



Generación de trayectorias para el brazo robótico (ArmX)

Daily Milanés Hermosilla Alejandro Castilla Pérez

RESUMEN / ABSTRACT

En el presente trabajo se realizó la modelación y simulación de la cinemática directa e inversa de un robot articular de cuatro grados de libertad que tiene pinza como efector final para manipular objetos, y tiene definida su superficie de trabajo en un espacio tridimensional. Se realiza el control cinemático del robot para el seguimiento de trayectorias punto a punto, coordinadas y continuas para el extremo. Se realiza el control dinámico desacoplado utilizando una acción de control on-off. Se empleó la plataforma de hardware libre Arduino, específicamente la placa Arduino Mega 2560 y su entorno de desarrollo. Además se diseñó e implementó una placa del tipo escudo con la electrónica adicional necesaria para el control del manipulador. Por último, se implementó una interfaz gráfica utilizando como herramienta el software LabWindows / CVI 9.0 de National Instruments, que permite al usuario enviar desde la PC los valores deseados en coordenadas articulares y cartesianas, incluyendo la posición de la pinza, así como la planificación de tareas y de tipos de trayectorias.

Palabras claves: Modelo cinemático, control cinemático, control dinámico, generación de trayectorias, interfaz de usuario.

In this paper the modeling and simulation of direct and inverse kinematics of a robot joint four degrees of freedom that clip as end effector to manipulate objects was made, and has defined its working surface in three dimensional space. Kinematic robot control trajectories for tracking point to point, coordinated and continuous to the end is performed. Dynamic control is performed using an action undocked on-off control. Free hardware platform Arduino, Arduino Mega2560 specifically and its development environment was used. It was also designed and implemented a shield plate type with additional electronics needed to control the manipulator. Finally, a graphical interface using as a tool the LabWindows / CVI 9.0 software from National Instruments, which allows the user to send from the PC the desired joint and Cartesian coordinate values, including the position of the clamp and planning was implemented tasks and types of trajectories.

Key words: Kinematic model, Kinematic control, dynamic control, tracking trajectories, user interface.

Trajectories generation for the robotic (ARMX).

INTRODUCCIÓN

En la actualidad se ve a la robótica como un campo de trabajo muy amplio desarrollando nuevas tecnologías en una serie de áreas interdisciplinarias como la cinemática, la dinámica, la planificación de sistemas, el control y la inteligencia artificial. Los sistemas robóticos están siendo desarrollados cada vez más con mayor autonomía, por lo que implica un reto involucrarse en este auge tecnológico. El campo de la robótica es muy amplio, así por ejemplo la vemos unida a la esfera productiva, a la investigación científica, a la medicina. De acuerdo a su uso y aplicación estos tienen diversas características[1].

La producción de los robots se destaca por traer consigo una disminución de la mano de obra; además ayuda a una mayor calidad del producto acabado, a la rapidez de la producción. Casi siempre, en la industria, los robots se unen a otras máquinas aportando mayor eficiencia en la producción. En la esfera científica, muchos de ellos son utilizados para hacer investigaciones en el campo donde el hombre se le hace difícil ir, tal vez por un medio hostil o tal vez demasiado peligroso. En el ámbito industrial, específicamente en las pequeñas y medianas empresas, se crea la necesidad de incentivar a la adquisición de estas tecnologías para utilizarlas en aplicaciones como el apilamiento de estructuras, tareas de soldadura,

pintura, trazado, corte y perforación de placas metálicas. Sin contar aquellas aplicaciones más complejas como son la industria de servicios, incluyendo los médicos, agricultura, etc.

Para lograr que un robot realice un tarea lo más precisa posible se requiere de varios elementos: los actuadores encargados de generar la fuerza para darle movimiento a las piezas mecánicas del manipulador, sensores que permiten medir las características del mundo que rodea al robot, modelos cinemáticos para evaluar la posición y orientación del extremo, generadores de trayectorias que incluyen los valores de posición, velocidad, tiempo invertido, tipo de trayectoria, etc. fijados por el usuario, el tipo de control implementado que puede ser PID, adaptivo, predictivo, par calculado, entre otros.

En la Universidad de Oriente, no existen mecanismos robóticos instalados para el desarrollo de trabajos científicos. Específicamente en la Facultad de Ingeniería Eléctrica, en el departamento de Control Automático fue donado un pequeño robot, se trata del robot ArmX K-680, el cual cuenta con un software propio para establecer la comunicación con una PC, pero los movimientos realizables por el robot a través del mismo se fundamentan en la medición de tiempo, no existiendo la posibilidad de implementar un control de posición adecuado para cada una de las articulaciones. La ejecución de tareas desde esta interfaz no posibilita además variar la velocidad con que se realizan los movimientos, limitando de esta forma las aplicaciones del robot, tampoco se cuenta con opciones de programación y memorización de tareas y seguimiento de trayectorias motivo por el cual no se explotan las potencialidades del manipulador.

Esta investigación tiene como objetivos la obtención de los modelos directo e inverso de la cinemática, el diseño e implementación de una placa impresa para incorporar la electrónica necesaria para realizar el control del manipulador y el diseño e implementación de una interfaz gráfica de usuario para la comunicación del Robot ArmX con la PC utilizando una tarjeta de desarrollo Arduino.

1.-ESTRUCTURA Y ARQUITECTURA DEL ROBOT ARMX

Debido a su estructura el Robot ArmX se clasifica como angular antropomórfico (Fig. 1). Esta configuración es útil cuando se requiere tener acceso a espacios cerrados y para tareas de manipulación complejas. Por esta razón es la configuración más utilizada en la industria.

En la estructura del robot manipulador se distingue el brazo, formado por eslabones con sus respectivas articulaciones y el efector final, así como la base del mismo con el correspondiente hardware asociado. Como elementos actuadores se dispone de cinco motores de CD para el control de las 4 articulaciones y el efector final.

Las estructuras de las articulaciones del Robot ArmX son todas rotacionales y con un grado de libertad, entonces teniendo en cuenta que el robot posee cuatro articulaciones se define como un robot de cuatro grados de libertad.

