

<https://ieomsociety.org/proceedings/2021dhaka/488.pdf>

<https://github.com/Abhimanyu-0/Xarm-6-robot>

<https://www.manualslib.com/manual/3298276/Ufactory-Xarm6.html>

[https://www.researchgate.net/profile/Md-Samiul-Sunny-2/publication/360128896\\_Kinematics\\_and\\_Workspace\\_Analysis\\_of\\_xArm6\\_Robot\\_for\\_Activities\\_of\\_Daily\\_Living/links/62643f3a1b747d19c2a1af14/Kinematics-and-Workspace-Analysis-of-xArm6-Robot-for-Activities-of-Daily-Living.pdf](https://www.researchgate.net/profile/Md-Samiul-Sunny-2/publication/360128896_Kinematics_and_Workspace_Analysis_of_xArm6_Robot_for_Activities_of_Daily_Living/links/62643f3a1b747d19c2a1af14/Kinematics-and-Workspace-Analysis-of-xArm6-Robot-for-Activities-of-Daily-Living.pdf)

<https://read.qxmd.com/read/35719427/kinematics-and-workspace-analysis-of-xarm6-robot-for-activities-of-daily-living>

<https://github.com/robotics-ETF/xarm6-etf-lab>

<https://ieomsociety.org/proceedings/2021dhaka/488.pdf>

<https://blog.csdn.net/ufactory/article/details/149772905>

<https://wiki.ros.org/xarm>

<https://www.manualslib.com/manual/3211390/Ufactory-Xarm.html>

<https://diku-dk.github.io/image-website/robotlab/xarm6>

estoy trabajando con el ufactory xarm6, quiero hacer el estudio cinemático, está publicado por el fabricante?

Sí, UFACTORY **sí publica información oficial** para poder realizar el estudio cinemático del xArm6, aunque no siempre aparece explícitamente como “estudio cinemático” en un único documento. Te resumo lo que existe y dónde encontrarlo, con base en lo que aparece en los resultados de búsqueda.



## 1. Documentación oficial del fabricante

En la página de descargas de UFACTORY se incluyen varios documentos técnicos del xArm6, entre ellos:

- **Developer Manual del xArm6**
- **Modelo 3D y URDF del robot**
- **Parámetros DH (Denavit–Hartenberg)**
- **User Manual completo**

Estos recursos están disponibles en la sección de descargas del fabricante.



**Los parámetros DH oficiales suelen estar en el Developer Manual o en los archivos URDF**, que son suficientes para derivar la cinemática directa e inversa.



## 2. Archivos URDF del xArm6

El URDF del xArm6 aparece también en repositorios públicos, como este en GitHub. El URDF contiene:

- Longitudes de eslabones
- Ejes de rotación
- Transformaciones entre articulaciones

Con eso puedes construir la cinemática directa e inversa sin problema.



## 3. Documentación no oficial pero útil

Hay un proyecto en GitHub que mapea la cinemática del xArm6 usando MATLAB, incluyendo:

- forward\_kinematics.m
- inverse\_kinematics.m
- workspace.m
- Un PDF explicativo

Este repositorio está aquí.



## Conclusión

Sí: **el fabricante publica la información necesaria para hacer el estudio cinemático**, aunque no siempre empaquetada como un “estudio cinemático” formal. Lo esencial (DH, URDF, geometría, límites articulares) está disponible en:

- **Developer Manual del xArm6**
- **URDF oficial del robot**
- **User Manual**

Si quieres, puedo ayudarte a **extraer los parámetros DH, montar la cinemática directa**, o incluso **generar el modelo en MATLAB o Python**.

quiero los parámetros denavit hartenberg, porque de ahí puedo hacer el proceso inverso para dibujar todos los sistemas de referencias

Ver mi razonamiento

Buena idea partir de los DH para luego construir todos los sistemas de referencia. Te marco algo importante primero:

- Los **parámetros Denavit–Hartenberg “oficiales” del xArm6 no están publicados de forma explícita por UFACTORY** en una tablita ya hecha (al menos no de acceso público directo).
- Pero sí se pueden obtener de dos sitios:
  - De trabajos académicos donde modelan el xArm6.
  - Del URDF/modelo CAD oficial del robot, a partir del cual se derivan los DH.

Vamos por partes.

### 1. DH del xArm6 en literatura académica

Hay al menos un artículo donde se hace el análisis cinemático y workspace del xArm6, y por tanto definen el modelo y los parámetros necesarios para la cinemática directa. Es el trabajo:

- **“Kinematics and Workspace Analysis of xArm6 Robot for Activities of Daily Living”**.

En ese paper formulan la cinemática del xArm6 y, aunque la tabla DH no esté siempre destacada con ese nombre, sí está toda la información geométrica y de juntas que permite reconstruirlos. Te puede servir como referencia “no oficial” pero bastante rigurosa.

### 2. Repositorios con modelo cinemático del xArm6

También hay proyectos tipo académico/curso donde ya han implementado la cinemática del xArm6, por ejemplo:

- Repositorio **“Xarm-6-robot”** en GitHub, donde incluyen cinemática directa, inversa, jacobianos, etc. Ahí suelen tener:
  - Scripts de cinemática directa.
  - Funciones donde se ve claramente cómo están fijados los frames.

- A veces una tabla DH o lo equivalente, aunque usen una convención ligeramente distinta.

Esto no es “del fabricante”, pero te da una base muy buena para comparar con lo que tú derives.

### 3. Cómo obtener los DH a partir del modelo oficial

Aunque el fabricante no te dé la tabla DH directamente, sí tienes la info clave en:

- **URDF oficial del xArm6** (en el SDK/ROS package).
- **Modelo CAD y documentación del xArm6** (en la web de UFACTORY; suelen venir longitudes de eslabones, offsets, etc.).

