

Universidad Politécnica de Cartagena

Escuela Técnica Superior de Ingeniería Industrial

Estudio de las posibilidades didácticas en ingeniería de control del LEGO Mindstorms NXT

Guillermo Nieves Molina
Septiembre 2008



- Introducción
- Objetivos
- Estado del arte
- Análisis del NXT
- Pruebas y ensayos
- Dos aplicaciones
- Conclusiones



- Un poco de historia... (I)
 - 1986 → Acuerdo de colaboración **LEGO – MIT**
 - LEGO financia investigaciones sobre aprendizaje a cambio aprovecha el trabajo del MIT
 - 1998 → LEGO Mindstorms **RCX** (1ª versión)
 - Gran éxito (**80K** unidades en 3 meses)
 - Más limitado que el NXT
 - Aspecto próximo al LEGO tradicional
 - Varias versiones sucesivas

Introducción





- Un poco de historia... (II)
 - 2006 → LEGO Mindstorms **NXT**
 - Más completo y potente
 - Utiliza sobre todo piezas de LEGO Technic
 - Por 250 €:
 - CPU 32 bit (\approx i486) + 4 sensores + 3 motores DC
 - Comunicación BT y USB, entre ellos y a PC
 - Muchas piezas de LEGO Technic...

Introducción





- Montaje propuesto por LEGO





- Introducción
- Objetivos
- Estado del arte
- Análisis del NXT
- Pruebas y ensayos
- Dos aplicaciones
- Conclusiones



- Objetivo principal: analizar y evaluar las posibilidades en la docencia de ingeniería de control
- Premisas:
 - Uso en laboratorios de prácticas
 - Mucho más **barato** que maquetas habituales
 - Gran **versatilidad**



- Desarrollo:
 - Búsqueda de información sobre la escena
 - Profundización en el sistema
 - Análisis, calibrado, y pruebas a componentes
 - Aplicaciones concretas útiles en docencia
 - Conclusiones



- Introducción
- Objetivos
- Estado del arte
- Análisis del NXT
- Pruebas y ensayos
- Dos aplicaciones
- Conclusiones



- *LEGO Mindstorms NXT* devuelve más de **medio millón** de resultados en Google
- Múltiples lenguajes y entornos de programación, libros, campeonatos mundiales de robótica,...
- Ya ha sido **usado en universidades** (MIT, UBC, Instituto RWTH, etc.)
- Hace unos meses: experimento HALE: varios NXT a **30 km de altura**



Imágenes del experimento HALE

Estado del arte





- Especificaciones, tanto de HW como de SW, **liberadas por LEGO**
- La mayoría de los proyectos son también libres
- Escena actual: en torno a **internet** (foros, blogs, grupos de usuarios, etc.)
- Casi al 100% en **inglés**
- Ingenieros, informáticos, profesores universitarios...
- Comunidad muy activa y *amigable*



- Esquemmatizando:
 - Programación del NXT
 - Lenguajes
 - Entornos
 - Hardware disponible
 - Proyectos en otras universidades
 - Bibliografía y fuentes de información
 - Otras utilidades



- Programación (I):
- Gran cantidad de lenguajes existentes
 - Basados en C/C++, Java, Assembler, etc.
 - La mayoría son **libres y multiplataforma**
 - Muchos disponen de su **propio firmware**
 - Los más usados:
 - NXT-G → Original suministrado por LEGO
 - Robot C → De pago, alto nivel
 - **NBC/NXC** → El más popular y completo



- Programación (II):
- Entornos integrados de desarrollo (IDE)
 - NXT-G → El original de LEGO
 - RobotC → Para el lenguaje del mismo nombre
 - **BricxCC** → Para NBC/NXC, completo y potente
 - Edición enriquecida
 - Explorador del NXT
 - Utilidades varias

