

# Programación del robot lego mindstorms mediante software de análisis numérico para ingenierías

*Programming of the Lego Mindstorms robot by means of numerical computing software for engineering*

*Jimmy Alexander Cortés Osorio\* Francisco Alejandro Medina Aguirre\*\* Jairo Alberto Mendoza Vargas\*\*\**

Presentado: 20 de febrero de 2010 Aprobado: 3 de mayo de 2010

## Resumen

**Introducción:** el siguiente artículo presenta, de manera abreviada, los diferentes componentes del Robot Lego Mindstorms NXT (en cuanto a su hardware). Este robot es utilizado en la enseñanza de la robótica y la programación en muchas Universidades de America y Europa. **Metodología:** tipo descriptivo, y evaluativo ya que se pretende mostrar las características que tiene el robot Lego Mindstorms NXT para la enseñanza de la robótica y evaluar el software de análisis numérico Matlab con algunos de sus toolboxes para la programación del robot lego como alternativa científica. **Resultados:** este artículo muestra como se puede programar mediante el software de análisis numérico Matlab los diferentes componentes del robot NXT (cerebro, servomotores, sensores de luz, tacto, ultrasonido y sonido). La programación del robot en Matlab le brinda la posibilidad al usuario de utilizar otros toolboxes que vienen con el lenguaje de programación dentro de los que se encuentran la visión artificial, la adquisición de datos y las redes neuronales entre otros. **Conclusiones:** de los diferentes robots educativos evaluados en la investigación, se llegó a la conclusión de que el Robot Lego NXT por su facilidad de armado y por los componentes de hardware que lo conforman, son una muy buena herramienta para la enseñanza de la robótica y la programación

**Palabras clave:** lego Mindstorms NXT, Matlab, sensor, cerebro, servomotor, robótica, programación, *toolbox*.

## Abstract

**Introduction:** this paper presents, in abbreviated form, the different components of the Lego Mindstorms NXT Robot (in terms of hardware), this robot is used in the teaching of robotics and programming in many universities in America and Europe. **Methodology:** descriptive and evaluative as it tries to show the characteristics that have the Mindstorms Lego nxt robot for teaching robotics and evaluate numerical analysis software Matlab with some of its toolboxes for programming the Lego robot as a scientific alternative. **Results:** this article shows how software can be programmed using the Matlab numerical analysis of different components of the NXT robot (brain, servo motors, light sensors, touch, ultrasonic and sound). The programming of the robot in Matlab gives the user the possibility of using other toolboxes that come with the programming language within which are machine vision, data acquisition and neural networks among others. **Conclusions:** of the different educational robots tested in the investigation concluded that the Lego NXT Robot for ease of assembly and hardware components that comprise it, are a very good tool for teaching robotics and programming

**Keywords:** Lego Mindstorms NXT, Matlab, sensor, brick, servo motor, robotics, programming, toolbox.

\* Docente de la Universidad Tecnológica de Pereira, Ingeniero Electricista, Magíster en Instrumentación Física, miembro fundador del grupo de investigación Robótica Aplicada reconocido por Colciencias y director del semillero en Robótica del programa Mecatrónica de la Universidad Tecnológica de Pereira. Coreo electrónico: jacoper@utp.edu.co

\*\* Docente Catedrático de la Universidad Cooperativa de Colombia, docente Catedrático de la Universidad Libre, sede de Pereira, Docente de la Universidad Tecnológica de Pereira, Ingeniero de Sistemas, magíster en Instrumentación Física, miembro de grupo de investigación Robótica Aplicada reconocido por Colciencias y del semillero en Robótica del programa Mecatrónica de la Universidad Tecnológica de Pereira. Coreo electrónico: famedina@utp.edu.co

\*\*\* Docente de la Universidad Tecnológica de Pereira, Ingeniero Electricista, Magíster en Instrumentación Física, director del Ciclo Profesional del Programa Ingeniería Mecatrónica. Coreo electrónico: jam@utp.edu.co

## Introducción

El robot lego Mindstorms NXT es la evolución de la fichas de lego, posee los elementos básicos de teorías de la robótica; inicialmente fue creado para niños, aunque algunas universidades lo han utilizado para la enseñanza de la robótica y la programación en sus asignaturas.

