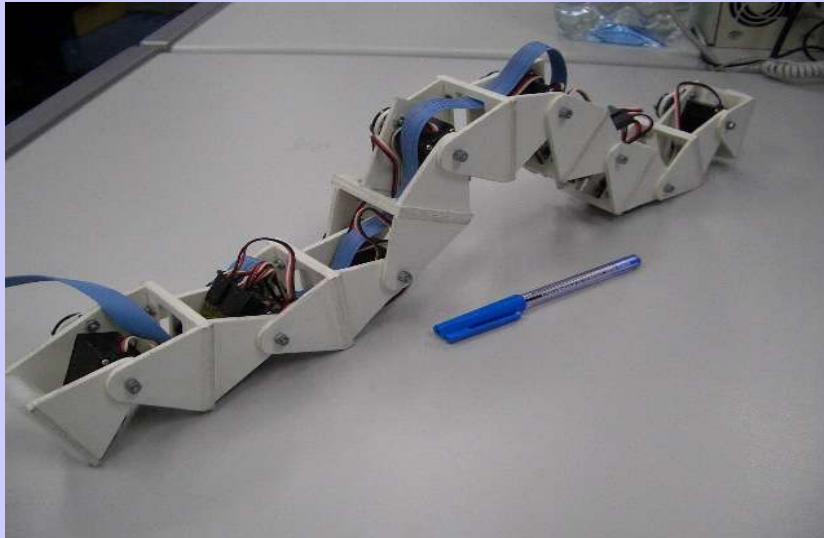


## Sesión III: Robótica e investigación

### Diseño de robots ápodos: *Cube Revolutions*



Juan González Gómez

Escuela Politécnica Superior  
Universidad Autónoma de Madrid

# ÍNDICE



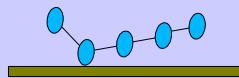
- Introducción



- Cube Revolutions "a fondo"



- Locomoción



- Simulación

Hypercube

- Líneas de investigación

# Introducción

# Robótica

## Dos grandes áreas:

- **Manipulación:** Capacidad de modificación de los objetos del entorno



Robótica Industrial

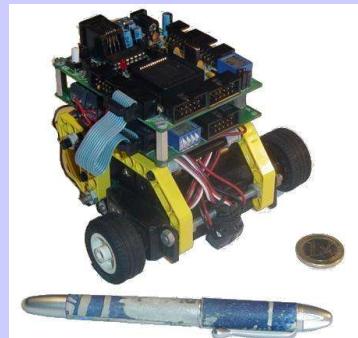
- **Locomoción:** Facultad para desplazarse de un lugar a otro



Robótica Móvil

# Clasificación de Robots móviles

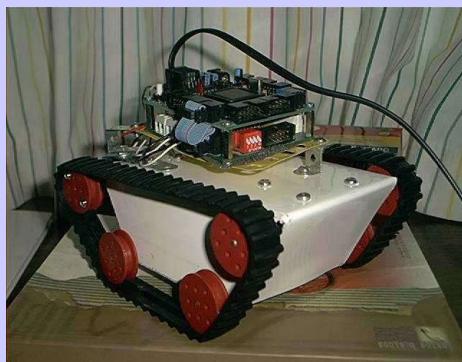
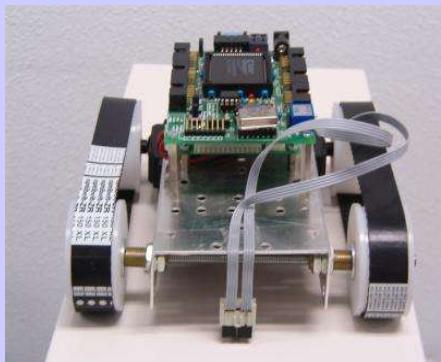
- Con ruedas:



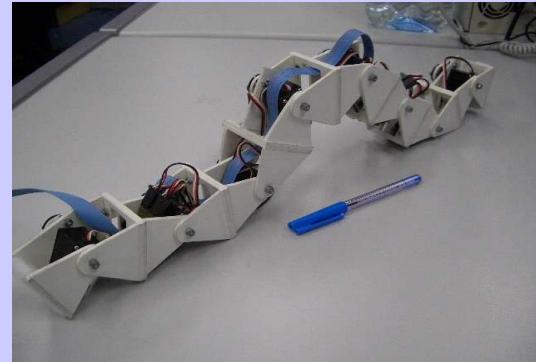
- Con patas:



- Con orugas:



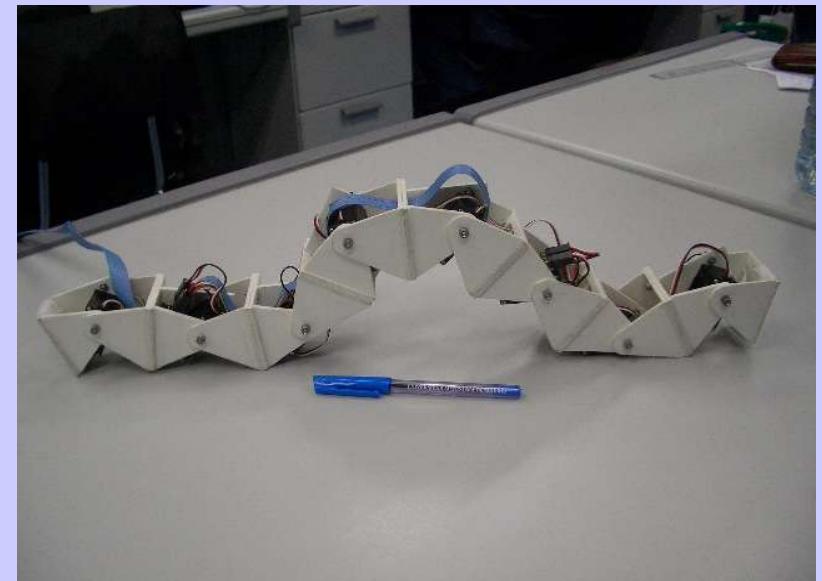
- Otros:



# Robots ápodos

- **Características:**

- Sección muy pequeña
- Muy modulares
- Mayor versatilidad en la locomoción
  - Movimiento por un plano
  - Por terrenos abruptos
  - Trepas
  - Movimiento por agujeros

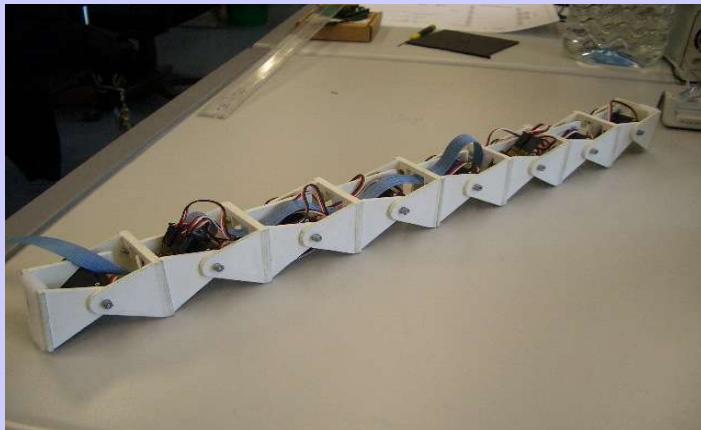


- **Aplicaciones: ¿Para qué construir un robot ápodo?**

- Por curiosidad :-)
- Busqueda y rescate en zonas catastróficas
- Inspección de tuberías
- Endoscopio "activo"

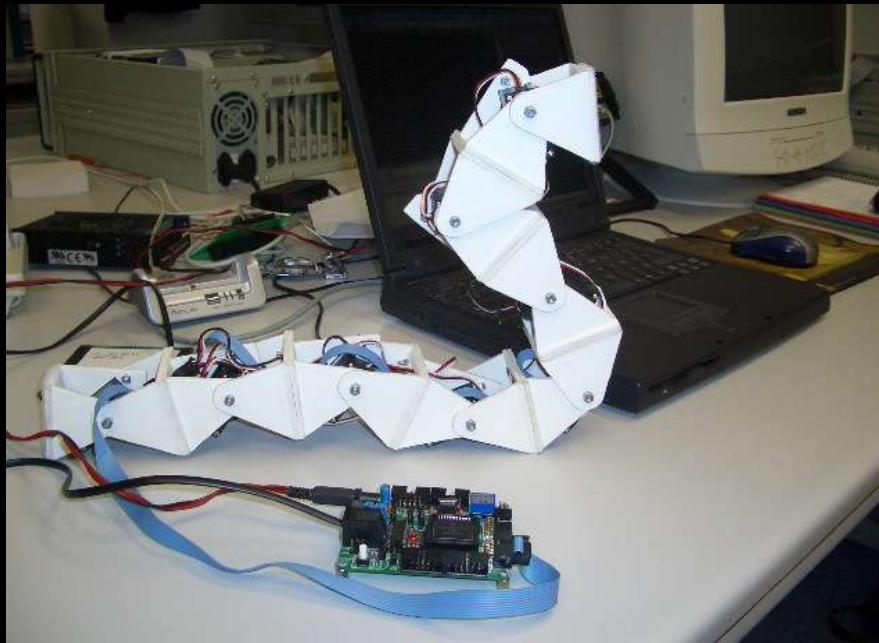
# Robots Modulares Reconfigurables

- Los Robots ápodos son muy modulares: Cadenas de módulos



- Un nuevo enfoque: **Robótica Modular Reconfigurable**
  - Mark Yim, 1994: Polypod
- Nuevo enfoque en la construcción de robots
- Diseño de robots a partir de **módulos** simples, que se unen y además se pueden **reconfigurar**
- Los robots ápodos son el caso más simple

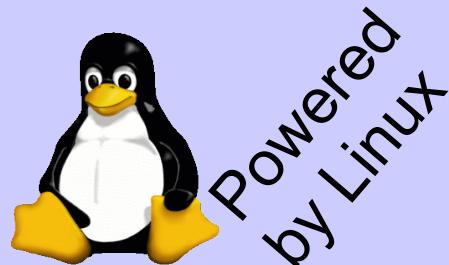
# Cube Revolutions



# Introducción

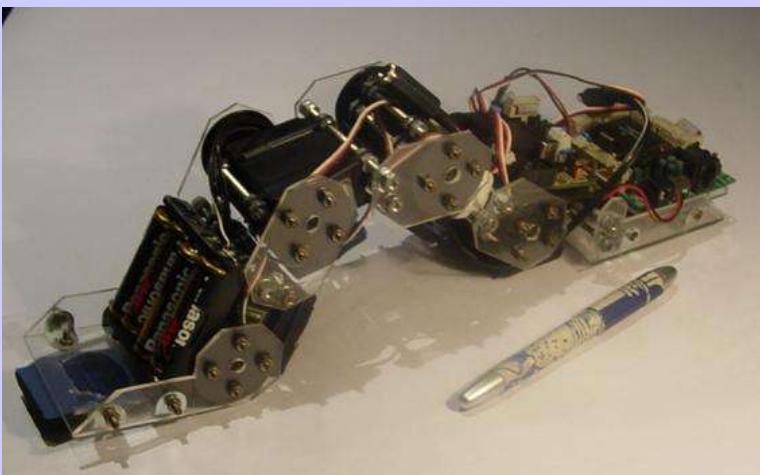
**Cube Revolutions:** Plataforma modular y abierta, para la investigación en el campo de los robots ápodos

- Constituido por 8 módulos
- **Abierto:** Toda la información está disponible para que cualquiera lo pueda construir o modificar: planos y software.
- Movimiento en línea recta
- Movimiento en un plano es mecánicamente viable, pero no implementado todavía
- Software de generación de secuencias de movimiento
- Modelo físico para la evaluación de las secuencias (en desarrollo)

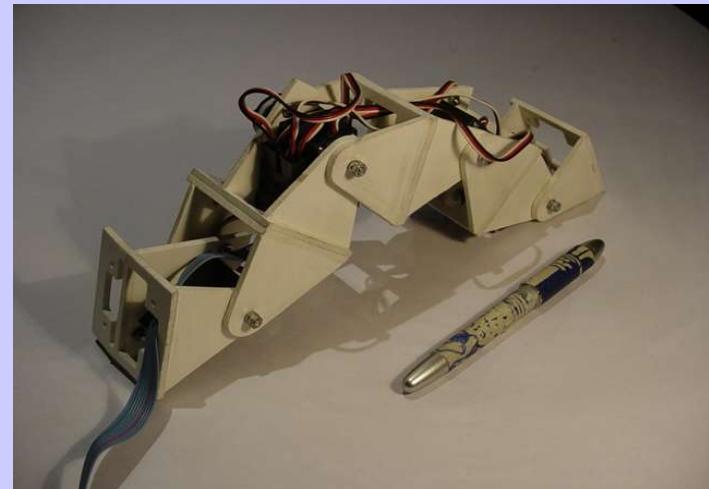


# Evolución

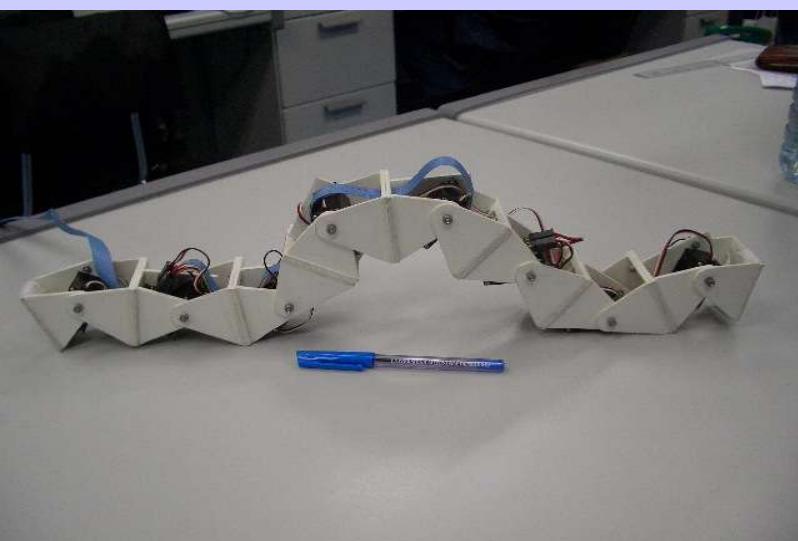
- Cube (2001)