Estado del arte



The screenshot shows the Bricx Command Center software interface. The title bar reads "Bricx Command Center - [Pid con entrada de valores.nxc]". The menu bar includes File, Edit, Search, View, Compile, Tools, Window, and Help. The toolbar contains various icons for file operations and execution. On the left, there are two tree views: "Functions" (move, normPwr) and "Tasks" (main). Below these is a "Sensor modes" list. The main area is a code editor displaying the following C++ code:

```
TextOut(0,LCD_LINE2,"Valor actual:");
NumOut(0,LCD_LINE3,P);
ok=ButtonCount(BTNCENTER,false);
}

//Then I...
while(ok==1){

if(resetki==1) //Ki starts from 0
{ButtonCount(BTNLEFT,true);ButtonCount(BTNRIGHT,true);resetki=0;}

TextOut(0,LCD_LINE1,"Introduce Ki:");
I=2*(ButtonCount(BTNRIGHT,false)-ButtonCount(BTNLEFT,false));

if(I<0 || I>100)
{I=0;ButtonCount(BTNLEFT,true);ButtonCount(BTNRIGHT,true);}

TextOut(0,LCD_LINE2,"Valor actual:");
NumOut(0,LCD_LINE3,I);
ok=ButtonCount(BTNCENTER,false);
}

//...and then D
while(ok==2){

if(resetkd==1) //Kd starts from 0
{ButtonCount(BTNLEFT,true);ButtonCount(BTNRIGHT,true);resetkd=0;}

TextOut(0,LCD_LINE1,"Introduce Kd:");
D=2*(ButtonCount(BTNRIGHT,false)-ButtonCount(BTNLEFT,false));

if(D<0 || D>100)
```

At the bottom of the window, there are status indicators: "1: 1", "no port", and "Insert".



- Hardware disponible (I):
 - Apertura de las especificaciones → Mercado de nuevos productos
 - Terceras empresas: Mindsensors, **HiTechnic**, Vernier, etc.
 - Nuevos sensores: acelerómetro, giróscopo, presión, temperatura, humedad,...
 - No hay nuevos actuadores (suficiente con motores)



- Hardware disponible (II):
 - Batería recargable: Realmente útil
 - (usa 6 pilas a la vez)
 - Mochila BlueTooth
 - Adaptador BlueTooth para PC





- Proyectos en otras universidades:
 - Tesis doctoral de Daniele Benedettelli, sobre robótica en grupo cooperativa (U. Siena)
 - **APRIL** (A PID Robot Implemented with LEGO), vehículo con un regulador PID (UBC)
 - **Matlab Meets Mindstorms**, sobre el uso de Matlab en el control del NXT (RWTH Aachen)
 - Proyecto de gran importancia
 - 300+ alumnos, 100+ robots
 - **Toolbox** de comunicación Bluetooth para Matlab

Estado del arte



Robot APRIL

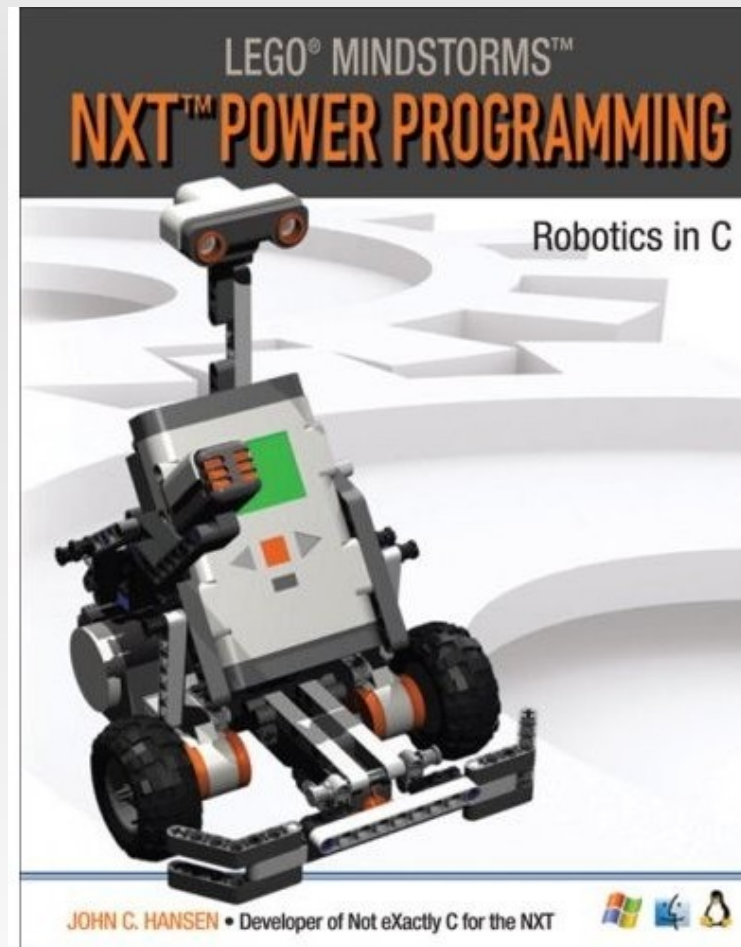


Matlab meets Mindstorms



- Bibliografía y fuentes de información
 - Blogs, portales y foros: NXTasy, blog y foro, NXTlog, The NXT Step, Mindstorms Community...
 - Libros sobre montajes concretos, de poco interés
 - Libros sobre programación en distintos lenguajes
 - No hay gran cantidad de libros publicados
→ la mayoría de la info está en **internet**

