver el conjunto de bloques disponibles así como también las distintas funcionalidades que proveen (como encender, apagar, repetir, accionar motor en reversa, entre otras)[18].

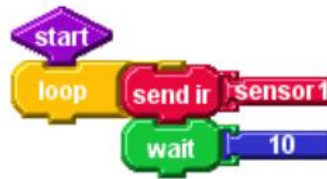


Figura 2.6. Fragmento de Código en LogoBlocks.

2.2.2 NXT-G Programming Environment (NXT)

Esta herramienta se distribuye con el NXT y es la forma más simple para crear la lógica de un robot. Está diseñada para ser utilizada por personas sin conocimientos de programación y consiste en un área de trabajo que posee una línea de acción sobre la cual se van lanzando acciones que el NXT ejecutar en forma secuencial.

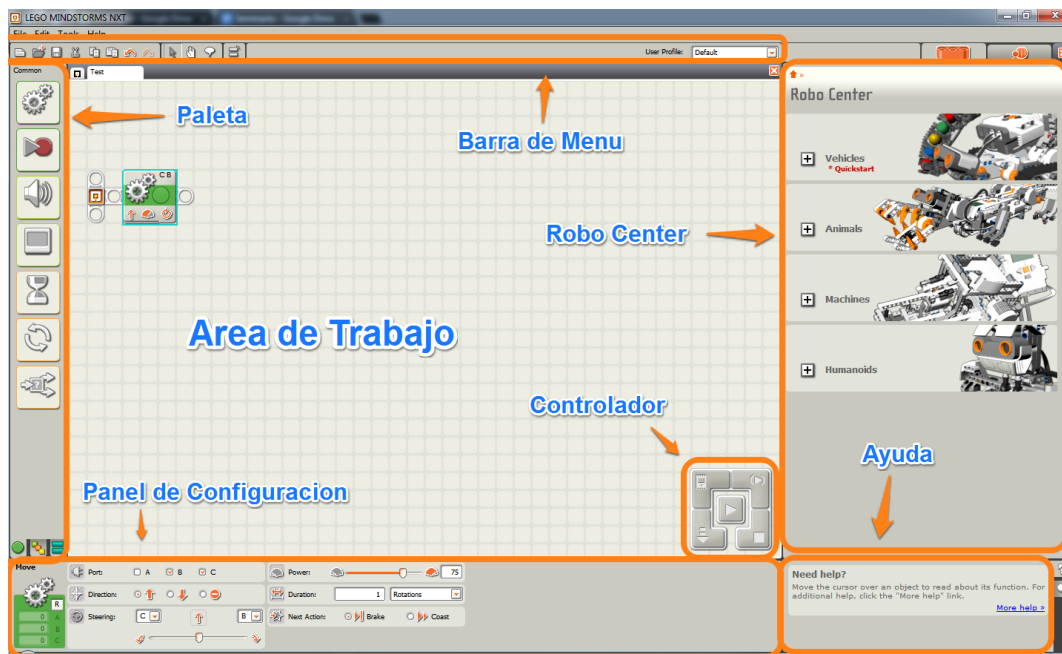


Figura 2.7. Interfaz del IDE NXT-G.

La interfaz se divide en 7 zonas, identificadas en la Figura 2.7:

- i. **Barra de Menú:** Barra estándar con las funcionalidades básicas
- ii. **Robo Center:** En esta área se listan varios ejemplos de robots para ser ensamblados

- iii. **Ayuda:** Un área que proporciona soporte así como consejos en línea sobre el uso de la herramienta
- iv. **Panel de Configuración:** Sirve para establecer los parámetros de las acciones que se van encadenando en el área de trabajo (Ver Figura 3.8)

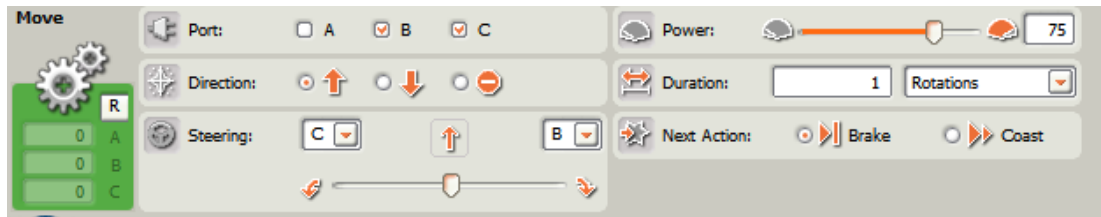


Figura 2.8. Panel de Configuración del IDE NXT-G.

- v. **Paleta:** En esta área se listan los comandos y acciones que puede ejecutar el NXT, como reproducir o grabar sonidos, imprimir información en pantalla, accionar motores, bloques condicionales y bucles
- vi. **Área de Trabajo:** En esta área se va encadenando la logica haciendo *drag and drop* de las acciones de la paleta
- vii. **Controlador:** Sirve para emular las acciones que se han armado en el área de trabajo.

2.2.3 AForge.NET (C# - NXT RCX)

AForge.NET es un *framework* escrito en C# que incorpora funciones que van desde procesamiento de imágenes hasta paquetes de inteligencia artificial.

Este framework, además de las ventajas básicas de C#, mediante sus *namespaces* (o paquetes) provee soporte para las siguientes funcionalidades:

- Logica Difusa
- Algoritmos Genéticos
- Redes Neuronales
- Algoritmos de *Machine Learning*
- Procesamiento de Imágenes (Filtros, texturas, reducción de color)
- Manejar el Microsoft Kinect
- Procesamiento de video

Entre estos *namespaces* se encuentra uno que incorpora soporte para los bloques NXT y RCX, aunque desde la version 2.2.5.0 el soporte para RCX fue desincorporado. Este

framework es de código abierto y se encuentra disponible bajo la licencia *LGPL (GNU Lesser General Public License)* y está bajo constante desarrollo y mantenimiento.