Las características que tiene el robot Lego Mindstorms NXT hacen de este kit una elección adecuada para ser utilizado en entornos educativos. Sus principales características son:

- Un relativo bajo precio
- Facilidad de configuración
- Fácil reprogramación
- Versatilidad y robustez

## Metodología

Las instigaciones realizadas por el grupo y el semillero de investigación en robótica para la elaboración de este documento son de tipo descriptivo y evaluativo, ya que se pretende mostrar las características que tiene el robot Lego Mindstoms NXT para la enseñanza de la robótica y evaluar el *software* de análisis numérico Matlab con algunos de sus *toolboxes* para la programación del robot lego como alternativa científica.

Para la realización de esta investigación se llevaron a cabo algunos procesos. Se realizó la búsqueda de una herramienta de *hardware* que facilitara a los estudiantes de robótica la construcción de robots de una forma rápida y ágil llegando al robot Lego Mindstorms como la herramienta más adecuada. Se pusieron a prueba cada una de las características que tiene el robot Lego Mindstorms NXT en cuanto su arquitectura y programación. Por otro lado, se consideraron algunos lenguajes

para la programación del robot NXT llegando a la conclusión de que la herramienta que se adecuaba más a las necesidades del grupo de investigación para manipular el Robot Lego NXT era el *software* de análisis numérico. Por otro lado, se dio uso a los diferentes *toolboxes* que tiene Matlab para la programación del robot y se desarrollaron algunos programas que permiten la manipulación de los diferentes dispositivos que posee el robot NXT.

## Definiciones

### *El robot lego NXT*

El kit en su versión NXT incluye, aparte de múltiples elementos estructurales como ruedas, engranajes, barras o bolas, motores de corriente continua con *encoders*, diferentes tipos de sensores y el denominado *cerebro* conocido como el NXT *brick*. El *brick* utiliza un procesador AT91SAM7S256 de 32 bits con 256 kb de memoria flash, 64 KB de RAM, un microcontrolador de 8 bits, un *display* LCD de 100x64 pixel, *bluetooth*, USB 2.0 e interfaces I2C. El sistema puede ser programado en varios lenguajes de programación bajo plataformas MacOS, Linux o Windows.



Figura 1. Robot Lego Mindstorms NXT  
Fuente: Compubot 2010

*El cerebro NXT*

El cerebro del Lego Mindstorms NXT permite que el robot cobre vida y realice diferentes operaciones.



**Figura 2.** Cerebro del NXT  
 Fuente: Compubot 2010

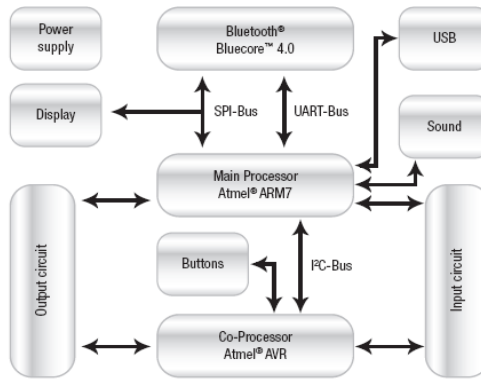
En *cerebro* NXT cuenta con siete cables similares a los que se usan en los teléfonos o conexiones de red de datos. Están divididos en conectores de entrada que se usan para enlazar los 4 sensores que utiliza el robot y 3 conectores de salida que se sirven para conectar los servomotores. También cuenta con un enlace por USB para cargar y descargar programas.

A nivel interno, el *cerebro* del NXT es un procesador ARM-7 de 32 bits (el AT91SAM7S256, para ser exactos), posee 256 kb de memoria flash para los programas (y para el firmware), 64 kb de RAM para los datos y una velocidad de funcionamiento de 48 MHz (The Lego Mindstorms Team, 2006).

Tiene, además, un procesador de 8 bits dedicado íntegramente a la gestión de las entradas y salidas del NXT y otro pequeño procesador que se encarga de las comunicaciones inalámbricas (*bluetooth*).