- Cube Reloaded (2003)



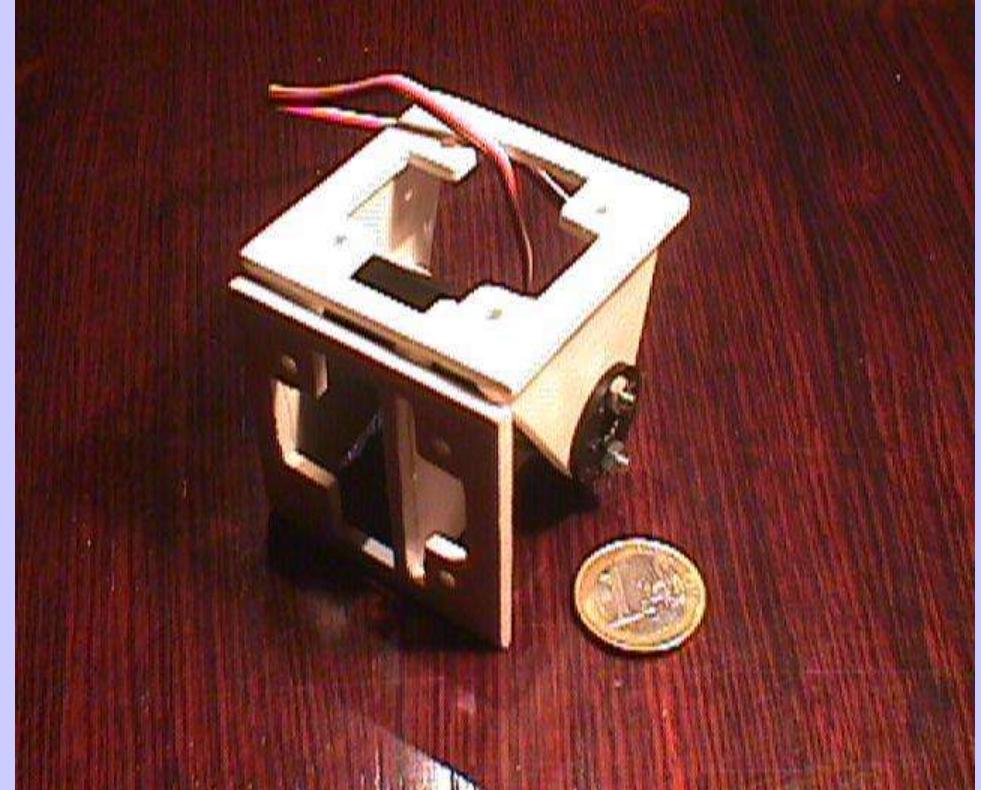
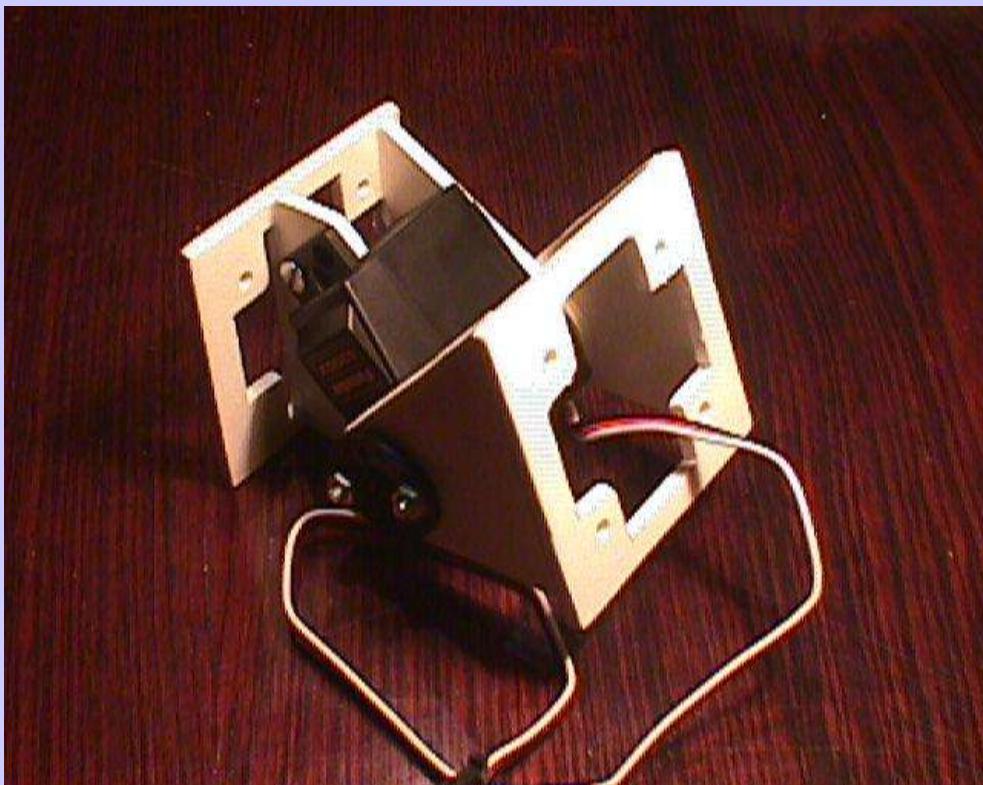
- Cube Revolution (2004)



- Hypercube (200?)

Hypercube

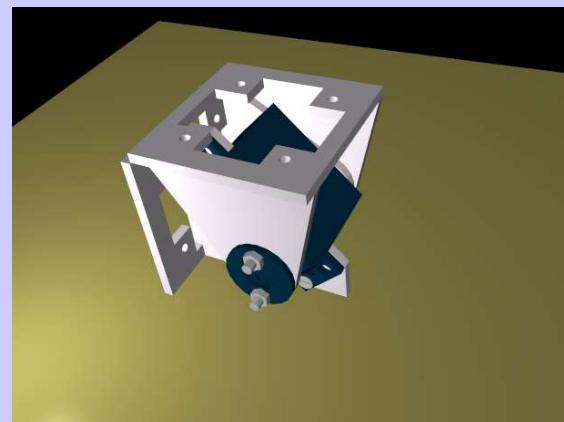
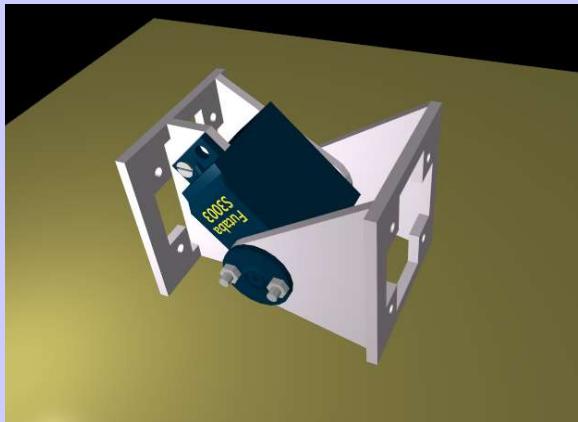
# Mecánica (I)



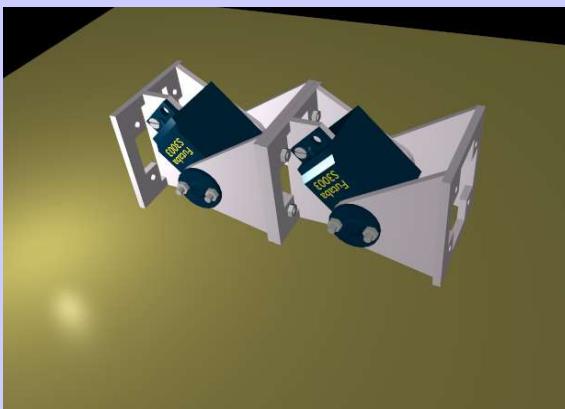
- Módulos Y1
- Basados en los realizados por Mark Yim para la primera generación de Polybot
- Se utilizan servos del tipo Futaba 3003
- No tiene sensores

# Mecánica (II)

- Modelo 3D de los módulos (Blender)



