

Generación de movimientos del torso de un robot humanoide de arquitectura abierta.

González Trejo José Enrique, López Sandoval Andrés Ernesto, Baliño
González Rosa Isela, Rodríguez González Andrea Jaqueline, Reyes Cruz Luis
Ángel, Pedraza Ortega Jesús Carlos.

en_gt@hotmail.com

Facultad de Informática, Universidad Autónoma de Querétaro.
Av. de las Ciencias S/N Campus Juriquilla, Juriquilla, Querétaro, Qro.
C.P. 76230 México. Tel. 1921200 EXT. 5900

Resumen

En el presente trabajo mostramos una estrategia para la generación de movimientos y telecontrol del torso de un robot humanoide de arquitectura abierta el cual se ha desarrollado gracias a la Universidad Autónoma de Querétaro con el apoyo del Fondo de Proyectos Especiales de Rectoría (FOPER).

Con el desarrollo de éste prototipo surgió la necesidad de generar secuencias básicas para el movimiento del robot, con las cuales se puede analizar la mecánica del robot humanoide e implementar funciones como básicas, así como de su manipulación a distancia.

Se utilizó el software de animación Blender, que permitió implementar el análisis cinemático realizado, así como incorporar el modelo CAD del robot y lograr controlarlo en tiempo real mediante la generación de una interfaz robot-Blender.

Palabras clave: Torso Robot Humanoide, Blender, Cinemática, CAD.

1. Introducción

Las interacciones humano-robot han sido motivo de estudio a lo largo de los últimos años, especialmente desde que se visualizó el potencial de la inclusión de los robots en actividades de la vida cotidiana de los seres humanos. Estas interacciones siempre se vieron limitadas debido a una falta de naturalidad en la estructura y, en especial, en los movimientos de los robots, en parte a causa del material de construcción y los componentes electrónicos que involucraban; de igual forma, estos estaban sujetos a procesos de programación complejos y largos, todo lo cual limitaba el rango de posibilidades en las posiciones que podían lograr.

Además la interacción directa entre humano y robot se había realizado hasta ahora por medio de controles que no eran lo suficientemente intuitivos (joysticks, mandos, pulsadores) para generar una correspondencia entre los movimientos humanos y los del robot [1].

El interés en naturalizar los movimientos de los robots se debe en gran parte a la necesidad que se tiene en diferentes campos para que estos puedan realizar movimientos complejos; pero además para que en campos como la robótica de servicio o la enseñanza estos sean aceptados más rápidamente por los usuarios [2].

La incorporación de los desarrollos de la robótica y la tecnología multimedia han permitido incrementar las capacidades de los sistemas remotos. La telerrobótica y la

teleoperación son las áreas de la robótica que se interesan en control y operación de robots a distancia. La telerrobótica es una técnica de teleoperación avanzada, donde el dispositivo maestro puede funcionar de manera automática, y solo requiere la supervisión mínima del operador [3].

Específicamente en el presente trabajo, la teleoperación se logra mediante la interfaz desarrollada en Blender y una comunicación inalámbrica.

2. Estrategia

El objetivo es desarrollar un método para generar secuencias de movimientos para un prototipo de robot humanoide con arquitectura abierta, basado en el análisis y modelado de la cinemática directa, que permita la libertad de manipular sus articulaciones mediante el software de animación Blender [4] y además, hacer una réplica de la secuencia de movimientos en el prototipo del robot humanoide en tiempo real. La estrategia utilizada se muestra en la figura 1.

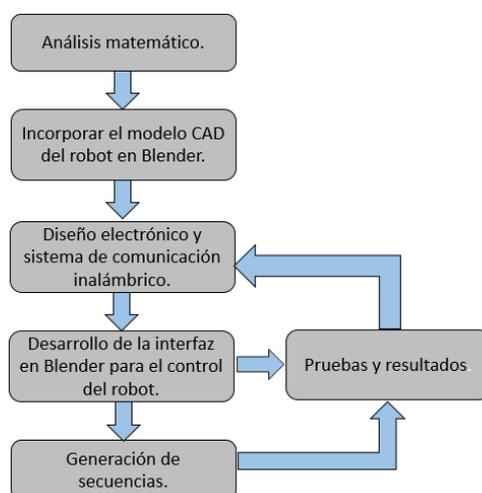


Figura 1. Estrategia propuesta.

