

## SIMULADOR DEL ROBOT DELTA EN LABVIEW UTILIZANDO METODOS GEOMETRICOS Y MATLAB

Ignacio Javier Vázquez Cuevas<sup>1</sup>, José Alejandro López Corella<sup>2</sup>, Sergio Iván Hernández Ruiz<sup>3</sup>, Maribel Herrera Velarde<sup>4</sup>, Jorge Alberto Baturoni Encinas<sup>5</sup>, Diana Belinda Hernández Barajas<sup>6</sup>

Tecnológico Nacional de México / I.T. de Nogales, Metal Mecánica Ingeniería Mecatrónica, Eléctrica y Electrónica Ingeniería en Electrónica, Av. Tecnológico 911, Nogales, Sonora, México  
+52(631) 311 1881. Ext 1129.  
[vazquez.ignacio@itnogales.edu.mx](mailto:vazquez.ignacio@itnogales.edu.mx)

### RESUMEN.

En el presente trabajo se demuestra el desarrollo de un simulador del Robot Delta utilizando métodos geométricos, en la cual primero se creó el modelo matemático de la cinemática directa e inversa en Matlab, para posteriormente generar una forma en Windows con la ayuda de la herramienta GUI (graphical user interface), una vez demostrada la matemática se desarrolló el robot en LabView, utilizando el modelo demostrado para crear una simulación del robot en vectores, de esta manera se pudo comparar el funcionamiento correcto del simulador en LabView, en la cual es una interfaz mucho más versátil y amigable para la creación de un prototipo físico del robot como futuro desarrollo ya que la simulación nos da la certeza del funcionamiento de la cinemática como proceso previo para la aplicación física del Robot Delta.

Palabras Clave: Robot Delta, LabView, Matlab.

### ABSTRACT.

In the present work the development of a simulator of the Delta Robot using geometric methods is demonstrated, in which first the mathematical model of the direct and inverse kinematics was created in Matlab, to later generate a form in Windows with the help of the GUI tool (graphical user interface), once the mathematics was demonstrated, the robot was developed in LabView, using the model shown to create a simulation of the robot in vectors, in this way it was possible to compare the correct functioning of the simulator in LabView, in which it is a much more versatile and friendly interface for the creation of a physical prototype of the robot as a future development since the simulation gives us the certainty of the operation of the kinematics as a previous process for the physical application of the Delta Robot.

Keywords: Robot Delta, LabView, Matlab.

## 1. INTRODUCCIÓN

Se han desarrollado algunos simuladores de Robots Delta con diferentes métodos matemáticos para la solución de la cinemática inversa, uno de los métodos para la solución de este modelado matemático es el uso de Redes Neuronales [1], otros de los métodos utilizados son números complejos en la cual se utiliza la rotación usual y variante de los complejos para modelar robots [2], también existe el Álgebra de Cuaterniones

la cual es una herramienta matemática utilizada para la solución de la cinemática directa [4]. Los Cuaterniones se crean Vectores Dinámicos las cuales se definen con las mismas características de movimiento que los eslabones que conforman al robot; por lo que representan una herramienta muy útil para obtener las ecuaciones cinemáticas de cadenas cerradas o abiertas [5].

Este trabajo se centra con el desarrollo de la Cinemática Inversa utilizando métodos geométricos como se demuestra en el artículo [6] que desarrollaron un robot paralelo de tres grados de libertad denominado manipulador delta, también en el artículo [7] demostraron el modelo matemático con métodos geométricos validados en Matlab utilizando Solidworks para su implementación mecánica. El presente artículo toma como base el algoritmo geométrico descrito en [8], en la cual se demuestra y valida creando un simulador en Matlab, con el fin de migrarlo a LabView para futuras aplicaciones en un prototipo.

Hoy en día los Robots Delta han tenido una gran aplicación, ya que pueden manipular objetos pesados, con una velocidad de operación alta, además tienen una mejor precisión y repetitividad, son muy utilizados en procesos en la etapa de empaquetamientos y en industrias alimenticias [8]. Los robots manipuladores se pueden clasificar de acuerdo a su estructura: serie, paralelo e híbridos.