2 módulos en fase



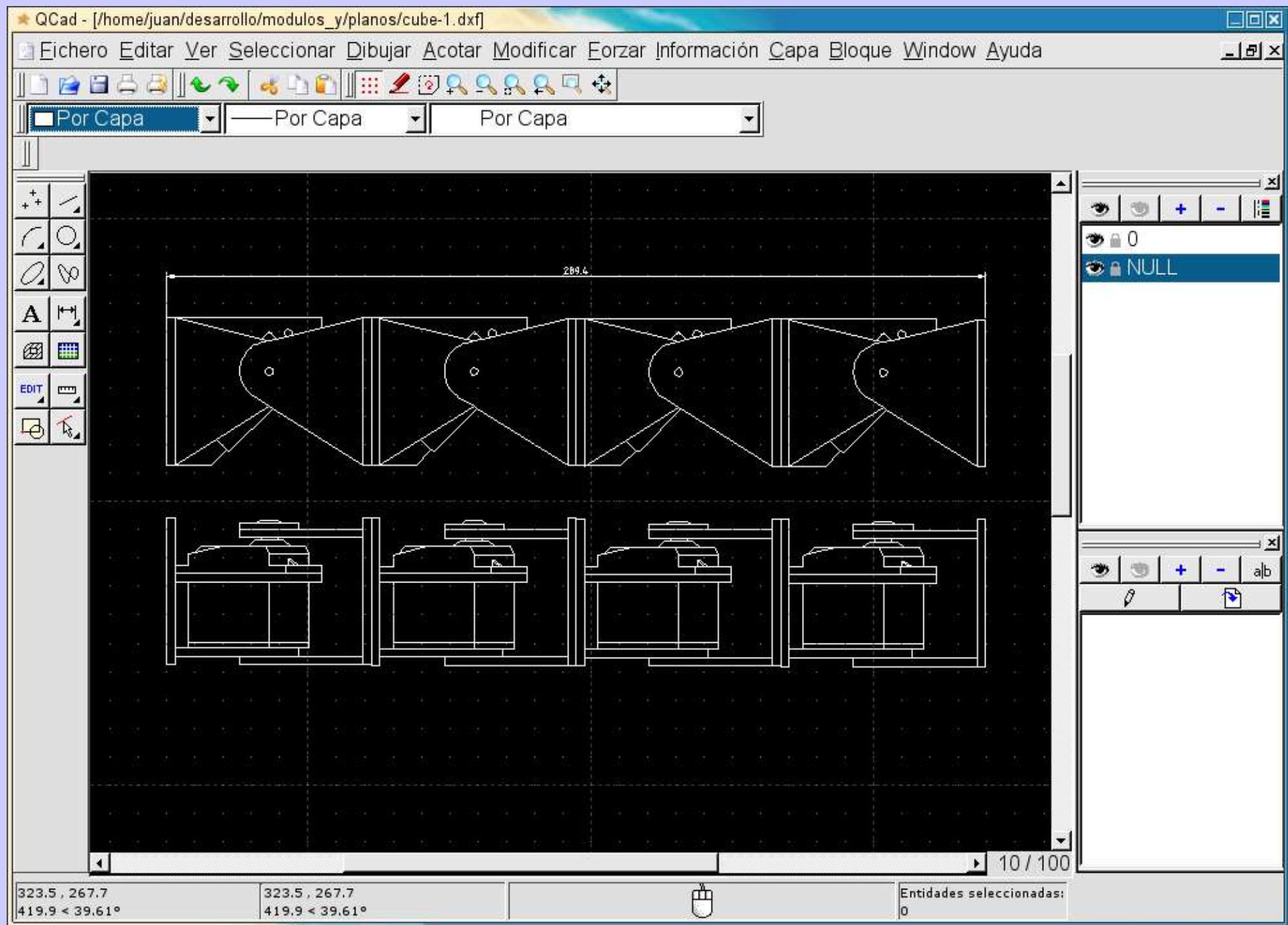
2 módulos desfasados



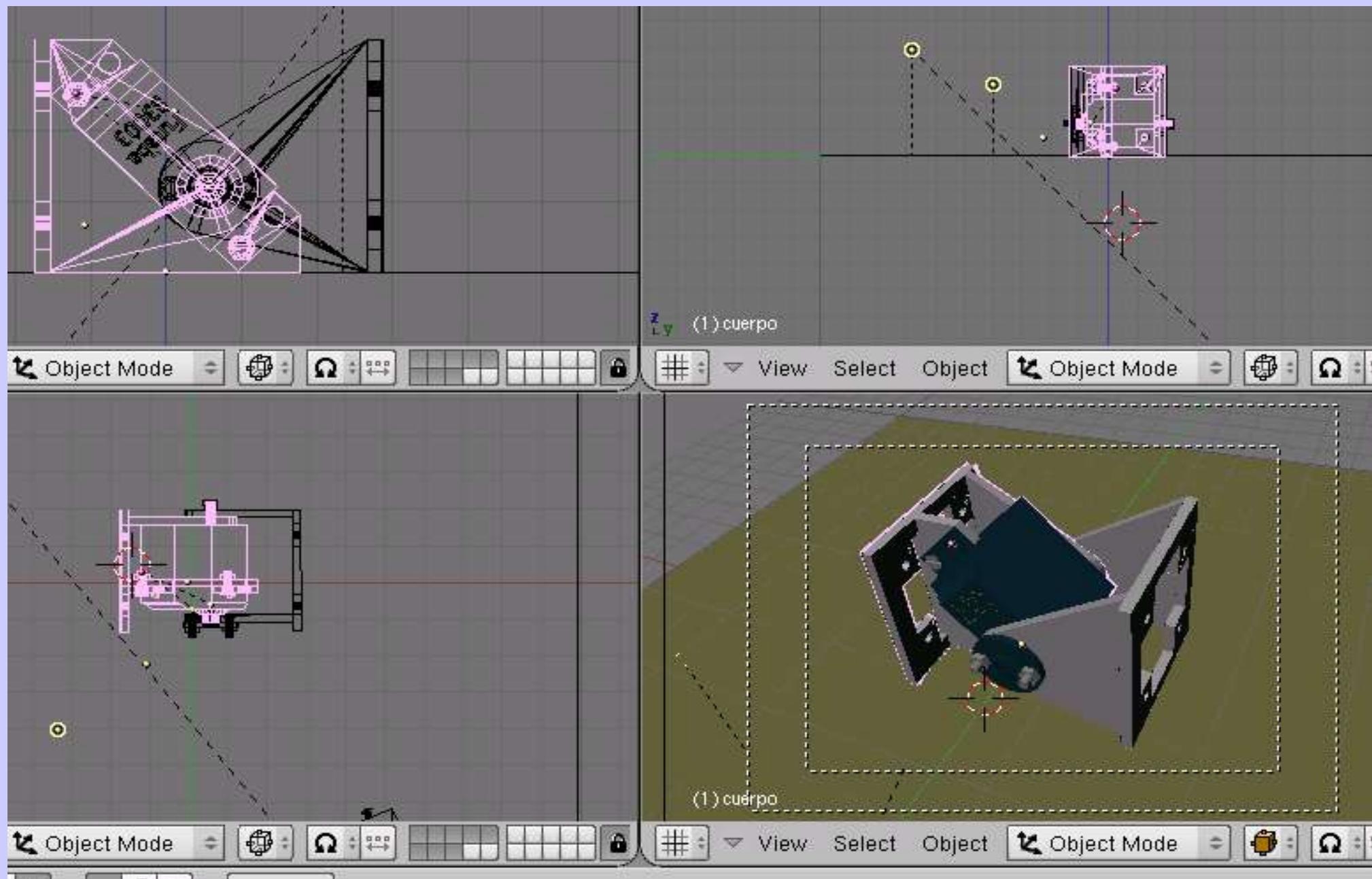
Vídeo



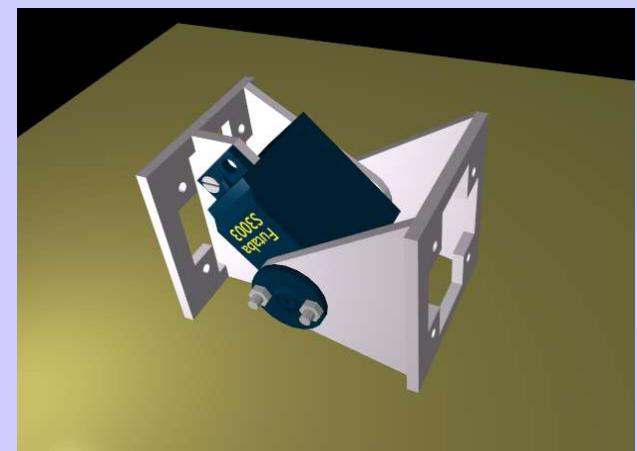
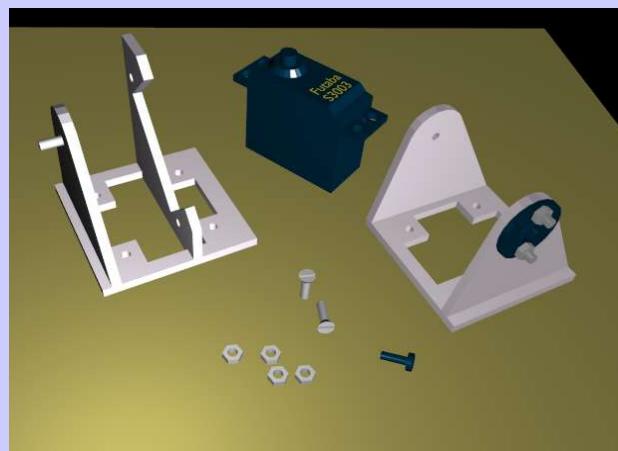
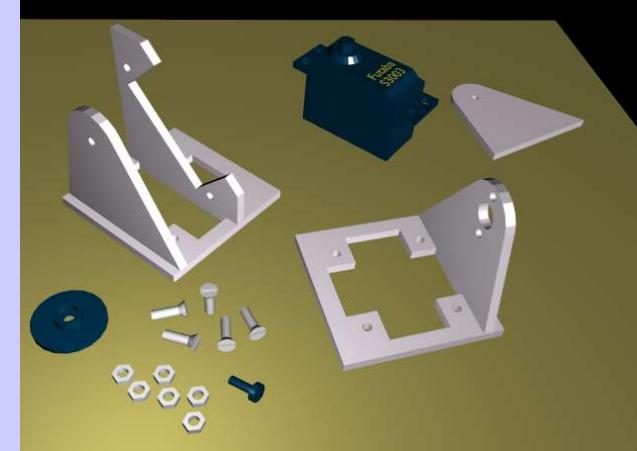
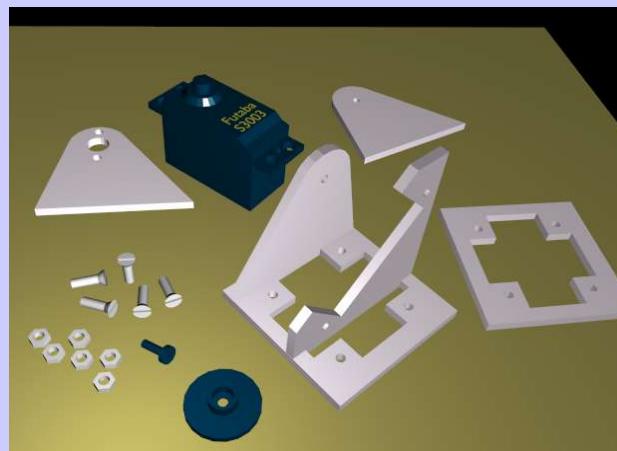
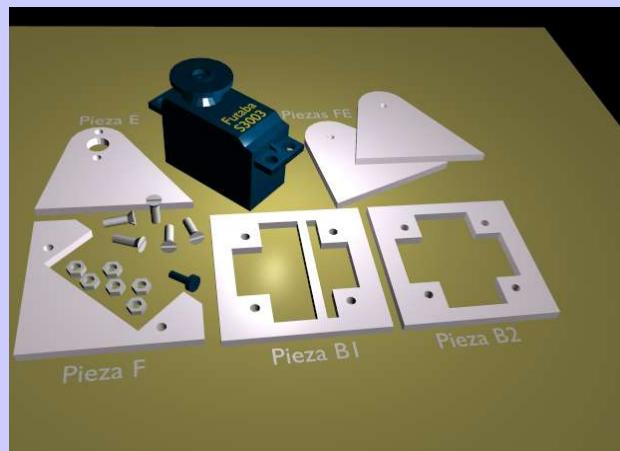
# Diseño de los planos con QCAD



# Diseño 3D con Blender



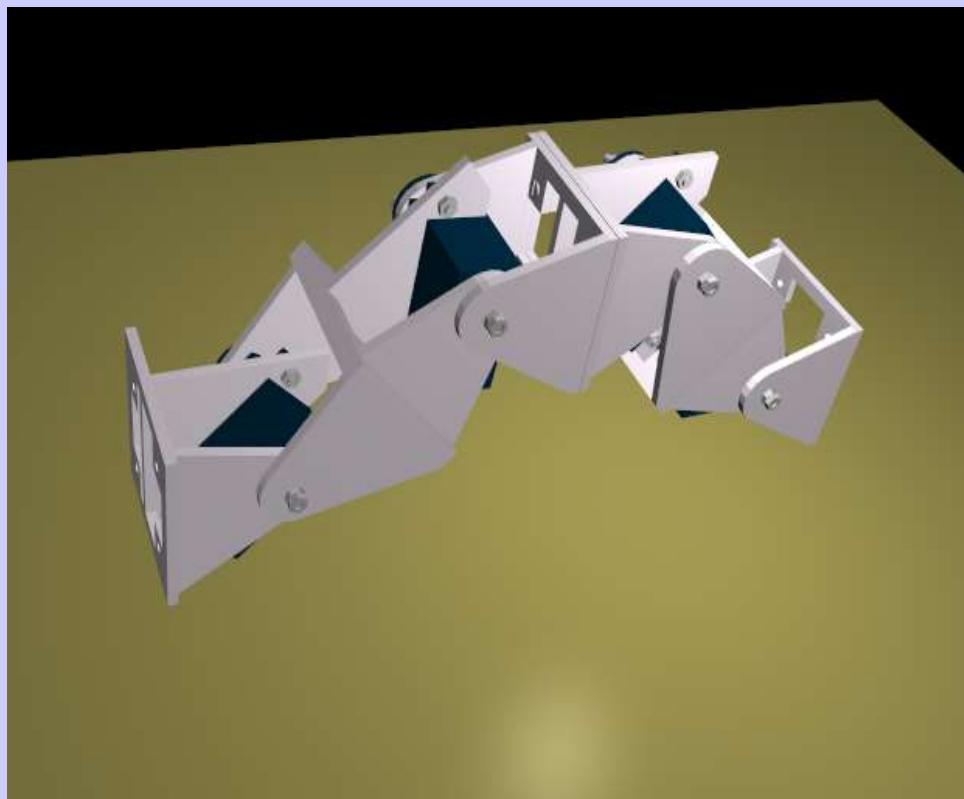
# Montaje de los módulos



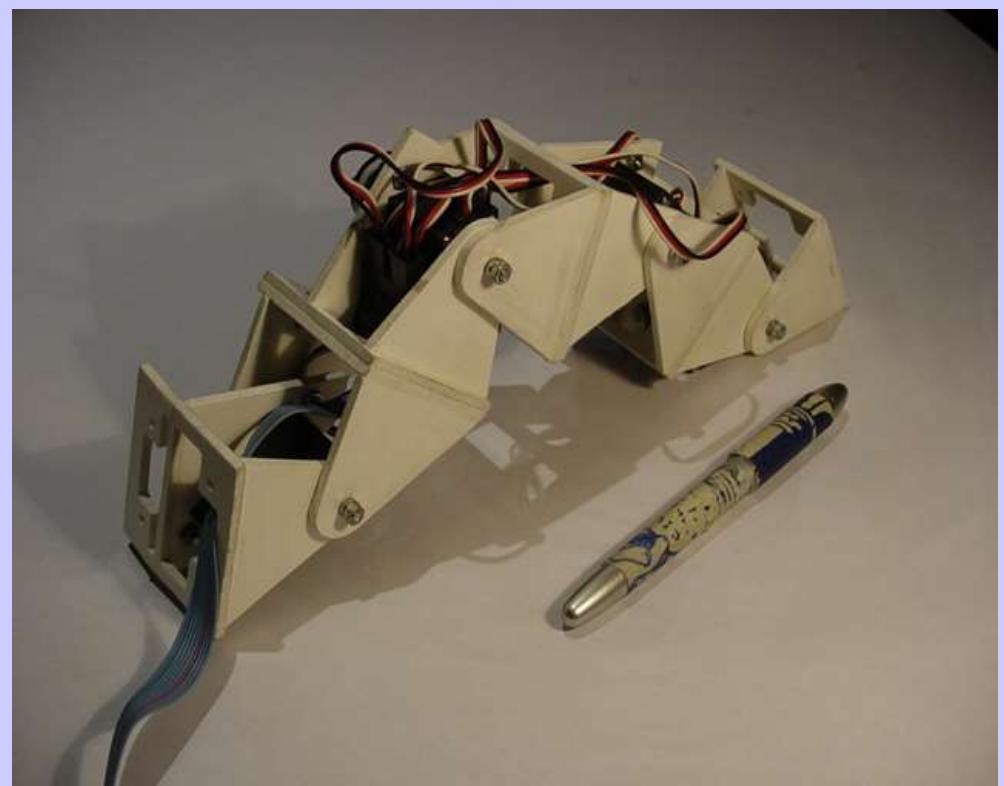
# Mecánica (IV)

