



**CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS AVANZADOS
DEL INSTITUTO POLITÉCNICO NACIONAL
UNIDAD ZACATENCO**

**DEPARTAMENTO DE INGENIERÍA ELÉCTRICA
SECCIÓN DE MECATRÓNICA**

Planeación óptima de trayectorias en robots industriales

Tesis que presenta:

Ing. Azael Javindee López Valle

Para obtener el grado de:

Maestro en Ciencias

En la especialidad de:

Ingeniería Eléctrica

Directores de tesis:

Dr. Carlos Alberto Cruz Villar

Dr. Alejandro Rodríguez Ángeles

Ciudad de México

Diciembre, 2017

*Hay hombres que luchan un día y son buenos,
hay otros que luchan un año y son mejores,
hay quienes luchan muchos años y son muy buenos,
pero hay unos que luchan toda la vida, esos son los imprescindibles.*

Bertolt Brecht.

*Las personas no son recordadas por el número de veces que fracasan,
sino por el número de veces que tienen éxito.*

Thomas Alva Edison.

*Hay una fuerza motriz más poderosa que el vapor,
la electricidad y la energía atómica, la voluntad.*

Albert Einstein.

Agradecimientos

A mis padres

Por todo el apoyo brindado a lo largo de mi vida y en especial durante la realización de mi maestría.

A mi hermano

Por todo su apoyo, conocimientos y recomendaciones que me ha dado durante toda mi formación de estudio y sobre todo en la realización de mi maestría.

A Dulce

Por el apoyo en la revisión de este trabajo y el apoyo al final de mi maestría.

Al CINVESTAV

Por ser un grandioso centro de investigaciones, en el cual, sus instalaciones permitieron exitosamente la realización de mi proyecto de tesis y en especial a la Sección de Mecatrónica por brindarme el apoyo con sus laboratorios, donde colabore con estudiantes y profesores, que me ayudaron a mejorar y dar más en mi tesis de maestría.

Al CONACYT

Por proporcionarme una beca de tiempo completo, para poder realizar mis estudios de Maestría.

Al Doctor Carlos A. Cruz Villar y al Doctor Alejandro Rodríguez Angeles

Por aceptarme en este proyecto de tesis, por su apoyo, asesorías y revisiones de mi tesis, que sin duda alguna fueron un aporte invaluable no solo al desarrollo de mi tesis, sino también a mi formación académica y científica .

Dedicatoria

Dedico este trabajo de titulación a mi familia, mi papá, mi mamá y mi hermano.

Para ustedes papá y mamá que me dieron todo el apoyo incondicional para estudiar esta maestría, por estar conmigo siempre en los buenos y malos momentos, por soportarme y entenderme en todo momento. Gracias por todo el cariño que siempre me han dado y por ser el ejemplo que siempre me impulso a seguir adelante. Porque para mí son y serán los mejores padres que alguien puede tener, y como una muestra de infinito agradecimiento dedico este título a ustedes.

Resumen

En este trabajo se propone la planeación de trayectorias en robots industriales por medio de la implementación de un control híbrido posición/fuerza, este control permite obtener una trayectoria de posición considerando la interacción entre el ambiente y el efector final del robot, es decir se controla la fuerza de contacto además de la posición. Con la implementación de este control híbrido se buscó resolver el problema de implementar un control de fuerza en un robot industrial que no cuenta con un kit ad-hoc de fuerza, el cual vende el fabricante y permite controlar fuerza en el robot industrial con un aditamento de software y hardware. El control híbrido posición/fuerza fue programado en el lenguaje de programación RAPID que es el lenguaje que utilizan los robots industriales ABB, dentro de la programación se utilizaron las instrucciones del fabricante para utilizar su propio control de posición (control PID industrial), junto con la implementación del algoritmo de control PI de fuerza, con las condiciones de conmutación y retroalimentación de los controles se obtuvo el control híbrido, el cual recibe una señal de fuerza a través de una interfaz de comunicación entre el robot industrial y el sensor de fuerza la cual fue implementada con un microcontrolador y circuitos de acondicionamiento de la señal.

Se presentan simulaciones y resultados experimentales para dicho esquema de control, mostrando los casos de regulación y seguimiento de trayectoria en el control de posición y regulación en el control de fuerza utilizando un Robot Industrial IRB2400 ABB.

Palabras clave: Robot industrial, control híbrido, impedancia planeación de trayectoria.

Abstract

In this work, we propose trajectory planning of industrial robots through the implementation of a hybrid position/force control, which allows to control a position path considering the interaction between the environment and the robot end-effector, i.e. the contact force is controlled as well as the position. With the implementation of this hybrid control, it was tried to solve the implementing problem of a force control on an industrial robot that does not have an ad-hoc force kit provided by the manufacturer, thus allowing force control on the industrial robot with software and hardware addition. Hybrid position/force control was programmed in the programming language RAPID which is used by ABB industrial robots, within the control program the manufacturer's instructions were employed, in order to implement the build joint position control (industrial PID control). Along with the PI implementation force control algorithm. The switching conditions and feedback controls, the hybrid control was obtained. This receives a force signal through a communication interface between the industrial robot and the force sensor which was implemented with a microcontroller and signal conditioning circuits.

Simulations and experimental results are presented for this control scheme, showing the cases of regulations and trajectory tracking in the position control and regulation in the force control using an Industrial Robot IRB2400 ABB.

Keywords: Industrial robot, hybrid control, impedance, trajectory planning.

Índice general

Índice general	XI
Índice de figuras	XIII
Índice de tablas	XV
1. Introducción	1
1.1. Estado del arte	5
1.2. Motivación	11
1.3. Objetivos	11
1.3.1. Objetivo general	11
1.3.2. Objetivos particulares	12
1.4. Contribuciones	12
1.5. Organización de la tesis	12
2. Modelos Matemáticos	15
2.1. Modelo cinemático	16
2.1.1. Modelo cinemático directo de un robot de 3GDL	16
2.1.2. Modelo cinemático inverso de un robot de 3GDL	19
2.2. Modelo diferencial	21
2.2.1. Principio de trabajo virtual	21
2.2.2. Jacobiano analítico para un robot de 3GDL	22
2.3. Modelo dinámico	22
2.3.1. Modelo dinámico para robot planar de 3GDL	24
2.4. Modelo del ambiente	28
3. Control híbrido posición/fuerza	33
3.1. Control PI de fuerza	35
3.2. Control PID de posición del fabricante	36
3.3. Condición de conmutación	37
3.4. Implementación del control híbrido en el robot industrial	38
3.5. Control híbrido posición/fuerza de un robot planar de 3GDL	39

4. Implementación de la arquitectura de control de fuerza	53
4.1. Sensor de fuerza	54
4.2. Microcontrolador ATmega2560	57
4.3. Tarjeta de entradas y salidas digitales	58
4.4. Circuitos de acondicionamiento de las señales	58
5. Casos de estudio: Resultados experimentales	63
5.1. Caso 1: Regulación de posición y regulación de fuerza	68
5.2. Caso 2: Seguimiento de trayectoria de posición y regulación de fuerza	72
5.3. Caso 3: Regulación de posición y seguimiento de fuerza	86
5.4. Conclusiones	92
6. Conclusiones y trabajo futuro	93
Trabajo futuro	94
Apéndice A	95
Apéndice B	109
Apéndice C	129
Bibliografía	133

Índice de figuras

1.1.	Robot IRB 6600 de la marca ABB realizando tareas pick and place [27]. . .	1
1.2.	Ensamblaje de un motor de automóvil con un robot industrial ABB [28]. . .	2
1.3.	Robot ABB realizando tareas de maquinado [28].	3
1.4.	Robot IRB2400 de la marca ABB con el que se realizaron los experimentos.	4
1.5.	Esquema de la tarea de control de fuerza.	5
1.6.	Interfaz de comunicación con el robot industrial IR7 ASEA [20].	8
1.7.	Esquema de comunicación con aplicación Win32 [19].	9
1.8.	Interfaz de comunicación con el robot industrial IRB140 ABB [16].	10
2.1.	Representación esquemática del robot IRB2400 ABB.	15
2.2.	Esquema del brazo robot planar de 3GDL.	17
2.3.	Configuraciones del robot de 3 GDL	19
2.4.	Triángulos formados para el método geométrico	19
2.5.	Robot de 3 GDL con centros de masa	24
2.6.	Circuito RLC con fuente de tensión variable	28
2.7.	Sistema masa-resorte-amortiguador	29
2.8.	Diagrama de enlace del modelo de impedancia	30
3.1.	Robot Industrial IRB2400 ABB con base de coordenadas	33
3.2.	Esquema de control híbrido posición/fuerza	34
3.3.	Diagrama de flujo de la condición de conmutación	37
3.4.	Esquema de control híbrido posición/fuerza para el robot industrial IRB 2400 ABB	38
3.5.	Modelo virtual del robot planar de 3GDL	39
3.6.	Esquema de control híbrido posición/fuerza en Simulink.	40
3.7.	Esquema de control PID en Simulink.	40
3.8.	Variables articulares para el caso de regulación con una superficie inclinada a 45°.	44
3.9.	Componentes de posición y fuerza para el caso de regulación con una su- perficie inclinada a 45°.	45
3.10.	Errores de posición y fuerza para el caso de regulación con una superficie inclinada a 45°.	45
3.11.	Pares de control para el caso de regulación con una superficie inclinada a 45°.	46

3.12. Señales de referencia para el caso de regulación con una superficie inclinada a 45°	46
3.13. Posiciones angulares para el seguimiento de trayectoria sobre el contorno de la superficie inclinada a 45°	47
3.14. Pares de control para el seguimiento de trayectoria sobre el contorno de la superficie inclinada a 45°	48
3.15. Componentes de posición y fuerza para el seguimiento de trayectoria sobre el contorno de la superficie inclinada a 45°	49
3.16. Errores de posición y fuerza para el seguimiento de trayectoria sobre el contorno de la superficie inclinada a 45°	49
3.17. Señales de referencia para el seguimiento de trayectoria sobre el contorno de la superficie inclinada a 45°	50
3.18. Señales comandadas para el seguimiento de trayectoria sobre el contorno de la superficie inclinada a 45°	50
3.19. Señal de conmutación para el seguimiento de trayectoria sobre el contorno de la superficie inclinada a 45°	51
4.1. Esquema de comunicación sensor de fuerza-robot.	53
4.2. Conexión del sensor de fuerza con controladora FT y Mux Box [29].	54
4.3. Disposición de pines analógicos de salida de la controladora FT [29].	55
4.4. Ensamble del sensor de fuerza.	55
4.5. Procedimiento para crear el TCP.	56
4.6. Método de los 4 puntos para crear el TCP [30].	57
4.7. Arduino Mega2560 [31].	57
4.8. Tarjeta de E/S digitales DSQC 652 de la unidad de control IRC5.	58
4.9. PCB del mapeo de la señal de fuerza.	59
4.10. Esquema del circuito de mapeo de la señal de fuerza para un canal del sensor.	59
4.11. PCB del circuito de amplificación.	60
4.12. Esquema de amplificación para un canal.	61
5.1. Robot industrial IRB 2400 ABB en estado inicial y en estado de interacción.	63
5.2. Interpolación de trayectoria desde la posición home del robot industrial IRB2400.	64
5.3. Programa principal del control híbrido posición/fuerza.	66
5.4. Rutina que implementa el controlador PI de fuerza.	67
5.5. Área de trabajo dentro de la mesa de acrílico.	67
5.6. Desplazamiento del robot industrial en el espacio cartesiano para el caso 1.	70
5.7. Graficas de posición y error de posición en el eje z para el caso 1.	70
5.8. Graficas de fuerza y error de fuerza en el eje z para el caso 1.	71
5.9. Comparación de la posición comandada y la posición actual para el caso 1.	71
5.10. Robot industrial IRB2400 ABB siguiendo una trayectoria recta.	73
5.11. Posición completa del robot industrial en el espacio cartesiano para el seguimiento de una trayectoria recta en el caso 2.	74

5.12. Graficas de posición y error de posición para el caso 2, siguiendo una trayectoria recta en posición.	75
5.13. Graficas de fuerza y error de fuerza para el caso 2, siguiendo una trayectoria recta en posición.	75
5.14. Seguimiento de una trayectoria recta en posición para el caso 2.	76
5.15. Comparación de la posición comandada con la posición actual para el caso 2, siguiendo una trayectoria recta de posición.	77
5.16. Posición completa del robot industrial en el espacio cartesiano para el seguimiento de una trayectoria recta con sentido contrario en el caso 2. . .	77
5.17. Graficas de posición y error de posición para el caso 2, siguiendo una trayectoria recta en posición con sentido contrario.	78
5.18. Comparación de la posición comandada con la posición actual para el caso 2, siguiendo una trayectoria recta de posición con sentido contrario.	78
5.19. Robot industrial IRB2400 ABB siguiendo una trayectoria circular.	79
5.20. Posición cartesiana del robot industrial siguiendo una trayectoria circular para el caso 2.	79
5.21. Graficas de posición y error de posición respecto al eje z siguiendo una trayectoria circular para el caso 2.	80
5.22. Graficas de fuerza y error de fuerza respecto al eje z siguiendo una trayectoria circular para el caso 2.	81
5.23. Seguimiento de una trayectoria circular en posición para el caso 2.	81
5.24. Comparación de la posición comandada con la posición actual para el caso 2 siguiendo una trayectoria circular en posición.	82
5.25. Robot industrial IRB2400 ABB siguiendo una trayectoria elíptica en posición.	82
5.26. Posición completa del robot industrial en el espacio cartesiano para el seguimiento de una trayectoria elíptica en posición para el caso 2.	83
5.27. Graficas de posición y error de posición respecto al eje z siguiendo una trayectoria elíptica para el caso 2.	84
5.28. Graficas de fuerza y error de fuerza respecto al eje z siguiendo una trayectoria elíptica para el caso 2.	84
5.29. Seguimiento de una trayectoria elíptica en posición para el caso 2.	85
5.30. Comparación de la posición comandada con la posición actual para el caso 2, siguiendo una trayectoria elíptica en posición.	85
5.31. Graficas de posición y error de posición respecto al eje z siguiendo una trayectoria coseno en fuerza para el caso 3.	87
5.32. Graficas de fuerza y error de fuerza respecto al eje z siguiendo una trayectoria coseno en fuerza para el caso 3.	88
5.33. Seguimiento de una trayectoria coseno en fuerza para el caso 3.	88
5.34. Comparación de la posición comandada con la posición actual para el caso 3 siguiendo una trayectoria coseno en fuerza.	89
5.35. Graficas de posición y error de posición respecto al eje z siguiendo una trayectoria diente de sierra en fuerza para el caso 3.	90

5.36. Graficas de fuerza y error de fuerza respecto al eje z siguiendo una trayectoria diente de sierra en fuerza para el caso 3.	90
5.37. Seguimiento de una trayectoria diente de sierra en fuerza para el caso 3. . .	91
5.38. Comparación de la posición comandada con la posición actual para el caso 3 siguiendo una trayectoria diente de sierra en fuerza.	91

Índice de tablas

2.1. Parámetros de Denavit-Hartenberg para un robot de 3GDL.	17
3.1. Ganancias del control híbrido posición/fuerza para un robot de 3GDL. . .	42
3.2. Parámetros de simulación del robot de 3GDL.	43
4.1. Parámetros del TCP.	56
4.2. Parámetros de diseño del amplificador.	60
5.1. Variables deseadas para el caso de estudio 1.	69
5.2. Parámetros del experimento y ganancias del control PI para el caso 1. . . .	69
5.3. Parámetros de la ecuación de línea recta paramétrica.	72
5.4. Parámetros de la ecuación de circunferencia paramétrica.	73
5.5. Parámetros de la ecuación de elipse paramétrica.	73
5.6. Parámetros del controlador PI de fuerza para el seguimiento de una línea recta en posición par el caso 2.	74
5.7. Parámetros del controlador PI de fuerza para el seguimiento de una circun- ferencia en posición.	80
5.8. Parámetros del controlador PI de fuerza para el seguimiento de elipse en posición.	83
5.9. Parámetros del controlador PI de fuerza para el seguimiento de un coseno de fuerza.	87
5.10. Parámetros del controlador PI de fuerza para el seguimiento de un diente de sierra de fuerza.	89

Capítulo 1

Introducción

Hoy en día los robots industriales de distintas marcas, son utilizados en su mayoría para tareas donde se controla solamente la posición, como son las tareas de pick and place (véase Figura 1.1) las cuales son implementadas en fábricas que requieren mover y posicionar objetos o piezas que se transportan en bandas, para ser procesadas por distintas máquinas industriales como tornos, fresadoras, cortadoras o simplemente son colocadas dentro de un almacén.



Figura 1.1: Robot IRB 6600 de la marca ABB realizando tareas pick and place [27].

En la Figura 1.1 se presenta un claro ejemplo de una tarea pick and place realizada por un robot IRB 6600 de la marca ABB con una capacidad de carga útil de 200Kg. La tarea que realiza este robot consiste en recoger de una banda transportadora garrafones de agua para después colocarlos en una planta embotelladora de la compañía Oasis Water Company en Dubái.

Sin embargo estos robots industriales pueden realizar tareas donde no solo se controla la posición sino también la fuerza, lo que genera la posibilidad de tener una relación segura entre el efector final del robot y un determinado ambiente, esto se puede ver en tareas donde el robot industrial se utiliza para el maquinado y desbastado de piezas de aluminio (véase Figura 1.3), de acero u otro metal maquinable, tareas de ensamblaje de piezas, para unir una pieza con otra en una fábrica armadora, por ejemplo, el ensamble correcto de un monoblock del motor de un automóvil, como se observa en la Figura 1.2.



Figura 1.2: Ensamblaje de un motor de automóvil con un robot industrial ABB [28].

Estas tareas son complicadas, ya que se debe cargar una pieza con cierta fuerza específica que no es fácilmente manipulable por una persona, además de que la unión de piezas necesita un elevado grado de precisión en posición y fuerza, de lo contrario, una fuerza excesiva dañaría la parte a ensamblar [6]. Los robots industriales en su configuración estándar o básica no incluyen los recursos necesarios para realizar control de fuerza, que permita realizar tareas como las mencionadas anteriormente. Lo que ofrecen los fabricantes es un kit extra para dotar al robot de las prestaciones para realizar control de fuerza, en estos kits el fabricante provee software y hardware que se instalan en el robot, un ejemplo de ello es la marca ABB que ofrece su kit llamado Integrated Force Control para control de fuerza [17]. El problema principal es el elevado costo que tiene el adquirir este equipamiento extra. Por ello en este trabajo de tesis se propone implementar un control híbrido posición/fuerza que con ayuda de hardware genérico y a menor costo, pueda realizar tareas de control de posición y fuerza con el robot industrial, sin utilizar el kit de fuerza que venden los fabricantes.

En la Figura 1.3 se observa un ejemplo de aplicación donde un robot industrial de la marca ABB controla la fuerza necesaria para maquinarse una pieza ensamblada en su efector final, esto con ayuda de un sensor de fuerza ensamblado entre el porta herramientas y el efector final. La tarea que realiza este robot es un devastado de una pieza de aluminio con un cortador vertical ensamblado en un soporte externo al robot.

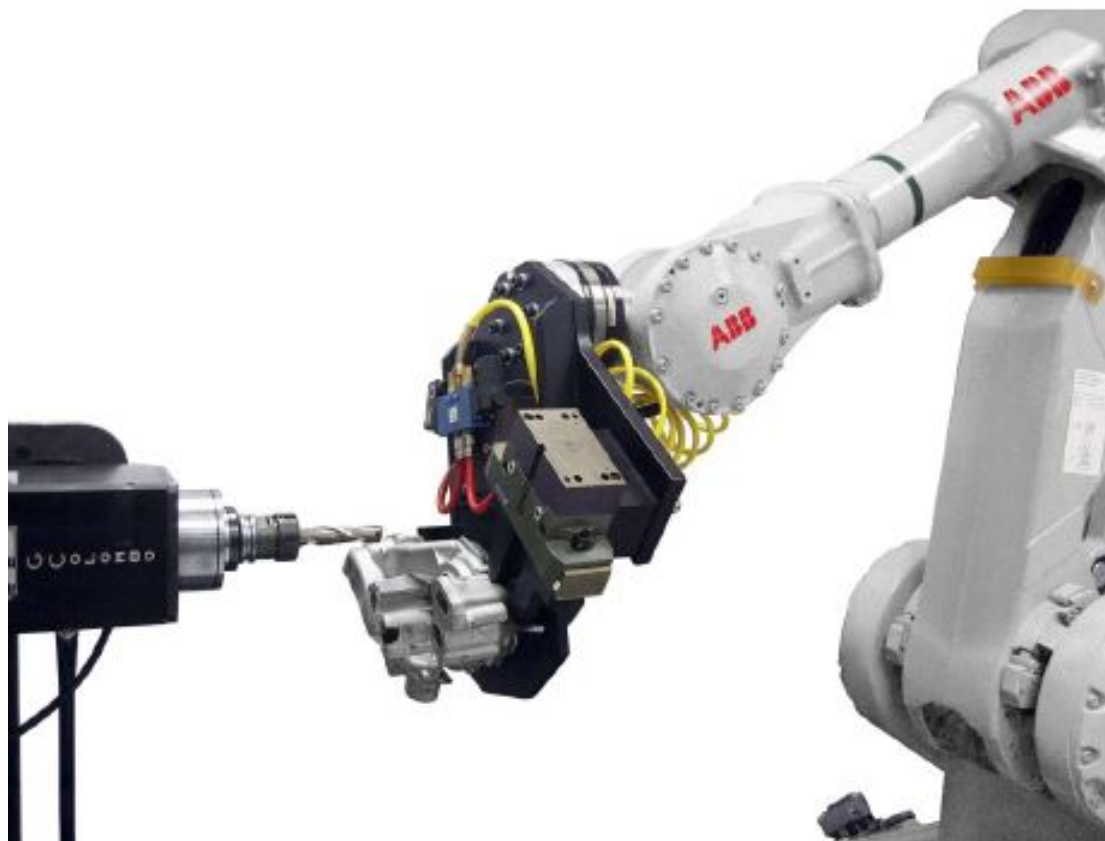


Figura 1.3: Robot ABB realizando tareas de maquinado [28].

En este trabajo se realizó la implementación de un control híbrido de posición/fuerza en un robot industrial IRB 2400 (véase Figura 1.4) de la marca ABB, junto con su software de programación y simulación ya incluido llamado RobotStudio, el cual se puede programar en el lenguaje de programación RAPID de ABB que es un lenguaje de programación basado en C y C++. Dentro de la implementación del control que se realizó, la parte de hardware está constituida por un sensor de fuerza mini40 de la marca ATI, junto con su controladora, una etapa de adquisición de datos, una etapa de conversión de la señal analógica a digital por medio del convertidor interno de un microcontrolador AVR y una etapa de amplificación de la señal de salida digital que se conecta a la unidad de control IRC5 ABB del robot industrial.



Figura 1.4: Robot IRB2400 de la marca ABB con el que se realizaron los experimentos.

En cuanto al software se utilizó el lenguaje de programación RAPID para programar el algoritmo de control, este programa se desarrolla en una computadora y después se carga a la controladora del robot, por medio de una conexión ethernet. Dicho algoritmo de control fue un control híbrido posición/fuerza, en el cual se utilizaron las instrucciones predefinidas en RAPID para configurar el control articular de posición interno del robot industrial, más la integración del algoritmo de control PI de fuerza que está basado en restricciones de fuerza y posición debido a la interacción con un ambiente determinado [4]. Este esquema de control híbrido implementado resultó de la modificación del esquema de control planteado en [1] y [5] con la diferencia de que, en lugar de usar un control óptimo de admitancia para controlar la parte de fuerza, se utilizó un control PI de fuerza que brinda a su salida la señal de referencia de posición al control de posición articular del robot.

La implementación se valida mediante resultados experimentales donde el efector final del robot interactúa con una superficie de acrílico, aplicando una fuerza deseada en el punto de contacto deseado como se muestra en la Figura 1.5. Ya que el control híbrido es un control de 2GDL (grados de libertad), siendo los grados de libertad el número de coordenadas (n) del sistema dinámico, menos el número de restricciones (m) entre coordenadas, esto es, $GDL = n - m$, donde las variables a controlar son posición y fuerza,

se plantean 2 casos de control y sus posibles combinaciones para ser validados de forma experimental. El caso 1 es donde se tiene control de regulación ya sea en posición o en fuerza y el caso 2 donde se tiene control de seguimiento de trayectoria en posición o en fuerza.

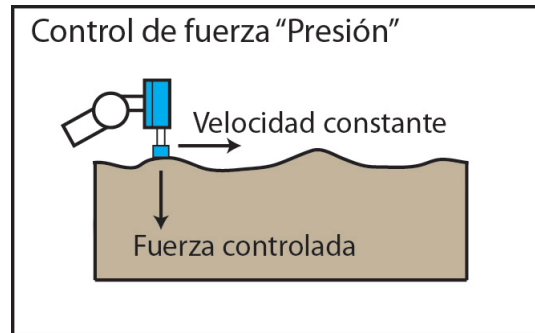


Figura 1.5: Esquema de la tarea de control de fuerza.

1.1. Estado del arte

Hoy en día gran parte de los robots manipuladores en el mundo son robots industriales, los cuales encontramos comúnmente en grandes fábricas, armadoras de automóviles o en empresas que fabrican piezas mecánicas para distintas aplicaciones. Un problema que se presenta a menudo en este tipo de robots es querer utilizarlos para distintas tareas donde sea necesario controlar la fuerza de contacto que se presenta en el efector final del robot, ya que en su configuración estándar no cuenta con esta capacidad, estas tareas pueden ser sencillas desde simplemente interactuar con alguna superficie (ambiente) y mantener cierta fuerza deseada en el contacto, hasta realizar tareas complicadas de ensamblaje y manipulación de objetos, estos tipos de ensambles o manipulación se hacen a través de distintos efectores finales que se encuentran en la parte terminal del robot (órgano terminal), para una aplicación específica.

El problema de controlar la posición y orientación del órgano terminal de un robot industrial ya se encuentra resuelto y muy estudiado pero, cuando el robot no solo tiene que controlar posición sino también la fuerza de contacto se vuelve un desafío importante, ya que para lograr esto es necesario que el controlador del robot permita modificar trayectorias cartesianas en tiempo real y los robots industriales más utilizados hoy en día tienen esta capacidad sin embargo, requieren una unidad adicional cuyo costo suele ser alto.

Una de las grandes limitaciones que existen en este tipo de robots manipuladores es que cuentan con una estructura predefinida de esquema de control, la cual no se puede cambiar de manera que se controle directamente el par de los actuadores, por ello se busca implementar a través del esquema de control existente en el robot, un control híbrido que controle posición y fuerza al mismo tiempo.

Para lograr que un robot manipulador pueda interactuar con un ambiente externo se han desarrollado distintas teorías de control, empezando por el control de impedancia propuesto por Neville Hogan [2]. En sus primeros trabajos mostró como controlar la interacción del órgano terminal de un robot con una superficie de contacto, introduciendo un modelo de impedancia deseada que describa y regule la interacción dinámica en el punto de contacto, por ello el control de impedancia se basa en regular una fuerza de contacto con el ambiente para asegurar una interacción deseada (impedancia deseada) [3]. A partir del planteamiento del control de impedancia surgieron distintos esquemas de control de fuerza, como el control por rigidez, control de fuerza pura, control de posición basado en impedancia y control híbrido de posición-fuerza.

El control por rigidez controla la interacción que existe entre el órgano terminal y el ambiente utilizando un control basado solo en una ganancia proporcional (P) [12], un tipo de control de fuerza está basado por lo general en modelos de control PI [14], ya que la medición de la fuerza genera ruido y es por ello que si se agrega una parte derivativa al controlador aumentaría este error, otro control de fuerza es utilizando un término proporcional con compensación ya sea por gravedad o por compensación de fuerza de Coriolis y centrífugas lo que daría como resultado controladores GPI de fuerza [13], sin embargo estos últimos tienen la desventaja de que, al no contar con el modelo dinámico exacto, no funcionarán con exactitud o no tendrán buenas prestaciones. Después surgió el control de posición basado en impedancia o admitancia [15], el cual tiene un doble lazo de control que trabaja de manera paralela ya que controla posición y regula la interacción física con el ambiente, lo que permite al robot seguir una trayectoria de posición y al mismo tiempo tener una interacción física deseada con el ambiente externo, pero cabe resaltar que en este tipo de controles no se controla la fuerza solo la posición, por ello surge el control híbrido de posición/fuerza [4], en el cual su esquema tiene un lazo interno que controla la posición, el cual puede ser un PID o distintas técnicas de control robusto como el control por regímenes deslizantes y un lazo de control externo el cual debe controlar la fuerza deseada que se establece, esto gracias a que se monta un sensor de fuerza entre el robot y su efector final, el cual brindará una señal de fuerza de reacción y al tener una fuerza deseada se genera el error de fuerza para ser retroalimentado en el lazo externo de control, que por lo regular es un PI de fuerza [6], por el problema ya mencionado que genera el agregar una parte derivativa solo se considera un PI y no un PD o PID.

Cuando surgieron las ideas de controlar tanto fuerza como posición se empezaron a crear etapas de control en los robots, ya que una etapa puede ser de movimiento libre, otra cuando el robot está en el instante de contacto con el ambiente que es denominada etapa de impacto y por último la etapa de interacción con el ambiente donde se busca controlar una fuerza deseada y controlar un seguimiento de trayectoria de posición [8].

A continuación se resumen los esquemas de control de fuerza más utilizados por los autores

citados:

- Control por impedancia.
- Control por impedancia adaptable.
- Control de fuerza pura.
- Control de posición basado en impedancia o admitancia.
- Control híbrido de posición/fuerza.

Estos controles mencionados anteriormente son para enmarcar la evolución del control de fuerza para robots manipuladores, que no solo pueden ser robots industriales sino también para distintos tipos de robots como robots de aplicación, robots de investigación, robots médicos y robots domésticos, etc.

Parte de los inicios para dotar al robot industrial de un control de fuerza fueron basados en los trabajos de N. Hogan [25], donde propone un control de impedancia aplicado a los robots industriales para resolver el problema de las limitaciones físicas impuestas por la interacción dinámica que existe entre un robot y una superficie cuando entran en contacto, con ello se fue desarrollando un enfoque unificado para realizar tareas como evasión de obstáculos, movimientos cinemáticamente restringidos e interacción dinámica. Posteriormente a las bases que se brindaron en distintos trabajos de control de fuerza en robots industriales basados en impedancia como el de N. Hogan, surgieron controles adaptativos de impedancia [20], donde se aplicó un control de impedancia adaptativa para resolver el problema de tener parámetros desconocidos del entorno como su rigidez y posición exacta de la superficie de contacto, en este se muestra con resultados experimentales las prestaciones del control de impedancia adaptativa en un robot industrial IR7 de ASEA, por medio de una interfaz de comunicación implementada en una PC entre un sensor de fuerza JR3 ensamblado en el órgano terminal del robot y el driver del motor, lo que permitió implementar el esquema de control desde la PC y actuar directamente los motores del robot industrial, esta implementación se puede ver de forma ilustrativa en la Figura 1.6.

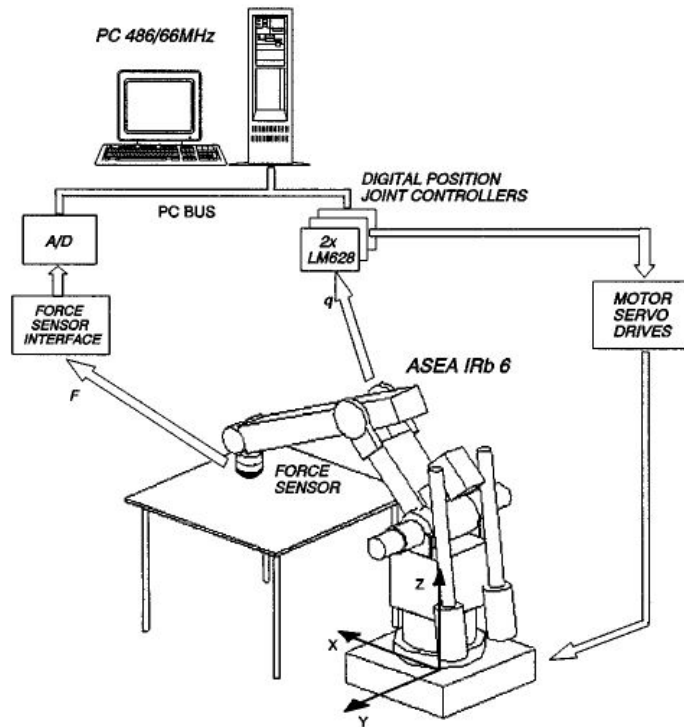


Figura 1.6: Interfaz de comunicación con el robot industrial IR7 ASEA [20].

Dentro de los trabajos donde se implantó un control de fuerza en un robot industrial a través de una PC, por ejemplo se han desarrollado trabajos en los cuales se utiliza el software Matlab para programar el esquema de control y realizar la adquisición de datos de fuerza para enviar a través del mismo software la señal de control al driver del motor [24].