Los Robots Delta también son conocidos como robots de cadena cerrada o paralela, estos están compuestos por dos plataformas paralelas, una base móvil y de menor dimensión que otra fija, tal como se observa en la figura 1. Además, están unidas por medio de varios brazos paralelos, cada brazo está controlado por un servomotor, estos generan un movimiento.

El robot Delta es considerado como uno de los robots paralelos que más éxito han tenido en cuanto a su diseño. Varios centenares de este tipo de robot se encuentran trabajando en la industria actualmente [7].

La idea básica detrás del diseño del robot paralelo Delta es el uso de paralelogramos. Un paralelogramo ofrece a un eslabón rendimiento para permanecer en una orientación fija con respecto a un eslabón de entrada. El uso de tales

paralelogramos ubica la orientación de la plataforma móvil que sólo permanece con los tres grados de libertad de translación. Las juntas de revoluta de las palancas de rotación están actuando de dos maneras diferentes: con motores rotatorios o con actuadores lineales. Finalmente, un cuarto dispositivo se usa para transmitir movimiento rotatorio de la base a un extremo final, montado en la plataforma móvil. [7]

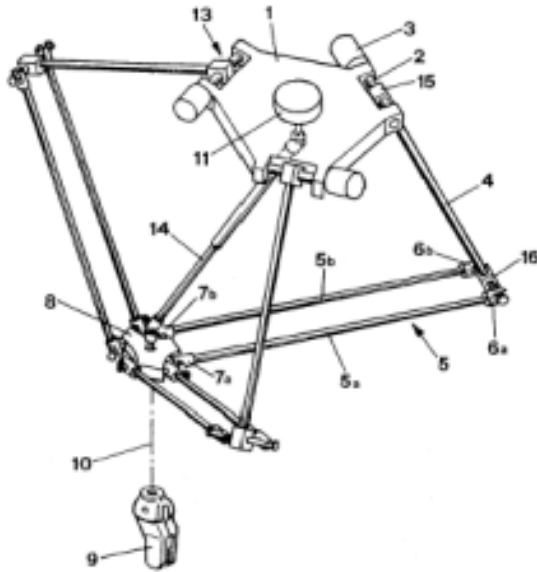


Figura 1. Partes del Robot Delta.

El dispositivo comprende un elemento base (1) y un elemento móvil (8). Tres brazos de control (4) están rígidamente montados en sus extremos (15) con tres flechas (2), las cuales pueden rotar. Los tres ensambles, cada uno formado por una flecha y un brazo, son las partes móviles de tres actuadores (13) en los cuales las partes fijas (3) están en conjunto con el elemento base. Los otros extremos (16) de cada brazo de control están en conjunto con el elemento móvil por medio de dos eslabones (5a, 5b), donde en uno de sus extremos están montados con una articulación de cada uno de los brazos de control y del otro extremo están montados en el elemento móvil. La inclinación y orientación en el espacio del elemento móvil permanecen inertes hasta que los brazos de control se pongan en movimiento. El elemento móvil soporta un elemento de trabajo (9), donde la rotación o movimiento es controlada por un actuador (11) montado en el elemento base. Un brazo telescópico (14) conecta el actuador al elemento de trabajo (ver figura 1) [2].

## 2. DESARROLLO

### 2.1. Cinemática Inversa.

En la figura 2 muestra el esquema simplificado de un robot tipo Delta que consiste de dos plataformas; la fija, en donde se ubican los actuadores que para este tipo de robot son motores rotativos, colocados a la misma distancia del centro O, en los puntos 1, 2 y 3 y la plataforma móvil que porta el efector del robot, ubicado en forma adecuada en el centro del triángulo.

Las dos plataformas están unidas entre sí, por tres estructuras de barras idénticas conformadas por un brazo en cada uno de los motores y por antebrazo que está unido al brazo y a la plataforma móvil, por medio de juntas esféricas [8].

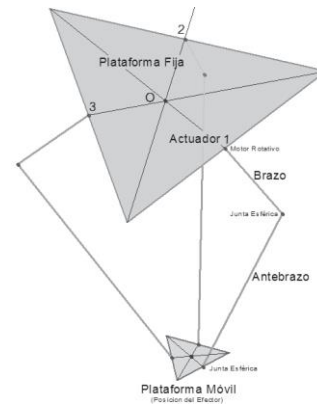


Figura 2. Esquema Robot Delta.