De esta manera, se pueden realizar diversas secuencias de movimientos para un posterior análisis que nos permita construir una estrategia de locomoción heurística que sirva como ayuda para trabajos futuros en esta misma rama de la robótica.

2.1 Análisis matemático.

El análisis matemático que se realizó involucra tanto el estudio geométrico como la cinemática directa del torso del robot. El modelo geométrico del robot es la relación entre los valores de las variables asociadas a las articulaciones del robot y la situación (posición y orientación de un sistema de referencia). [5]

Se utilizan las coordenadas homogéneas para representar vectores de posición en un espacio tridimensional, y las matrices de rotación se ampliarán a matrices de transformación homogénea 4×4 para incluir las operaciones translacionales del sistema de coordenadas ligado al cuerpo, siendo utilizado por primera vez este método por Denavit-Hartenberg en 1995. [6]

Tal y como se muestra en la figura 3, se asocia un cuadro o sistema de referencia a cada articulación (excepto por {D} y {H} que pertenecen al efecto final), representado por un sistema de coordenadas cartesianas:

- Flecha roja (eje x)
- Flecha verde (eje y)
- Flecha azul (eje z)
- Flecha negra (GDL)

El sistema de referencia base OXYZ se encuentra ubicado en el centro del torso del robot.

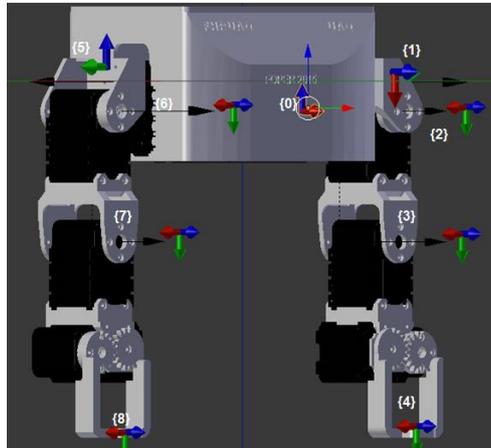


Figura 2. Torso del robot con su sistema de referencia.

Para la solución de la cinemática directa, se obtiene la posición y orientación del efecto final del robot en función de las variables de las articulaciones; se emplearon ecuaciones de transformación homogéneas básicas para resolver la relación entre los eslabones del robot, descrita en la ec. (1) y obtener así una matriz de transformación homogénea compuesta, ec. (3) que relacione el sistema de referencia base con el sistema de referencia de la pinza 0A_4 .

$$\begin{aligned}
 & T_{trans z}(d_1) T_x(-\pi/2) T_{trans x}(d_2) T_x(-\pi/2) * T_{trans z}(d_3) T_z(-\pi/2) T_y(\pi/2 + q_1) T_z(\pi/2) \\
 & * T_{trans z}(d_4) T_{trans x}(d_5) T_x(-\pi/2) T_z(-\pi/2 + q_2) * T_{trans y}(d_6) T_z(-\pi/2 + q_3) T_{trans y}(d_6)
 \end{aligned}
 \tag{1}$$

Las ecuaciones (2.1), (2.2), (2.3), expresan las rotaciones en los ejes (α , Φ , Θ) y la ecuación (2.4) la translación en los ejes (x, y, z):

$$T_{x, \alpha} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\text{sen}\alpha & 0 \\ 0 & \text{sen}\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \tag{2.1}$$

$$T_{y, \Phi} = \begin{bmatrix} \cos\Phi & 0 & \text{sen}\Phi & 0 \\ 0 & 1 & 0 & 0 \\ -\text{sen}\Phi & 0 & \cos\Phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \tag{2.2}$$

$$T_{z, \theta} = \begin{bmatrix} \cos\theta & -\text{sen}\theta & 0 & 0 \\ \text{sen}\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \tag{2.3}$$

$$(2.4) \quad T_{trans} \begin{bmatrix} 1 & 0 & 0 & dx \\ 0 & 1 & 0 & dy \\ 0 & 0 & 1 & dz \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Dónde:

Tabla 1. Elementos del brazo izquierdo

Elemento	Valor	Elemento	Valor
q_1	θ_1	d_3	1
q_2	θ_2	d_4	-1.4
q_3	θ_3	d_5	2.84
d_1	-1	d_6	6.3
d_2	7.5	d_7	5.8

Estos valores están reflejados en la figura 3. Donde los elementos q_i representan GDL. Y los elementos d_i las dimensiones de los eslabones del robot, el cual está conformado por dos brazos articulados y estos a su vez cuentan con 3 eslabones y 3 grados de libertad ($q_i(t)$).

