

# Análisis dinámico de un robot Hexápodo e implementación en FPGA

Borghi Orué, Emiliano Javier

*emiliano.borghi@gmail.com*

El presente trabajo práctico analiza el comportamiento dinámico de un robot Hexápodo dividiendo el informe en dos partes principales. Primero se estudiará dinámicamente una pata usando la formulación de Lagrange-Euler, y luego, se presentará un método de manejo articular utilizando un FPGA.

## 1. Modelo dinámico del Hexápodo

El estudio de la siguiente sección estará subdividido en:

- cálculo teórico de las ecuaciones dinámicas
- simulación de los movimientos ante distintas trayectorias

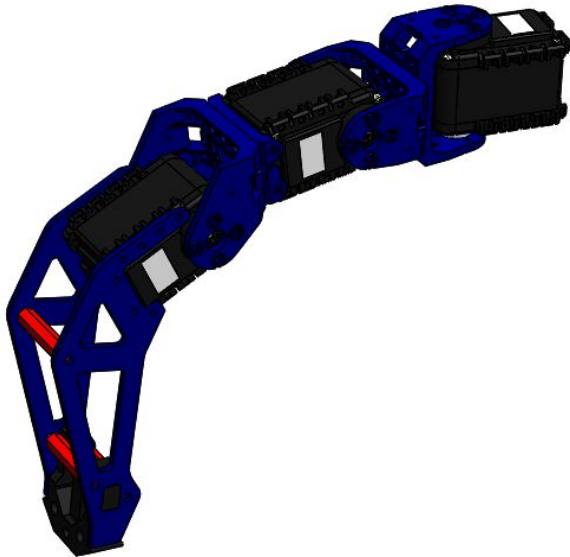
El robot elegido es uno comercial llamado **PhantomX Hexapod Mark II**.



## 1.1. Introducción teórica

En la presente sección se obtendrán las ecuaciones dinámicas de una pata de un robot hexápodo usando la formulación de Lagrange-Euler. Se ha optado por desarrollar únicamente una sola extremidad debido a la elevada complejidad que presenta realizar el análisis completo.

Los **robots móviles** son sistemas mecatrónicos muy complejos, donde sus extremidades, con varios grados de libertad (DOF) están conectadas entre sí a través de su cuerpo. Además, un robot hexápodo en movimiento, en cualquier instante, es una combinación compleja de cadenas cinemáticas abiertas y cerradas. Las patas del robot, cuando están en contacto con el piso, forman una cadena cerrada con dicha superficie de contacto y su propio cuerpo, mientras que las patas que no están haciendo contacto alguno, se comportan como cadenas cinemáticas abiertas.

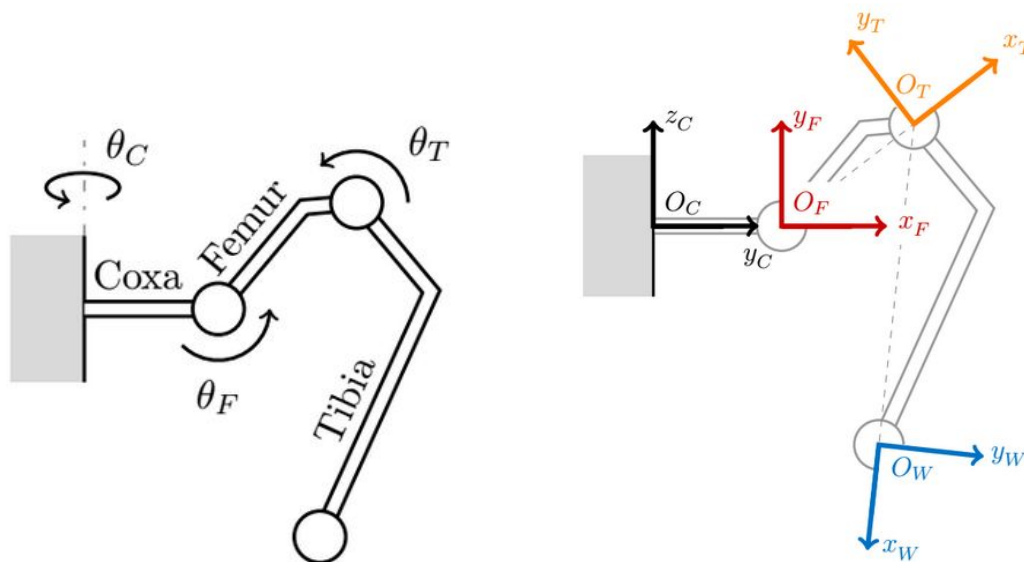


Para obtener estas ecuaciones se utilizó un modelo CAD realizado en **Solidworks 2015** por Pedro Larez [1]. La imagen muestra la pata completa del robot.

La formulación de Lagrange-Euler propuesta por Uicker (1965) utiliza la representación de Denavit-Hartenberg basada en matrices de representación homogénea.

A continuación se explicará el código empleado para obtener las ecuaciones dinámicas.

**L-E 1:** Asignar a cada eslabón un sistema de referencia de acuerdo a las normas D-H [2].



L-E 2: Obtener matrices de transformación  ${}^0A_i$  para cada elemento  $i$ .

La siguiente tabla muestra los parámetros DH para la pata media izquierda.

$i$	$\alpha_i$ [rad]	$a_i$ [mm]	$\theta_i$ [rad]	$\theta_i^{\text{off}}$ [rad]	$d_i$ [mm]
C (coxis)	$\pi/2$	52	$\theta_C$	$\pi/2$	0
F (fémur)	0	66	$\theta_F$	-0.22	0
T (tibia)	0	130	$\theta_T$	-0.59	0

L-E 3: Obtener las matrices  $U_{ij}$  definidas por:

$$U_{ij} = \frac{\partial {}^0A_i}{\partial q_i}$$

L-E 4: Obtener las matrices  $U_{ijk}$  definidas por:

$$U_{ijk} = \frac{U_{ij}}{\partial q_k}$$

L-E 5: Obtener las matrices de pseudoinercias  $J_i$  para cada elemento, que vienen definidas por:

$$J_i = \begin{bmatrix} \int x_i^2 dm & \int x_i y_i dm & \int x_i z_i dm & \int x_i dm \\ \int y_i x_i dm & \int y_i^2 dm & \int y_i z_i dm & \int y_i dm \\ \int z_i x_i dm & \int z_i y_i dm & \int z_i^2 dm & \int z_i dm \\ \int x_i dm & \int y_i dm & \int z_i dm & \int dm \end{bmatrix}$$

donde las integrales están extendidas al elemento  $i$  considerado, y  $(x_i, y_i, z_i)$  son las coordenadas del diferencial de masa  $dm$  respecto al sistema de coordenadas del elemento.