El problema cinemático en un robot, es encontrar la relación de la posición del efector final y los ángulos de articulación de los actuadores, en el caso del robot delta, el análisis cinemático inverso busca encontrar la relación entre la posición de la plataforma móvil, especialmente el punto  $p[x_0 \ y_0 \ z_0]$ , y los ángulos de los brazos  $\theta_1$ ,  $\theta_2$  y  $\theta_3$  donde están colocados los motores como se muestra en la figura 3. El brazo tiene una longitud  $L_1$  y el antebrazo  $L_2$ . Se toman los puntos A, B y C como referencia para el análisis geométrico.

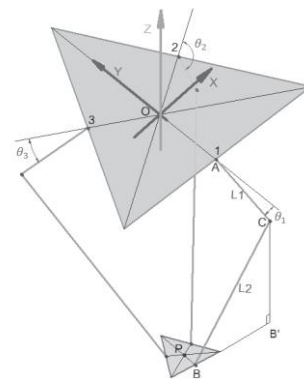


Figura 3. Geometría Robot Delta.

El objetivo de la cinemática inversa es encontrar el ángulo de cada actuador, conociendo la posición del efector final, este problema se produce cuando se tienen las coordenadas de un objeto que se quiere manipular, y se desea saber el ángulo que debe suministrar el sistema de control a cada motor.

Debido a la restricción de la junta A, el brazo describe una circunferencia de radio  $L1$ , mientras que con respecto del punto B, el punto antebrazo puede describir una esfera de Radio  $L2$ . La intersección de la circunferencia y la esfera se produce en dos puntos, se toma como solución el punto con menor valor en la coordenada y. Al determinar la posición del punto C, se puede obtener el ángulo  $\theta_1$  del actuador, como se muestra en la figura 4.

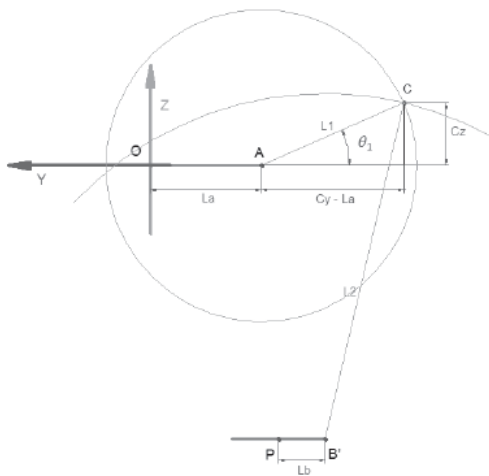


Figura 4. Cadena cinemática de un brazo del Robot Delta.

Coordenadas del Punto P, B, A y B'

$$P(x_0, y_0, z_0); B(x_0, y_0 - L_b, z_0); A(0, -L_a, 0); B'(0, y_0 - L_b, z_0)$$

Con las coordenadas de los puntos descritos, se plantea un sistema de dos ecuaciones no lineales que permita encontrar la posición del punto C, con la cual se puede calcular el ángulo que forma el brazo con el plano horizontal, y se obtiene así, la solución esperada como se puede observar en las siguientes ecuaciones

$$(C_y - L_a)^2 + (C_z - A_z)^2 = L_1^2 \dots \dots \text{Ec1}$$

$$(C_y - B'_y)^2 + (C_z - B'_z)^2 = L_2^2 - x_0^2 \dots \dots \text{Ec2}$$

$$\theta_1 = \arctan\left(\frac{C_{z1}}{L_a - C_{y1}}\right) \dots \dots \text{Ec3}$$

Para los otros brazos, se usa la matriz de rotación con un ángulo de  $120^\circ$  para el brazo 2 y  $240^\circ$  para el 3. Esta matriz de rotación permite girar el sistema de coordenadas, de manera que se pueda usar la solución descrita para el cálculo de restantes ángulos, esta solución se validó con una aplicación en Matlab.

## 2.2. Validación Matlab.

Se desarrolló el modelado de la cinemática inversa del Robot aplicando el algoritmo geométrico anteriormente explicado, primero se crearon las funciones en Matlab, para posteriormente utilizarlas en una aplicación GUI siendo una interfaz gráfica con el usuario.