En [20] y [24] se expone la idea de implantar un control de fuerza basado en impedancia en el robot industrial, debido a que en estos casos se tiene acceso al control de los pares del motor por medio de un driver, entonces se implanta una nueva arquitectura de control al robot industrial, a diferencia de estas referencias este trabajo de tesis tiene el objetivo no solo de implementar un control de fuerza en el robot industrial sino también de no modificar la arquitectura del control de posición que trae de fábrica el robot industrial, esto debido a las grandes prestaciones que tiene el control de posición de los robots industriales, además de reducir costos en la implementación, con ello se puede mencionar los siguientes trabajos más enfocados a la idea planteada en la tesis.

Con base a la hipótesis de la tesis y buscando trabajos más actuales se tiene el trabajo [21], donde se diseñó e implementó una plataforma para la integración rápida de sensores externos en un sistema de control de robots industriales (ABB S4CPlus), lo cual permitió el desarrollo de una aplicación basada en API Win 32, que corre en una PC, la cual tiene una tarjeta PCI conectada a un sensor de fuerza JR3, lo que permite tener una adquisición de datos del valor de fuerza con una razón de 10ms. Paralelamente desarrollaron un

servidor que corre en la unidad de control de un robot industrial ABB IRB2400, utilizando el lenguaje RAPID para programar el robot. La comunicación y la sincronización entre el servidor de la unidad de control y la aplicación en la PC que funge de cliente, fueron realizadas por medio un servidor RPC (Remote Procedure Calls) que brinda ABB para sus robots. Un esquema ilustrativo de la aplicación Win32 utilizada en estos trabajos se observa en la Figura 1.7.

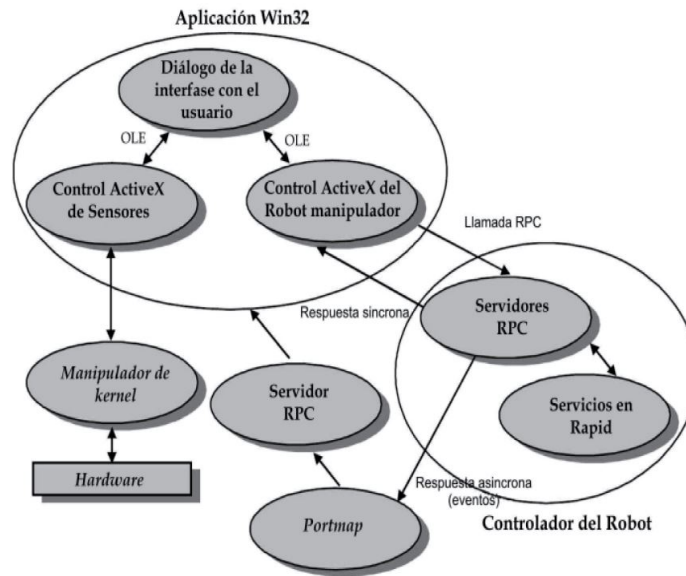


Figura 1.7: Esquema de comunicación con aplicación Win32 [19].

En [19], utilizan el mismo tipo de comunicación con un robot industrial como en la referencia anterior, pero proponen utilizar un control difuso para control de fuerza, específicamente un PI difuso de fuerza que pretende resolver el problema de parametrizar correctamente los valores de ganancia del control de fuerza.

Otra forma de solucionar el problema de comunicar el sensor de fuerza con algún dispositivo que procese la señal de fuerza con la intención de controlar fuerza en un robot industrial es por medio de la creación de software e instrucciones, para cambiar las posiciones y orientaciones del robot en tiempo real, y con ello modificar la trayectoria de referencia del control de posición del robot industrial para tener un control de fuerza [16]. Esto fue posible creando una interfaz de comunicación entre Matlab y RAPID de ABB, desarrollando un programa en RAPID consistente en una serie de procesos para acceder vía Ethernet desde Matlab mediante un protocolo RPC/RAP, que permite conectarse con el controlador del robot en tiempo real, además el sensor de fuerza que se utilizó tiene una controladora que permite comunicación serial con la unidad de control del robot industrial, el esquema de comunicación se puede visualizar en la Figura 1.8, sin embargo muchas veces este protocolo de comunicación se tiene que adquirir como un aditamento extra en el robot y con un costo elevado.

Para mejorar el rendimiento del control de fuerza en robots industriales, se utilizaron en conjunto dos tipos de robots comunicados, un dispositivo robótico de impedancia controlada (RCID) que como su nombre lo dice es un pequeño robot que se encarga del control de impedancia para modular la fuerza de contacto y el robot industrial el cual se utiliza para controlar la posición con su propio control de fábrica [23], el RCID se conecta en serie con el robo industrial y con ello se tiene el rendimiento dinámico de impedancia del RCID, así como el espacio de trabajo y ancho de banda del seguimiento de trayectoria del robot industrial, este trabajo resulto de la evolución de la adaptabilidad pasiva, ya que en un inicio para que el robot industrial pudiera actuar de una forma restringida con su ambiente pero sin controlar la fuerza, se instalaba un dispositivo de adaptabilidad de centro remoto (RCC) [26] y más que nada era utilizado en tareas de ensamblaje para corregir o evitar el atascamiento teniendo un control pasivo de fuerza.

En los últimos años se han estado proponiendo controladores más sofisticados para brindar un mejor desempeño en el control de fuerza, donde se utiliza un controlador híbrido posición /fuerza neuro-difuso adaptativo PD+I [22], sin embargo solo se muestra su respuesta en simulación en Matlab con un robot industrial PUMA-560.

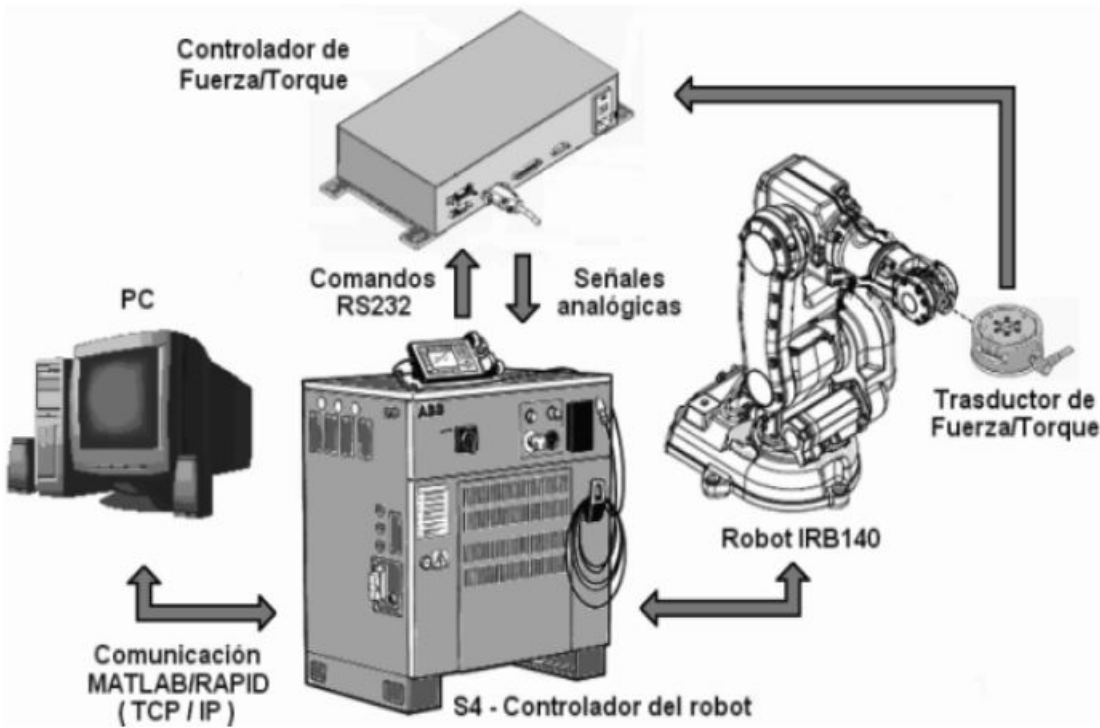


Figura 1.8: Interfaz de comunicación con el robot industrial IRB140 ABB [16].

1.2. Motivación

Hoy en día los robots industriales son usualmente utilizados para realizar tareas de posicionamiento conocidas como tareas de pick and place, donde se controla solamente la posición y para ello se tienen distintos controles de posición industriales, por lo general PIDs que tienen un buen desempeño, sin embargo con el avance de las fábricas y sobre todo en las grandes fábricas automotrices o de automatización se requieren robots industriales que interactúen con su entorno, por ejemplo al realizar tareas de agarre y colocación de piezas automotrices en el chasis de un carro. Es por ello que se buscó implementar un esquema de control donde se controle además de la posición, la fuerza de interacción entre el órgano terminal del robot industrial y su ambiente, para ser posible una interacción segura.

A pesar del avance en el control de los robots industriales, aún siendo de última generación pueden no contar con los recursos para realizar un control de fuerza, por ello los fabricantes ofrecen un kit de fuerza que provee de hardware y software al robot industrial para realizar tareas donde se desea un control de posición y fuerza, pero este kit tiene un costo elevado, lo cual motivó a realizar un control híbrido posición/fuerza que realice una planeación de trayectorias donde se controle de manera simultánea la fuerza y la posición para realizar tareas que lo requieran como el desplazamiento sobre una superficie manteniendo un contacto con una fuerza específica.

Por otro lado el control híbrido posición/fuerza propuesto tiene a diferencia de otros controles híbridos implementados en robots industriales, que este utiliza el propio control de posición industrial el cual es retroalimentado con una señal de referencia proveniente de un control PI de fuerza, lo que permite planear una trayectoria la cual sea segura en la interacción ya que se controla una fuerza específica que es permitida en la superficie de contacto en base a la rigidez del material.

Debido a los recursos estándar con los que cuenta un robot industrial y específicamente en la parte de comunicación, fue necesario implementar una interfaz de comunicación para comunicar el sensor de fuerza y el robot industrial debido a que los fabricantes venden de forma externa aditamentos de software para tener una comunicación con algún protocolo de comunicación, como una comunicación serial RS-232 o señales I/O digitales.

1.3. Objetivos

1.3.1. Objetivo general

Realizar una planeación de trayectoria a través de la implementación de un control híbrido posición/fuerza en robots industriales que permita una interacción segura con un determinado ambiente.

1.3.2. Objetivos particulares

- Estudio del estado de arte en control de fuerza de robots manipuladores.
- Implementación y programación del control híbrido posición/fuerza en robots industriales.
- Implementar una etapa de comunicación entre el sensor de fuerza y el robot industrial.
- Proponer trayectorias que se controlen en tiempo real donde el robot industrial interactúe con su entorno.
- Pruebas experimentales del control híbrido para el caso de regulación en posición y regulación en fuerza y para el caso de seguimiento de trayectoria en posición y seguimiento de trayectoria de fuerza.

1.4. Contribuciones

Las contribuciones de esta tesis son las siguientes:

- Planteamiento del problema de controlar la fuerza en un robot industrial sin utilizar el kit de control de fuerza del fabricante.
- Solución a la planeación de trayectorias de posición donde el robot industrial interactúe con su ambiente por medio de un control híbrido posición/fuerza.
- Desarrollo de interfaz de comunicación entre el sensor de fuerza y el robot industrial.

1.5. Organización de la tesis

La tesis está organizada de la siguiente forma:

En el primer capítulo se presenta la introducción general de la tesis, así como un conjunto de antecedentes expuestos en forma de estado del arte para dar a conocer los trabajos realizados antes del presente, y que fueron estudiados para dar base y principio a esta misma. También se expone la motivación del trabajo, junto con los objetivos generales y parciales que se cumplieron a lo largo de la tesis.

En el capítulo 2 se presentan los modelos cinemático y dinámico del robot industrial expresados de forma general, una deducción completa de los mismos modelos para un robot planar de 3GDL y el modelo del ambiente representado como una impedancia para describir la interacción del robot con su ambiente cuando existe contacto entre los mismos.

En el capítulo 3 se plantea el problema de planeación de trayectoria por medio de un control híbrido posición/fuerza para realizar una tarea, donde se controle simultáneamente posición y fuerza, para ello se desarrolló el esquema general de control híbrido basado en algoritmos PID, con un ejemplo de aplicación en un robot planar de 3GDL. Además se realizó el planteamiento en el robot industrial de 6GDL utilizando el propio control de posición del robot. En el capítulo 4 se presenta la interfaz de comunicación entre el sensor de fuerza y el robot industrial. Esto se llevó a cabo por medio de un microcontrolador y circuitos de acondicionamiento de señal, donde el sensor de fuerza se conecta al microcontrolador para convertir la señal analógica a digital y este a su vez con la unidad de control del robot industrial por medio de una etapa de amplificación de señales digitales provenientes del microcontrolador. En el capítulo 5 se presentan los resultados experimentales obtenidos al programar el esquema de control en el lenguaje RAPID de ABB, descargándolo en la unidad de control del robot industrial y teniendo conectado el sensor de fuerza que permite tener en tiempo real la fuerza de reacción en un contacto puntual a la superficie de contacto, estos resultados experimentales se muestran en forma gráfica y son de las variables a controlar, sus errores y variables derivadas de estas. En el capítulo 6 se exponen las conclusiones de la tesis basadas en los resultados obtenidos y se propone el trabajo futuro como continuación del presente trabajo.

Capítulo 2

Modelos Matemáticos

En este capítulo se presentan los modelos generales cinemático y dinámico para el robot industrial, así como el de un robot de 3 GDL de libertad para ejemplificar el control de posición en el siguiente capítulo y el modelo del ambiente o impedancia. La solución que se propuso fue utilizar el propio control de posición del robot industrial más la introducción de un control de fuerza que permite tener un control híbrido posición/fuerza, este control de posición industrial se realiza con instrucciones predefinidas en el lenguaje de programación RAPID de ABB. Estas instrucciones tienen un código fuente que contiene los modelos cinemáticos y dinámicos del robot industrial, es por ello que no es necesario deducir los modelos del mismo.

El robot industrial con el que se trabajó fue un IRB2400 de la marca ABB el cual podemos observar en la Figura 2.1 junto con su modelo esquemático que representa 6 grados de libertad (GDL) del robot, los primeros 3GDL representados por 3 variables articulares: q_1 , q_2 , q_3 que representan el movimiento de la base y de los dos eslabones o brazos, y los últimos 3GDL se encuentran localizados en la muñeca del robot y que representan la orientación del efector final como ángulos de Euler: q_4 , q_5 , q_6 .

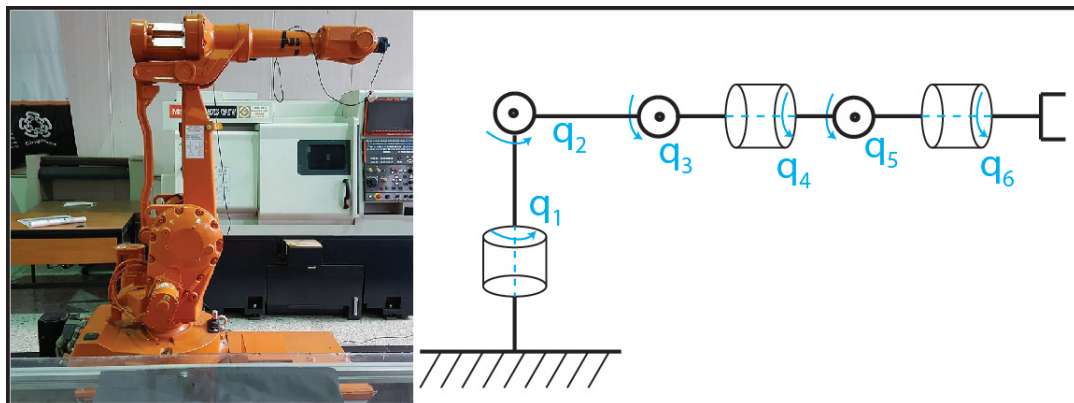


Figura 2.1: Representación esquemática del robot IRB2400 ABB.

El robot industrial tiene 6 articulaciones rotacionales que están esquematizadas en la Figura 2.1. Dentro del esquema la figura cilindro representa un movimiento rotacional alrededor del eje sobre su eslabón solidario, mientras que la articulación representada por un círculo y un punto indica una rotación sobre un eje perpendicular al eje del cilindro, el eje del círculo se encuentra saliendo de la hoja, estas variables angulares siguen la regla de la mano derecha como se representa con la flecha asignada a cada variable articular.

2.1. Modelo cinemático

El modelo cinemático se divide en 2 modelos: directo e inverso. El modelo cinemático directo es aquel donde es conocido el valor de las variables articulares $q \in R^n$ y se desea obtener las variables cartesianas y orientaciones $X \in R^m$ que adopta el extremo del robot que es expresado como,

$$X = F_{CD}(q) \quad (2.1)$$

La ecuación (2.1) representa el modelo cinemático directo del robot y tiene como variables independientes a las variables articulares.

q .

En el modelo cinemático inverso se conocen las variables cartesianas y la orientación del efector final del robot y a partir de ello se determinan las variables articulares, esto es presentado a continuación,

$$q = F_{CI}(X) \quad (2.2)$$

La ecuación (2.2) representa el modelo cinemático inverso del robot donde se tiene como variables independientes a las variable cartesianas y orientaciones del extremo del robot.

X .

X representan el movimiento x, y, z y orientación α, β, γ en angulos de Euler del efector final del robot.

A continuación se muestra el desarrollo para obtener los modelos cinemáticos para un robot planar de 3GDL, con 3 articulaciones de rotación (RRR). La cinemática directa se obtuvo por el método de Denavit-Hartenberg utilizando la convención propuesta en [8], y el modelo cinemático inverso fue obtenido por métodos geométricos utilizando identidades trigonométricas.

2.1.1. Modelo cinemático directo de un robot de 3GDL

En la Figura 2.2 podemos observar un esquema simplificado del robot a controlar, el cual es un brazo robot planar RRR de 3 grados de libertad los cuales son movimientos angulares referentes a cada eslabón.

Las variables articulares son:

$$q = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix}$$

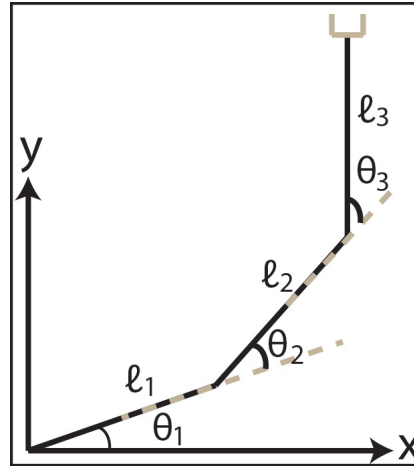


Figura 2.2: Esquema del brazo robot planar de 3GDL.

En base al algoritmo de Denavit-Hartenberg [8] se obtuvieron los parámetros de D-H mostrados en la Tabla 2.1.

i	θ_i	d_i	a_i	α_i
1	θ_1	0	l_1	0
2	θ_2	0	l_2	0
3	θ_3	0	l_3	0

Tabla 2.1: Parámetros de Denavit-Hartenberg para un robot de 3GDL.

En base a las 3 variables articulares del brazo robot obtenemos las matrices de transformación homogénea referentes a cada grado de libertad, para posteriormente multiplicarlas y obtener la matriz general de transformación homogénea que describe la cinemática directa del robot.

$$T = {}^0 A_1 {}^1 A_2 {}^2 A_3$$

$${}^0 A_1 = \begin{bmatrix} \cos(\theta_1) & -\sin(\theta_1) & 0 & l_1 \cos(\theta_1) \\ \sin(\theta_1) & \cos(\theta_1) & 0 & l_1 \sin(\theta_1) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^1A_2 = \begin{bmatrix} \text{Cos}(\theta_2) & -\text{Sin}(\theta_2) & 0 & l_2\text{Cos}(\theta_2) \\ \text{Sin}(\theta_2) & \text{Cos}(\theta_2) & 0 & l_2\text{Sin}(\theta_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2A_3 = \begin{bmatrix} \text{Cos}(\theta_3) & -\text{Sin}(\theta_3) & 0 & l_3\text{Cos}(\theta_3) \\ \text{Sin}(\theta_3) & \text{Cos}(\theta_3) & 0 & l_3\text{Sin}(\theta_3) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Para reducir las expresiones se utilizaron las siguientes notaciones

$$\begin{aligned} \text{Cos}(\theta_1) &= C_1, & \text{Sin}(\theta_1) &= S_1 \\ \text{Cos}(\theta_2) &= C_2, & \text{Sin}(\theta_2) &= S_2 \\ \text{Cos}(\theta_3) &= C_3, & \text{Sin}(\theta_3) &= S_3 \\ \text{Cos}(\theta_1 + \theta_2) &= C_{12}, & \text{Sin}(\theta_1 + \theta_2) &= S_{12} \\ \text{Cos}(\theta_1 + \theta_2 + \theta_3) &= C_{123}, & \text{Sin}(\theta_1 + \theta_2 + \theta_3) &= S_{123} \end{aligned}$$

La matriz de transformación homogénea obtenida fue la siguiente

$$T = \left[\begin{array}{ccc|c} C_{123} & -S_{123} & 0 & l_2C_{12} + l_1C_1 + l_3C_{123} \\ S_{123} & C_{123} & 0 & l_2S_{12} + l_1S_1 + l_3S_{123} \\ 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 \end{array} \right]$$

De la matriz de transformación homogénea T se obtuvieron las siguientes componentes de posición y orientación que describen por completo la cinemática directa del robot.

Posiciones del efector final

$$\begin{aligned} x &= l_1C_1 + l_2C_{12} + l_3C_{123} \\ y &= l_1S_1 + l_2S_{12} + l_3S_{123} \\ z &= 0 \end{aligned}$$

La posición z se considera igual a cero ya que los tres grados de libertad del robot solo cambian en el plano $x - y$.

Orientaciones del efector final

$$\begin{aligned} n &= [C_{123} \quad S_{123} \quad 0]^T \\ o &= [-S_{123} \quad C_{123} \quad 0]^T \\ a &= [0 \quad 0 \quad 1]^T \end{aligned}$$

Los vectores n , o y a definen una terna ortonormal y representan la orientación del efector final del robot en función de las coordenadas articulares.

- a : vector en la dirección de aproximación del efector final del robot a su objetivo.
- o : vector perpendicular a a en el plano definido por el efector final del robot.
- n : vector que forma una terna ortonormal con los dos vectores anteriores.

2.1.2. Modelo cinemático inverso de un robot de 3GDL

La solución a la cinemática inversa del robot de 3GDL se realizó con el método geométrico, el cual puede presentar dos configuraciones en su solución como se muestra en la Figura 2.3.

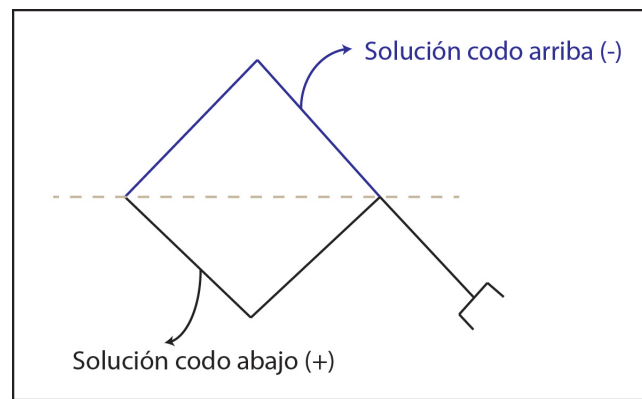


Figura 2.3: Configuraciones del robot de 3 GDL

Como el robot es planar se tienen como variables de entrada las posiciones x , y del efector final en el espacio cartesiano y la orientación del efector final quedará definida por un solo ángulo (θ), con ello podremos obtener las 3 variables articuladas del robot a través de la cinemática inversa y su deducción se realizó con el esquema de la Figura 2.4.

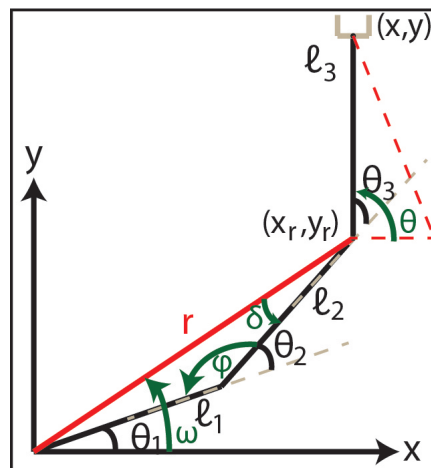


Figura 2.4: Triángulos formados para el método geométrico

Se observa en la Figura 2.4 que el ángulo de la orientación del efector final (θ) es igual a la suma de las variables articulares del robot.

$$\theta = \theta_1 + \theta_2 + \theta_3$$

En la Figura 2.4 se visualizan los triángulos formados para obtener ángulos auxiliares que permiten tener las variables angulares en función de las posiciones del efector final (x, y) y su orientación (θ).

$$x_r = x - l_3 \text{Cos}(\theta)$$

$$y_r = y - l_3 \text{Sin}(\theta)$$

$$r^2 = x_r^2 + y_r^2$$

$$\varphi = \pi - \theta_2$$

$$\text{Cos}(\varphi) = \text{Cos}(\pi - \theta_2) = -\text{Cos}(\theta_2) = \frac{l_1^2 + l_2^2 - r^2}{2l_1l_2}$$

$$\text{Cos}(\theta_2) = C_2 = \frac{r^2 - l_1^2 - l_2^2}{2l_1l_2}$$

Aplicando las siguientes identidades trigonométricas

$$\text{Sin}^2(\theta_2) + \text{Cos}^2(\theta_2) = 1$$

$$\text{Sin}^2(\theta_2) = S_2^2 = 1 - C_2^2$$

$$\text{Tan}(\theta_2) = \frac{S_2}{C_2} = \frac{\pm\sqrt{1 - C_2^2}}{C_2}$$

$$\theta_2 = \text{atan2}(\pm\sqrt{1 - C_2^2}, C_2)$$

Ahora se obtiene el ángulo θ_1 que es la diferencia de w y el ángulo formado por el primer eslabón y la distancia r .

$$\text{Tan}(w - \theta_1) = \frac{l_2 S_2}{l_1 + l_2 C_2}$$

$$\theta_1 = w - \text{Tan}\left(\frac{l_2 S_2}{l_1 + l_2 C_2}\right)$$

$$w = \text{atan2}(y, x)$$

$$\theta_1 = \text{atan2}(y, x) - \text{atan2}(l_2 S_2, l_1 + l_2 C_2)$$

Finalmente se dedujeron las tres ecuaciones que describen la cinemática inversa para el robot de 3GDL

$$\theta_1 = \text{atan2}(y, x) \pm \text{atan2}(l_2 S_2, l_1 + l_2 C_2)$$

$$\theta_2 = \text{atan2}(\pm\sqrt{1 - C_2^2}, C_2)$$

$$\theta_3 = \theta - \theta_1 - \theta_2$$

Para la simulación se utilizó la solución codo abajo (+) como se observa en las Figuras 2.3 y 2.4.

2.2. Modelo diferencial

Una vez que se conoce el modelo cinemático directo se puede obtener el modelo diferencial, obteniendo el Jacobiano analítico $J(q) \in R^{m \times n}$ que relaciona las velocidades cartesianas \dot{X} y las velocidades articulares \dot{q} con fuerzas F y pares τ del sistema.

$$\dot{X} = J(q)\dot{q} \quad (2.3)$$

La ecuación (2.3) representa el modelo diferencial del robot.

El Jacobiano analítico es obtenido al derivar respecto a las variables articulares y está representado por,

$$J(q) = \frac{\partial F_{CD}(q)}{\partial q} \quad (2.4)$$

2.2.1. Principio de trabajo virtual

Utilizamos el principio de trabajo virtual para poder obtener una ecuación que relacione el par τ con las fuerza externas al sistema F . Este principio nos dice que si un cuerpo rígido se encuentra bajo la acción de fuerzas externas, los puntos donde se aplica dicha fuerza sufren un desplazamiento virtual, y estas fuerzas producen un trabajo virtual.

A partir del Jacobiano se obtiene un mapeo entre los desplazamientos infinitesimales en el espacio cartesiano δX , y los desplazamientos infinitesimales en el espacio articular δq , por medio de,

$$\delta X = J(q)\delta q \quad (2.5)$$

Definimos el trabajo virtual δw en la siguiente ecuación,

$$\delta w = F^T \delta x - \tau^T \delta q \quad (2.6)$$

Sustituimos la ecuación (2.5) en la ecuación (2.6) para obtener,

$$\delta w = (F^T J(q) - \tau^T) \delta q \quad (2.7)$$

El la ecuación (2.7), el trabajo virtual es igualado a cero cuando se está en equilibrio, esto es,

$$\tau = F^T J(q) \quad (2.8)$$

Ya que el control híbrido involucra control de fuerza y control de posición, entonces por medio del Jacobiano analítico se tiene la relación entre pares y fuerzas del sistema, esto es descrito por la ecuación (2.8).

2.2.2. Jacobiano analítico para un robot de 3GDL

Al utilizar la ecuación (2.3) obtenemos la matriz jacobiana para el robot planar de 3 GDL, para ello, la cinemática directa está dada por:

$$f_1 = x = l_1 C_1 + l_2 C_{12} + l_3 C_{123}$$

$$f_2 = y = l_1 S_1 + l_2 S_{12} + l_3 S_{123}$$

$$f_3 = z = 0$$

Se obtiene la matriz jacobiana de la siguiente forma

$$J(q) = \begin{bmatrix} J_{11} & J_{12} & J_{13} \\ J_{21} & J_{22} & J_{23} \\ J_{31} & J_{32} & J_{33} \end{bmatrix} = \begin{bmatrix} \frac{\partial f_1}{\partial q_1} & \frac{\partial f_1}{\partial q_2} & \frac{\partial f_1}{\partial q_3} \\ \frac{\partial f_2}{\partial q_1} & \frac{\partial f_2}{\partial q_2} & \frac{\partial f_2}{\partial q_3} \\ \frac{\partial f_3}{\partial q_1} & \frac{\partial f_3}{\partial q_2} & \frac{\partial f_3}{\partial q_3} \end{bmatrix}$$

$$J(q) = \begin{bmatrix} -l_1 S_1 - l_2 S_{12} - l_3 S_{123} & -l_2 S_{12} - l_3 S_{123} & -l_3 S_{123} \\ l_1 C_1 + l_2 C_{12} + l_3 C_{123} & l_2 C_{12} + l_3 C_{123} & l_3 C_{123} \\ 0 & 0 & 0 \end{bmatrix}$$

2.3. Modelo dinámico

Para obtener el modelo dinámico del robot planar de 3GDL se utilizó un método energético basado en la formulación de Euler-Lagrange [11], que se describe a continuación.

Empezamos por definir los grados de libertad (GDL) de un sistema, como el número de coordenadas (n), menos el número de restricciones entre coordenadas (m). Esto es:

$$GDL = n - m$$

Se define la forma restringida de la ecuación de Lagrange:

$$\frac{d}{dt} \left[\frac{\partial T(\dot{x}_1, \dot{x}_2, \dots, \dot{x}_n)}{\partial \dot{x}_k} \right] + \frac{\partial V(x_1, x_2, \dots, x_n)}{\partial x_k} = Q_i \quad (2.9)$$

$$\forall k = 1, 2, \dots, n$$

Si la interconexión de los elementos resultan m ecuaciones adicionales de restricción, entonces se tendrán $n + m$ ecuaciones con sólo n incógnitas, por lo que no todas las $n + m$ ecuaciones pueden ser independientes. Las funciones T y V , representan la energía cinética y potencial respectivamente del sistema y Q_i representa los pares y las fuerzas involucradas.

Para sistematizar esta formulación energética se define la siguiente función conocida como el Lagrangiano,

$$\mathcal{L}(\dot{q}, q, t) = T(\dot{q}, q, t) - V(\dot{q}, q, t) \quad (2.10)$$

$$\frac{d}{dt} \left[\frac{\partial \mathcal{L}(\dot{q}, q, t)}{\partial \dot{q}_i} \right] - \frac{\partial \mathcal{L}(\dot{q}, q, t)}{\partial q_i} + \frac{\partial \mathcal{F}(\dot{q})}{\partial \dot{q}_i} = Q_i \quad (2.11)$$

para $i = 1, 2, \dots, k$

En términos del Lagrangiano, tenemos la forma completa de la ecuación de Lagrange (2.11) considerando elementos disipativos, donde $\mathcal{F}(\dot{q})$ es la función de disipación de Rayleigh, para modelar los efectos disipativos que tenga el sistema.

La ecuación (2.11) permite modelar matemáticamente todo tipo de sistemas dinámicos no conservativos o conservativos cancelando la parte disipativa. Tomando en cuenta las ecuaciones (2.10) y (2.11), y siguiendo la metodología de Euler-Lagrange podemos deducir una expresión general para modelar la dinámica de un robot.

Siguiendo este método energético se define la energía cinética del sistema como:

$$T(\dot{q}, q) = \frac{1}{2} \dot{q}^T M(q) \dot{q}$$

La energía potencial viene dada por:

$$V(q) = \sum_{i=1}^n m_i g^T r_{ci}$$

Donde m_i es la masa de cada eslabón del robot, g^T el vector gravedad y r_{ci} son las distancias de cada centro de masa correspondiente a su eslabón i .