2.2.4 - nxtOSEK (ANSI C/C++ - NXT)

Es una plataforma *open-source* para Lego Mindstorms NXT, está integrada por el *driver* (o manejador) de dispositivos de leJOS NXJ, hecho en C/Ensamblador, TOPPERS/ATK (Automotive Kernel), TOPPERS/JSP (Sistema Operativo en Tiempo Real que incluye específicamente partes portadas para la arquitectura ARM7 del NXT y código adicional para integrar todas estas partes hacerlas funcionar en conjunto[19].

Entre las funcionalidades que ofrece se encuentran:

- Ambiente de Programación ANSI C/C++ mediante el uso del compilador GCC
- API en C/C++ para los sensores (incluidos algunos no oficiales en C++), motores y otros dispositivos del NXT
- Provee multitarea en tiempo real de TOPPERS/ATK, probadas en la industria automotriz
- Provee multitarea en tiempo real de TOPPERS/JSP, que cumple con la certificación japonesa original de la especificación RTOS “ μ ITRON 4.0”
- Rápida ejecución y menor consumo de memoria (el programa nxtOSEK es ejecutado nativamente en el procesador ARM7 y consume alrededor de 10 KBytes)
- Múltiples maneras de cargar la aplicación en nxtOSEK al bloque NXT

2.2.5 nxt-python (Python - NXT)

Es un paquete para controlar bloques NXT en el lenguaje de programación Python principalmente mediante Bluetooth, utilizando comandos del Lego *Bluetooth Development Kit* para el bloque NXT[20]. Provee soporte para muchos sensores de terceros, tiene licencia GPLv3 y está basado en el paquete *nxt_python*, escrito por Douglas P. Lau, cuyo desarrollo fue detenido y abandonado en 2007[21].

Su versión 2, mediante un conjunto de paquetes adicionales proporciona la opción de controlar el bloque mediante USB como una alternativa a Bluetooth. La versión 1 puede ser utilizada aún, pero la versión 2 no es compatible hacia atrás con ésta. Entre las mejoras que presenta la versión 2 se encuentran [20]:

- Nueva interfaz para sensores, con mejoras al módulo del sensor y al framework para sensores digitales.

- Detección automática de tipos de sensores y creación del objeto de sensor correcto.
- Nueva interfaz para motores, con clase de motor sincronizada y soporte para frenado.
- Soporte para 24 sensores diferentes, incluyendo 8 de HiTechnic y 11 de MindSensors.

2.2.6 hackage-NXT (Haskell- NXT)

hackage-NXT es una interfaz en Haskell para Lego Mindstorms NXT a través de Bluetooth. Con soporte para comandos directos, mensajes, sensores (incluidos no oficiales) y para control remoto del bloque NXT mediante mensajes simples [22].

Está compuesta básicamente por 3 programas principales, *nxt-upload* para hacer la carga de archivos al bloque, *nxt-status* para mostrar su estado y *nxt-shutdown* que permite apagarlo remotamente.

2.2.7 leJOS NXJ (Java - NXT RCX)

leJOS NXJ es un API para desarrollar programas para los bloques en lenguaje Java. Estos programas corren sobre la máquina virtual leJOS JVM, escrita en C. leJOS permite crear programas más complejos que NXT-G. Entre las funcionalidades que provee leJOS se encuentran [23]:

- Programación orientada a Objetos
- Hilos de ejecución
- Arreglos multidimensionales
- Recursión
- Sincronización
- Excepciones
- Todos los tipos de datos de Java
- La mayoría de las funciones de los paquetes `java.lang`, `java.util` y `java.io` classes
- Documentación del API
- Paquetes para manejar Sensores y Motores
- PC API

Al ser un API para Java, leJOS permite desarrollar desde cualquier sistema operativo, aprovechándose de la cualidad multiplataforma de este lenguaje. Y no es limitativo para el uso

en Mindstorms, ya que también se puede desarrollar en leJOS para GameBoy (excluyendo los elementos del paquete NXJ).

Aparte de su API principal también posee un PC API, el cual permite la ejecución de un gran subconjunto de instrucciones del API principal, como mover los motores o reproducir sonidos en el bloque, desde otros dispositivos con Java mediante el uso de la tecnología Bluetooth.

Es llamado PC API debido a que inicialmente estaba enfocado para PCs, pero debido a la expansión de los teléfonos inteligentes, estos también se ven incluidos dentro de su alcance, siendo los dispositivos Android los más usados para su integración con bloques Mindstorms debido a que también funcionan sobre Java.

Para hacer que un dispositivo Android y un bloque Mindstorms interactúen entre ellos se puede utilizar el PC API de leJOS[24], que utiliza comunicación mediante Bluetooth, desde Android, esto se puede lograr de dos maneras:

- Comunicándose con programas hechos en leJOS corriendo en el Mindstorms. Esto requiere que el firmware de leJOS se encuentre instalado en el bloque
- Comunicándose con el bloque mediante el Protocolo de Comunicación de LEGO (LCP por sus siglas en inglés). Esto no requiere tener el firmware de leJOS en el bloque

La única limitación de utilizar estas librerías es que no proporcionan soporte para el paquete *robotics* del API de leJOS, lo cual impide la manipulación directa de motores o sensores desde la aplicación de Android, pero que puede ser solucionado utilizando técnicas para controlar el bloque, como paso de mensajes, para enviar órdenes y obtener información de dichos componentes.

2.2.8 Android SDK (Dispositivos Android)

El Software Development Kit (SDK)[25] de Android comprende un conjunto de librerías, documentación, herramientas y APIs necesarias para desarrollar, compilar, probar y depurar aplicaciones para el Sistema Operativo Android.