Como se describe en el algoritmo se requiere de una matriz de rotación en Z, con el objetivo de rotar el punto a resolver para los otros dos brazos que están a  $120^\circ$  y  $240^\circ$  respecto al primer brazo, esta función también se utiliza para rotar las cadenas cinemáticas del segundo y tercer brazo para posteriormente graficarlos. En la figura 5(a) y (b) se puede observar la matriz de rotación en Z.

$$HR_z(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Figura 5(a). Matriz Homogénea de Rotación en Z.

Función de rotación en Z

```
function R = Rz(theta)
```

```
R=[cos(theta), sin(theta), 0, 0; sin(theta), cos(theta), 0, 0; 0, 0, 1, 0; 0, 0, 0, 1];
```

```
end
```

Figura 5(b). Matriz Homogénea de Rotación en Z en Matlab.

También se creó una función que forma la cadena cinemática del brazo, esta función también regresa la articulación del servomotor, cabe mencionar que de los parámetros de entrada de esta función es la solución del sistema de ecuaciones antes mencionada, así como las dimensiones de los brazos y el punto que se desea que articule el robot, a continuación, se puede observar la función creada en Matlab como se muestra en la figura 5(c).

```
Function [matriz, th] =
delta_cadena(s1, s2, l11, l12, l1a, l1b, p)
syms cy la cz az l1 lb bz l2 x y z
o=[0;0;0;0];
b=[p(1,1); p(2,1)-l1b; p(3,1); 0];
a=[0; -l1a; 0; 0];
bb=[0; p(2,1)-l1b; p(3,1); 0];
```

```
rescy=subs(s1,[la lb l1 l2 x y z],[l1a
l1b l11 l12 p(1,1) p(2,1) p(3,1)]);
rescz=subs(s2,[la lb l1 l2 x y z],[l1a
l1b l11 l12 p(1,1) p(2,1) p(3,1)]);

c=[0;rescy;rescz;0]
matriz=[o,a,c,b,p];
th=(atan(rescz/rescy)*180/pi);
end
```

Figura 5(c). Formación de la cadena cinemática en Matlab.

Con la ayuda de estas funciones nos facilita la aplicación del algoritmo geométrico para la cinemática inversa del robot, en la figura 6 se puede observar un ejemplo de la articulación del robot en la coordenada (0,0,-5)

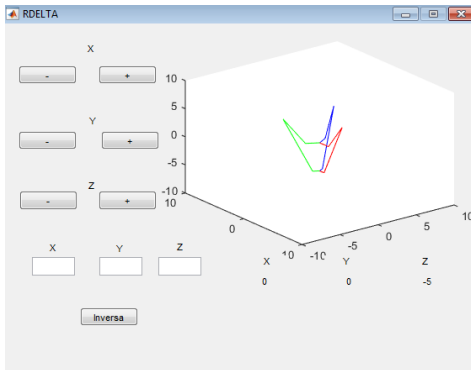


Figura 6. Demostración del algoritmo geométrico.

La aplicación del algoritmo consiste en los siguientes pasos:

- 1.- Resolver el sistema de ecuaciones utilizando la función *Solve* y guardarlas en *s1* y *s2*.
- 2.- Asignar los valores de los eslabones así como de los platos del Robot Delta (*la* y *lb*)
- 3.- Asignar el punto *p* que se desea que articule el Robot
- 4.- utilizar la función *delta\_cadena*, para resolver la primera cadena cinemática (obtener *m1* y *t1*).
- 5.- Rotar el punto *p* a  $-120^\circ$ , utilizar la función *delta\_cadena*, para resolver la primera cadena cinemática (obtener *m2* y *t2*).
- 6.- Rotar la cadena cinemático *m2* a  $-120^\circ$
- 7.- Rotar el punto *p* a  $-240^\circ$ , utilizar la función *delta\_cadena*, para resolver la primera cadena cinemática (obtener *m3* y *t3*).
- 8.- Rotar la cadena cinemático *m3* a  $-240^\circ$
- 9.- Graficar las codenas cinemáticas con *plot3*