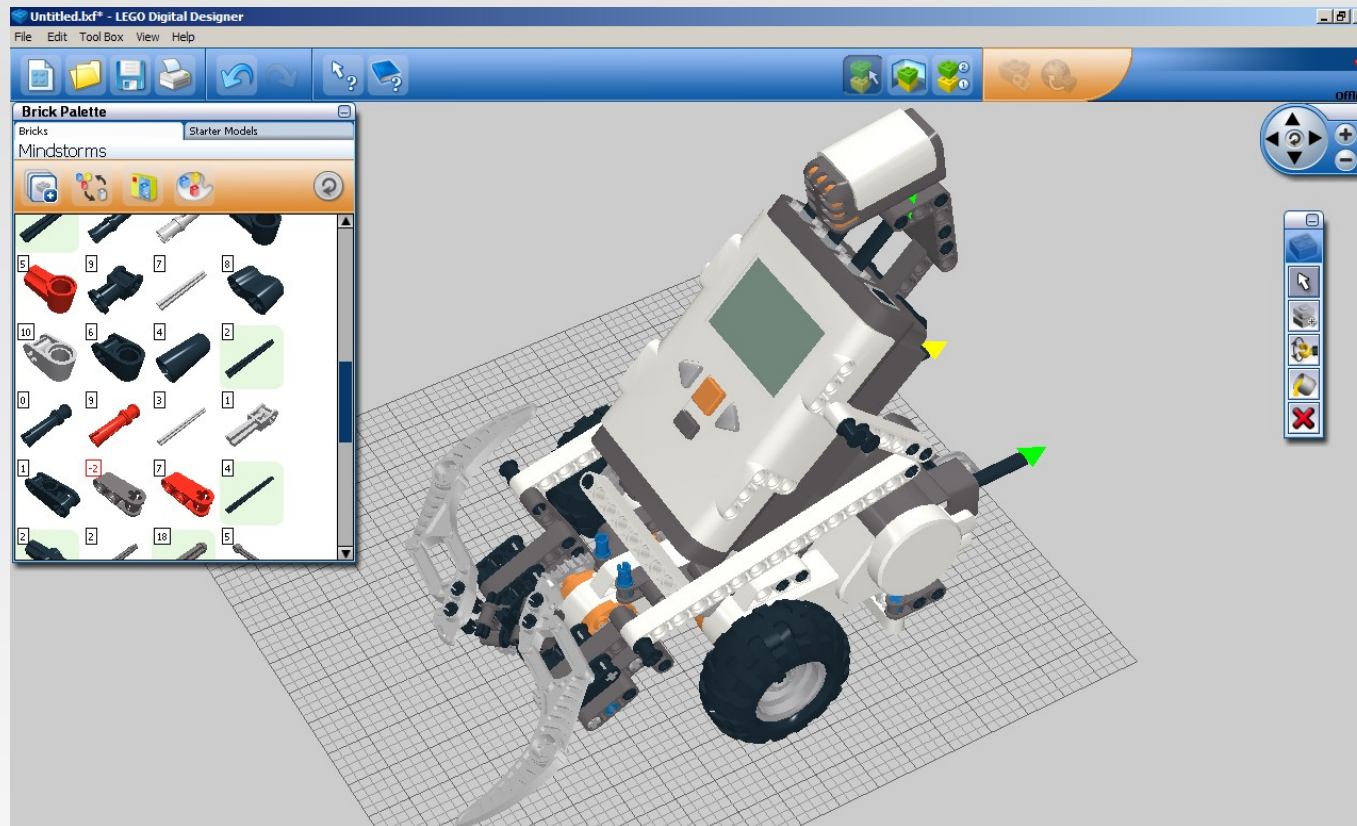
Estado del arte





- Otras utilidades
 - Herramientas CAD para generar manuales de instrucciones
 - Ldraw
 - LeoCAD
 - **LEGO Digital Designer**