---

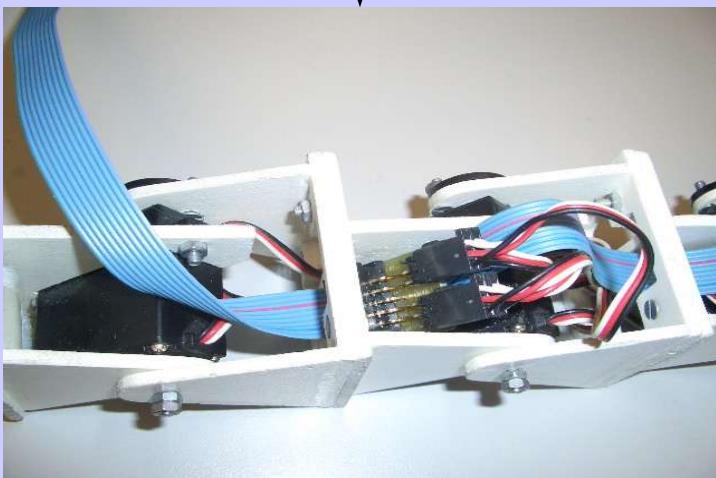
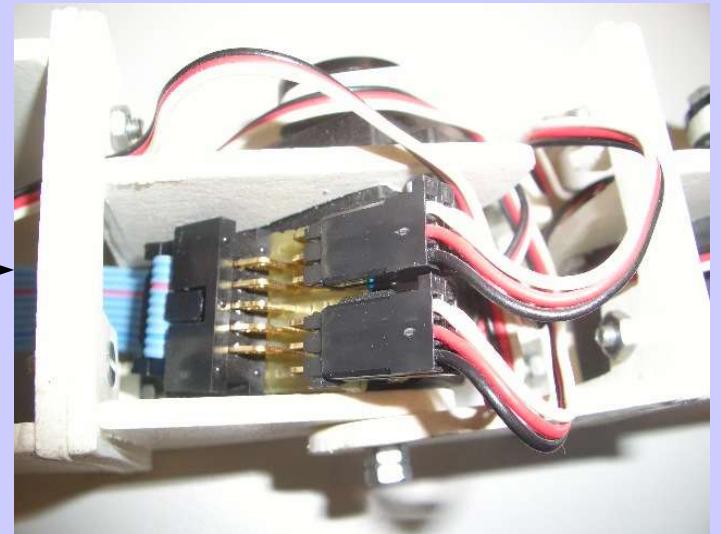
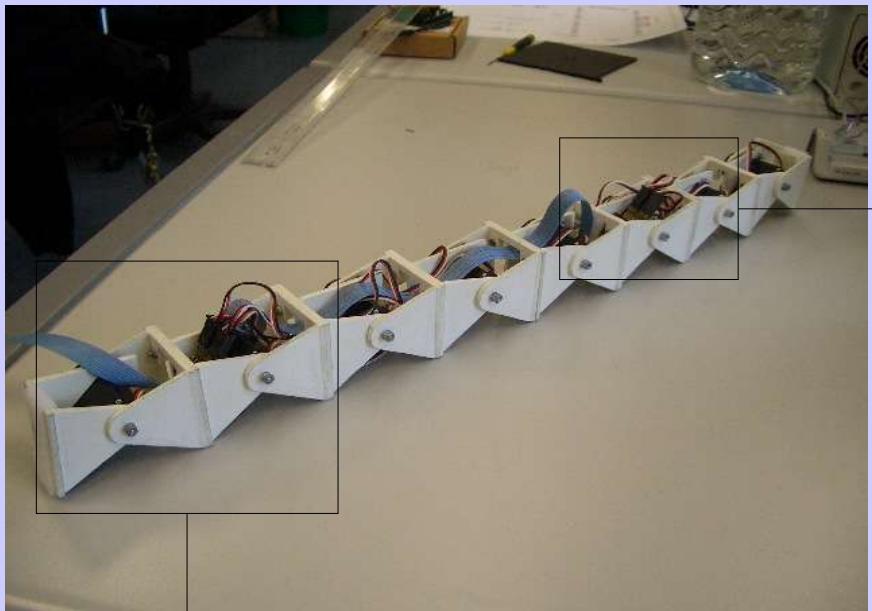
Gusano virtual



Gusano real



# Cableado



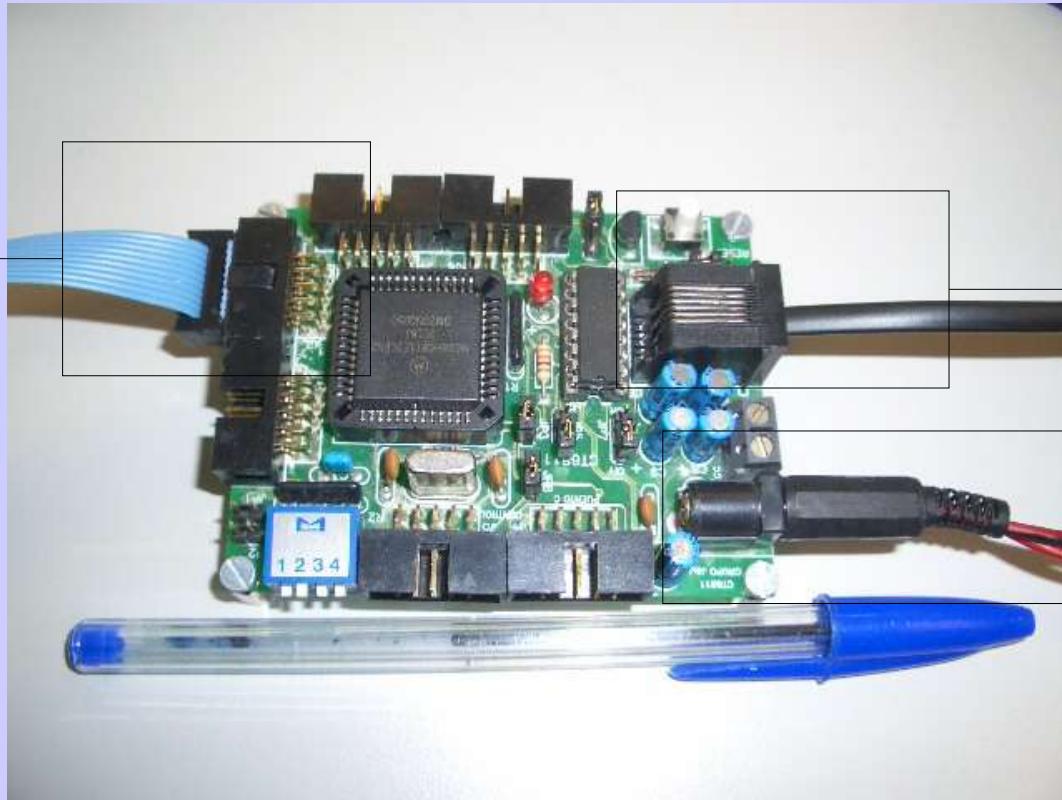
- Dos placas "pasivas"
- Se conectan 4 servos a cada una
- Interconectadas mediante cable plano de bus
- El bus lleva las señales de los 8 servos + la alimentación (VCC y GND)

# Electrónica

Al gusano

Al PC (RS-232)

Alimentación



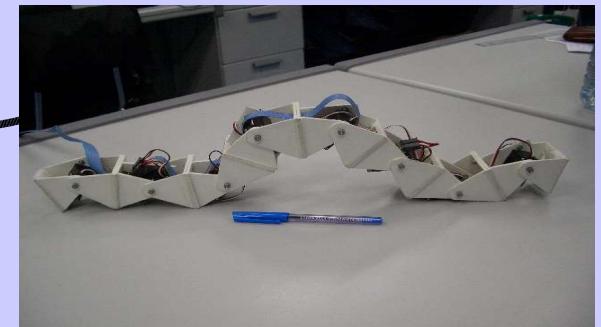
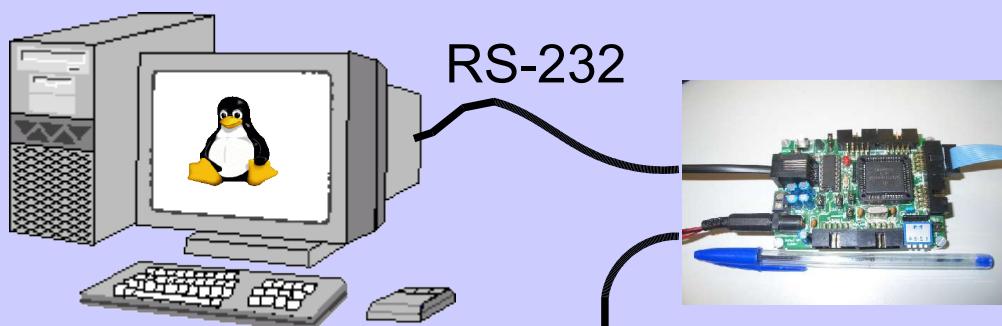
- Electrónica "Off-board"
- Un microcontrolador de 8 bits hace de "puente" entre el PC y los servos
- Servidor que permite mover 8 servos
- Implementaciones para los micros 6811 (CT6811) y 16F876 (SKYPIC)

# Software (I)

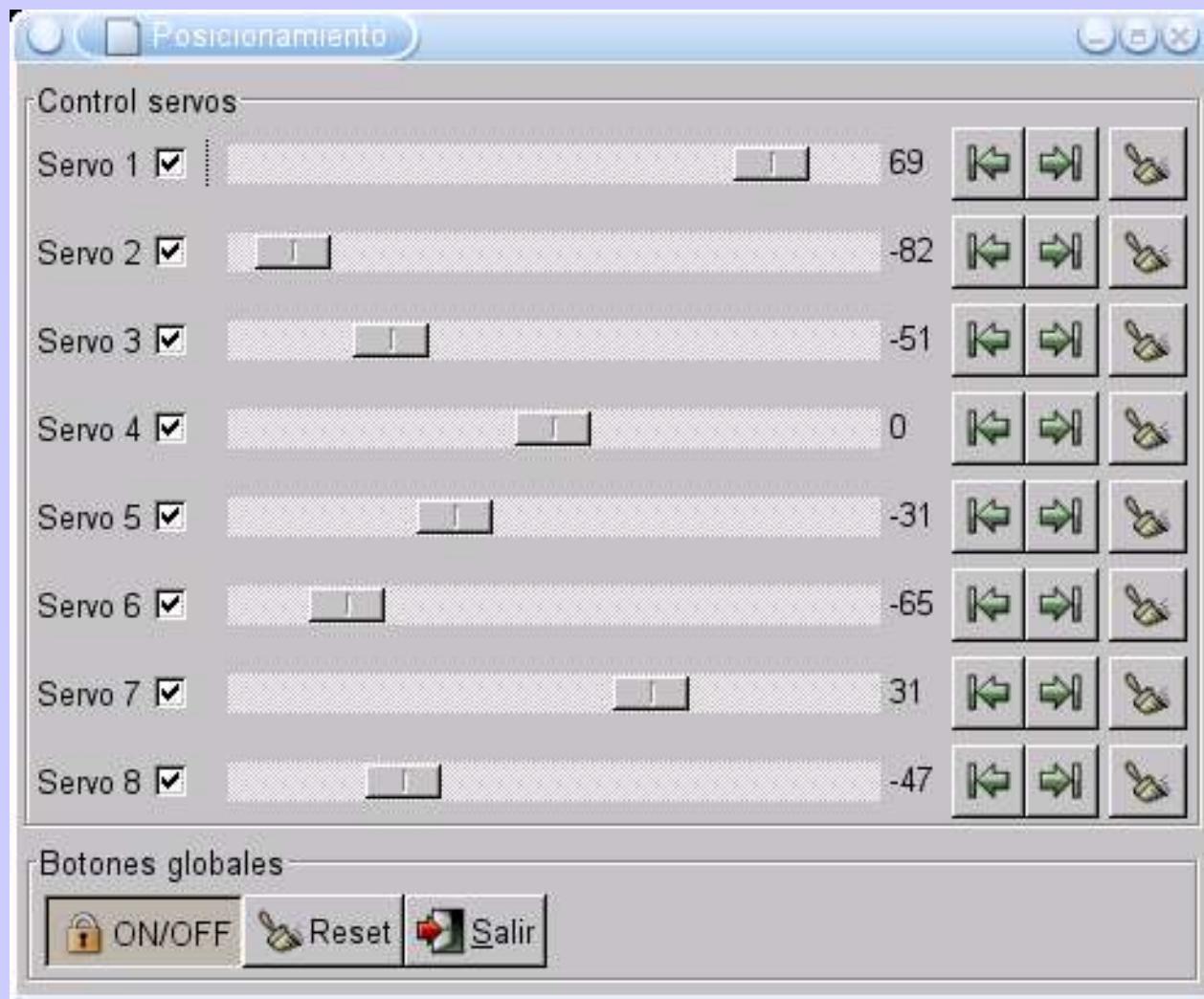
API en C que permite situar cualquier servo en una posición

- sg-servos8-enable(int mask) → Activación/desactivación servos
- sg-servos8-pos1(int servo, int pos) → Posicionamiento servo

[1-8]      [-90,90]



# Software (II)



- La API permite establecer la posición de cualquiera de los 8 servos
- No es un software específico para CUBE: es válido para otros robots articulados
- A partir de ella se pueden hacer programas para consola o en modo gráfico:

**DEMO 1**

# **Locomoción**

# Coordinación (I)

¿ Cómo coordinar todas las articulaciones para conseguir movimiento?