L-E 6: Obtener la matriz de inercias  $D=[d_{ij}]$  cuyos elementos vienen definidos por:

$$d_{ij} = \sum_{K=(\max i,j)}^n \text{Traza}(U_{kj} J_k U_{ki}^T)$$

con  $i,j=1,2,\dots,n$  (DOF)

L-E 7: Obtener los términos  $h_{ikm}$  definidos por:

$$d_{ikm} = \sum_{j=(\max i,k,m)}^n \text{Traza}(U_{jkm} J_j U_{ji}^T)$$

L-E 8: Obtener la matriz columna de fuerzas de Coriolis y centrípeta  $H=[h_i]^T$  cuyos elementos vienen definidos por:

$$h_i = \sum_{k=1}^n \sum_{m=1}^n h_{ikm} \dot{q}_k \dot{q}_m$$

con  $i,k,m=1,2,\dots,n$

L-E 9: Obtener la matriz columna de fuerzas de gravedad  $C=[c_i]^T$  cuyos elementos están definidos por:

$$c_i = \sum_{j=1}^n -m_j g U_{ji}^T \mathbf{r}_j$$

con  $i=1,2,\dots,n$

$g$ : vector de gravedad expresado en el sistema base  $\{S_0\}$  y viene expresado por  $\{g_{x0}, g_{y0}, g_{z0}, 0\}$

$\mathbf{r}_j$ : vector de coordenadas homogéneas del centro de masa del elemento  $j$  expresado en el sistema de referencia del elemento  $i$ .

L-E 10: La ecuación dinámica del sistema será:

$$\tau = D \ddot{q} + H + C$$

donde  $\tau$  es el vector de fuerzas y pares motores efectivos aplicados a la coordenada  $q$ .

Una vez obtenidas las ecuaciones de torques que permiten dimensionar los motores a utilizar, se puede realizar el control dinámico del robot o de la pata en cuestión, pero éste presenta un inconveniente particular.

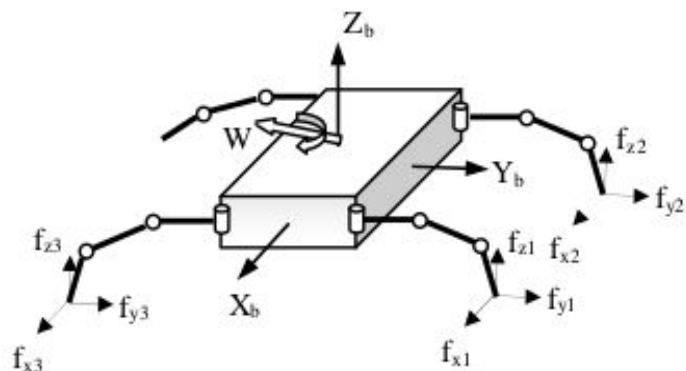
El problema básico de controlar las cadenas cinemáticas del robot radica en su coordinación. A esto se suma el problema de la coordinación local que consta en

controlar las articulaciones de cada pata y así controlar de manera efectiva los extremos del robot.

Respecto a la dinámica, existe un problema de coordinación global que involucra todas las extremidades del hexápodo. Las fuerzas y momentos se propagan a través de las cadenas cinemáticas a través de sus extremidades, produciendo un acoplamiento dinámico.

Cuando la pata se encuentra en el aire, no existe contacto alguno con el piso, entonces la fuerza aplicada al extremo es nula ( $\mathbf{f} = \mathbf{0}$ ). Sin embargo, durante la fase de soporte, existe contacto con la superficie por lo que la ecuación anterior se vuelve *indeterminada*, por lo que tiene que resolverse por alguno de los siguientes métodos:

- Usando multiplicadores de Lagrange para minimizar alguna función energética.
- Modelando la interacción pata/terreno.
- Usando sensores de fuerza para medir  $\mathbf{f}$ .



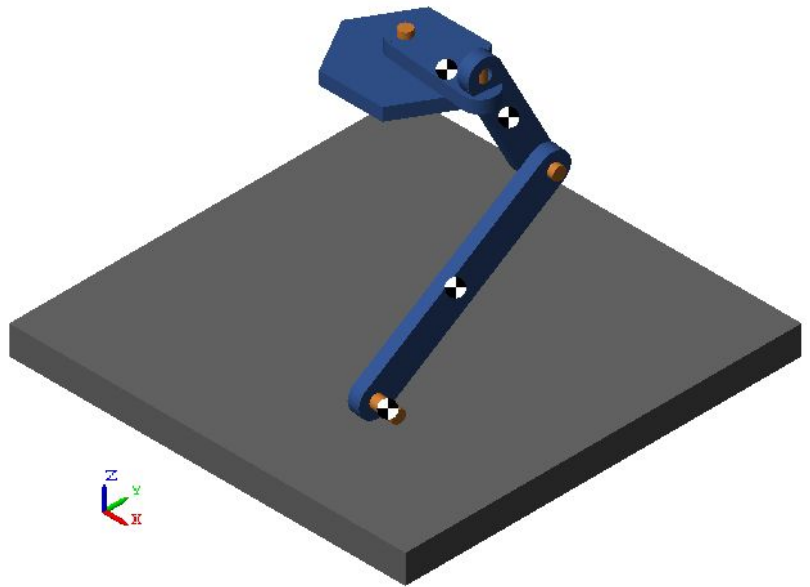
Fuerzas de contacto actuantes sobre un robot cuadrúpedo

Para evitar realizar estos cálculos, en la siguiente sección se simulará un diseño simple en 3D de una de las extremidades de un Hexápodo y así dimensionar los motores, lo cual es el objetivo principal del informe.

## 1.2. Implementación en SimMechanics

Para diseñar algoritmos que controlen robots móviles de muchas patas es importante tener buenos modelos que describan el comportamiento del mismo [3].

Para analizar la dinámica de una pata del robot Hexápodo se utilizó el toolbox **SimMechanics** de Simulink. El segmento analizado se muestra en la imagen donde se observan, además, los centros de masa (CoM) de los eslabones y el sistema de referencia global empleado.



SimMechanics es un toolbox que posee bloques de Simulink especiales para implementar enlaces mecánicos, actuadores, sensores, etcétera, y permite de manera muy sencilla realizar análisis cinemático y dinámico de un robot, observar características esenciales como la posición, velocidad, torque y demás parámetros de las articulaciones.

Otra característica importante es que pueden implementarse sistemas de control para el robot sin la necesidad de calcular las ecuaciones del robot de manera directa ya que, de eso, se encarga el programa.

