

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ**  
**FACULTAD DE CIENCIAS E INGENIERÍA**



PONTIFICIA  
**UNIVERSIDAD**  
**CATÓLICA**  
DEL PERÚ

IMPLEMENTACIÓN DE UNA ESTACIÓN DE TRABAJO  
MEDIANTE UN ROBOT SERIAL DE 3 GRADOS DE LIBERTAD  
PARA EL CETAM

Tesis para optar el Título de Ingeniero Electrónico, que presenta el bachiller:

**Oscar Benjamín Cieza Aguirre**

ASESOR: Julio Cesar Tafur Sotelo

**Lima, Agosto del 2011**

## RESUMEN

El presente trabajo de tesis tiene como objetivo diseñar e implementar las interfaces hardware y software, el algoritmo de control PID Desacoplado, así como la interface de usuario que permita comandar los movimientos punto a punto, línea recta y circunferencia de un robot serial planar de 3 grados de libertad, con accionamiento eléctrico construido por el CETAM.

En un principio se describen algunas de las tecnologías de sistemas de control más comunes implementadas en la actualidad en la robótica de manipuladores seriales, para luego pasar a explicar el modelo teórico del funcionamiento de todo el sistema propuesto.

Posteriormente, el presente trabajo realiza un énfasis en el desarrollo del algoritmo de generación de trayectoria, la cinemática directa e inversa y la dinámica directa e inversa para las simulaciones e implementación del sistema.

Luego se describe la Interface Usuario a utilizar, su modo de funcionamiento así como el protocolo de comunicación entre el controlador y la computadora.

Finalmente se desarrolla el algoritmo de control PI Desacoplado de corriente sintonizándolo de forma analítica así como el control PID Desacoplado de posición sintonizándolo de forma experimental comparando los resultados con la sintonización hecha por software propio de los controladores EPOS UserInterface.

## INDICE

<b>INTRODUCCIÓN .....</b>	<b>10</b>
<b>CAPÍTULO 1.- DESCRIPCIÓN DEL PROBLEMA .....</b>	<b>11</b>
1.1 Descripción de la problemática materia del estudio .....	11
1.1.1 Teórico Académica .....	11
1.1.2 Realidad .....	11
1.2 Declaración del Marco problemático .....	12
<b>CAPÍTULO 2.- MARCO TEÓRICO .....</b>	<b>14</b>
2.1 Estado del Arte .....	14
2.1.1 Presentación del asunto de estudio .....	14
2.1.2 Estado de la investigación .....	15
2.1.3 Síntesis sobre el asunto de estudio .....	18
2.2 Conceptualizaciones Generales .....	19
2.3 Modelo Teórico .....	21
2.4 Descripción mínima de aspectos teóricos materia de estudio .....	24
2.4.1 Coordenadas Homogéneas .....	24
2.4.2 Modelos Cinemáticos .....	24
2.4.3 Modelos Dinámicos .....	25
2.4.4 Controlador PID Desacoplado .....	26
2.5 Descripción de materia del estudio vigente .....	27
2.5.1 Generación de trayectoria .....	27
2.5.2 Sintonización del controlador PID-Reglas Ziegler y Nichols .....	28
2.5.3 Sintonización del controlador PI de corriente de forma Analítica .....	28
<b>CAPÍTULO 3.- DESARROLLO DE LA TESIS .....</b>	<b>29</b>
3.1 Objetivos .....	29
3.1.1 Objetivo General .....	29
3.1.2 Objetivos Específicos .....	29
3.2 Descripción y justificación del diseño .....	29
3.2.1 Modelos Cinemáticos .....	29

3.2.2	Modelo Dinámico.....	31
3.2.3	Algoritmo generación de trayectoria .....	33
3.2.4	Control PI de corriente y PID de posición del brazo robot.....	36
3.3	Selección y justificación, componentes y/o equipos.....	42
3.3.1	Descripción del sistema Mecánico.....	42
3.3.2	Cálculo de los Accionamientos Eléctricos.....	43
3.3.3	Otros .....	47
3.4	Implementación y construcción del control de posición e Interface del Robot de 3 grados de Libertad Construido por el CETAM .....	48
3.4.1	Interface de Usuario .....	48
3.4.2	Comunicación EPOS24/1 .....	50
<b>CAPÍTULO 4.-</b>	<b>.....</b>	<b>52</b>
4.1	Ensayos y Pruebas.....	52
4.2	Resultados. ....	53
4.2.1	PI Calculado .....	53
4.2.2	PI Calculado Vs PI Hallado por el software.....	54
4.2.3	PID Calculado Vs PID Hallado por el software.....	54
4.2.4	Retardo de Comunicación PC – EPOS24/1 .....	54
4.3	Propuesta económica.....	55
<b>CONCLUSIONES.....</b>	<b>.....</b>	<b>56</b>
<b>OBSERVACIONES .....</b>	<b>.....</b>	<b>56</b>
<b>RECOMENDACIONES.....</b>	<b>.....</b>	<b>57</b>
<b>BIBLIOGRAFÍA .....</b>	<b>.....</b>	<b>58</b>
<b>APENDICE .....</b>	<b>.....</b>	<b>60</b>

## Lista Figuras

Figura: 1	Diagrama de Flujo del desarrollo de la estación de trabajo del Robot.....	12
Figura: 2	Diagrama de bloques del sistema de control del EPOS 24/1 .....	14
Figura: 3	Esquema del algoritmo de control PD con compensación .....	16
Figura: 4	Esquema del algoritmo de control par calculado.....	16
Figura: 5	Trayectoria de referencia de la estrategia de un control predictivo .....	18
Figura: 6	Diagrama de conexión de los controladores. ....	22
Figura: 7	Funcionamiento global del sistema.....	23
Figura: 8	Brazo robot con Coordenadas Cartesiana y articulares.....	24
Figura: 9	Cinemática Directa e Inversa.....	24
Figura: 10	Diagrama de Bloques del controlador PID.....	26
Figura: 11	Perfil de trayectoria de un punto a otro .....	27
Figura: 12	Oscilaciones sostenidas en respuesta del sistema al $K_{cr}$ .....	28
Figura: 13	Modelo dinámico del robot con los actuadores. ....	33
Figura: 14	Perfil de posición .....	35
Figura: 15	Perfil de velocidad .....	35
Figura: 16	Perfil de Aceleración.....	35
Figura: 17	Movimiento en Línea Recta .....	36
Figura: 18	Movimiento en Circunferencia .....	36
Figura: 19	Modelo del sistema desacoplado de los motores con sus cargas respectivas (Planta).....	37
Figura: 20	Control de Corriente Desacoplado de los motores.....	39
Figura: 21	Respuesta del motor 3 en lazo cerrado sin modelamiento de retardo con un periodo de muestreo de 10-5 segundos.....	39

Figura: 22	Respuesta del motor 2 en lazo cerrado sin modelamiento de retardo con un periodo de muestreo de $10^{-5}$ segundos.....	39
Figura: 23	Respuesta del motor 3 en lazo cerrado sin modelamiento de retardo con un periodo de muestreo de $10^{-4}$ segundos.....	40
Figura: 24	Respuesta del motor 2 en lazo cerrado sin modelamiento de retardo con un periodo de muestreo de $10^{-4}$ segundos.....	40
Figura: 25	Respuesta del motor 3 en lazo cerrado con modelamiento de retardo con un periodo de muestreo de $10^{-4}$ segundos.....	40
Figura: 26	Respuesta del motor 2 en lazo cerrado con modelamiento de retardo con un periodo de muestreo de $10^{-4}$ segundos.....	40
Figura: 27	Oscilaciones sostenidas del motor 1 producidas ante una respuesta escalón y un $K_{critico}$ .....	41
Figura: 28	Oscilaciones sostenidas del motor 2 producidas ante una respuesta escalón y un $K_{critico}$ .....	41
Figura: 29	Oscilaciones sostenidas del motor 3 producidas ante una respuesta escalón y un $K_{critico}$ .....	41
Figura: 30	Brazo robot de 3 Grados de Libertad.....	42
Figura: 31	Cinemática Inversa y Dinámica Inversa para el cálculo del torque necesario.....	43
Figura: 32	Trayectoria en forma de circunferencia en el Espacio X,Y.....	44
Figura: 33	Trayectoria en línea recta en el Espacio X,Y.....	44
Figura: 34	Torque necesario para la trayectoria en forma de circunferencia... .....	44
Figura: 35	Torque necesario para la trayectoria en línea recta.....	44
Figura: 36	Potencia necesaria para la trayectoria en forma de circunferencia. .....	44
Figura: 37	Potencia necesaria para la trayectoria en línea recta.....	44
Figura: 38	Velocidad Angular necesaria para la trayectoria en forma de circunferencia.....	45
Figura: 39	Velocidad Angular necesaria para la trayectoria en línea recta...45	

Figura: 40	Controlador Epos.....	46
Figura: 41	Diagrama de conexiones físicas del controlador Epos 24/1 .....	47
Figura: 42	Pantalla de la Interface usuario .....	48
Figura: 43	Menú pop-up de elección del tipo de movimiento .....	49
Figura: 44	Tipos de curvas .....	49
Figura: 45	Comunicación PC -> Epos24/1 .....	50
Figura: 46	Comunicación Epos24/1 -> PC.....	50
Figura: 47	Curva de respuesta del motor 1 con sintonización calculada.....	52
Figura: 48	Curva de respuesta del motor 1 con sintonización por software EPOS UserInterface .....	52
Figura: 49	Curva de respuesta del motor 2 con sintonización calculada.....	52
Figura: 50	Curva de respuesta del motor 2 con sintonización por software EPOS UserInterface .....	52
Figura: 51	Curva de respuesta del motor 3 con sintonización calculada.....	52
Figura: 52	Curva de respuesta del motor 3 con sintonización por software EPOS UserInterface .....	52
Figura: 53	Curva de respuesta del motor 1 con sintonización calculada.....	53
Figura: 54	Curva de respuesta del motor 1 con sintonización por software EPOS UserInterface .....	53
Figura: 55	Curva de respuesta del motor 2 con sintonización calculada.....	53
Figura: 56	Curva de respuesta del motor 2 con sintonización por software EPOS UserInterface .....	53
Figura: 57	Curva de respuesta del motor 3 con sintonización calculada.....	53
Figura: 58	Curva de respuesta del motor 3 con sintonización por software EPOS UserInterface .....	53

## Lista de Tablas

Tabla: 1	Conceptos Generales.....	19
Tabla: 2	Acciones proporcional, integral y derivativa en un algoritmo de control PID.....	26
Tabla: 3	Calculo de las ganancias del segundo método de las Reglas de Ziegler y Nichols. ....	28
Tabla: 4	Relaciones entre $\theta, \dot{\theta}, \ddot{\theta}$ y $\phi, \dot{\phi}, \ddot{\phi}$ .....	30
Tabla: 5	Polos del Planta.....	38
Tabla: 6	Ganancias del control de Corriente calculadas con el periodo cercano al ideal (Sin modelamiento de retardo).....	38
Tabla: 7	Ganancias del control de Corriente calculadas con el periodo real y modelamiento de retardo .....	39
Tabla: 8	Respuestas del sistema de control de corriente PI de calculado analíticamente .....	39
Tabla: 9	Oscilaciones sostenidas para hallar las ganancias del control de posición PID. ....	41
Tabla: 10	Ganancias del control de posición halladas de forma experimental .....	41
Tabla: 11	Características del Brazo Robot Construido por el CETAM .....	42
Tabla: 12	Cálculo de Torques, Potencias y Velocidades Angulares para los movimientos requeridos.....	44
Tabla: 13	Características de los motores .....	45
Tabla: 14	Requisitos máximos en los motores .....	45
Tabla: 15	Fuentes tentativas .....	47
Tabla: 16	Características del Adaptador USB-RS232 .....	48
Tabla: 17	Respuestas del control de corriente.....	52
Tabla: 18	Respuestas del control de posición. ....	53
Tabla: 19	Costos del proyecto de la Estación de trabajo del robot de 3 grados de Libertad.....	55



## INTRODUCCIÓN

En el actual mundo globalizado, la robótica está cambiando la forma en la que construimos, mantenemos la seguridad, salud, producimos; y distribuimos energía y alimentos al mundo. Es en este contexto en donde los robots manipuladores se han convertido en excelentes herramientas para producir, trabajar y realizar labores peligrosas en la Tierra y fuera de ella.

Dado que Actualmente el Centro de Tecnologías Avanzadas de Manufactura (CETAM) de la Pontificia Universidad Católica del Perú (PUCP) ha construido un brazo robot manipulador de 3 grados de libertad, con el objetivo de servir de soporte educacional así como de desarrollo e investigación, siendo una de sus primeras aplicaciones la teleoperación robótica en donde ejecutará tareas de control de posición; entonces es necesario el desarrollo de un sistema de control del brazo robot.

El presente trabajo tiene como objetivo implementar una interface Hombre-máquina para realizar el control de movimiento del robot manipulador de 3 grados de libertad construido por el CETAM, para el cual se propone un sistema control PID desacoplado. Dicha interface es implementada mediante una computadora personal utilizando el software Matlab – Simulink, mientras que el control PID desacoplado se implementa en los controladores Epos 24/1.

## CAPÍTULO 1.- DESCRIPCIÓN DEL PROBLEMA

### 1.1 Descripción de la problemática materia del estudio

#### 1.1.1 Teórico Académica

El desarrollo de robots manipuladores ha tenido un rápido y significativo avance debido a las necesidades del mundo actual, es en este contexto que se han mejorado los mecanismos y sistemas de control bajo los cuales funcionan, llevando a progresos en la eficiencia, precisión, rapidez, confiabilidad y capacidad de los robots. Es así que el control de robots manipuladores tiene muchas más posibilidades de continuar desarrollándose, además los beneficios que trae consigo permiten el progreso de diversas áreas para las cuales puede ser aplicada la robótica como es el caso de la medicina, la industria, la exploración y situaciones riesgosas para el ser humano.

#### 1.1.2 Realidad

Actualmente el Perú es un país en proceso de industrialización, donde la robótica industrial está ingresando paulatinamente, tomando un gran auge en los últimos años. Es en esta situación donde las industrias peruanas para ser competitivas internacionalmente se ven obligadas en la importación de estos robots. Por tales razones, el uso de tecnologías modernas y económicas resulta conveniente, ya que deberían ser desarrolladas, enseñadas y aplicadas en nuestro país. En ese contexto, el Centro de Tecnologías Avanzadas de Manufactura (CETAM) de la PUCP ha fabricado un brazo manipulador de tres grados de libertad, con el objetivo de servir de soporte educacional así como de desarrollo e investigación, siendo una de sus primeras aplicaciones la teleoperación robótica, además de otras aplicaciones de alcance de posición.

Dado que Actualmente el Centro de Tecnologías Avanzadas de Manufactura (CETAM) de la Pontificia Universidad Católica del Perú (PUCP) ha construido un brazo robot manipulador de 3 grados de libertad; entonces surge la motivación y necesidad de diseñar e implementar el control de dicho brazo a través de una Interface Hombre-Máquina implementada en una computadora personal utilizando el software Matlab, además de una estrategia control PID Desacoplado, para el control de los motores del Robot.

## 1.2 Declaración del Marco problemático

A continuación se muestra el diagrama de flujo del proceso de desarrollo de la Estación de Trabajo para Robot de 3 Grados de Libertad construido por el CETAM en la figura 1.

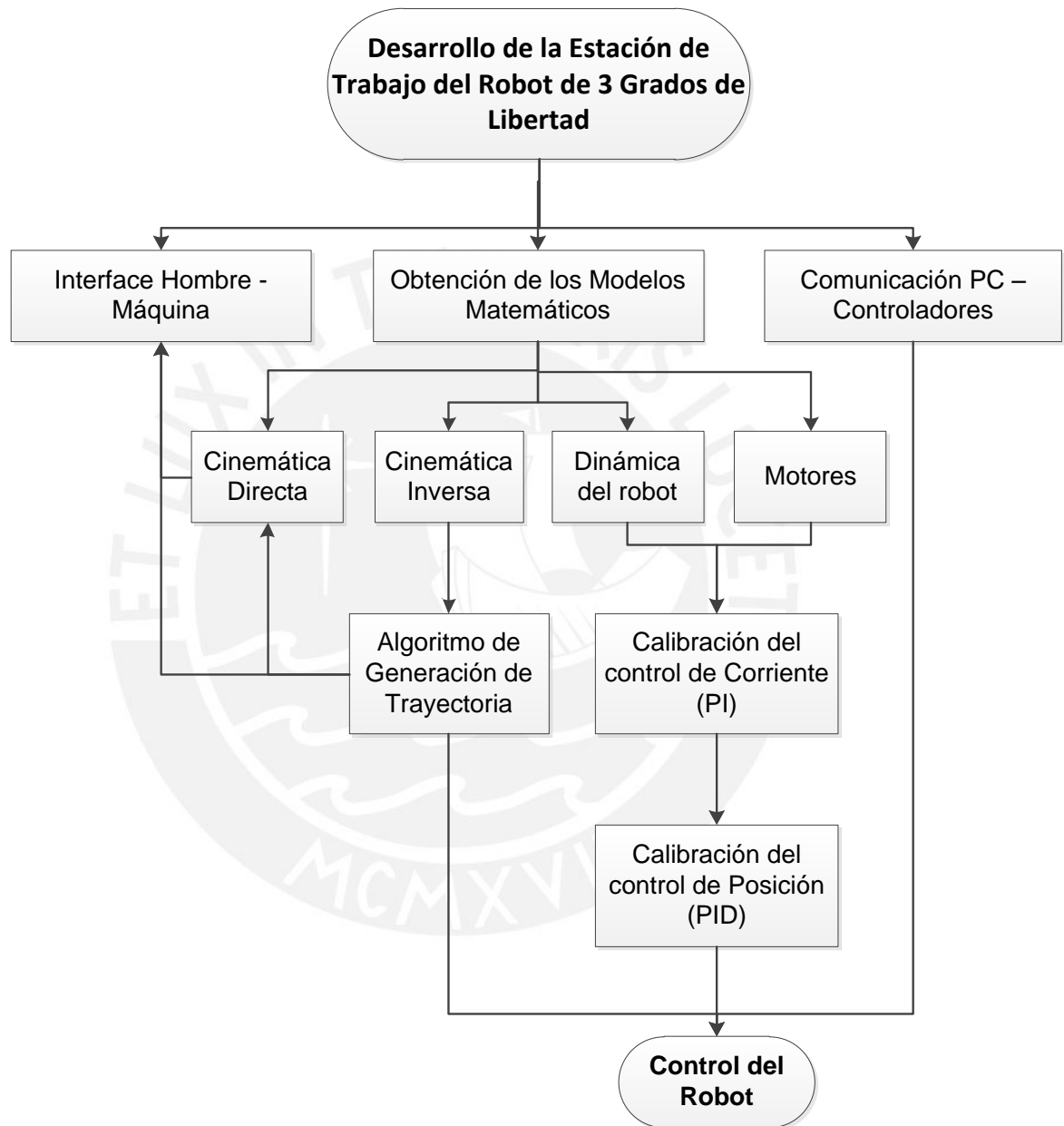


Figura: 1 Diagrama de Flujo del desarrollo de la estación de trabajo del Robot

Para la implementación del control del Robot manipular es necesario en primer lugar, la Obtención de los Modelos Matemáticos del Robot, dentro de los cuales se encuentran: Los modelos Cinemáticos, los cuales relacionan las posiciones en el espacio cartesiano con las posiciones en el espacio de coordenadas articulares de

los eslabones; y los modelos dinámicos, los cuales relacionan los torques en los ejes de los eslabones con las posiciones, velocidad y aceleraciones angulares en coordenadas articulares de los eslabones.