Oficialmente el *Integrated Development Environment* (IDE) soportado por Google para el desarrollo de aplicaciones Android es *Eclipse* con el plugin *Android Development Tools* (ADT), también se encuentra en fase de *Early Access Preview* una variante del IDE IntelliJ IDEA, llamado *Android Studio*, que será directamente mantenido por Google.

Aunque no es obligatorio el uso de un IDE para desarrollar aplicaciones para Android, se puede utilizar cualquier editor de texto para desarrollar y una interfaz de línea de comandos

para compilar. Estos IDEs proveen soporte para herramientas que facilitan tanto la estructuración de los proyectos como su correcta compilación y mantenimiento.

Entre dichas herramientas se encuentran:

- *Apache Ant*, que permite automatizar la compilación y mantenimiento, similar a *make* para C/C++,
- *Apache Maven*, al igual que *Ant*, permite automatizar tareas de compilación y mantenimiento, de una manera más simple ya que su configuración es almacenada en archivos XML, está enfocado a ciclos de vida del proceso de compilación e incluye un *Project Object Model* (POM) en el cual se describe el proyecto, sus dependencias, componentes externos e incluso el orden de compilación. Está diseñado para funcionar en red y puede administrar repositorios externos, tanto para descargar dependencias u otros elementos como para subir el resultado final de la compilación a repositorios preestablecidos.
- *Gradle*, proporciona la misma funcionalidad de las dos herramientas previas, pero utilizando archivos DSL basados en Groovy en vez de XML y utiliza un Grafo Acíclico Dirigido (DAG por sus siglas en inglés) en vez de ciclos de vida, como *Gradle*, u ordenado parcial dependiente de otros elementos de compilación, como *Ant*, para determinar el orden en que las tareas deben ser ejecutadas. Soporta compilación de multi-proyectos así como también compilación incremental, determinando cuales compilaciones de componentes están al día para evitar ejecutar las tareas que afectan a esos componentes.

Otras herramientas incluidas en el SDK son *Fastboot*, utilizada para modificar el contenido de la memoria Flash de los dispositivos mediante USB, empleando la línea de comandos y *Android Debug Bridge* (ADB) que le permite a PC y dispositivos Android comunicarse mediante línea de comandos, muy parecido a como funciona un *Secure Shell* (SSH).

2.2.9 OpenCV

OpenCV⁹ (*Open Source Computer Vision Library*) es una librería especializada en visión computacional, está escrita en C/C++ y ofrece versiones para C++, C, Python y Java. Gracias a esta variedad de lenguajes, puede ser utilizado en aplicaciones para Windows, Linux, Mac OS, iOS y Android. Dado que se publica bajo la licencia BSD, puede ser utilizada tanto para fines comerciales como para fines académicos.

Entre sus principales funcionalidades se encuentran:

- Procesamiento de imágenes
- Procesamiento de videos
- Diseño de GUIs
- Reconstrucción de objetos 3D
- Detección de objetos
- Machine Learning

OpenCV fue diseñada teniendo en cuenta el procesamiento en tiempo real lo que hace que sea altamente eficiente.

2.2.10 VTK

VTK¹⁰ (*Visualization Toolkit*) es una librería multiplataforma para el procesamiento de imágenes y visualización 3D, escrita en C++, con interfaces para Java y Python.

Entre los principales algoritmos que implementa esta librería se encuentran:

- Algoritmos de visualización, incluyendo escalares, vectoriales y de texturas
- Métodos volumétricos
- Algoritmos de modelado avanzado, como modelado implícito, reducción de polígonos y triangulación de Delaunay.

También posee un extenso framework de visualización, widgets 3D, soporte para procesamiento en paralelo, integración con librerías como Qt y Tk.

Al igual que OpenCV, VTK se distribuye con la licencia BSD lo que permite su uso en cualquier tipo de proyectos.

⁹ <http://opencv.org/>

¹⁰ <http://www.vtk.org/>

2.2.11 CGAL

CGAL¹¹ (*Computational Geometry Algorithms Library*) es una librería compuesta por un conjunto de algoritmos de geometría, implementados en C++, estos algoritmos han sido implementados lo más eficientemente posible para hacer que esta librería sea robusta y confiable. Las principales funcionalidades de de CGAL son:

- Estructuras de datos geométricas
- Algoritmos de búsqueda
- Algoritmos de ordenamiento espacial

CGAL es distribuida con un esquema de doble licencia. Puede ser utilizada con software *open source* libre de cargos gracias a que su código base está disponible bajo licencia LGPL y los niveles superiores de la librería bajo GPL. O de manera comercial, en aquellos casos en que el uso que se le vaya a dar se encuentre limitado por las licencias LGPL y GPL, adquiriendo una licencia comercial en GeometryFactory¹².

¹¹ <http://www.cgal.org/>

¹² <http://geometryfactory.com/>

Capítulo 3 - Propuesta de desarrollo

En este capítulo se presenta la propuesta de desarrollo, elegida luego de analizar y descartar un número importante de propuestas de desarrollo de robots utilizando Lego Mindstorms, haciendo uso de diferentes componentes y cubriendo diferentes campos de trabajo.

Gracias a la capacidad del bloque NXT para comunicarse por Bluetooth, es posible emparejarse con otros dispositivos para conformar sistemas distribuidos, con diferentes dispositivos ocupándose de tareas diferentes, un ejemplo de esto, para procesamiento de imágenes, se pueden capturar imágenes utilizando un teléfono celular, las cuales serían transmitidas al bloque Mindstorms para hacer procesamiento de imágenes como detección de bordes, identificación de patrones, entre otros.

3.1 Motivación

En los últimos años las impresoras 3D han logrado gran popularidad dada la capacidad de generar prácticamente cualquier objeto a partir de un modelo generado por computadora, estos objetos pueden ser tan simples como un cubo o tan complejos como un brazo mecánico. En sitios web como *Thingiverse*¹³ y *3D Printing Model Marketplace*¹⁴, se pueden conseguir un gran número de modelos que en cuestión de minutos se pueden imprimir y utilizar.