Estas expresiones de energía se sustituyen en la ecuación del Lagrangiano (2.10) y después se deriva el lagrangiano como lo indica la ecuación de Euler-Lagrange (2.11), de esta manera se tiene

$$\begin{aligned} \frac{d}{dt} (M(q) \dot{q}) - \frac{\partial T(\dot{q}, q)}{\partial q} + \frac{\partial V(q)}{\partial q} &= \tau_{ext} \\ M(q) \ddot{q} + \dot{M}(q) \dot{q} - \frac{\partial T(\dot{q}, q)}{\partial q} + \frac{\partial V(q)}{\partial q} &= \tau_{ext} \end{aligned}$$

Se define:

$$C(\dot{q}, q) \dot{q} \triangleq \dot{M}(q) \dot{q} - \frac{\partial T(\dot{q}, q)}{\partial q}, \quad G(q) \triangleq \frac{\partial V(q)}{\partial q}$$

La matriz $C(\dot{q}, q)$ es la matriz de Coriolis y fuerzas centrípetas, $G(q)$ es el vector de gravedad y τ_{ext} los pares externos involucrados. La matriz $C(\dot{q}, q) \dot{q}$ puede ser simplificada de la siguiente manera:

$$\begin{aligned} C(\dot{q}, q) \dot{q} &= \dot{M}(q) \dot{q} - \frac{\partial T(\dot{q}, q)}{\partial q} = \dot{M}(q) \dot{q} - \frac{1}{2} \dot{q}^T \frac{\partial M(q)}{\partial q} \dot{q} \\ C(\dot{q}, q) \dot{q} &= \frac{1}{2} \dot{q}^T \frac{\partial M(q)}{\partial q} \dot{q} \quad \text{donde} \quad \dot{M}(q) = \frac{\partial M(q)}{\partial q} \dot{q} = \dot{q}^T \frac{\partial M(q)}{\partial q} \end{aligned}$$

Una forma general de expresar y deducir la matriz $C(\dot{q}, q)$ es mediante el uso de los símbolos de Christoffel dados por la siguiente expresión:

$$C_{kj} = \frac{1}{2} \sum_{k=1}^n \left[\frac{\partial M_{kj}}{\partial q_i} + \frac{\partial M_{ki}}{\partial q_j} - \frac{\partial M_{ij}}{\partial q_k} \right] \dot{q}_k$$

Finalmente podemos deducir una ecuación general que represente el modelo dinámico de un robot manipulador y está expresada en su forma canónica por,

$$M(q)\ddot{q} + C(\dot{q}, q)\dot{q} + g(q) + D\dot{q} = \tau_{ext} + J^T F_R \quad (2.12)$$

En la ecuación (2.12), \ddot{q}, \dot{q}, q son la aceleración, velocidad y las posiciones articulares, respectivamente. La matriz $M(q) \in R^{n \times n}$ es la matriz de inercia generalizada, $C(q, \dot{q}) \in R^{n \times n}$ es la matriz de fuerzas centrífugas y de Coriolis, $D(q) \in R^{n \times n}$ es la matriz de disipación de energía descrita por la función de disipación de Rayleigh, $G(q) \in R^n$ es el vector que representa los efectos gravitacionales en el sistema, $\tau_{ext} \in R^n$ son los pares que se generan en cada articulación debido a su respectivo actuador y el Jacobiano transpuesto multiplicado por las fuerzas de reacción $F_R \in R^m$ que surgen en la interacción del robot y su ambiente.

2.3.1. Modelo dinámico para robot planar de 3GDL

Basado en la metodología de Euler-Lagrange descrita se deduce el modelo dinámico para un robot planar RRR de 3GDL con centros de masa localizados en el robot como se muestra en la Figura 2.5.

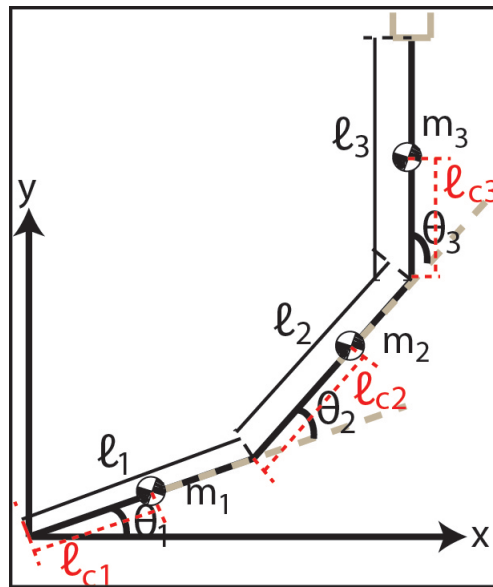


Figura 2.5: Robot de 3 GDL con centros de masa

Se definen las posiciones y velocidades de cada centro de masa correspondientes a cada eslabón del robot, para obtener su dinámica en función de sus variables generalizadas.

Posición del centro de masa del primer eslabón

$$\begin{aligned}x_{c1} &= l_{c1}S_1 \\y_{c1} &= -l_{c1}C_1\end{aligned}$$

Posición del centro de masa del segundo eslabón

$$\begin{aligned}x_{c2} &= l_1S_1 + l_{c2}S_{12} \\y_{c2} &= -l_1C_1 - l_{c2}C_{12}\end{aligned}$$

Posición del centro de masa del tercer eslabón

$$\begin{aligned}x_{c3} &= l_1S_1 + l_2S_{12} + l_{c3}S_{123} \\y_{c3} &= -l_1C_1 - l_2C_{12} - l_{c3}C_{123}\end{aligned}$$

Se deriva respecto al tiempo para obtener las velocidades en cada centro de masa

$$\begin{aligned}v_{c1} &= \begin{bmatrix} \frac{dx_{c1}}{dt} \\ \frac{dy_{c1}}{dt} \end{bmatrix} = \begin{bmatrix} l_{c1}C_1\dot{\theta}_1 \\ l_{c1}S_1\dot{\theta}_1 \end{bmatrix} \\v_{c2} &= \begin{bmatrix} \frac{dx_{c2}}{dt} \\ \frac{dy_{c2}}{dt} \end{bmatrix} = \begin{bmatrix} l_1C_1\dot{\theta}_1 + l_{c2}C_{12}(\dot{\theta}_1 + \dot{\theta}_2) \\ l_1S_1\dot{\theta}_1 + l_{c2}S_{12}(\dot{\theta}_1 + \dot{\theta}_2) \end{bmatrix} \\v_{c3} &= \begin{bmatrix} \frac{dx_{c3}}{dt} \\ \frac{dy_{c3}}{dt} \end{bmatrix} = \begin{bmatrix} l_1C_1\dot{\theta}_1 + l_2C_{12}(\dot{\theta}_1 + \dot{\theta}_2) + l_{c3}C_{123}(\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3) \\ l_1S_1\dot{\theta}_1 + l_2S_{12}(\dot{\theta}_1 + \dot{\theta}_2) + l_{c3}S_{123}(\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3) \end{bmatrix}\end{aligned}$$

Ya que la metodología para la obtención del modelo dinámico se basa en las energías del sistema, obtenemos la energía cinética y potencial total.

Energía cinética

$$T_T = T_1 + T_2 + T_3$$

Siendo I_1, I_2, I_3 los momentos de inercia referentes a cada centro de masa.

$$\begin{aligned}T_1 &= \frac{1}{2}m_1V_{c1}^TV_{c1} + \frac{1}{2}I_1\dot{\theta}_1^2 \\T_1 &= \frac{1}{2}m_1l_{c1}^2\dot{\theta}_1^2 + \frac{1}{2}I_1\dot{\theta}_1^2 \\T_2 &= \frac{1}{2}m_2V_{c2}^TV_{c2} + \frac{1}{2}I_2(\dot{\theta}_1 + \dot{\theta}_2)^2 \\T_2 &= \frac{1}{2}m_2l_1^2\dot{\theta}_1^2 + \frac{1}{2}m_2l_{c2}^2(\dot{\theta}_1 + \dot{\theta}_2)^2 + m_2l_1l_{c2}(\dot{\theta}_1^2 + \dot{\theta}_1\dot{\theta}_2)C_2 + \frac{1}{2}I_2(\dot{\theta}_1 + \dot{\theta}_2)^2\end{aligned}$$

$$\begin{aligned}
T_3 &= \frac{1}{2}m_3V_{c3}^TV_{c3} + \frac{1}{2}I_3(\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3)^2 \\
T_3 &= \frac{1}{2}m_3l_1^2\dot{\theta}_1^2 + \frac{1}{2}m_3l_2^2(\dot{\theta}_1 + \dot{\theta}_2)^2 + \frac{1}{2}m_3l_{c3}^2(\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3)^2 + \\
&\quad + m_3l_1l_2(\dot{\theta}_1^2 + \dot{\theta}_1\dot{\theta}_2)C_2 + m_3l_1l_{c3}(\dot{\theta}_1^2 + \dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_1\dot{\theta}_3)C_{12} \\
&\quad + m_3l_2l_{c3}(\dot{\theta}_1^2 + \dot{\theta}_2^2 + 2\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_1\dot{\theta}_3 + \dot{\theta}_2\dot{\theta}_3)C_3 + \frac{1}{2}I_3(\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3)^2
\end{aligned}$$

Energía potencial

$$\begin{aligned}
V_T &= V_1 + V_2 + V_3 = mgh \\
V_1 &= -m_1l_{c1}C_1 \\
V_2 &= -m_2l_1gC_1 - m_2l_{c2}gC_{12} \\
V_3 &= -m_3l_1gC_1 - m_3l_2gC_{12} - m_3l_{c3}gC_{123}
\end{aligned}$$

Obtenemos el Lagrangiano del sistema que se define como la diferencia de la energía cinética total y la energía potencial total de la planta.

$$\mathcal{L} = T_T(\theta, \dot{\theta}) - V_T(\theta)$$

$$\begin{aligned}
\mathcal{L} &= \frac{1}{2}m_1l_{c1}^2\dot{\theta}_1^2 + \frac{1}{2}I_1\dot{\theta}_1^2 + \frac{1}{2}m_2l_1^2\dot{\theta}_1^2 + \frac{1}{2}m_2l_{c2}^2(\dot{\theta}_1 + \dot{\theta}_2)^2 + m_2l_1l_{c2}(\dot{\theta}_1^2 + \dot{\theta}_1\dot{\theta}_2)C_2 + \frac{1}{2}I_2(\dot{\theta}_1 + \dot{\theta}_2)^2 + \\
&\quad + \frac{1}{2}m_3l_1^2\dot{\theta}_1^2 + \frac{1}{2}m_3l_2^2(\dot{\theta}_1 + \dot{\theta}_2)^2 + \frac{1}{2}m_3l_{c3}^2(\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3)^2 + m_3l_1l_2(\dot{\theta}_1^2 + \dot{\theta}_1\dot{\theta}_2)C_2 + m_3l_1l_{c3}(\dot{\theta}_1^2 + \dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_1\dot{\theta}_3)C_{12} \\
&\quad + m_3l_2l_{c3}(\dot{\theta}_1^2 + \dot{\theta}_2^2 + 2\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_1\dot{\theta}_3 + \dot{\theta}_2\dot{\theta}_3)C_3 + \frac{1}{2}I_3(\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3)^2 + m_1l_{c1}C_1 + m_2l_1gC_1 + m_2l_{c2}gC_{12} \\
&\quad + m_3l_1gC_1 + m_3l_2gC_{12} + m_3l_{c3}gC_{123}
\end{aligned}$$

Obtenemos las ecuaciones de Euler-Lagrange

$$\frac{d}{dt} \left[\frac{\partial \mathcal{L}}{\partial \dot{\theta}_1} \right] - \frac{\partial \mathcal{L}}{\partial \theta_1} + \frac{\partial D}{\partial \dot{\theta}_1} = \tau_1 \quad (2.13)$$

$$\frac{d}{dt} \left[\frac{\partial \mathcal{L}}{\partial \dot{\theta}_2} \right] - \frac{\partial \mathcal{L}}{\partial \theta_2} + \frac{\partial D}{\partial \dot{\theta}_2} = \tau_2 \quad (2.14)$$

$$\frac{d}{dt} \left[\frac{\partial \mathcal{L}}{\partial \dot{\theta}_3} \right] - \frac{\partial \mathcal{L}}{\partial \theta_3} + \frac{\partial D}{\partial \dot{\theta}_3} = \tau_3 \quad (2.15)$$

Reescribimos el modelo dinámico en su forma canónica (2.12), agregando la parte de disipación modelada por la función de disipación de Rayleigh.

Donde $\ddot{q}, \dot{q}, q = \ddot{\theta}, \dot{\theta}, \theta$ son la doble derivada de las variables articulares, la derivada de las variables articulares y las variables articulares, respectivamente. La matriz $M(q)$ es la matriz de inercias generalizada, $C(q, \dot{q})$ es la matriz de fuerzas centrífugas y de Coriolis,

$D(q)$ la matriz de disipación de energía y $G(q)$ el vector que representa la parte gravitacional en el sistema.

A partir de las ecuaciones (2.13), (2.14), (2.15) se reorganizan los términos para obtener una forma (2.12) y deducir las matrices siguientes:

Matriz de inercia generalizada

$$M(q) = \begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix}$$

$$M_{11} = m_1 l_{c1}^2 + I_1 + m_2 l_1^2 + m_2 l_{c2}^2 + 2m_2 l_1 l_{c2} C_2 + I_2 + m_3 l_1^2 +$$

$$+ m_3 l_2^2 + m_3 l_{c3}^2 + 2m_3 l_1 l_2 C_2 + 2m_3 l_2 l_{c3} C_3 + 2m_3 l_1 l_{c3} C_{12} + I_3$$

$$M_{12} = m_2 l_{c2}^2 + m_2 l_1 l_{c2} C_2 + m_3 l_2^2 + I_2 + m_3 l_{c3}^2 +$$

$$+ m_3 l_1 l_2 C_2 + m_3 l_1 l_{c3} C_{12} + 2m_3 l_2 l_{c3} C_3 + I_3$$

$$M_{13} = m_3 l_{c3}^3 + m_3 l_1 l_{c3} C_{23} + m_3 l_2 l_{c3} C_3 + I_3$$

$$M_{21} = M_{12}$$

$$M_{22} = m_2 l_{c2}^2 + I_2 + m_3 l_{c3}^2 + m_3 l_2^2 + 2m_3 l_2 l_{c3} C_3 + I_3$$

$$M_{23} = m_3 l_{c3}^2 + m_3 l_2 l_{c3} C_3 + I_3$$

$$M_{31} = M_{13}$$

$$M_{32} = M_{23}$$

$$M_{33} = m_3 l_{c3}^2 + I_3$$

Matriz de fuerzas centrífugas y de Coriolis

$$C(\dot{q}, q) = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix}$$

$$C_{11} = [-m_2 l_1 l_{c2} S_2 - m_3 l_1 l_2 S_2 - m_3 l_1 l_{c3} S_{23}] \dot{\theta}_2 + [-m_3 l_2 l_{c3} S_3 - m_3 l_1 l_{c3} S_{23}] \dot{\theta}_3$$

$$C_{12} = [-m_3 l_1 l_2 S_2 - m_2 l_1 l_{c2} S_2 - m_3 l_1 l_{c3} S_{23}] \dot{\theta}_1 + [-m_2 l_1 l_{c2} S_2 - m_3 l_1 l_2 S_2 - m_3 l_1 l_{c3} S_{23}] \dot{\theta}_2$$

$$+ [-m_3 l_2 l_{c3} S_3 - m_3 l_1 l_{c3} S_{23}] \dot{\theta}_3$$

$$C_{13} = [-m_3 l_2 l_{c3} S_3 - m_3 l_1 l_{c3} S_{23}] \dot{\theta}_1 + [-m_3 l_2 l_{c3} S_3 - m_3 l_1 l_{c3} S_{23}] \dot{\theta}_2 + [-m_3 l_2 l_{c3} S_3 - m_3 l_1 l_{c3} S_{23}] \dot{\theta}_3$$

$$C_{21} = [m_3 l_1 l_2 S_2 + m_2 l_1 l_{c2} S_2 + m_3 l_1 l_{c3} S_{23}] \dot{\theta}_1 + [-m_3 l_2 l_{c3}] \dot{\theta}_3$$

$$C_{22} = [-m_3 l_2 l_{c3} S_3] \dot{\theta}_3$$

$$C_{23} = [-m_3 l_2 l_{c3} S_3] \dot{\theta}_1 + [-m_3 l_2 l_{c3}] \dot{\theta}_2 + [-m_3 l_2 l_{c3}] \dot{\theta}_3$$

$$C_{31} = [m_3 l_2 l_{c3} S_3 + m_3 l_1 l_{c3} S_{23}] \dot{\theta}_1 + [m_3 l_2 l_{c3} S_3] \dot{\theta}_2$$

$$C_{32} = [m_3 l_2 l_{c3} S_3] \dot{\theta}_1 + [m_3 l_2 l_{c3} S_3] \dot{\theta}_2$$

$$C_{33} = 0$$

Vector gravedad

$$G(q) = \begin{bmatrix} G_1 \\ G_2 \\ G_3 \end{bmatrix}$$

$$G_1 = m_1 l_{c1} g S_1 + m_2 l_1 g S_1 + m_2 l_{c2} g S_{12} + m_3 l_1 g S_1 + m_3 l_2 g S_{12} + m_3 l_{c3} g S_{123}$$

$$G_2 = m_2 l_{c2} g S_{12} + m_3 l_2 g S_{12} + m_3 l_{c3} g S_{123}$$

$$G_3 = m_3 l_{c3} g S_{123}$$

Matriz de disipación de energía

$$D(q) = \begin{bmatrix} \mu_1 & 0 & 0 \\ 0 & \mu_2 & 0 \\ 0 & 0 & \mu_3 \end{bmatrix}$$

2.4. Modelo del ambiente

El modelo del ambiente representa la interacción entre el órgano terminal del robot y la superficie donde hace contacto y se modela como una impedancia, viendo este modelo desde el punto de vista eléctrico y su equivalencia al mecánico, que es en el cual el presente trabajo se enfocó para su utilización.

La impedancia (Z) es una medida de oposición a la corriente eléctrica en un circuito cuando se aplica una diferencia de potencial eléctrico en sus terminales. La impedancia está conformada por 3 elementos pasivos, un resistor con resistencia R , un condensador con capacitancia C y una bobina con inductancia L , conectados en serie con una fuente de tensión variable en el tiempo V y la corriente de malla $I(t)$, como se muestra en la Figura 2.6, en base a la ley de ohm tenemos que $Z = V/I$ y estas variables son cantidades fasoriales ya que varían con el tiempo.

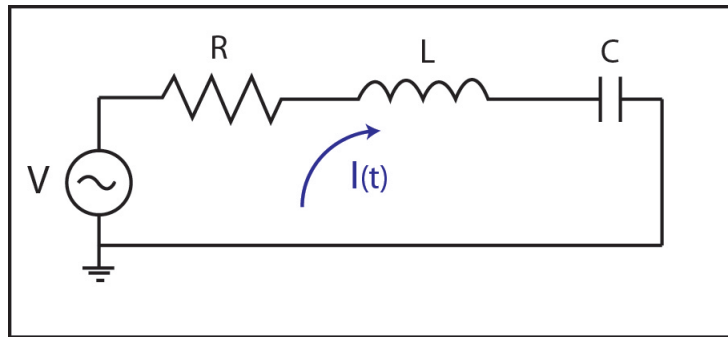


Figura 2.6: Circuito RLC con fuente de tensión variable

Para obtener una representación matemática de la impedancia se obtiene la ecuación integro-diferencial del circuito de la Figura 2.6, utilizando la ley de mallas de Kirchoff.

$$L \frac{dI(t)}{dt} + RI(t) + \frac{1}{C} \int_0^t I(\tau) d\tau = V \quad (2.16)$$

Expresamos la ecuación (2.16) en función de la carga eléctrica q ya que $I(t) = dq(t)/dt$

$$L \frac{d^2q(t)}{dt^2} + R \frac{dq(t)}{dt} + \frac{1}{C}q(t) = V \quad (2.17)$$

Reescribimos la ecuación (2.17) con una notación simplificada

$$L\ddot{q}(t) + R\dot{q}(t) + \frac{1}{C}q = V$$

Aplicando la transformada de Laplace a la ecuación (2.17) con condiciones iniciales cero, se obtiene,

$$\left(Ls^2 + Rs + \frac{1}{C} \right) q(s) = V(s) \quad (2.18)$$

A partir de la ecuación (2.18) se reescribe la ecuación que representa la impedancia en el dominio de Laplace,

$$Z(s) = Ls^2 + Rs + 1/C \quad (2.19)$$

Con de la ecuación (2.19) se deduce la analogía a un sistema mecánico, que es un sistema masa-resorte-amortiguador (véase Figura 2.7), con masa m , rigidez k y amortiguamiento b .

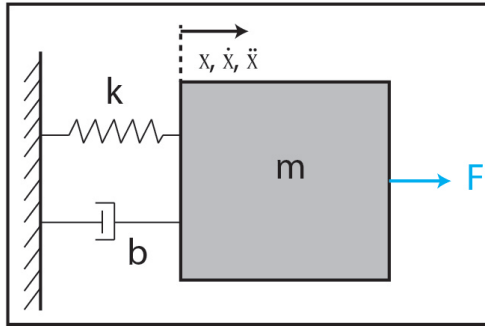


Figura 2.7: Sistema masa-resorte-amortiguador

$$m\ddot{X} + b\dot{X} + kX = F \quad (2.20)$$

El sistema de la Figura (2.7) está representado por la ecuación diferencial (2.20) equivalente a la ecuación diferencial (2.17).

Se aplica la transformada de Laplace a la ecuación (2.20) y se obtiene

$$(ms^2 + bs + k) X(s) = F(s) \quad (2.21)$$

De la ecuación (2.21) se deduce la impedancia en el dominio de Laplace, pero ahora para el sistema mecánico

$$Z(s) = ms^2 + bs + k \quad (2.22)$$

Se puede observar la equivalencia del termino impedancia en los sistemas eléctrico y mecánico con la deducción de las ecuaciones (2.22) y (2.19).

$$F_R = M(\ddot{X} - \ddot{X}_a) + B(\dot{X} - \dot{X}_a) + K(X - X_a) \quad (2.23)$$

La ecuación general que representa el modelo de impedancia es la ecuación (2.23) que considera la posición actual del robot X y la posición del ambiente X_a tal que $X - X_a$ representa penetración en el ambiente.

$$F_R = k(X - X_a) \quad (2.24)$$

Este modelo de impedancia completo es muy adecuado cuando se quiere realizar control por impedancia [25], las matrices M, B, K son matrices diagonales que representan la masa, amortiguamiento y rigidez respectivamente del modelo, sin embargo en este trabajo se utiliza para modelar el sensor de fuerza y obtener una fuerza de reacción, es por ello que se usa un ambiente capacitivo (2.24) el cual modela la interacción como un resorte y se considera una constante de rigidez k , adecuada para los materiales en contacto del efector final y de la superficie de contacto.

Cuando se tiene contacto con el ambiente es utilizada la expresión (2.24) para modelar la fuerza de reacción o la lectura del sensor de fuerza, en otro caso se considera un modelo libre de contacto con $F_R = 0$, este modelo se utiliza solamente en las simulaciones ya que en la implementación se instaló un sensor de fuerza en el efector final del robot para medir la fuerza de reacción mencionada.

De manera más general se esquematiza la impedancia con un diagrama de enlace que se observa en la Figura 2.8

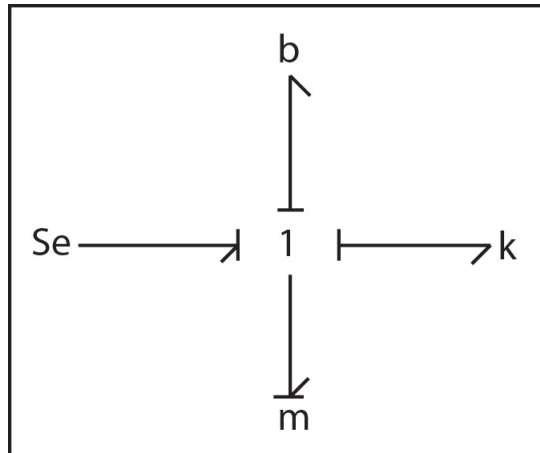


Figura 2.8: Diagrama de enlace del modelo de impedancia

El diagrama de enlace mostrado en la Figura 2.8 tiene como entrada una fuente de esfuerzo S_e la cual puede ser una fuerza como en el caso mecánico o una tensión en el caso eléctrico, esta fuente de esfuerzo va conectado con un elemento 1 que representa una conexión serie con los elementos generales b , m , k que son los mismos del sistema mecánico, pero que también pueden ser los elementos pasivos de un circuito eléctrico L , R , C .

Capítulo 3

Control híbrido posición/fuerza

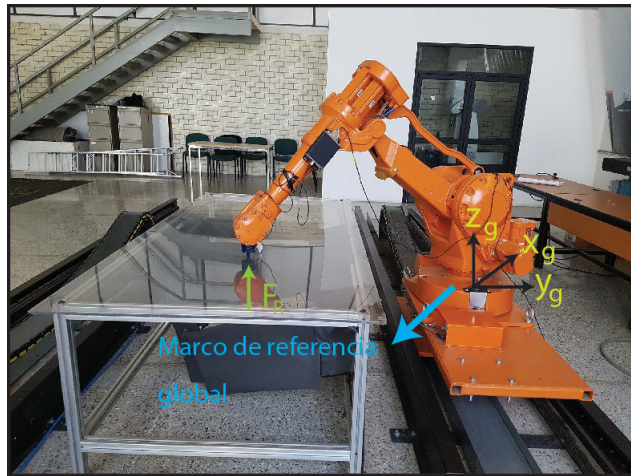


Figura 3.1: Robot Industrial IRB2400 ABB con base de coordenadas

La planeación de trayectorias realizada en este trabajo de tesis es aquella donde se pretende seguir una posición deseada considerando la interacción dinámica del órgano terminal del robot con un determinado ambiente, esta interacción dinámica es dada por la relación de fuerzas que se presentan al entrar en contacto el robot y la superficie, las cuales son la fuerza de contacto y la fuerza de reacción, que tienen la misma magnitud pero signo contrario. Para que la planeación de la trayectoria considere posición y fuerza controlada se utilizó un esquema de control híbrido posición/fuerza basado en esquemas de control PID, este control se puede ver como un control con dos grados de libertad ya que parte de la acción integral y proporcional es repartida en el camino directo que sale de la referencia y en el camino de la realimentación. Teniendo un control PID como control interno de posición del robot industrial se puede ver el control híbrido como un control PID-PI, en el cual el control PID se encarga de controlar la posición y se encuentra en el lazo interno del esquema de control y el PI controla la fuerza de contacto y se encuentra en el camino de la realimentación. La relación entre los dos controles descritos compone al control híbrido así como su conmutación, la cual se pudo observar como un diagrama de bloques en la

Figura 3.2.

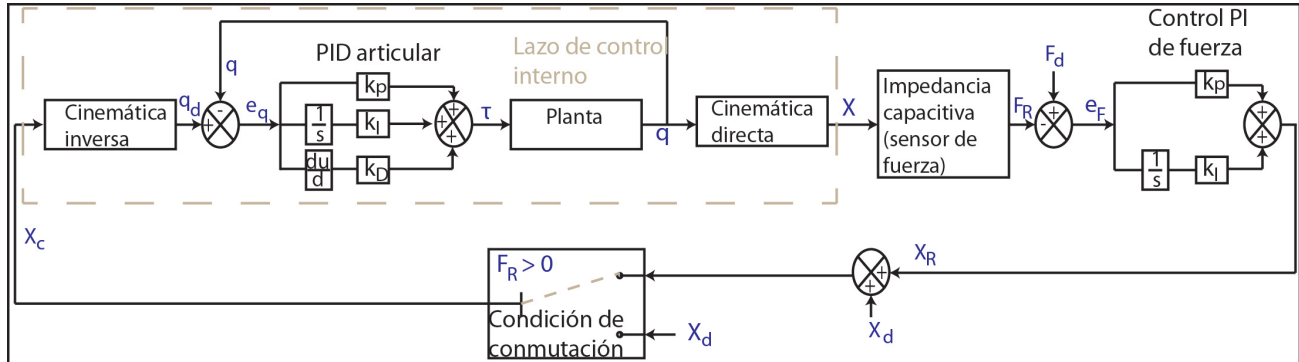


Figura 3.2: Esquema de control híbrido posición/fuerza

La planeación de trayectoria se realiza desde la posición de inicio del robot (home del robot), que se establece en la configuración del robot industrial, desde esta posición el control interno de posición empieza a funcionar, donde el mismo genera una interpolación de la trayectoria necesaria para llegar al punto deseado de contacto, una vez que hace contacto conmutan los modelos de control y se inicia el control híbrido que se encargará en ese momento de la planeación de la trayectoria y esta será una donde se controle la posición y la fuerza para resolver el problema de regulación o de seguimiento de trayectoria tanto en posición como en fuerza, para un contacto con una superficie de acrílico como se muestra en la Figura 3.1.

El esquema de control híbrido posición/fuerza realizado resultó de la modificación del esquema de control planteado en [1] y [5] con la diferencia de que, en lugar de usar un control óptimo de admitancia para controlar la fuerza.

El control híbrido posición/fuerza está conformado por dos controles, un control PID para controlar la posición articular, este control es el que se asume que está programado como control interno por el fabricante y un control PI para controlar la fuerza, el cual se encuentra en un lazo externo en el esquema de control y genera una señal de referencia de posición, que retroalimenta al control de posición del fabricante.

En la Figura 3.2 se observa el esquema completo del control del sistema robótico, el cual consiste en un control PID articular de posición, un control PI de fuerza, modelos cinemáticos, modelo dinámico, modelo del ambiente (impedancia capacitiva) y condición de conmutación. Ya que el control de posición es articular es necesario conocer los modelos cinemáticos para transformar las variables articulares a cartesianas y viceversas, como se aprecia la cinemática inversa recibe el valor de la posición comandada (X_c), la cual es transformada a la posición articular deseada, para generar el error de posición articular que retroalimenta al control PID de posición, el cual brinda a su salida una señal de control en forma de par τ , que es enviada al modelo dinámico para controlar la dinámica

del sistema, este modelo dinámico permite tener la relación entre el par y las posiciones articulares, las cuales son convertidas a una posición cartesiana actual que se utiliza en la impedancia capacitiva para obtener una relación directa con la fuerza de reacción, que restada a la fuerza deseada, genera el error de fuerza que retroalimenta al control PI de fuerza para obtener a su salida la posición de referencia que es adicionada con la posición cartesiana deseada y así se obtiene una posición comandada, la cual es la señal a seguir por el control interno de posición del robot. En el lazo interno del control híbrido se controla solo la posición articular por el PID de posición que actuará de manera independiente desde el estado de inicio donde el robot se encuentra en movimiento libre, es decir sin contacto con la superficie. En el lazo externo se observa el control PI de fuerza el cual se acciona por medio de la condición de conmutación, ya que cuando el robot entra en contacto, se debe controlar tanto posición como fuerza. Cabe mencionar que este esquema de control completo solo es posible implementarlo si se conocen los modelos cinemático y dinámico del robot y si el robot permite modificar su arquitectura de control, por ello más adelante se explica la reducción de este esquema para implementarse en el robot industrial sin modificar su arquitectura de control.

3.1. Control PI de fuerza

$$X_R(t) = k_p e_F(t) + k_I \int_0^t e_F(T) dT \quad (3.1)$$

$$e_F = F_d - F_R \quad (3.2)$$

Para controlar la fuerza de contacto en el efector final del robot, se implementó un control PI de fuerza, descrito por la ecuación (3.1), el cual tiene como entrada el error de fuerza (3.2) y solo se ocupa la parte integral y proporcional ya que el monitorear señales de fuerza genera ruido y al derivar el error de fuerza se obtendrá un ruido considerable que afecte al control híbrido. Este control de fuerza que se implementó, fue un control activo ya que se tiene manipulación directa del valor de fuerza de contacto obtenida a través del sensor de fuerza en los experimentos reales y del modelo del ambiente en las simulaciones en Matlab y en RAPID.

La ecuación (3.2) define el error de fuerza como la fuerza deseada F_d menos la fuerza de reacción F_R , esta fuerza de reacción proviene del modelo de impedancia (modelo del ambiente) ó del sensor de fuerza.

La sintonización del control PI de fuerza se obtiene de forma experimental y se basa en encontrar ciertos valores de ganancia proporcional k_p y ganancia integral k_I en base al comportamiento de la interacción con el ambiente.

Cuando el robot está en contacto con el ambiente, la salida del control PI será la posición

de referencia X_R , esta posición se adiciona con la posición deseada X_d , que corresponde con la posición del ambiente X_a , para generar la posición comandada $X_c = X_d + X_R$ que es la señal que se retroalimenta al control de posición del fabricante, motivo por el cual el control PI de fuerza se encuentra en el lazo externo del esquema de control y el de posición en un lazo interno. La relación entre estas señales se observa en forma de esquema en las Figuras 3.2 y 3.4.