Al ser simétricos los brazos, los elementos d_i del brazo izquierdo son iguales al derecho excepto por d_9 cambia el signo de d_2 , al igual de los q_i .

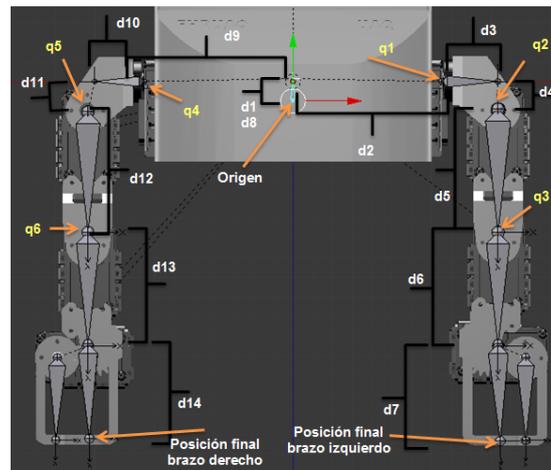


Figura 3. Sistema de referencias de los grados de libertad (GLD) del torso del robot.

Solución de la cinemática directa para el brazo izquierdo:

$${}^0A_4 = \begin{bmatrix} r_{nx} & r_{ox} & 0 & p_x \\ r_{ny} & r_{oy} & r_{ay} & p_y \\ r_{nz} & r_{oz} & r_{az} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Dónde:

$$\begin{aligned}
 (4) \quad r_{nx} &= \cos((-\pi/2 + q_2) + (-\pi/2 + q_3)) \\
 r_{ox} &= \sin((-\pi/2 + q_2) + (-\pi/2 + q_3)) \\
 (5) \quad p_x &= d_2 + d_5 - d_7 \sin((\pi/2 + q_2) + (\pi/2 + q_3)) \\
 (6) \quad r_{ny} &= \sin((-\pi/2 + q_2) + (-\pi/2 + q_3)) * \sin(-\pi/2 + q_1) \\
 (7) \quad r_{oy} &= \cos((-\pi/2 + q_2) + (-\pi/2 + q_3)) * \sin(\pi/2 + q_1) \\
 r_{ay} &= \cos(\pi/2 + q_1) \\
 (9) \quad p_y &= \sin(\pi/2 + q_1) (d_7 \cos((-\pi/2 + q_2) + (-\pi/2 + q_3)) - d_4 + d_6 \cos(-\pi/2 + q_2)) \\
 (10) \quad r_{nz} &= -\sin((-\pi/2 + q_2) + (-\pi/2 + q_3)) * \cos(\pi/2 + q_1) \\
 (11) \quad r_{oz} &= \cos((-\pi/2 + q_2) + (-\pi/2 + q_3)) * \cos(\pi/2 + q_1) \\
 (12) \quad r_{az} &= \sin(\pi/2 + q_1) \\
 (13) \quad p_x &= d_1 + d_3 + d_4 \cos(\pi/2 + q_1) - d_6 \cos(\pi/2 + q_1) \cos(-\pi/2 + q_2) \\
 &\quad - d_7 \cos(\pi/2 + q_1) \cos(-\pi/2 + q_2) + d_7 \cos(\pi/2 + q_1) \sin(-\pi/2 + q_2) \\
 &\quad * \sin(-\pi/2 + q_3) \\
 (14)
 \end{aligned}
 \tag{8}$$

q1: 1° grado de libertad del brazo izquierdo.

q2: 2° grado de libertad del brazo izquierdo.

q3: 3° grado de libertad del brazo izquierdo.

Para la solución de la cinemática directa del brazo derecho se emplea la misma ecuación con las respectivas variables de cada brazo, debido a que son simétricos.