A partir de ahí:

#### 1. Lees el URDF:

- Cada `joint` te da:
  - Tipo (revolute).
  - Eje de rotación.
  - Transformación estática `origin` (xyz + rpy) entre links.

#### 2. Fijas la convención DH (clásica o modificada) y vas encajando:

- $a_i$ : distancia entre ejes  $Z_i$  y  $Z_{i+1}$  a lo largo de  $X_i$ .
- $\alpha_i$ : ángulo entre  $Z_i$  y  $Z_{i+1}$  en torno a  $X_i$ .
- $d_i$ : desplazamiento a lo largo de  $Z_i$ .
- $\theta_i$ : ángulo en torno a  $Z_i$  (variable articular).

Si quieres, en el siguiente paso puedo hacer contigo algo concreto:

- O bien: **derivamos juntos una tabla DH** a partir de una convención clara (me dices si usas DH clásico o modificado, y si trabajas en ROS, MATLAB, etc.).
- O bien: **te doy una tabla DH propuesta** (basada en la geometría estándar del xArm6) y tú la ajustas a tu sistema de referencia global.

Para poder darte ya una tabla DH directamente, dime:

1. ¿Quieres **DH clásico** o **DH modificado**?
2. ¿En qué entorno vas a usarla (MATLAB, Python/ROS, otra cosa)?
3. ¿Te da igual que la tabla no sea “exactamente” la del fabricante mientras sea geoméricamente equivalente (mismas poses finales)?

y qué me dices del control dinámico?

Ver mi razonamiento

Buena pivotada: pasar de la cinemática a la dinámica es donde el robot “enseña los dientes” de verdad.

## 1. ¿El fabricante da un modelo dinámico completo?

No, UFACTORY no publica (que yo haya visto de forma abierta) un **modelo dinámico completo**

**tipo: matrices**  $M(q)$ ,  $C(q, \dot{q})$ ,  $g(q)$  ni una tabla de **parámetros inerciales identificados** (masas, tensores de inercia de cada eslabón, etc.) como hace, por ejemplo, KUKA o UR grandes. En el manual de usuario se centran en:

- **Datos nominales:** cargas máximas, alcances, velocidades, etc.
- **Aspectos de seguridad, instalación y uso**, pero no ecuaciones dinámicas ni parámetros inerciales detallados.

Para control dinámico “serio” (computed torque, inverse dynamics, etc.) tienes que tirar de:

- Modelo genérico + estimación de masas/inercia.
- O trabajos académicos y proyectos que ya hicieron parte del camino.

## 2. Qué sí existe (y te sirve como base)

Hay al menos un repositorio docente sobre el xArm6 donde ya incluyen:

- **Dinámica, jacobianos, generación de trayectorias, etc.** para el xArm6, dentro de un paquete de prácticas de robótica.

Ese tipo de repositorios suelen tener:

- Una **estructura de Denavit–Hartenberg** o equivalente.
- Una implementación de la dinámica con Euler–Lagrange o Newton–Euler.
- A veces un conjunto de **parámetros inerciales supuestos** para poder simular.

No es “oficial UFACTORY”, pero es el punto de partida perfecto si lo que quieres es:

- Validar tu propio modelo.
- Tener algo funcional mientras no conoces los parámetros reales.

## 3. Si quieres hacer control dinámico “de verdad”

Para un control tipo **computed torque** o **inverse dynamics control**, necesitas:

### 1. Modelo simbólico o numérico:

$$\tau = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) + \tau_{\text{fric}}$$

### 2. Parámetros inerciales por eslabón:

- **Masa.**
- **Centro de masas** respecto al frame del eslabón.
- **Tensor de inercia.**

### 3. **Modelo de fricción** (al menos viscosa + Coulomb).

Dado que el fabricante no los da:

- Puedes **suponerlos** a partir de:
  - Pesos totales y distribución aproximada.
  - Geometría CAD (si la tienes) calculando inercias.
- O bien hacer un **identificado experimental**: excitar el robot con trayectorias conocidas, leer pares de motor y resolver un problema de mínimos cuadrados para los parámetros del modelo.

### 4. En la práctica, ¿qué hace la mayoría?

Para robots tipo xArm6, en entorno académico se suele:

- Usar una **dinámica aproximada** para simulación y control por par calculado.
- En el robot real, apoyarse más en:
  - **Control de par interno del fabricante** que ya compensa parte de gravedad y fricción.
  - **Control en espacio de tareas o de articulaciones** por posición/velocidad alrededor de ese lazo interno.

Es decir, tu modelo dinámico sirve para:

- Planificación y simulación.
- Controles avanzados (impedancia, admisión, etc.), donde la exactitud ayuda mucho.

### 5. Cómo puedo ayudarte de forma concreta

Si me dices qué quieres exactamente:

- **a)** ¿Quieres ecuaciones dinámicas simbólicas (Euler–Lagrange) a partir de unos DH que definamos?
- **b)** ¿Quieres un script en MATLAB/Python que te devuelva  $M(q)$ ,  $C(q, \dot{q})$ ,  $g(q)$ ?
- **c)** ¿Quieres una estrategia para identificar parámetros inerciales en el propio xArm6?

Puedo:

- Proponer una **estructura DH consistente** (a partir de lo que ya hablamos).
- A partir de ahí, guiarte paso a paso para montar el modelo dinámico y el controlador que tengas en mente.

¿En qué entorno trabajas el control: MATLAB/Simulink, ROS + Python, C++...? Y, siendo sincero: ¿aspiras a control dinámico implementado en el propio robot, o “solo” simulación y estudio académico? Ahí está el cruce de caminos.