3.2. Control PID de posición del fabricante

El control de posición del robot industrial se encuentra pre-programado en su unidad de control y por lo general se basa en un control PID articular [18], el cual tiene como entrada la posición deseada q_d obtenida de la cinemática inversa, la cual es resuelta por el software del fabricante. El control PID articular es descrito por,

$$\tau = k_p e_p(t) + k_i \int_0^t e_p(T) dT + k_d \frac{de_p(t)}{dt} \quad (3.3)$$

$$e_p = q_d - q \quad (3.4)$$

El control de posición PID industrial que contiene el robot, es implementado por medio de la instrucción de movimiento articular `moveJ`, la cual se muestra con argumentos de ejemplo.

```
moveJ p1, v50, z30, tool1;
```

Esta instrucción contiene en su código fuente el control PID de posición articular y una interpolación polinomial que ABB ha creado para ser utilizada desde RAPID, que es el lenguaje de programación que utilizan los robots industriales de la marca ABB, sin embargo el fabricante no permite modificar este código fuente.

La instrucción contiene los siguientes argumentos:

`p1` = $[x, y, z, \alpha, \beta, \gamma]$ posición cartesiana deseada conformada de 3 componentes de posición y 3 de orientación, cabe mencionar que el software del robot resuelve la cinemática inversa y no es necesario especificarla, solamente se le ingresa el valor de posición y orientación.

`v50` representa una velocidad deseada, en este caso de $50[mm/s]$, la cual puede cambiarse a cualquier valor de hasta $5000[mm/s]$ que es la velocidad máxima del robot.

`z30` indica la precisión con la que se llega al punto deseado, entre menor sea su valor se tendrá mayor precisión.

`tool1` es el nombre de una variable tipo `tooldata` que contiene todos los parámetros

del efector final que se esté utilizando, como el centro de masa, posiciones cartesianas, centro de gravedad y masa total aproximada del efector.

$$p1 = X_c = X_d + X_R \quad (3.5)$$

A la instrucción `moveJ` se le manda la posición comandada de la ecuación (3.5) en tiempo real, que es la salida del control de fuerza (posición de referencia) más la posición deseada del ambiente.

3.3. Condición de conmutación

Esta condición establece la conmutación de dos esquemas de control, uno cuando no existe fuerza de reacción ($F_R = 0$), que corresponde al movimiento libre del robot y es controlado solo por el control interno del fabricante, y el otro esquema entra en función cuando existe fuerza de reacción en el contacto ($F_R > 0$), activándose el control híbrido, en el cual un control de fuerza retroalimenta al de posición y debido a esta condición la entrada al control interno será la posición deseada X_d o la posición comandada X_c . En general esta condición lo que realiza es conmutar entre el control de posición y el control híbrido como se observa en el diagrama de flujo de la Figura 3.3.

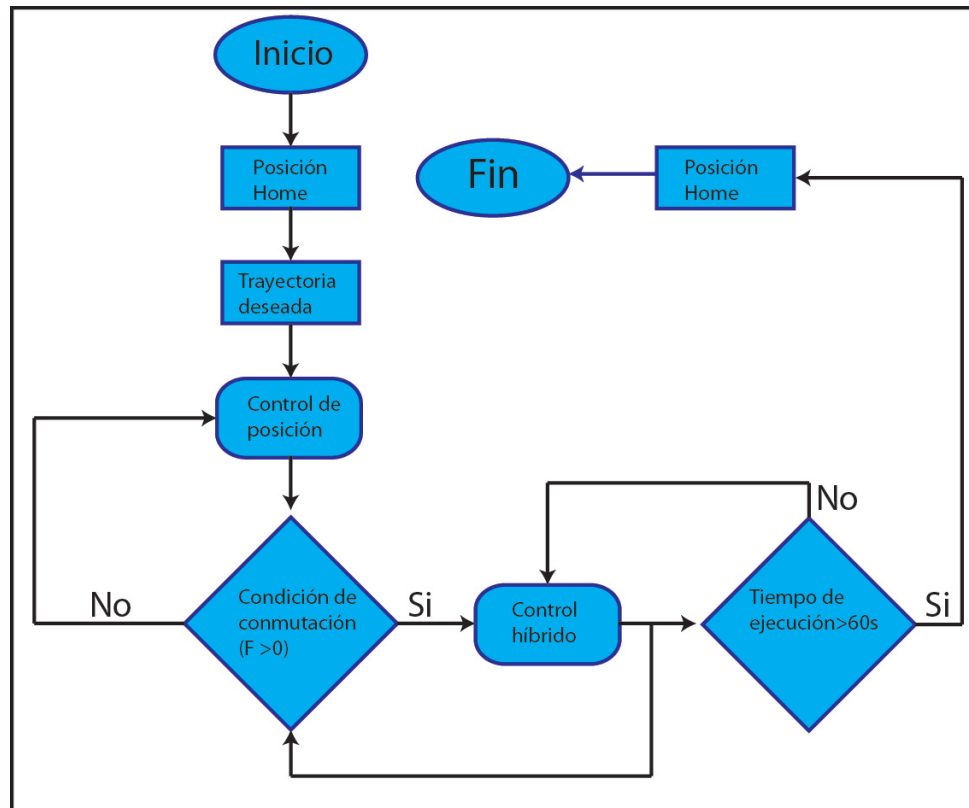


Figura 3.3: Diagrama de flujo de la condición de conmutación

En la Figura 3.3 se observa el último bloque de condición, en el cual se establece un tiempo de ejecución por medio de un condicional “if” dentro del programa principal, que al monitorear una variable tiempo, si se supera el tiempo establecido (60 segundos), el robot detendrá su ejecución con la instrucción STOP y se manda a la rutina de home, por medio de la cual el robot regresa a su posición inicial.

También cabe señalar que el control híbrido posición/fuerza tiene un compromiso entre el control de posición y el control de fuerza, ya que si se exige un mejor desempeño en el control de posición, el control de fuerza se verá disminuido en su desempeño y viceversa, por ello se deben de ajustar las ganancias tomando en cuenta la tarea a realizar para buscar mayor precisión en posición o en fuerza.

3.4. Implementación del control híbrido en el robot industrial

El esquema de control de la Figura 3.2 se implementó en simulación para un robot planar de 3GDL en el software Simulink de Matlab, para desarrollar por completo la teoría aplicada del control de fuerza en un robot manipulador, que sirvió como base para implementar este mismo control en el robot industrial.

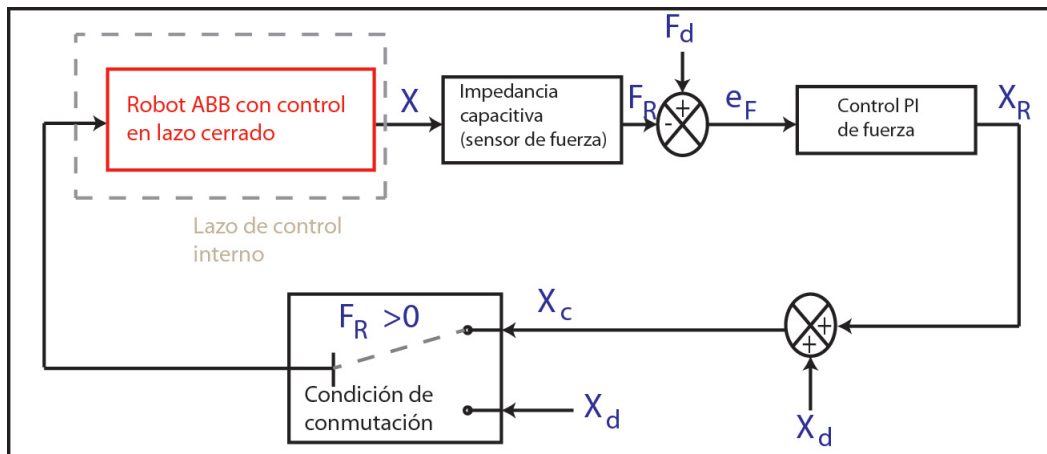


Figura 3.4: Esquema de control híbrido posición/fuerza para el robot industrial IRB 2400 ABB

La idea de implementar un control de fuerza en un robot industrial fue sin modificar su arquitectura y utilizar su propio control de posición interno para verificar las altas prestaciones que se podían obtener, el esquema de la Figura 3.2 se reduce al esquema de control de la Figura 3.4, ya que el propio sistema del robot industrial realiza las transformaciones de las variables involucradas, con un control propio de posición, entonces como se mencionó anteriormente en la parte del control de posición, la etapa de conversión de variables cartesianas a articulares y viceversa por medio de los modelos cinemáticos lo

3.5. CONTROL HÍBRIDO POSICIÓN/FUERZA DE UN ROBOT PLANAR DE 3GDL39

realiza las instrucciones predefinidas en RAPID, así como el control de posición lo realiza automáticamente el robot, con solo indicarle o mandarle una posición de referencia a seguir para generar un cambio de fuerza cuando el efector final este en contacto y este cambio de fuerza es leído por el sensor de fuerza que retroalimenta al control PI de fuerza programado en el robot que a su vez genera la posición de referencia.

3.5. Control híbrido posición/fuerza de un robot planar de 3GDL

El siguiente desarrollo del control híbrido aplicado a un robot de 3GDL se obtuvo siguiendo el esquema presentado en la Figura 3.2

El control híbrido posición/fuerza se desarrolló primeramente para un robot de 3GDL, con la intención de mostrar el desarrollo completo del esquema de control de la Figura 3.2, ya que la implementación en el robot industrial se reduce en gran parte por no ser necesario utilizar los modelos matemáticos, ni implementar directamente un algoritmo de control de posición, solo de fuerza, además que en el control híbrido aplicado al robot industrial no se tendrá control del par en los motores, ya que este tiene una arquitectura cerrada, donde solo se controlará la posición comandada. Este robot se diseñó como un modelo de CAD desde el software Solidworks (véase Figura 3.5b), donde se establecieron sus medidas y parámetros, este modelo CAD se utilizó para generar una animación desde Simulink de Matlab. La animación del robot de 3GDL se realizó por medio de la exportación del diseño CAD como un archivo .xml, el cual se carga al toolbox de Simulink llamado SimMechanics, con ello Simulink genera un modelo virtual del mismo que se puede observar en la Figura 3.5a, en el cual se ubican en automático los centros de masa y se puede acceder a sus variables articulares, así como ingresar una señal de control al modelo virtual, entonces se generó un esquema de control desde Simulink (véase Figura 3.6), el cual permite tener una simulación del manipulador de 3 gdl, además de la animación generada con SimMechanics.

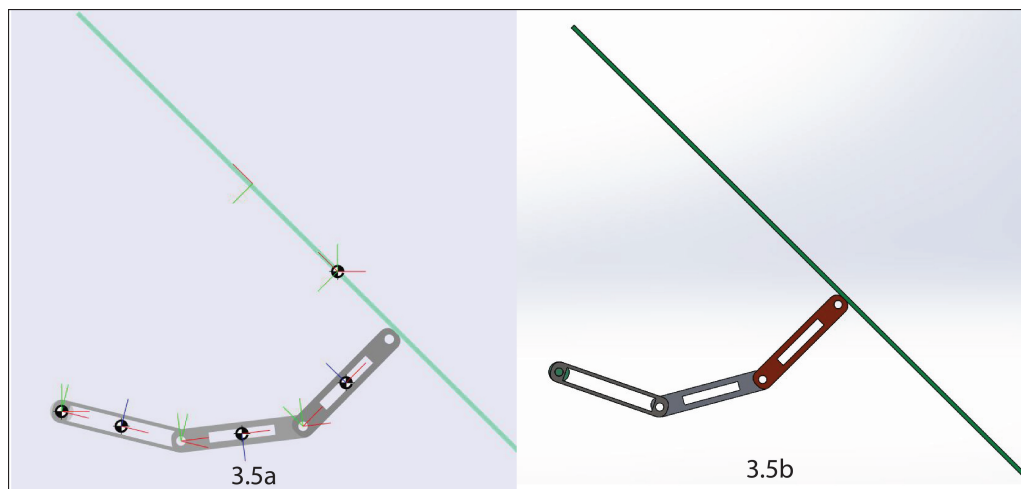


Figura 3.5: Modelo virtual del robot planar de 3GDL

La Figura 3.6 que representa un diagrama de bloques de Simulink tiene la misma estructura del control híbrido planteado de manera general, solo difiere en que en este se controlan dos componentes de fuerza, ya que la tarea a realizar es el contacto del robot de 3 gdl con un plano inclinado a 45° , sin embargo como el contacto es normal al plano de la superficie de contacto, se tiene un control de 1 gdl en la fuerza, solo que en este caso se descompone en dos componentes $F_x = (1N) * \text{Cos}(45^\circ)$ y $F_y = (1N) * \text{Sin}(45^\circ)$, estas componentes representan la fuerza deseada de contacto.

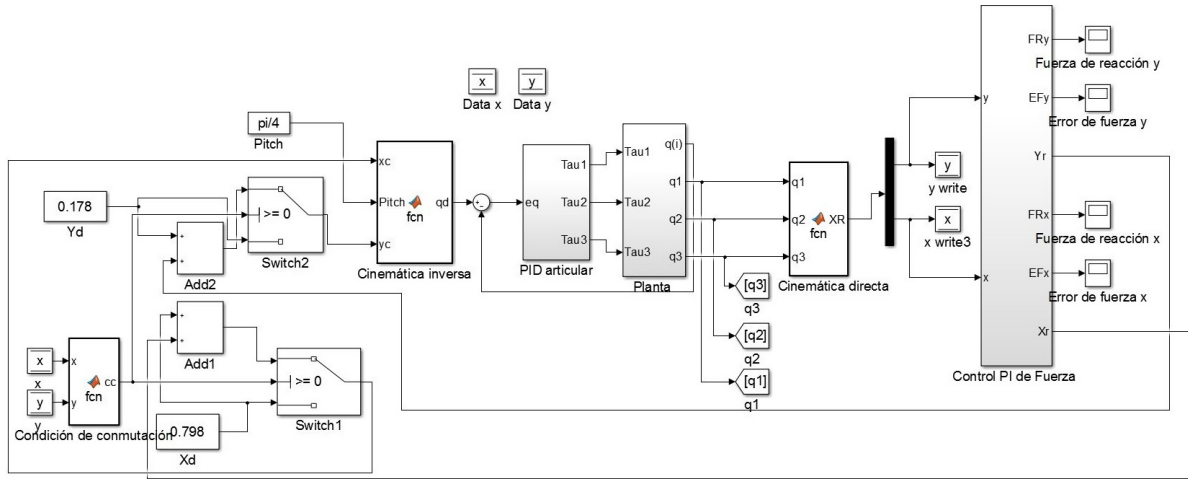


Figura 3.6: Esquema de control híbrido posición/fuerza en Simulink.

El control de posición se basa en un PID articular el cual tiene como entrada el error de posición en variables articulares, para ello se implementa la ecuación (3.3) en bloques de Simulink. Como se ilustra en la Figura 3.7, se genera un error de posición e_q , el cual se integra y se deriva para generar los errores derivativo e integrativo y a su vez se multiplican por una ganancia integral y derivativa para ser sumados con el proporcional, que es el error mismo por una ganancia proporcional. A la salida del controlador se tendrán los pares τ necesarios para controlar al robot, los cuales pasan por bloques de saturación como se observa en la Figura 3.7 que limitan el par en un rango de $[-10[Nm], 10[Nm]]$ para regulación y $[-30[Nm], 30[Nm]]$ para seguimiento de trayectoria.

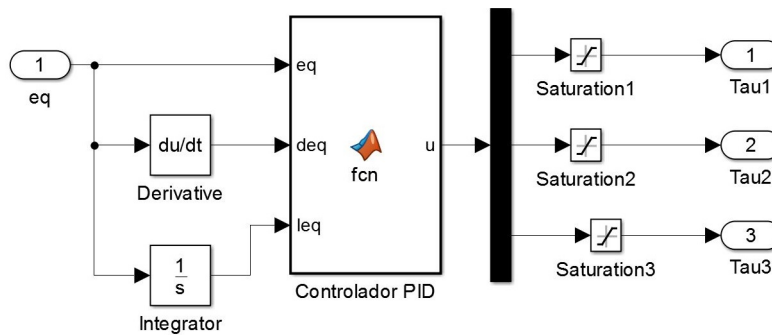


Figura 3.7: Esquema de control PID en Simulink.

3.5. CONTROL HÍBRIDO POSICIÓN/FUERZA DE UN ROBOT PLANAR DE 3GDL41

Ya que se tienen 3 gdl y tres actuadores uno en cada articulación del robot, tendremos tres matrices para las ganancias k_p , k_I , k_d de dimensión 3x3 que tendrán solo valores en su diagonal ya que se tiene el sistema desacoplado.

El control de fuerza se implementó de la misma forma utilizando la ecuación (3.1), solo que no se añadió la parte de control derivativa por el ruido que se tiene al medir la señal de fuerza.

Como la condición de conmutación detecta si el robot esta en movimiento libre o interactuando con el ambiente, entonces se obtiene la ecuación del plano inclinado (superficie de contacto), para saber si la posición actual del robot está en contacto o no. A continuación se deduce la ecuación del plano y la condición de conmutación:

Considerando dos puntos (x_1, y_1) y (x_2, y_2) obtenemos la ecuación del plano con un ángulo de 45° , que se encuentra dentro del rango de trabajo del robot.

$$x_1 = 0.976[m], \quad y_1 = 0[m]$$

$$x_2 = 0[m], \quad y_2 = 0.976[m]$$

$$\frac{x - x_1}{x_2 - x_1} = \frac{y - y_1}{y_2 - y_1}$$

$$y = -x + 0.976$$

$$y + x - 0.976 = 0$$

Las posiciones (x_1, y_1) y (x_2, y_2) representan puntos específicos de las posiciones actuales del robot (x, y) sobre la superficie de contacto (plano inclinado) y las posiciones en general de la recta o del plano inclinado son las posiciones deseadas (x_d, y_d) que se establecen como las mismas posiciones del ambiente (x_m, y_m) .

Si $y + x - 0.976 > 0$ la condición nos indica que se penetró en la superficie.

Si $y + x - 0.976 < 0$ la condición nos indica que no hay contacto con la superficie.

El diseño de la trayectoria a seguir consiste en generar una ecuación paramétrica de la recta para que el robot siga el contorno del plano inclinado, manteniendo un contacto con 1N de fuerza, esta trayectoria se define a continuación:

La trayectoria diseñada permite hacer un barrido de movimientos sobre el contorno de la superficie inclinada a 45° .

Para el caso de regulación tenemos los siguientes puntos deseados:

$$x_d = 0.798[m], \quad y_d = 0.178[m]$$

Para el caso de seguimiento de trayectoria tenemos los siguientes puntos deseados:

Punto inicial

$$x_0 = 0.798[m], \quad y_0 = 0.178[m]$$

Punto final

$$x_n = 0.607[m], \quad y_n = 0.369[m]$$

Para el seguimiento de trayectorias se diseñaron las siguientes funciones:

Trayectoria en x

$$x = -\frac{0.191}{2}\cos(0.5wt) + \frac{1.405}{2}$$

Trayectoria en y

$$y = \frac{0.191}{2}\cos(0.5wt) + \frac{0.547}{2}$$

Fuerzas deseadas

$$F_{dx} = F_{dy} = 0.7071[N]$$

Simulación

Para mostrar el comportamiento del robot de 3 gdl ante el control híbrido presentado al inicio del capítulo, se muestra una simulación y animación con Simulink/Matlab utilizando una rutina de integración Runge-Kutta de cuarto orden con un paso fijo de 0.001 y en la Tabla 3.2 se presentan los parámetros de simulación.

Ganancias del control híbrido	Valor numérico
Ganancia proporcional del PID articular	$k_p=[75 \ 0 \ 0;0 \ 22.5 \ 0;0 \ 0 \ 22.5]$
Ganancia integral del PID articular	$k_I=[100 \ 0 \ 0;0 \ 40 \ 0;0 \ 0 \ 40]$
Ganancia derivativa del PID articular	$k_D=[3 \ 0 \ 0;0 \ 1 \ 0;0 \ 0 \ 0.6]$
Ganancia proporcional del PI de fuerza	$k_p=[0.0004 \ 0;0 \ 0.0009]$
Ganancia integral del PI de fuerza	$k_I=[0.1 \ 0;0 \ 0.04]$

Tabla 3.1: Ganancias del control híbrido posición/fuerza para un robot de 3GDL.

La simulación se emplea para resolver dos problemas de control, uno donde se realiza regulación tanto en posición como en fuerza, y otro donde se realiza seguimiento de trayectoria en posición y regulación en fuerza, para ello se realizó una sintonización dando preferencia al desempeño del control de posición y se obtuvieron las ganancias de la Tabla 3.1 para los dos problemas de control.

Cabe mencionar que el robot de 3GDL se inició (condición inicial) cerca del punto deseado de contacto definido en el diseño de la trayectoria a seguir, por ello no se generó una

3.5. CONTROL HÍBRIDO POSICIÓN/FUERZA DE UN ROBOT PLANAR DE 3GDL43

interpolación de trayectoria para llegar a este punto.

Parámetros de simulación	Valor numérico
Rigidez del ambiente	
Masa del primer eslabón	$m_1 = 0.5 [Kg]$
Masa del segundo eslabón	$m_2 = 0.3 [Kg]$
Masa del tercer eslabón	$m_3 = 0.2 [Kg]$
Rigidez del ambiente	$k_a = 100200 [N/m]$
Momento de inercia del primer centro de masa	$I_1 = 0.3 [Kg m^2]$
Momento de inercia del segundo centro de masa	$I_2 = 0.3 [Kg m^2]$
Momento de inercia del tercer centro de masa	$I_3 = 0.3 [Kg m^2]$
Longitud del primer eslabón	$l_1 = 0.3 [m]$
Longitud del segundo eslabón	$l_2 = 0.3 [m]$
Longitud del tercer eslabón	$l_3 = 0.3 [m]$
Distancia de la base al primer centro de masa	$l_{c1} = 0.15 [m]$
Distancia de la segunda articulación al segundo centro de masa	$l_{c2} = 0.15 [m]$
Distancia de la tercera articulación al tercer centro de masa	$l_{c3} = 0.15 [m]$
Aceleración producida por la gravedad	$g = 9.81 [m/s^2]$
Amortiguamiento referente al primer eslabón	$\mu_1 = 0.175$
Amortiguamiento referente al segundo eslabón	$\mu_2 = 0.175$
Amortiguamiento referente al tercer eslabón	$\mu_3 = 0.175$

Tabla 3.2: Parámetros de simulación del robot de 3GDL.

A continuación se establece la condición inicial q_0 de las variables articulares, como un punto cercano al plano inclinado para una mejor convergencia del control, ya que de no ser así se tendría que realizar una interpolación de trayectoria, desde el inicio del robot al punto de contacto.

$$q_0 = [-0.2750[rad], 0.4130[rad], 0.6472[rad]]$$

Simulación a un punto deseado sobre una superficie inclinada a 45°

En la Figura 3.8 se observa las posiciones angulares correspondientes a cada eslabón, donde la mayor oscilación se presenta en el tercer eslabón, debido a que se inicia cerca de la superficie de contacto, entonces se tiene contacto en el punto deseado sobre el plano inclinado en los primeros segundos.

Las componentes de fuerza y posición se muestran en la Figura 3.9, el valor de la posición en x y y empieza a estabilizarse cerca del segundo 2, a diferencia de las componentes de fuerza que tienen una mejor convergencia, al estabilizar su valor en el primer segundo, sin embargo en el valor de fuerza tanto en x como en y tiene un incremento instantáneo en forma de pico al principio de la simulación, debido al contacto inicial del robot.

Comparando los errores de fuerza y posición (véase en la Figura 3.10), se observa el compromiso que existe entre los dos controles involucrados, ya que el error en la posición es mayor que el error en fuerza, si se buscara un error menor en la posición, el error de fuerza tendría que ser un poco mayor.

En la Figura 3.11 se muestran los pares generados a la salida del control híbrido, donde se aprecia que el saturador entra en función solo al principio de la simulación, que es donde se requiere un mayor par, después de ello el valor par es menor en estado estacionario.

Las señales de referencia (véase en la Figura 3.12) que son las salidas del control PI de fuerza, presentan un valor de pequeña magnitud, ya que van adicionadas a los valores deseados en x y y para obtener la posición comandada a seguir por el control de posición.

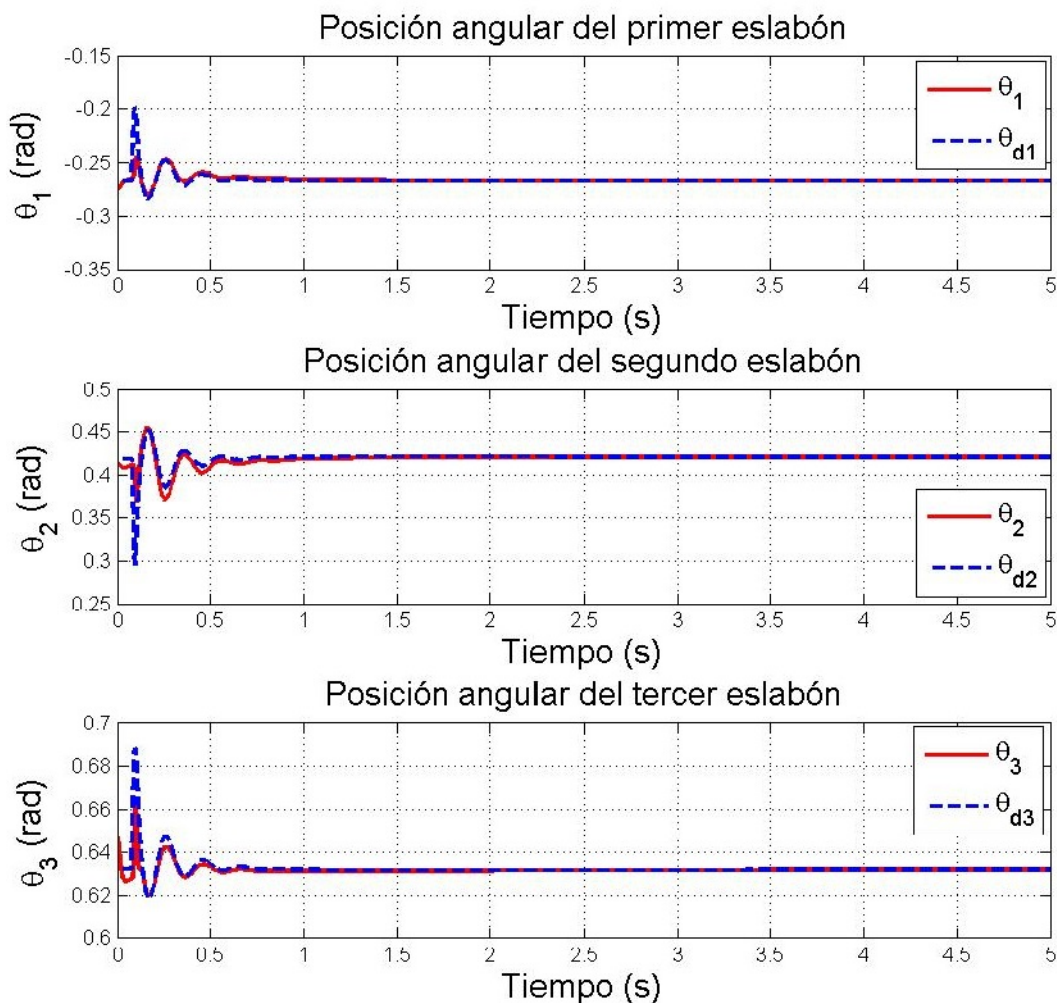


Figura 3.8: Variables articulares para el caso de regulación con una superficie inclinada a 45° .

3.5. CONTROL HÍBRIDO POSICIÓN/FUERZA DE UN ROBOT PLANAR DE 3GDL45

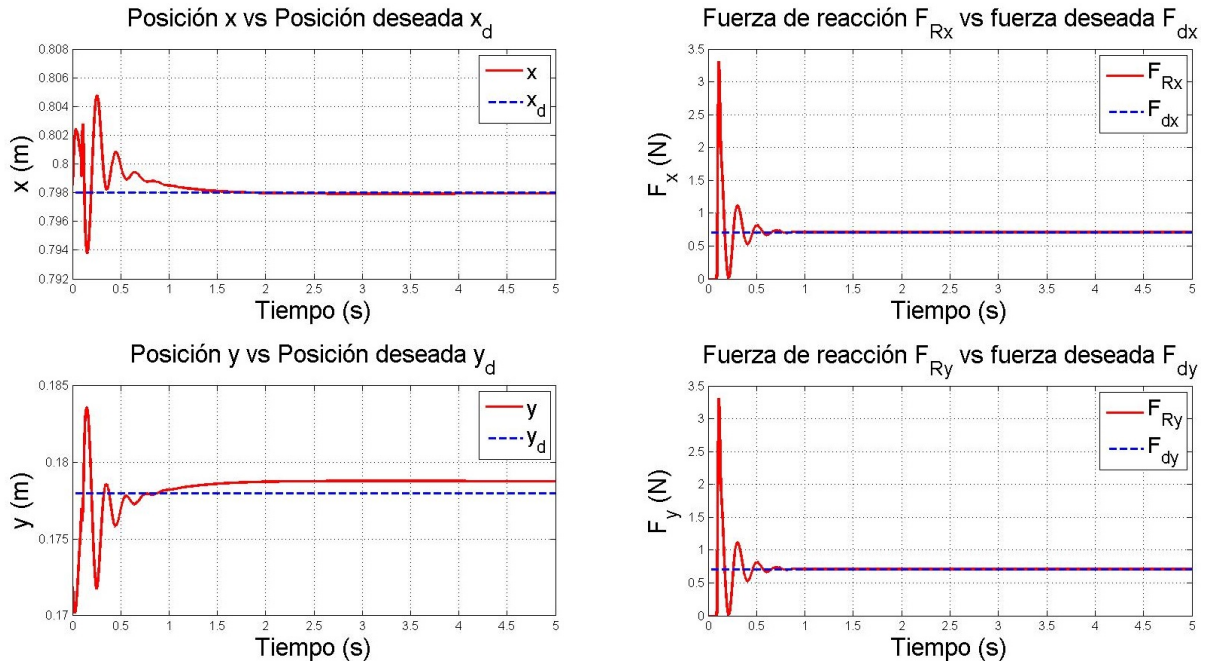


Figura 3.9: Componentes de posición y fuerza para el caso de regulación con una superficie inclinada a 45° .

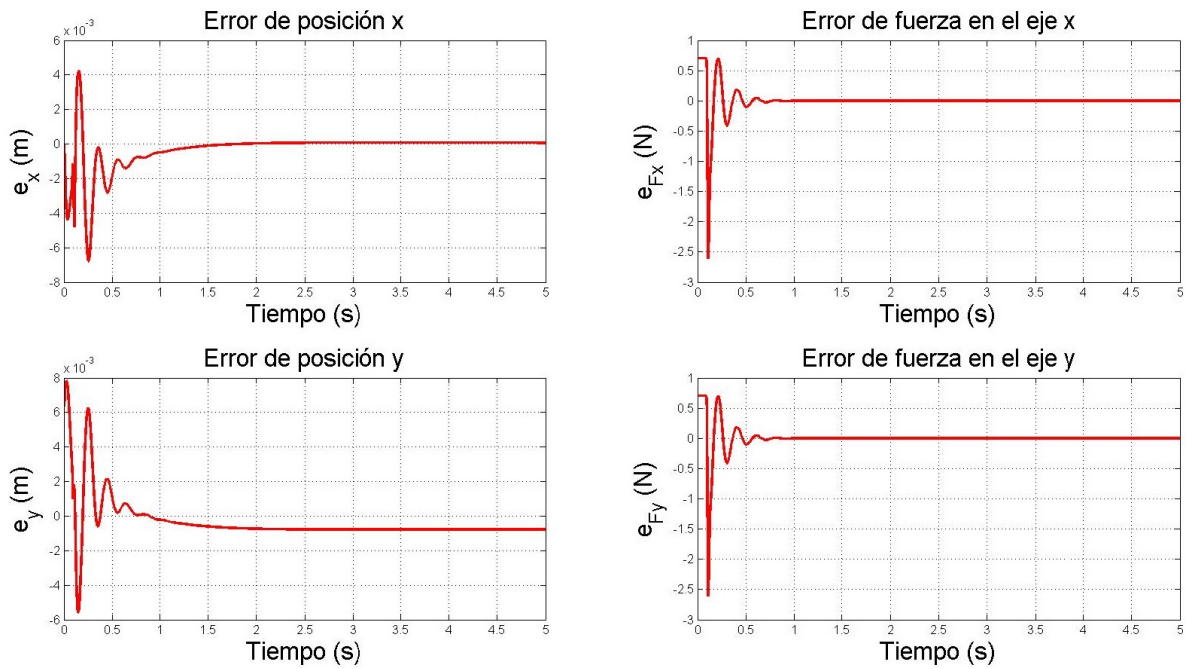


Figura 3.10: Errores de posición y fuerza para el caso de regulación con una superficie inclinada a 45° .