Para generar estos modelos a partir de objetos cotidianos es necesario disponer de un escáner 3D y un ambiente controlado donde hacer el escaneo de los objetos, por lo que surge la idea de desarrollar una propuesta de escáner 3D móvil, con componentes de bajo coste disponibles en el mercado.

Éste, con una calidad del modelo final menor que la de sus contrapartes profesionales y estacionarios, no estaría enfocado al sector profesional, sino al mercado joven y de *power users*¹⁵, un target similar al del robot *Ziphius*[14] mencionado previamente en la sección dedicada a los *Raspberry Pi*.

¹³ <http://www.thingiverse.com>

¹⁴ <http://3dprintingmodel.com>

¹⁵ Usuario final de tecnologías de computación con un conocimiento considerablemente mayor que el usuario final común

Al ser móvil haría relativamente sencillo al usuario escanear objetos del mundo real, siendo sólo necesario tener consigo la guía, el robot y un smartphone/tablet Android para controlarlo. Haciendo una sesión de escaneo con el robot comparable a una sesión sencilla de fotos semi-profesional, en cuanto a que el robot es el equivalente a las cámaras y accesorios que un fotógrafo debe cargar consigo y que requiere menos experticia en su uso que una cámara fotográfica réflex.

Una vez obtenidos los modelos generados por el robot, así como para imprimirlos con una impresora 3D, podrían ser utilizados para otras aplicaciones, por ejemplo en software para modelos de videojuegos, renders para material audiovisual y diseño gráfico en general. También, en una siguiente fase de desarrollo de esta propuesta, se podría implementar un repositorio público de modelos escaneados por los usuarios, implementando una opción para compartir el contenido escaneado dentro de la aplicación, siguiendo modelos como el de DeviantArt¹⁶, con un enfoque más específico, estableciendo las bases para el desarrollo de una comunidad alrededor del escaneo de modelos con estos dispositivos.

Actualmente los modelos para imprimir están limitados a aquellos que se consigan en sitios web como los mencionados previamente o de tomarse el tiempo de diseñar el objeto usando un programa tipo CAD o un modelador 3D, para esto se necesita tener conocimiento y dominio de estas herramientas. Esto hace que surja la necesidad de tener un escáner para objetos 3D que sea fácil de usar y que requiera del mínimo de conocimientos de estas herramientas por parte de los usuarios.

3.2 Objetivo General

Diseñar e implementar un escáner 3D basado en un robot y cámaras para capturar diferentes ángulos del objeto 3D y finalmente desplegar el modelo generado.

3.3 Objetivos Específicos

- Diseñar un robot que se encargue de trasladarse alrededor de un objeto designado para capturar todos los lados posibles del objeto
- Implementar un esquema de calibración del robot y las cámaras para reducir el error generado por el desplazamiento del robot

¹⁶ <http://www.deviantart.com/>

- Implementar una aplicación Android que capture imágenes bajo demanda y realice el pre-procesamiento necesario a las mismas para los pasos previos a la creación del modelo 3D
- Implementar una aplicación Android que se comuniquen con el robot y con las cámaras para coordinar el proceso de captura de imágenes, que procese dichas imágenes para generar un modelo 3D del objeto y presente este como una visualización que el usuario pueda manipular mediante la aplicación.

3.4 Propuesta de Solución

3.4.1 Esquema propuesto

Se propone crear un sistema compuesto por 3 dispositivos Android y un bloque Mindstorms. De los 3 dispositivos Android 2 serán teléfonos, para ser utilizados como cámaras y para procesamiento de imágenes, y el tercero una tablet, que cumplirá las funciones de controlador del resto de los dispositivos, así como para generar una nube de puntos en base a imágenes capturadas para crear el modelo 3D final.

El controlador se comunicará mediante una conexión Bluetooth con el bloque y mediante WiFi con las cámaras como se muestra en la Figura 3.1.