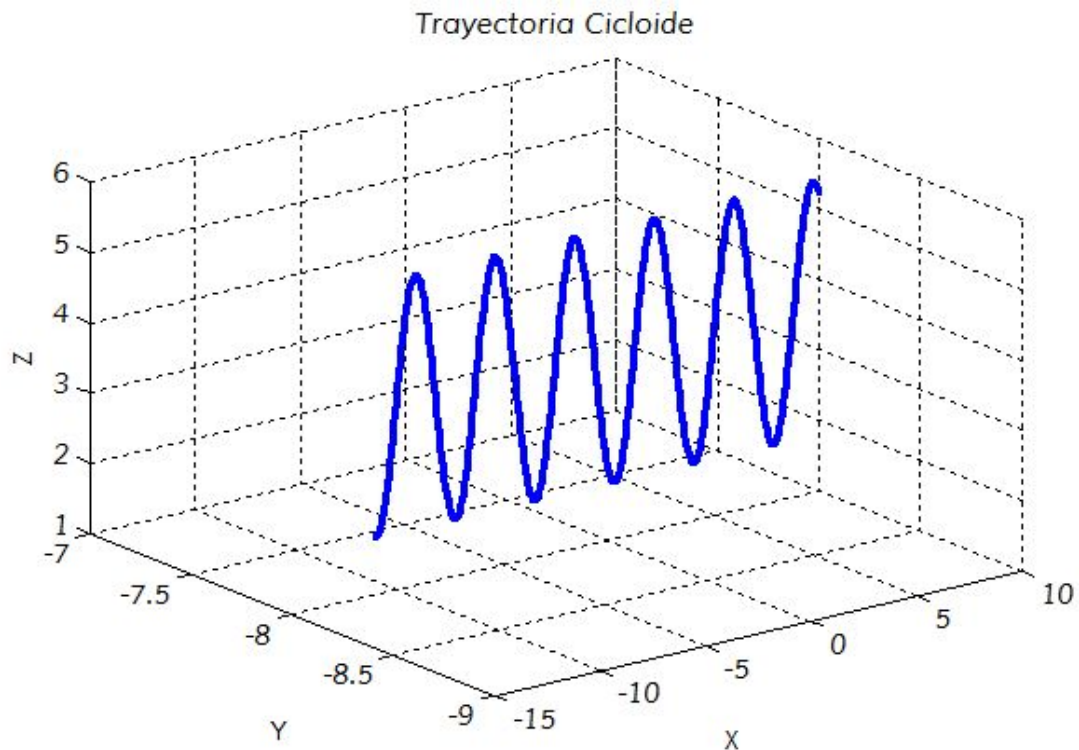
Como puede observarse en la imagen de la página siguiente, el diagrama de Simulink consta de:

- En la mitad superior están todos los bloques que representan los enlaces y los actuadores de revolución de la pata, como así también el piso.
- El bloque más grande (6-DOF Joint) relaciona la posición del extremo de la pata respecto del sistema de referencia. Esto permite aplicarle trayectorias de referencia y analizar los movimientos generales de dicha extremidad.
- Los bloques ubicados a la izquierda del mencionado anteriormente son las trayectorias de referencia que se le asignan al extremo.
- Los bloques de la derecha, en cambio, miden algunos parámetros como posición, velocidad y fuerza del extremo de la pata.

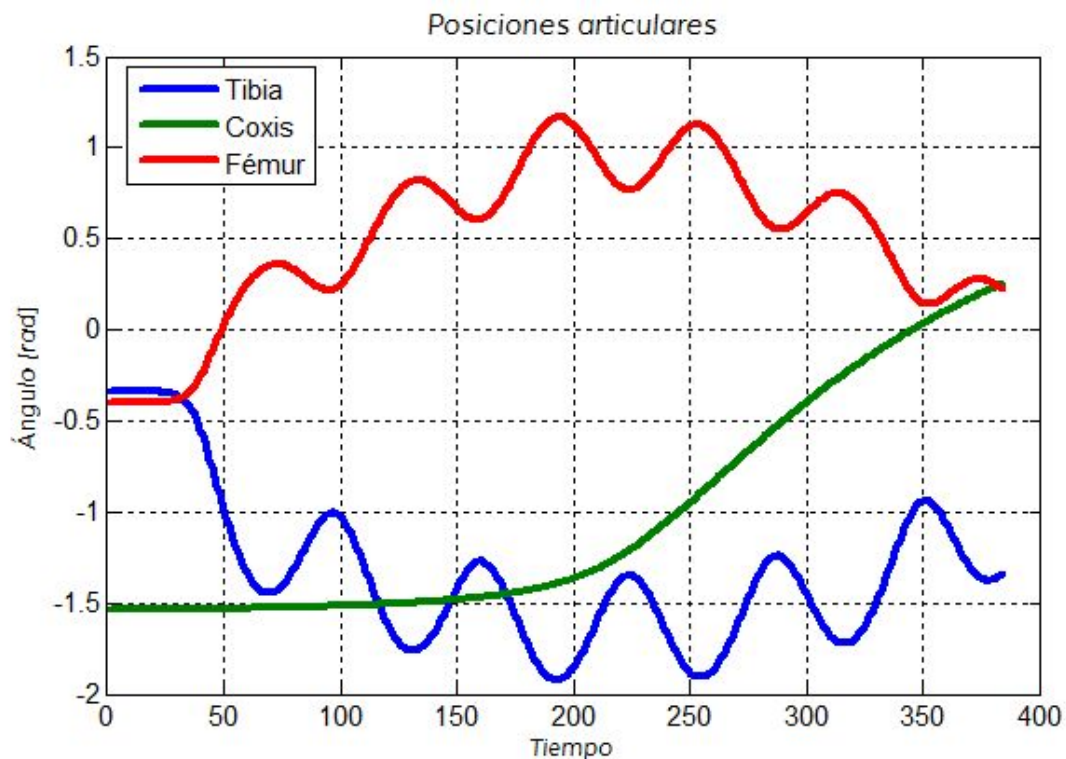




A continuación se muestra la trayectoria utilizada para analizar la dinámica del robot.