2.2 Incorporar el modelo CAD en Blender.

Esta parte del proyecto consiste en digitalizar el torso del robot humanoide en un software CAD (Solidworks) [7] para posteriormente ser incorporado en el entorno gráfico de Blender, realizando las configuraciones adecuadas (agregando un esqueleto empleado en la animación de personajes en Blender con la orientación descrita en la Figura 3) para que cada una de las articulaciones reproduzca de manera similar los movimientos del modelo real (los huesos del robot forman los eslabones del cuerpo de éste y la cabeza del hueso forma la articulación). Es necesario tener el modelo del robot como archivo .stl, .obj o una extensión similar para que pueda ser incorporado en el entorno de trabajo de Blender, con la mayor precisión al modelo real del robot.

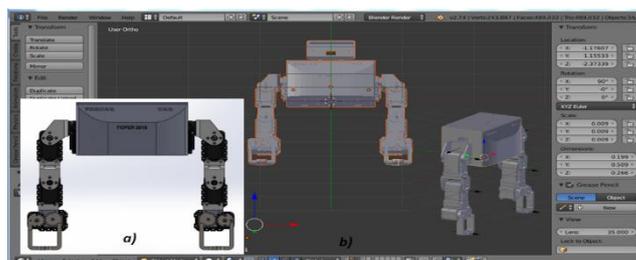


Figura 4. a) Diseño del torso en Solidwoks. b) Incorporación del diseño a Blender.

La API (Application Programming Interfaces) que incorpora Blender, permite al usuario interactuar con el contenido de la aplicación en tiempo real, así como también la posibilidad de ejecutar e implementar scripts y comandos en el lenguaje de programación Python, de esta forma es posible gestionar y establecer una comunicación con otro software o sistema que se requiera mediante Middleware (lógica de intercambio de información entre aplicaciones).

Por lo tanto el entorno de Blender cuenta con los componentes y elementos necesarios para desarrollar un sistema de comunicación inalámbrico que permita la comunicación entre la interfaz del modelo gráfico y el sistema real, teniendo como resultado un sistema de telecontrol de un robot.

2.3 Diseño electrónico y sistema de comunicación inalámbrico.

Una vez finalizado el análisis teórico del torso del robot humanoide se desarrolla el sistema electrónico y de alimentación para el control de los servomotores (Dynamixel AX-12) [8] y el sistema embebido utilizado (BeagleBone Black) [9], como se muestra en la figura 5, así como el sistema de comunicación inalámbrica para el robot humanoide y el área de control empleando el protocolo 802.11.

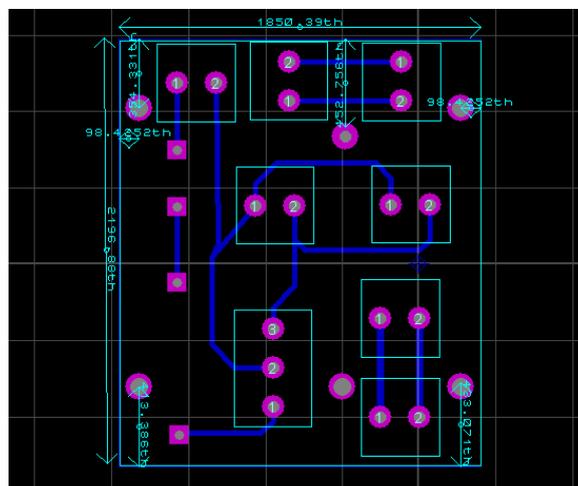


Figura 5. Circuito realizado en Proteus para la alimentación de los servomotores y la BeagleBone Black.

2.4 Desarrollo de la interfaz en Blender para el control del robot.

Los principios básicos para el modelo y simulación de la cinemática de un robot son también empleados en la animación por computadora al describir la orientación y posición de cada uno de los objetos de una escena en un sistema de coordenadas de referencia, [10] siendo en este caso un sistema de coordenadas (x, y, z). Por lo cual el motor gráfico 3D de Blender, en conjunto con sus herramientas de animación y modelado 3D permiten el desarrollo de una interfaz de telecontrol que incorpore el modelo digitalizado de un robot y una estación de telecontrol.

La interfaz de Blender consiste en una interfaz que muestra el torso del robot humanoide y el ángulo en el que se encuentra posicionada cada articulación (servomotor) como se puede apreciar en la Figura 7. Esta interfaz al ser ejecutada, envía un mensaje al robot para establecer una comunicación cliente-servidor.