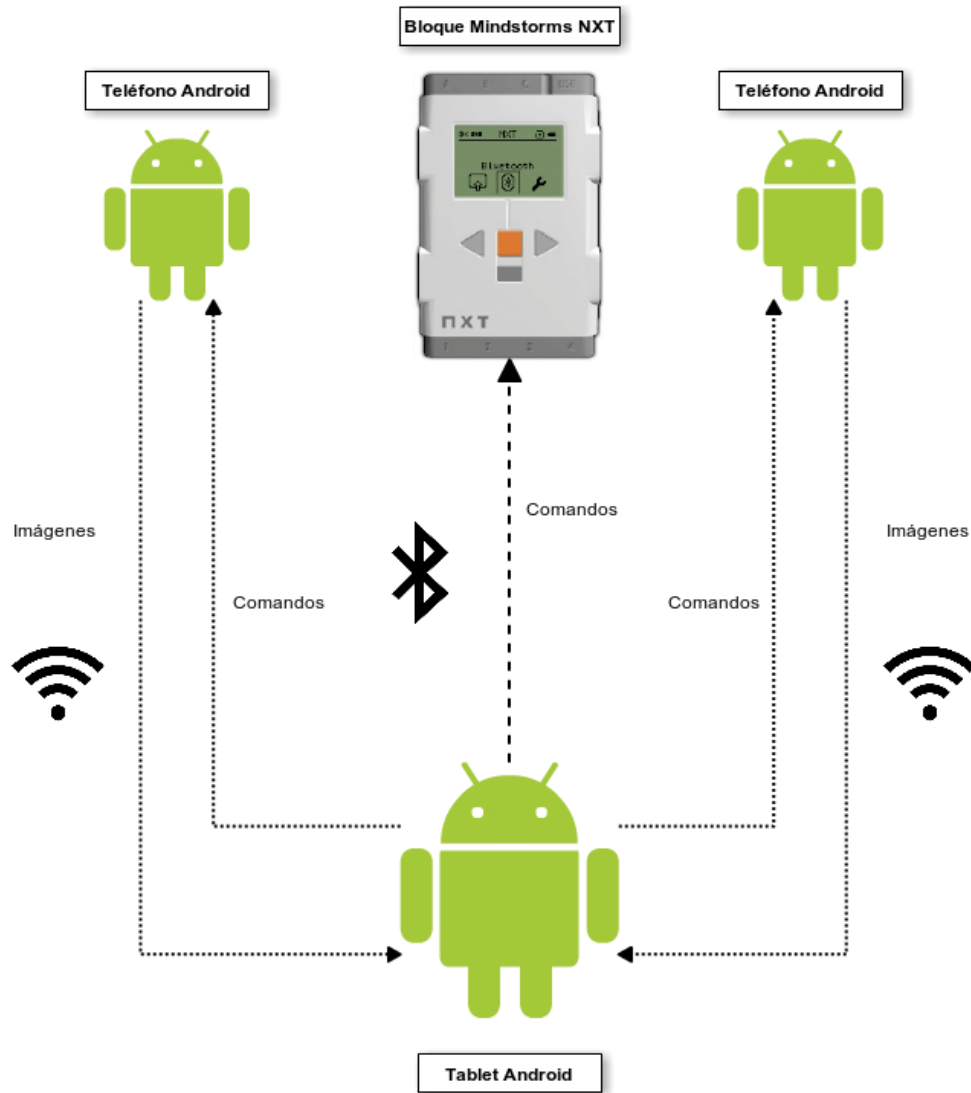


Figura 3.1. Esquema Propuesto.

El controlador permitirá al usuario controlar el movimiento del robot, una vez que se ubique el objeto a escanear, este se deberá posicionar sobre una guía especial para el robot. La misma debe ser desplegada en una superficie plana sobre la que el robot se pueda desplazar.

Esta guía está compuesta por una serie de círculos concéntricos con marcas especiales que el robot seguirá para mantenerse a una distancia uniforme del objeto a escanear mientras se desplaza alrededor del mismo, así como para determinar la distancia óptima desde la cual realizar la captura de las imágenes.

Cuando llegue a un punto de control notificará a la tablet, la cual ordenará a las cámaras iniciar la captura de las imágenes, una vez completada la captura de imágenes por las cámaras, éstas notificarán de nuevo a la tablet, la cual ordenará al robot iniciar su movimiento hacia el siguiente punto de control.

Cuando el robot termine de recorrer la guía, las imágenes capturadas por cada cámara serán preprocesadas previamente a ser enviadas a la tablet, para mejorar la calidad de las mismas y limpiar todos los elementos que no sean parte del objeto a escanear mediante un conjunto de algoritmos diseñados con este fin, como el GrabCut. Una vez en la tablet ésta se encargará de procesar las imágenes para generar la representación 3D del objeto escaneado.

En las siguientes secciones se describen en detalle los componentes de hardware y software necesarios para esta propuesta.

3.4.2 Hardware

En esta propuesta se utilizará un Lego Mindstorms NXT para crear el robot, para las cámaras se utilizarán 2 teléfonos con sistema operativo Android, y como controlador de todo el escáner se utilizará una tableta igualmente con el sistema operativo Android.

3.4.3 Software

Para esta propuesta se deben desarrollar 3 aplicaciones, una que se ejecuta en el NXT, otra para los dispositivos que se utilizarán como cámaras y una que coordine la interacción entre todos los dispositivos que conforman el sistema así como de encargarse de generar el modelo 3D final y desplegarlo en pantalla para el usuario, que se ejecutará en la tablet.

1. Controlador del NXT

Una aplicación escrita en Java con leJOS que controle el movimiento del robot, para mantener al mismo a una distancia constante del objeto y moverse alrededor de este, trazando una circunferencia, para que las cámaras puedan capturar las imágenes necesarias.

2. Controlador de la Cámara

Una aplicación para Android que se encargue de capturar imágenes bajo demanda, que las procese para eliminar todos los elementos que no sean relevantes ni sean parte del objeto a escanear, así como para aumentar la calidad de las mismas.

3. *Controlador General*

Una aplicación para Android que funcione como control de todo el escáner. Esta aplicación se encargaría de controlar el desplazamiento del robot, mediante Bluetooth, hasta el próximo punto de control dentro de la guía. Así como también ordenaría a las cámaras iniciar el proceso de captura y transmisión de imágenes una vez que el robot esté en posición. Y, una vez que se hayan recorrido todos los puntos de control, procesará las imágenes capturadas para generar el objeto 3D, que finalmente desplegará en pantalla para ser manipulado por el usuario.

3.4.4 Herramientas a Utilizar

Para el desarrollo de estas aplicaciones se utilizarán diferentes herramientas.

Para APIs, SDKs y frameworks:

- Android SDK
- leJOS JVM (Java Virtual Machine) y NXJ API

Interfaces de Desarrollo:

- Android Studio
- Eclipse

Sistemas de Manejo de Versiones:

- GIT

3.4.5 Prototipos

Robot Mindstorms

El robot debe ser capaz de llevar ambas cámaras sujetadas firmemente sobre él, así como tener la capacidad de mover el brazo con los sensores. En la Figura 3.2 se muestra un prototipo para el robot, en el cual se incorporan 2 sensores de luz, que utiliza para seguir la guía, y un sensor ultrasónico, para estimar la distancia con respecto al objeto a escanear.

Los sensores de luz están fijados al brazo, lo cual permite que el robot pueda ajustarse a la cuerda más cercana en la circunferencia que sigue dentro de la guía. Por encima de este brazo se encuentra la base para las cámaras. Para minimizar el número de partes móviles del robot, las cámaras siempre están orientadas hacia el frente del mismo.