- **Observando los gusanos en la naturaleza...** por ejemplo los gusanos de seda...
  - El movimiento comienza por la cola
  - Se produce una perturbación inicial...
  - ...que se propaga hasta que alcanza la cabeza

Se produce una contracción inicial que se propaga...



# Coordinación (II)

1. Situación inicial



2. Contracción



3. Propagación



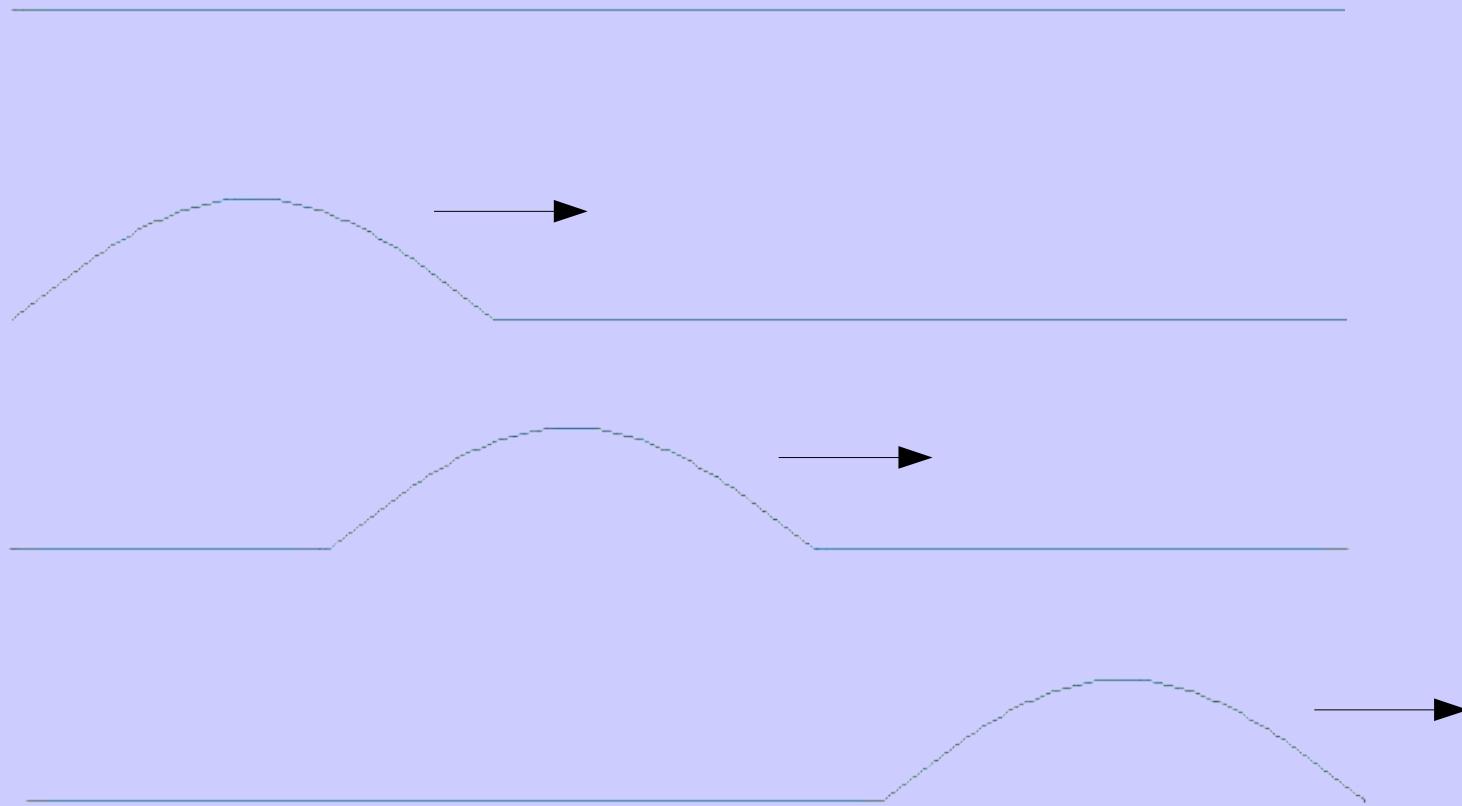
4. Situación final



Avance

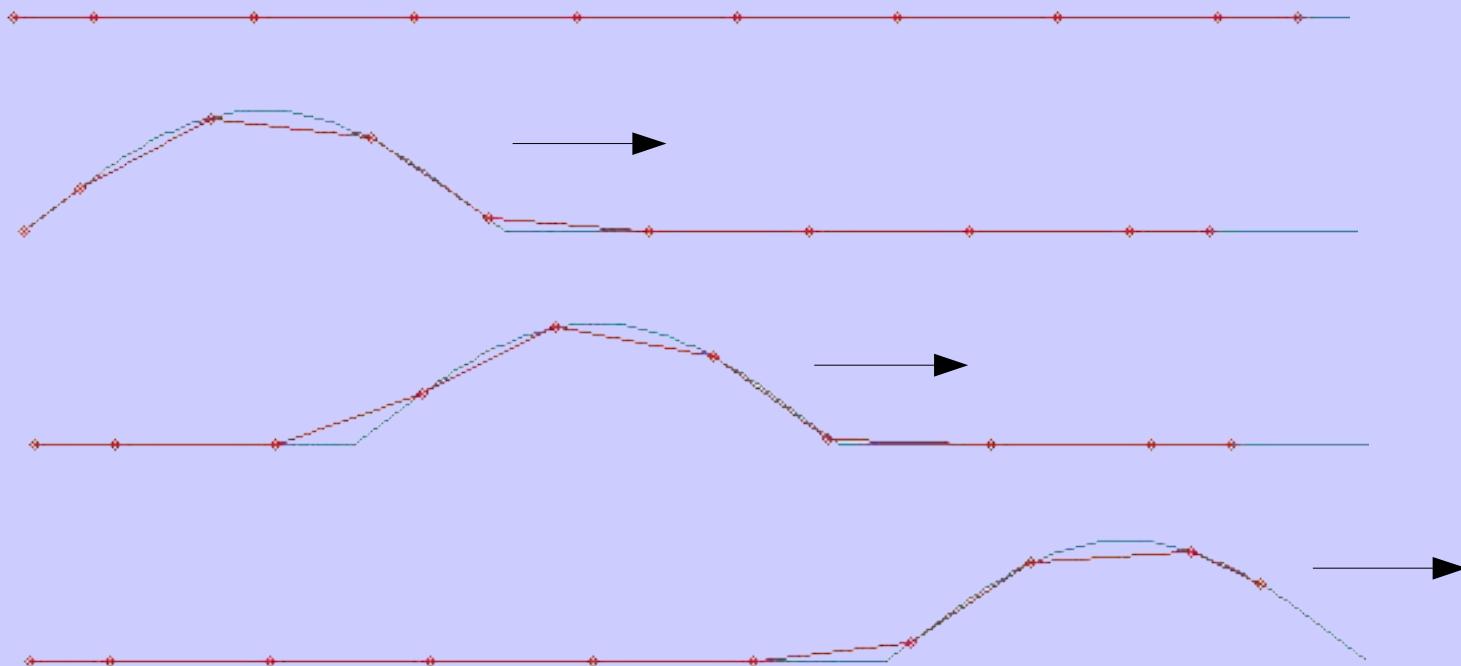
# Coordinación (III)

Modelo de propagación de ondas...



# Coordinación (IV)

Para conseguir avance hay que hacer que las articulaciones se sitúen sobre la onda

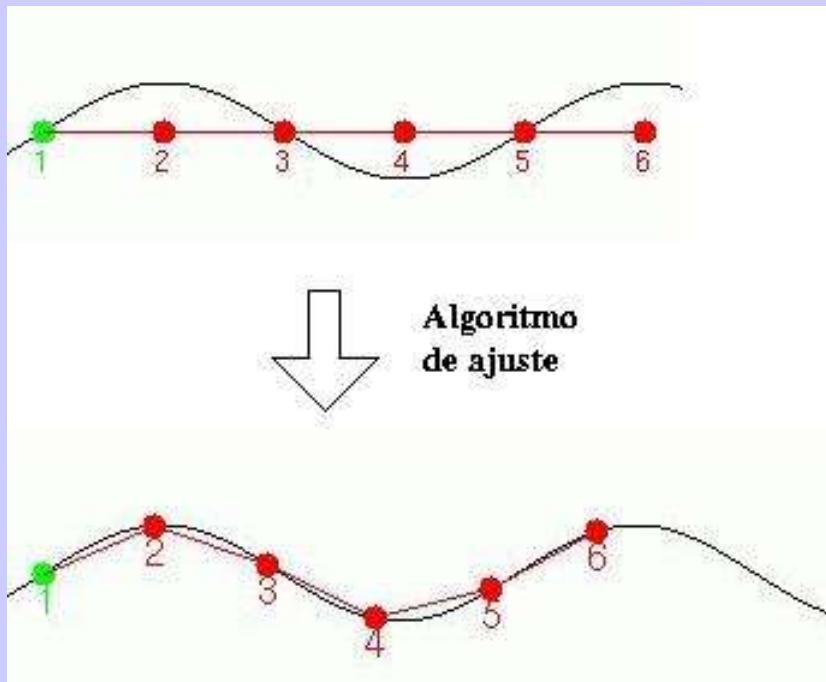


# Coordinación (V)

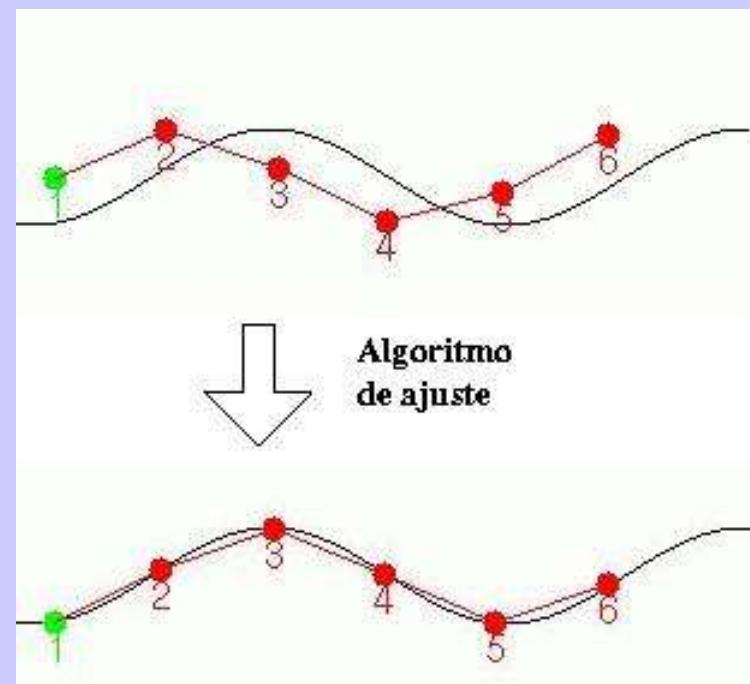
- La generación de las secuencias de movimiento se hace aplicando el modelo de propagación de ondas:

- A partir de la onda en un instante  $t$ , se calcula la posición de las articulaciones para situarlas sobre ella (AJUSTE)
- Se desplaza la onda y se vuelve a "ajustar" el gusano

1

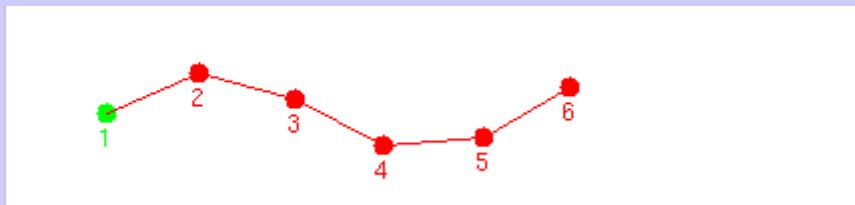


2



# Coordinación (VI)

- **RESULTADO: Conseguimos las posiciones de los servos en cada instante de tiempo que logran que el gusano avance**
- La evolución de los ángulos es la siguiente:



- Para cada instante de tiempo, obtenemos un **vector de estado**, que contiene los ángulos en los que se sitúan cada una de las articulaciones

$t_1 \longrightarrow [a_1, b_1, c_1, d_1, e_1, f_1, g_1, h_1]$

$t_2 \longrightarrow [a_2, b_2, c_2, d_2, e_2, f_2, g_2, h_2]$

...