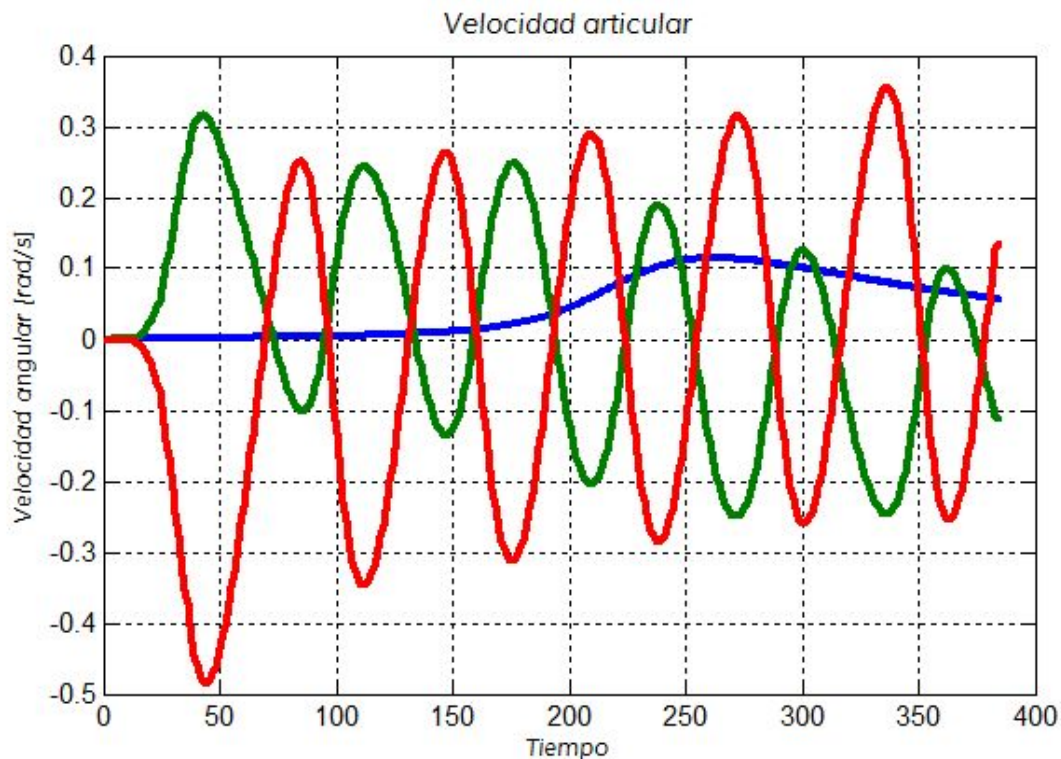


Las posiciones que coordinadas articulares ( $\theta$ ) que deben seguir las articulaciones son calculadas automáticamente por el programa. A continuación se grafican tales cálculos.





De lo anterior se puede observar un movimiento progresivo de la articulación del Coxis, mientras que las articulaciones de la Tibia y el Fémur, siguen de manera similar el movimiento de la trayectoria propuesta.

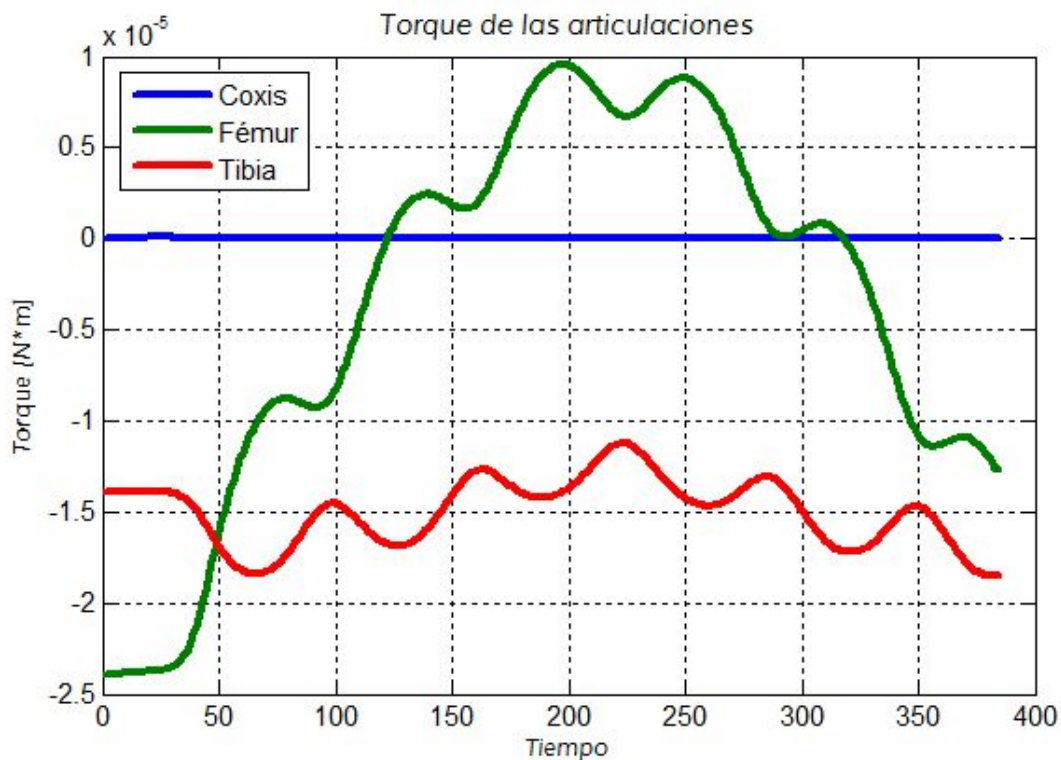


La imagen anterior muestra cómo varían drásticamente las velocidades de las articulaciones. Esto podría mejorarse usando un sistema de control (PID, control difuso o Aprendizaje por Refuerzo), aplicando algún criterio de optimización (energía).

El torque aplicado por las articulaciones es una de los parámetros más importantes del informe, sin embargo, esta gráfica no es del todo exacta debido a que no se han modelado las fuerzas de contacto que se ejercen cuando la pata se encuentra en la fase de soporte.

Si se hubieran considerado estas fuerzas, se podrían dimensionar de manera más exacta los motores empleados.

Este fenómeno se puede observar claramente en el torque que aplica la articulación del Coxis, la cual no requiere esfuerzo alguno en ninguna parte de la trayectoria.