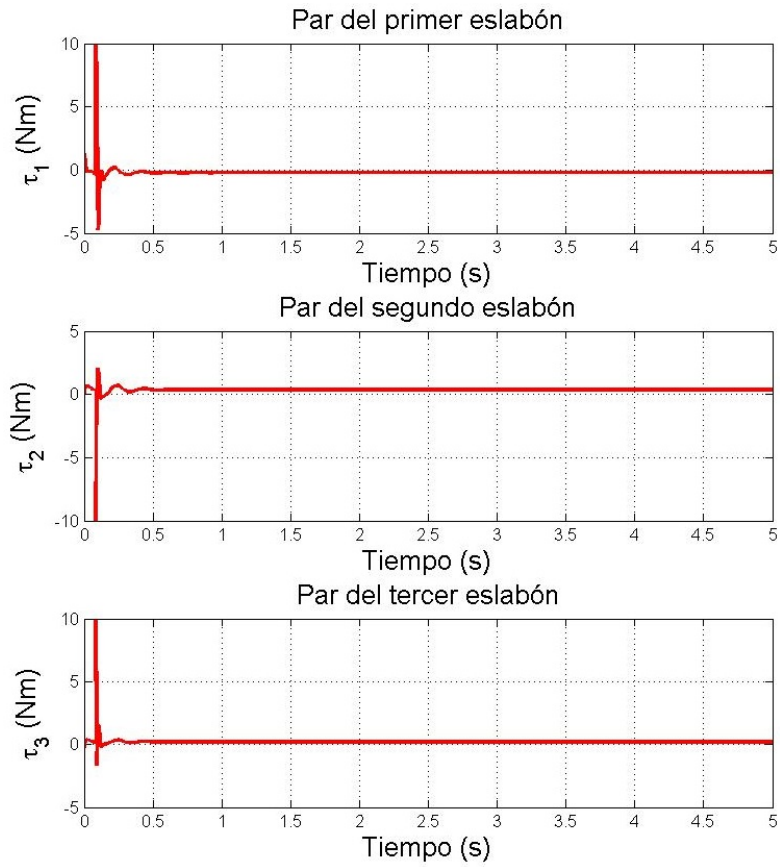


Figura 3.11: Pares de control para el caso de regulación con una superficie inclinada a 45° .

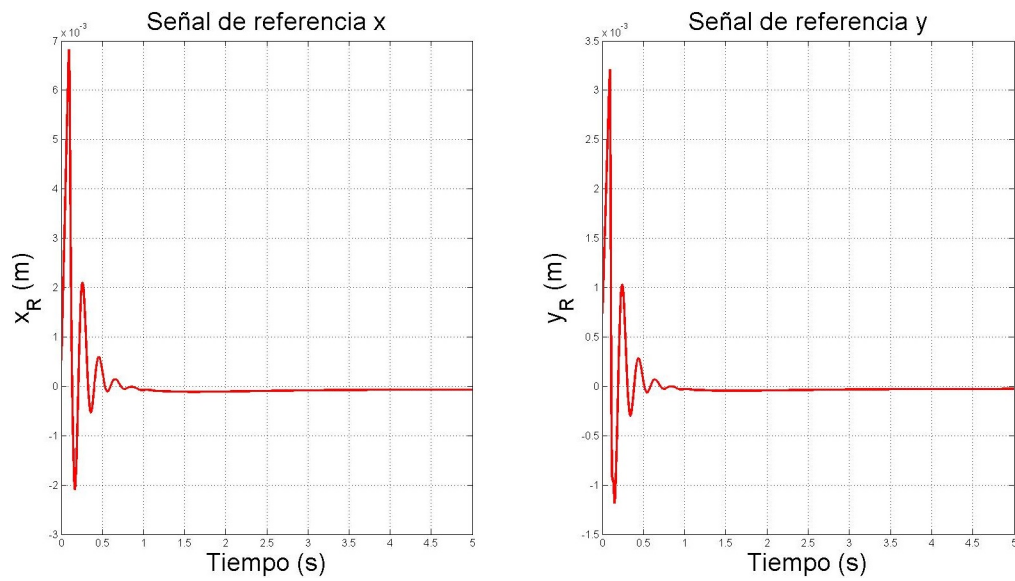


Figura 3.12: Señales de referencia para el caso de regulación con una superficie inclinada a 45° .

3.5. CONTROL HÍBRIDO POSICIÓN/FUERZA DE UN ROBOT PLANAR DE 3GDL47

Cabe mencionar que debido a que el problema de control es regulación, se pudo tener contacto con el plano inclinado todo el tiempo, con una buena sintonización de las ganancias del control híbrido.

Simulación de seguimiento de trayectoria de una superficie plana inclinada a 45°

En la Figura 3.13 se observa las posiciones angulares respecto al centro de masa de cada eslabón, donde el seguimiento es mejor a partir del segundo 2 y en los primeros 2 segundos se tienen irregularidades en la trayectoria a seguir de color azul, debido a que la trayectoria a seguir es generada a partir del control de fuerza, ósea es una trayectoria que involucra una regulación de fuerza.

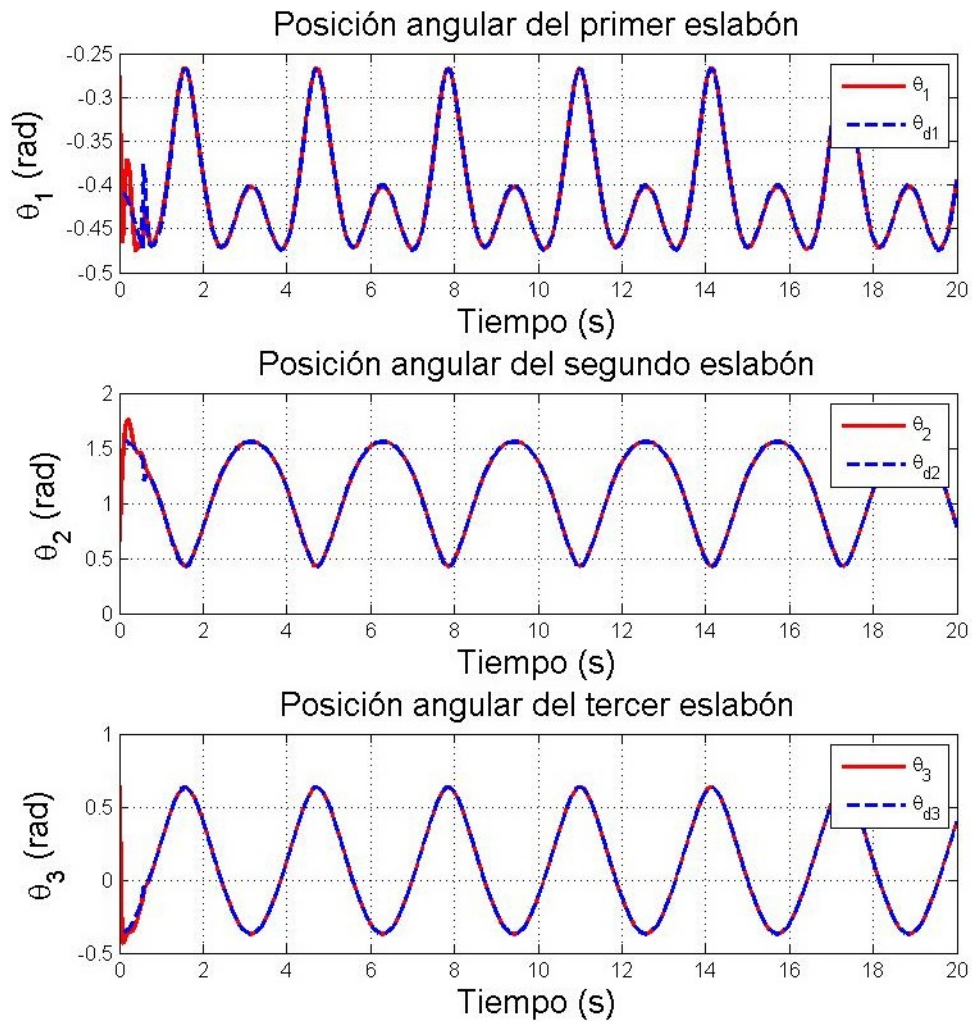


Figura 3.13: Posiciones angulares para el seguimiento de trayectoria sobre el contorno de la superficie inclinada a 45° .

En las Figuras 3.15 y 3.16 se observan las posiciones, fuerzas y sus errores durante la simulación, se muestra un buen comportamiento en el seguimiento de posición, sin embargo en las componentes de fuerza se observan picos de forma periódica en estado estacionario, esto es debido a los instantes donde robot cambia de dirección para volver a recorrer el plano inclinado, sin embargo en el tiempo que se mantiene el movimiento recto se regula la fuerza con un mínimo error.

Al igual que los pares de control del caso de regulación, con una buena sintonización de las ganancias y considerando el instante de contacto, se logra tener un elevado par solamente al inicio, si es muy elevado entra en saturación (véase en la Figura 3.14), debido a que se necesita un mayor par cuando el robot pierde contacto con la superficie.

En las Figuras 3.17 y 3.18 se muestran las posiciones de referencia y comandadas, las cuales tiene el mismo comportamiento periódico pero distinta magnitud, dentro de su comportamiento se observan cambios constantes en su magnitud debido a las oscilaciones que se presentan en el seguimiento de la trayectoria.

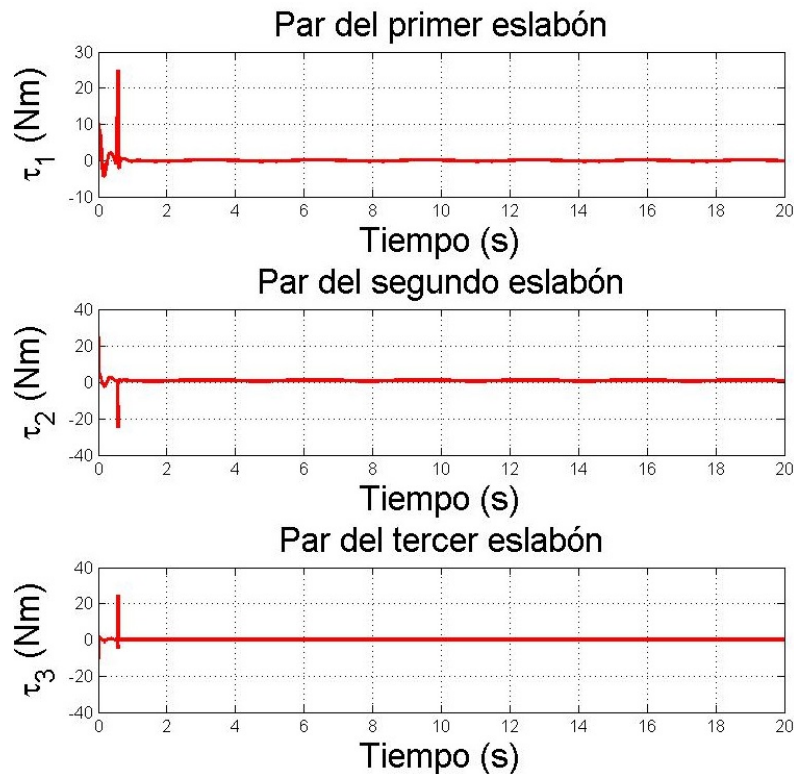


Figura 3.14: Pares de control para el seguimiento de trayectoria sobre el contorno de la superficie inclinada a 45° .

3.5. CONTROL HÍBRIDO POSICIÓN/FUERZA DE UN ROBOT PLANAR DE 3GDL49

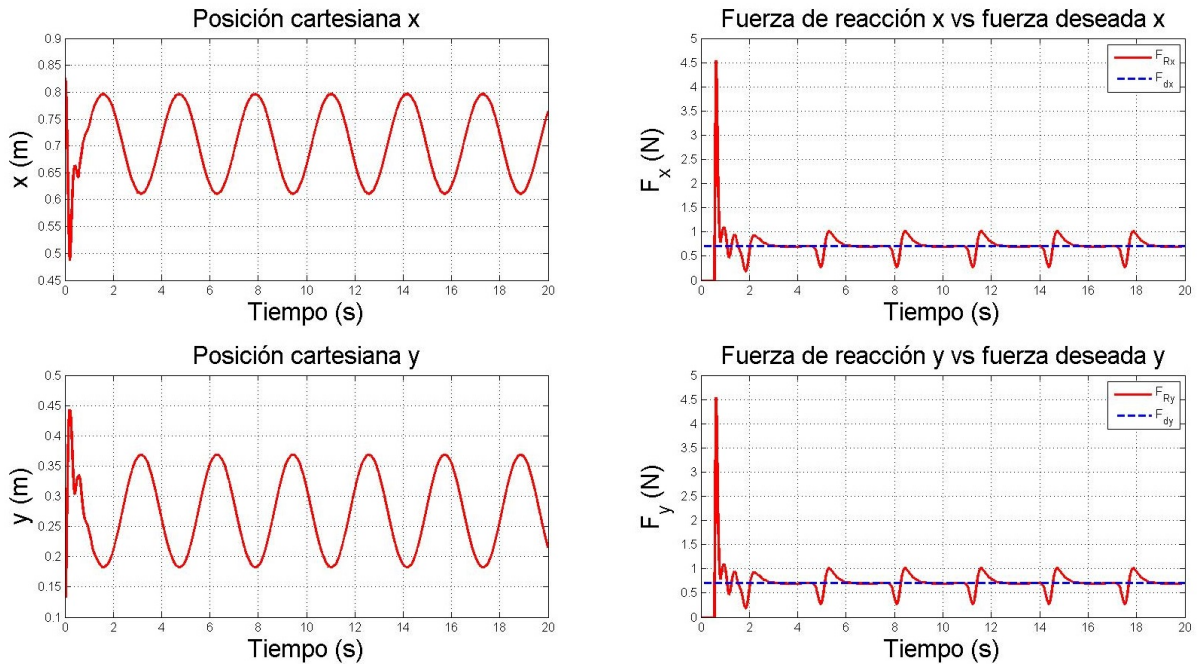


Figura 3.15: Componentes de posición y fuerza para el seguimiento de trayectoria sobre el contorno de la superficie inclinada a 45° .

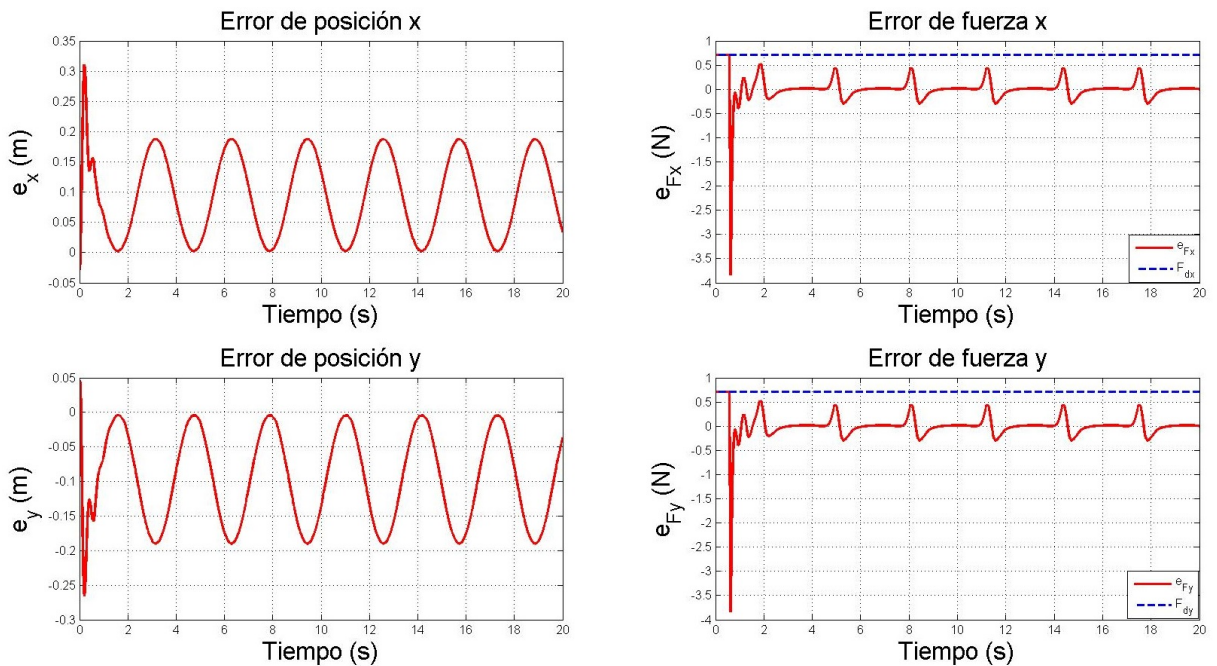


Figura 3.16: Errores de posición y fuerza para el seguimiento de trayectoria sobre el contorno de la superficie inclinada a 45° .

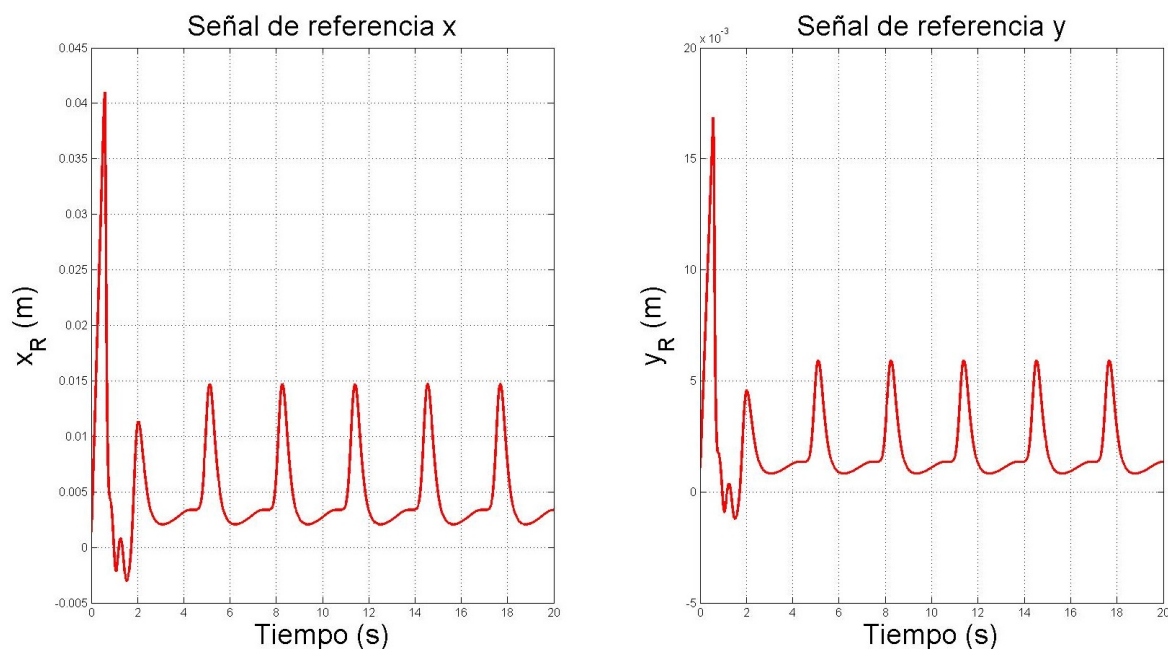


Figura 3.17: Señales de referencia para el seguimiento de trayectoria sobre el contorno de la superficie inclinada a 45° .

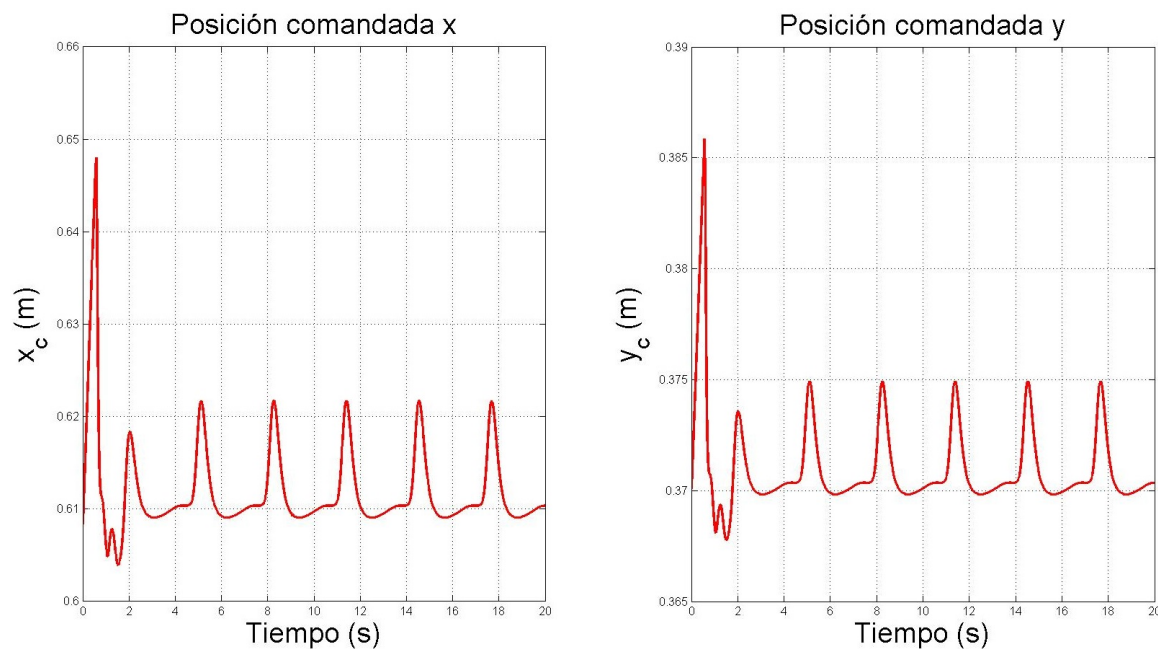


Figura 3.18: Señales comandadas para el seguimiento de trayectoria sobre el contorno de la superficie inclinada a 45° .

Para corroborar lo dicho anteriormente en las gráficas de posición y fuerza, donde no se presentaron problemas por la pérdida de contacto, se muestran los estados de conmutación

3.5. CONTROL HÍBRIDO POSICIÓN/FUERZA DE UN ROBOT PLANAR DE 3GDL51

del control de posición y del control híbrido (véase la Figura 3.19), ya que cuando el robot pierde contacto con la superficie entra en función solamente el control de posición, que reubica al robot al punto de contacto, y como se aprecia en la gráfica cuando el robot entra en contacto por primera vez, logra mantener ese estado todo el tiempo de la simulación.

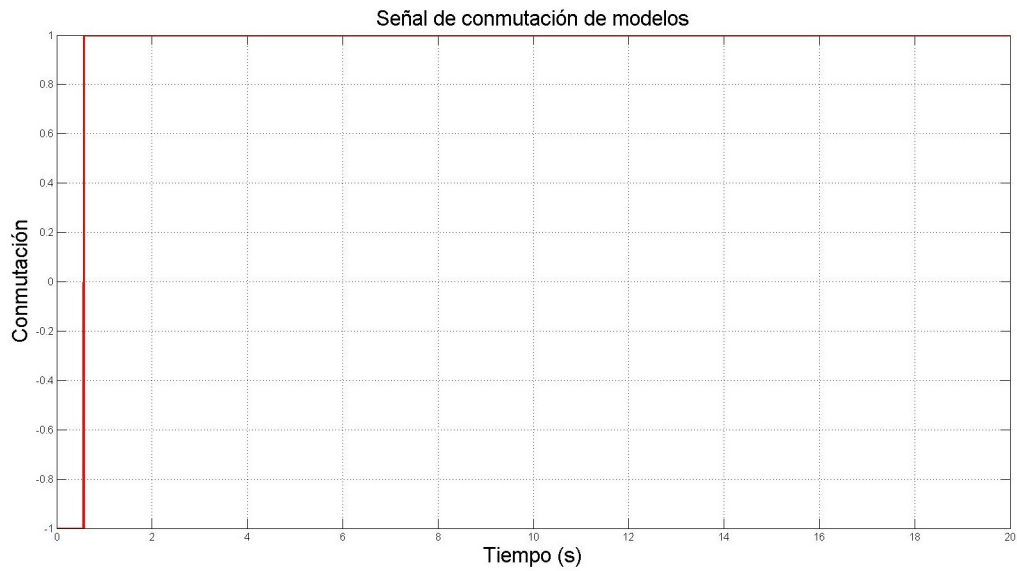


Figura 3.19: Señal de conmutación para el seguimiento de trayectoria sobre el contorno de la superficie inclinada a 45° .

Capítulo 4

Implementación de la arquitectura de control de fuerza

El control híbrido posición/fuerza se implementó en un robot industrial IRB2400 de la marca ABB que posee una unidad de control IRC5, con un sistema operativo RobotWare 5.15 y el software de simulación y programación de ABB, RobotStudio 6.00.01 de 64bits con licencia RobotStudio Premium.

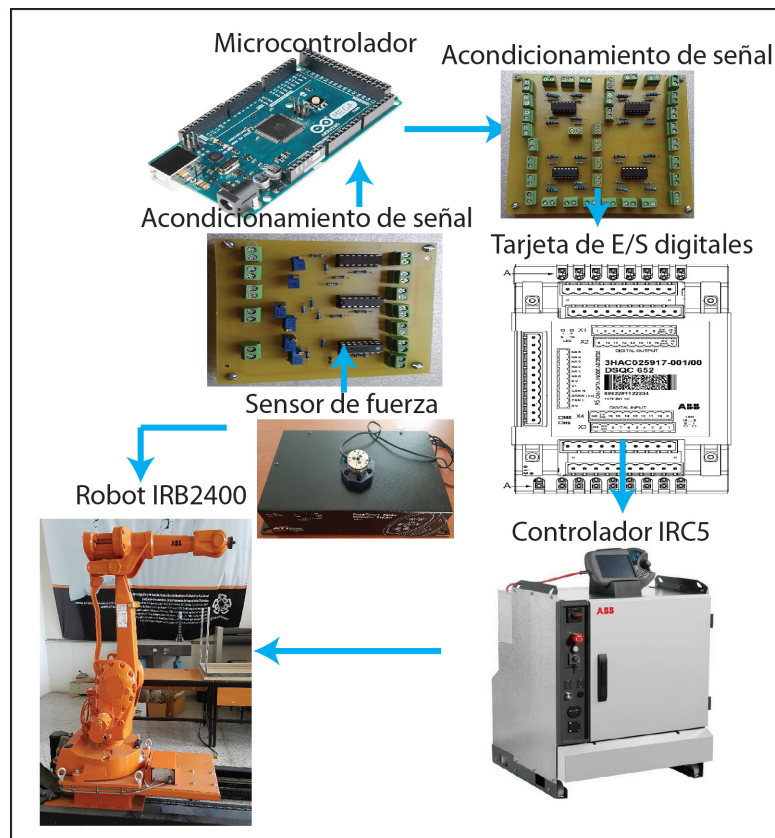


Figura 4.1: Esquema de comunicación sensor de fuerza-robot.

Para ser posible la implementación del control mencionado sin utilizar el kit de fuerza de ABB llamado integrated force control, se utilizaron los requerimientos mínimos que contiene un robot industrial como los mencionados en el párrafo anterior (sin protocolos de comunicación que el robot puede tener como aditamento extra), junto con el algoritmo de control desarrollado y programado en el lenguaje RAPID, y una etapa de comunicación que permite tener lectura de la señal de fuerza en el software para generar el perfil de fuerza.

La arquitectura del control híbrido posición/fuerza se observa en la Figura 4.1 y está constituida por los siguientes componentes:

4.1. Sensor de fuerza

El sensor de fuerza utilizado fue un mini40 de la marca ATI junto con su controladora FT y su MUX box que es donde contiene toda la etapa de electrónica, ya que el sensor es pequeño para contener todo en su encapsulado, esto se puede observar en la Figura 4.2. La controladora FT permite adquirir la señal de fuerza de cada una de las 6 galgas de fuerza que integran al sensor de fuerza, estas mediciones son multiplicadas por la matriz de calibración y así se obtiene el conteo de fuerza, donde 200 conteos equivalen a un 1N de fuerza. La controladora FT emplea un filtro para mejorar la señal y así permite tener acceso a la señal de fuerza a través de una comunicación serial o como salida analógica en un conector DB 26 (véase Figura 4.3).

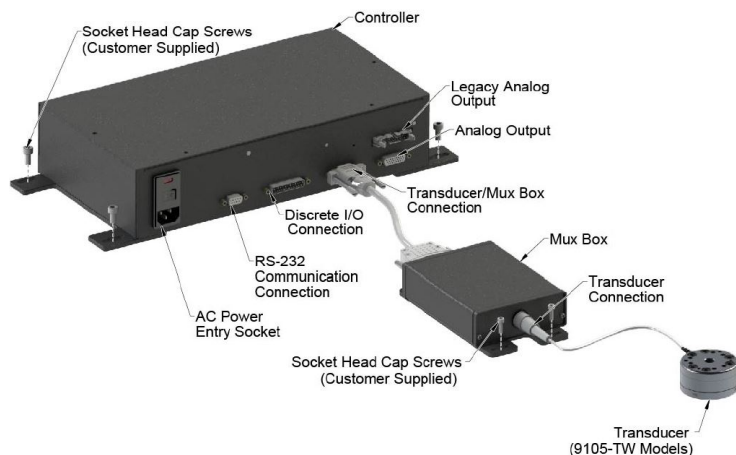


Figura 4.2: Conexión del sensor de fuerza con controladora FT y Mux Box [29].

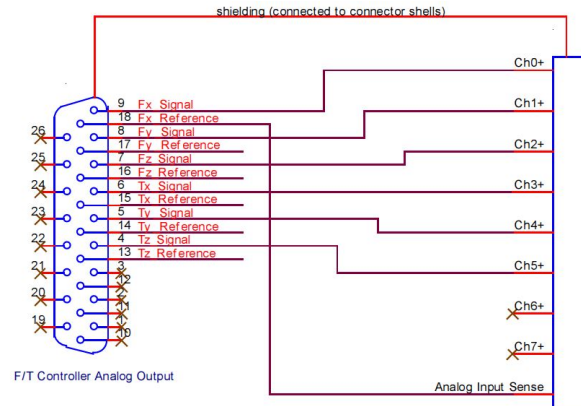


Figura 4.3: Disposición de pines analógicos de salida de la controladora FT [29].

Este sensor de fuerza ATI tiene un rango máximo de medición de fuerza de 80N en F_x , F_y y 250N en F_z , con una resolución de 1v por 48N para la salida analógica.

En la Figura 4.3 se observa la disposición de los pines analógicos de salida del sensor de fuerza, este conector se encuentra en la controladora FT del sensor.

Para instalar el sensor de fuerza se diseñó un efector final en el cual se tiene un contacto directo con el encapsulado del sensor, para tener mediciones de fuerza en el eje z del sensor, este se monta entre la base del porta herramientas y el efector final del robot (véase las Figuras 4.4 y 4.5).



Figura 4.4: Ensamble del sensor de fuerza.

Una vez montado el efector final en el robot industrial, se establece el tool center point (TCP) para que el robot reconozca el efector final instalado, esto se realiza experimentalmente tomando cuatro puntos de referencia al tocar puntualmente la referencia con el

efector final como se observa en la Figura 4.5, además, se establecen los parámetros de la herramienta que se muestran en la Tabla 4.1.

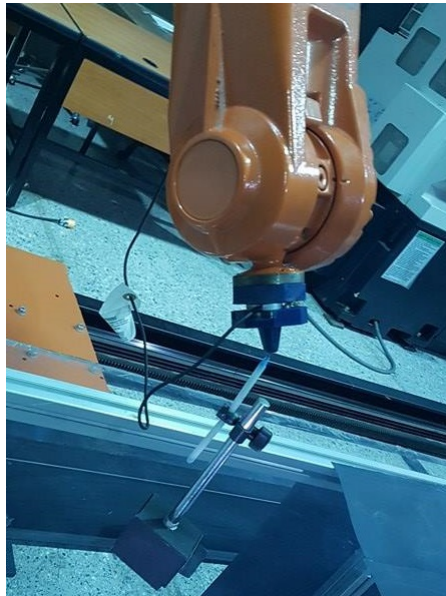


Figura 4.5: Procedimiento para crear el TCP.

Los parámetros del efector final establecidos en la Tabla 4.1 se generan automáticamente con el software del robot, al establecer el TCP encontrando 4 puntos de referencia como se observa en la Figura 4.6, a excepción del valor de la masa que es establecido por usuario.

Método	Cuatro puntos
Máximo error	1.66 [mm]
Mínimo error	1.36 [mm]
Error promedio	0.66 [mm]
Posición x	3.37831 [mm]
Posición y	0.159115 [mm]
Posición z	59.6913 [mm]
Orientación en cuaterniones	[1,0,0,0]
Masa del efector final	0.09 [Kg]
Centro de gravedad	[0,0,1]

Tabla 4.1: Parámetros del TCP.