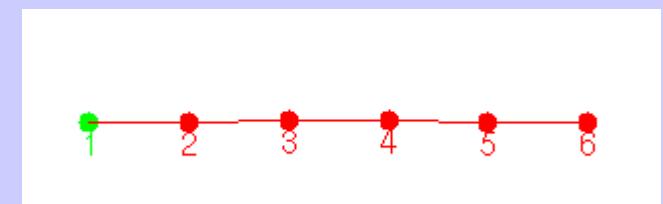
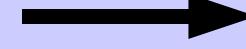
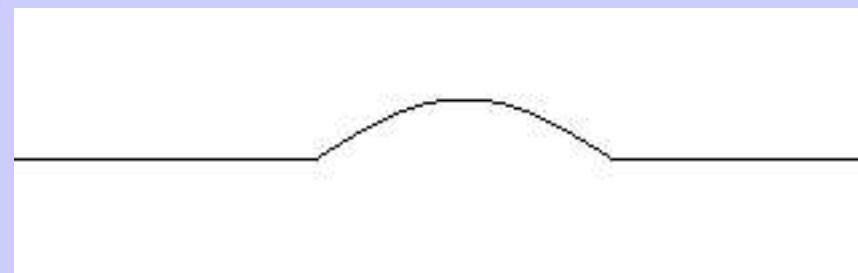
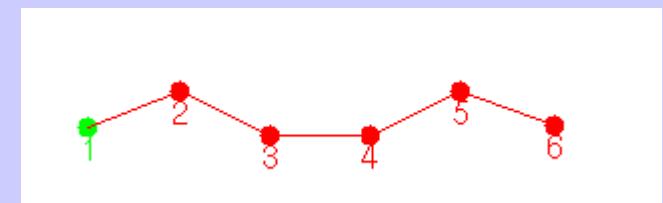
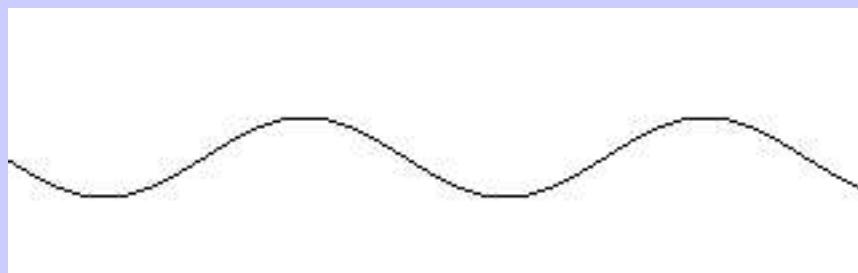
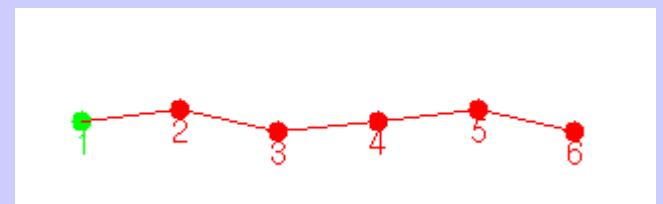
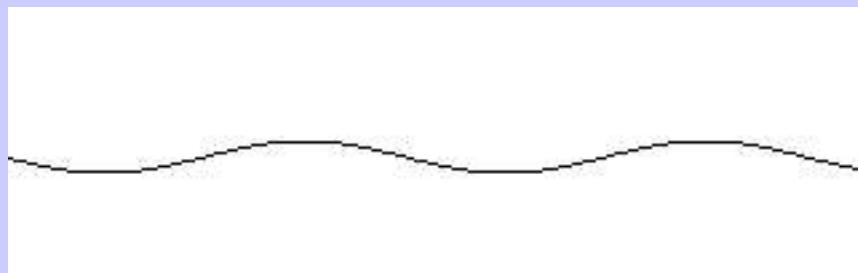
# Coordinación (VII)

- Cada tipo de movimiento está definido por una matriz, que contiene todos los vectores de estado:

$$\begin{bmatrix} -19 & -9 & 24 & -4 & -22 & 17 & 12 & -18 \end{bmatrix} \longrightarrow \text{Instante t1}$$
$$\begin{bmatrix} -17 & -14 & 21 & 4 & -25 & 11 & 19 & -15 \end{bmatrix} \longrightarrow \text{Instante t2}$$
$$\begin{bmatrix} -12 & -20 & 18 & 11 & -25 & 3 & 23 & -7 \end{bmatrix} \longrightarrow \text{Instante t2}$$
$$\begin{bmatrix} -5 & -24 & 12 & 17 & -22 & -4 & 24 & -2 \end{bmatrix} \dots$$
$$\begin{bmatrix} 1 & -25 & 6 & 21 & -17 & -12 & 24 & 4 \end{bmatrix} \dots$$
$$\begin{bmatrix} 7 & -23 & -3 & 25 & -11 & -18 & 20 & 12 \end{bmatrix} \dots$$
$$\begin{bmatrix} 12 & -19 & -9 & 24 & -4 & -22 & 17 & 14 \end{bmatrix}$$
$$\begin{bmatrix} 17 & -13 & -15 & 21 & 4 & -25 & 11 & 19 \end{bmatrix}$$
$$\begin{bmatrix} 20 & -6 & -21 & 17 & 12 & -24 & 2 & 21 \end{bmatrix}$$
$$\begin{bmatrix} 21 & 0 & -23 & 12 & 17 & -21 & -6 & 22 \end{bmatrix}$$
$$\begin{bmatrix} 19 & 9 & -24 & 4 & 22 & -17 & -12 & 18 \end{bmatrix} \longrightarrow \text{Instante tn}$$

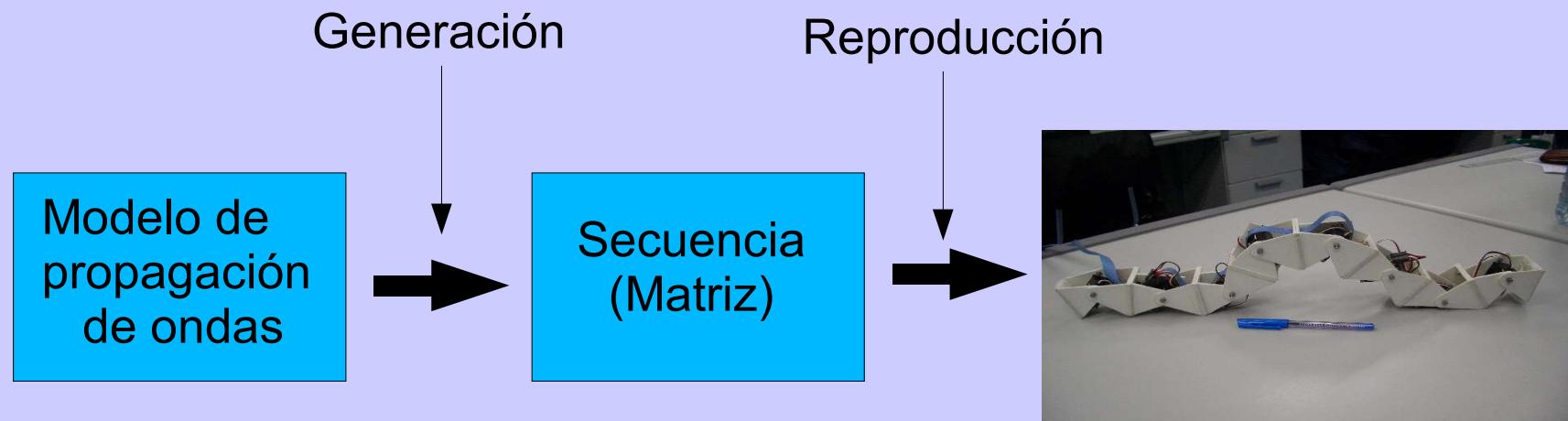
# Coordinación (VIII)

- A partir del **tipo de onda, amplitud y longitud de onda**, se consiguen secuencias de movimiento diferentes



# Locomoción

- La **locomoción** se implementa según el siguiente diagrama de bloques:



# DEMO 2

★ Cube Virtual.mono

Archivo Ayuda

**Configuracion**

**Onda**

Sin  Semi  Comb

Visible

**Tipos**

Sin | Semi | Comp |

Ampitud 15 Fase 0

Long. onda 120 Freq 5

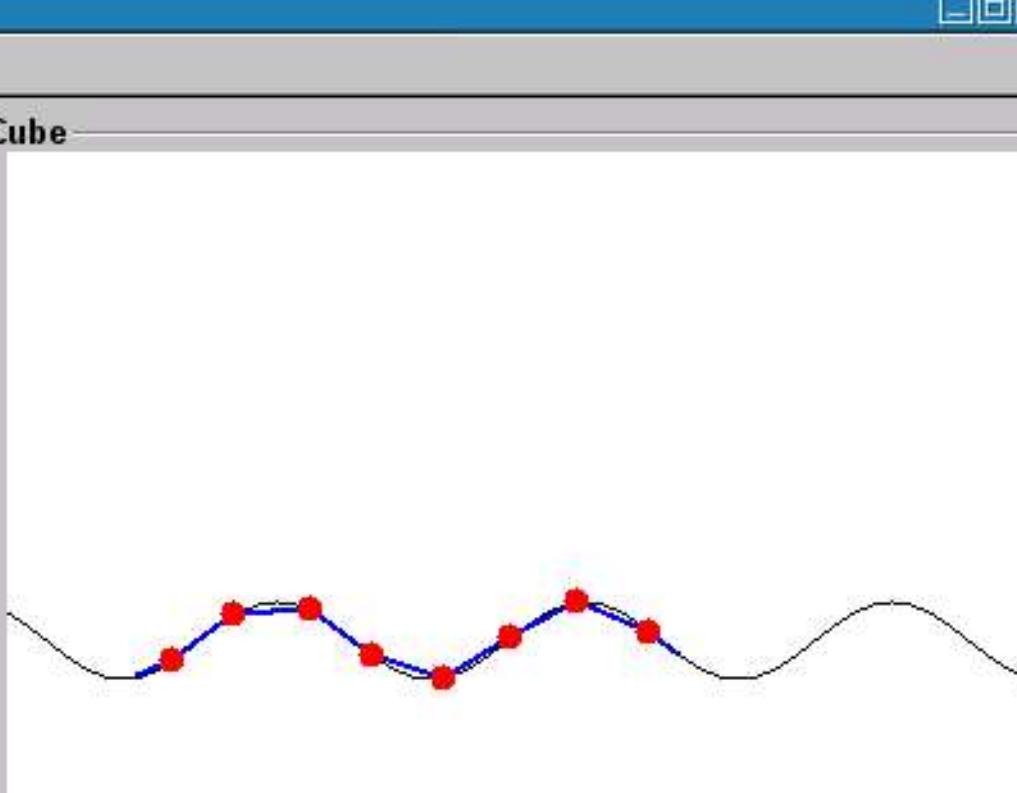
**Gusano**

Long. Seg 15 Radio 5 Arts 8

Eje x  Panel

Num. Art  Visible

**Cube**



[12 -33 -40 19 50 -4 -51 -13]  
X: [0 14 38 68 92 120 146 172 200 212 ]  
Y: [-14 -8 10 12 -6 -15 1 15 3 -6 ]