La responsabilidad de la interface USB recae sobre el propio procesador central y puede alcanzar velocidades de hasta 12 Mbit/s.

La figura 3 muestra el esquema interno de cerebro del robot Lego NXT (Compubot, 2010).



**Figura 3.** Mapa del cerebro del robot Lego Mindstorms NXT  
 Fuente: Compubot 2010

*Servomotores*

Los servomotores permiten dar al robot la capacidad de movimiento. Los motores del robot se pueden sincronizar automáticamente, para que éste se desplace en línea recta.



**Figura 4.** Servomotor  
 Fuente: Compubot 2010

Cada servomotor incorpora un sensor de posición (un grado de resolución) que permite al controlador recibir información del

movimiento ajustando éste a los requerimientos del programa.

De esta forma se puede hacer que el robot avance una distancia concreta o que realice un giro de un ángulo determinado, sin utilizar temporizaciones.

Los servomotores del NXT poseen una velocidad de rotación del eje de 170 rpm (The Lego Minsdtorms Team, 2006)

#### *Sensor de sonido*

Este sensor hace que el robot escuche y reaccione ante el sonido.



Figura 5. Sensor de Sonido  
Fuente: Compubot 2010

El sensor de sonido le permite detectar la intensidad de esta medida en decibeles (db medida de presión del sonido) (The Lego Minsdtorms Team, 2006):

- 4-5% es casi como silencio en una habitación.
- 5-10% es como el sonido producido por una conversación a distancia.
- 10-30% es como el sonido producido por una conversación normal.
- 30-100% es como poner música a alto volumen.

#### *Sensor de tacto*

El sensor de tacto le da al robot la posibilidad de sentir cuándo se presiona la base naranja, además éste tiene, en su parte frontal, una cavidad donde se pueden adaptar diferentes piezas de lego para darle mayor versatilidad.

Con este sensor se pueden hacer aplicaciones como detección de objetos por contacto.



Figura 6. Sensor de tacto  
Fuente: Compubot 2010

#### *Sensor ultrasónico*

El sensor ultrasónico permite al robot detectar objetos próximos. También se puede utilizar para hacer el robot evite obstáculos, tome medidas de distancias, y detecte el movimiento.



Figura 7. Sensor de ultrasonido  
Fuente: Compubot 2010

El sensor ultrasónico permite que el robot pueda detectar objetos; este sensor se puede utilizar para hacer aplicaciones como evitar obstáculos, hacer una medición de una distancia, detectar movimientos, entre otros.

El sensor ultrasónico permite medir distancias entre 0 y 255 cm, con error de +/-3 cm. El principio de funcionamiento de este sensor es el mismo que utilizan algunos animales como los murciélagos (The Lego Minsdtorms Team, 2006).

Sensor de luz



Figura 8. Sensor de luz  
Fuente: Compubot 2010

El sensor de luz le permite al robot dotarlo de visión, permitiéndole al robot distinguir entre luz y oscuridad, también puede medir la intensidad de luz, en el caso de los colores el robot los ve con diferentes intensidades de color negro y blanco y todas las combinaciones intermedias.

Con este sensor se pueden montar aplicaciones como alarmas, detector de colores, selección de objetos entre otros (The Lego Minsdtorms Team, 2006).

#### Programando el NXT

El Lego Mindstorms NXT se puede programar en diferentes entornos de desarrollo de

software; para la programación del NXT existen varios lenguajes de programación comerciales, también los hay desarrollados por la comunidad de *software* libre (Lrobotikas, 2010).

Tabla 1. Lenguajes de programación para el NXT

	Tipo de licencia	
	Lenguajes de programación comerciales	Lenguajes de programación de libre uso
Lenguajes de programación	- NXT-G Educación	- BricxCC
	- NXT-G	- NBC (Next Byte Codes)
	- Robolab 2.9	- NXC
	- RobotC	- leJOS NXJ
	- LabVIEW	- leJOS OSEK (C/C++)
	- Microsoft Robotics Developer Studio	- pbLua
	- Matlab	- nxt#
	- Simulink	- LOGO for nxt
	- RoboRealm	- URBI para LEGO MINDSTORMS
		- Python

Fuente: Lrobotikas (2010)

#### Programando el nxt en Matlab

Matlab es un programa que se usa para realizar cálculos numéricos con vectores y matrices, es utilizado para resolver problemas de ingeniera.