Estado del arte





- Introducción
- Objetivos
- Estado del arte
- Análisis del NXT
- Pruebas y ensayos
- Dos aplicaciones
- Conclusiones



- Componentes:
 - El Ladrillo Inteligente (*The Brick*)
 - Actuadores:
 - 3 motores DC con tacómetro integrado
 - Sensores:
 - Distancia
 - Luz
 - Sonido
 - Tacto



- El ladrillo inteligente

- CPU:

- Atmel 32 bit ARM (principal) @ **48 MHz**, 256 KB Flash + 64 KB de RAM
- Atmel 8 bit AVR (secundario) @ 8 MHz, 4 KB Flash + 512 B de RAM

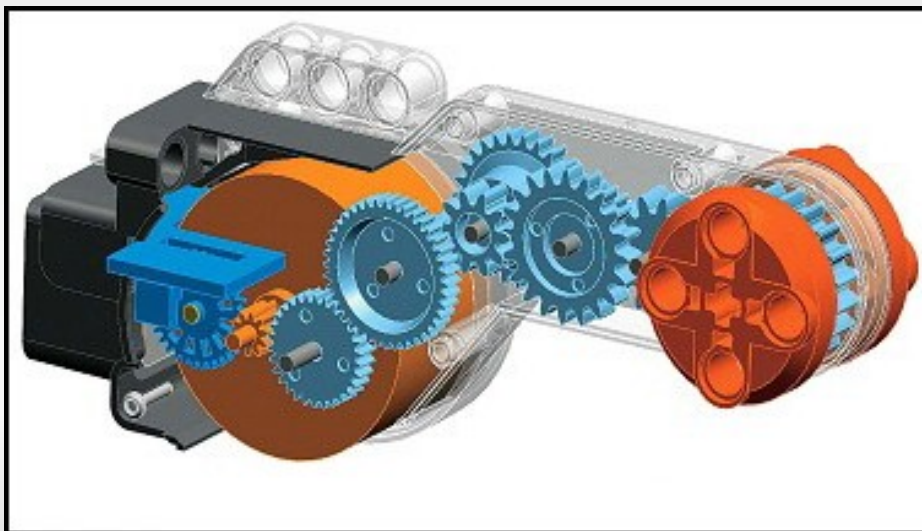
- Comunicaciones:

- USB 2.0 (12 Mbit/s)
- Bluetooth (clase II, procesador CSR BlueCore 4)





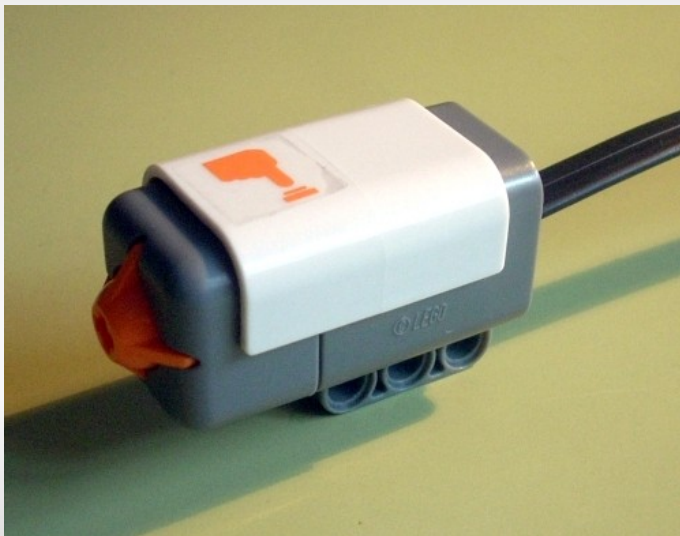
- Los motores DC:
 - *Motor* es motor + tren de engranajes + tacómetro
 - Funcionamiento: PWM
 - Hasta 160 rpm sin carga





- Los sensores (I):
 - Sensor de luz: diodo LED + fototransistor
 - Medida de distancias cortas y detección de obstáculos
 - Poco preciso
 - Sensor acústico
 - Hasta 90 dB
 - Puede funcionar en dB o dBA (ajustados)
 - Sensor de presión
 - Funcionamiento binario: pulsado - liberado