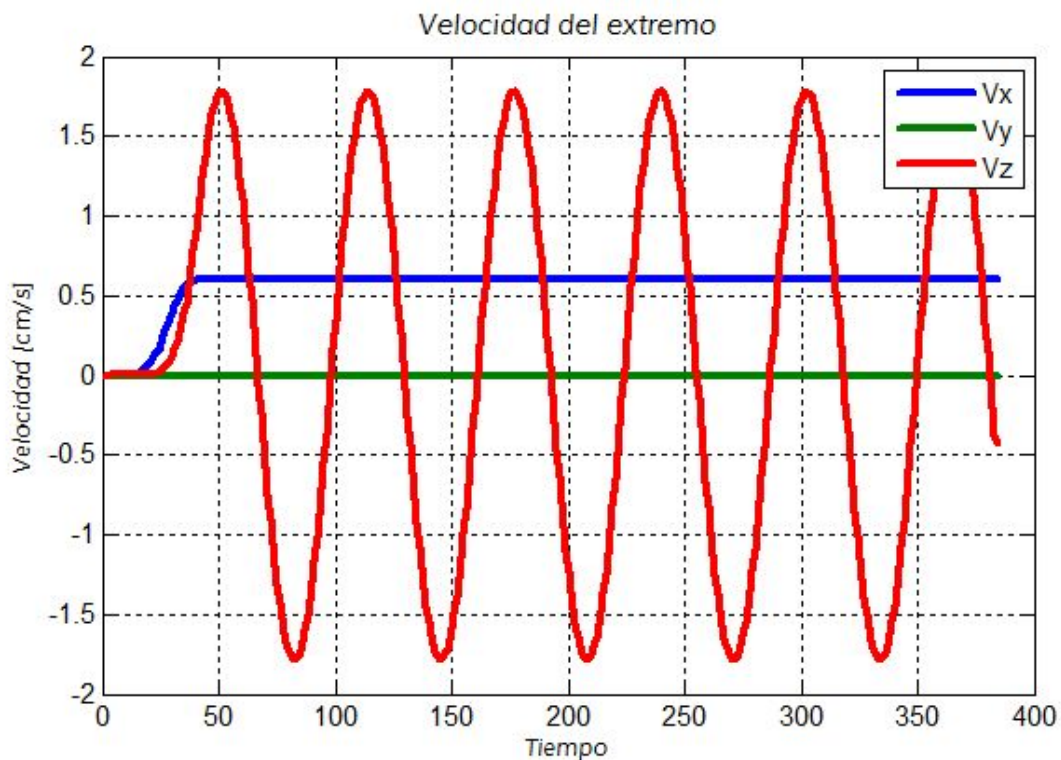
Un servomotor comercial, utilizado para Hexápodos, posee un torque de entre 5 kg\*cm y 20 kg\*cm.

El robot analizado (*PhantomX Hexapod Mark II*) posee 18 servomotores Dynamixel AX-12A (15,3 kg\*cm) o AX-18A(18,3 kg\*cm).

Para utilizar la misma nomenclatura que en el gráfico, el robot simulado necesitaría servomotores de  $1,02 \cdot 10^{-4}$  kg\*cm.

Este valor también puede deberse a que no están consideradas las masas de los servomotores y que las partes del robot analizadas como si hubieran sido construidas por una impresora 3D, las cuales utilizan un plástico conocido como **PLA** o **Poliácido Láctico**, y que tiene una densidad de 1,3 kg/m<sup>3</sup>.

La siguiente gráfica muestra las tres componentes de velocidad del extremo de la pata del Hexápodo. Como puede verse, la componente en y es cero debido a que la trayectoria en ese sentido es constante. Para la componente en x, el transitorio que se origina al principio es producido por la ubicación del inicio de la trayectoria.



### 1.3. Comparación con V-REP

Para constatar las pruebas anteriores se mostrarán simulaciones con un simulador para robótica llamado **V-REP** de Coppelia Robotics.

V-REP, para analizar la dinámica, utiliza una librería física de código libre llamada **Bullet Physics Library** [4]. La precisión de las siguientes simulaciones se ha dejado por defecto a una velocidad  $dt=50ms$ .

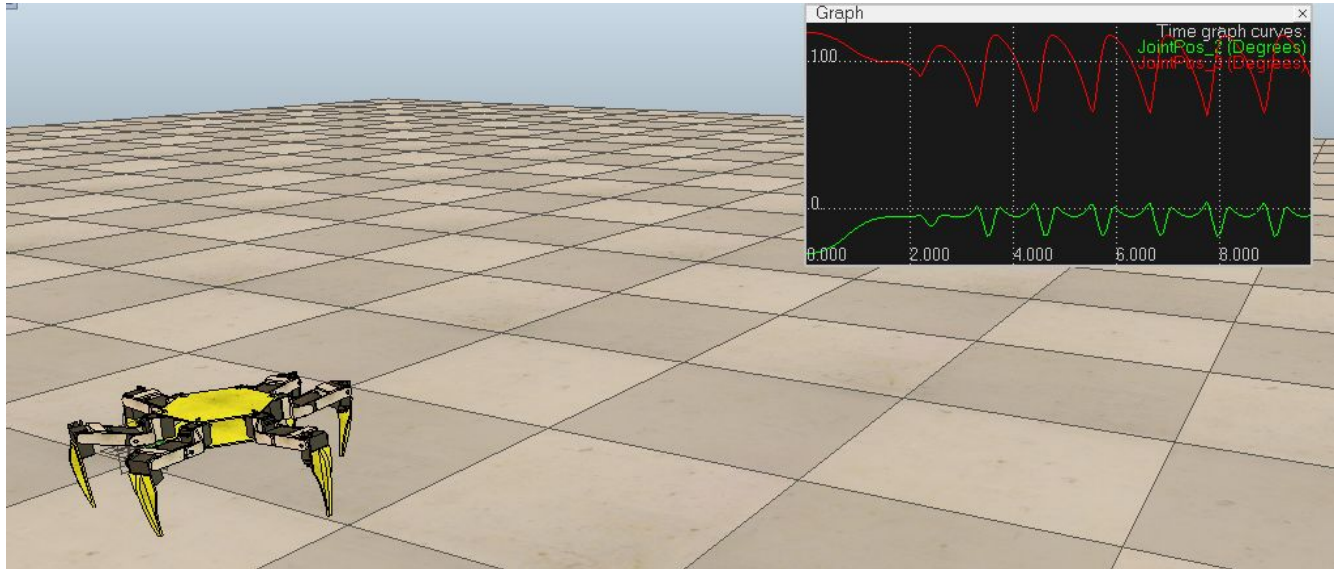
Las siguientes suposiciones se han hecho en orden de simplificar el análisis dinámico del robot.

- (a) El robot se mueve en línea recta sobre una superficie plana usando una marcha tipo Trípode.
- (b) El cuerpo se mantiene a una altura constante, paralela al plano del piso durante el movimiento.

La simulación necesitó de algunos parámetros, que a continuación se indican.