Para la obtención del modelo Dinámico del Robot se utiliza la representación Denavit-Hartenberg [1], para describir el desplazamiento espacial entre los sistemas de coordenadas de los eslabones vecinos obteniendo la información cinemática del robot; la Formulación Euler – Lagrange (escogida principalmente por su forma estructurada) para la deducción de las ecuaciones que representan la dinámica del robot; el modelamiento de los motores brushless [19]; y adicionalmente la relación de transmisión de movimiento producida por el sistema de fajas, debida a que robot presenta los motores en la base y no en los eslabones como tradicionalmente se fabrican los robots seriales.

El robot manipulador que se desea controlar tiene la característica de poseer un comportamiento dinámico altamente no-lineal, multivariable, acoplado y de parámetros variantes, sin embargo, debido a las reducidas velocidades angulares de los eslabones (producto de las elevadas relaciones de reducción); y al libre desplazamiento del robot producido por movimientos que no interactúan con el medio ambiente (carga de objetos o fuerzas externas al robot que alteran de forma significativa el modelo dinámico del robot), los efectos negativos de un control desacoplado (Vibraciones innecesarias, velocidad de respuesta y amortiguamiento reducido, limitaciones en precisión y velocidad del efector final) se reducen significativamente. Por tales motivos además del hardware disponible para el control, se torna factible el diseño y sintonización los parámetros un algoritmo de control PID Desacoplado en los controladores EPOS 24/1.

Adicionalmente el hardware de control (Epos 24/1) contiene un lazo interno PI de corriente (control en cascada) como se muestra en la figura 2, el cual mejora las prestaciones del control (ya que durante el movimiento fuerzas y torques deben ser controlados), además de ofrecer protección de corriente a los motores. [17]

Debido a que el sistema se puede desacoplar, es factible para la sintonización del controlador PID realizar los métodos experimentales y analíticos, los cuales son puestos en comparación con la sintonización hallada por el Software EPOS UserInterface de los controladores que ofrecen la capacidad realizar la sintonización automática de las ganancias de los controles PI y PID en cascada.

El algoritmo de generación de trayectorias es de carácter fundamental en el sistema ya que realiza el perfil (posiciones, velocidad y aceleraciones) de la trayectoria realizada por cada articulación cumpliendo con las restricciones de los actuadores para el movimiento punto a punto, además del cálculo de todos puntos de paso y perfil de cada uno estos para la ejecución de los movimientos más complejos como son Línea recta y Circunferencia. Las trayectorias generadas por dicho algoritmo serán isócronas y continuas. [15]

Finalmente, para el control del brazo robot es necesario el desarrollo de una interface Hombre – Máquina en una computadora personal que permita la comunicación con el driver EPOS 24/1, siendo esencial el análisis del protocolo de comunicación del controlador, el cual será explicado posteriormente.

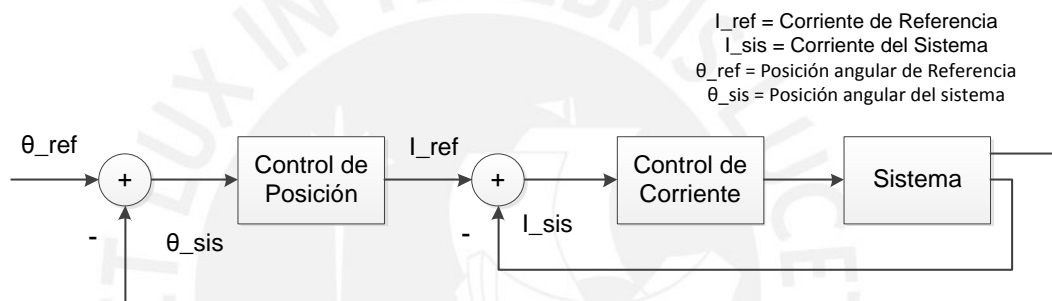


Figura: 2 Diagrama de bloques del sistema de control del EPOS 24/1

## CAPÍTULO 2.- MARCO TEÓRICO

### 2.1 Estado del Arte

#### 2.1.1 Presentación del asunto de estudio

Actualmente vivimos en un mundo globalizado en donde la alta competitividad industrial genera estándares de producción cada vez más elevados, además de tasas de producción más altas, eficientes y flexibles, es en este contexto donde la robótica de manipuladores ha tenido un gran auge.

Los robots manipuladores son comúnmente empleados en tareas repetitivas, de precisión y velocidad, así como en actividades peligrosas para operadores humanos. Sus principales ventajas están en la reducción de costos de producción, incremento en la precisión, calidad y la productividad, y una mayor flexibilidad comparada con las máquinas especializadas. Por lo tanto para que estas máquinas

puedan realizar todas estas tareas es necesario el desarrollo de mecanismos y algoritmos de control cada vez más eficientes.

A continuación se muestra el avance logrado en el control digital de los robots manipuladores para diferentes aplicaciones.

### 2.1.2 Estado de la investigación

A mediados de los años cincuenta, George C. Devol desarrolló un dispositivo que él llamo “Dispositivo de transferencia programada articulada”, un manipulador cuya operación podía ser programada (y por tanto, cambiada) y que podía seguir una secuencia de pasos de movimientos determinados por las instrucciones en el programa. Posteriores desarrollos de este concepto por Devol y Joseph F. Engelberger condujo al primer robot industrial, introducido por Unimation Inc. en 1959 [1]. Con el lanzamiento de este robot la forma del control empezó a tomar un auge en investigaciones y el desarrollo.

El control de robots manipuladores puede clasificarse según su aplicación en dos clases:

- 1.- **Control de posición:** El robot se desplaza libremente en su espacio de trabajo realizando movimientos sin interactuar con el medio ambiente. Tareas como el pintado y soldadura pueden ser realizadas por esta clase de manipuladores.
- 2.- **Control de movimiento:** El robot interactúa con su medio ambiente, por ejemplo, aplicando una fuerza sobre éste, como manipuladores de pulido, ensamblado de precisión, transporte. [2]

A continuación se mencionan los algoritmos de control que han tenido mayor énfasis en la robótica de manipuladores:

#### 2.1.2.1 Control PID desacoplado

Control ampliamente usado en las industrias para satisfacer el control de posición para el caso de robots modelados con gravedad. Esta técnica de control se basa en que los efectos de la dinámica acoplada del sistema pueden ser modelados como perturbaciones, realizado un algoritmo control PID de forma independiente para cada articulación como se muestra en la figura 3. [8]

En 1995 Rafael Kelly, propone reglas simples para la optimización del control PID en robots manipuladores las cuales conllevan siempre a la estabilidad. Para llevar a cabo dicho procedimiento se requiere de conocimientos de la estructura, de las matrices de inercia y torque gravitacional de la dinámica del robot. [3]

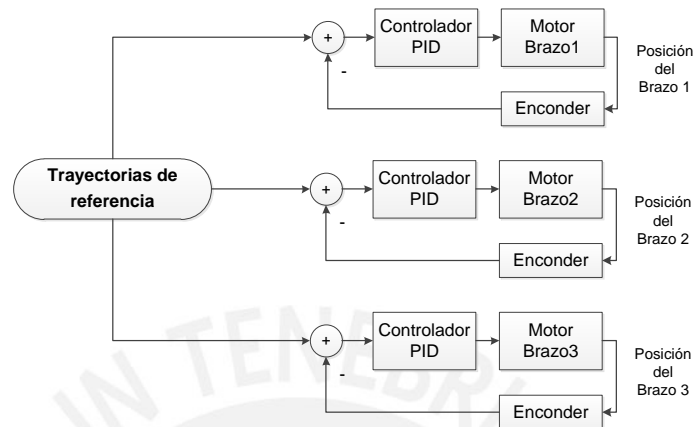


Figura: 3 Esquema del algoritmo de control PD con compensación

### 2.1.2.2 Control Par Calculado

Esta técnica de control permite describir el sistema de control del robot (el cual presenta una dinámica no lineal) en malla cerrada mediante una ecuación diferencial lineal, utilizando el modelo dinámico del robot como se muestra en la figura 4. La razón de su importancia es que puede satisfacer el objetivo de control de movimiento de forma global, con una selección trivial de sus parámetros de diseño ( $K_p$  y  $K_v$ ). [2]

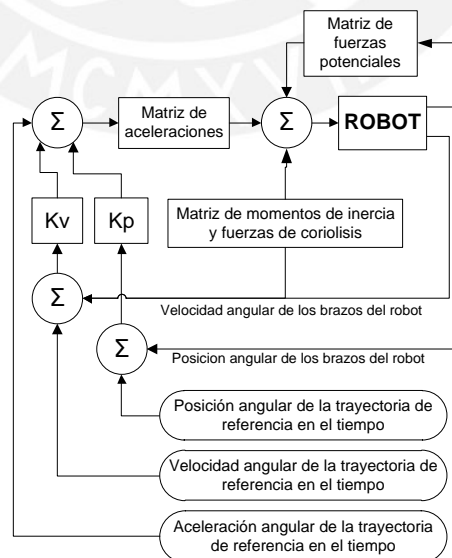


Figura: 4 Esquema del algoritmo de control par calculado

### 2.1.2.3 Control Adaptativo

La técnica del control adaptativo permite reajustar los parámetros del controlador cada vez que el comportamiento del robot se modifica (no-lineal), de modo que los movimientos de éste se aproximen siempre a las especificaciones deseadas, por ello se ha utilizado esta técnica en diferentes estrategias de control como PID, par calculado, entre otras.

En Diciembre del 2005 C. C. Cheah, C. Liu y J.J.E. Slotine resolvieron el problema de adaptación concurrente para robots con cinemática y dinámica incierta (Control Adaptativo Jacobiano), estos parámetros se actualizan en línea por las leyes de actualización de parámetros. El algoritmo requiere sólo medir la posición del efector, además de los ángulos y las velocidades de las articulaciones del robot. [4]

### 2.1.2.4 Control PID Lógica Difusa

En el 2004 G. M. Khoury, M. Saad, H. Y. Kanaan and C. Asmar estudiaron la aplicación del control de lógica difusa en un brazo robot de cinco grados de libertad (DOF). Simulan los resultados en comparación de un controlador PID normal mostrando el éxito del PID difuso sobre el normal, el control par calculado y el control adaptativo directo. Estos hallazgos lo realizan utilizando el mismo modelo dinámico del robot. [5]

### 2.1.2.5 Control Predictivo funcional

La estrategia del control predictivo MPC se resume en la figura 5, en donde, dado un Set point definido en el horizonte finito  $[0, h]$ , la predicción de la salida futura  $y_P$  deberá alcanzar el Set point futuro siguiendo una trayectoria de referencia  $y_R$ . [14]

En el 2006 Oscar Vivas publica un artículo en el que presenta una aplicación eficiente de un control por modelo de referencia (MPC) sobre un robot de cuatro grados de libertad tipo SCARA. El control estudiado fue un control predictivo funcional, el que hace uso del modelo dinámico simplificado del robot. Con el fin de comparar diferentes estrategias de control, se diseña un controlador clásico tipo PID y dos controladores basados en el modelo de referencia, en donde los mejores resultados se obtuvieron por parte del control predictivo funcional, gracias a su capacidad de anticipar los cambios presentes en el sistema.



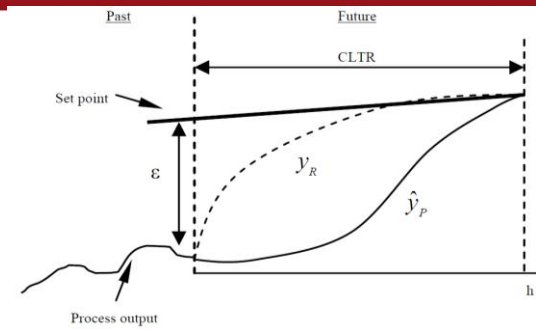


Figura: 5 Trayectoria de referencia de la estrategia de un control predictivo [14]

### 2.1.3 Síntesis sobre el asunto de estudio

Los innumerables beneficios que pueden brindar los robots manipuladores en diversos campos ha llevado al hombre a desarrollar a lo largo del tiempo una gran variedad de formas de controlarlos. Sin embargo actualmente no existe un tipo de control que se imponga sobre los demás.

De las investigaciones realizadas [1], [2], [3], [4], [5] y [14] se llega a la siguiente conclusión. La selección del algoritmo de control en robots manipuladores depende de las siguientes características:

- El tipo de tarea a realizar por el robot, control de posición o de movimiento.
- Disponibilidad o factibilidad de obtener el modelo dinámico del robot manipulador.
- La complejidad del control que podría requerir gran cantidad de procesamiento computacional – Relacionado con los grados de libertad.
- Velocidad de trabajo.
- La precisión, exactitud en la tarea asignada (Error en estado estable).
- Conocimiento previo de la trayectoria (Control predictivo).

## 2.2 Conceptualizaciones Generales

Tabla: 1 Conceptos Generales

Robot	Definición	Maquina diseñada para la realización de trabajos productivos, de imitación de movimientos y comportamientos de seres vivos. [8]		
	Robot Manipulador	Robot articulado formado de eslabones conectados a través de uniones o articulaciones que permiten un movimiento relativo entre dos eslabones consecutivos. Además debe ser controlado automáticamente, reprogramable y de múltiples usos, pudiendo estar en un lugar fijo o móvil para su empleo en aplicaciones industriales.[2]		
	Driver	Definición	Dispositivo de potencia que permite interconectar el motor con el sistema de control.	
	Sensores	Definición	Son los dispositivos primarios de medición de variables de un proceso, siendo algunos usados para lectura e indicación y otros para la transformación de la variable medida en una señal eléctrica.	
		Tipos usados en Robots Manipuladores	De velocidad (Tacómetro, Enconder) De posición (Enconder, Potenciómetro) De fuerza (Galgas extensiométricas)	
	Grados de Libertad	Definición	Número total de movimientos independientes que un dispositivo puede efectuar. Un objeto en el espacio de 3 dimensiones puede trasladarse y rotar en cualquiera de sus 3 ejes. Dando un máximo de 6 grados de libertad. [11]	
Control	Definición	Se refiere a una operación que, en presencia de perturbaciones, tiende a reducir la diferencia entre la salida de un sistema y alguna entrada de referencia y lo continúa haciendo con base en esta diferencia.[7] Para la realización de un Control Realimentado es		

		necesario: Sistema (Robot) Controlador Sensores Actuadores (Motores y Drivers)	
	Control Digital	Es aquel que en el cual una o más señales sólo pueden cambiar en valores discretos de tiempo. Estos instantes especifican el tiempo en el que se miden las señales o se extraen los datos de la memoria de una computadora digital. Dichas señales tiene una amplitud cuantificada y se representa mediante una secuencia de números binarios [10].	
	Control PID	La combinación de una acción de control proporcional, una acción de control integral y una acción de control derivativa, para disminuir el tiempo de respuesta, el sub-amortiguamiento y el error en estado estable. [7]	
	Modelo Matemático	Definición	En un sistema dinámico se define a modelo matemático como un conjunto de ecuaciones que representan la dinámica del sistema con precisión o, al menos, bastante bien. [7]
Generación de Trayectoria	Definición	Consiste en parametrizar en el tiempo la curva deseada en coordenadas del espacio de trabajo a coordenadas articulares.[2]	
Software de Simulación	Matlab	MATLAB es un lenguaje de computación técnica de alto nivel y un entorno interactivo para desarrollo de algoritmos, visualización de datos, análisis de datos y cálculo numérico. Con el uso de MATLAB se pueden resolver los problemas técnicos de computación más rápido que con los lenguajes de programación tradicionales, tales como C, C y Fortran. [12]	
	Simulink	Definición	Es un entorno para simulación multidominio y diseño basado en modelos para sistemas dinámicos y embebidos. Proporciona un entorno gráfico interactivo y un conjunto de

			bibliotecas de bloques personalizables que permiten diseñar, simular, implementar y probar una variedad de sistemas de tiempo variable. [9]
	Interface Usuario GUI-Matlab	Definición	GUI (Interface gráfica de usuario) Es una pantalla grafica en una o más ventanas que contiene controles, llamados componentes que permiten al usuario realizar tareas interactivas. Los componentes pueden ser: Menús, Barras de Herramientas, Botones, Radio botones, Listas, Sliders, etc.