De los tres motores de los que dispone el robot, se utilizan dos para el desplazamiento del mismo y el tercero para mover el brazo con los sensores. Este último tiene la limitante de que no puede hacer un recorrido de 360° , cada sensor tiene un rango de 90° para efectuar las mediciones necesarias.

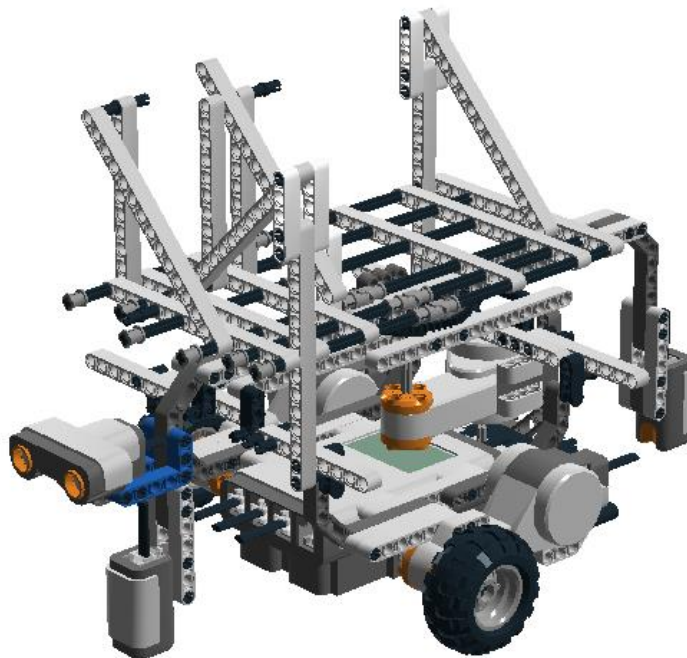


Figura 3.2. Prototipo robot Mindstorms.

Aplicación del Controlador General

La aplicación del Controlador General contará con dos vistas, la principal que mostrará la vista de una de las cámaras del robot, así como botones para controlar el movimiento del mismo, para iniciar el escaneo y para abandonar la aplicación (Ver Figura 3.3).

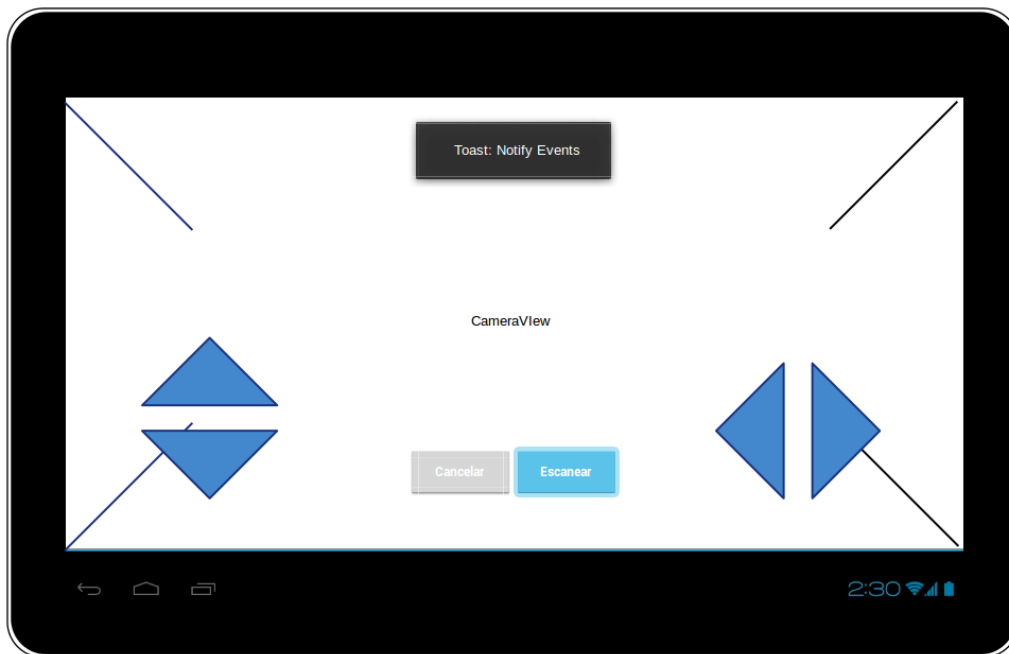


Figura 3.3. Vista principal aplicación tablet Android.

La vista secundaria de la aplicación presentará un diálogo bloqueante mientras se esté realizando el proceso de escaneo en el punto de control, este proporcionará las opciones de cancelar el proceso junto con el poder iniciar el visualizador de modelos 3D mientras se espera que el proceso se complete (Ver Figura 3.4).