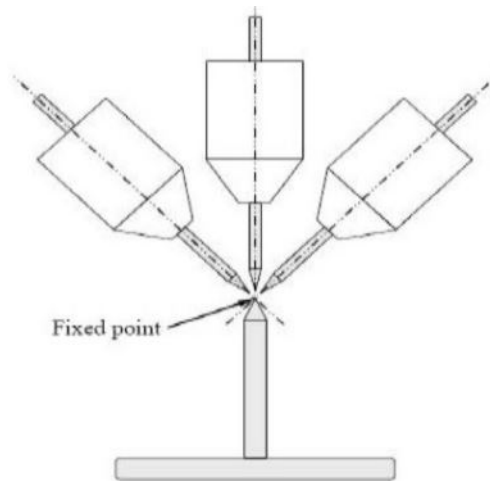


Figura 4.6: Método de los 4 puntos para crear el TCP [30].

4.2. Microcontrolador ATmega2560

El microcontrolador utilizado fue un ATmega2560 de la familia AVR montado en la tarjeta embebida Arduino (véase en la Figura 4.7), de la cual se programa su convertidor analógico/digital para recibir la señal analógica del sensor de fuerza a través de la controladora ATI en sus pines analógicos, para después enviar el dato leído en forma digital utilizando 10 bits de resolución.

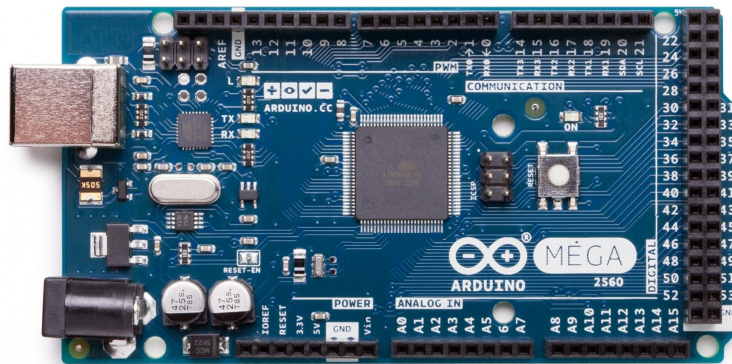


Figura 4.7: Arduino Mega2560 [31].

Ya que el microcontrolador ATmega2560 contiene solo puertos de 8bits, fue necesario concatenar otros dos bits para tener una resolución de 10bits, el programa que implementa el convertidor analógico digital de 10bits se muestra en el Apéndice B.

4.3. Tarjeta de entradas y salidas digitales

La forma en que se comunicó externamente el sensor de fuerza con el robot industrial fue por medio de una tarjeta de E/S digitales DSQC 652 de ABB (véase la Figura 4.8), que está internamente conectada dentro de la unidad de control IRC5, esta tarjeta permite tener en todo momento el valor de 16 señales digitales, que comúnmente se utilizan para conectar sensores de interrupción.



Figura 4.8: Tarjeta de E/S digitales DSQC 652 de la unidad de control IRC5.

4.4. Circuitos de acondicionamiento de las señales

Ya que la señal analógica que proviene del controlador ATI se encuentra en un rango de $+/- 5v$, se mapean estos valores a un rango de $0 - 5v$ para leerlos con el convertidor analógico/digital del microcontrolador, este mapeo está representado por la ecuación lineal (4.1), donde el voltaje de entrada es v_{es} que corresponde a la salida del sensor y v_{ss} es la salida de voltaje que se conecta al convertidor analógico/digital del microcontrolador.

$$v_{ss} = 0,5v_{es} + 2,5 \quad (4.1)$$

El circuito PCB de la Figura 4.9 corresponde a la configuración electrónica mostrada esquemáticamente en la Figura 4.10, donde cada integrado LM32A contiene 4 amplificadores lo que permite tener 6 configuraciones en la PCB, como se muestra en la Figura 4.9, esto permite tener el mapeo de las 6 señales que se pueden leer desde el sensor de fuerza: F_x , F_y , F_z , τ_x , τ_y , τ_z .

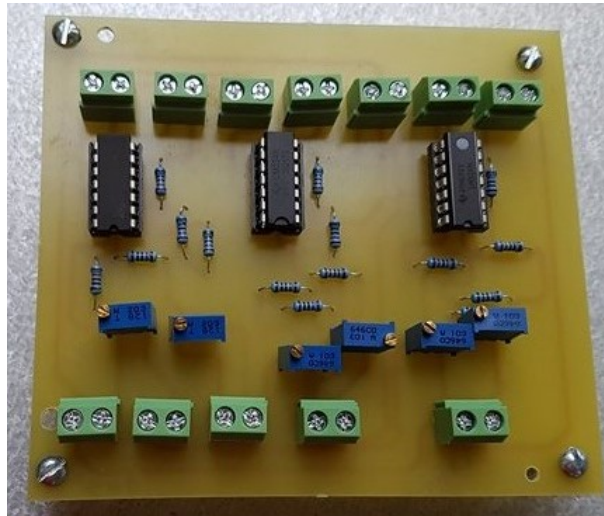


Figura 4.9: PCB del mapeo de la señal de fuerza.

Sin embargo para los experimentos que se muestran en el siguiente capítulo se utilizó únicamente la medición de la fuerza en z , ya que sólo se ejerce la fuerza en esta dirección para la tarea realizada.

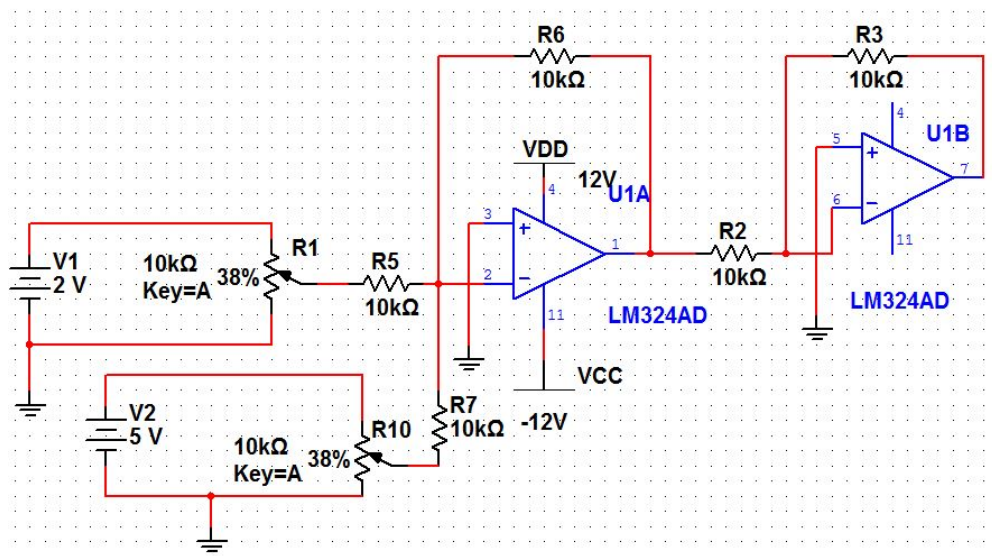


Figura 4.10: Esquema del circuito de mapeo de la señal de fuerza para un canal del sensor.

Otro circuito de acondicionamiento necesario es una etapa de amplificación de las señales digitales de salida del microcontrolador, debido a que la tarjeta de E/S de la controladora IRC5 recibe pulsos de 0 – 24v y la salida del microcontrolador es de 0 – 5v, este circuito se observa en la Figura 4.11 y su esquemático para un canal del sensor de fuerza en la Figura 4.12.

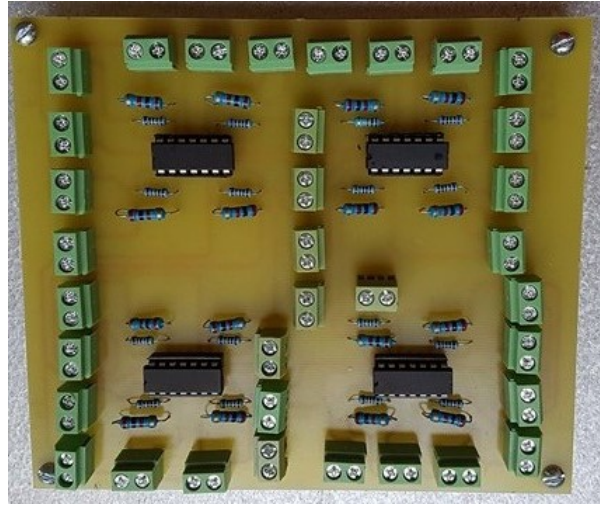


Figura 4.11: PCB del circuito de amplificación.

Con las resistencias propuestas R_1 y R_2 , con un voltaje de entrada v_{di} en forma de señal digital de pulsos de 0 y 5v y con la ecuación (4.2) se amplificó la señal 5 veces, para obtener pulsos de 0 y 24v a la salida del amplificador, así se obtuvo una ganancia G y el voltaje de salida amplificado v_c .

$$v_c = v_{di} * G \quad (4.2)$$

Los valores obtenidos para el diseño de este amplificador se muestran en la Tabla 4.2.

Parámetros	Valor
Resistencia 1	$R_1 = 10k\Omega$
Resistencia 2	$R_2 = 2.7k\Omega$
Ganancia de amplificación	$G = R_1/R_2 = 4.7$
Tensión de entrada	$v_{di} = 5v$
Tensión de salida	$v_c = 23.4 \approx 24v$

Tabla 4.2: Parámetros de diseño del amplificador.

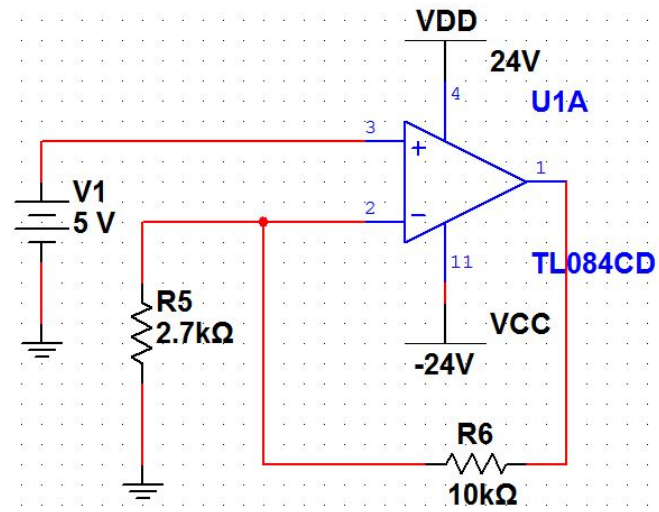


Figura 4.12: Esquema de amplificación para un canal.

Capítulo 5

Casos de estudio: Resultados experimentales

Las pruebas experimentales se realizaron con la finalidad de validar la implementación del esquema de control híbrido posición/fuerza mostrado en la Figura 3.4, por medio de la interfaz de comunicación mostrada en el esquema de la Figura 4.1 que permite dicha implementación.



Figura 5.1: Robot industrial IRB 2400 ABB en estado inicial y en estado de interacción.

La tarea realizada por el robot industrial IRB2400 consiste en partir desde una posición inicial del robot o también llamada home del robot (véase Figura 5.2) respecto al marco de referencia global o local ya que se encuentran en la misma posición, que por medio de su control interno de posición vaya acercándose al punto de contacto con una mesa de acrílico (véase Figura 5.1), con una interpolación de trayectoria que realiza el propio controlador del robot industrial (véase Figura 5.2) y en el instante en que se realiza el contacto, se activa el control híbrido, el cual controla la posición z y la fuerza de reacción F_z sobre

el eje z , (la conmutación de estos modelos se puede observar en forma de diagrama de flujo en la Figura 3.3), una vez que se hace contacto se pueden tener distintos casos de estudio dependiendo del problema de control a considerar en posición y en fuerza, como son regulación y seguimiento de trayectoria.

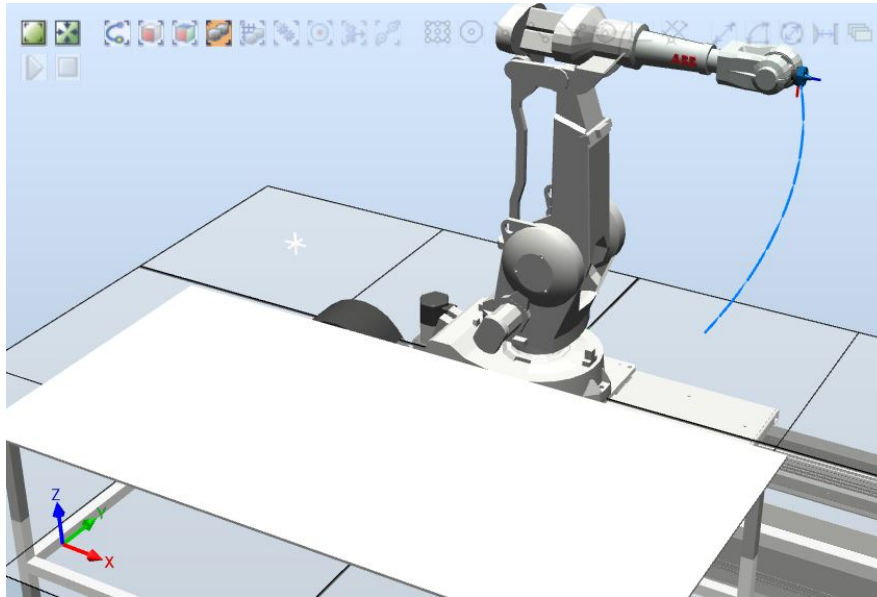


Figura 5.2: Interpolación de trayectoria desde la posición home del robot industrial IRB2400.

De manera independiente, el control de posición interno del robot controla las posiciones x y y , sin tener relación con las fuerzas F_x y F_y , debido a que el experimento consiste en realizar un contacto normal con la superficie de acrílico, con ello se tiene control de la componente de fuerza en la dirección del eje z , respecto al sistema de coordenadas base mostrado en la Figura 5.1. Cuando se realiza control de seguimiento de trayectoria en posición, el robot industrial se desplaza sobre la mesa siguiendo la trayectoria y manteniendo un contacto con dicha superficie con una fuerza de $10N$, que sería equivalente a ejercer una presión con una masa de $1,019Kg$. El motivo por el cual se seleccionó una superficie de acrílico es que la rigidez del material permite un rango considerable de deformación elástica, permitiendo tener una deformación visual con un valor pequeño de masa, el valor de rigidez inicial considerado fue $k_0 = 100000[N/m]$, el cual se extrajo de tablas normadas de rigidez del acrílico y cabe mencionar que este valor fue el punto de partida para estimar de manera experimental un valor adecuado para el material de contacto que se utilizó, ya que este valor inicial es un valor promedio que viene en tablas y no puede ser el valor exacto para todo tipo de acrílico, con este material se evitó tener fuerzas excesivas de reacción que podrían dañar el sensor de fuerza debido a que el sensor que se utilizó es de un bajo rango de mediciones de fuerza y una región pequeña de saturación comparado con sensores de fuerza para aplicaciones industriales, sin embargo permite mostrar la efectividad y desempeño del control híbrido propuesto.

Condición de conmutación entre movimiento libre y movimiento restringido

El caso de estudio considera un plano horizontal (mesa de acrílico) como superficie de contacto y un modelo de impedancia, que es representado por el sensor de fuerza, con el cual se tiene medición de la fuerza en el eje z , que se compara con la fuerza deseada z_d respecto al marco de referencia global, para ello deben aplicarse las siguientes condiciones de conmutación:

Condición de conmutación en posición

Si $z \leq z_d$, la condición respecto al marco de referencia global del robot indica que se penetró en la superficie y se tiene que cerrar el lazo de control para que funcione el control híbrido (movimiento restringido).

Si $z > z_d$, la condición respecto al marco de referencia global del robot indica que no hay contacto todavía con la superficie por lo cual solo debe funcionar el control interno de posición (movimiento libre).

Condición de conmutación en fuerza

Si $F_z > 0$, la condición respecto al marco de referencia global del robot indica que se penetró en la superficie y se tiene que cerrar el lazo de control para que funcione el control híbrido (movimiento restringido).

Si $F_z \leq 0$, la condición respecto al marco de referencia global del robot indica que no hay contacto todavía con la superficie por lo cual solo debe funcionar el control interno de posición (movimiento libre).

Resulta útil considerar la condición de conmutación, vista de la posición y desde la fuerza, ya que se complementan, debido a las condiciones que se presentan en los experimentos, por ejemplo en un supuesto caso donde el sensor de fuerza llegara a tener una pequeña medición en movimiento libre, la cual puede ser interpretada como un contacto y para ello se compara con la condición de conmutación en posición, para verificar si existe contacto o no.

Programación del esquema de control híbrido en RAPID

La programación del esquema de control híbrido como se mencionó en los capítulos anteriores se realizó desde RobotStudio en el lenguaje de programación RAPID, este programa parte de un programa principal que se encuentra en la estructura principal `main()`, como se muestra en el Algoritmo 5.3.

En este programa principal se inicializa la variable tiempo declarada como `tt2`, por medio de la instrucción `pos1:=CPos()`; se lee el valor de las posiciones cartesianas x , y y z y se establece la condición de conmutación definida por un `IF`, donde si se cumple la condición $z \leq z_d$ el programa llama a la rutina `ControladorPI`; el cual contiene el algoritmo de control PI de fuerza basado en una aproximación integral trapezoidal, en caso contrario con la instrucción `moveJ` ejecutará el control interno de posición siguiendo las posiciones deseadas iniciales x_d , y_d , z_d , que representan la posición inicial de contacto en la mesa de acrílico.

```

!-----
!Programa principal
PROC main()
  k:=k+1;!Conteo de ciclos
  ClkStart time2;!Inicio del Reloj de RAPID
  tt2:=ClkRead(time2)!Variable de tiempo
  t_vector2{k}:=tt2;
  pos1:=CPos(); !Vector que lee las posicipones x y z
  x:=pos1.x;!Lectura de la posición x del efector final
  y:=pos1.y;!Lectura de la posición y del efector final
  z:=pos1.z;!Lectura de la posición z del efector final
]
  IF z<=zd THEN
    ControladorPI;
  ELSE
MoveJ [[xd,yd,zd],[0.00171871,-0.779389,-0.626535,-0.00193248],[-1,0,0,0]
  ENDIF
ENDPROC

```

Algoritmo 5.3: Programa principal del control híbrido posición/fuerza.

En el Algoritmo 5.4 se muestra el fragmento del código que contiene la rutina `ControladorPI()`; en el cual primero se manda a llamar la subrutina `Datos1`; que se encarga de leer la señal de fuerza a través de las entradas digitales de la unidad de control IRC5, en esta rutina se implementó un algoritmo para leer los datos a través de las funciones `Bitset` y `Bitclear` para indicar el valor binario en 10 bits que componen una variable numérica y así obtener en la variable `sensor_F` el valor numérico de la señal de fuerza. Una vez que se tiene el valor de fuerza en la variable `FR` se genera el error de fuerza, se integra por medio de una suma que se obtiene al calcular la suma de rectángulos o trapecios, de esta manera se establece una ganancia integral k_i que multiplica al error integral y se suma con el error de fuerza por una ganancia proporcional, que dará el valor de salida del controlador o la penetración que tendrá el robot en la posición z , después obtiene la posición comandada z_c que es el aumento de la posición deseada z_d con la salida del controlador `Pref`, ya que el eje z del marco de referencia global tiene dirección contraria al eje z del marco de referencia de la herramienta o del efector final, la posición comandada resulta ser una diferencia como se observa en el código. En la parte final de la rutina se escriben las ecuaciones paramétricas de la trayectoria de posición para generar las posiciones deseadas actuales a seguir, y tanto las posiciones deseadas que salen de estas trayectorias paramétricas como la posición comandada se establecen en el control de posición descrito por la función `moveJ`.

Los códigos de RAPID para cada caso de estudio se anexan en el Apéndice B.

```

PROC ControladorPI()
Datos1;!Rutina para adquirir los datos de la señal de fuerza
sensor_F:=DnumToNum(SF);!valor del sensor de fuerza mini40
FR:=((sensor_F*5)/1024)*24-0.1;!Fuerza de reacción
FR_vector{i}:=FR;
e:=Fd-FR;!Error de fuerza
e_vector{i}:=e;
!Error integral
ie:=ia+(h/2)*(e+ea);
!Salida del controlador
Pref:=kp*e+ki*ie;
IF Pref*1000>=2.5 THEN !Saturador de posición
Pref:=2/1000;
ENDIF
Penetracion_vector{i}:=Pref;
!Actualización de variables-----
i:=i+1;! variable de ciclos
ia:=ie;
!-----
zc:=zd-Pref*1000;!Posición comandada
PF_vector{i}:=zc;
e_posicion{i}:=zc-z;!Error de posición
!Generación de trayectoria circular-----
theta:=theta+1;
xd:=cx+rx*cos(theta);
yd:=cy+ry*sin(theta);
MoveJ [[xd,yd,zc],[0.00171871,-0.779389,-0.626535,-0.00193248],[-1,0,0,0]

```

Algoritmo 5.4: Rutina que implementa el controlador PI de fuerza.

Definición del área de trabajo sobre la superficie de acrílico

La región seleccionada para realizar los experimentos fue definida en base al alcance, ubicación del robot y sin estar cerca de los límites de carrera de cada articulación del mismo, esta área es un rectángulo de 684[mm]x 800.8[mm] que se observa en la Figura 5.5, con su sistema de coordenadas que tiene las mismas direcciones y sentidos que el sistema de referencia base del robot.

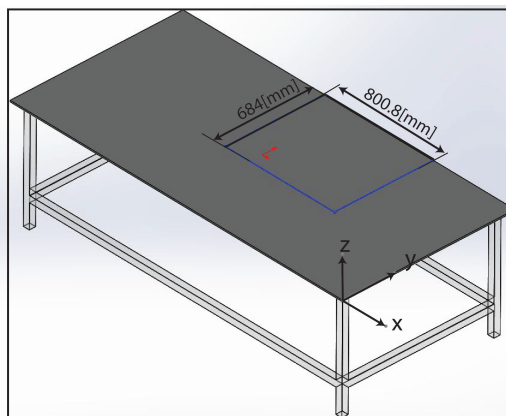


Figura 5.5: Área de trabajo dentro de la mesa de acrílico.

Cabe mencionar que esta área de trabajo que fue definida sobre la mesa de acrílico no es utilizada por completo y los experimentos se realizaron en distintos puntos de contacto dentro de esta área, ya que el material de acrílico no presenta uniformidad en toda su superficie.

Los resultados para los 3 casos que se explicarán a continuación, se obtuvieron primeramente en base a simulaciones, con la finalidad de emular el comportamiento del robot industrial y evitar posibles errores y daños en los experimentos reales. El comportamiento que tendría el robot industrial ABB se simuló por medio de RobotStudio, donde se programó exactamente igual que el programa que funciona con el robot real, con la diferencia de que en el programa de simulación la fuerza de reacción se obtiene del modelo de impedancia capacitiva definido en la ecuación (2.24) donde $X = z$, $k = k_0$ y siendo así resulta ya no ser necesario utilizar la rutina Datos1(); en simulación, que se encarga de aplicar el algoritmo de adquisición de datos digitales de la señal de fuerza.

Periodo de muestreo en la comunicación de tiempo real

Un problema importante que cabe resaltar dentro de la implementación, es debido al periodo de muestreo mínimo que se puede utilizar, el cual es aproximadamente entre $5[ms]$ y $10[ms]$ debido a las características del sistema operativo, software y tarjetas de memoria de la unidad de control IRC5 que cuenta el robot industrial ABB utilizado, sin embargo no se pudo utilizar este periodo de muestreo, ya que al utilizar el control de posición del robot se genera gran consumo de procesamiento en el sistema, lo que provoca tener un retardo en la señal, entonces para los experimentos realizados se comprobó que el periodo mínimo a utilizar fue aproximadamente $40[ms]$, para que no se perdieran lectura de datos, específicamente esto es provocado cuando se ejecuta la instrucción `moveJ`, debido a que esta instrucción no solo implementa el control de posición interno, sino también una interpolación de trayectoria provocando un aumento del ciclo de procesamiento.

Los experimentos fueron realizados para resolver los siguientes tres casos:

5.1. Caso 1: Regulación de posición y regulación de fuerza

En este caso se resolvió el problema de regulación tanto en fuerza como en posición, buscando una posición deseada sobre el eje z , posiciones deseadas fijas de x y y y una fuerza deseada de contacto con dirección sobre el eje z , como el control híbrido en este caso controla la fuerza en el eje z , se obtiene como posición final del robot la posición comandada z_c , que es la posición deseada z más la penetración debido a la fuerza ejercida sobre la superficie de acrílico. En la Tabla 5.1 se muestran las posiciones, orientaciones del efector final y fuerza deseada para el experimento del caso 1.

Posición deseada	Orientación deseada
$x_d = 468.14[mm]$	$\alpha = 180^\circ$
$y_d = -1205.94[mm]$	$\beta = 0^\circ$
$z_d = 367.4[mm]$	$\gamma = 77.6^\circ$
Velocidad constante	Fuerza deseada
$V_{cte} = 300[mm/s]$	$F_d = 10[N]$

Tabla 5.1: Variables deseadas para el caso de estudio 1.

En la Figura 5.6 se tiene una gráfica en 3D de la trayectoria interpolada que genera el propio robot desde su posición inicial hasta el punto de contacto, con solo indicar la posición deseada en la instrucción `moveJ`. Los parámetros del experimento y ganancias del control PI de fuerza se presentan en la Tabla 5.2.

Ganancia derivativa	$k_p = 6.25 \times 10^{-5}$
Ganancia integral	$k_i = 3.75 \times 10^{-4}$
Paso de integración	$h = 0.01$
Dimensiones de la mesa de acrílico	$120 \times 244 \times 367[mm]$
Posición inicial del robot industrial	$x=1000[mm], y=0.0[mm], z=1451.5[mm]$
Tiempo de ejecución	$65[s]$

Tabla 5.2: Parámetros del experimento y ganancias del control PI para el caso 1.

La evolución de posición z del efector final se puede observar en la Figura 5.7, así como su respectivo error, las posiciones x y y no se muestran ya que se establecen en el robot como posiciones fijas y el cambio tanto en posición como en fuerza solo es respecto al eje z .

En las gráficas de posición y fuerza en las Figuras 5.7 y 5.8 se observa que empiezan aproximadamente en el segundo 6.5, debido a que es el tiempo que tarda el robot en llegar al punto de contacto y es el instante en donde inicia el control híbrido. En la Figura 5.7 se aprecia que tanto en la posición z como en el error e_z se tiene una oscilación entre los segundos 7 y 10, debido al instante en que el robot hace contacto con la superficie, que es la etapa de impacto, donde el controlador tarda aproximadamente 8 segundos en estabilizar la posición, este tiempo es debido a que se tuvo un bajo periodo de muestreo en la señal de fuerza de 30ms.

El error de posición e_z mostrado en la Figura 5.7 es el resultado de la diferencia de la posición actual y la posición comandada.

La fuerza de reacción F_R y el error de fuerza e_F se muestran en la Figura 5.8, donde se observa que la señal contiene cierto valor de ruido no considerable para la tarea realizada, y presenta un error un poco mayor cuando termina la ejecución de la tarea, debido a que el robot detiene de manera instantánea los motores y provoca una pequeña oscilación.

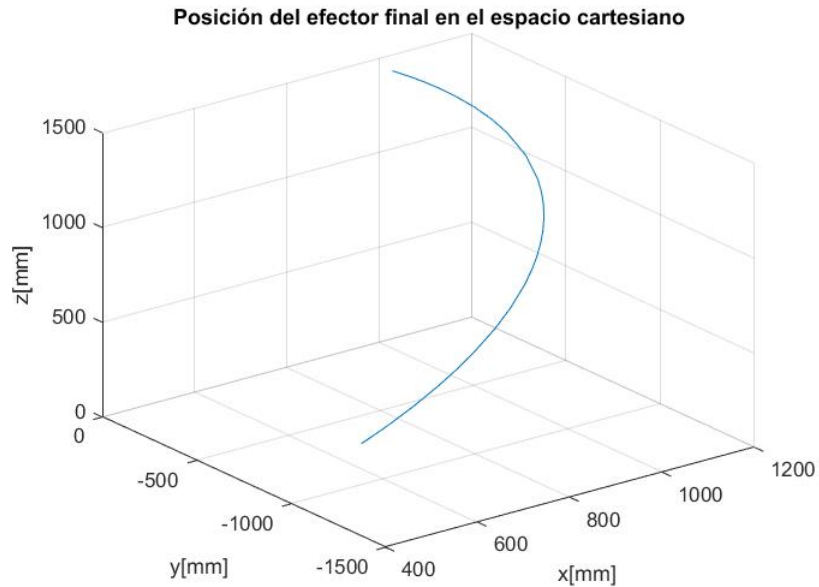


Figura 5.6: Desplazamiento del robot industrial en el espacio cartesiano para el caso 1.

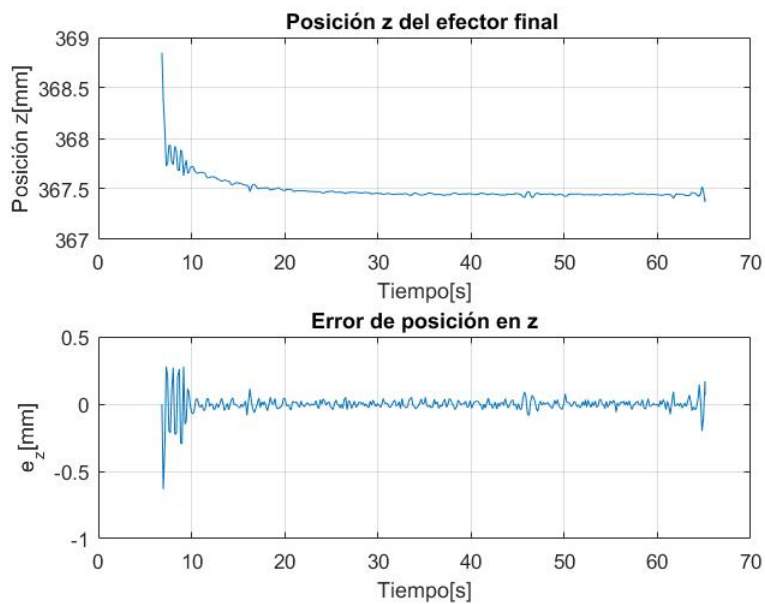


Figura 5.7: Graficas de posición y error de posición en el eje z para el caso 1.

La Figura 5.9 presenta la comparación de la posición actual y la posición comandada, donde se aprecia que se tiene una diferencia de magnitud entre estas dos señales, debido a que la posición comanda es el valor deseado z_d más las penetración que se requiere para mantener el contacto con una fuerza de $10[N]$.

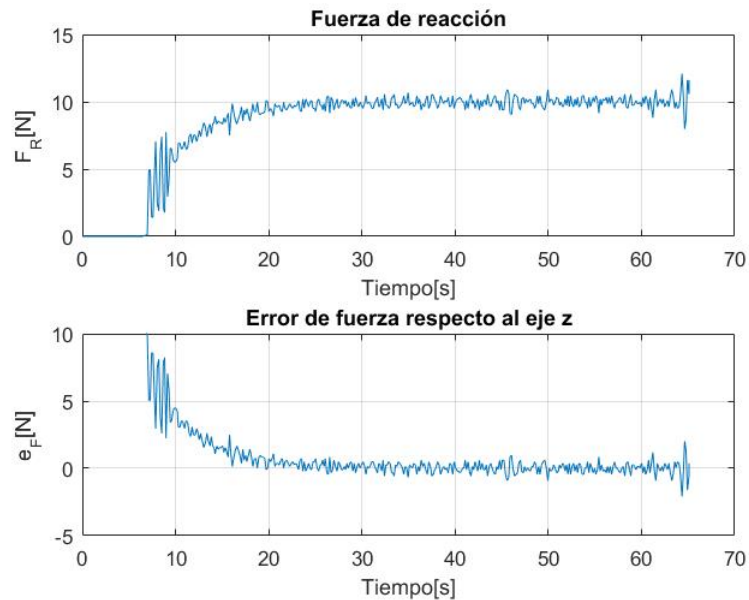


Figura 5.8: Graficas de fuerza y error de fuerza en el eje z para el caso 1.

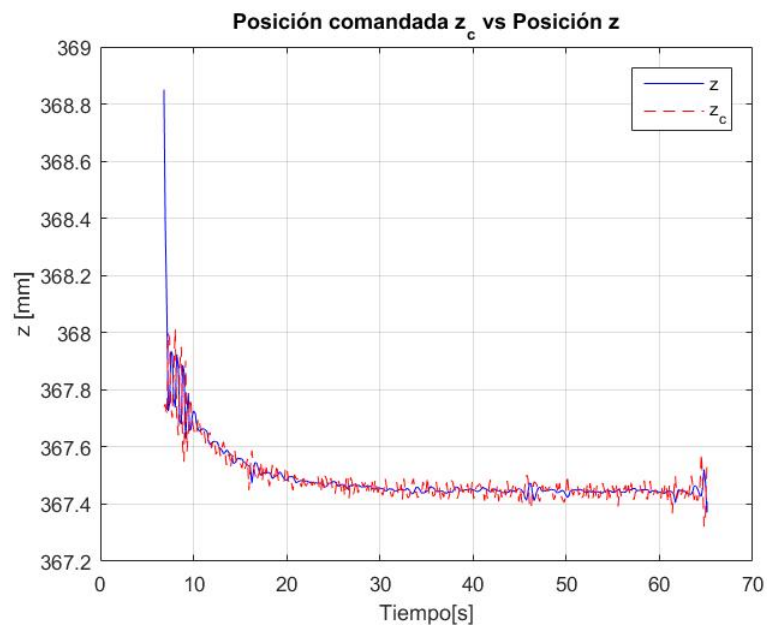


Figura 5.9: Comparación de la posición comandada y la posición actual para el caso 1.