El programa desarrollado en Python, que funciona como servidor en la tarjeta BeagleBone Black se encarga de enviarle las instrucciones a los servomotores. Estas instrucciones son enviadas por la interfaz de Blender que se activan al presionar una tecla, como se muestra en la tabla 2. Estos comandos son enviados mediante sockets. El diagrama de flujo del programa servidor se muestra en la figura 6.

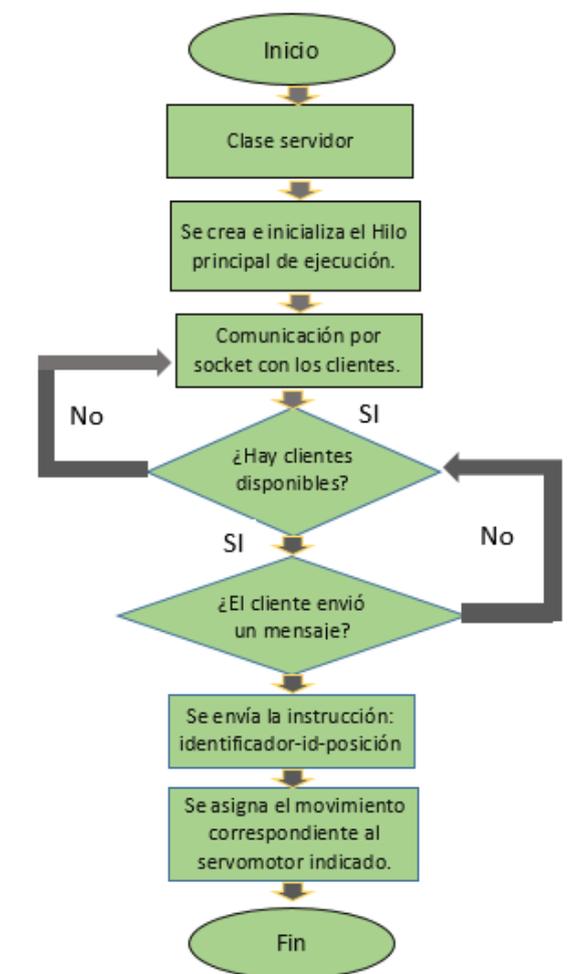


Figura 6. Diagrama de flujo programa servidor en la BeagleBone Black.

2.5 Generación de secuencias

La interfaz de Blender en modo de animación permite realizar secuencias de movimientos de los brazos del robot, cambiando el ángulo de rotación de los grados de libertad con los que cuenta el robot (como si fuera un simulador), esto nos permite realizar las pruebas para definir las limitaciones y alcances de los movimientos del robot. Una vez establecidos, se puede recrear o realizar la manipulación en tiempo real mediante el motor de videojuegos de Blender (Blender Game Engine).

Cada uno de los servomotores con los que cuenta el robot, tienen un rango de rotación entre $0^\circ - 300^\circ$, la posición por de default es de 150° permitiendo rotarla 150° hacia ambos sentidos.

Tabla 2. Comandos para manipular el robot mediante Blender.

Comando	Instrucción	Comando	Instrucción
Q	$q_1 += 10^\circ$	T	$q_4 += 10^\circ$
A	$q_1 -= 10^\circ$	G	$q_4 -= 10^\circ$
W	$q_2 += 10^\circ$	Y	$q_5 += 10^\circ$
S	$q_2 -= 10^\circ$	H	$q_5 -= 10^\circ$
E	$q_3 += 10^\circ$	U	$q_6 += 10^\circ$
D	$q_3 -= 10^\circ$	J	$q_6 -= 10^\circ$
R	$tenaza_{izq} += 10^\circ$	I	$tenaza_{der} += 10^\circ$
D	$tenaza_{izq} -= 10^\circ$	K	$tenaza_{der} -= 10^\circ$

2.6 Pruebas y resultados.

La última etapa del proyecto consistió en realizar pruebas del sistema de telecontrol completo, es decir, se realizaron las pruebas de telemando manipulando las articulaciones del robot humanoide a través de la interfaz desarrollada en BGE, comprobando la réplica de los movimientos del sistema esclavo-maestro así como el análisis del sistema de comunicación, determinando las posibles mejoras para el sistema y la interfaz de telecontrol para un trabajo futuro.