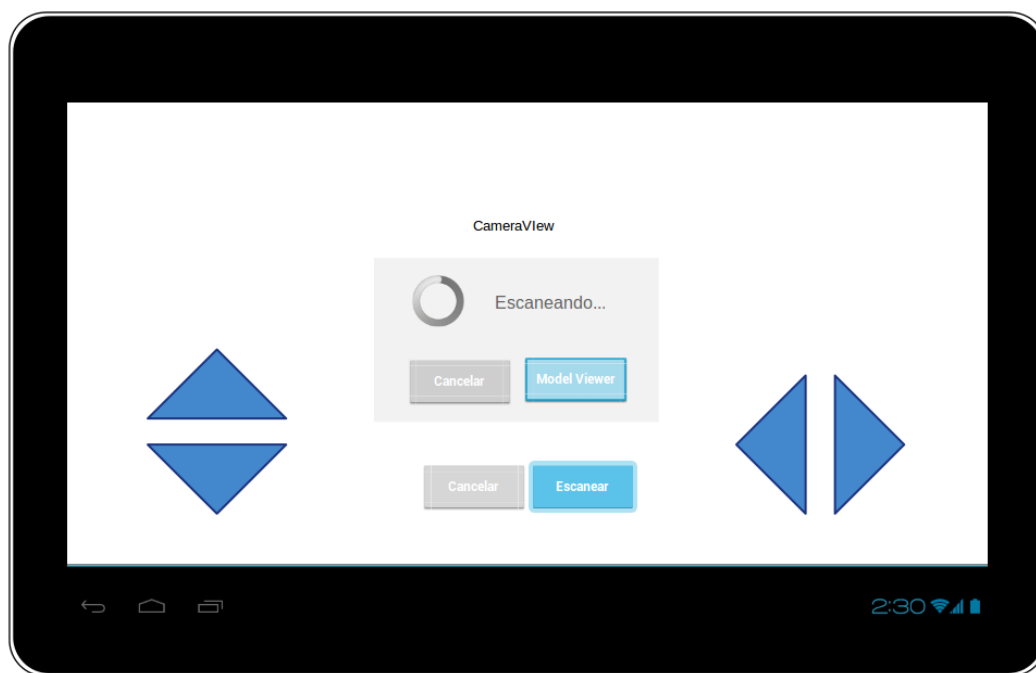


Figura 3.4. Vista alterna “Escaneando” aplicación tablet Android.

Aplicación del Controlador de las Cámaras

La aplicación para los teléfonos que serán usados como cámaras contará con una interfaz simple que refleje el estado en el que se encuentren en el proceso de escaneo (Ver Figura 3.5).

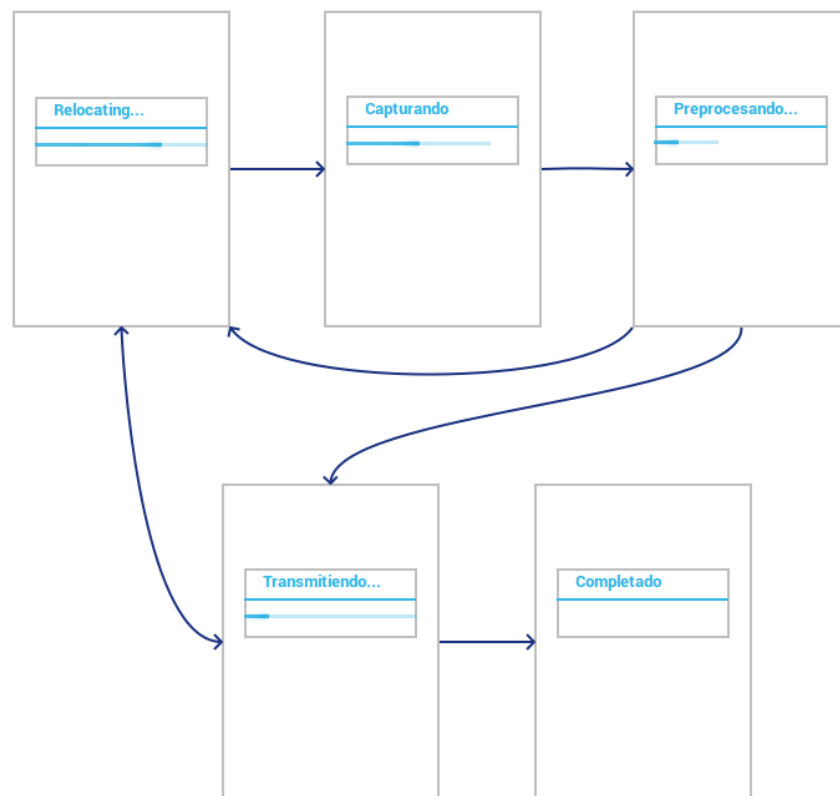


Figura 3.5. Vistas aplicación teléfonos Android.

Guía

La guía, que se requerirá colocar en la superficie sobre la que será escaneado el objeto, estará compuesta por una serie de círculos concéntricos, con un conjunto de puntos de control ubicados a intervalos de entre 5 a 30 grados aproximadamente (Ver Figura 3.6). Esta guía permitirá al robot mantener un desplazamiento uniforme con respecto al objeto a escanear, para incrementar la exactitud del modelo generado.

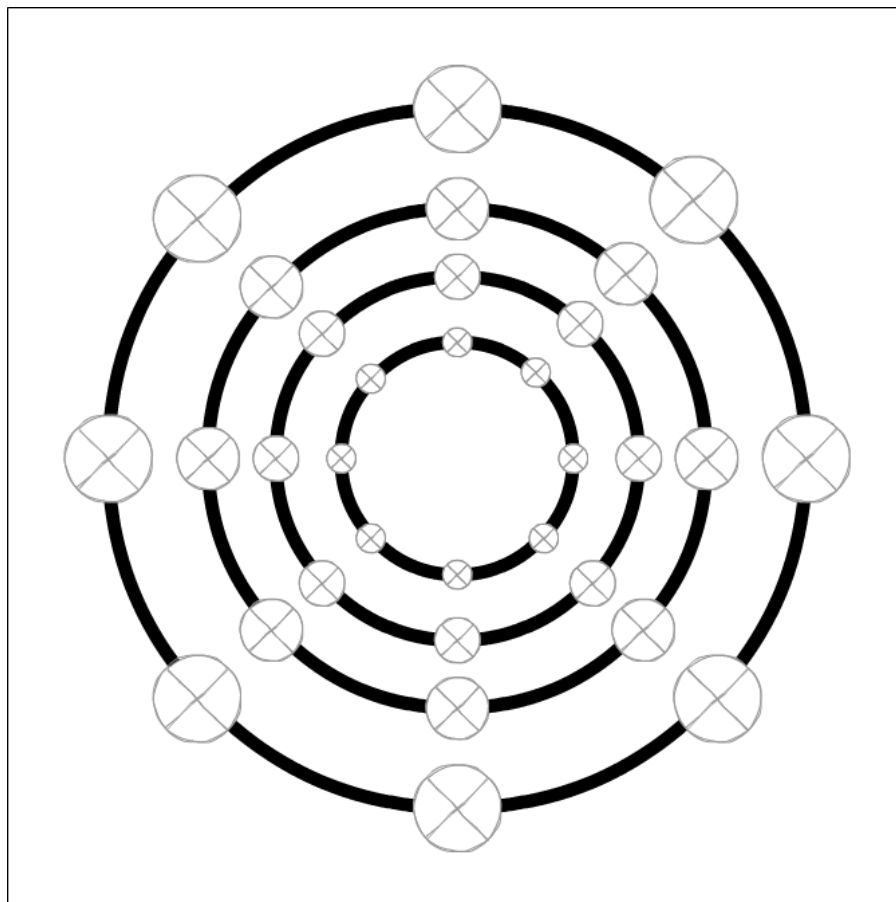


Figura 3.6. Guía con puntos de control para diferentes distancias para el robot.