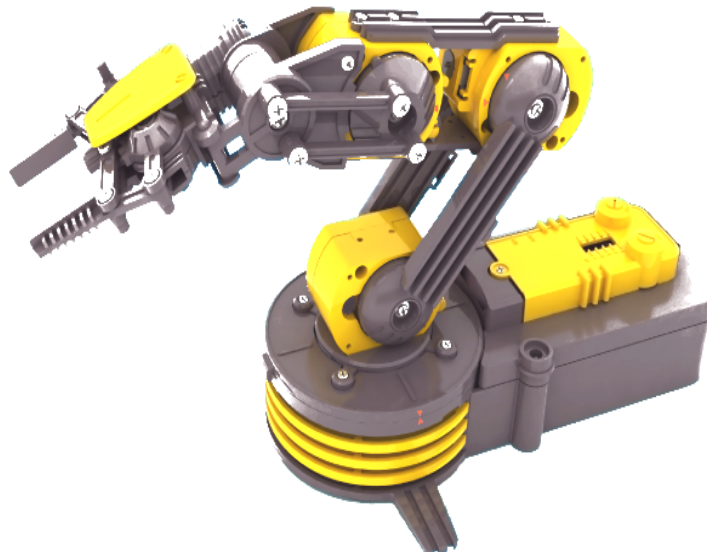


Figura 1

Estructura del Robot ArmX.

En la Tabla 1 se muestran los valores de los eslabones que conforman la estructura del manipulador, así como el rango de valores para las rotaciones de cada articulación, la apertura máxima de la pinza y el peso máximo de los objetos que puede soportar.

Tabla 1
Características de eslabones y articulaciones

Eslabón	Longitud (mm)	Rango (°)
1 (L1)	65.5	0 a 270
2 (L2)	91	0 a 180
3 (L3)	113	0 a 300
4 (L4)	113	0 a 120
Pinza	1.77 pulgadas	hasta 3.5 oz

2.- MODELADO CINEMÁTICO DEL ROBOT ARMX

La cinemática del robot estudia el movimiento del mismo con respecto a un sistema de referencia. Así, la cinemática se interesa por la descripción analítica del movimiento espacial del robot como una función del tiempo, y en particular por las relaciones entre la posición y orientación del extremo final del robot y los valores que toman sus coordenadas articulares. Lograr obtener esta descripción del movimiento del robot se plantea desde dos puntos de vista, el problema cinemático directo y el problema cinemático inverso.

2.1.- MODELO CINEMÁTICO DIRECTO

La cinemática directa consiste en determinar la posición y orientación del extremo final del robot con respecto al sistema de la base del robot a partir del conocimiento de los valores de las articulaciones y sus parámetros geométricos. Para articulaciones rotacionales su valor estará determinado por un ángulo.

Para dar solución al problema cinemático directo del robot Armx se utilizó el algoritmo de Denavit-Hartenberg [1], mediante el cual el problema cinemático directo se reduce a encontrar la matriz de transformación homogénea T que relaciona la posición y orientación del extremo del robot respecto a su sistema de referencia fijo (la base). Esta matriz T está en función de los parámetros de las articulaciones del robot. A partir del desarrollo del algoritmo los sistemas de coordenadas que se obtienen se muestran en la Fig. 2.

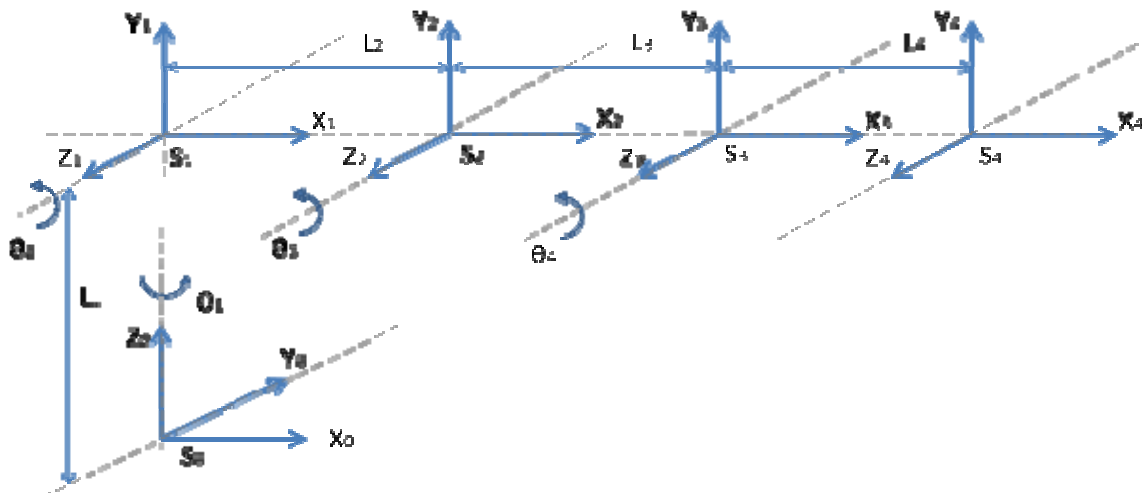


Figura 2

Representación del robot en sistemas de coordenadas según algoritmo de Denavit-Hartenberg.

Para el robot ArmX los parámetros de Denavit-Hartenberg se muestran en la Tabla 2.

L1, L2, L3 y L4 son las longitudes de los eslabones.

$\theta_1, \theta_2, \theta_3, \theta_4$: Posición angular de cada articulación.