### 2.3 Modelo Teórico

El sistema de control digital de un brazo Robot requiere de hardware y software para su elaboración. A partir del hecho que se cuenta con el Brazo robot de 3 grados de libertad construido por el CETAM y los controladores (EPOS 24/1) con capacidad de procesamiento de algoritmos de control (procesador DSP), control de motores brushless (Maxon EC), capacidad de interconexión entre controladores y comunicación serial RS-232 [17], se posibilita la opción de realizar los algoritmos de control en los controladores EPOS y comunicar estos con una computadora personal a través de una interface hombre-máquina, que permita visualización gráfica, control de los movimientos del brazo robot (Punto a punto, Línea recta y Circunferencia) y comunicación con los controladores.

A continuación se explica el funcionamiento total del sistema, como se puede apreciar en la figura 7.

El sistema de control empieza con el usuario el cual al abrir la interface usuario, habilita el puerto serial y realiza las configuraciones previas requeridas por los controladores; una vez abierta la GUI (Interface gráfica de usuario de Matlab) el usuario establece el modo de movimiento, la posición inicial, final y el tiempo de desplazamiento, posteriormente dicha información es enviada al algoritmo de cinemática inversa el cual calcula los puntos indicados en coordenadas articulares y envía la información al algoritmo de generación de trayectoria, el cual calcula los perfiles de movimiento enviando la información secuencialmente a los controladores EPOS 24/1 para la ejecución del movimiento ( a medida que el robot alcanza las

posiciones previamente indicadas) y a la interface para la visualización de curvas de movimiento. Dentro de la información enviada a los controladores, están, las posiciones angulares deseadas y las velocidades angulares máximas en cada desplazamiento por par de puntos.

Una vez que la información llega a los controladores estos realizan el control PID Desacoplado de posición angular sobre cada articulación (Como se mencionó anteriormente) como se muestra en la figura 3. La sintonización de los parámetros de los controladores se realiza de forma analítica, experimental (utilizando la plataforma Matlab - Simulink) y a través del software de los controladores (EPOS UserInterface), comparando sus resultados.

El envío de información entre la computadora el controlador deseado (articulación deseada) es a través de un puerto USB de la computadora a un adaptador USB-RS232 y luego a un controlador (EPOS 24/1) que realiza el papel de Gateway enviado la información de forma transparente entre RS-232 y la interfaz CAN OPEN (a través de la cual todos los controladores se interconectan) hacia el controlador deseado, como se muestra en la figura 6.

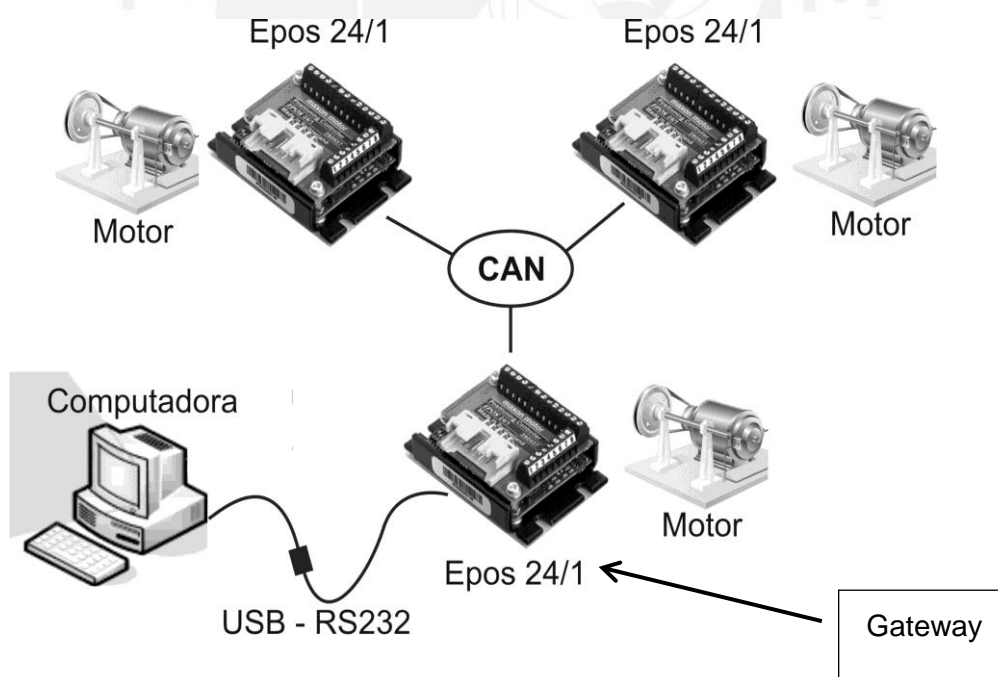


Figura: 6 Diagrama de conexión de los controladores.

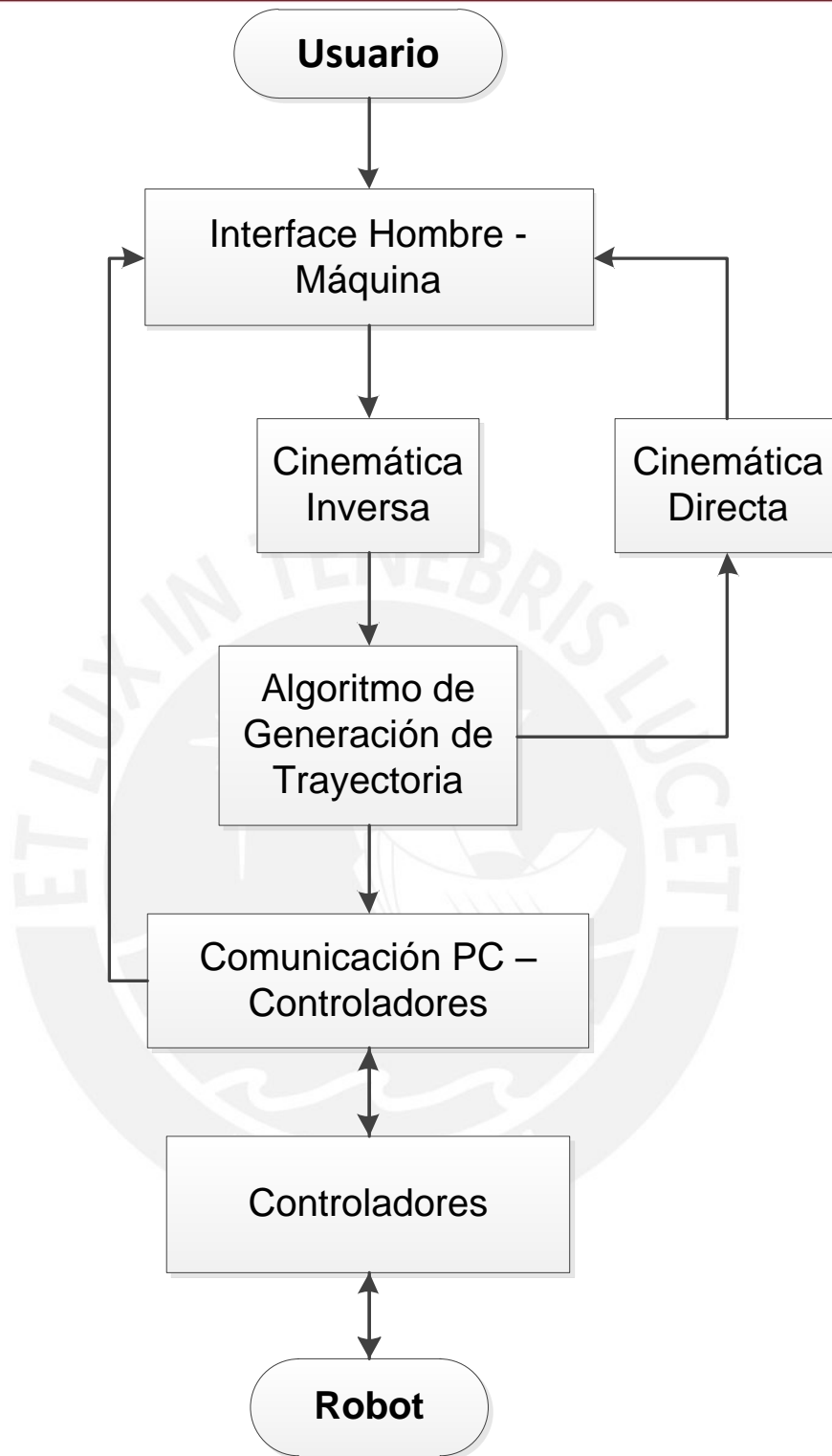


Figura: 7 Funcionamiento global del sistema.

## 2.4 Descripción mínima de aspectos teóricos materia de estudio

### 2.4.1 Coordenadas Homogéneas

Es una matriz 4x4 que representa la transformación de un vector de coordenadas homogéneas de un sistema de coordenadas a otro, esta contiene una matriz de traslación (ubicación), de Rotación, de escalado y finalmente una de Perspectiva:

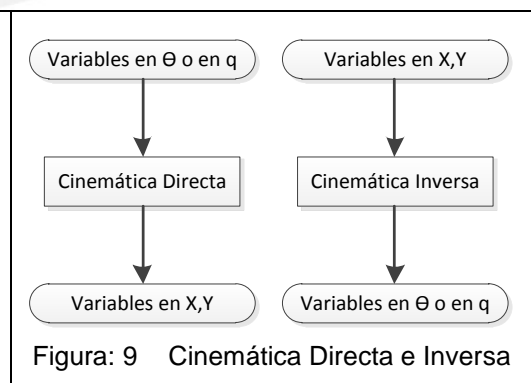
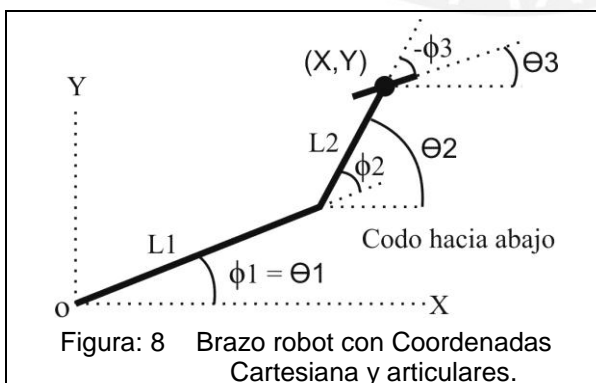
$$T = \begin{bmatrix} R_{3 \times 3} & p_{3 \times 1} \\ f_{1 \times 3} & w_{1 \times 1} \end{bmatrix} = \begin{bmatrix} \text{Rotación} & \text{Traslación} \\ \text{Perspectiva} & \text{Escalado} \end{bmatrix} = \begin{bmatrix} R_{3 \times 3} & p_{3 \times 1} \\ 0 & 1 \end{bmatrix} \quad (1)$$

En robótica la submatriz  $f_{1 \times 3}$ , que representa una transformación de perspectiva, es nula; y la submatriz  $w_{1 \times 1}$ , que representa un escalado global, es la unidad,  $R_{3 \times 3}$  representa la orientación de un sistema OUVW rotado y  $p_{3 \times 1}$  representa la traslación con respecto al sistema de referencia OXYZ. [1]

### 2.4.2 Modelos Cinemáticos

Los controladores de motores suelen controlar en el espacio de las variables de la articulación, mientras que los objetos que se manipulan se suelen expresar en el sistema de coordenadas cartesianas. Con el fin de controlar la posición y orientación del efector final de un robot para alcanzar su objeto, es fundamental la obtención de la solución cinemática inversa. En otras palabras, dada la posición y orientación del efector final, calcular los ángulos de cada articulación. [1]

A su vez también es importante el modelo cinemático directo el cual, dadas determinadas posiciones angulares, calcula las posiciones en el espacio con un determinado sistema de referencia, como en la figura 8.



### 2.4.3 Modelos Dinámicos

El modelo dinámico del robot es una expresión matemática que representa las ecuaciones de movimiento del robot que describen su conducta dinámica. Tales ecuaciones son útiles para la simulación en computadora, diseño de ecuaciones de control apropiadas para el robot y la evaluación del diseño y estructura del Brazo Robot. [1]

Las ecuaciones dinámicas de un robot se pueden obtenerse a partir de la formulación Newton-Euler, de las ecuaciones de movimiento generalizadas de D'Alembert o de la formulación Lagrange-Euler junto con la convención Denavit-Hartenberg, siendo utilizado este último por poseer una formulación altamente estructurada [1]. Utilizando esta formulación se obtiene la dinámica inversa del sistema, la cual calcula el torque necesario para mover el robot desde una determinada posición angular a una velocidad angular y aceleración angular deseada. La ecuación de la formulación Lagrange-Euler [1], [2] es la siguiente:

$$\tau_c = D(\phi)\ddot{\phi} + h(\phi, \dot{\phi})\dot{\phi} + g(\phi) \quad (2)$$

En donde:

$\tau_c$  = Torque necesario para el movimiento deseado

$\phi$  = Posición articular

$\dot{\phi}$  = Velocidad articular

$\ddot{\phi}$  = Aceleración articular

$D(\phi)$  = Matriz simétrica relacionada con la aceleración

$h(\phi, \dot{\phi})$  = Vector de Fuerzas de Coriolis y Centrifugas

$g(\phi)$  = Vector de fuerza de carga gravitatoria

Finalmente para la simulación es necesario el modelo dinámico directo, el cual se halla con la misma ecuación despejando en función de la aceleración articular, para luego integrar dicha aceleración obteniendo velocidad y posición articular, es así que el modelo dinámico directo calcula la aceleración angular del sistema debido a un determinado torque.



### 2.4.4 Controlador PID Desacoplado

La estructura del brazo robot que deseamos controlar consta de 3 grados de libertad, por lo cual podemos realizar el control a los 3 grados de forma simultanea (Todo el sistema) o realizar el control a cada uno de los grados basándonos en que los efectos de un grado de libertad no influyen en gran medida a otros y es representado como una perturbación. A este último se le conoce como control desacoplado y es como se llevará a cabo nuestra solución. [8]

Los miembros de la familia de controladores PID, incluyen tres acciones: Proporcional (P), Integral (I) y Derivativa (D) como se muestra en la figura 10. A su vez las acciones se pueden encontrar en paralelo (sumándose), serie (multiplicándose) o una combinación de ambos [1]. Para el presente tema de estudio utilizaremos la estructura paralela siguiente:

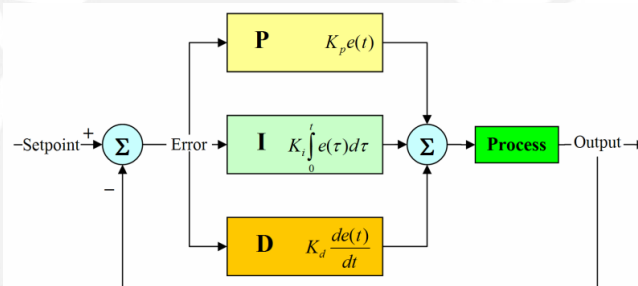


Figura: 10 Diagrama de Bloques del controlador PID

La función de las acciones es descrita a continuación en la tabla 2:

Tabla: 2 Acciones proporcional, integral y derivativa en un algoritmo de control PID

Acción Proporcional (P)	Acción Integral (I)	Acción Derivativa (D)
Da una salida del controlador correctiva que es proporcional al error.	Da una salida del controlador que es proporcional al error	Da una salida del controlador que es proporcional al cambio del error. (derivada)
Un controlador proporcional puede controlar cualquier planta estable, pero posee desempeño limitado y un error estacionario (offset).	Un controlador acumulado (integral), lo que implica que es un modo de controlar lento. Su función principal es desaparecer el error en estado estacionario.	Tiene como principal característica que disminuye el tiempo de respuesta.

## 2.5 Descripción de materia del estudio vigente

### 2.5.1 Generación de trayectoria

Definir el movimiento de un robot implica controlar dicho robot de manera que siga un camino pre planificado. El objetivo es por tanto establecer cuáles son las trayectorias que debe seguir cada articulación del robot a lo largo del tiempo para conseguir los objetivos. [15]

Por lo cual un algoritmo de generación de trayectoria es aquel que tiene la función de suavizar el movimiento y de determinar las posiciones, velocidades y aceleraciones durante todo el trayecto de un punto a otro garantizando que se cumplan las restricciones de los actuadores, por tal razón se definen 3 intervalos, uno de aceleración, otro de velocidad constante y finalmente uno de desaceleración interpolándose en puntos en donde las posiciones y velocidades son iguales [1], como se muestra en la figura 11.