Referencias

1. David Mindell (2000). *LEGO Mindstorms The Structure of an Engineering (R)evolution*. Massachusetts Institute of Technology (MIT). Recuperado de:
<http://web.mit.edu/6.933/www/Fall2000/LegoMindstorms.pdf>
2. *About the LOGO Foundation*. The Logo Foundation. Consultado en Junio, 2013. Disponible en
<http://el.media.mit.edu/logo-foundation/about/index.html>
3. *LHTC - Low Threshold High Ceiling*. Mind, Matter & Media Lab. Universidad Vanderbilt. Consultado en Junio, 2013. Disponible en <http://www.m3lab.org/research/lthc>
4. *Lego Mindstorms RCX Kit and Add-ons*. RJMcNamara. Consultado en Junio, 2013. Disponible en
<http://www.rjmcnamara.com/lego-mindstorms-rcx/>
5. Bagnall, B. (2007). *Maximum Lego NXT: building robots with Java brains, Meet NXT (1-5)*. Canada: Variant Press.
6. *Brickpicker - LEGO 8547-1: Mindstorms NXT 2.0*. Brickpicker Lego Price and Investment Guide. Consultado en Julio, 2013. Disponible en <http://bit.ly/HyRLG6>
7. *Brickpicker - LEGO 9747-1: Mindstorms RIS*. Brickpicker Lego Price and Investment Guide. Consultado en Julio, 2013. Disponible en <http://bit.ly/HyRR0c>
8. *Lego Mindstorms EV3 – A new Generation!*. Robot Square. Laurens Valk. Consultado en Junio, 2013. Disponible en <http://bit.ly/HyPfQ8>
9. *Afrel - LEGO Mindstorms Education EV3 Intelligent Brick*. Afrel Co. Ltd. Consultado en Junio, 2013. Disponible en <http://www.ev-3.net/en/archives/850>
10. *Arduino vs Raspberry Pi Comparison*. Codeduino - All Things Microcontroller. Consultado en Octubre, 2013. Disponible en <http://bit.ly/HyRXFI>
11. *Arduino vs Raspberry Pi: Which Is The Mini Computer For You?* makeUseOf. Consultado en Octubre, 2013. Disponible en <http://bit.ly/HySK9e>
12. *Guest Blog #1: Repurposing a Rooba by Ben J*. The Raspberry Pi Foundation. Consultado en Octubre, 2013. Disponible en <http://www.raspberrypi.org/archives/2348>
13. *Livebots*. The Raspberry Pi Foundation. Consultado en Octubre, 2013. Disponible en
<http://www.raspberrypi.org/archives/3493>
14. *Ziphius: The Aquatic Drone*. Kickstarter Inc. Consultado en Octubre, 2013. Disponible en
<http://kck.st/1hbS58A>
15. *Lego.com Digital Designer Virtual Building Software - Support*. Lego.com. The LEGO Group. Consultado en Junio, 2013. Disponible en <http://ldd.lego.com/en/support/>

16. *Autodesk partners with Lego to create 3D Interactive Mindstorm EV3 Instructions*. Autodesk Latinoamérica. Autodesk Inc. Consultado en Junio, 2013. Disponible en <http://autode.sk/HyPIBF>
17. *Autodesk Inventor Publisher*. Autodesk Latinoamérica. Autodesk Inc. Consultado en Junio, 2013. Disponible en <http://autode.sk/HyPLxy>
18. Andrew Begel (1996). *LogoBlocks: A Graphical Programming Language for Interacting with The World* (Epistemology and Learning Group, MIT Media Laboratory). Recuperado de: <http://research.microsoft.com/en-us/um/people/abegel/mit/begel-aup.pdf>
19. *What is nxtOSEK?*. Sourceforge.net. Takashi Chikamasa. Consultado en Junio, 2013. Disponible en <http://lejos-osek.sourceforge.net/whatislejososek.htm>
20. *nxt-python 2.2.2 : Python Package Index*. Python Software Foundation. Consultado en Junio, 2013. Disponible en <https://pypi.python.org/pypi/nxt-python/2.2.2>
21. *nxt-python - A pure python driver/interface/wrapper for the Lego Mindstorms NXT robot*. Google Developers. Consultado en Junio, 2013. Disponible en <https://code.google.com/p/nxt-python/>
22. *HackageDB: NXT 0.2.3*. Industrial Hackage Group (IHG). Consultado en Junio, 2013. Disponible en <http://hackage.haskell.org/package/NXT>
23. *LeJOS, Java for Lego Mindstorms*. Sourceforge. Consultado en Julio, 2013. Disponible en. <http://lejos.sourceforge.net/nxj.php>
24. *Using leJOS with Android*. LEJOS. Java for LEGO Mindstorms. Sourceforge. Consultado en Octubre, 2013. Disponible en <http://www.lejos.org/nxt/nxj/tutorial/Android/Android.htm#3.4>
25. *Android SDK*. Android Developers. Consultado en Octubre, 2013. Disponible en <http://developer.android.com/sdk/index.html>