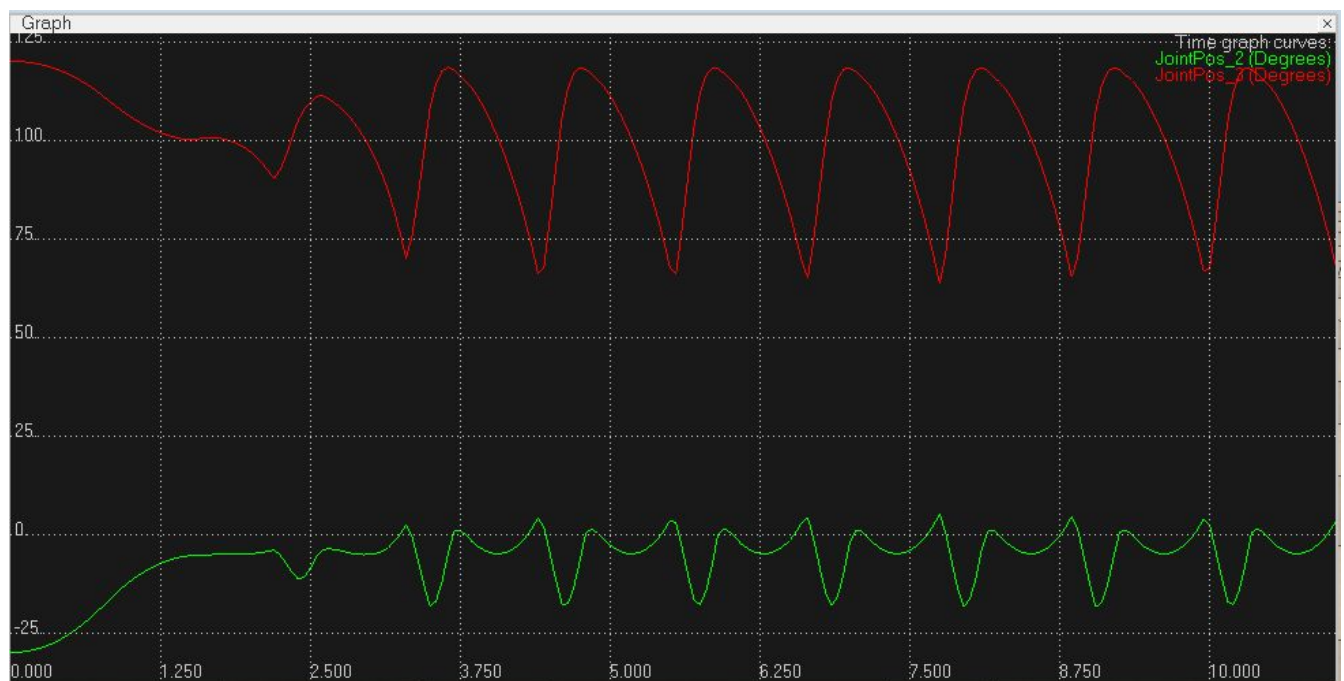
- Tiempo de ciclo ( $T$ ) = 2s
- Longitud de zancada ( $\lambda$ ) = 0.16m
- Altura de zancada = 0.04m
- Velocidad de marcha = 0.9 m/s

A continuación se observa una impresión de pantalla durante la simulación.



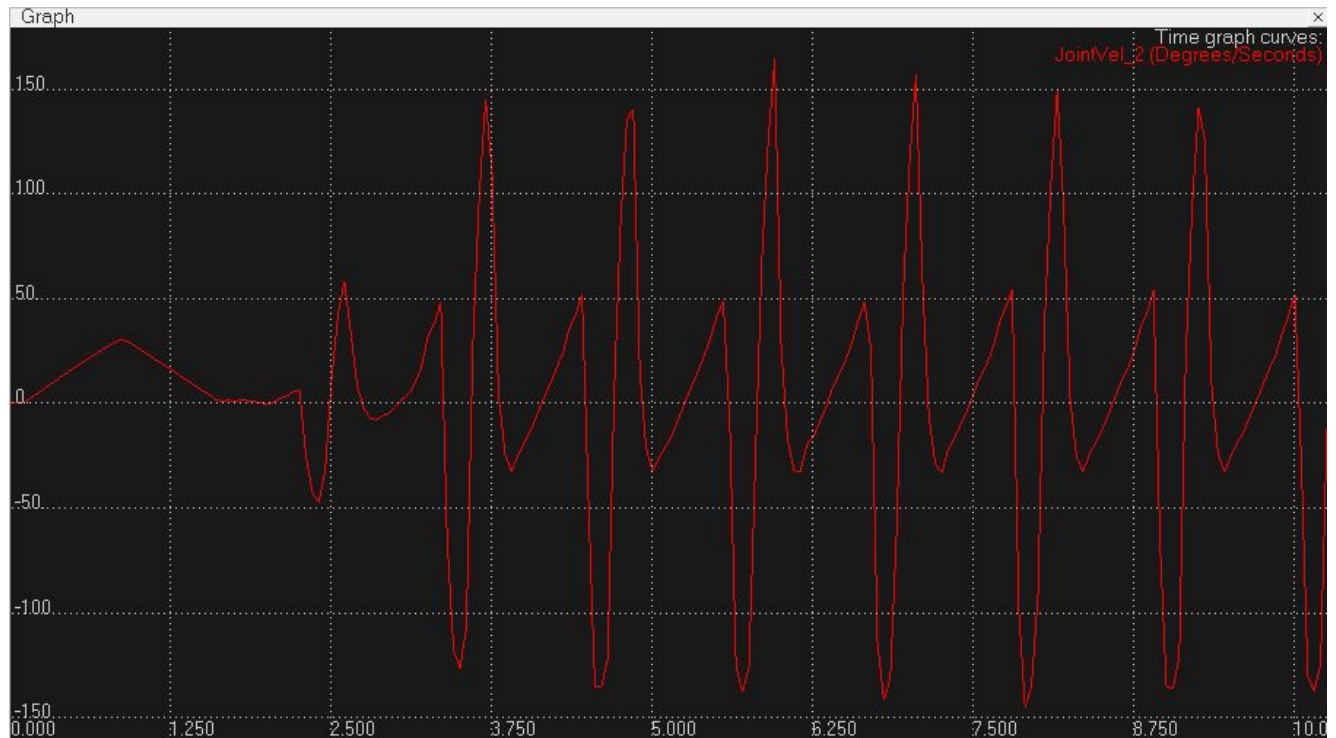
El diseño en 3D es propiedad de Lyan Randell.

Y luego, los gráficos obtenidos.



Posiciones angulares de las articulaciones del fémur y la tibia, en grados.





Velocidad angular de la articulación del fémur, en grados/segundo.

A diferencia de las simulaciones implementadas en SimMechanics, las de V-REP poseen una gran cantidad de parámetros que mejoran el realismo en las simulaciones, pero contrapartida poseen herramientas más complicadas para el manejo y análisis dinámico del robot.

## 2. Manejo articular con un FPGA

El robot hexápodo utilizará **servomotores**, accionados mediante señales PWM a través de un **FPGA**.