En la figura 6 (a), se muestra el código del algoritmo mencionado anteriormente.

```
syms cy la cz az l1 lb bz l2 x y z
f1='(cy-la)^2+cz^2=l1^2';
```

```
f2='(cy-(y-lb))^2+(cz-z)^2=l2^2-x^2';
[s1,s2]=solve(f1,f2,'cy','cz');
s1=simplify(s1);
s2=simplify(s2);
s1=s1(1,1);
s2=s2(1,1);

l11=7;
l12=10;
l1a=1.5;
l1b=.75;

p=[0;0;-5;0];

[m1,t1]=delta_cadena(s1,s2,l11,l12,l1a,l1b,p);

p2=Rz(-120*pi/180)*p;
[m2,t2]=delta_cadena(s1,s2,l11,l12,l1a,l1b,p2);
m2=Rz(120*pi/180)*m2;

p3=Rz(-240*pi/180)*p;
[m3,t3]=delta_cadena(s1,s2,l11,l12,l1a,l1b,p3);
m3=Rz(240*pi/180)*m3;

plot3(m1(1,:),m1(2,:),m1(3:),'r')
axis([-10 10 -10 10 -10 10])
hold on
plot3(m2(1,:),m2(2,:),m2(3:),'b')
plot3(m3(1,:),m3(2,:),m3(3:),'g')
hold off
```

Figura 6(a). Demostración del algoritmo geométrico en Matlab.

Con la aplicación de este programa en Matlab se puede comprobar la matemática y el algoritmo planteado para el desarrollo del simulador del robot delta, a través de este simulador se toma como referencia para posteriormente aplicarlo en Labview.

### 2.3. Simulador en Labview.

Se desarrollaron las SubVI, para crear el simulador, en la figura 7 se puede observar la *SubVI Rz* la cual es la matriz de rotación en z, esta función recibe como parámetro el punto y el ángulo para rotarlo, como se explica anteriormente esta función es utilizada para rotar los puntos del segundo y tercer brazo del Robot Delta.



Figura 7.- matriz de rotación en Z.

En la figura 8 se muestra la matemática desarrollada en Labview utilizando la estructura de *Formula Node*.

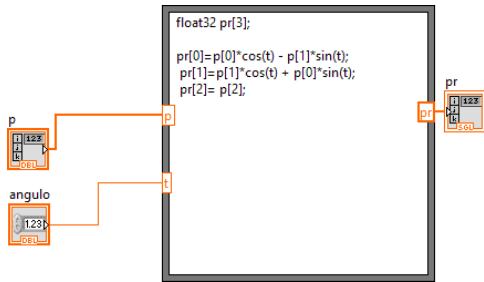


Figura 8. Matemática de Rz.

También se creó la *SubVI Cadena Delta*, esta función recibe como parámetros los tamaños de los eslabones, así como el punto donde se desea que articule el robot resolviendo la cinemática inversa, se utilizó la solución del sistema de ecuaciones resuelto en Matlab; Esta función da como resultado los puntos de articulación de la cadena cinemática, así como el ángulo del actuador, la función se puede observar en la figura 9.

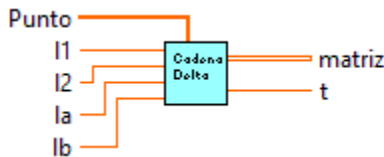


Figura 9. SubVI Cadena delta.

Como se puede observar en la figura 10, se crearon los puntos para formar la cadena cinemática del Robot Delta, en esta estructura se crea una matriz donde se indican los puntos para formar la cadena cinemática vista en la figura 4, los puntos más complicados son Cy y Cz, estos puntos se encuentran con la solución de sistema de ecuaciones, utilizando el comando *solve* en Matlab.

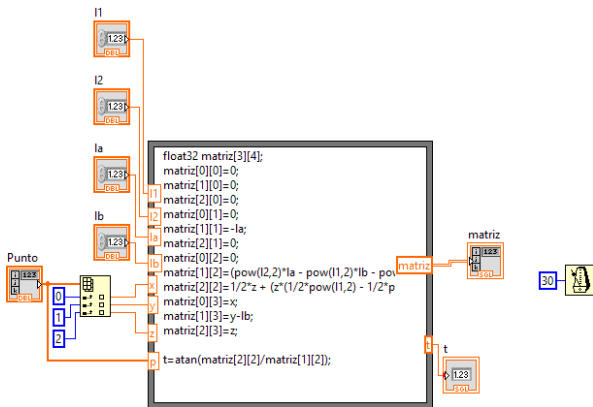


Figura 10. SubVI Cadena delta.