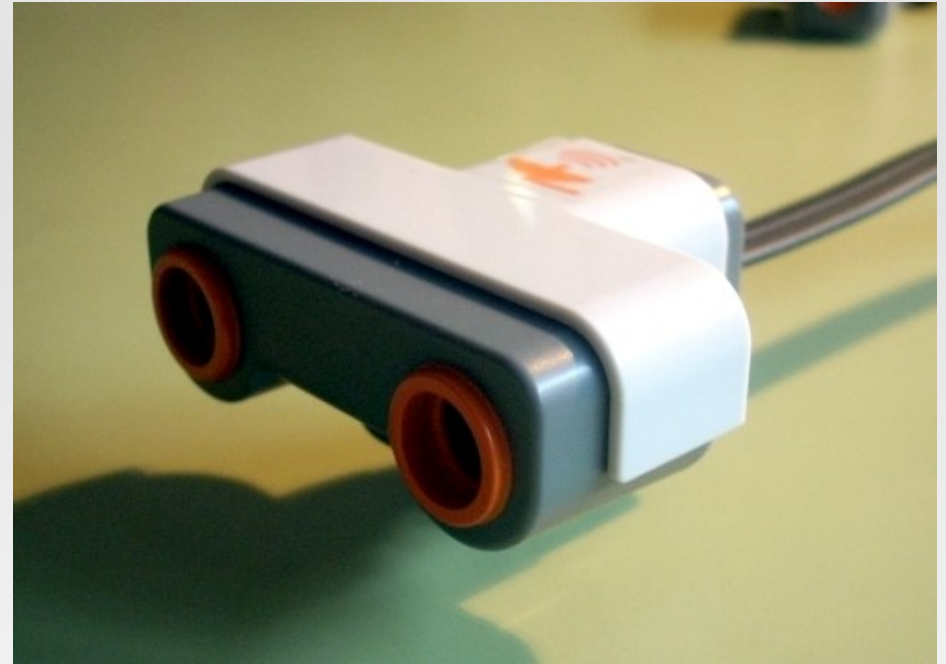
Análisis del NXT



Sensores de luz (izquierda arriba),
sonido (arriba) y presión (izquierda abajo)



- Los sensores (II):
 - Sensor de distancia
 - Funcionamiento:
ultrasonidos
 - Mucho más complejo y preciso
(μ procesador propio)
 - Opera mediante el protocolo I²C
 - Alcance máximo: 255 cm





- Introducción
- Objetivos
- Estado del arte
- Análisis del NXT
- Pruebas y ensayos
- Dos aplicaciones
- Conclusiones

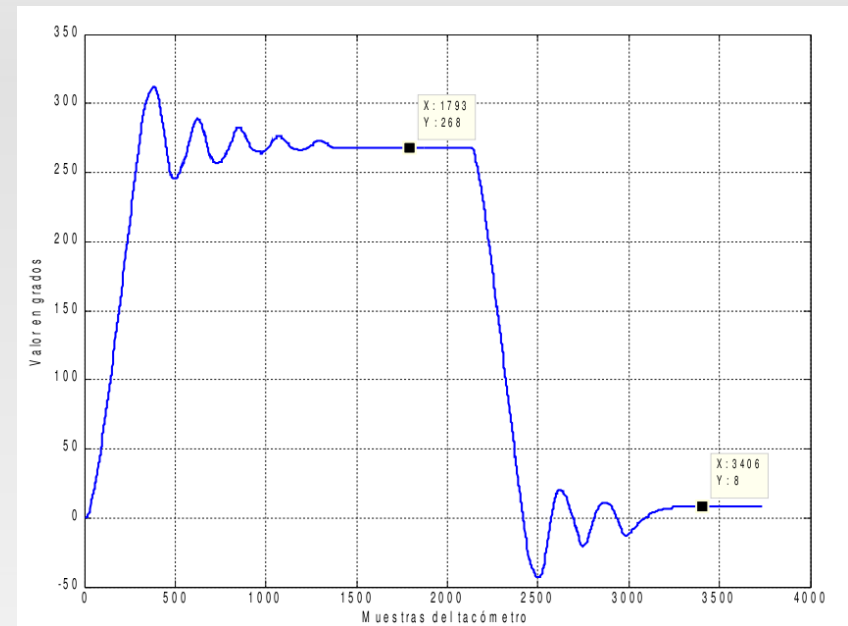
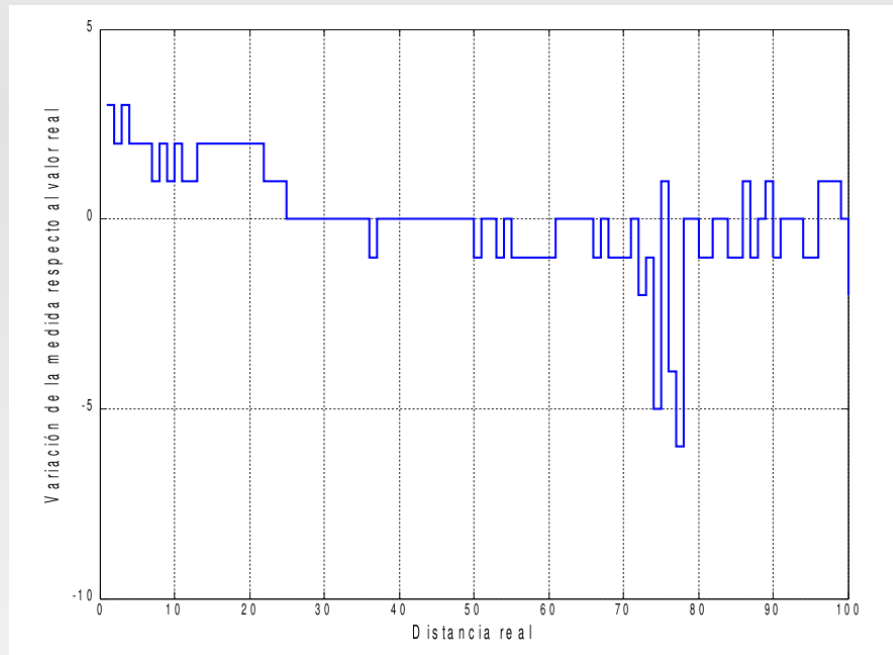