Tabla 2
Parámetros de Denavit-Hartenberg

Articulación	θ	d	a	α
1	θ_1	L1	0	90°
2	θ_2	0	L2	0
3	θ_3	0	L3	0
4	θ_4	0	L4	0

A partir de estos parámetros se obtuvieron las ecuaciones que permiten calcular los valores de la posición del extremo del robot en coordenadas cartesianas.

$$\begin{aligned}
 X_0 = & 91 * \cos(\theta_1) * \cos(\theta_2) + 113 * \cos(\theta_4) * (\cos(\theta_1) * \cos(\theta_2) * \cos(\theta_3) - \cos(\theta_1) * \sin(\theta_2) * \sin(\theta_3)) - 113 \\
 & * \sin(\theta_4) * (\cos(\theta_1) * \cos(\theta_2) * \sin(\theta_3) + \cos(\theta_1) * \cos(\theta_3) * \sin(\theta_2)) + 113 * \cos(\theta_1) * \cos(\theta_2) \\
 & * \cos(\theta_3) - 113 * \cos(\theta_1) * \sin(\theta_2) * \sin(\theta_3)
 \end{aligned} \tag{1}$$

$$\begin{aligned}
 Y_0 = & 91 * \cos(\theta_2) * \sin(\theta_1) - 113 * \cos(\theta_4) * (\sin(\theta_1) * \sin(\theta_2) * \sin(\theta_3) - \cos(\theta_2) * \cos(\theta_3) * \sin(\theta_1)) - 113 \\
 & * \sin(\theta_4) * (\cos(\theta_2) * \sin(\theta_1) * \sin(\theta_3) + \cos(\theta_3) * \sin(\theta_1) * \sin(\theta_2)) - -113 * \sin(\theta_1) * \sin(\theta_2) \\
 & * \sin(\theta_3) + +113 * \cos(\theta_2) * \cos(\theta_3) * \sin(\theta_1)
 \end{aligned} \tag{2}$$

$$\begin{aligned}
 Z_0 = & 91 * \sin(\theta_2) + 113 * \cos(\theta_2) * \sin(\theta_3) + 113 * \cos(\theta_3) * \sin(\theta_2) + 113 * \cos(\theta_4) * \\
 & * (\cos(\theta_2) * \sin(\theta_3) + \cos(\theta_3) * \sin(\theta_2)) + 113 * \sin(\theta_4) * (\cos(\theta_2) * \cos(\theta_3) - \sin(\theta_2) * \\
 & * \sin(\theta_3)) + 66.5
 \end{aligned} \tag{3}$$

2.2.- MODELO CINEMÁTICO INVERSO

El objetivo del problema cinemático inverso consiste en encontrar los valores que deben adoptar las coordenadas articulares del robot, para que su extremo se posicione y oriente según una determinada localización espacial. La resolución no es sistemática, depende de la configuración del robot y pueden existir soluciones múltiples.

Para robots de pocos grados de libertad, como es el caso del manipulador ArmX, es conveniente usar métodos geométricos (Fig. 3) los cuales utilizan relaciones trigonométricas y geométricas en los elementos que se forman en el robot [2- 5].

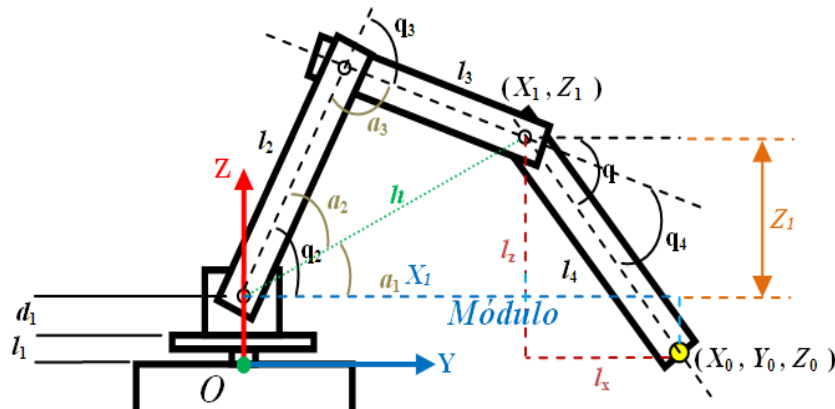


Figura 3

Representación del manipulador para obtener el modelo cinemático inverso mediante el método geométrico.

Las ecuaciones (4)-(12) permiten calcular los ángulos de rotación de las articulaciones [3].

$$q_1 = \tan^{-1} \left(\frac{Y_0}{X_0} \right) \quad (4)$$

$$l_x = l_4 \cos(q) \quad (5)$$

$$l_z = l_4 \text{sen}(q) \quad (6)$$

$$X_1 = \sqrt{X_0^2 + Y_0^2} - l_x \quad (7)$$

$$Z_1 = Z_0 - l_z - (d_1 + l_1) \quad (8)$$

$$h = \sqrt{X_1^2 + Z_1^2} \quad (9)$$

$$q_2 = \tan^{-1} \left(\frac{Z_1}{X_1} \right) + \cos^{-1} \left(\frac{l_2^2 - l_3^2 + h^2}{2 l_2 h} \right) \quad (10)$$

$$q_3 = \cos^{-1} \left(\frac{l_2^2 - l_3^2 - h^2}{2 l_2 l_3} \right) - 180^\circ \quad (11)$$

$$q_4 = q - q_2 - q_3 \quad (12)$$

donde:

q_1, q_2, q_3, q_4 : posiciones articulares.

x_0, y_0, z_0 : posición deseada para el extremo.

3.-CONTROL CINEMÁTICO Y DINÁMICO DEL ROBOT ARMX

3.1- GENERACIÓN DE TRAYECTORIAS

Para realizar una tarea determinada el robot debe moverse desde un punto inicial a uno final. Este movimiento puede ser realizado según infinitas trayectorias espaciales. De todas ellas hay algunas que, bien por su sencillez de implementación por parte del control cinemático o bien por su utilidad y aplicación a diversas tareas, son las que en la práctica incorporan los robots comerciales. De este modo, puede encontrarse que los robots dispongan de trayectorias punto a punto, coordinadas y continuas [1,2,6-8].

3.1.1- TRAYECTORIAS PUNTO A PUNTO

En este tipo de trayectorias cada articulación evoluciona desde su posición inicial a la final sin realizar consideración alguna sobre el estado o evolución de las demás articulaciones. Normalmente, cada actuador trata de llevar a su articulación al punto de destino en el menor tiempo posible, pudiéndose distinguir dos casos: movimiento eje a eje y movimiento simultáneo de ejes.