3. Análisis de resultados

En la figura 7 podemos observar las pruebas que se realizaron con la interfaz en Blender funcionando y con la parte del torso del robot humanoide, que cuenta con 4 grados de libertad por brazo y replicalos movimientos que se le van asignando en la animación en Blender, como podemos notar.

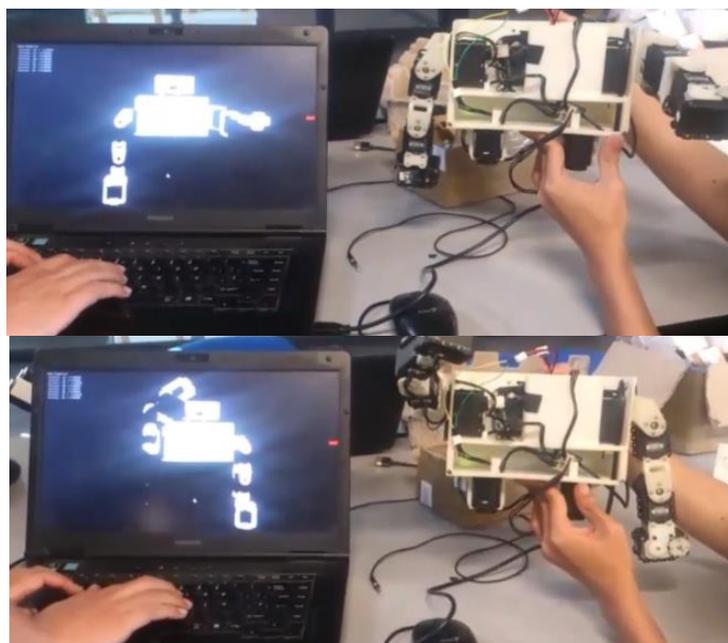


Figura 7. Pruebas con la comunicación entre la interfaz en Blender con el torso del robot

4. Conclusiones

Con ayuda de la interfaz desarrollada en Blender, se han generado secuencias de movimiento en el torso del robot previamente simulado en el software de animación, así como también se hizo uso de comunicación en tiempo real para generar movimientos en el mismo. La comunicación se realizó mediante una función implementando sockets, en la cual la interfaz en Blender trabaja como cliente y la tarjeta de desarrollo como servidor.

Referencias

- [1] C.I. Penaloza, Y. Mae, K. Ohara, and T. Arai, "Social human behavior modeling for robot imitation learning", *Mechatronics and Automation (ICMA), 2012 International Conference on*, vol., no., pp. 457-462, Agosto. 5, 2012.
- [2] S. Hak, N. Mansard, O. Ramos, L. Saab, and O. Stasse, "Capture, recognition and imitation of anthropomorphic motion", *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, vol., no., pp. 3539-3540. Mayo 1, 2012.
- [3] Turner, M. L., Findley, R. P., Griffin, W. B., Cutkosky, M. R., & Gomez, D. H. 2000. Development and testing of a telemanipulation system with arm and hand motion. In *ASME IMECE Symp. on Haptic Interfaces*.
- [4] Blender Foundation. (1995). Información técnica. Agosto 20, 2015, de Blender, Sitio web: <https://www.blender.org/>.
- [5] Fu, K. S. (1988). *Robótica, Control, detección, visión e inteligencia*. McGraw-Hill Interamericana de España
- [6] Martínez, V. J. & Sabater, N. J. M. (2012). *Guía Docente para el Diseño de Robots de Servicio*. Madrid, España.
- [7] Dassault Systèmes SolidWorks Corp. (1993). Consulta general. Agosto 21, 2015, de SolidWorks Sitio web: http://www.solidworks.es/sw/183_ESN_HTML.htm
- [8] ROBOTIS. (1999). Consulta técnica servomotores Dynamixel AX-12, AX-18. Agosto 19, 2015, de ROBOTIS Sitio web: <http://en.robotis.com/index/>
- [9] The BeagleBoard.org Foundation. (2013). Consulta técnica. Agosto 20, 2015, de Texas Instruments Sitio web: <http://beagleboard.org/>
- [10] Bruyninckx, H. 2004. Blender for robotics and robotics for Blender. Dept. of Mechanical Engineering. KU Leuven, Belgium.