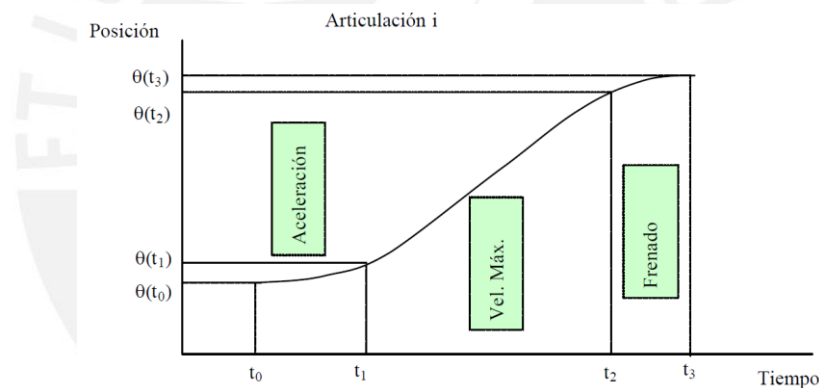


Figura: 11 Perfil de trayectoria de un punto a otro [1]

Para el desarrollo de las funciones se define un tiempo  $t \in [0,1]$  para cada intervalo, en donde:

$\tau =$  Tiempo real

$\tau_i =$  tiempo real al final de cada trayecto

$$t_i = \tau_i - \tau_{i-1} \tag{3}$$

$$t = \frac{\tau - \tau_{i-1}}{\tau_i - \tau_{i-1}} = \frac{\tau - \tau_{i-1}}{t_i} \quad \tau \in [0,1] \tag{4}$$

$$\text{Función Posición} = h_i \tag{5}$$

$$\text{Función Velocidad} = V_i = \frac{dh_i}{d\tau} = \frac{dh_i}{dt} \frac{dt}{d\tau} = \dot{h}_i \frac{1}{t_i} \tag{6}$$

$$\text{Aceleración} = A_i = \frac{d^2h_i}{d\tau^2} = \ddot{h}_i \frac{1}{t_i^2} \tag{7}$$

### 2.5.2 Sintonización del controlador PID-Reglas Ziegler y Nichols

Ziegler y Nichols propusieron reglas para determinar la sintonización de los controladores P, PI y PID, basando en las características de la respuesta transitoria de una planta. Existen 2 métodos para la aplicación de la reglas, el primer método utiliza la curva de reacción de la planta en lazo abierto a una entrada escalón unitario y es aplicado cuando la planta no contiene integrados ni polos dominantes complejos, por lo cual no puede ser usado en nuestro caso [7]. Ya que el control de corriente del controlador Epos 24/1 como se muestra en la figura 2 se encuentra en lazo cerrado y este no presenta la opción de abrir el lazo, además que el lazo de control de posición presenta integradores, se emplea el segundo método, el cual utiliza la ganancia crítica ( $K_{cr}$ ) que genera oscilaciones sostenidas como se muestra en la figura 12, en donde el periodo de estas oscilaciones es llamado periodo crítico ( $P_{cr}$ ); este método no es válido si para todo valor de  $K > 0$  no se presentan oscilaciones.

Tabla: 3 Cálculo de las ganancias del segundo método de las Reglas de Ziegler y Nichols.

Tipo de controlador	$K_p$	$K_i$	$K_d$
PID	$0.6K_{cr}$	$1.2K_{cr}/P_{cr}$	$0.075K_{cr}P_{cr}$

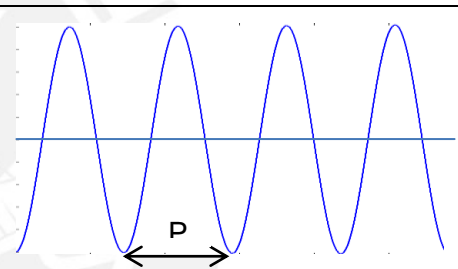


Figura: 12 Oscilaciones sostenidas en respuesta del sistema al  $K$  crítico ( $K_{cr}$ )

### 2.5.3 Sintonización del controlador PI de corriente de forma Analítica

La sintonización del control PI puede realizarse de forma analítica con la función de transferencia de la planta, aproximando la respuesta de la planta en lazo cerrado a una función de segundo o de primer orden, con los tiempos de respuesta deseados aproximados.

Para cumplir dichos objetivos es necesario factorizar el polinomio del denominador de la función de transferencia y encontrar los polos dominantes del sistema (polos a los cuales se aproximará el comportamiento), los cuales son los más cercanos al eje imaginario, en esta situación si el sistema presenta un cero en lazo cerrado cerca de un polo en lazo cerrado la respuesta transitoria de este par se vuelve pequeña pudiéndolos despreciar y/o simplificar. [7]

## CAPÍTULO 3.- DESARROLLO DE LA TESIS

### 3.1 Objetivos.

#### 3.1.1 Objetivo General

Diseñar e implementar las interfaces hardware y software, el algoritmo de control PID Desacoplado, así como la interface de usuario que permita comandar los movimientos de un robot serial planar de 3 grados de libertad, con accionamiento eléctrico construido por el CETAM.

#### 3.1.2 Objetivos Específicos

- Sintonización analítica y experimental de los algoritmos de control PI y PID desacoplado en Matlab-Simulink del driver EPOS24/1 para control de posición del brazo robot.
- Implementación de un algoritmo de generación de trayectoria (en Matlab) para ejecutar los comandos de: Movimiento Punto a Punto; Movimiento Línea Recta; y Movimiento Circular.
- Desarrollo de una interface usuario para visualización de gráfica, comunicación y control del brazo robot.

### 3.2 Descripción y justificación del diseño.

#### 3.2.1 Modelos Cinemáticos

Para la elaboración de los modelos cinemáticos es fundamental el conocimiento de la estructura del robot, para el presente trabajo se ha empleado el método geométrico por su simplicidad relativa a la cantidad de grados de libertad del robot, Además se ha tomado en cuenta la característica de diseño en la cual el movimiento de los eslabones es transmitido desde los motores, los cuales se encuentran en la base, mediante fajas, esta característica genera que las posiciones articulares  $\Phi_i$  las cuales son con respecto al eslabón anterior, varíen si el motor  $i$  se acciona (hecho que no ocurre en los robots que presentan los motores en los ejes), por lo tanto para expresar las posiciones angulares del brazo en función de los motores se definen las posiciones angulares  $\theta_i$ , como se muestra en la figura 8, Las relaciones son las siguientes:

Tabla: 4 Relaciones entre  $\theta, \dot{\theta}, \ddot{\theta}$  y  $\phi, \dot{\phi}, \ddot{\phi}$

Posición angular	Velocidad angular	Aceleración angular
$\begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} = \begin{bmatrix} \phi_1 \\ \phi_1 + \phi_2 \\ \phi_1 + \phi_2 + \phi_3 \end{bmatrix} \quad (8)$	$\begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix} = \begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_1 + \dot{\phi}_2 \\ \dot{\phi}_1 + \dot{\phi}_2 + \dot{\phi}_3 \end{bmatrix} \quad (9)$	$\begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} = \begin{bmatrix} \ddot{\phi}_1 \\ \ddot{\phi}_1 + \ddot{\phi}_2 \\ \ddot{\phi}_1 + \ddot{\phi}_2 + \ddot{\phi}_3 \end{bmatrix} \quad (10)$

### 3.2.1.1 Modelo Cinemático Directo

La función del modelo cinemático directo como se explicó en el capítulo 2.4.2 es la de calcular Las coordenadas espaciales (Sistemas Cartesiano para la traslación XY producida por el brazo y el antebrazo) en base a las coordenadas Articulares o las posiciones angulares definidas en la tabla 4, producidas por el algoritmo de generación de trayectorias, el cual es explicado posteriormente. Este modelo es importante ya que permite la visualización gráfica en la Interface Usuario de las posiciones, velocidades o aceleraciones del robot en el espacio cartesiano.

Posiciones en el espacio cartesiano:

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} 0.21 \cos(\theta_1) + 0.16 \cos(\theta_2) \\ 0.21 \sin(\theta_1) + 0.16 \sin(\theta_2) \end{bmatrix} \quad (11)$$

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \end{bmatrix} = \begin{bmatrix} -0.21 \sin(\theta_1) \dot{\theta}_1 - 0.16 \sin(\theta_2) \dot{\theta}_2 \\ 0.21 \cos(\theta_1) \dot{\theta}_1 + 0.16 \cos(\theta_2) \dot{\theta}_2 \end{bmatrix} \quad (12)$$

$$\begin{bmatrix} \ddot{X} \\ \ddot{Y} \end{bmatrix} = \begin{bmatrix} 0.21 \cos(\theta_1) + 0.16 \cos(\theta_2) \\ 0.21 \sin(\theta_1) + 0.16 \sin(\theta_2) \end{bmatrix} \quad (13)$$

Dónde:

$X, Y =$  Posiciones en el espacio cartesiano

$\dot{X}, \dot{Y} =$  Velocidad en el espacio cartesiano

$\ddot{X}, \ddot{Y} =$  Aceleraciones en el espacio cartesiano

### 3.2.1.2 Modelo Cinemático Inverso

La función del modelo cinemático inverso es la de calcular las posiciones  $\phi_i$  o  $\theta_i$  utilizando como dato las posiciones en el espacio cartesiano, ingresadas por el usuario o las posiciones calculadas por el algoritmo de generación de trayectoria a ser usadas como puntos de paso para los movimientos en línea recta o circunferencia.

Posiciones en coordenadas articulares:

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} L1 \cos(\theta_1) + L2 \cos(\theta_2) \\ L1 \sin(\theta_1) + L2 \sin(\theta_2) \end{bmatrix} = \begin{bmatrix} 0.21 \cos(\phi_1) + 0.16 \cos(\phi_1) \cos(\phi_2) - 0.16 \sin(\phi_1) \sin(\phi_2) \\ 0.21 \sin(\phi_1) + 0.16 \sin(\phi_1) \cos(\phi_2) + 0.16 \sin(\phi_1) \cos(\phi_2) \end{bmatrix} \quad (14)$$

$$\phi_2 = Codo \times Acos\left(\frac{X^2 + Y^2 - L1^2 - L2^2}{2L1L2}\right) = Codo \times Acos\left(\frac{X^2 + Y^2 - 0.0696}{0.0335}\right) \quad (15)$$

$$\begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} L1 + L2 \cos(\phi_2) \\ L2 \sin(\phi_2) \end{bmatrix} \quad (16)$$

$$\phi_1 = \text{atan2} \left( \frac{aY - bX}{a^2 + b^2}, \frac{bY + aX}{a^2 + b^2} \right) \quad (17)$$

$$\text{codo} = [-1; 1]$$

Dónde:

- $\text{Atan2}(Y,X)$  es un algoritmo de Matlab que permite calcular el arco tangente en los cuatro cuadrantes (rango de  $[-\pi, \pi]$  que es el doble del rango que posee el Arco tangente normal).
- $\text{Codo} = [-1; 1]$  Es la representación de la posición del codo del brazo robot (hacia abajo es representado por 1, mientras lo opuesto con -1) como se muestra en la figura 8.

### 3.2.2 Modelo Dinámico

La función del modelo dinámico para el presente trabajo de tesis es la de servir de representación del modelo real del robot para la realización de simulaciones de movimiento del robot, así como la comprobación de los algoritmos de control y cálculo de los actuadores eléctricos.

Para el presente modelo dinámico del robot de 3 grados de libertad se utilizó:

- La formulación Euler – Lagrange (2)
- Representación Matricial Denavit-Hartenberg [1]
- Relaciones entre  $\theta, \dot{\theta}, \ddot{\theta}$  y  $\phi, \dot{\phi}, \ddot{\phi}$  (8), (9) y (10)
- Modelo de los motores Brushless [19]:

$$\ddot{\theta}_i(J_{mi}) = \frac{I_i K_t}{n} - \dot{\theta}_i(b_{mi}) - \tau_c \quad (18)$$

$$\frac{V_i - \frac{K_b i}{n_i} \dot{\theta}_i}{R_i + sL_i} = I_i \quad (19)$$

Empleando las ecuaciones (2), (9), (10) y (18) se obtiene (20) que es la unificación del modelo dinámico del robot con una parte del modelo de los motores, esto es realizado para obtener la corriente como variable de entrada, necesaria para el control:

$$\frac{K_t}{n} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} - b \begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_1 + \dot{\phi}_2 \\ \dot{\phi}_1 + \dot{\phi}_2 + \dot{\phi}_3 \end{bmatrix} - H(\theta, \dot{\theta}) - G(Q) = D(\theta) \begin{bmatrix} \ddot{\phi}_1 \\ \ddot{\phi}_2 \\ \ddot{\phi}_3 \end{bmatrix} + J \begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_1 + \dot{\phi}_2 \\ \dot{\phi}_1 + \dot{\phi}_2 + \dot{\phi}_3 \end{bmatrix} \quad (20)$$

$$D(\theta) + \begin{bmatrix} J_1 & 0 & 0 \\ J_2 & J_2 & 0 \\ J_3 & J_3 & J_3 \end{bmatrix} = Df(\theta) \quad (21)$$

$$H(\theta, \dot{\theta}) + b \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix} = Hf(\theta, \dot{\theta}) \quad (22)$$

Reemplazando (21) y (22) en (20), se obtiene:

$$Df(\theta)^{-1} \left[ \frac{Kt}{n} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} - Hf(\theta, \dot{\theta}) - g(\theta) \right] = \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 - \dot{\theta}_1 \\ \ddot{\theta}_3 - \dot{\theta}_2 \end{bmatrix} \quad (23)$$

La cual es la función de transferencia principal del modelo dinámico directo del robot, esta función presenta como entradas: la corriente de cada motor, las posiciones y velocidades angulares; y como resultado de cálculo, las aceleraciones angulares del robot, las cuales al integrarse, permiten hallar las velocidades y posiciones angulares, necesarias como variables de entrada, como es mostrado en la figura 13. Finalmente Unificado el modelo dinámico directo (23) con la parte restante del modelo del motor (19) se obtiene el modelo total del sistema el cual puede es representado en la figura 13, este sistema presenta el bloque Dinámica Directa, el cual es diseñado en Simulink basado en las ecuaciones (23), (8), (9) y (10).

$J_{mi}$  = Momento de Inercia del rotor del brazo  $i$  [ $Kg m^2$ ]

$J_c$  = Momento de Inercia de la carga del brazo  $i$  [ $Kg m^2$ ]

$b_m$  = Coeficiente de Viscosidad del motor del brazo  $i$   $\left[ \frac{N - m}{rad s^{-1}} \right]$

$b_c$  = Coeficiente de Viscosidad de la carga del brazo  $i$   $\left[ \frac{N - m}{rad s^{-1}} \right]$

$b = \left( \frac{b_{mi}}{n^2} + b_{ci} \right)$  = Matriz de coeficiente de Viscosidad del sistema  $\left[ \frac{N - m}{rad s^{-1}} \right]$

$J = \frac{J_m}{n^2}$  Matriz de momento Inercia de los motores [ $Kg m^2$ ]

$n$  = Relación de Reducción  $\left[ \frac{N_m}{N_c} \right] \in [0,1]$

$\tau_c$  = Torque resultante del motor visto desde la carga [ $N - m$ ]

$L$  = Inductancia por fase del motor ( $H$ )

$R$  = Resistencia por fase del motor ( $\Omega$ )

$V$  = Tensión de motor [ $V$ ]

$Kt$  = constante de motor  $\left[ \frac{N - m}{A} \right]$

$Kb$  = constante de la armadura del motor  $\left[ \frac{V - s}{rad} \right]$

$n$  = Relación de Reducción  $\left[ \frac{N_m}{N_c} \right]$

Reemplazando la función de transferencia principal (23) con los datos del sistema (Distancias, masas, momentos de inercia, centros de masa) se obtiene:

$$Df(\theta) =$$

$$\begin{bmatrix} 8.1779 \times 10^{-3} \cos(\theta_2 - \theta_1) + 0.16823 & 4.089 \times 10^{-3} \cos(\theta_2 - \theta_1) + 1.1594 \times 10^{-3} & 1.2710 \times 10^{-5} \\ 4.089 \times 10^{-3} \cos(\theta_2 - \theta_1) + 0.1168 & 0.1168 & 1.2710 \times 10^{-5} \\ 0.1542 & 0.1542 & 0.1542 \end{bmatrix} \quad (24)$$

$$Hf(\theta, \dot{\theta}) = \begin{bmatrix} 0.1969 \dot{\theta}_1 + 4.089 \times 10^{-3} \sin(\theta_2 - \theta_1) (\dot{\theta}_1 + \dot{\theta}_2) (\dot{\theta}_1 - \dot{\theta}_2) \\ 4.0890 \times 10^{-3} \sin(\theta_2 - \theta_1) \dot{\theta}_1^2 + 0.5338 \dot{\theta}_2 \\ 0.1969 \dot{\theta}_3 \end{bmatrix} \quad (25)$$

$$G(\theta) = \begin{bmatrix} 0.9127 \cos(\theta_1) + 0.1908 \cos(\theta_2) \\ 0.1908 \cos(\theta_2) \\ 0 \end{bmatrix} \quad (26)$$

$$\begin{bmatrix} \frac{Kt_1}{n_1} \\ \frac{Kt_2}{n_2} \\ \frac{Kt_3}{n_3} \end{bmatrix} = \begin{bmatrix} \frac{Kb_1}{n_1} \\ \frac{Kb_2}{n_2} \\ \frac{Kb_3}{n_3} \end{bmatrix} = \begin{bmatrix} 11.1 \\ 18.1 \\ 1.1 \end{bmatrix} 10^{-3} \quad (27)$$

$$\begin{bmatrix} R_1 \\ R_2 \\ R_3 \end{bmatrix} = \begin{bmatrix} 2.28 \\ 12.4 \\ 2.28 \end{bmatrix} \quad (28)$$

$$\begin{bmatrix} L_1 \\ L_2 \\ L_3 \end{bmatrix} = \begin{bmatrix} 0.214 \\ 0.488 \\ 0.214 \end{bmatrix} 10^{-3} \quad (29)$$

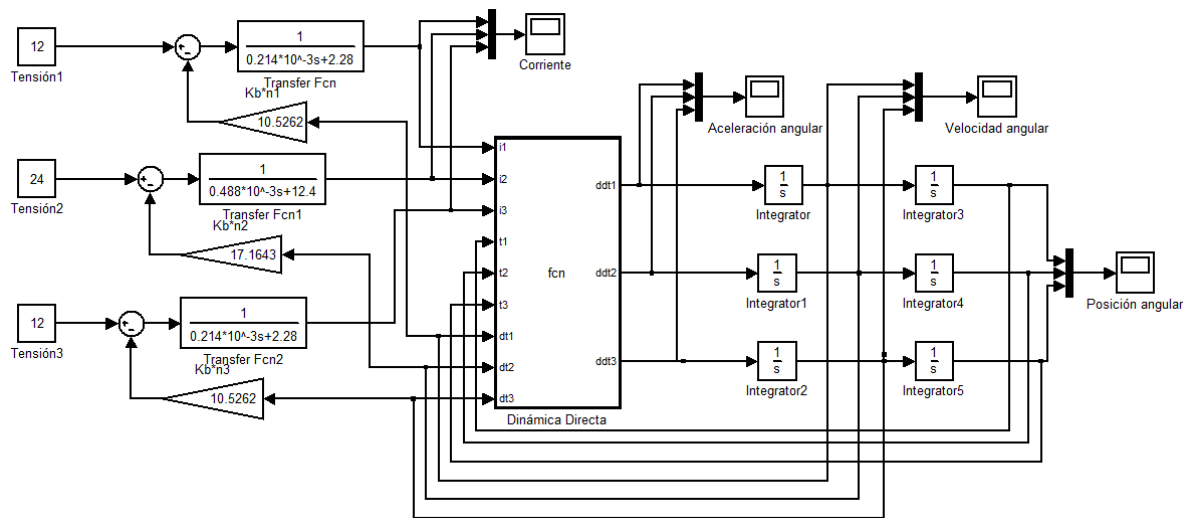


Figura: 13 Modelo dinámico del robot con los actuadores.