3.1.2.- TRAYECTORIAS COORDINADAS O ISOCRONAS

Para evitar que algunos actuadores trabajen forzando sus velocidades y aceleraciones, teniendo que esperar después la conclusión del movimiento de la articulación más lenta, puede hacerse un cálculo previo, averiguando cuál es esta articulación y qué tiempo invertirá. Se ralentizará entonces el movimiento del resto de los ejes para que inviertan el mismo tiempo en su movimiento, acabando todos ellos simultáneamente. Se tiene así que todas las articulaciones se coordinan comenzando y acabando su movimiento a la vez, adaptándose todas a la más lenta. El tiempo total invertido en el movimiento es el menor posible y no se piden aceleraciones y velocidades elevadas a los actuadores de manera inútil. Desde el punto de vista del usuario la trayectoria que describe el extremo del robot no es significativa, siendo ésta impredecible, aunque como es obvio, un conocimiento del modelo y control cinemático del robot permitiría su cálculo.

3.1.3.- TRAYECTORIAS CONTINUAS

Cuando se pretende que la trayectoria que siga el extremo del robot sea conocida por el usuario (trayectoria en el espacio cartesiano o de la tarea), es preciso calcular de manera continua las trayectorias articulares. Típicamente, las trayectorias que el usuario pretende que el robot describa son trayectorias en línea recta o en arco de círculo. El resultado será que cada

articulación sigue un movimiento aparentemente caótico con posibles cambios de dirección y velocidad y sin coordinación con el resto de las articulaciones. Sin embargo, el resultado conjunto será que el extremo del robot describirá la trayectoria deseada.

Para el manipulador ArmX se implementaron todos los tipos de trayectorias mencionados anteriormente, vale destacar que para el caso de la trayectoria continua se diseñó un generador de trayectoria lineal basado en la ecuación paramétrica de una recta en tres dimensiones (13), donde:

Pi: punto inicial

Pf: punto final

n: cantidad de puntos intermedios deseados por el usuario.

La variación del parámetro (t) permite hallar todos los puntos intermedios entre el punto inicial y el final, el rango de valores posibles es: $(0 \leq (t) \leq 1)$, coincidiendo los extremos con los puntos inicial y final respectivamente. Para cada punto calculado por el generador se obtienen las coordenadas cartesianas, y mediante el modelo cinemático inverso se obtienen las correspondientes coordenadas articulares. Finalmente el sistema de control del manipulador ejecuta cada posición entregada por el modelo de forma secuencial asegurando que la trayectoria descrita por el extremo sea la deseada. Mientras mayor sea el valor de (n) la trayectoria resulta más precisa.

$$\begin{aligned} P_x &= P_i(x) + (P_f(x) - P_i(x)) * t \\ P_y &= P_i(y) + (P_f(y) - P_i(y)) * t \\ P_z &= P_i(z) + (P_f(z) - P_i(z)) * t \end{aligned} \quad (13)$$

3.2.- LA PLATAFORMA ARDUINO

Arduino es una plataforma de hardware libre, basada en una placa con un microcontrolador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios [9].

3.2.1.- PLACA ARDUINO MEGA2560

En el desarrollo del trabajo se utilizó como dispositivo controlador una placa Arduino Mega 2560 [9- 11] (Fig. 4).

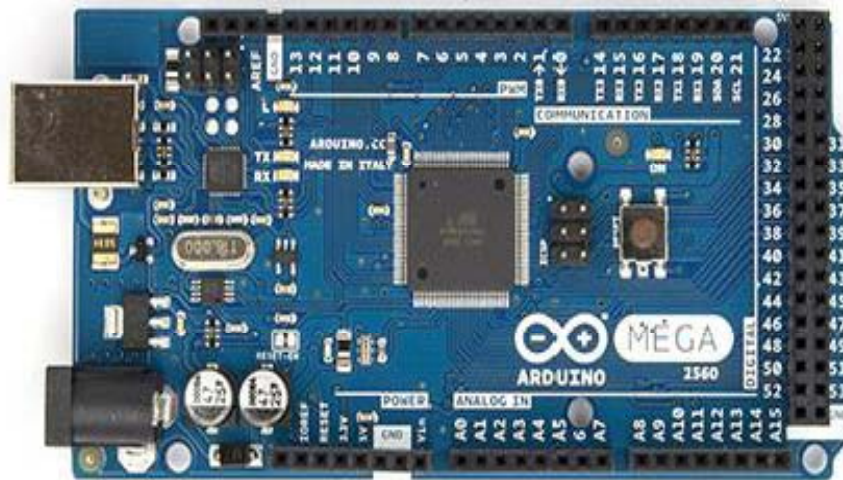


Figura 4

Placa Arduino Mega 2560.

Características de la Placa Arduino Mega 2560:

- Microcontrolador: ATmega2560.
- Voltaje de operación: 5 V.

- Voltaje de entrada límite: 6 - 20 V.
- Voltaje de entrada recomendado: 7 - 12 V.
- 54 pines de E/S digitales (15 pueden ser utilizados como PWM).
- 16 entradas analógicas.
- Intensidad DC por cada pin E/S: 40 mA
- Intensidad DC por cada pin 3.3V: 50 mA
- 4 UARTs (puertos serie).
- Memoria flash: 256kB.
- EEPROM: 4kB
- SRAM: 8kB.
- Frecuencia del Reloj: 16MHz.

3.3.- CONTROL DE LOS MOTORES DE CD

El control de los motores de CD posibilita el movimiento de las articulaciones y del efector final a posiciones deseadas y fijadas por los usuarios. Con este fin se utilizó el driver L293D, este circuito está conformado por dos puentes H, y está perfectamente capacitado para controlar los cinco motores de CD del brazo manipulador, para lo cual se necesitaron tres circuitos de este tipo.

3.3.1.- INCORPORACIÓN DE CODIFICADORES ÓPTICOS

Para realizar un control efectivo de cada articulación, es necesario incorporar sensores para conocer la posición de cada articulación. Se propone el uso de dispositivos electrónicos conocidos como codificadores ópticos. En este trabajo se utilizaron codificadores ópticos reciclados, se extrajeron de antiguos mouse de PC, en los cuales se utiliza para determinar la posición del cursor en pantalla. Se instalaron en el interior de la caja reductora de cada motor, generando una señal periódica entre 0 y 5V de DC o 0 y 1 lógico.