En la figura 11 se observa la SubVI *Delta* utiliza internamente las funciones anteriormente mencionadas, esta función recibe como parámetros la posición que deseamos que articule el robot resolviendo la cinemática inversa, así como un arreglo donde se almacena los tamaños de los eslabones, esta función da como resultado las articulaciones de los tres actuadores y la gráfica del robot como se puede observar en la figura 12.

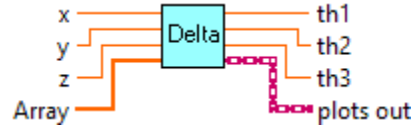


Figura 11. SubVI Delta.

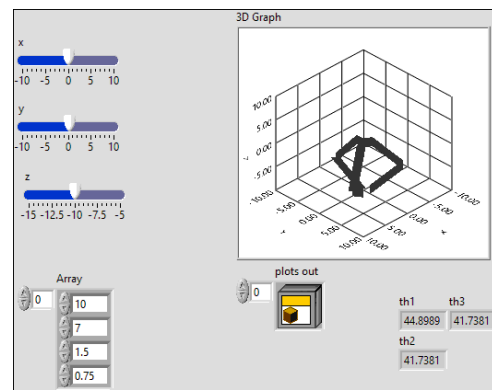


Figura 12. Gráfica del Robot Delta.

En la figura 13 muestra el programa siguiendo el algoritmo geométrico, en la cual resuelve las tres cadenas cinemáticas con sus respectivas rotaciones para graficar el Robot.

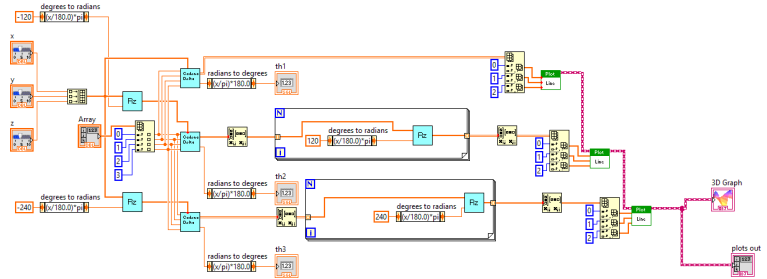


Figura 13. Programa del Robot Delta.

### 3. RESULTADOS

Se realizó un análisis del simulador del Robot Delta, dando distintos puntos para observar la articulación, en la cual se puede determinar la eficiencia del simulador. En la figura 14 y 15 muestra la interfaz del usuario del simulador utilizando los



controladores e indicadores de LabView, el indicador más importante del simulador es la gráfica en vectores, este simulador tiene como indicadores la posición actual del Robot y tiene como controladores las barras para mover el robot en xyz así como un botón para mover el robot a una coordenada definida

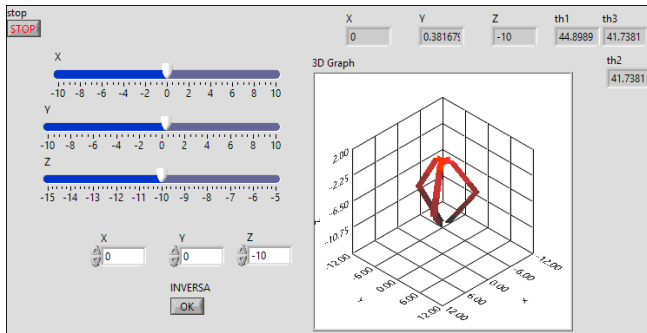


Figura 14. Interfaz del simulador en Labview.