### 3.2.3 Algoritmo generación de trayectoria

EL presente trabajo de Tesis tiene como objetivo la realización de 3 tipos de movimiento por parte del brazo robot de 3 grados de libertad, los cuales son:

- Punto a punto
- Línea recta
- Circunferencia.



Para la realización de los movimientos del brazo robot de 3 grados de libertad se plantea la elaboración de Trayectorias coordinadas o isócronas [15].

El control de las articulaciones del robot es realizado por los Controladores EPOS 24/1, los cuales presentan un algoritmo de generación de trayectorias de forma trapezoidal y Sinusoidal [15], el presente trabajo utiliza el perfil de trayectoria trapezoidal para todos los movimientos a realizar.

Es muy común encontrar trayectorias interpoladas con polinomios de orden 4-3-4, 3-5-3 o 5 cubicas (splines cúbicos) [1] y el cálculo de estas es similar pese al diferente orden de los polinomios que presentan, es por ello que se toman de referencia para el cálculo del algoritmo de generación de trayectoria del EPOS 24/1.

La trayectoria trapezoidal generada por el EPOS 24/1 tiene la característica de ser 2-1-2 es decir dos polinomios de segundo orden y uno de primer orden [17], este controlador presenta la limitación que no se puede realizar trayectorias más complejas en la cuales se definen puntos iniciales, finales y puntos de paso. Esta limitación influye directamente en los movimientos: Línea recta y Circunferencia, generando que en el movimiento entre punto y punto de movimiento el controlador acelere y desacelere completamente, presentando paradas en cada punto.

### 3.2.3.1 Cálculo de la trayectoria punto a punto

Para el Epos24/1 se Define las siguientes funciones de posición:

$$h_1 = a_2 t^2 + a_1 t + a_0 \quad (30)$$

$$h_2 = b_1 t + b \quad (31)$$

$$h_3 = c_2 t^2 + c_1 t + c_0 \quad (32)$$

Datos de Cálculo:

$$\theta_{t_0} = \text{Posición angular Inicial (Rad)}$$

$$\theta_{t_3} = \text{Posición angular Final (Rad)}$$

$$V_m = \text{Velocidad máxima alcanzada (Rad/s)}$$

$$A_c = \text{Aceleración máxima } \left( \frac{\text{Rad}}{\text{s}^2} \right)$$

$$D_a c = \text{Desaceleración máxima } \left( \frac{\text{Rad}}{\text{s}^2} \right)$$

Tomando las limitaciones (Velocidad de  $t_0$  y  $t_3 = 0$ ) del EPOS 24/1 y resolviendo (1), (2) y (3) para las velocidades, aceleraciones y reemplazando los datos de Cálculo se llega a:

Tiempo total de movimiento

$$T_{TOTAL} = \frac{Vm^2 Dac - Vm^2 Ac + 2 Dac Ac (\theta_{t3} - \theta_{t0})}{2 Dac Ac Vm} \quad (33)$$

Velocidad Máxima de movimiento

$$Delta = Dac^2 Ac^2 T^2 - 2 Dac Ac (Dac - Ac) (\theta_{t3} - \theta_{t0}) \quad (34)$$

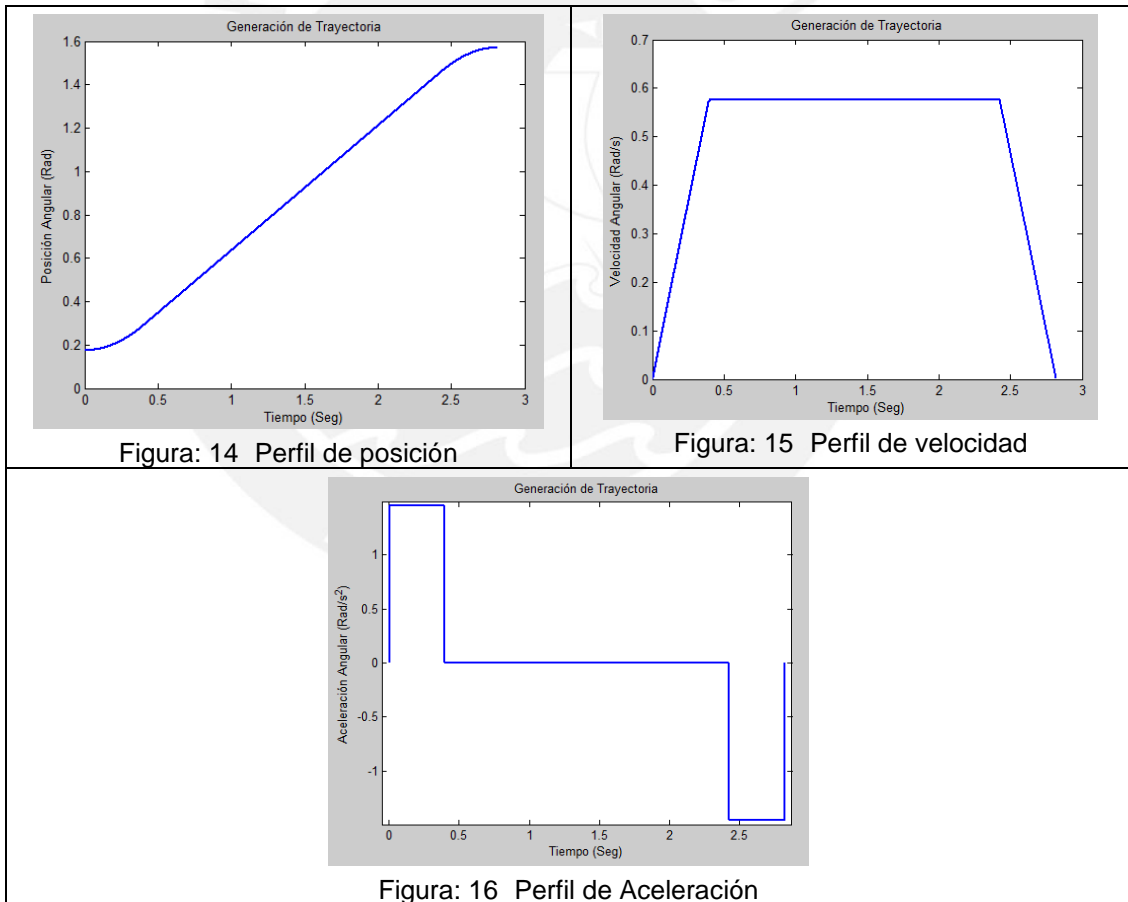
Si  $Delta < 0 \Rightarrow$  Tiempo insuficiente

$$Vm = \frac{2 Dac Ac T + \sqrt{Delta}}{2 (Dac - Ac)} \quad (35)$$

$t_1 = \frac{Vm}{Ac} \quad (36)$	$t_3 = \frac{Vm}{Dac} \quad (37)$	$t_2 = \frac{\theta_{t3} - \theta_{t0} + \frac{t_3^2 Dac - t_1^2 Ac}{2}}{Vm} \quad (38)$
----------------------------------	-----------------------------------	--

Ejemplo de generación de trayectoria

Para:  $\theta_{t0} = 10^\circ = 0.1745 \text{ (Rad)}$  ;  $\theta_{t3} = 90^\circ = 1.5708 \text{ (Rad)}$  ;  $T_{TOTAL} = 2.82 \text{ (s)}$



Hallándose:  $Vm = 0.5759 \text{ (Rad/s)}$

### 3.2.3.2 Movimiento en Línea Recta y Circunferencia

Para los movimientos en línea recta y circunferencia es necesario definir cierta cantidad de puntos por los cuales el brazo robot debe pasar, dicha cantidad está definida por la distancia deseada entre estos desplazamientos y por la trayectoria ideal total deseada como se puede observar en la figura 17 y 18 en donde la trayectoria ideal está representada por una línea roja y la real calculada por la línea azul.

Dada la limitación del controlador de no poder realizar un movimiento continuo entre tres o más puntos definidos (solo desde un punto inicial a uno final), para realizar los movimientos en línea recta o circunferencia, se realiza un desplazamiento punto a punto entre los intervalos calculados por los cuales debe pasar el brazo robot, esto genera que los errores de desplazamiento sean proporcionales a la dimensión de los intervalos y el tiempo de desplazamiento sea proporcional a la cantidad de intervalos, ya que en cada intervalo el robot acelera y frena.

Ejemplo de generación de trayectoria en Línea recta y circunferencia.

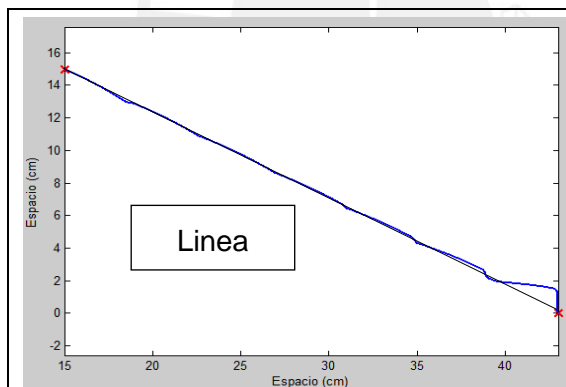


Figura: 17 Movimiento en Línea Recta

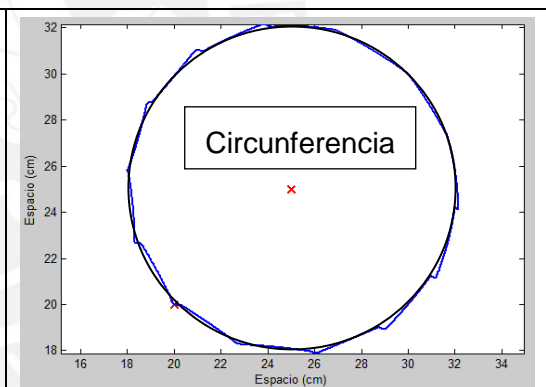


Figura: 18 Movimiento en Circunferencia

### 3.2.4 Control PI de corriente y PID de posición del brazo robot

El controlador EPOS24/1 presenta un control en cascada que consta de un control PI de corriente interno seguido por un control PID para la posición como lo muestra la figura 2. [17] por dicho motivo para realizar la sintonización total del controlador, primero se debe realizar la sintonización del control de corriente, seguido por el posición.

### 3.2.4.1 PI – Corriente

Para la determinación de las ganancias del control PI de corriente se empleó el método analítico en el espacio continuo para luego ser transferido al espacio discreto y posteriormente el método iterativo (Automático) del software EPOS UserInterface del controlador.

En primer lugar se obtiene la función de transferencia del sistema desacoplado (39), así como el diagrama de bloques para su simulación como se muestra en la figura 19.

$$\frac{V \frac{b_i}{L J_i} (s \frac{J_i}{b_i} + 1) + \tau_c \frac{n K b}{L J_i}}{(s^2 + s (\frac{b_i}{J_i} + \frac{R}{L}) + \frac{K t K b}{L J_i} + \frac{R b_i}{L J_i})} = I \tag{39}$$

En donde:

$b_i = (J_m + J_c n^2) =$  Coeficiente de Viscosidad del motor y la carga del brazo  $i$   $[\frac{N - m}{rad s^{-1}}]$

$J_i = (b_m + b_c n^2) =$  Momento Inercia del motor y la carga del brazo  $i$   $[Kg m^2]$

$\tau_c =$  Torque externo que representa los rozamiento; fuerzas centrifugas; y de coriolisis  $[N - m]$

$T =$  Periodo de muestreo del controlador  $[s]$

$K_m = K_t = K_b$

Otras = Definidas anteriormente

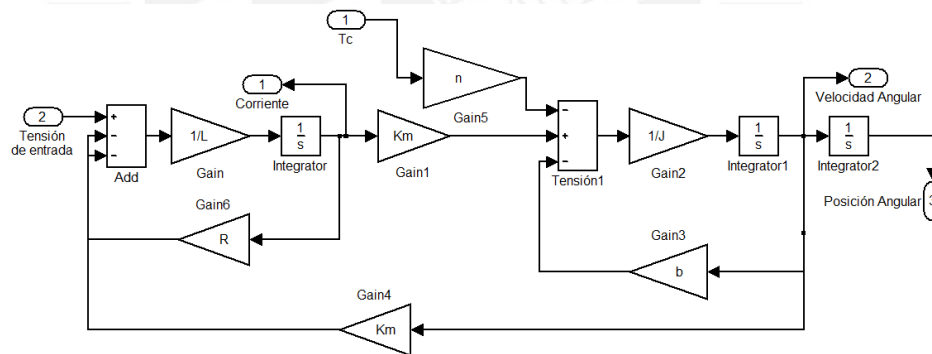


Figura: 19 Modelo del sistema desacoplado de los motores con sus cargas respectivas (Planta) [17]

Una vez obtenida (39), surge la dificultad que el tiempo de muestreo del controlador no es el necesario para ya que su Periodo de muestreo del control de corriente es de  $100 \times 10^{-6}$  [segundos] [17] y el requerido, basado en los polos de las funciones de transferencia desacopladas como se muestra en la tabla 5 y la ecuación (40) es de  $7.9070 \times 10^{-6}$  [segundos], generando inestabilidades en el control. Por esta razón se propone una solución parcial al problema, modelando el retardo del controlador (41) en el espacio continuo y calculando las ganancias con dicho retardo.

Tabla: 5 Polos del Planta

	Polo 1	Polo 2
Motor 1	$-1.0510 \times 10^4$	$-1.4561 \times 10^2$
Motor 2	$-2.5294 \times 10^4$	$-1.2003 \times 10^2$
Motor 3	$-1.0471 \times 10^4$	$-1.8457 \times 10^2$

Donde por regla teórica [7], [16] el periodo de muestreo debe ser:

$$T_{necesario} < \frac{1}{5|\text{polo}_{\text{mínimo}}|} = \frac{1}{5 \times 2.5294 \times 10^4} = 7.9070 \times 10^{-6} [\text{s}] \quad (40)$$

Modelo Retardo:  $\frac{2/T}{s+2/T} \quad (41)$

Una vez resuelta la función de transferencia en lazo cerrado del sistema con el modelo del retardo (41) y el controlador PI como se muestra en la figura 10 se procede a aproximar el orden del sistema a uno de segundo orden con las siguientes características:

Función de Transferencia en Lazo Cerrado:

$$I = \frac{I_{ref} \left( s^2 \frac{2Kp}{TL} + s \left( \frac{2Ki}{TL} + \frac{2Kpb_i}{TLJ_i} \right) + \frac{2b_i Ki}{J_i TL} \right)}{s^4 + s^3 a + s^2 \left( b + \frac{2Kp}{TL} \right) + s \left( c + \frac{2Kpb_i}{TLJ_i} + \frac{2Ki}{TL} \right) + fKi} \quad (42)$$

Diseño de una respuesta críticamente estable para  $\zeta_1 = 1$

$$(s^2 + 2\zeta_1 \omega_1 s + \omega_1^2)(s^2 + 2\zeta_2 \omega_2 s + \omega_2^2) = s^4 + s^3 a + s^2 \left( b + \frac{2Kp}{TL} \right) + s \left( c + \frac{2Kpb_i}{TLJ_i} + \frac{2Ki}{TL} \right) + fKi \quad (43)$$

Donde las ganancias se hallan de:

$$\begin{bmatrix} \omega_1^2 + (a - 2\omega_1)\omega_1 - b \\ (a - 2\omega_1)\omega_1^2 - c \end{bmatrix} = \begin{bmatrix} \frac{2}{TL} & -\frac{f}{\omega_1^2} \\ \frac{2b_i}{TLJ_i} & \left( \frac{2}{TL} - \frac{2f}{\omega_1} \right) \end{bmatrix} \begin{bmatrix} Kp \\ Ki \end{bmatrix} \quad (44)$$

Los polos del sistema en Lazo Cerrado son los siguientes:

$$x_{1,2} = -\omega_1 \pm \omega_1 ; \quad x_{3,4} = \omega_1 - \frac{a}{2} \pm \left( \sqrt{\frac{fKi}{\omega_1^2}} \right) \left( \sqrt{\frac{(a-2\omega_1)^2}{4 \frac{fKi}{\omega_1^2}} - 1} \right) \quad (45)$$

Una vez realizados los cálculos de las ganancias KP y KI (e44) se obtiene los siguientes resultados para el periodo de muestreo cercano al ideal (Tabla 6) y para el calculado con el periodo real (Tabla 7).

Tabla: 6 Ganancias del control de Corriente calculadas con el periodo cercano al ideal (Sin modelamiento de retardo)

Calculado con = T 10 <sup>-5</sup> s				
	Kp	Ki	Kp Epos	Ki Epos
Motor 1	2.6114	27630	669	707
Motor 2	5.3454	159900	1368	4093
Motor 3	2.6114	27540	669	705

Tabla: 7 Ganancias del control de Corriente calculadas con el periodo real y modelamiento de retardo

Calculado con modelo de retardo $y = T 10^{-4}s$				
	Kp	Ki	Kp Epos	Ki Epos
<b>Motor 1</b>	0.1748	7261	45	186
<b>Motor 2</b>	0.0401	34319	10	879
<b>Motor 3</b>	0.1705	7175	44	184

Con estos resultados se modela el control de corriente del sistema desacoplado como se muestra en la figura 20, la cual contiene el bloque planta representado por la figura 19, con este modelo se simulan las respuestas a un escalón de corriente obteniéndose como resultado la tabla 8.