3.3.2.- CONTROL DINÁMICO

El control dinámico tiene como objetivo procurar que las trayectorias realmente seguidas por el robot sean lo más parecidas posible a las propuestas por el control cinemático como trayectorias deseadas. Para llevar a cabo esa tarea hace uso del conocimiento del modelo dinámico del robot y de las herramientas de análisis y diseño aportadas por la teoría de control (estabilidad de Lyapunov, control PID, control adaptativo, etc.).

La estrategia de control utilizada se fundamenta en el uso de un controlador de dos posiciones (on /off) (Fig. 5), que varía entre dos estados fijos, activado o desactivado. El bloque G1 representa el algoritmo matemático para la conversión de la posición deseada (entrada del sistema) a su valor equivalente en pulsos, dicho valor será comparado con la señal proveniente del codificador óptico, equivalente a la posición real (salida del sistema), y así generar la señal de error como entrada del controlador. En este tipo de control la señal de salida del controlador $u(t)$ permanece en un valor ya sea máximo o mínimo, dependiendo de si la señal de error $e(t)$ es positiva o negativa. Destacar que se implementó un control desacoplado o monoarticular, es decir, un control para cada articulación del manipulador de modo independiente.

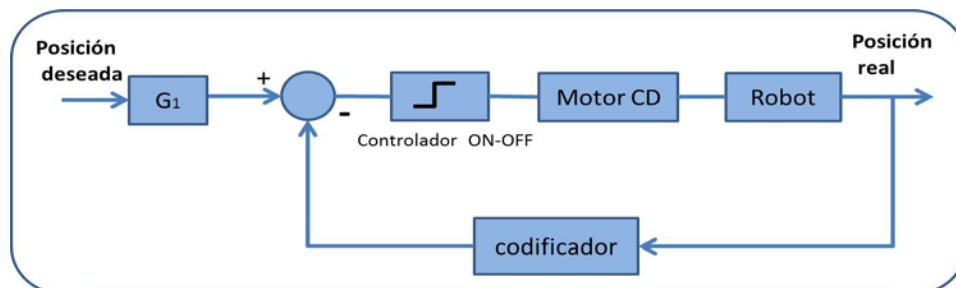


Figura 5

Esquema del control de posición para una articulación del manipulador.

3.4.- IMPLEMENTACIÓN DE LA PLACA IMPRESA

Se agrupó la electrónica adicional necesaria para el control del manipulador en una tarjeta de circuito impreso (Fig. 6), para su diseño se utilizó el software multidisciplinario Fritzing versión 0.9.2.

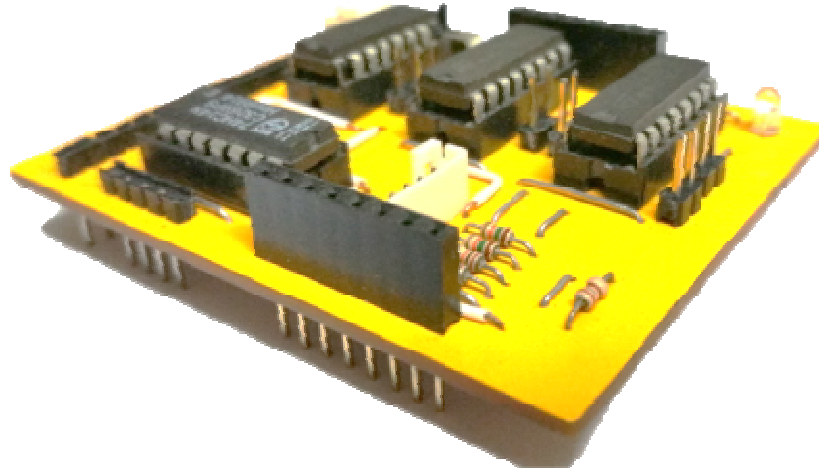


Figura 6

Circuito Impreso desarrollado.

Los componentes incorporados a la tarjeta fueron:

- 3 Driver tipo L293D.
- 1 Inversor Schmitt –triggers 74HC14.
- 4 Resistencias de 0.5 W.
- Múltiples conectores para motores y codificadores ópticos.

La versatilidad de esta tarjeta radica en la eliminación de un gran número de conductores, ya que la misma se diseñó del tipo escudo, capaz de conectarse directamente a la placa Arduino Mega.

A continuación se muestra una imagen del manipulador conectado con la placa de desarrollo Arduino Mega 2560 y la placa impresa desarrollada (Fig. 7).

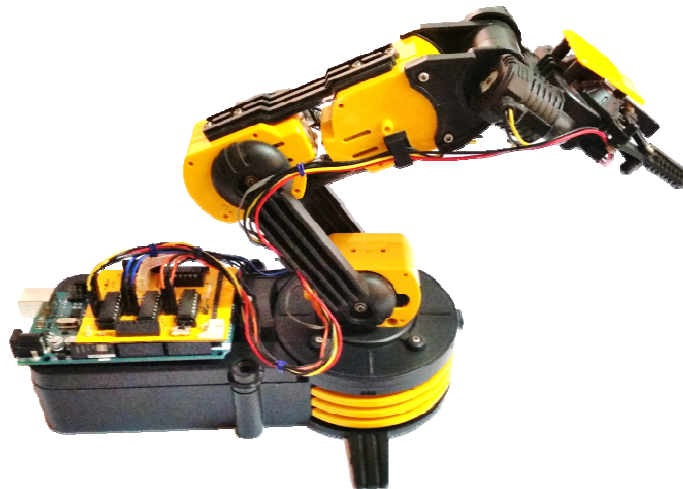


Figura 7

Vista del ensamblaje final del manipulador, la placa Arduino Mega 2560 y la placa electrónica.

4.- INTERFAZ GRÁFICA

Una interfaz de comunicación es un dispositivo de enlace entre un puerto de salida de una computadora y los componentes eléctricos o electrónicos de un sistema [12- 15](Fig. 8).