Para controlar el robot Lego Mindstorms NXT desde Matlab existen en la actualidad solo tres *toolboxes*.

#### Toolbox diseñados en Matlab

##### Toolbox NXT de MathWorks

Este *toolbox* fue creado por la casa fabricante de *software* MathWorks (desarrollado por los diseñadores del Matlab) (se puede descargar de la dirección web <http://www.mathworks.com/programs/lego-mindstorms-nxt-software/>).

El *toolbox* permite controlar los motores y sensores del robot desde Matlab y Simulink, permite programar rápidamente el NXT, además de combinar con éste otras *toolboxes* propios de Matlab como el *toolbox* de adquisición de datos y de visión artificial, también permite al usuario utilizar todo el potencial matemático que posee esta herramienta (The MatlabWorks, 2010).

Para comunicar el robot con Matlab, este hace uso del *toolbox* de entrada y salida (I/O), que le da la posibilidad al *software* de enviar comandos al robot lego NXT por medio de una conexión bluetooth.

#### RWTH-Mindstorms NXT *Toolbox* for Matlab

La Universidad RWTH Aachen de Alemania (en la actualidad es la segunda universidad tecnológica con más estudiantes de Alemania y una de las más renombradas de Europa) diseñó un *toolbox* para manipular el Robot Lego NXT desde Matlab; en este proyecto participaron más de 300 estudiantes y 60 supervisores de la RWTH. Este *toolbox* permite avanzar hacia la interacción con sensores del Lego NXT y motores, de forma transparente a través de controles y comandos. Posee un protocolo muy bien documentado (Institute of Imaging & Computer Vision, 2010).

El *toolbox* es licenciado como *software* de código abierto como parte de la GNU GPL (General Public License) y puede ser descargado gratuitamente desde la página oficial <http://mindstorms.lfb.rwth-aachen.de/>.

#### RealTerm y *bluetooth* para controlar el Lego Mindstorms NXT

Gregory Gutt del Laboratorio de la Universidad George Mason Neural Engineering, desarrolló un paquete de *software* para

controlar el Lego Mindstorms NXT usando *bluetooth*, RealTerm y el *software* de MathWorks Matlab (nxtasy.org, 2010). El *toolbox* se puede descargar en la dirección web: <http://nxtasy.org/wp-content/uploads/2006/11/matlab-realterm-lego-mindstorm-nxt.zip>.

Realterm es un programa de terminal diseñado especialmente para la captura, control y depuración de binarios. Puede descargarse de manera gratuita en <http://realterm.sourceforge.net/>.

#### Programando el robot en Matlab

A continuación se muestran algunos programas elaborados en Matlab utilizando los servomotores y sensores del Lego NXT. Estos programas se realizaron a través del *toolbox* NXT de MathWorks.

Para la elaboración estos programas se utilizó la versión de R2009b de Matlab.

Para programar el robot Lego NXT en Matlab se debe tener en cuenta lo siguiente:

- *Nombre de los motores y puertos en Matlab:* estos nombres deben coincidir con la ubicación real de cada dispositivo.

- Motores
  - nxt.PortA
  - nxt.portB
  - nxt.PortC
- Sensores
  - nxt.Port1 (touch)
  - nxt.Port2 (sound)
  - nxt.Port3 (light\_fon)
  - nxt.Port4 (distance)

- *Controlar el motor conectado a un puerto:*

```
>> start(nxt.PortA, 30);
```

La instrucción <Start> permite iniciar un motor al porcentaje de velocidad indicada.

Se debe tener en cuenta que el porcentaje de velocidad puede ir en 0% y 10%

```
>> stop(nxt.PortA)
```

La instrucción <Stop> se utiliza para parar el motor que en su parámetro.