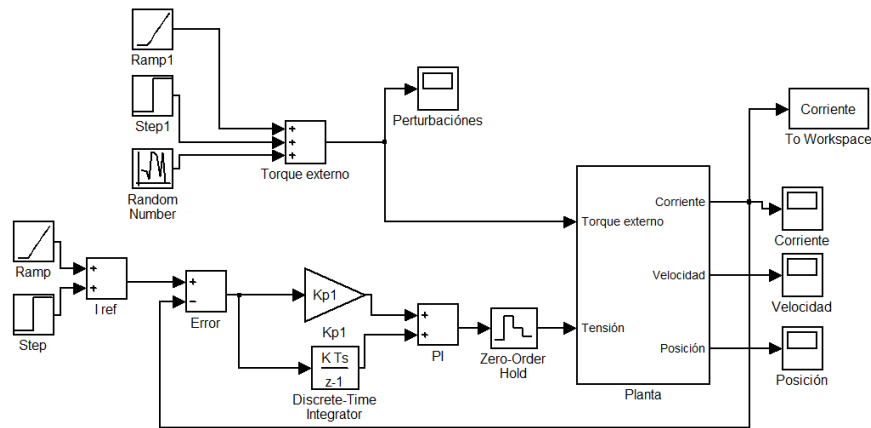
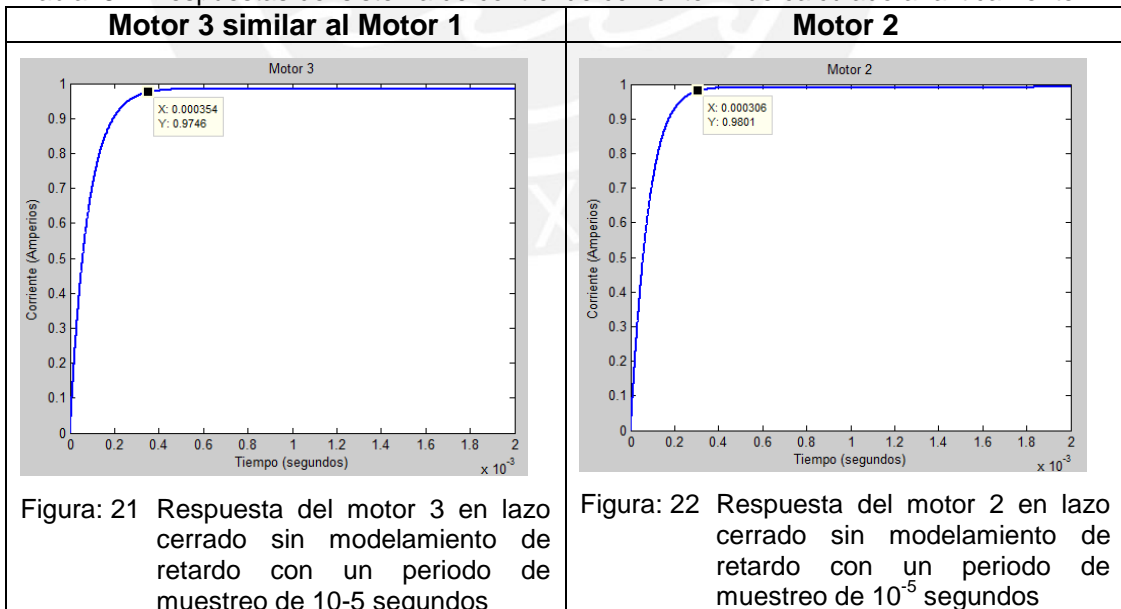


Figura: 20 Control de Corriente Desacoplado de los motores

Tabla: 8 Respuestas del sistema de control de corriente PI de calculado analíticamente



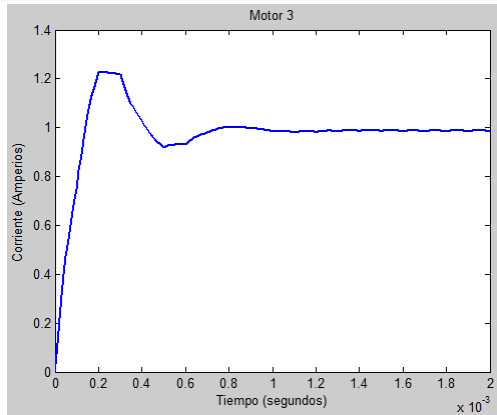


Figura: 23 Respuesta del motor 3 en lazo cerrado sin modelamiento de retardo con un periodo de muestreo de  $10^{-4}$  segundos

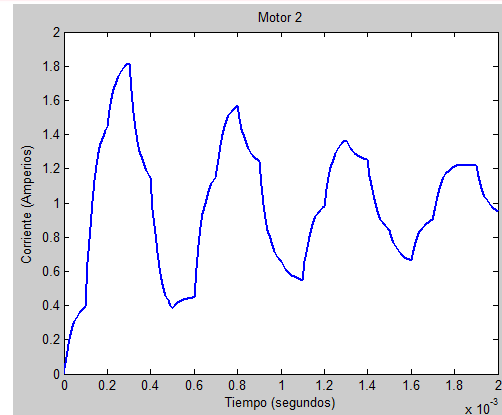


Figura: 24 Respuesta del motor 2 en lazo cerrado sin modelamiento de retardo con un periodo de muestreo de  $10^{-4}$  segundos

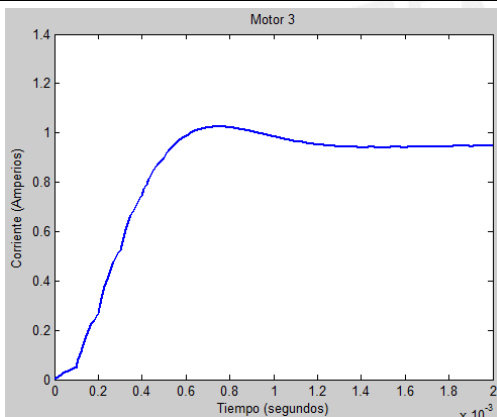


Figura: 25 Respuesta del motor 3 en lazo cerrado con modelamiento de retardo con un periodo de muestreo de  $10^{-4}$  segundos

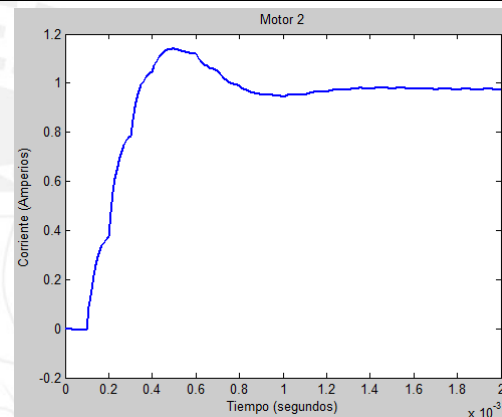


Figura: 26 Respuesta del motor 2 en lazo cerrado con modelamiento de retardo con un periodo de muestreo de  $10^{-4}$  segundos

De la Tabla anterior se puede apreciar que para un modelamiento sin retardo con una frecuencia de muestreo ideal, la respuesta al escalón es muy cercana a la de amortiguamiento crítico; mientras que la respuesta sin modelamiento de retardo con la frecuencia de muestreo real, tiende ligeramente a la inestabilidad, presentándose oscilaciones; finalmente el modelamiento con retardo a frecuencia de muestreo real, genera respuestas aceptables por el sistema en tiempo de respuesta, y sobre impulso. Estos resultados son comparados en el capítulo posterior con las respuestas obtenidas por la sintonización realizada de forma iterativa por el software EPOS UserInterface.

### 3.2.4.2 PID - Posición

Para el control PID se utilizó segundo método de las reglas de Reglas Ziegler y Nichols de forma experimental hallando el  $K_{critico}$  en cada uno de los motores para el cual el sistema presentaba oscilaciones sostenidas.

De la experimentación se llegó a las gráficas de la tabla 9 de donde se hallan el  $K_{crítico}$  y el  $P_{crítico}$ , con estos valores se calculan las ganancias de los controladores PID como se expone en la tabla 10.

Tabla: 9 Oscilaciones sostenidas para hallar las ganancias del control de posición PID.

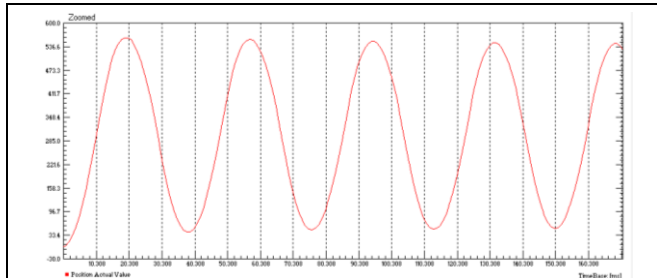


Figura: 27 Oscilaciones sostenidas del motor 1 producidas ante una respuesta escalón y un  $K_{crítico}$

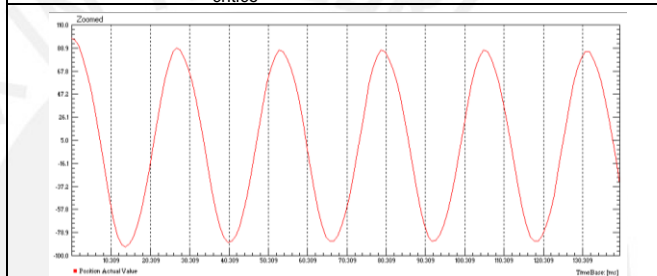


Figura: 28 Oscilaciones sostenidas del motor 2 producidas ante una respuesta escalón y un  $K_{crítico}$

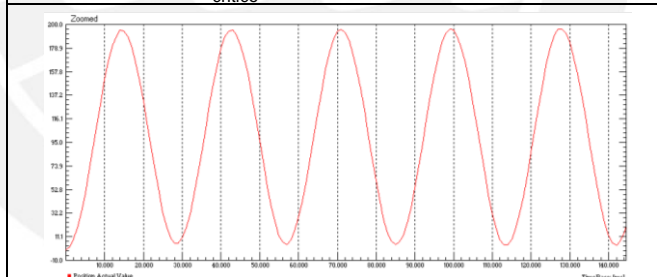


Figura: 29 Oscilaciones sostenidas del motor 3 producidas ante una respuesta escalón y un  $K_{crítico}$

Tabla: 10 Ganancias del control de posición halladas de forma experimental

Calculo de la Experimentación									
	<b>Kcr</b>	<b>Pcr</b>	<b>Kp</b>	<b>Ki</b>	<b>Kd</b>	<b>Kp Epos</b>	<b>Ki Epos</b>	<b>Kd Epos</b>	<b>Kcr Epos</b>
<b>Motor 1</b>	1.53	0.036	0.92	51.00	0.004	360	160	207	600
<b>Motor 2</b>	1.02	0.026	0.61	47.08	0.002	240	148	99	400
<b>Motor 3</b>	0.77	0.028	0.46	32.79	0.002	180	103	80	300



Las respuestas del controlador PID sintonizado son mostradas en el capítulo 4 por ser parte experimental.

### 3.3 Selección y justificación, componentes y/o equipos.

#### 3.3.1 Descripción del sistema Mecánico

El sistema Mecánico está conformado por el brazo manipulador de 3 grados de Libertad, el cual ha sido fabricado completamente en el Laboratorio de Producción de la sección de Ingeniería Mecánica, bajo diseño y supervisión del CETAM con el fin de servir de soporte educaciones así como desarrollo e investigación, siendo una de sus primeras aplicaciones de control de posición para teleoperación robótica. A continuación se muestran sus características principales en la tabla 11.

Tabla: 11 Características del Brazo Robot Construido por el CETAM


	Características	
<b>Material</b>	Acero Inoxidable 360	
<b>Peso del Brazo</b>	0.658 Kg sin la Base	
<b>Capacidad-Carga</b>	0.5 Kg	
<b>Tipo de Estructura</b>	Serial	
<b>Grados de Libertad</b>	3 ( $Y, X$ y $\theta$ )	
<b>Estructura</b>	Liviana, esbelta, con el propósito de movilizar cargas de reducida masa.	
<b>Transmisión de movimiento</b>	Transmisión de torque a los eslabones mediante fajas.	
<b>Motores</b>	Brushless DC, Marca: <b>Maxon Motors</b>	
<b>Posición-Motores</b>	En la Base del Robot	
<b>Potencia-Motores</b>	Motor1 y 3 = 20 Watts Motor2 = 12 Watts	
<b>Movimiento</b>	Planar	
<b>Fabricante</b>	CETAM	

Figura: 30 Brazo robot de 3 Grados de Libertad

### 3.3.2 Cálculo de los Accionamientos Eléctricos

#### 3.3.2.1 Motores

El robot de 3 grados de libertad cuenta con 3 motores Maxon EC (Brushless), para su movimiento, a continuación se presenta la justificación de su elección.

Se eligieron motores brushless por su alta relación potencia – Tamaño y mediante simulaciones utilizando la Dinámica Inversa del sistema, que permite calcular El torque necesario para una determinada posición, velocidad y aceleración angular.

La figura 31 contiene los bloques necesarios para el cálculo de los motores, en la cual se diseña el bloque Cinemática Inversa que contiene las ecuaciones definidas en el capítulo 3.2.1.2 presentando como entrada las trayectorias posibles que realizará el brazo robot (Línea recta y Circunferencia) y calculando las posiciones angulares, las cuales se derivan respecto al tiempo para obtener las velocidades y aceleraciones angulares que son ingresadas al bloque diseñado Dinámica Inversa, el cual contiene las ecuaciones definidas en el capítulo 3.2.2 para calcular de los torques, velocidades y potencias necesarias por los motores.

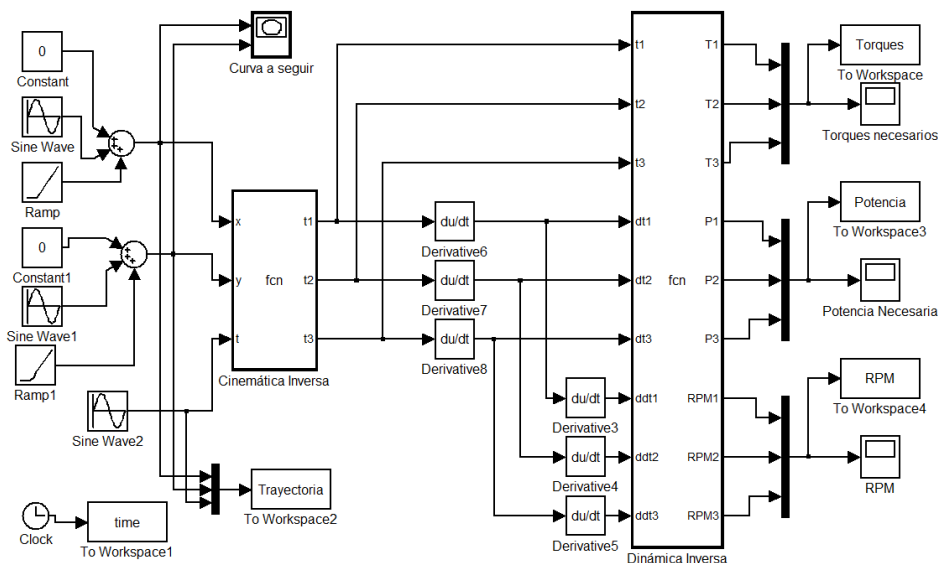
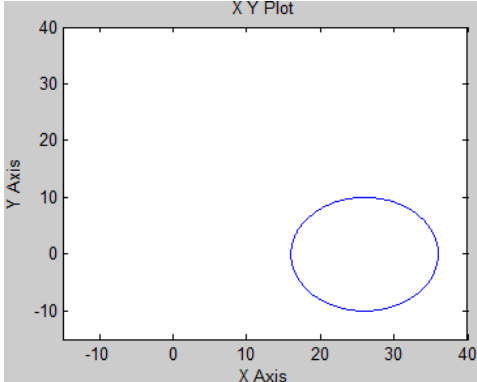
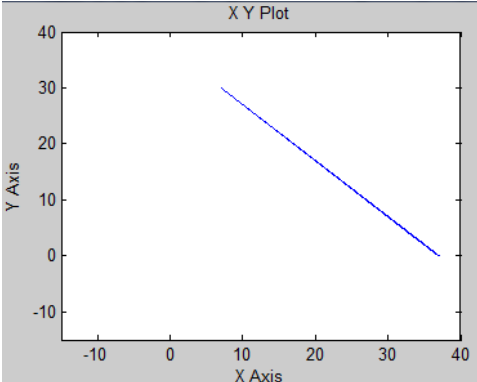
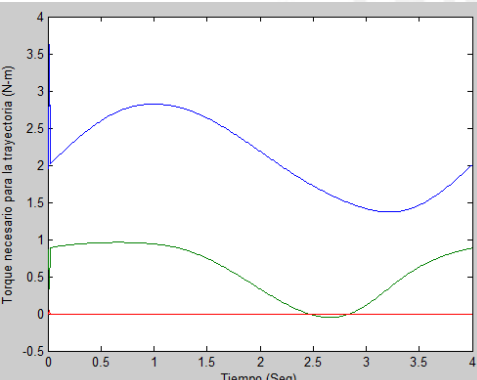
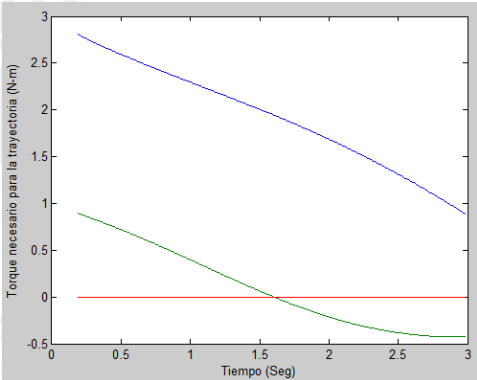
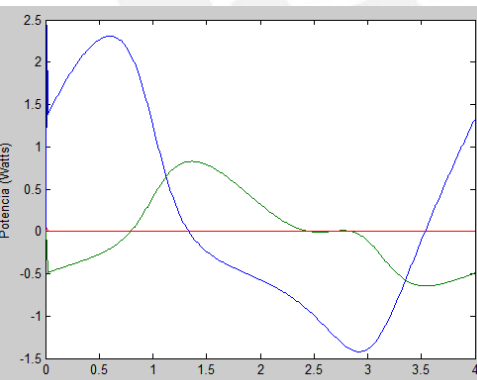
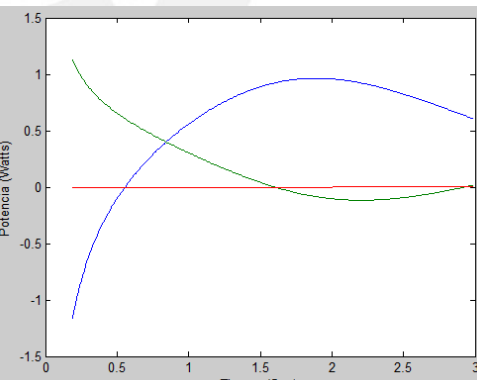


Figura: 31 Cinemática Inversa y Dinámica Inversa para el cálculo del torque necesario.