5.2. Caso 2: Seguimiento de trayectoria de posición y regulación de fuerza

En este caso fue resuelto el problema de control de regulación en el control de fuerza y el problema de seguimiento de trayectoria en el control de posición, las trayectorias a seguir fueron definidas en el plano $x - y$ por lo cual se tiene un control de posición independiente entre las trayectorias definidas en el plano $x - y$ y el control de posición z que interactúa directamente con el control de la fuerza F_z , se siguieron trayectorias de funciones conocidas como son la línea recta, circunferencia y elipse.

En el caso 2 la fuerza deseada es la misma que en el caso 1, $F_d = 10[N]$, y las trayectorias a seguir son formas paramétricas de las siguientes funciones:

Diseño de la trayectoria línea recta:

$$x_d = \frac{x_0 - x_n}{2} \cos(\omega t) + \frac{x_n + x_0}{2}$$

$$y_d = \frac{y_0 - y_n}{2} \cos(\omega t) + \frac{y_n + y_0}{2}$$

Los parámetros de la ecuación paramétrica de la recta se observan en la Tabla 5.3.

Posición inicial de x	$x_0 = 468.14[mm]$
Posición inicial de y	$y_0 = -914.9[mm]$
Posición final de x	$x_n = 221.1[mm]$
Posición final de y	$y_n = -914.9[mm]$
Tiempo de ejecución	$t = 100[s]$

Tabla 5.3: Parámetros de la ecuación de línea recta paramétrica.

Diseño de la trayectoria circunferencia

$$x_d = a + r \cos(\theta)$$

$$y_d = b + r \sin(\theta)$$

Para obtener una ecuación paramétrica de la circunferencia como nuestra trayectoria deseada a seguir, se establecen sus parámetros en la Tabla 5.4.

Diseño de la trayectoria elipse

$$x_d = c_x + r_x \cos(\theta)$$

$$y_d = c_y + r_y \sin(\theta)$$

5.2. CASO 2: SEGUIMIENTO DE TRAYECTORIA DE POSICIÓN Y REGULACIÓN DE FUERZA

Posición del centro en x	$a = 221.3[mm]$
Posición del centro en y	$b = -1039.9[mm]$
Radio de la circunferencia	$r = 50[mm]$
Cambio angular	$\theta = 0^\circ - 1080^\circ$

Tabla 5.4: Parámetros de la ecuación de circunferencia paramétrica.

Radio del eje mayor	$r_x = 100[mm]$
Radio del eje menor	$r_y = 12.5[mm]$
Posición del centro en x	$c_x = 221.3[mm]$
Posición del centro en y	$c_y = -1039.9[mm]$
Cambio angular	$\theta = 0^\circ - 1080^\circ$

Tabla 5.5: Parámetros de la ecuación de elipse paramétrica.

Para obtener una ecuación paramétrica de la elipse como nuestra trayectoria deseada a seguir, se establecen sus parámetros en la Tabla 5.5.

Resultados siguiendo una trayectoria de línea recta, regulando una fuerza de 10N

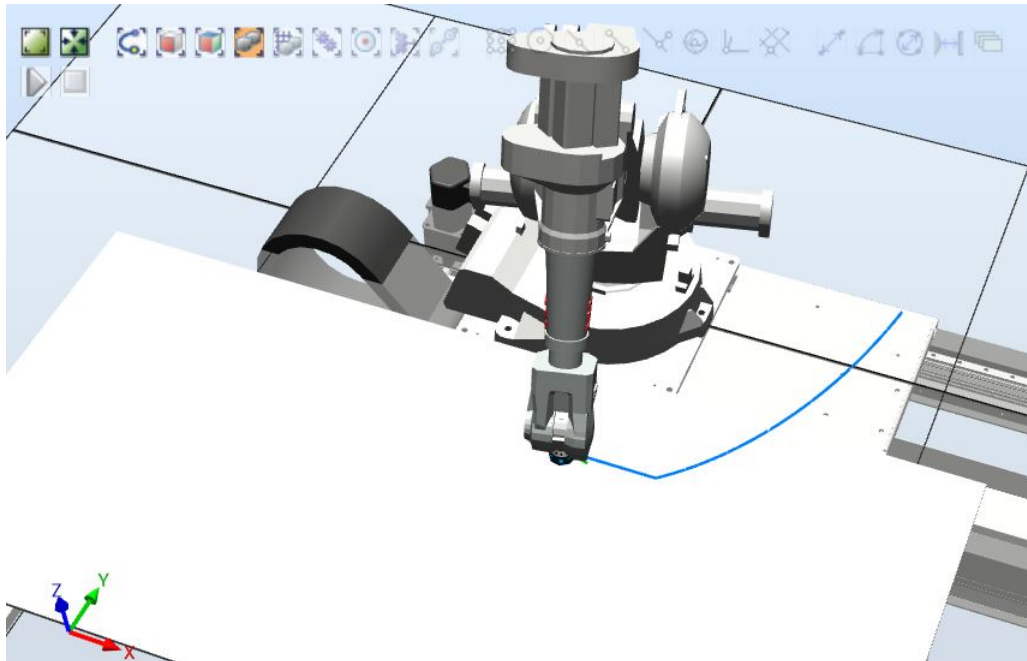


Figura 5.10: Robot industrial IRB2400 ABB siguiendo una trayectoria recta.

En la Figura 5.11 se muestra la trayectoria completa ejecutada por el robot industrial siguiendo una trayectoria recta en el plano xy (véase Figura 5.10), en la cual se aprecian

las dos etapas de movimiento, el movimiento libre que es la misma trayectoria interpolada por el software del robot que en el caso 1 y el movimiento restringido donde sigue una línea recta en el plano xy . Las ganancias y el paso de integración utilizados para este experimento se muestran en la Tabla 5.6.

La posición z presenta un aumento constante de penetración en la superficie de acrílico debido a una pequeña inclinación que presenta esta superficie (véase en la Figura 5.12), por ello el controlador adapta la penetración en base a esta inclinación para mantener la fuerza de contacto de $10N$, además que el acrílico presenta irregularidades en su superficie provocando algunos picos de error de posición y de fuerza (véase Figura 5.12 y 5.13).

Ganancia proporcional	$k_p = 1.25 \times 10^{-5}$
Ganancia integral	$k_i = 3.75 \times 10^{-4}$
Paso de integración	$h = 0.01[s]$
Tiempo de ejecución	$t = 116[s]$

Tabla 5.6: Parámetros del controlador PI de fuerza para el seguimiento de una línea recta en posición par el caso 2.

En la Figura 5.13 se observa que la señal de fuerza presenta un ruido de pequeña amplitud y un error pequeño que se encuentra aproximadamente entre $+/- 0.4[N]$.

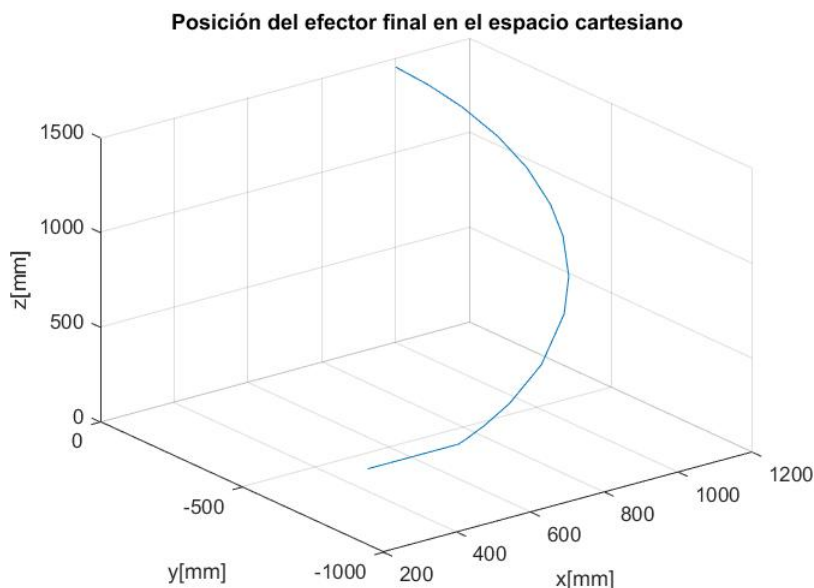


Figura 5.11: Posición completa del robot industrial en el espacio cartesiano para el seguimiento de una trayectoria recta en el caso 2.

5.2. CASO 2: SEGUIMIENTO DE TRAYECTORIA DE POSICIÓN Y REGULACIÓN DE FUERZA

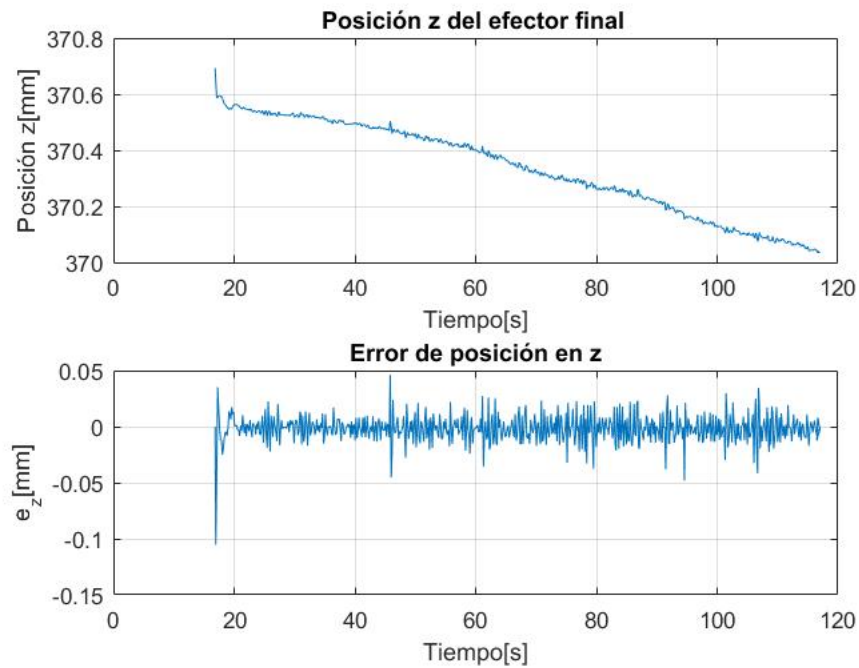


Figura 5.12: Graficas de posición y error de posición para el caso 2, siguiendo una trayectoria recta en posición.

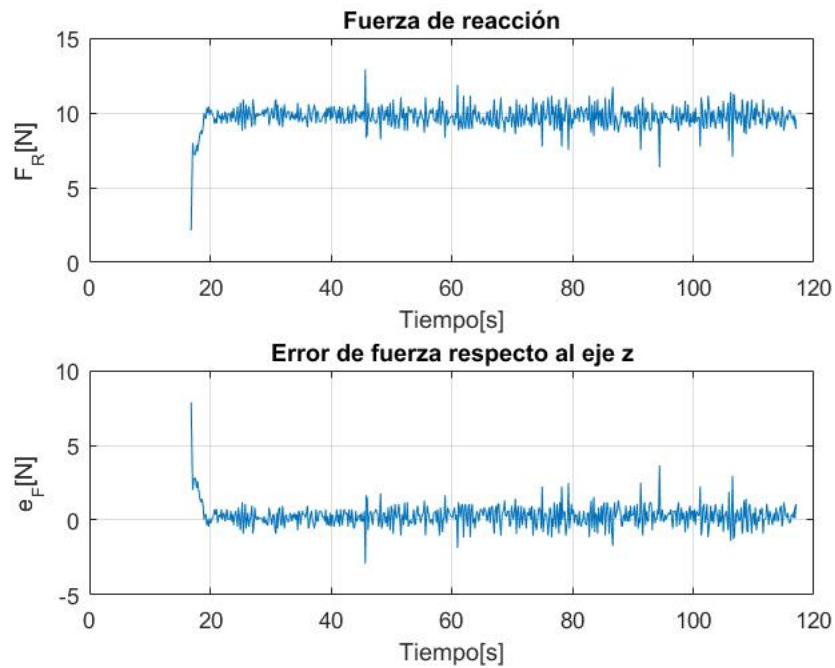


Figura 5.13: Graficas de fuerza y error de fuerza para el caso 2, siguiendo una trayectoria recta en posición.

La trayectoria deseada en el plano $x - y$ se puede observar en la Figura 5.14 como una línea de color azul y la línea punteada en color rojo es el resultado de la trayectoria que se obtuvo, que a simple vista parece no ser una recta, sin embargo la variación que tiene esta dentro del rango de $\pm 0.01[mm]$ que es un valor de distancia muy pequeño y puede ser despreciado, esto quiere decir que tiene un ligero movimiento en el eje y pero sigue correctamente la línea sobre el eje x .

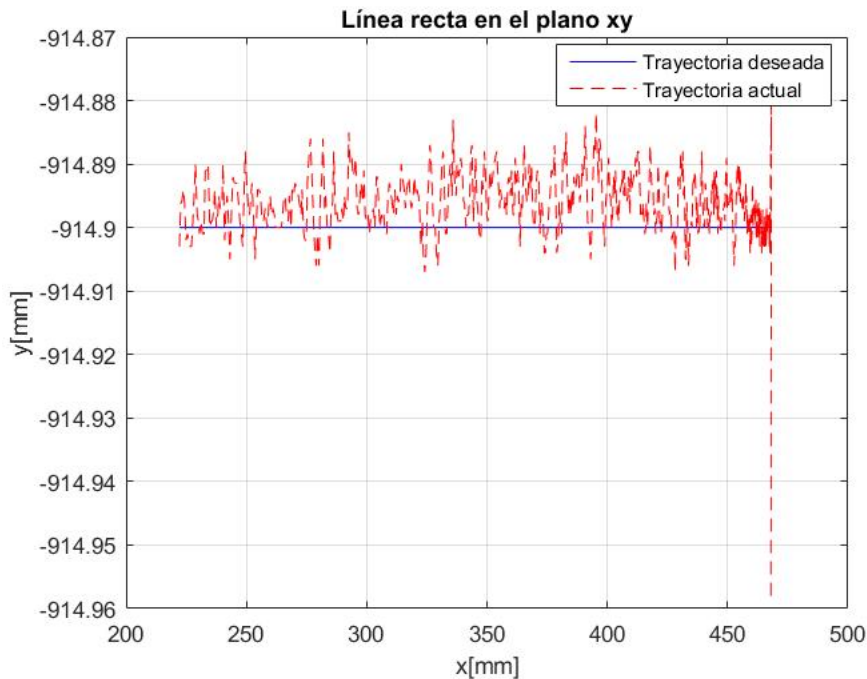


Figura 5.14: Seguimiento de una trayectoria recta en posición para el caso 2.

En la Figura 5.15 se presenta la posición comandada z_c que como era de esperarse evoluciona de la misma forma que la posición z con un error muy pequeño en decimas de milímetro.

A continuación se presentan las gráficas de las Figuras 5.16, 5.17 y 5.18 para corroborar que el comportamiento que presenta el seguimiento de la trayectoria recta tiene una inclinación debido a la superficie y no por cuestiones del controlador o de la implementación del mismo, para ello se presenta en estas gráficas el seguimiento de la misma trayectoria con la misma fuerza deseada pero, iniciando en el punto donde termina la trayectoria recta anterior, con las mismas dimensiones y ubicación, con la finalidad de observar que la recta presenta una inclinación contraria en el eje z y empieza con mayor penetración para ir disminuyéndola debido a la inclinación de la superficie.

5.2. CASO 2: SEGUIMIENTO DE TRAYECTORIA DE POSICIÓN Y REGULACIÓN DE FUERZA

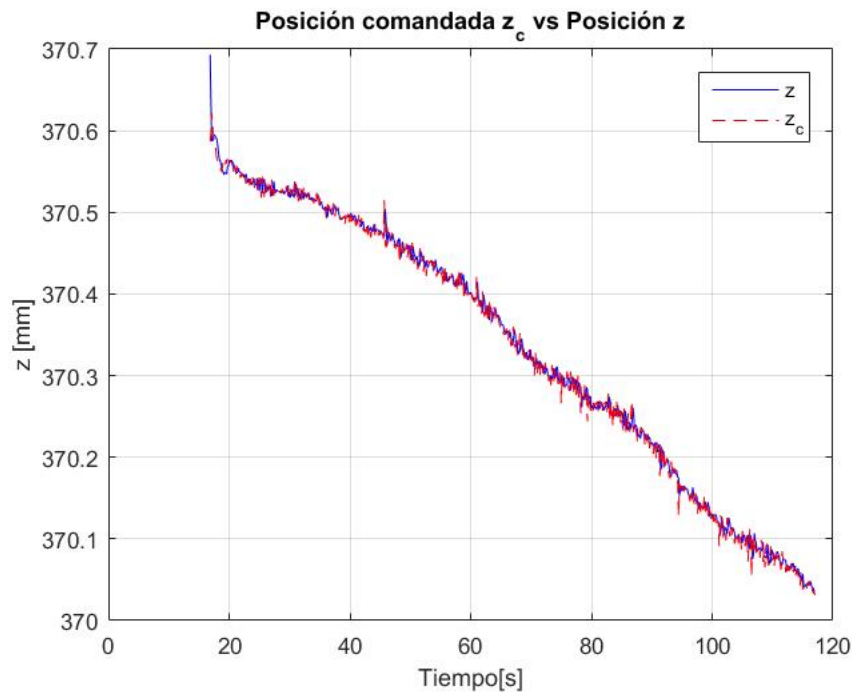


Figura 5.15: Comparación de la posición comandada con la posición actual para el caso 2, siguiendo una trayectoria recta de posición.

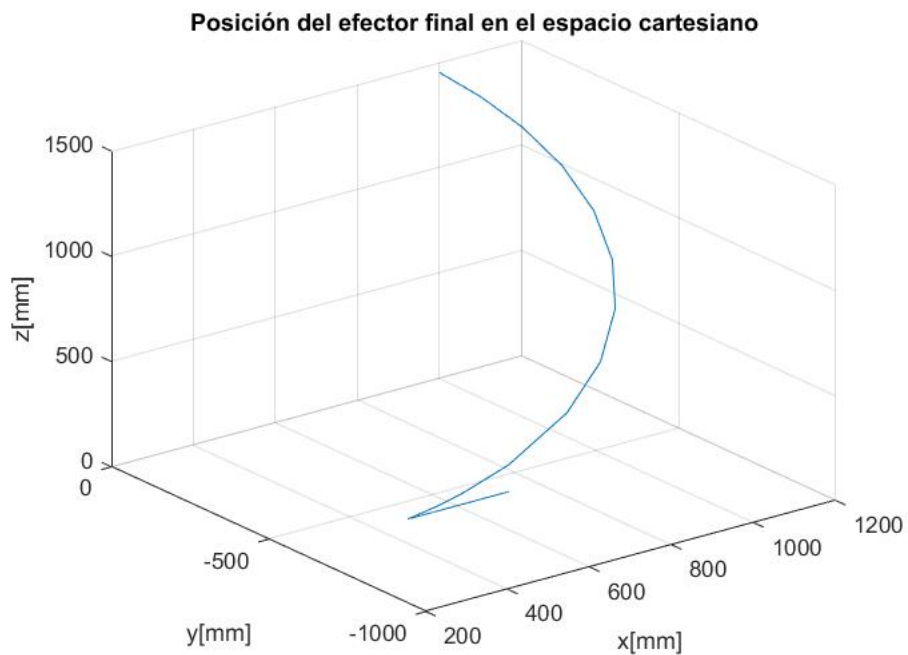


Figura 5.16: Posición completa del robot industrial en el espacio cartesiano para el seguimiento de una trayectoria recta con sentido contrario en el caso 2.

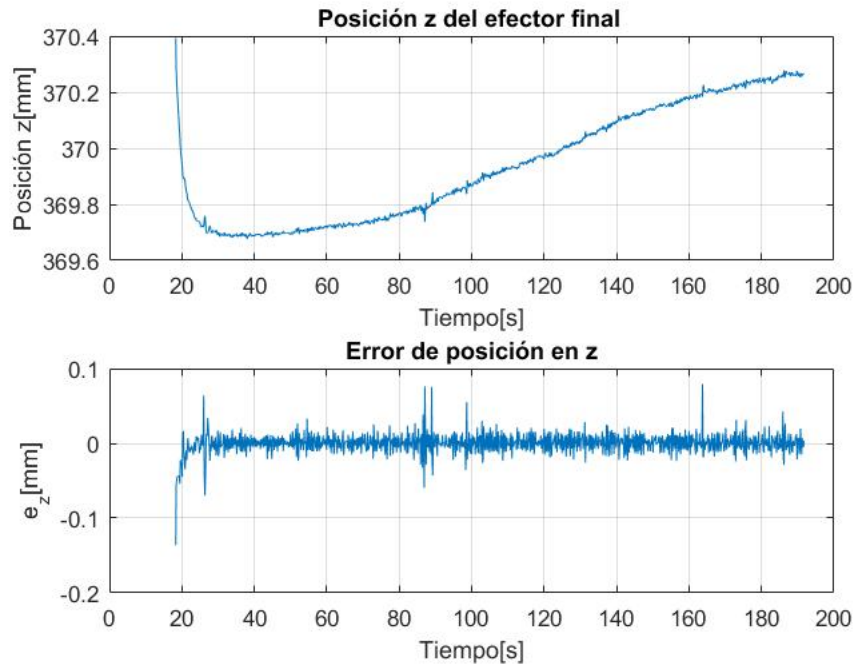


Figura 5.17: Graficas de posición y error de posición para el caso 2, siguiendo una trayectoria recta en posición con sentido contrario.

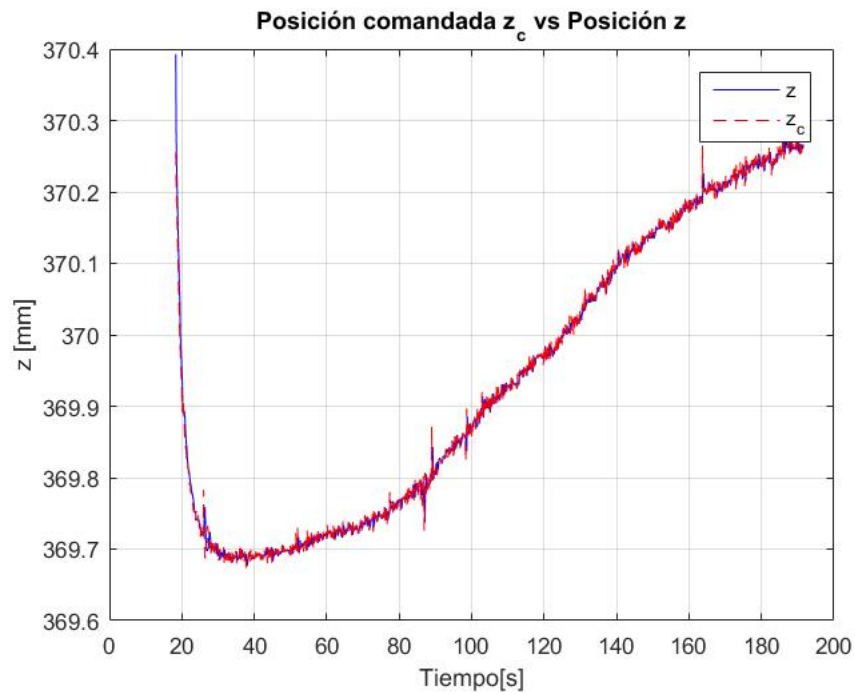


Figura 5.18: Comparación de la posición comandada con la posición actual para el caso 2, siguiendo una trayectoria recta de posición con sentido contrario.

5.2. CASO 2: SEGUIMIENTO DE TRAYECTORIA DE POSICIÓN Y REGULACIÓN DE FUERZA

Resultados siguiendo una trayectoria de circunferencia, regulando una fuerza de $10N$

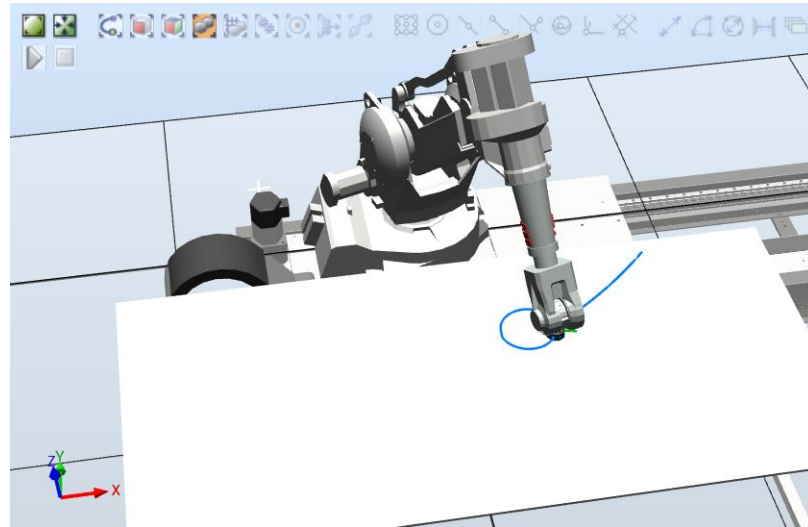


Figura 5.19: Robot industrial IRB2400 ABB siguiendo una trayectoria circular.

En la Figura 5.20 se observa la trayectoria completa cartesiana seguida por el efector final del robot, donde realiza su propia interpolación de trayectoria para llegar a la superficie de contacto en el plano $x - y$, para generar el seguimiento de la circunferencia de radio $r = 50[mm]$. El movimiento que realiza el robot durante la ejecución del experimento se observa en la Figura 5.19 y los parámetros del control PI de fuerza para este caso se muestran en la Tabla 5.8.

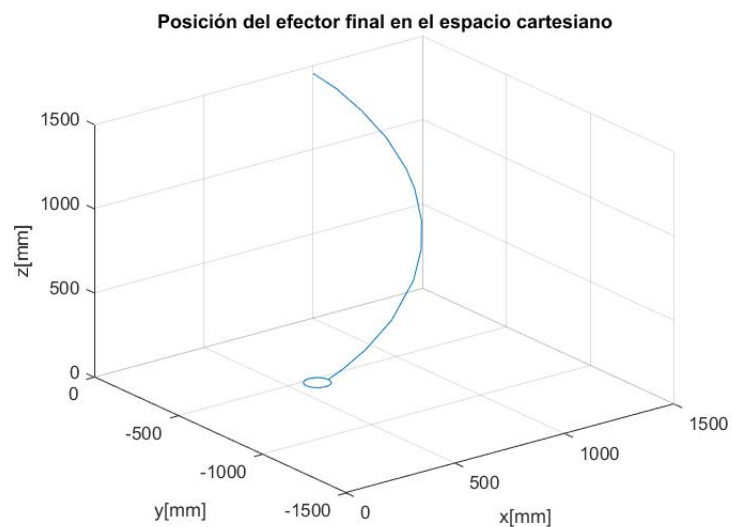


Figura 5.20: Posición cartesiana del robot industrial siguiendo una trayectoria circular para el caso 2.

Ganancia proporcional	$k_p = 1.1 \times 10^{-6}$
Ganancia integral	$k_i = 14 \times 10^{-4}$
Paso de integración	$h = 0.01[s]$
Tiempo de ejecución	$t = 190[s]$
Numero de vueltas	3

Tabla 5.7: Parámetros del controlador PI de fuerza para el seguimiento de una circunferencia en posición.

En las Figuras 5.19 y 5.20 se observa que se siguieron trayectorias pequeñas, esto fue debido a las pequeñas irregularidades que tenía toda la superficie a lo largo de su material, que afectaban el rendimiento del controlador, entonces se consideró una región pequeña donde fuera un caso más ideal.

La trayectoria de la circunferencia es de un radio pequeño en comparación al área de trabajo disponible, se seleccionó esta dimensión debido a las irregularidades de la superficie que provocan no tener un buen desempeño en el seguimiento de posición y provocando picos altos en la fuerza.

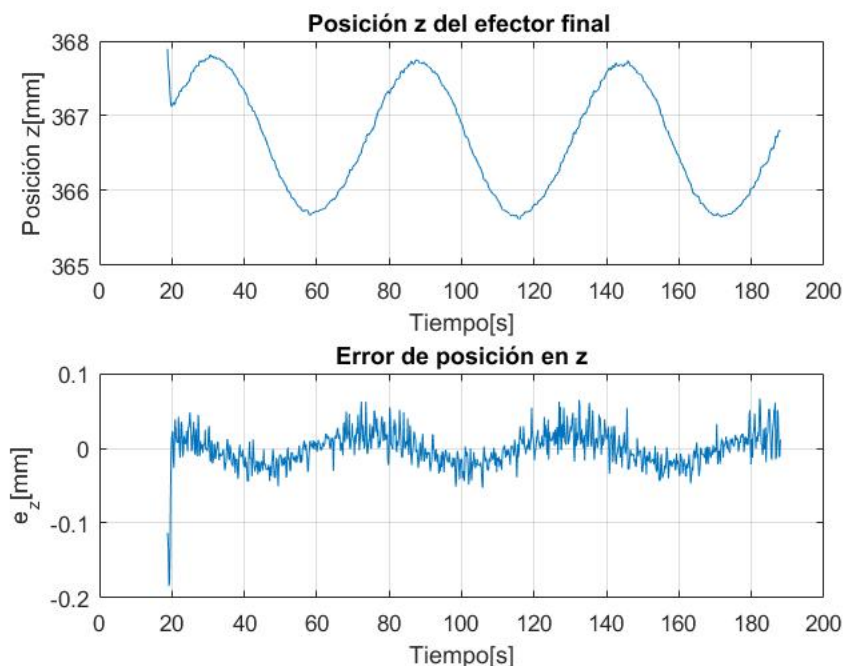


Figura 5.21: Graficas de posición y error de posición respecto al eje z siguiendo una trayectoria circular para el caso 2.

Como el experimento se realizó dando 3 vueltas a la circunferencia, se aprecia en la Figura 5.21 que la posición z presenta las mismas variaciones que un comportamiento cosenoidal,

5.2. CASO 2: SEGUIMIENTO DE TRAYECTORIA DE POSICIÓN Y REGULACIÓN DE FUERZA

además se presentan pequeñas irregularidades en el seguimiento de trayectoria de posición, debido a que el acrílico al colocarlo sobre soportes espaciados provocan una ligera deformación en distintas zonas del material, sin embargo en el error de fuerza e_F se observa que se mantiene la regulación de fuerza con un error promedio aproximado de $+/- 2[N]$, con alguno picos que llegan hasta $4N$, sin embargo no se pierde la regulación aunque se tengan que seguir las irregularidades.

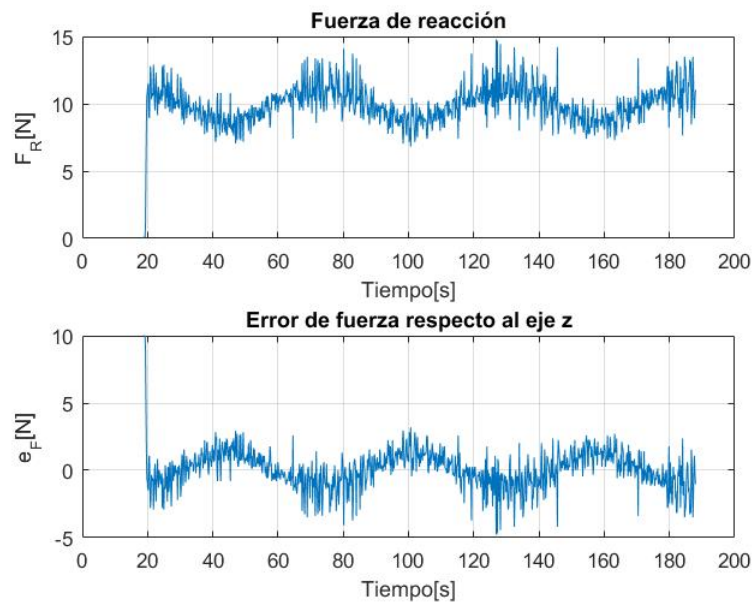


Figura 5.22: Graficas de fuerza y error de fuerza respecto al eje z siguiendo una trayectoria circular para el caso 2.