- *Configurar un sensor conectado a un puerto:*

```
>> set(nxt.Port1, 'type', 'touch')
```

La anterior instrucción se utiliza para indicar en qué puerto se encuentra un determinado sensor

Para obtener datos del sensor se utiliza el comando <getdata> se le debe indicar el puerto de donde se capturan los datos.

```
>> v = getdata(nxt.Port1)
```

```
v =
```

```
20
```

- *El dispositivo bluetooth:* El dispositivo *bluetooth* que se utiliza para conectar el robot NXT, debe contener el protocolo Serial Port Profile (SPP). Éste protocolo permite que el dispositivo *bluetooth* emule un puerto serial RS232 en el computador.

Para los programas que se muestran en este documento se utilizó un dispositivo *bluetooth* ENUBT-C1E marca Encore, este dispositivo tiene las siguientes características:

- Rango de Cobertura: hasta 100 metros
- Dimensiones: 52 mm x 21mm x 9 mm
- Peso: 8 g
- Velocidad: 3Mbps
- Operando en rangos: hasta 100 m

- *Programando los motores:* en este programa el robot avanza en línea recta y luego gira a la izquierda o a la derecha. Para ello se utiliza un número aleatorio entre 0 y 1.  
TIEMPO\_GIRO = 0.60;  
TIEMPO\_MOVIMIENTO=1;  
a=1;

```
while a<=5
start(nxt.PortB, 75);
start(nxt.PortC, 75);
pause(TIEMPO_MOVIMIENTO);
stop(nxt.PortB)
stop(nxt.PortC)
x = rand(1,1); % genera el numero aleatorio
if x <= 0.5
start(nxt.PortC, -30);
pause(TIEMPO_GIRO)
else
start(nxt.PortB, -30);
pause(TIEMPO_GIRO)
end
pause(1)
a=a+1;
end
stop(nxt.PortB)
stop(nxt.PortC)
```

- *Programando el sensor de tacto:* en este programa implementado en Matlab, el robot avanza evitando los obstáculos que se encuentre en su camino. Cada vez que el robot golpee un objeto, retrocede un poco, gira y continúa avanzando.

```
a=1;
set(nxt.Port1, 'type', 'touch')
while a<=4
start(nxt.PortB, 50);
start(nxt.PortC, 50);
while (true)
v = getdata(nxt.Port1);
if v<100
stop(nxt.PortB);
stop(nxt.PortC);
break
end
end
start(nxt.PortB, 75);
```

```

pause(1)
stop(nxt.PortB);
a=a+1;
end
    
```

- *Programando el sensor ultrasónico:* el sensor de ultrasonido funciona como un sonar, éste envía un frente de ondas de ultrasonidos para que se refleje en algún objeto y luego mide el tiempo que tarda en recibir su eco, para así conocer a qué distancia se encuentra.

El siguiente código muestra cómo se puede programar el sensor de ultrasonido en Matlab:

```

distancia = 20;
a=1;
set(nxt.Port4, 'type', 'distance')
while a<=4
start(nxt.PortB, 50);
start(nxt.PortC, 50);
while (true)
v = getdata(nxt.Port4);
if v<=distancia
stop(nxt.PortB);
stop(nxt.PortC);
break
end
end
start(nxt.PortB, 75);
pause(1)
stop(nxt.PortB);
a=a+1;
end
    
```

- *Programando el sensor luz:* el principio básico en el que se basa el seguidor de línea es que el robot intente mantenerse en el borde de la línea negra, alejándose de ella si el nivel de luz es muy pequeño (el sensor de luz se sitúa en mitad de la línea negra) y girando hacia ella si el sensor se sale de ésta y detecta un nivel de luz alto.