A continuación se presenta una tabla con los datos obtenidos después de las simulaciones de las trayectorias en circunferencia y Línea Recta deseadas.

Tabla: 12 Cálculo de Torques, Potencias y Velocidades Angulares para los movimientos requeridos

Trayectoria en Circunferencia	Trayectoria en Línea Recta
	
<p>Figura: 32 Trayectoria en forma de circunferencia en el Espacio X,Y.</p>	<p>Figura: 33 Trayectoria en línea recta en el Espacio X,Y.</p>
	
<p>Figura: 34 Torque necesario para la trayectoria en forma de circunferencia.</p>	<p>Figura: 35 Torque necesario para la trayectoria en línea recta.</p>
	
<p>Figura: 36 Potencia necesaria para la trayectoria en forma de circunferencia.</p>	<p>Figura: 37 Potencia necesaria para la trayectoria en línea recta.</p>

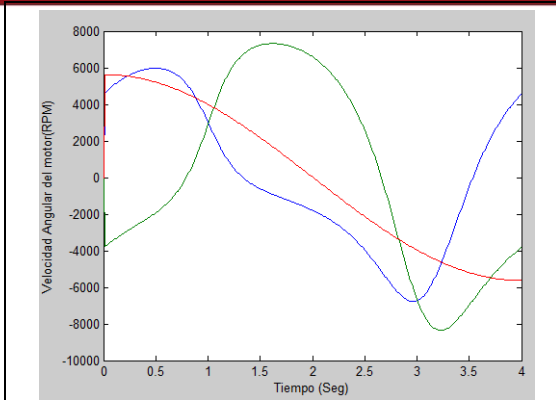


Figura: 38 Velocidad Angular necesaria para la trayectoria en forma de circunferencia.

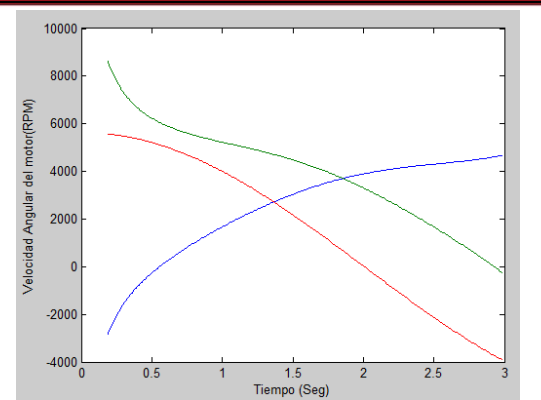


Figura: 39 Velocidad Angular necesaria para la trayectoria en línea recta.

De las gráficas anteriores para la trayectoria en forma de circunferencia, se toma el caso en el cual el robot se encuentra ligeramente estirado en una posición horizontal con una carga de 0.5 Kg en el eje final, mientras que para la trayectoria en línea recta el robot se encuentra completamente estirado en posición horizontal con la misma carga, estos casos fueron elegidos por ser uno de los requiere mayores requisitos en el motor del hombro (motor1-Linea azul), calculando de esta forma la Velocidad, potencia y torque angular.

Tabla: 13 Características de los motores [17]

Motores	Código Maxon	Torque Nominal Mot.(mNm)	Velocidad Nominal motor (RPM)	Reducción total	Torque Nominal Eje (Nm)	Velocidad Nominal Eje (Rad/s)
<b>Motor 1 Hombro</b>	169008	16.9	16500	716.95 : 1	12.12	2.41
<b>Motor 2 Codo</b>	283840	11	8240	716.95 : 1	7.89	1.2
<b>Motor 3 Efactor final</b>	169008	16.9	16500	716.95 : 1	12.12	2.41

De la tabla 12 se puede observar que:

Tabla: 14 Requisitos máximos en los motores

	Torque Máximo [N-m]	Potencia Máxima [Watts]	Velocidad Máxima [RPM]
Motor 1	3	2.5	<8000
Motor 2	1	2	<6000
Motor 3	< 0.1	< 0.1	<8000

De la tabla 13 y 14 podemos concluir que la selección de motores cumple con todos los requisitos satisfactoriamente teniendo un factor de seguridad mayor a 3 para los 3 casos.

### 3.3.2.2 Controladores

Como se mencionó anteriormente el robot consta 3 motores Maxon EC, los cuales son Brushless. Debido al alto costo de los motores, su complicado control y los objetivos para los que fue diseñado el robot, la mejor opción fue comprar controladores que ofrecieran una protección de corriente a los motores, preservándolos, realicen el control de motores brushless y ofrezcan opciones de comunicación, por lo cual se eligió el controlador EPOS 24/1 de la compañía Maxon Motors mostrado en la figura 40, el cual presenta las siguientes características:

- Permite la operación de micro motores DC y EC hasta los 20 Watts.
- Control punto a punto.
- Interconexión de controladores a través del Bus CAN.
- 6 Entradas Digitales.
- 2 Salidas Digitales.
- 2 Entradas Analógicas (10-bit-ADC).
- Modelo miniatura.
- Comunicación Maestro – Esclavo por interface RS-232.
- Posee algoritmos de control de posición angular, velocidad angular y corriente de los motores.
- Posee un software propietario “Epos UserInterface” que dentro de sus principales funciones permite: comunicación con los controladores, configuración de estos y sintonización de los parámetros de los algoritmos de control de forma experimental.
- Aplicaciones típicas:
  - Aplicaciones de pequeños aparatos
  - Sistema de automatización
  - Controladores de motores

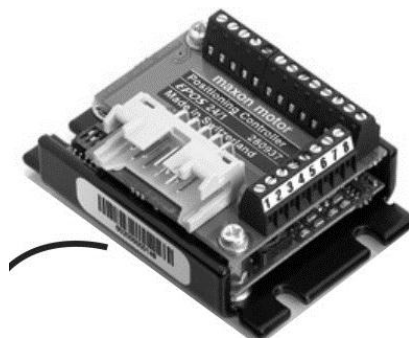


Figura: 40 Controlador Epos

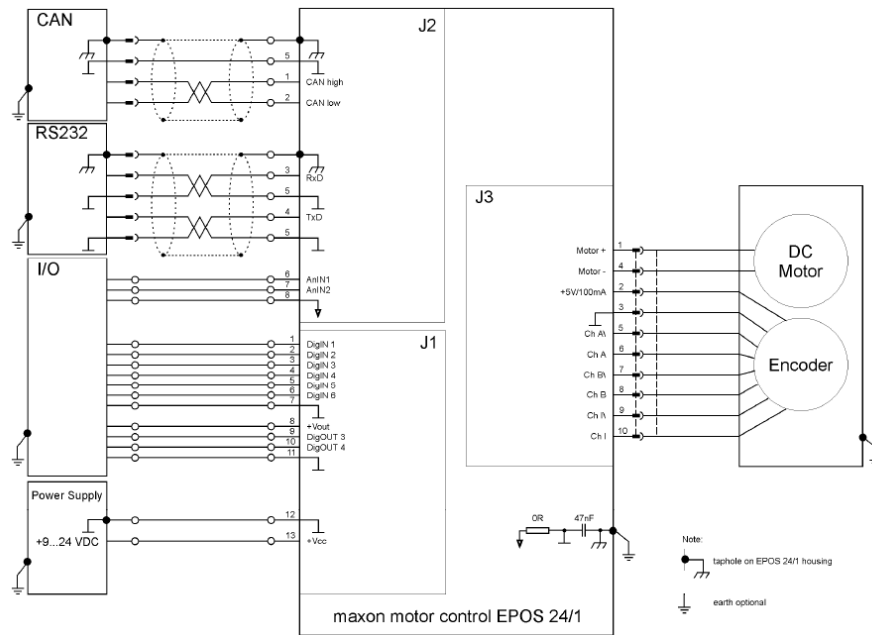


Figura: 41 Diagrama de conexiones físicas del controlador Epos 24/1

### 3.3.3 Otros

#### 3.3.3.1 Fuente de energía del Brazo robot

Para el presente trabajo de tesis se energizó el brazo robot con una fuente Switching de 12 V, siendo la tensión nominal de los motores del brazo robot, 24V, esto se realizó debido a que era necesaria una fuente de aproximadamente 6 Amperios (peor caso en donde los 3 motores están acelerando a límite permitido por el controlador) y solo se contaba con una fuente de computadora marca Intec de 200W con capacidad de 12V – 8Amp.

Debido a que un diseño y construcción de una fuente de 24V – 6Amp. Escapaba de los objetivos del presente trabajo de tesis y que el costo de estas es ligeramente elevado, se trabajó con la fuente de 12V con la restricción de trabajo de aproximadamente la mitad de la velocidad nominal en todos los motores. A continuación se muestran dos alternativas de fuentes posibles para la energización del brazo robot.


Tabla: 15 Fuentes tentativas. [18]

Marca	ABB	Siemens
Modelo	SD823	6EP1334-3BA00
Entrada	A.C. 115/230V; 50/60Hz	AC 120/230V-500V; 50/60Hz
Salida	D.C. 24V, 10A, 240W	24VDC, 10A, 240W (24V-28.8V)
Costo	\$ 195	\$100.00

### 3.3.3.2 Adaptador USB-Serial

Debido a que el presente trabajo se realizó con un Laptop modelo Dv4-1275mx, la cual no dispone de un puerto serial RS-232, fue necesario utilizar un Adaptador USB-RS232 para realizar la comunicación con los controladores con los que se cuenta. A continuación se presentan las características del adaptador elegido.

Tabla: 16 Características del Adaptador USB-RS232

Marca	Gigaware	
Modelo	26-949	
Salidas	1 x USB, 1 x DB-9(RS232 Serial)	
Entradas	1 x USB, 1 x DB-9(RS232 Serial)	
Interface Supported	USB 2.0, RS232	
Technology Features	Plug-n-play y funciones hot swap	

## 3.4 Implementación y construcción del control de posición e Interface del Robot de 3 grados de Libertad Construido por el CETAM

### 3.4.1 Interface de Usuario

La interface usuario para el control, visualización, configuración y comunicación con el brazo robot ha sido desarrollada utilizando la herramienta GUI de Matlab.

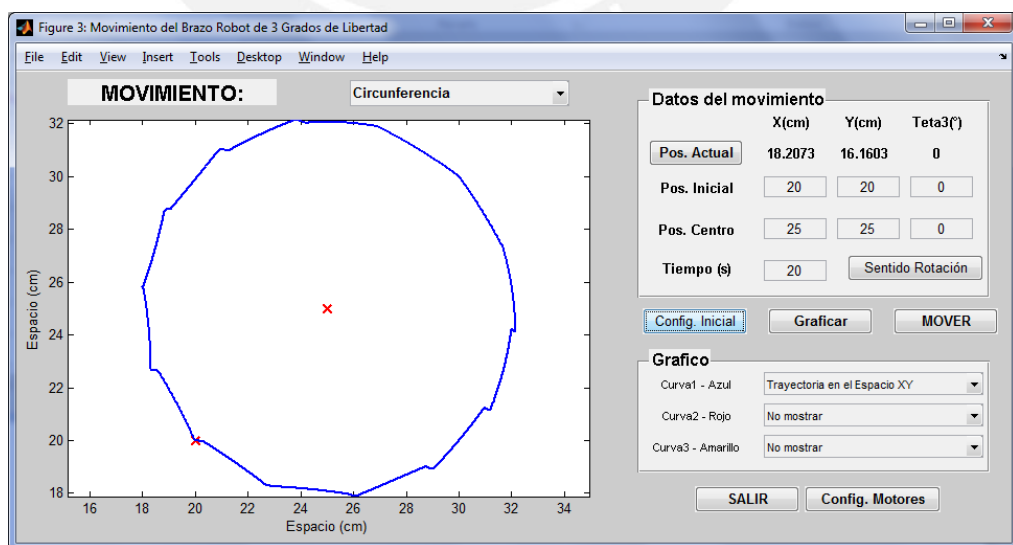


Figura: 42 Pantalla de la Interface usuario

Esta interface consta de las siguientes partes:

- Cuadro de Gráficas

Es el espacio en donde se visualizan todas la graficas deseadas, producto de los movimientos escogidos.

- Menú de tipo de movimiento

Permite seleccionar entre 3 tipos de movimiento:

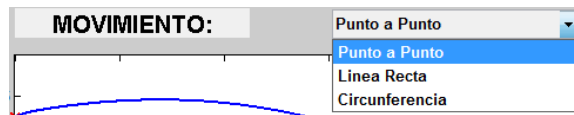


Figura: 43 Menú pop-up de elección del tipo de movimiento

- Panel de Datos del movimiento

Es la zona en donde se visualiza la posición actual del robot, así como se ingresan los datos necesarios para realizar el tipo de movimiento.

- Panel de Curvas de Gráficos

Es la zona en donde se seleccionan los datos a ser graficados en el Cuadro de gráficas, se pueden graficar hasta 3 curvas las cuales presentan las siguientes opciones:

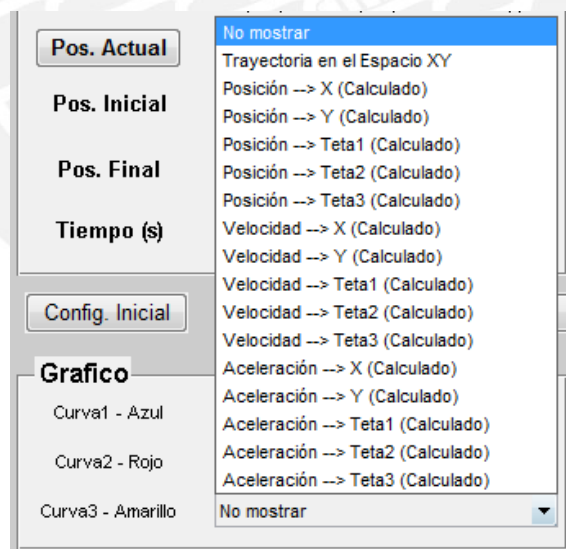


Figura: 44 Tipos de curvas

### Modo de Empleo

1. Abrir la interface “Movimiento del Brazo Robot de 3 Grados de Libertad”.
2. Seleccionar el tipo de movimiento deseado a realizar.
3. Ingresar los datos correspondientes al tipo de movimiento deseado.

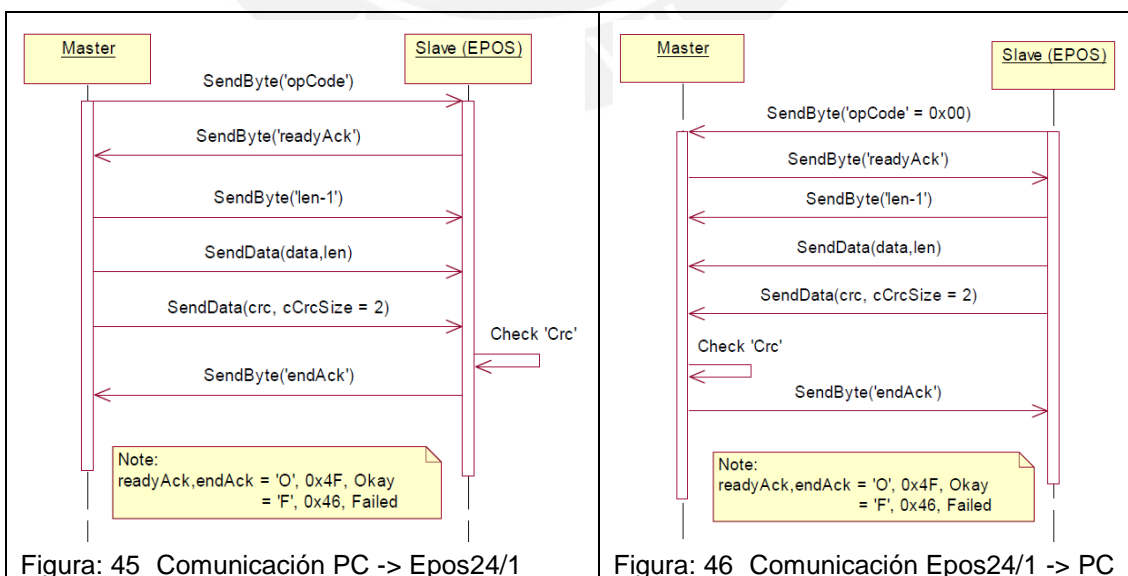


4. Dar Clic en el botón configuración Inicial, La interface verificará si es posible dicho movimiento y mostrará un mensaje: “Robot listo para moverse a la posición inicial”.
5. Liberar el espacio del Robot por seguridad.
6. Dar clic en ok y esperar a que el robot se halla desplazado.
7. Si se desea visualizar el movimiento que realizará el robot, seleccionar las curvas a visualizar en el panel gráfico y dar clic al botón graficar.
8. Finalmente, dar clic al botón MOVER donde aparecerá una mensaje que dice robot listo para moverse, dar clic en OK y el robot se moverá.