### 2.1. Introducción teórica

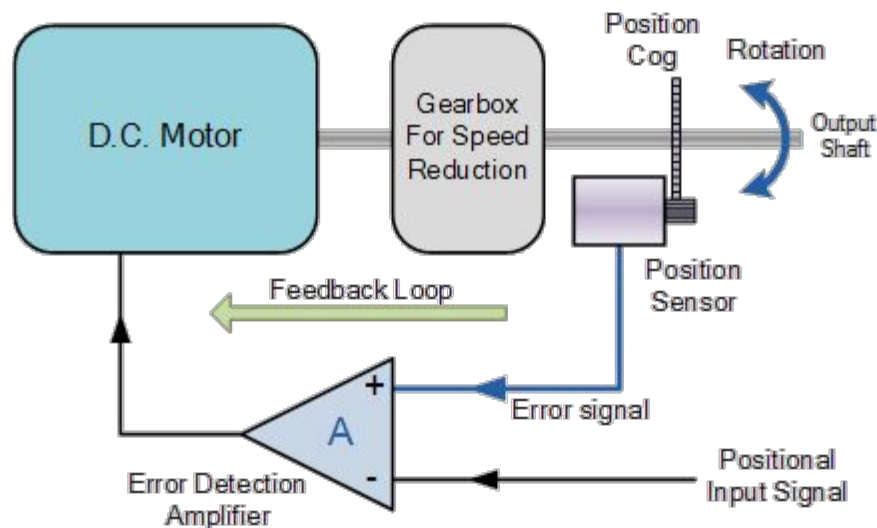
#### 2.1.1. Servomotor

Un servomotor es un actuador rotacional que permite un control preciso de posición, velocidad y aceleración. Básicamente consiste en un motor acoplado con un sensor que realimenta la posición del eje.

Estos tipos de motores, suelen usarse en robótica por su bajo costo y liviano peso.



A continuación se presenta un diagrama en bloque del mismo.



Puede verse que el lazo de realimentación está dado por la posición del eje, el cual se compara con la señal PWM de entrada.

A la salida posee una caja de engranajes para reducir la velocidad y aumentar el torque.

El servomotor es un sistema con un mecanismo de control a lazo cerrado incorporado. Este control se produce realimentando la señal de salida y la de referencia de entrada. Cuando el motor alcanza la posición deseada, no habrá ninguna diferencia entre la señal sensada y la referencia.

Para poder controlar un servomotor es necesario proporcionar una señal PWM de período fijo (normalmente de 10 ó 20 ms) y un ciclo de actividad proporcional a la posición angular deseada (entre 0.7 y 2.5 ms, dependiendo del fabricante).

### 2.1.2. FPGA

Un FPGA (del inglés *Field-Programmable Gate Array*) es un circuito integrado diseñado para ser configurado por el programador según las necesidades mediante un lenguaje de descripción de hardware (HDL).

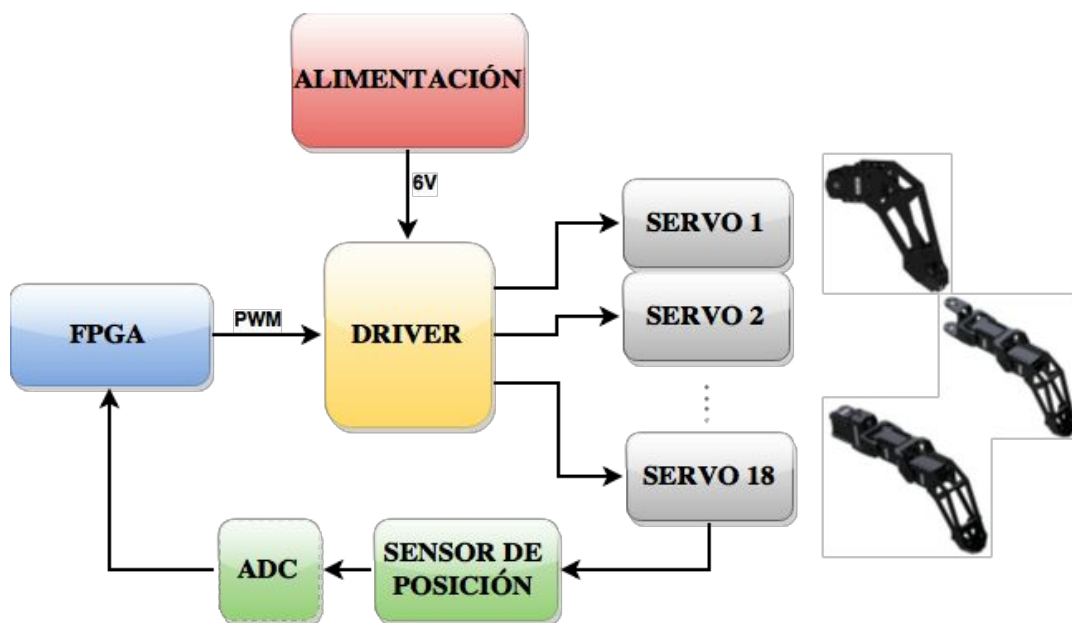
Un FPGA contiene un array de bloques lógicos programables con conexiones reconfigurables que permiten realizar funciones muy complejas. Algunos de estos dispositivos también contienen bloques de memoria.



La implementación de este sistema con una FPGA, en lugar de un microcontrolador, se justifica en que se está controlando cada motor por hardware, con un tiempo de respuesta mucho menor que el de un microcontrolador. Este tiempo es vital debido a que el robot debe recalcular sus variables y obtener un sistema que actúe rápidamente es de vital importancia.

## 2.2. Hardware

Para poder controlar los servomotores se necesita lo siguiente:



Para poder lograr un robot móvil sin cables y de liviano peso, para alimentarlo es recomendable utilizar baterías de 6 ó 7,2V de alta capacidad. Y en caso de que se necesite disminuir la tensión, conviene utilizar un **UBEC**, el cual es simplemente un regulador switching con capacidad para suministrar gran cantidad de corriente por períodos prolongados y aún mucha mayor corriente en el encendido de los motores. Debe tenerse en cuenta que en ese momento, los servos suelen consumir alrededor de 1A cada uno.

El driver puede ser un optoacoplador con salida a transistor o Darlington para aumentar la corriente y ajustarlo a la tensión requerida, pero principalmente para aislar la etapa digital ante fuerzas contraelectromotrices.

Para sensar la posición basta con utilizar un encoder (donde es necesario utilizar un ADC junto con una etapa de filtro de ruido y anti aliasing). Algunos servomotores poseen incorporado un sensor de posición, pero son muy costosos. Existen alternativas baratas, como es la implementación de servos para aeromodelismo. Estos motores, para poder leer su posición, es necesario realizarle algunas



modificaciones internas en su mecanismo, si es que se quiere evitar utilizar un encoder. Otro sensor que puede agregarse para el control de la dinámica, es un sensor de fuerza en la extremidad de cada pata.