Figura 11. Robot NXT seguidor de línea  
Fuente: Compubot 2010

El siguiente código muestra un programa sencillo realizado en Matlab que hace un seguimiento de la línea con un único valor umbral de luz.

```

umbral = 40;
a=1;
set(nxt.Port3, 'type', 'light_fon')
while a<=50
start(nxt.PortB, 20);
start(nxt.PortC, 20);
pause(0.2);
stop(nxt.PortB);
stop(nxt.PortC);
v = getdata(nxt.Port3);
while (v<=umbral)
start(nxt.PortB, -20);
pause(0.2);
stop(nxt.PortB);
pause(0.2);
v = getdata(nxt.Port3);
end
a=a+1;
end
    
```

## Conclusiones

El *software* de análisis numérico, como es Matlab, resulta ser una herramienta de alta ingeniería que permite hacer usos de complejos algoritmos para el procesamiento de los datos y la toma de decisiones que de otra forma serían altamente demandantes en el consumo de tiempo de su implementación. Matlab, siendo el líder en el mercado, no resulta ser una herramienta libre, por lo que se encuentra limitada a la adquisición de una licencia de uso. Como alternativa al manejo de Matlab para el control de los robots Lego NXT en la actualidad se está buscando el desarrollo de un *toolbox* que complemente su uso mediante plataformas clones de Matlab tales como Scilab u Octave, para las cuales no existen, a la fecha, trabajos importantes que permitan su uso.

De los diferentes robots educativos evaluados en la investigación se llegó a la conclusión de que el Robot Lego NXT, por su facilidad de armado y por los componentes de *hardware* que lo conforman, es una muy buena herramienta para la enseñanza de la robótica y la programación.

La programación del robot NXT mediante el *software* de análisis numérico Matlab permite la integración de otros *toolboxes* que esta herramienta posee, como los *toolboxes* de visión artificial, redes neuronales, adquisición de datos entre otros. Esto le da un panorama muy interesante en la enseñanza de la robótica con el robot Lego NXT con herramientas muy robustas.

El grupo de investigación en Robótica Aplicada y el Semillero de investigación en Robótica del programa de Mecatrónica de la Universidad Tecnológica de Pereira, están comprometidos con la idea de desarrollar un

*toolbox* para el manejo de robot Lego NXT mediante Scilab para el cual no existe referente.

## Resultados

Esta investigación da como resultado principal la evaluación del *software* de análisis numérico Matlab; para la programación del robot Lego NXT se realizaron algunos algoritmos que permitieron manipular los diferentes dispositivos de *hardware* que posee el robot Lego.

Con base a esta investigación algunos estudiantes del programa de Mecatrónica están realizando proyectos en los que se involucran *toolboxes* de visión artificial y redes neuronales con el robot Lego NXT.

Tomando como base el trabajo realizado con el robot Lego NXT y Matlab, la Universidad Tecnológica, en cabeza del programa de Mecatrónica, se encuentra en proceso diseñar un *toolbox* para el manejo del robot NXT que responda a; dicho *toolbox* se diseña en el programa de análisis numérico Scilab (*software* libre). El *toolbox* fortalecerá el trabajo de la programación y la robótica de los grupos de investigación que trabajen con el robot Lego Mindstorms NXT.

## Referencias

- The MatlabWorks, (s.f) [en línea], disponible en: <http://www.mathworks.com>, recuperado: 1º de enero de 2010.
- Compubot [en línea] disponible en: [http://complubot.educa.madrid.org/pruebas/lego\\_nxt\\_version\\_educativa/lego\\_nxt\\_version\\_educativa\\_index.php](http://complubot.educa.madrid.org/pruebas/lego_nxt_version_educativa/lego_nxt_version_educativa_index.php), recuperado: 15 de enero de 2010.
- Institute of Imaging & Computer Vision [en línea], disponible en: <http://mindstorms.lfb.rwth-aachen.de/index.php/en>, recuperado: 5 de enero de 2010.

Lrobotikas [en línea], disponible en: [http://lrobotikas.net/mediawiki/index.php?title=Programaci%C3%B3n\\_NXT](http://lrobotikas.net/mediawiki/index.php?title=Programaci%C3%B3n_NXT), recuperado: 4 de enero de 2010.

nxtasy.org [en línea], disponible en: <http://nxtasy.org/2006/11/28/controlling-nxt-from-math-works-matlab/>, recuperado: 4 de enero de 2010.

The Lego Minsdtorms Team, Lego minsdtorms User Guide (2006) (guía de usuario que viene con el robot).