- Pruebas a los componentes (I):
 - Sensor de distancia:
 - Distancias hasta 100 cm: Respuesta precisa
 - Distancias a partir de 100 cm: Dificultad de detección
 - Los valores varían si hay obstáculos cercanos
 - Brújula (HiTechnic):
 - Los motores y los aparatos electrónicos interfieren
 - Distancia de seguridad



- Pruebas a los componentes (II):
 - Escritura a archivos (`CreateFile()`)
 - No es exactamente una prueba, pero es muy importante (logs)
 - El NXT puede manejar hasta **4 archivos a la vez**
 - Los datos se pueden importar y trabajar con ellos
 - Motores
 - Para giros de x grados → regulador **PID**
 - Los valores por defecto no son los idóneos
 - No es posible trabajar en lazo abierto

Pruebas y ensayos



Diferentes gráficas obtenidas en las pruebas



- Introducción
- Objetivos
- Estado del arte
- Análisis del NXT
- Pruebas y ensayos
- Dos aplicaciones
- Conclusiones

Dos aplicaciones



- NXTWay (balancín sobre dos ruedas)
 - Basado en el original de Steve Hassenplug
 - Diferentes versiones (LegWay, NXTWay-GS,...)
- Autoparker (vehículo que aparca solo)
 - Utiliza sensores de distancia y orientación



- NXTWay:
 - Funcionamiento:
con sensor óptico
 - Toma la consigna al inicio → Lazo con PID
 - También puede usarse un sensor giróscopico:
 - Consigna: movimiento angular cero
 - Consigna: ángulo cero (\int velocidad angular)





- Autoparker
 - Aparca en batería
 - Funcionamiento más secuencial
 - Sensor de distancia para *buscar* el hueco
 - Brújula para los giros
 - Sensor de luz para recular hasta el final





- Introducción
- Objetivos
- Estado del arte
- Análisis del NXT
- Pruebas y ensayos
- Dos aplicaciones
- Conclusiones



- Positivas
 - Buen precio
 - Comunidad sólida y *amigable*
 - Tendencia a lo libre y abierto
 - Versatilidad
 - Programación en lenguajes habituales
 - Calidad y solidez de los componentes



- Negativas
 - Precisión y potencia pueden ser insuficientes
 - Dificultades de montaje
 - Programas como el LDD



- Trabajos futuros
 - Ampliación de las dos aplicaciones
 - Uso de BlueTooth
 - Comunicación
 - Trabajo en grupo
 - Toolbox de Matlab
 - Aplicaciones más complejas

Universidad Politécnica de Cartagena

Escuela Técnica Superior de Ingeniería Industrial

Estudio de las posibilidades didácticas en ingeniería de control del LEGO Mindstorms NXT

Guillermo Nieves Molina
Septiembre 2008