**Generacion**

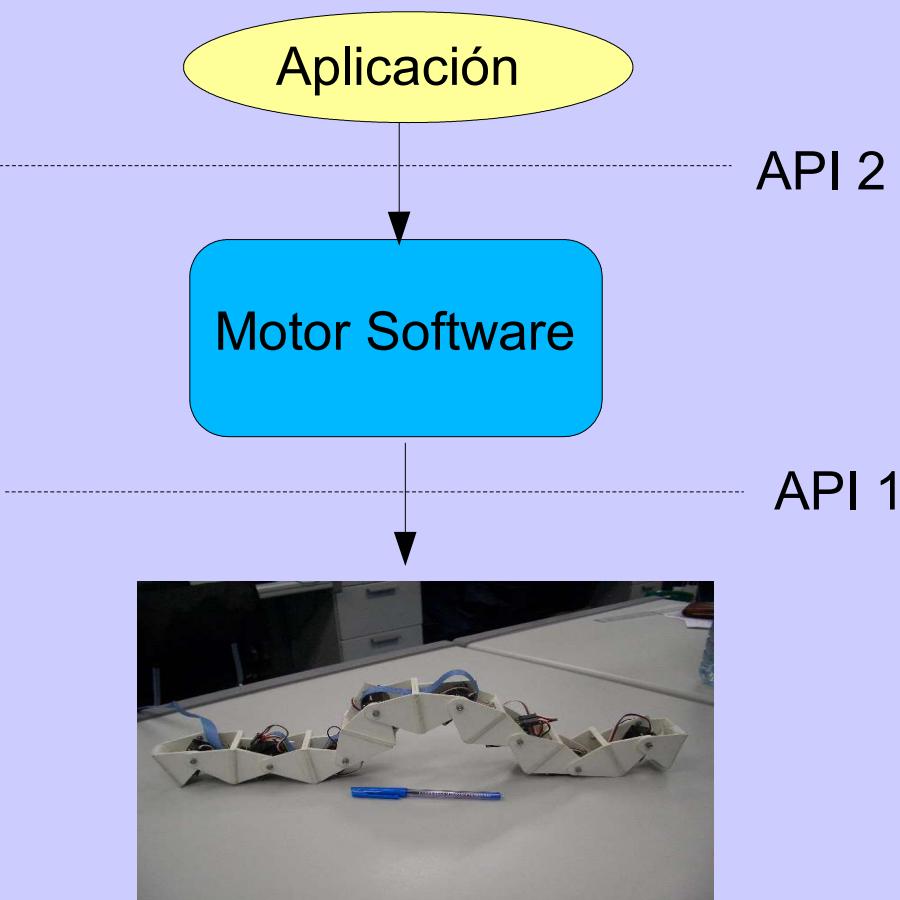
Ajustar  Play  Stop  Guardar

Tiempo 149   Enganchar

# Simulación

# Recapitulando

- **Coordinación:** usando ondas
- **Parámetros:** Amplitud, longitud de onda y tipo de onda

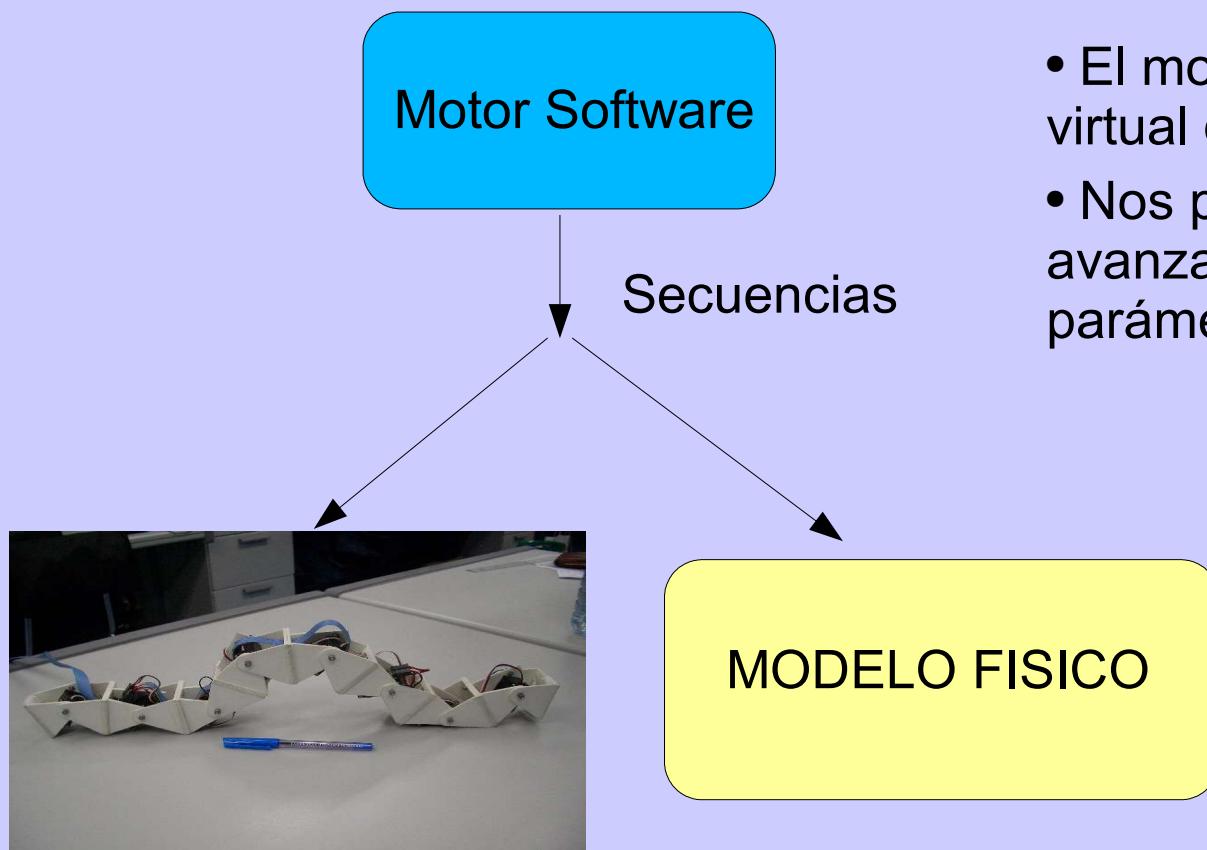


- La aplicación puede mover el gusano a partir de los parámetros
- No tiene que conocer nada sobre la coordinación

**¿Qué parámetros son los óptimos en cuanto a consumo, estabilidad o velocidad?**

# Simulación (I)

- Necesitamos tomar medidas
- Lo más sencillo es tener un **MODELO FÍSICO** del gusano



- El modelo físico es un gusano virtual que se mueve por la "pantalla"
- Nos permite determinar cuánto avanza con cada tipo de onda y sus parámetros

# Simulación (III)

- El modelo físico está implementado en C
- Genera resultados que se pueden visualizar en OCTAVE/MATHLAB

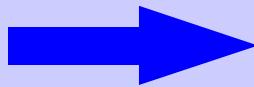
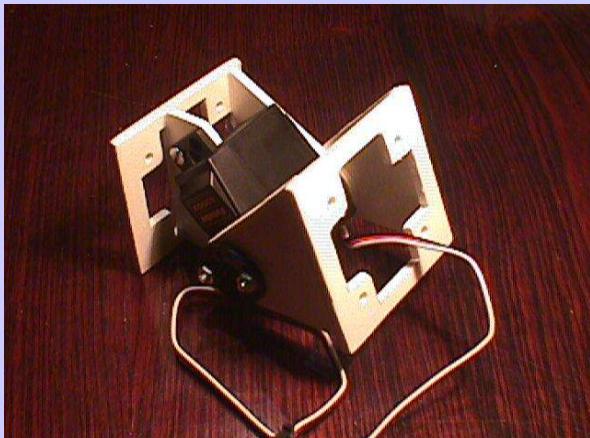


# **Líneas de investigación abiertas**

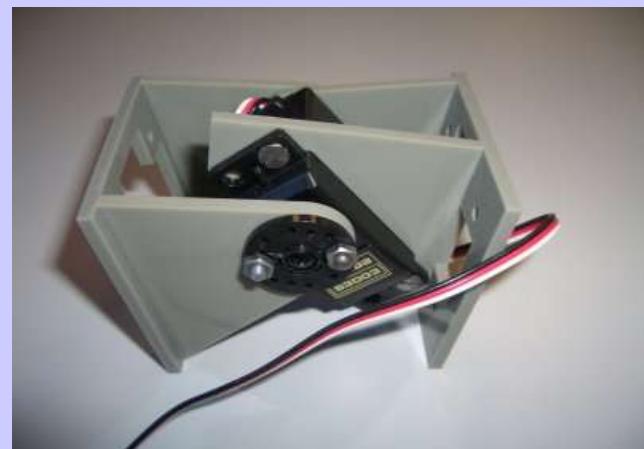
# Actualmente...

- Nueva versión de los módulos: Y1.1
  - Más robustos
  - Corte por láser

Módulo Y1

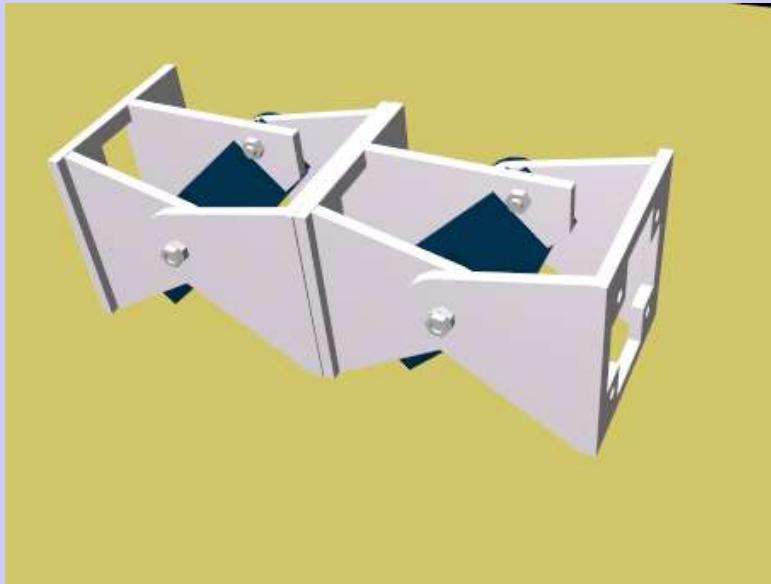


Módulo Y1.1



# Línea de investigación 1: Robótica modular (I)

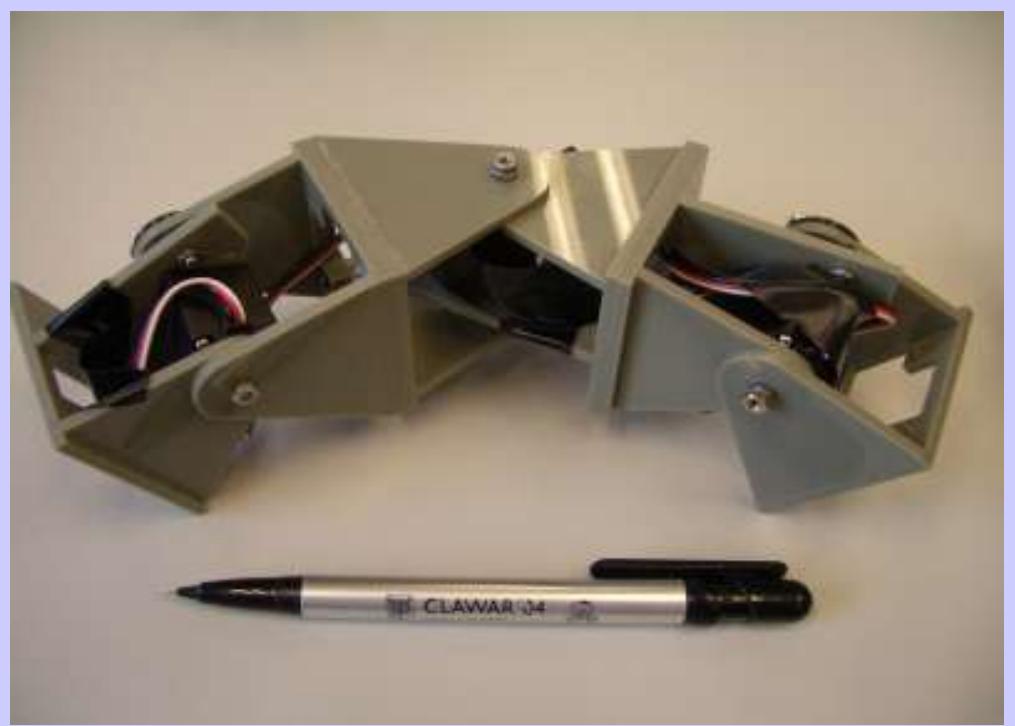
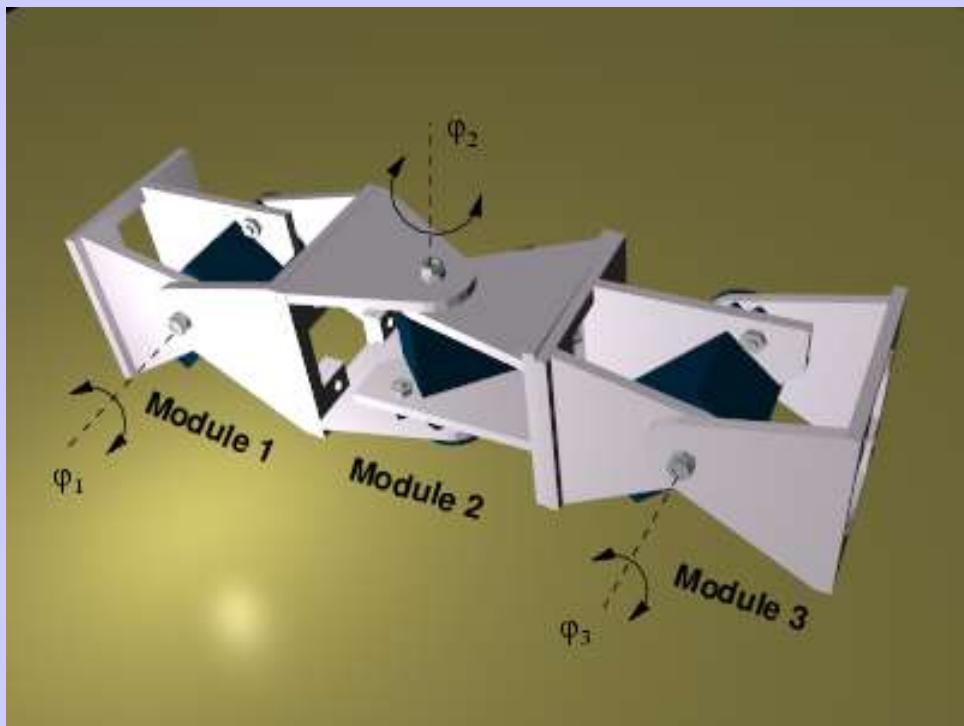
¿Cuáles son las configuraciones mínimas con las que se consigue locomoción?