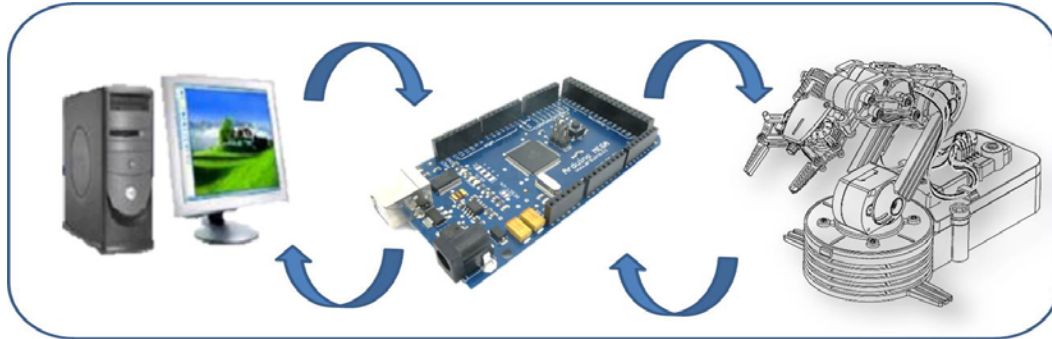


Figura 8

Interfaz de comunicación mediante la plataforma Arduino.

Con los objetivos de posibilitar la comunicación con el brazo manipulador y el control del mismo, así como de comprobar los modelos cinemáticos obtenidos, el generador de trayectoria lineal implementado y el diseño de hardware propuesto, se implementó una interfaz de usuario (Fig. 9). Esta interfaz se realizó en el software LabWindows / CVI 9.0 de National Instruments.

De modo general la interfaz se compone de tres paneles principales: Cinemática, Modo Programable y Configuración.

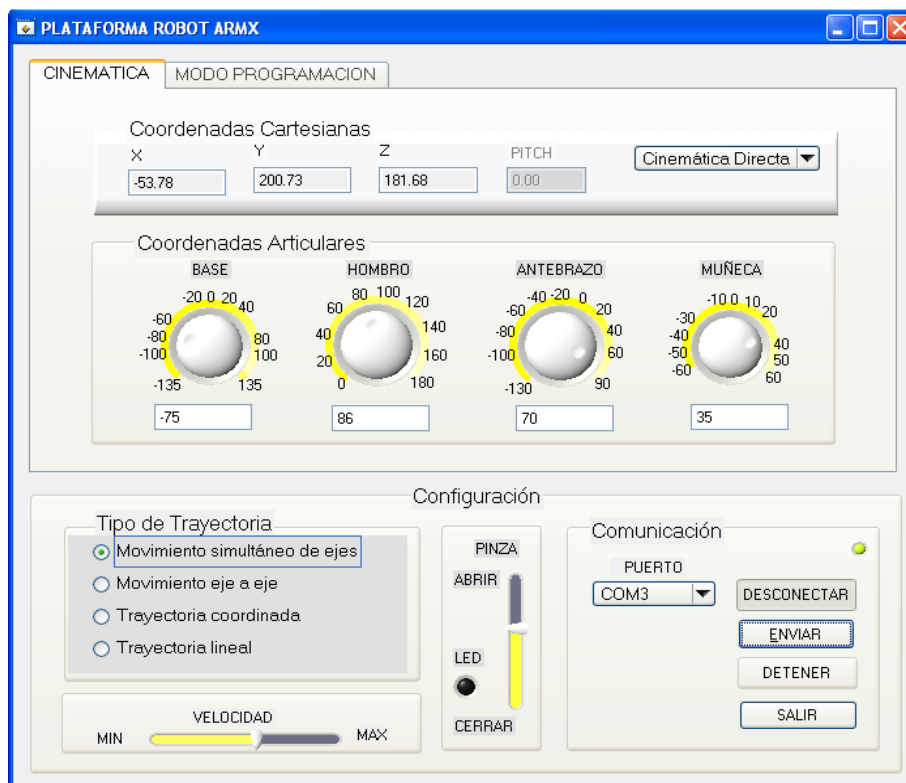


Figura 9

Interfaz de Usuario elaborada en LabWindows.

El panel Cinemática (Fig. 10 y 11) permite la realización de movimientos por el manipulador desde un punto inicial a un punto final, para ello se puede seleccionar entre el modo cinemática directa o cinemática inversa mediante el menú desplegable ubicado en el extremo superior derecho del panel.



Figura 10
Panel Cinemática directa.

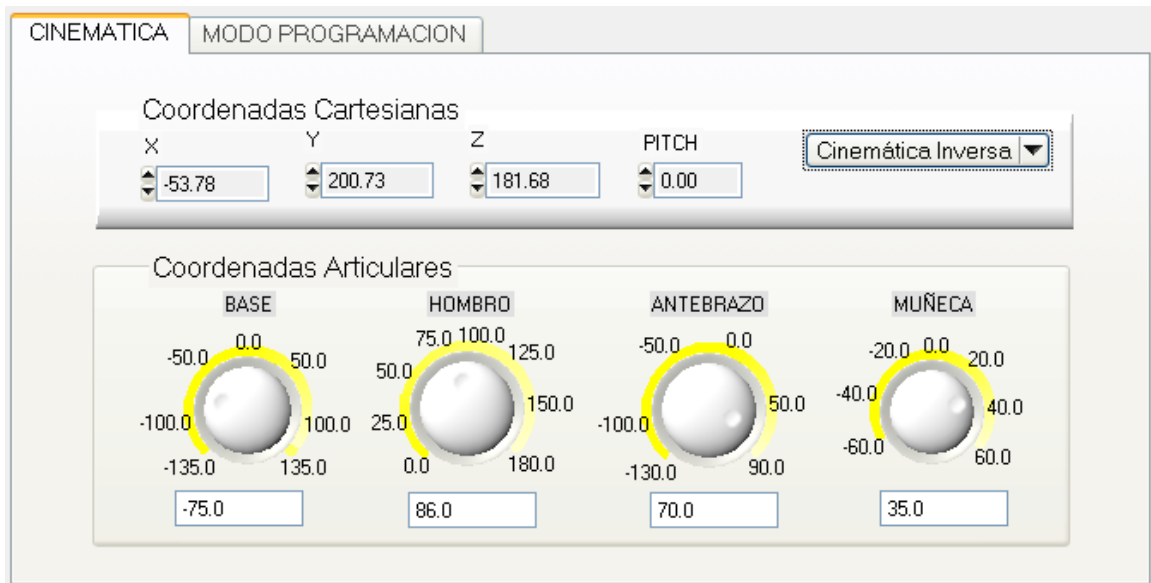


Figura 11
Panel Cinemática inversa.