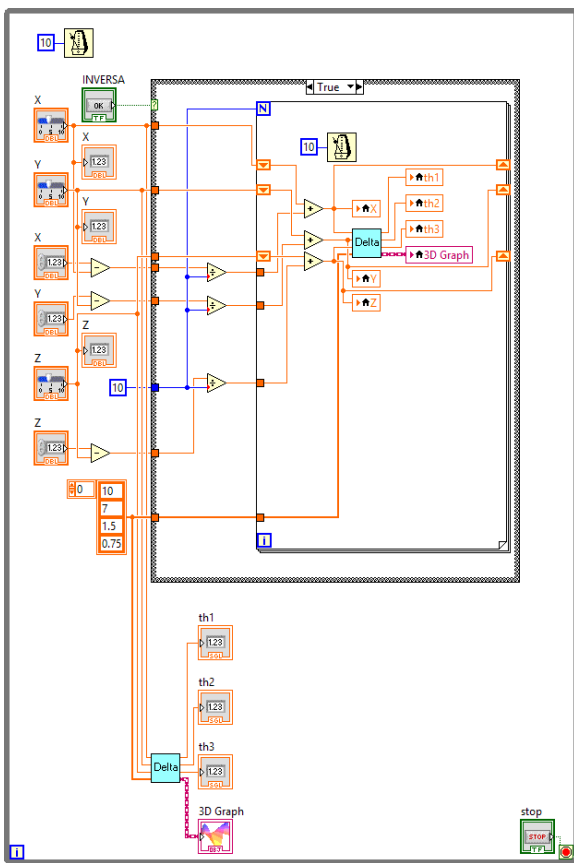


Figura 15. Programa principal en Labview.

#### 4. CONCLUSIONES

Para el desarrollo de simuladores de robots se requiere de la aplicación de modelos matemáticos, así como de la aplicación de Softwares, en este caso se utilizó Matlab para validar la matemática y el algoritmo geométrico, para posteriormente aplicarlo en LabView. Cabe destacar que al utilizarse métodos meramente geométricos y trigonométricos, como fue en este caso, la programación no fue tan robusta; sin embargo, es importante saber que existen también otros métodos, como lo son los métodos numéricos, inteligencia artificial así como algebra de cuaterniones, y que análogamente conllevan a los mismos resultados al buscar solución virtual a través del software de análisis experimental LabView, el cual permitió manipular al Robot Delta de manera directa en cada una de sus articulaciones, resolviendo la trayectoria trigonométricamente, como se mencionó anteriormente.

#### 5. REFERENCIAS

- [1] M. Dehghani, «Neural Network Solution for Forward Kinematics Problem of HEXA Parallel Robot,» American Control Conference, p. 6, 2008.
- [2] P. A. O. ESCOTO, MODELACIÓN Y SIMULACIÓN CINEMÁTICA DE UN ROBOT DELTA PLANAR TIPO RR, Ciudad Universitaria, México, D. F.: UNAM, 2006.
- [3] Ignacio Vázquez Cuevas, «Modelación y Diseño de un Simulador de un Robot Paralelo Manejado por un Controlador Manual Didáctico,» Congreso Nacional de Mecatrónica, vol. 8, pp. 329-324, 2009.
- [4] F. C. JIMÉNEZ, TESIS: ANÁLISIS CINEMÁTICO Y DINÁMICO DE UN ROBOT PARALELO ESPACIAL RUS, Departamento de Posgrado en Ingeniería Mecánica de la Facultad de Ingeniería: UNAM, 2008.
- [5] M. J. X. Atonatiúh, «Diseño y Control de un Robot Paralelo,» Asociación Mexicana de Mecatrónica A.C., vol. 6, pp. 219-225, 2007.
- [6] W. Pachacama, «Diseño y construcción de un prototipo de robot delta para aplicaciones pick & place,» Jornadas en Ingeniería Eléctrica y Electrónica, vol. 27, pp. 100-107, 2017.
- [7] J. GUDIÑO-LAU, «Diseño de un robot delta y su análisis cinemático,» Revista de Aplicación Científica y Técnica, vol. 3, n° 8, pp. 13-23, 2017.
- [8] C. A. P. Cortés, «OPTIMIZACIÓN DIMENSIONAL DE UN ROBOT PARALELO TIPO DELTA BASADO EN EL MENOR CONSUMO DE ENERGÍA,» Ciencia e Ingeniería Neogranadina, vol. 21, n° 1, pp. 77-88, 2011.