Con sólo dos módulos unidos en fase, se consigue locomoción en línea recta

# Línea de investigación 1: Robótica modular (II)

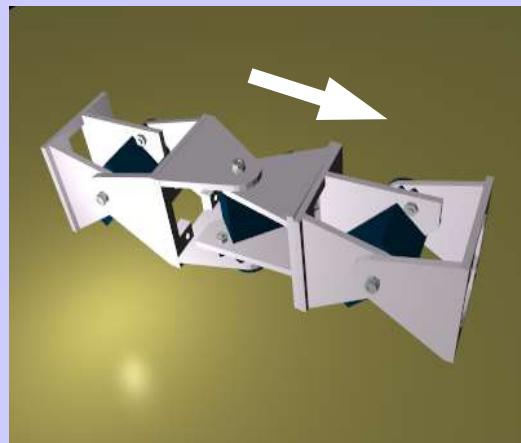
¿Y si añadimos un módulo más conectado en desfase?



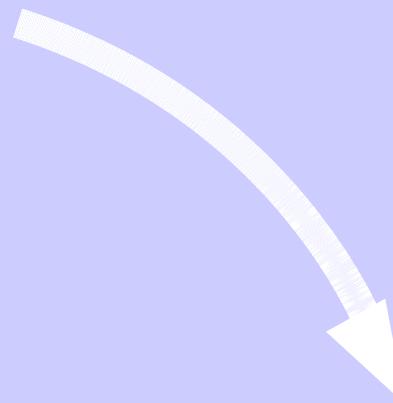
# Línea de investigación 1: Robótica modular (III)

Se pueden realizar los siguientes movimientos:

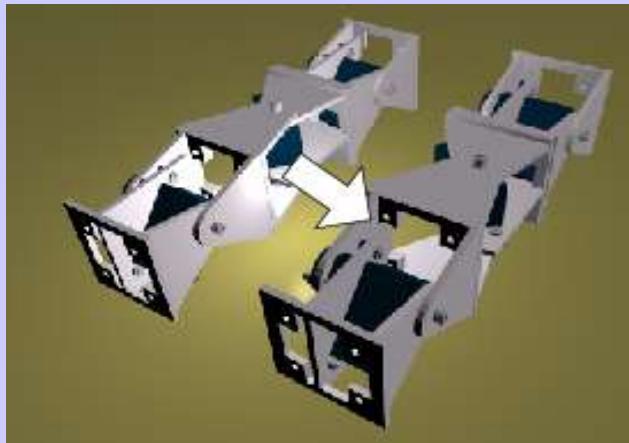
Avance en línea recta



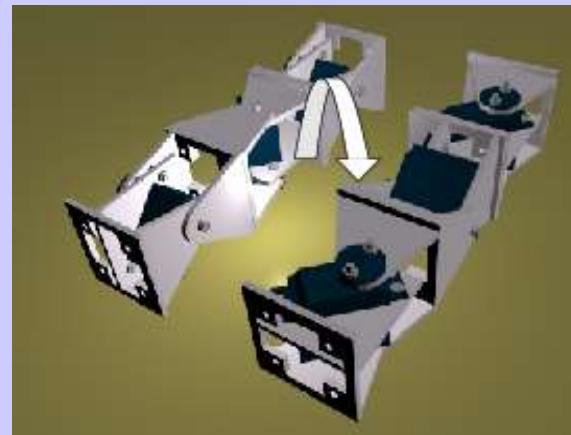
Trayectorias curvas



Desplazamiento lateral



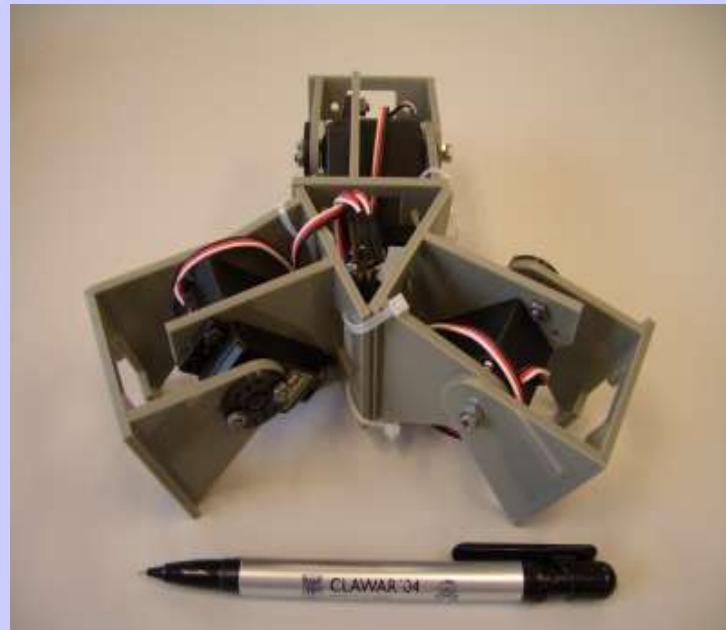
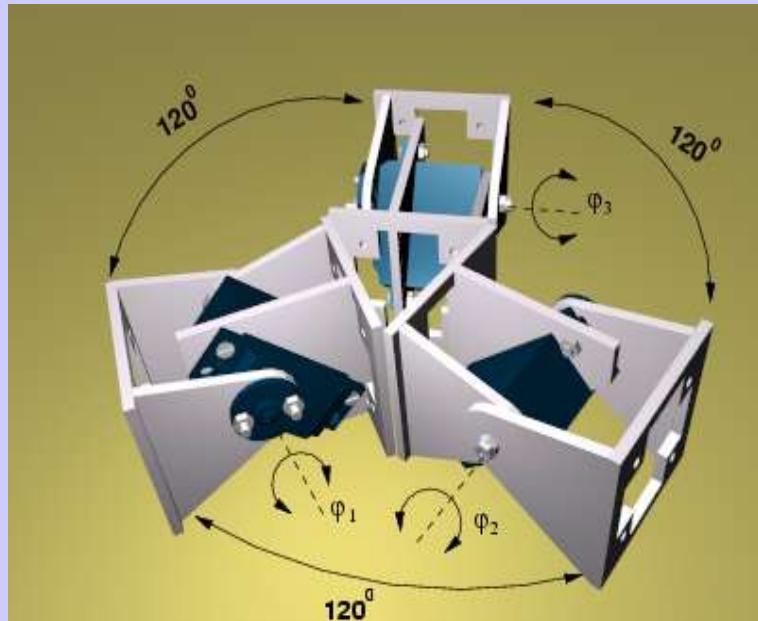
Rotación lateral



# Línea de investigación 1: Robótica modular (VI)

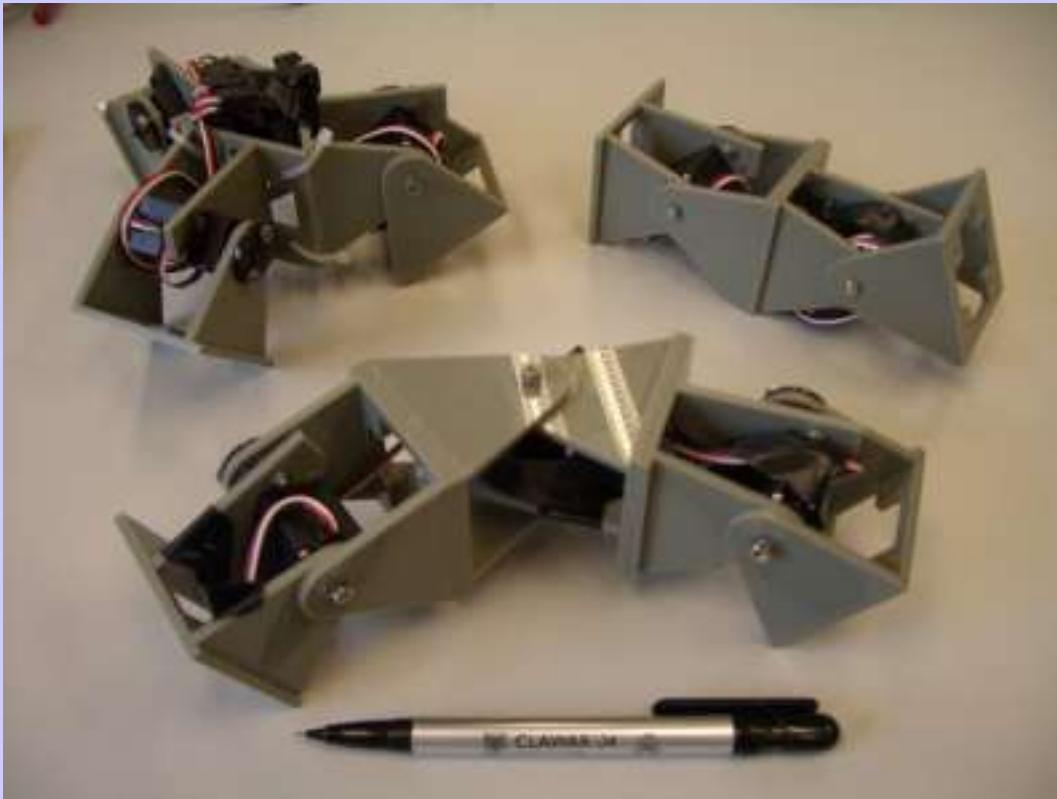
¿Qué otras configuraciones se pueden tener con tres módulos?

- Configuración en estrella
- Puede moverse en tres direcciones
- Puede rotar



# Línea de investigación 1: Robótica modular (V)

**MULTICUBE**



**DEMO 4**

## Línea de investigación 2: Algoritmos genéticos

- Aplicación de **algoritmos genéticos** para la generación de secuencias de movimiento

¿Qué secuencias se obtienen?

¿Son ondas sinusoidales?

¿De qué amplitud?

¿Qué longitud de onda?

¿Cuáles son las secuencias óptimas?

# Línea de investigación 3

---

The logo consists of the word "Hypercube" in a bold, black, sans-serif font. The letters are slightly shadowed, giving them a 3D effect. The logo is set against a white rectangular background, which is itself centered on a larger light blue rectangular background.

- 8 módulos. 4 paralelos al suelo, 4 perpendiculares
- Estudio del movimiento en 3D

# Enlaces

---

Algunas direcciones web son difíciles de recordar... Para acceder a toda la información hacer lo siguiente:

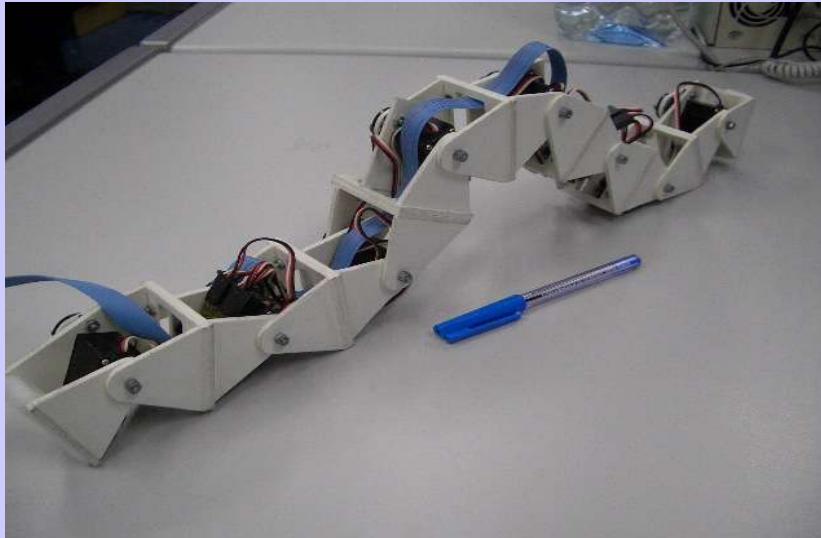
- 1) Ir a Google
- 2) Teclear: "Cube Reloaded"
- 3) Pinchar en voy a tener suerte

También podéis acceder directamente a la dirección:

[www.iearobotics.com](http://www.iearobotics.com)

## Sesión III: Robótica e investigación

### Diseño de robots ápodos: *Cube Revolutions*



Juan González Gómez

Escuela Politécnica Superior  
Universidad Autónoma de Madrid