En dependencia del modo seleccionado se habilitan o deshabilitan los paneles secundarios denominados Coordenadas Cartesianas y Coordenadas Articulares, permitiendo así su funcionamiento como controles o como indicadores:

- En el panel denominado Coordenadas Articulares se encuentran los controles asociados a cada una de las articulaciones del brazo manipulador, los cuales permiten el ajuste del ángulo deseado por el usuario (en grados) cuando se tiene

seleccionado el modo cinemática directa, teniendo en cuenta el rango permitido por la articulación. Además posibilita la visualización de las coordenadas articulares necesarias para posicionar el manipulador en un punto determinado cuando es seleccionado el modo cinemática inversa.

- El panel titulado Coordenadas Cartesianas permite fijar las coordenadas (x, y, z) del punto final deseado para posicionar el manipulador, así como el ángulo de cabeceo de la herramienta (pitch), mediante el modo cinemática inversa, tomando como referencia la base del mismo. Igualmente se obtienen en él los valores correspondientes a las coordenadas (x, y, z) en las que se encuentra el extremo del manipulador luego de ser ejecutado algún movimiento en el modo cinemática directa.

El panel Modo Programación (Fig. 12) posee las herramientas necesarias para la programación de tareas para el manipulador. Luego de ser seleccionada la tarea a programar, el botón “nueva” posibilita iniciar la programación. Las tareas pueden estar conformadas por tantas posiciones como sean necesarias para su implementación. La especificación de las posiciones se realiza en coordenadas cartesianas y se almacenan mediante el uso del botón “añadir”. Además es posible incluir el accionamiento de la pinza como parte de las posiciones. El botón “guardar” permite culminar con la programación de una tarea. Mediante el control “repeticiones” se puede definir el número de veces que se ejecutará la tarea seleccionada.

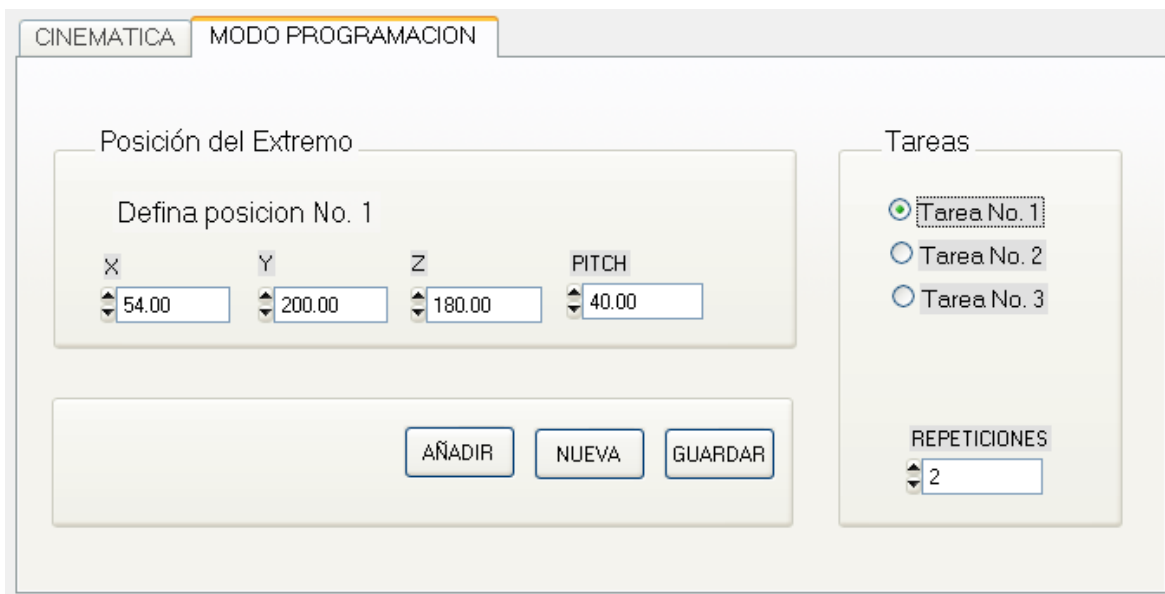


Figura 11

Panel Modo Programación.

En la parte inferior de la interfaz se localiza el panel Configuración (Fig. 13), el cual posee varias funcionalidades relacionadas con la comunicación y la trayectoria a realizar por el manipulador:

- En el extremo inferior derecho se encuentra el panel secundario Comunicación, que permite realizar los ajustes necesarios para la conexión con el dispositivo. Mediante un menú desplegable se selecciona el puerto por el cual se realizará la comunicación y el botón “conectar/desconectar” nos permite iniciar o terminar la misma. Al presionar el botón “enviar” se realiza el movimiento deseado por el usuario, teniendo en cuenta el modo cinemático que esté seleccionado, o la tarea previamente programada que esté seleccionada, en caso de encontrarse en el Modo Programación. El botón “detener/continuar” permite pausar o detener la realización de un movimiento en cualquier instante así como la reanudación del mismo.
- En el extremo inferior izquierdo se sitúa el panel secundario Tipo de Trayectoria, donde es posible seleccionar entre las trayectorias punto a punto, coordinada y lineal. El tipo de trayectoria seleccionada condicionará el movimiento realizado por el manipulador independientemente del modo cinemático escogido.



Figura 12
Panel Configuración.

El panel Configuración incorpora además una barra deslizante que permite ajustar la apertura de la pinza (entre totalmente abierta y totalmente cerrada). La barra deslizante denominada Velocidad, permite realizar ajustes de velocidad entre un valor mínimo y uno máximo, para todos los movimientos que realice el manipulador.

5.- RESULTADOS Y DISCUSIÓN

Los experimentos realizados a través de la interfaz y el manipulador permitieron validar los modelos cinemáticos directo e inverso, posicionando el extremo del manipulador en los valores fijados por el usuario, esto se comprobó de manera práctica utilizando instrumentos de medición.

Se establecieron varias posiciones para el extremo y para cada articulación, y siempre la diferencia entre los valores reales a los que llegaba el manipulador y los deseados por el usuario eran menos de 1 cm, o sea, que el método desarrollado fue bastante preciso para este tipo de aplicación.