### 3.4.2 Comunicación EPOS24/1

La comunicación entre la computadora y los controladores EPOS24/1 es una comunicación maestro esclavo, en la cual la computadora se comunica directamente con un único controlador que hace la función de Gateway interconectando de forma transparente el RS-232 con la interfaz CAN OPEN (a través de la cual todos los controladores se interconectan) enviando a la información a el controlador deseado especificado en el protocolo, generando así una comunicación transparente Maestro-Esclavo con cualquier controlador del bus CAN. [17]

Para implementar la comunicación fue necesario desarrollar en Matlab un programa con todo el protocolo del Epos24/1, dicho protocolo consta de 2 etapas que son similares entre sí.



Esquema del protocolo, Etapa 1 (PC -> Epos), figura 45

1. Computador envía el OP, que indica la acción a realizar escritura o lectura.
2. El maestro envía una confirmación de recepción del código OP.
3. El computador envía un byte con la cantidad de Word que serán enviadas.
4. El computador envía los datos y el CRC de los datos enviados.
5. El EPOS24/1 verifica el CRC y envía una confirmación del mensaje recibido correctamente.

Esquema del protocolo, Etapa 2 (Epos -> PC), figura 46

6. El EPOS24/1 envía el OP '00'.
7. La computadora responde con una confirmación de OP recibido.
8. El EPOS24/1 envía un byte con la cantidad de Word que serán enviadas.
9. El EPOS24/1 envía los datos y el CRC de los datos enviados.
10. El computador verifica el CRC y envía una confirmación del mensaje recibido correctamente.

Definiciones:

- OP = Comando de operación (Define el tipo de operación a realizarse durante la comunicación, Eje: lectura o escritura).
- CRC = Palabra (16 bits) enviada para la verificación del correcto envío del mensaje, el algoritmo utilizado es CRC-CCITT [17].

## CAPÍTULO 4.-

### 4.1 Ensayos y Pruebas

Para el control PI así como para el control PID, se realizaron pruebas utilizando las ganancias de los controladores halladas analíticamente y experimentalmente por reglas en comparación con las ganancias halladas de forma automática por el software EPOS UserInterface propietario del controlador.

Obteniéndose los siguientes resultados mostrados en las tablas 17 y 18.

Tabla: 17 Respuestas del control de corriente ante un escalón con sintonización calculada analíticamente y por el software EPOS UserInterface.

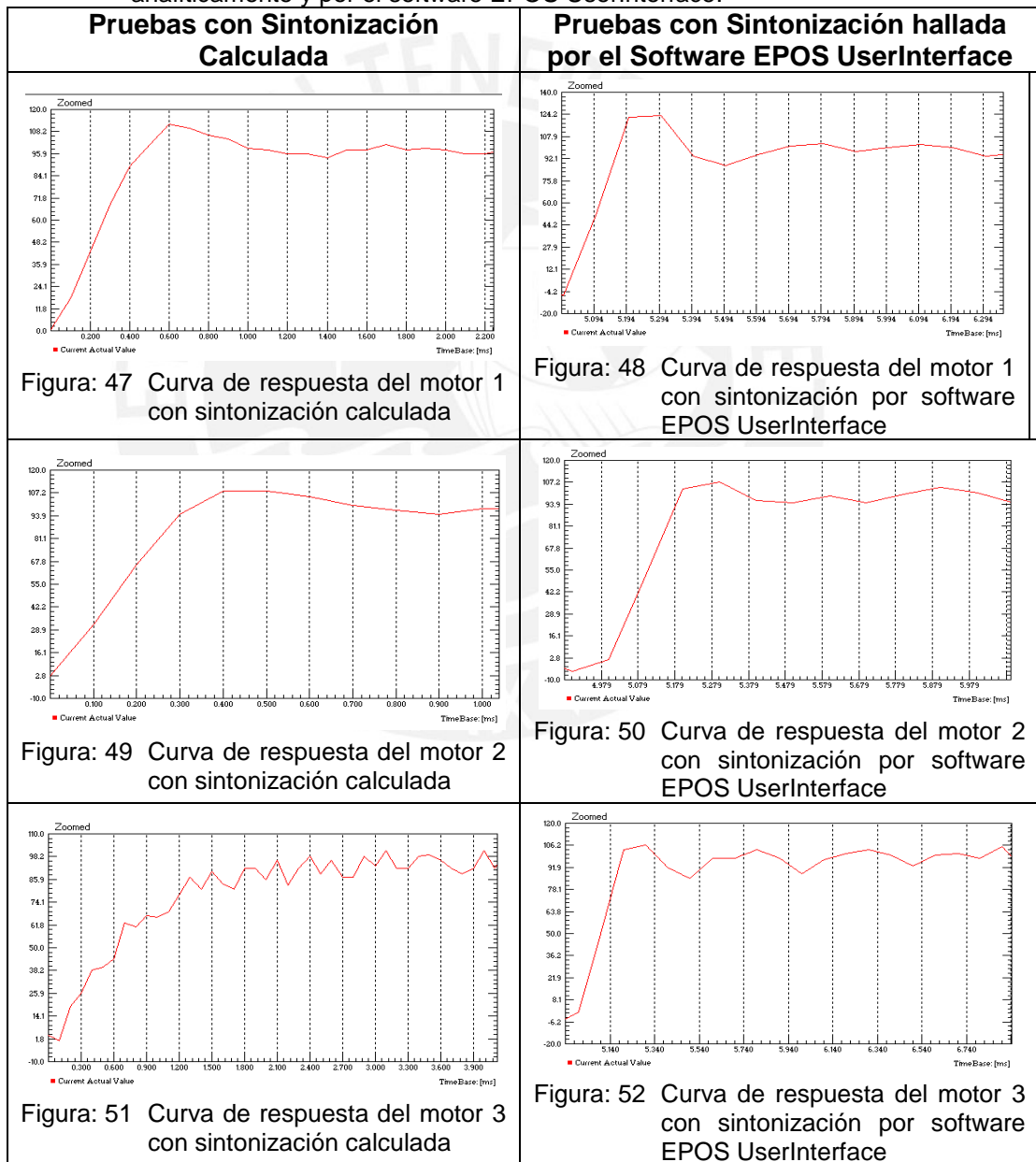


Tabla: 18 Respuestas del control de posición ante un escalón con sintonización calculada por Ziegler y Nichols y por el software EPOS UserInterface.

Pruebas con Sintonización Calculada	Pruebas con Sintonización hallada por el Software EPOS UserInterface
<p data-bbox="304 663 844 723">Figura: 53 Curva de respuesta del motor 1 con sintonización calculada</p>	<p data-bbox="871 656 1401 745">Figura: 54 Curva de respuesta del motor 1 con sintonización por software EPOS UserInterface</p>
<p data-bbox="304 1070 844 1131">Figura: 55 Curva de respuesta del motor 2 con sintonización calculada</p>	<p data-bbox="871 1059 1401 1149">Figura: 56 Curva de respuesta del motor 2 con sintonización por software EPOS UserInterface</p>
<p data-bbox="304 1485 844 1545">Figura: 57 Curva de respuesta del motor 3 con sintonización calculada</p>	<p data-bbox="871 1469 1401 1559">Figura: 58 Curva de respuesta del motor 3 con sintonización por software EPOS UserInterface</p>

## 4.2 Resultados.

### 4.2.1 PI Calculado

De los resultado obtenidos se puede observar que si el controlador presentará un tiempo de muestreo menor, el modelamiento y la respuesta al control de corriente serían mejores en tiempo de respuesta y sobre impulso, sin embargo debido a las características del controlador, se puede observar la inestabilidad que genera el controlador debido a su frecuencia baja de muestreo, por lo cual se modela en el

espacio continuo un retardo y se halla nuevamente las ganancias, mejorando las respuestas de sobre impulso y estabilidad. La tarea fundamental del control de corriente es la protección de motores, por lo cual prima disminuir el efecto del sobre impulso de corriente.

#### **4.2.2 PI Calculado Vs PI Hallado por el software**

Debido a que en ambas situaciones (PI calculado analíticamente y por software) el tiempo de respuesta del lazo interno (PI de corriente) es mucho menor que el tiempo de respuesta del lazo externo del control en cascada. Para los motores 1 y 2 las sintonizaciones halladas de forma teórica y experimental cumplen con las restricciones del lazo interno, siendo la sintonización calculada de forma teórica en este caso superior que la hallada por el software, ya que obtiene sobre impulsos menores.

#### **4.2.3 PID Calculado Vs PID Hallado por el software**

De la sintonización realizada experimentalmente se obtiene que pese al alto orden del sistema las reglas de sintonización del control PID Desacoplado de Ziegler y Nichols utilizando el segundo método son eficientes como se había supuesto en un inicio, sin embargo de acuerdo a las gráficas se puede observar que la respuesta en lazo cerrado de la sintonización del PID por iteración, empleando el software EPOS UserInterface es en este caso superior ya que sobre impulsos y tiempos de establecimiento menores.

#### **4.2.4 Retardo de Comunicación PC – EPOS24/1**

De las pruebas realizadas en los movimiento del brazo robot se encontró que debido al retardo de Matlab en la comunicación serial y la forma de comunicación Maestro – Esclavo, el robot presenta un retardo considerable (0.15segundos) en la orden de activación entre un motor generando un error en el seguimiento de la trayectoria. Por estas razones se propone la solución de agregar el retardo producido por la comunicación al algoritmo generación de trayectoria, para que las trayectorias de los eslabones sean coordinadas o isócronas.

### 4.3 Propuesta económica.

A continuación se muestra la tabla 19 la cual representa una propuesta económica aproximada del costo total del proyecto.

Tabla: 19 Costos del proyecto de la Estación de trabajo del robot de 3 grados de Libertad

	Unidades	Costo Unidad \$	Total \$
Construcción Brazo	1	2,000.00	2,000.00
Drivers	3	520.00	1,560.00
Motores	3	305.50	916.50
Reducción	3	161.20	483.60
Adaptador USB – Serial	1	30.00	30.00
Fuente	1	15.00	15.00
Encoders	3	130.00	390.00
Computadora Personal	1	1000.00	1000.00
Matlab-Simulink	1	1,250.00	1,250.00
		<b>Total</b>	<b>7645.10</b>
Horas Hombre	360	30.00	11800.00
		<b>Total del proyecto</b>	<b>18,445.10</b>

## CONCLUSIONES

- La sintonización analítica de los algoritmos de control PI cumple satisfactoriamente los requerimientos de movimiento por su aceptable tiempo de respuesta y menor sobre impulso que la sintonización realizada por el software EPOS UserInterface ante una entrada escalón. Este menor sobre impulso da como consecuencia que puedan utilizarse valores aceleración y desaceleración mayores (perfil de aceleración) sin que produzca errores en el controlador por sobre corriente.
- La sintonización experimental del PID realizada por el segundo método de las reglas de Ziegler y Nichols cumple satisfactoriamente los objetivos de control de movimiento, sin embargo no es tan eficiente como la hallada por el software EPOS UserInterface, ya que genera oscilaciones innecesarias, mayor tiempo de respuesta y mayor sobre impulso.
- Es factible la implementación de un algoritmo de generación de trayectoria para el control de los movimientos deseados por el robot, reflejando su correcto funcionamiento en el desplazamiento de las trayectorias isócronas entre puntos y en el seguimiento de las trayectorias circulares y línea recta.
- Los Errores de seguimiento producidos en los desplazamientos en línea recta y circunferencia son proporcionales al tamaño de los intervalos tomados en los recorridos (línea recta y circunferencia).
- Es factible la implementación una interface a través de una computadora personal utilizando el software Matlab para el control y visualización de los movimientos del brazo robot de tres grados de libertad construido por el CETAM

## OBSERVACIONES

- El Hardware de los controladores EPOS 24/1 no permite realizar un control de movimiento continuo, es decir, especificar puntos y velocidades de paso en el camino de un punto a otro.
- Para los movimientos Línea Recta y Circunferencia el controlador acelera y frena deteniendo su movimiento en cada punto de paso para efectuar la trayectoria.
- Las respuestas del brazo robot ante una fuente de 12 Voltios (Mitad de la Tensión de trabajo de los motores) es la adecuada si se trabaja con las

restricciones de velocidades máximas a la mitad de las velocidades nominales características de los motores.

- El modelamiento del retardo por parte del puerto serial en Matlab incluido en el algoritmo de generación de trayectorias, genera que los movimientos de los eslabones del robot finalicen al mismo tiempo.

## RECOMENDACIONES

- Para disminuir el tiempo de respuesta del movimiento de un punto a otro, utilizar una fuente de 24 voltios, que permite trabajar a velocidades nominal características de los motores.
- Intercambiar el motor del antebrazo o codo (Motor 2) por el motor del efector final (Motor 3) por:
  - El torque necesario en el Griper es mucho menor que el torque necesario en el Antebrazo.
  - La velocidad nominal del motor 3 es mayor que la del motor 2, por lo cual el intercambio permitiría traslaciones a mayor velocidad.
  - Debido a bajo torque requerido en Griper en comparación del con el Antebrazo, es factible disminuir la reducción de los engranajes para aumentar la velocidad.
- Adicionar los switches límites para efectuar de forma automática la movilización del robot a la posición de inicio además de la autocalibración del sistema de referencia.



**BIBLIOGRAFÍA**

- [1] FU, K. S., GONZALES, R. C., LEE, C. S.  
1988 *Robótica: Control, detección, visión e inteligencia*. Madrid: McGraw-Hill
- [2] Kelly, Rafael – Santibáñez, Victor  
2003 *Control de movimiento de Robots Manipuladores*. Madrid: Pearson Educación.
- [3] Kelly, Rafael.  
1995, *A tuning procedure for stable PID control of robot manipulators*,  
*Robótica*, Marzo-Abril, Vol 13, Parte 2, pp. 141-148.
- [4] C. C. Cheah, C. Liu and J.J.E. Slotine  
2005, *Adaptive Tracking Control for Robots with Unknown Kinematic and Dynamic Properties*,
- [5] G. M. Khoury, M. Saad, H. Y. Kanaan and C. Asmar  
2004, *Fuzzy PID Control of a Five DOF Robot Arm*  
*Journal Of Intelligent & Robotic Systems* Volume 40, Number 3, 299-320
- [6] [http://www.mathworks.com/mechatronics/?s\\_cid=HP\\_MI\\_tech\\_mechatronics](http://www.mathworks.com/mechatronics/?s_cid=HP_MI_tech_mechatronics)
- [7] OGATA, Katsuhiko.  
2003 *Ingeniería de control moderno*. Madrid: Pearson Educación.
- [8] Aníbal Ollero Baturon  
2001, *Robótica: Manipuladores y robots móviles*
- [9] <http://www.mathworks.com/products/simulink/?BB=1>
- [10] OGATA, Katsuhiko.  
1996 *Sistemas de control en tiempo discreto*. México, D.F.: Prentice Hall

- [11] Santos, Victor M. F.  
2004 *Robótica Industrial*. Departamento de Ing. Mecánica. Universidad de Aveiro
- [12] [http://www.mathworks.com/help/techdoc/learn\\_matlab/f0-14059.html](http://www.mathworks.com/help/techdoc/learn_matlab/f0-14059.html)
- [13] Antonio Barrientos, Luis Peñin, Carlos Balaguer, Rafael Aracil  
2007 *Fundamentos de Robótica*. Madrid: McGraw-Hill
- [14] Oscar Andrés Vivas Albán  
2006, *Control Predictivo de un Robot Tipo Scara*.  
Ingeniare - Revista Chilena de Ingeniería, vol. 14 N° 2, 2006, pp. 135-145
- [15] R.Saltarén, M.Almonacid, J.M.Azorín, J.M.Sabater  
ROBOTICA: MODELADO, SIMULACIÓN Y DISEÑO CON MATLAB.
- [16] Gene F. Franklin; J. David Powell; Abbar Emami-Naeini  
Feedback Control of Dynamics Systems, Fourth Edition; Prentice Hall
- [17] Hojas de Datos del EPOS24/1
- [18] Catálogo de precios de ABB Perú y Siemens Perú.
- [19] J.R. Hendershot Jr. and Tje Miller  
Design of Brushless Permanent-Magnet Motors

## APENDICE

- Anexo: 1 Tabla de cálculos de las ganancias PI y PID
- Anexo: 2 Programa en Matlab: Interface\_Usuario.m (Abre la Interface de Usuario)
- Anexo: 3 Programa en Matlab: cinematica\_directa.m
- Anexo: 4 Programa en Matlab: cinematica\_inversa.m
- Anexo: 5 Programa en Matlab: Dinam\_Euler\_Lagrange.m (Calculo de las Matrices para la Dinámica Directa e Inversa)
- Anexo: 6 Programa en Matlab: generación\_trayectoria.m
- Anexo: 7 Programa en Matlab: Plot\_mov\_BrazoB (Permite Visualizar los movimientos del brazo robot calculados en Simulink)
- Anexo: 8 Programa en Matlab: Calculo\_PI\_corriente2\_EPOS (Permite realizar la sintonización analítica del control PI)
- Anexo: 9 Video: Línea Recta.avi (simulación del seguimiento del brazo robot de la trayectoria en forma de línea recta)
- Anexo: 10 Video Punto a Punto.avi (simulación del seguimiento del brazo robot de la trayectoria punto a punto).
- Anexo: 11 Video Circunferencia.avi (simulación del seguimiento del brazo robot de la trayectoria en forma de circunferencia).
- Anexo: 12 Video: Demostración.mov (Demostración de la comunicación, control y visualización de los movimientos del brazo robot a través de la interface usuario)
- Anexo: 13 Simulaciones del sistema en Matlab – Simulink (Controles de corriente y posición, global y desacoplado; y modelos matemáticos del sistema)