Realizar un control de posición en un servomotor es necesario si es que se desea tener un conocimiento exacto de esta variable en todo momento. Este parámetro también es importante debido a que puede saberse si el motor se encuentra estancado o sobrecargado.

## 2.3. Software

Para programar las señales PWM del FPGA se utilizó el **ModelSim PE Edición Estudiantil**. A continuación se explicarán las partes más importantes del programa.

El principio de funcionamiento básico para la generación de la señal PWM consiste en el uso de un registro que almacena un valor que se cargará en un contador creciente/decreciente cada vez que termine una cuenta la cual determinará el ancho del pulso [6].

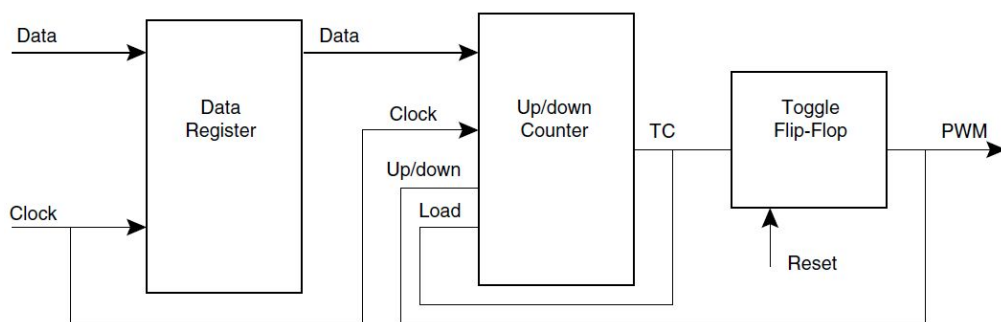


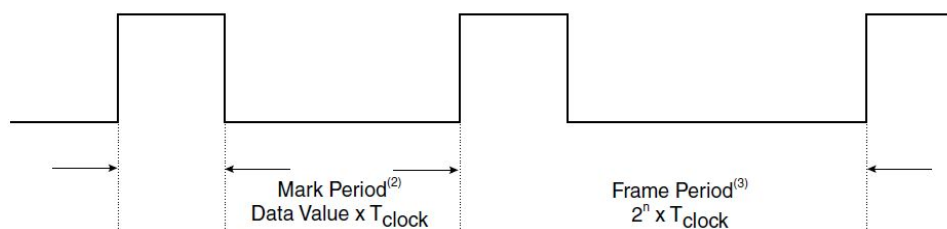
Diagrama en bloques del funcionamiento

Un registro de datos es utilizado para almacenar el valor del contador, el cual determinará la posición del servomotor. Un Flip-Flop T (Toggle) será el encargado de generar la señal PWM.

Cuando el dato se carga, el contador comenzará a decrementarse, desde el valor hasta cero. Durante esta fase, la señal PWM permanece en un nivel bajo. Cuando el contador llega a cero, la señal pasa al estado alto. El dato es recargado al nuevo valor, repitiendo el ciclo.

El ciclo de actividad de la señal PWM es controlado con el dato que se le envía al contador. Mientras más grande es el dato, mayor será el ciclo de actividad.

Los servomotores convencionales utilizan ciclos de actividad entre 3,5 y 12,5% si el período es de 20 ms o entre 7 y 25% si es de 10 ms.



Forma de onda de una señal PWM (1)

#### NOTAS:

- (1) El ciclo de actividad se calcula como el cociente entre la duración del pulso y el período.  
 $D = (1 - \tau) / T = \text{Data} / 2^n$ .
- (2)  $1 - \tau = \text{Data} * T_{\text{clock}}$
- (3)  $T = T_{\text{clock}} * 2^n$ , donde  $n$  es la longitud del contador binario.

### 3. Posibles mejoras

Una de las mejoras imprescindibles a lograr es la implementación de la fuerza de contacto de la pata con el suelo, para poder dimensionar de manera mucho más exacta a los servomotores.

Otra mejora sustancial sería la de lograr generar la trayectoria deseada a partir de la utilización de las ecuaciones cinemáticas inversas del robot, con lo que, se evitaría tener que definir a mano la trayectoria para que no genere una singularidad durante la simulación. Esto podría hacerse a partir del trabajo realizado en el trabajo práctico N°1: *Implementación de una Matriz Cinemática en DSP para un robot Hexápodo*.

Algo muy útil de implementar a la hora de la simulación, debido a que se pueden implementar conceptos avanzados, sería utilizar sistemas de control para posicionar de manera rápida y eficiente las articulaciones teniendo como objetivo, por ejemplo, la minimización del consumo de energía. Esto puede implementarse con controladores PID básicos o procesos más complejos como los controladores difusos o Aprendizaje por Refuerzo.

Debido a la complejidad inicial que presentaba analizar una pata sola, no se analizó el robot por completo. En cambio, para un futuro, se podría observar cómo afecta el cuerpo (y la marcha elegida) a la dinámica del robot.

Para agregar complejidad aún más superior, el robot podría incorporar conceptos de estabilidad dinámica, lo que implicaría modificar su centro de gravedad (CoG) o también permitir rotar su cuerpo y modificar su altura durante la marcha elegida.

### 4. Conclusiones finales

Se obtuvieron las ecuaciones dinámicas del robot y se simuló en un entorno 3D que

permitió obtener parámetros de velocidad y torque para dimensionar los motores a utilizar, sin necesidad de realizar cálculos complejos ni emplear sistema de control alguno.

Este análisis abre una rama de posibilidades de estudio definidas en la sección anterior.

## 5. Referencias

[1] <https://grabcad.com/library/hexapod-11>

[2] *Design of Motion Primitives for a Hexapod Walking Robot Operation in a Rough Environment*. Jakub Mrva. Master's Thesis. Czech Technical University in Prague. Faculty of Electrical Engineering. Department of Cybernetics. May 2014.

[3] *Modeling and Simulation of Wave Gait of a Hexapod Walking Robot: A CAD/CAE Approach*. International Journal of Robotics and Automation (IJRA) Vol. 2, No. 3, September 2013, pp. 104~111. ISSN: 2089-4856

[4] <http://bulletphysics.org/>

[5] Atmel PWM - 2324A-10/01. Nota de aplicación.

# Índice

1. Modelo dinámico del Hexápodo
  - 1.1. Introducción teórica
  - 1.2. Implementación en SimMechanics
  - 1.3. Comparación con V-REP
2. Manejo articular con un FPGA
  - 2.1. Introducción teórica
    - 2.1.1. Servomotor
    - 2.1.2. FPGA
  - 2.2. Hardware
  - 2.3. Software
3. Posibles mejoras
4. Conclusiones finales
5. Referencias