Los resultados mostrados en la Tabla 3 muestran algunos ejemplos experimentales para posicionar el extremo.

Tabla 3
Resultados experimentales

Experimento	Valores fijados por el usuario (mm)	Valores reales alcanzados por el manipulador (mm)
1	X=100 Y=100 Z=150	X=98.5 Y=97.6 Z=143
2	X=111 Y= -80 Z=60	X=114 Y= -74 Z=61.5
3	X=226 Y= 0 Z=157	X=224 Y= 2.3 Z=152
4	X= 240 Y=200 Z=160	X= 232 Y=206 Z=154

Relacionado con los diferentes tipos de trayectorias implementadas, se comprobó mediante ensayos prácticos un correcto funcionamiento para las trayectorias punto a punto (movimiento eje a eje y simultáneo de ejes), sin embargo para trayectorias coordinadas y continuas el movimiento del manipulador se realiza de forma aproximada en comparación al movimiento deseado por el usuario, debido a las características inherentes de los motores de CD y del manipulador. En el caso de trayectorias coordinadas al tener que realizar movimientos que requieran una diferencia elevada entre las

velocidades de las articulaciones para que todas concluyan al mismo tiempo, puede ocurrir que el cálculo realizado para determinar la velocidad de una articulación arroje un valor tan pequeño que no pueda ser físicamente realizado por el motor. Para la trayectoria continua lineal ocurre que los juegos mecánicos del manipulador en los sistemas de transmisión de la caja reductora hacen que la realización de movimientos entre puntos sucesivos se aleje de los valores de referencia a los que debe de ir cada articulación.

6.- CONCLUSIONES

El presente trabajo permitirá el desarrollo de prácticas de laboratorio en la asignatura de Robótica y en cursos de postgrado en la carrera de Ingeniería en Automática. El algoritmo implementado permite la programación de tareas en los espacios articular y cartesiano, así como aplicaciones de manipulación de objetos en las cuales se conozca la posición del mismo en el espacio de trabajo, realizando trayectorias punto a punto, coordinada y lineal en el extremo. Se pretende continuar la investigación mediante la incorporación de sensores externos que permitan ampliar las esferas de aplicaciones, además de la incorporación de una cámara fija para la integración robot-visión en el espacio de trabajo.

RECONOCIMIENTOS

Agradecemos de manera especial a los profesores de la universidad de Ecuador por haber realizado su aporte al departamento de Control Automático de la Universidad de Oriente de la ciudad de Santiago de Cuba, sin su colaboración este trabajo no existiría.

REFERENCIAS

1. Barrientos, A. Fundamentos de Robótica. 2da Edición. S.A. MCGRAW-HILL / INTERAMERICANA DE ESPAÑA; 2007. Disponible en: <http://www.casadellibro.com/libro-fundamentos-de-robotica-2-ed/9788448156367/1132459>
2. Bolton, W. Mecatrónica. Sistemas de control electrónico en ingeniería mecánica y eléctrica. 2da Edición. Alfaomega; 2011. Disponible en: www.teczamora.blogspot.com
3. Medina J. Simulador 3D para brazo robot de 4 grados de libertad. Revista Iberoamericana para la Investigación y el Desarrollo Educativo. 2014.
4. Aguirre I. Sistema manipulador antropomórfico de tres grados de libertad. ITECKNE. 2011.8(1): 87 – 95.
5. Molina M. Diseño y Construcción del Prototipo de un Brazo Robótico con Tres Grados de Libertad, como Objeto de Estudio. INGENIARE. 2015; 18(1): 87-94.
6. Saleiro M, Carmo B, Rodrigues JM, du Buf JH. A low-cost classroom-oriented educational robotics system. Social Robotics. 2013; pp. 74–83.
7. Valera A, Soriano A, Vallés M. Plataformas de Bajo Coste para la Realización de Trabajos Prácticos de Mecatrónica y Robótica. Revista Iberoamericana de Automática e Informática industrial. 2014; 11(3): 363–376.
8. Busquets J, Busquets JV, Tudela D, Pérez F, Busquets-Carbonell J, Barberá A, et al. Low-cost auv based on arduino open source microcontroller board for oceanographic research applications in a collaborative long term deployment missions and suitable for combining with an usv as autonomous automatic recharging platform. Autonomous Underwater Vehicles (AUV).2012; pp.1–10.
9. Evans, M. Arduino in Action. Manning Publications Co; 2013. Disponible en: <http://www.ifmbe.org>.
10. Torrente, O. Arduino curso práctico de formación. Alfaomega; 2013. Disponible en: <http://www.arduino.cc>.
11. Al-Busaidi AM. Development of an educational environment for online control of a biped robot using matlab and arduino. Mechatronics (MECATRONICS). 2012; 337–344.
12. Rafael CM. Desarrollo de control adaptable y HMI para robot articulado serial. Tesis de maestría. Universidad Autónoma de Querétaro; 2011. Disponible en: <http://www.kansasuni.edu>.
13. Cuellar MP, Pegalajar MC. Design and implementation of intelligent systems with lego mindstorms for undergraduate computer engineers. Computer Applications in Engineering Education. 2014; 22 (1): 153–166.

14. Jiménez-González A, de Dios JRM, de San Bernabé A, Núñez G, Ollero A. Un banco de pruebas remoto para experimentación en robótica ubicua. Revista Iberoamericana de Automática e Informática Industrial RIAI. 2014; 11 (1): 68 – 79.
15. Chaos D, Chacon JnJ, Lopez-Orozco JA, DormidoS. Virtual and remote robotic laboratory using ejs, matlab and labview. Sensors. 2013;13(2): 2595–2612.

AUTORES

Daily Milanés Hermosilla, Ingeniera en Automática, Master en Automática, Universidad de Oriente, Santiago de Cuba, Cuba, daily@uo.edu.cu. Trabaja las líneas de investigación de control basado en modelo y robótica.

Alejandro Castilla Pérez, Ingeniero en Automática, Universidad de Oriente, Santiago de Cuba, Cuba, alejandro.castilla@cedai.com.cu Trabaja actualmente en el CEDAI, provincia de Santiago de Cuba.