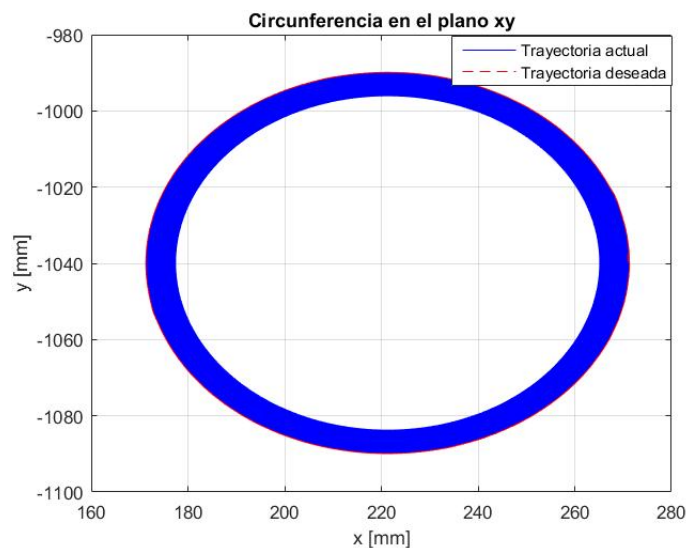


Figura 5.23: Seguimiento de una trayectoria circular en posición para el caso 2.

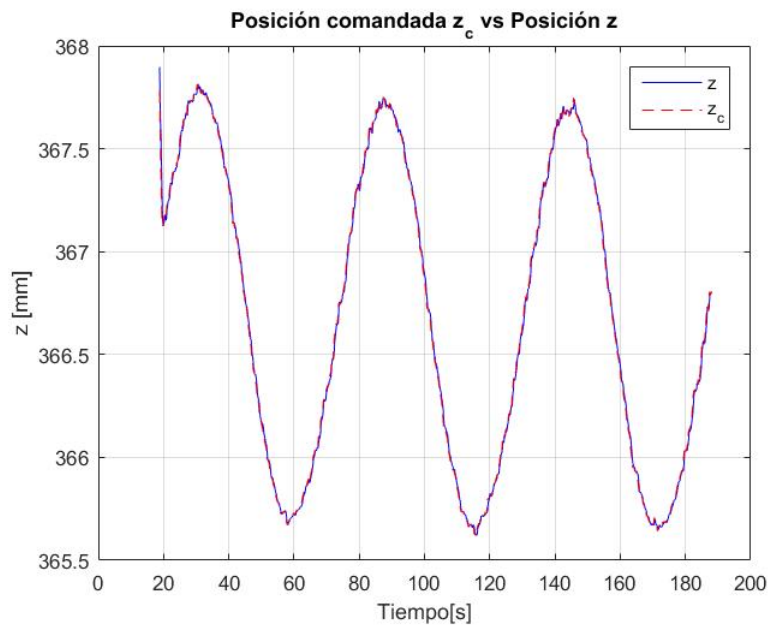


Figura 5.24: Comparación de la posición comandada con la posición actual para el caso 2 siguiendo una trayectoria circular en posición.

En la Figura 5.24 se tiene la comparación de la posición comandada vs la posición actual en z , en la cual se comprueba la efectividad del control interno de posición con el que cuenta este robot industrial, ya que permite tener un muy buen seguimiento en posición lo que provoca a su vez no tener un buen comportamiento en el control de fuerza.

Resultados siguiendo una trayectoria de elipse, regulando una fuerza de 10N

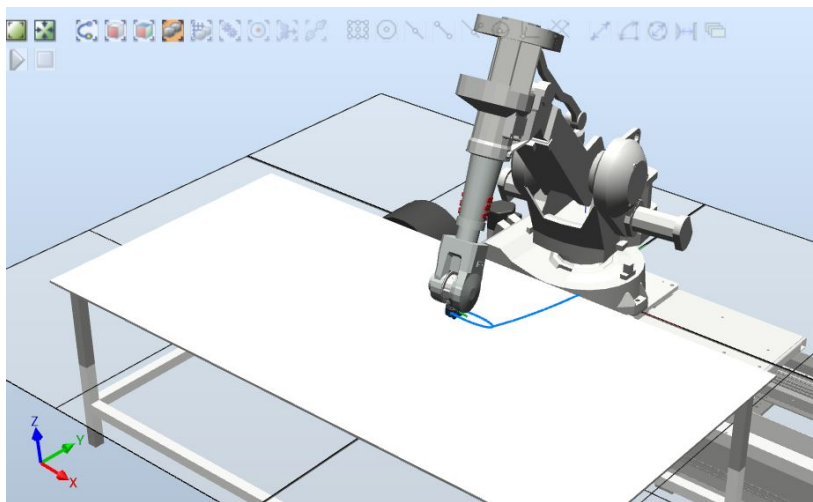


Figura 5.25: Robot industrial IRB2400 ABB siguiendo una trayectoria elíptica en posición.

5.2. CASO 2: SEGUIMIENTO DE TRAYECTORIA DE POSICIÓN Y REGULACIÓN DE FUERZA

A continuación se presentan los resultados donde el robot industrial sigue una trayectoria elíptica (véase Figuras 5.25 y 5.26). En base a los resultados en forma de graficas se dedujo que el desempeño que tiene el controlador en el seguimiento de este tipo de trayectorias se encuentra en un punto intermedio entre la recta y la circunferencia, ya que en los extremos de la elipse se asemeja el comportamiento a una recta y no cuenta con giros tan prolongados como la circunferencia, en conclusión presenta un mejor desempeño en la regulación de la fuerza con un error aproximado de $\pm 0.8[N]$, esto lo podemos ver en comparación con la circunferencia comparando las gráficas 5.28 y 5.22. Las ganancias del control PI de fuerza y el tiempo de ejecución se observan en la Tabla 5.8.

Ganancia proporcional	$k_p = 1.1 \times 10^{-6}$
Ganancia integral	$k_i = 14 \times 10^{-4}$
Paso de integración	$h = 0.01[s]$
Tiempo de ejecución	$t = 75[s]$
Numero de vueltas	1

Tabla 5.8: Parámetros del controlador PI de fuerza para el seguimiento de elipse en posición.

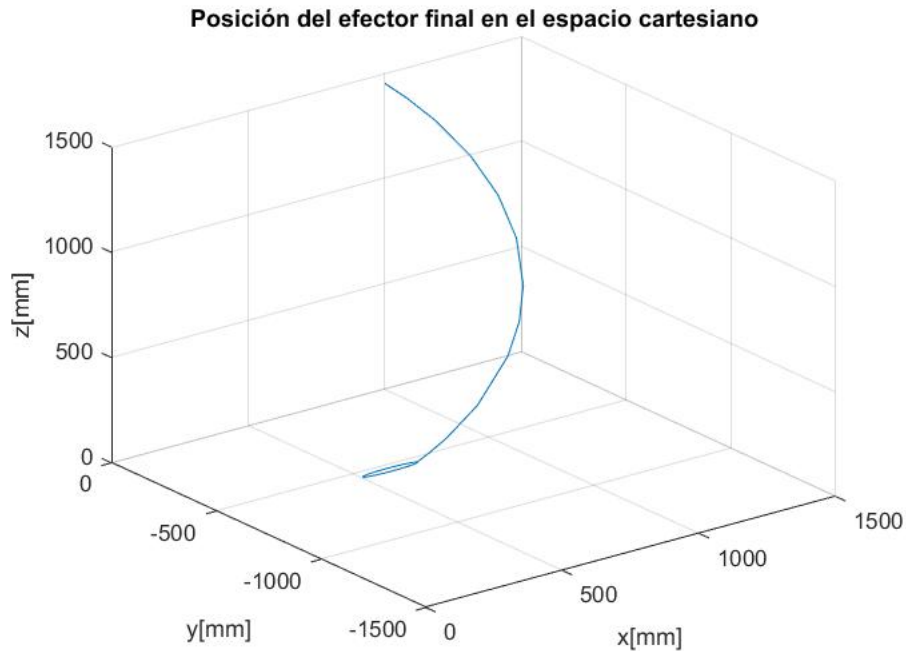


Figura 5.26: Posición completa del robot industrial en el espacio cartesiano para el seguimiento de una trayectoria elíptica en posición para el caso 2.

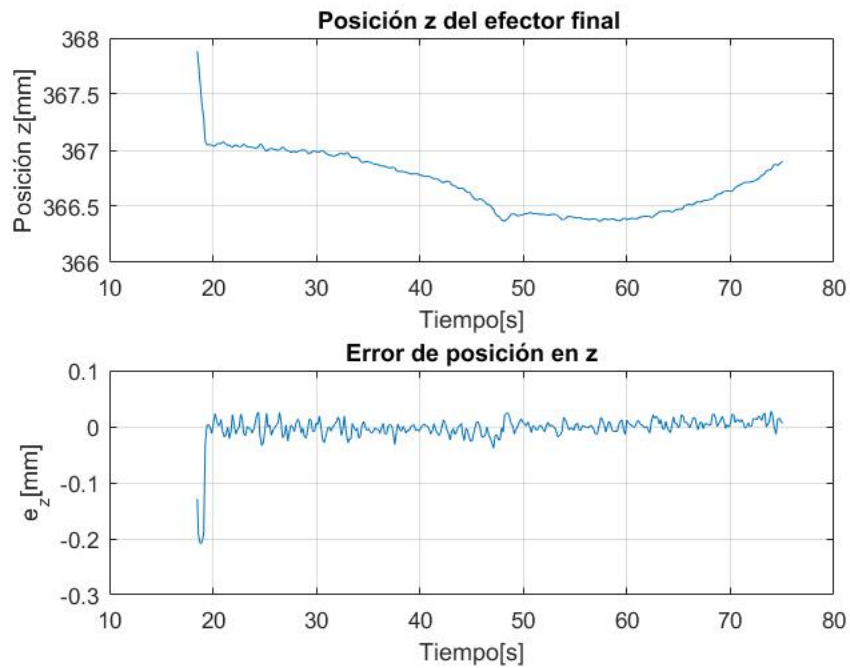


Figura 5.27: Graficas de posición y error de posición respecto al eje z siguiendo una trayectoria elíptica para el caso 2.

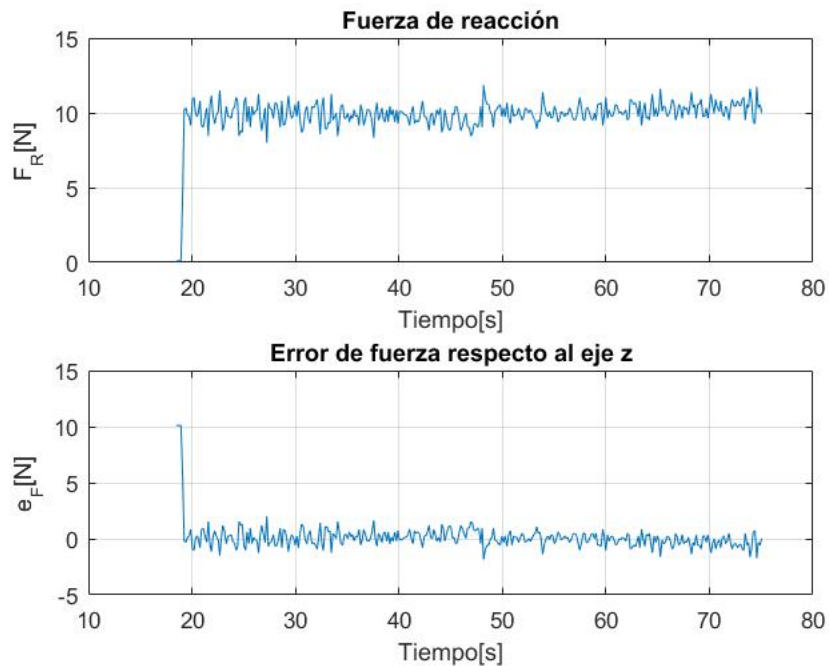


Figura 5.28: Graficas de fuerza y error de fuerza respecto al eje z siguiendo una trayectoria elíptica para el caso 2.

5.2. CASO 2: SEGUIMIENTO DE TRAYECTORIA DE POSICIÓN Y REGULACIÓN DE FUERZA

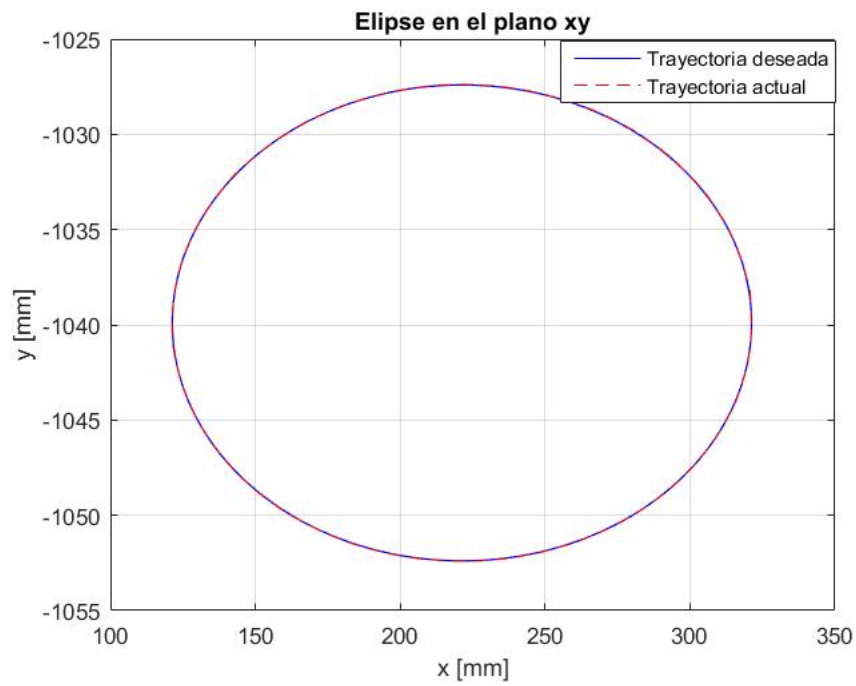


Figura 5.29: Seguimiento de una trayectoria elíptica en posición para el caso 2.

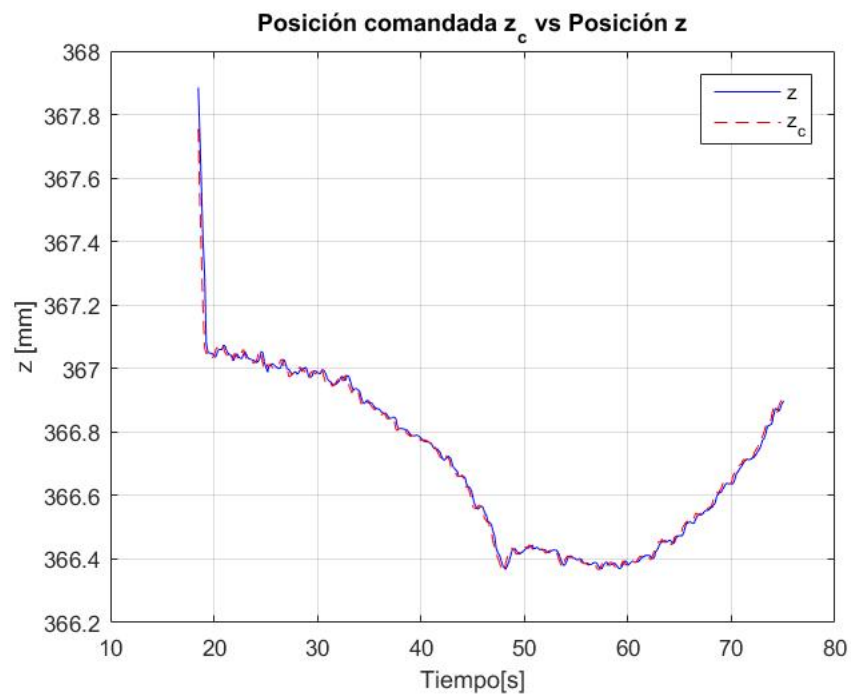


Figura 5.30: Comparación de la posición comandada con la posición actual para el caso 2, siguiendo una trayectoria elíptica en posición.

Como se aprecia en la Figura 5.29 el seguimiento de la elipse es mejor comparado con el de la circunferencia debido a la geometría a seguir y por las irregularidades de la superficie, donde en la circunferencia se presentan más por el área que abarca.

En la Figura 5.30 se observa un buen seguimiento de la posición comandada z_c como lo fue en los otros seguimientos de trayectoria, básicamente no se observa diferencia entre el desempeño del controlador en la aparte de posición siguiendo distintas geometrías regulares.

5.3. Caso 3: Regulación de posición y seguimiento de fuerza

Para este caso se resolvió el problema de seguimiento de trayectoria en el control de fuerza y el problema de regulación en el control de posición, las trayectorias a seguir fueron un coseno de fuerza y un diente de sierra de fuerza con referencia de $10N$ de fuerza. En este caso las posiciones deseadas fueron las mismas que en el caso 1 y las trayectorias de fuerza a seguir fueron las siguientes:

Diseño de trayectoria de coseno de fuerza

$$F_d = A * \cos(\theta) + b$$

En el coseno de fuerza se utilizó el parámetro $A = 2$ para modificar la amplitud del coseno y el parámetro $b = 10$ para realizar un corrimiento vertical.

Diseño de trayectoria de diente de sierra de fuerza

$$F_d = \begin{cases} 0.2t + 8 & 0 \leq t \leq 20 \\ 0.2t + 4 & 20 \leq t \leq 40 \\ 0.2t & 40 \leq t \leq 60 \\ 0.2t - 4 & 60 \leq t \leq 80 \end{cases}$$

Resultados siguiendo una trayectoria coseno en fuerza, regulando la posición del órgano terminal

En los siguientes experimentos se muestra el desempeño del controlador cuando se realiza el seguimiento de una trayectoria de fuerza, como se observa en la Figura 5.33 la trayectoria seguida es un coseno con referencia de $10[N]$ y una amplitud de $2[N]$, esta trayectoria es seguida por el controlador presentando el mismo ruido que los experimentos de los casos anteriores, con parámetros del control PI de fuerza mostrados en la Tabla 5.9.

Ganancia proporcional	$k_p = 5 \times 10^{-6}$
Ganancia integral	$k_i = 25 \times 10^{-4}$
Paso de integración	$h = 0.01[s]$
Tiempo de simulación	$t = 125[s]$
Fuerza de referencia	$10[N]$
Amplitud del coseno	$2[N]$

Tabla 5.9: Parámetros del controlador PI de fuerza para el seguimiento de un coseno de fuerza.

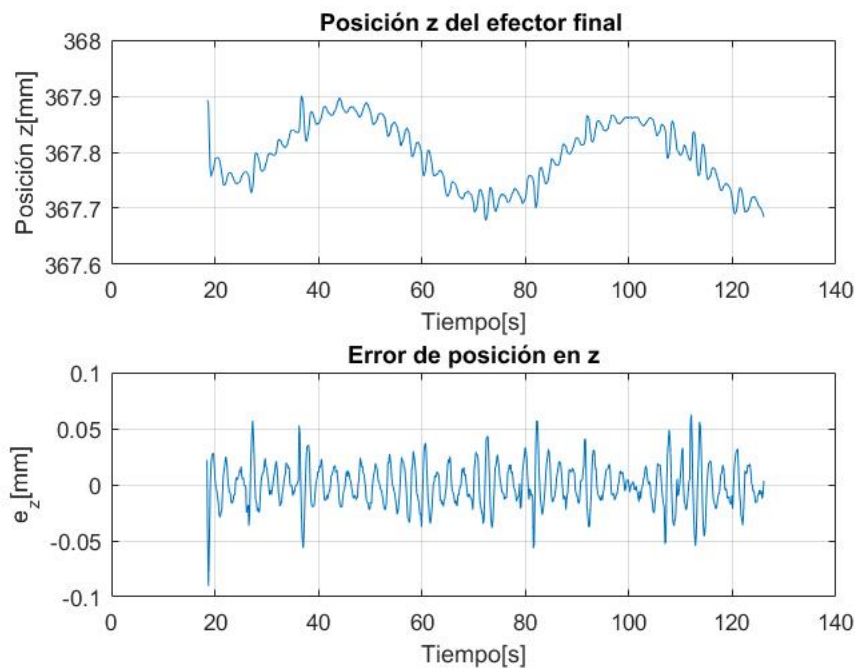


Figura 5.31: Graficas de posición y error de posición respecto al eje z siguiendo una trayectoria coseno en fuerza para el caso 3.

En la Figura 5.31 se observa un comportamiento cosenoidal en la posición z , que era de esperarse ya que la trayectoria a seguir en fuerza es una función coseno. En el error de posición e_z se aprecia un pequeño aumento a diferencia de los casos anteriores en un rango $\pm 0.05[mm]$ que es un valor muy pequeño de error.

El error de fuerza e_z (véase Figura 5.32) resultar menor que el error que se obtuvo siguiendo una trayectoria de posición como la circunferencia sin embargo se siguen presentando valores pico de fuerza.

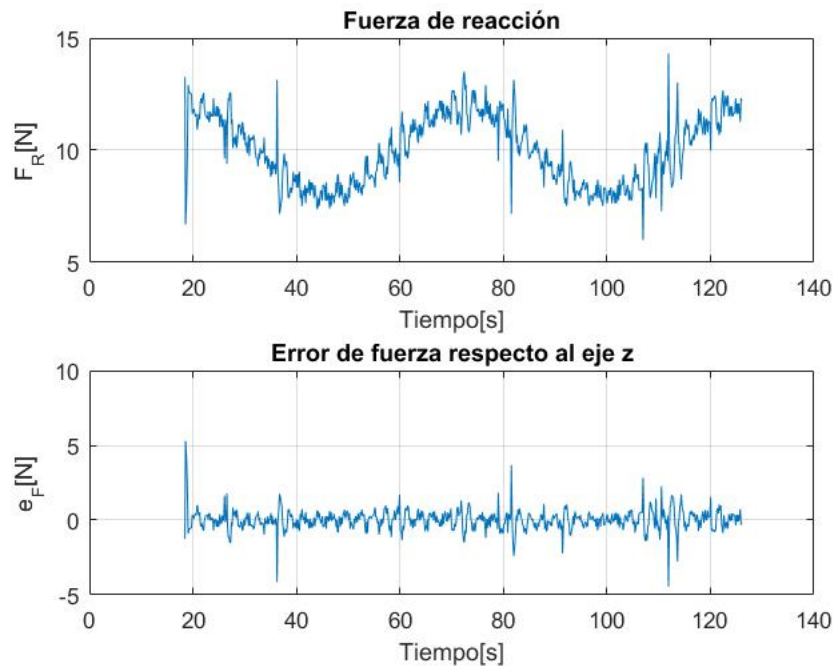


Figura 5.32: Graficas de fuerza y error de fuerza respecto al eje z siguiendo una trayectoria coseno en fuerza para el caso 3.

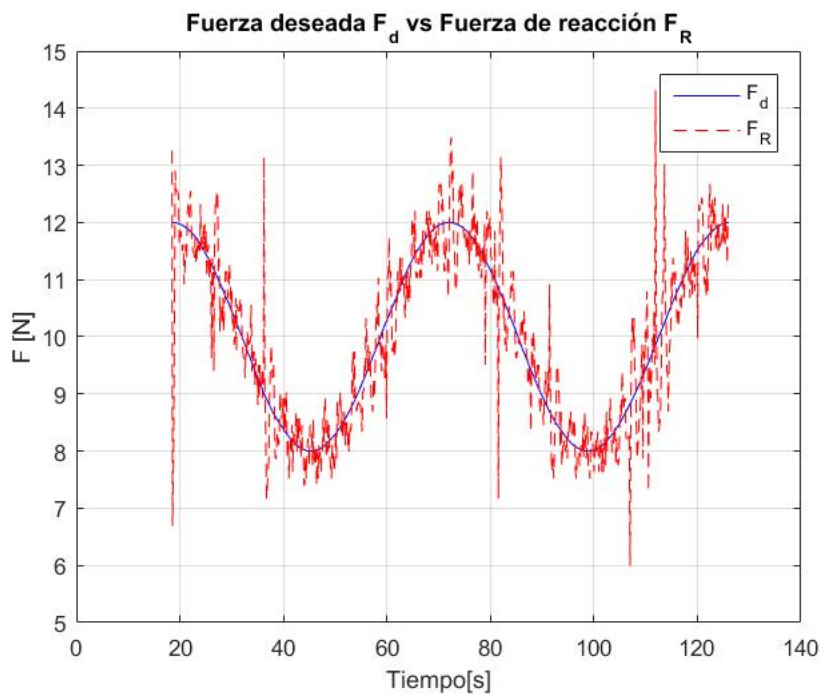


Figura 5.33: Seguimiento de una trayectoria coseno en fuerza para el caso 3.

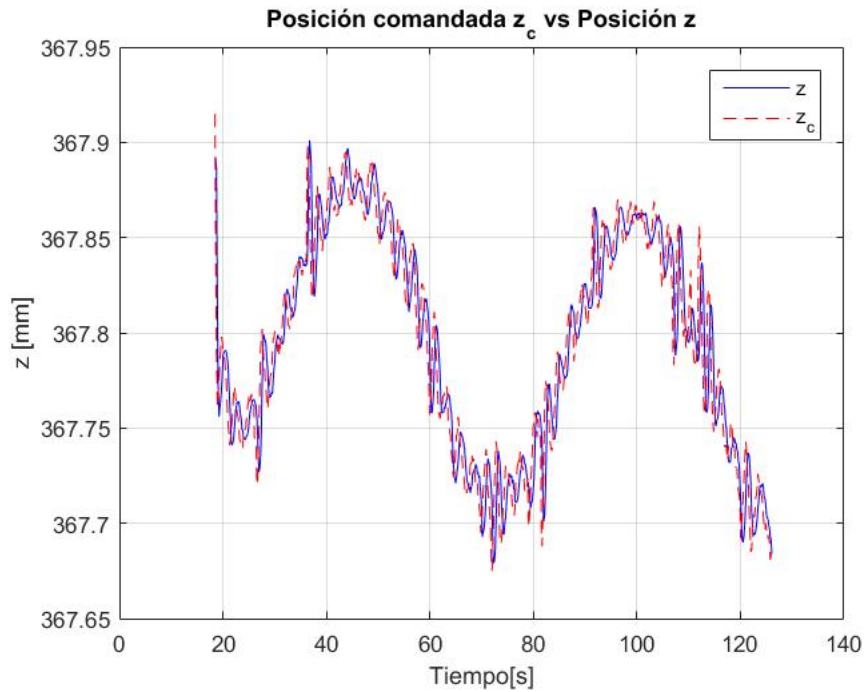


Figura 5.34: Comparación de la posición comandada con la posición actual para el caso 3 siguiendo una trayectoria coseno en fuerza.

Resultados siguiendo una trayectoria diente de sierra en fuerza, regulando la posición del órgano terminal

Para comparar la respuesta que se tiene en el seguimiento de trayectorias de fuerza, se presenta a continuación el seguimiento de un diente de sierra en fuerza, con una fuerza de referencia de $10 + / - 2N$ (véase Figura 5.37), con parámetros del control PI de fuerza mostrados en la Tabla 5.10.

Ganancia proporcional	$k_p = 5 \times 10^{-6}$
Ganancia integral	$k_i = 25 \times 10^{-4}$
Paso de integración	$h = 0.01[s]$
Tiempo de simulación	$t = 98[s]$
Fuerza de referencia	$10[N]$
Amplitud del coseno	$2[N]$

Tabla 5.10: Parámetros del controlador PI de fuerza para el seguimiento de un diente de sierra de fuerza.

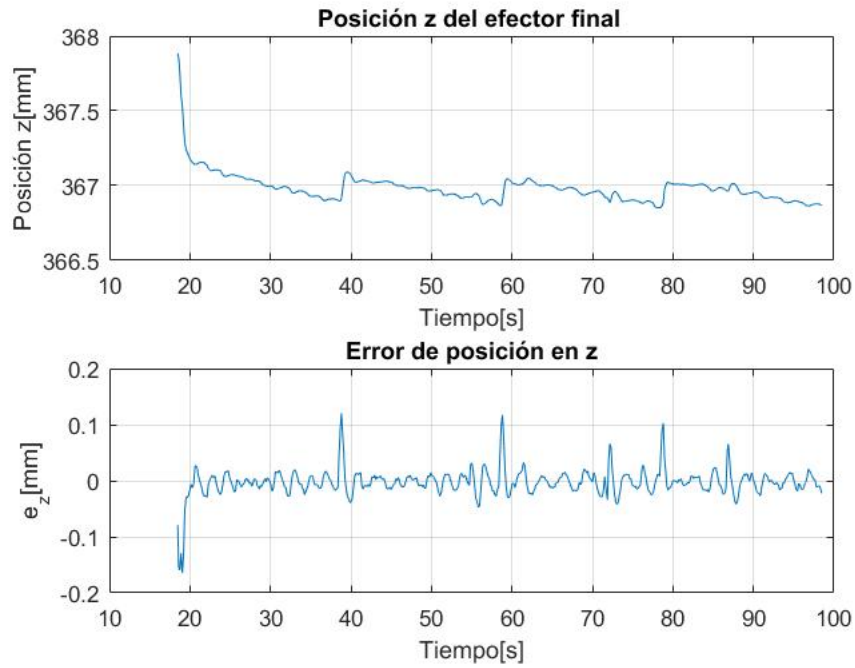


Figura 5.35: Graficas de posición y error de posición respecto al eje z siguiendo una trayectoria diente de sierra en fuerza para el caso 3.

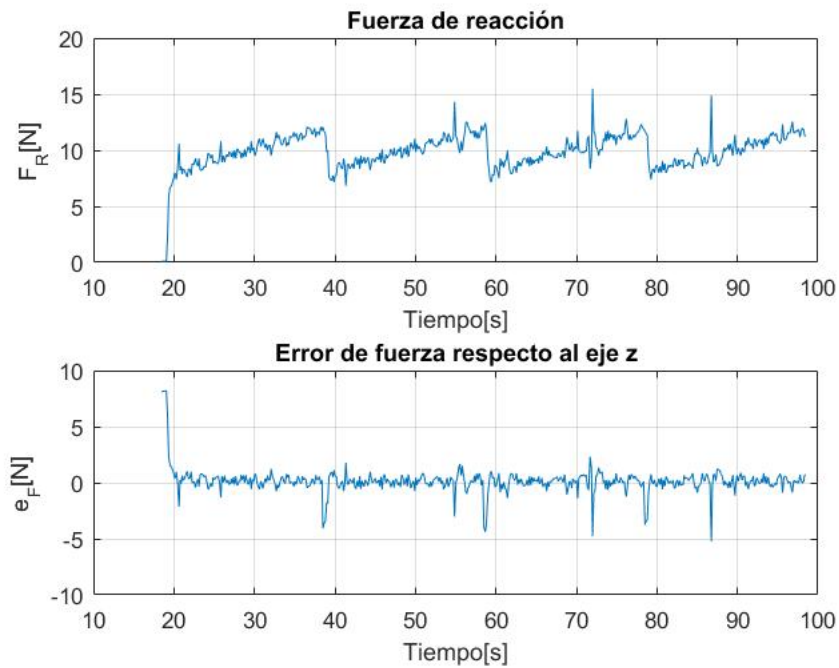


Figura 5.36: Graficas de fuerza y error de fuerza respecto al eje z siguiendo una trayectoria diente de sierra en fuerza para el caso 3.

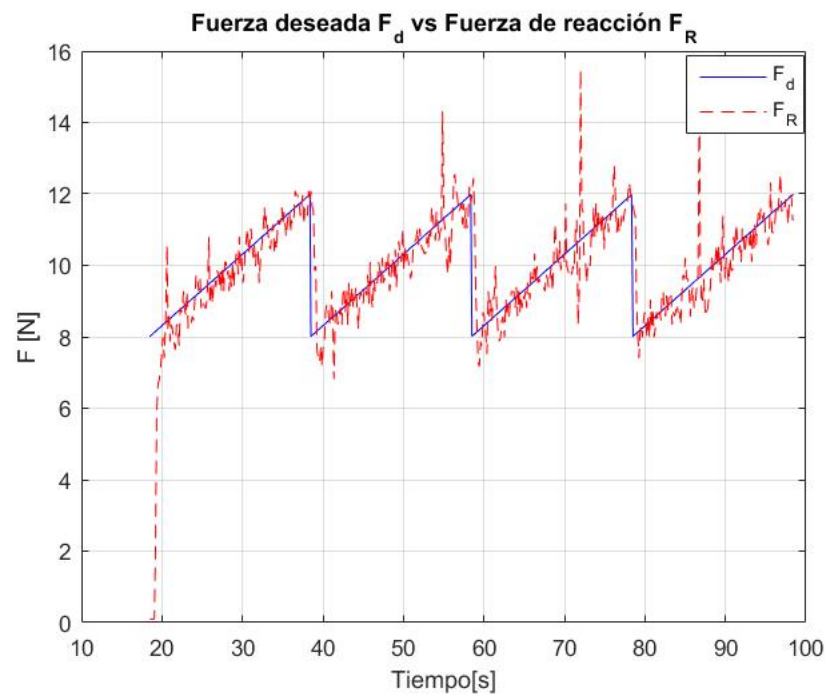


Figura 5.37: Seguimiento de una trayectoria diente de sierra en fuerza para el caso 3.

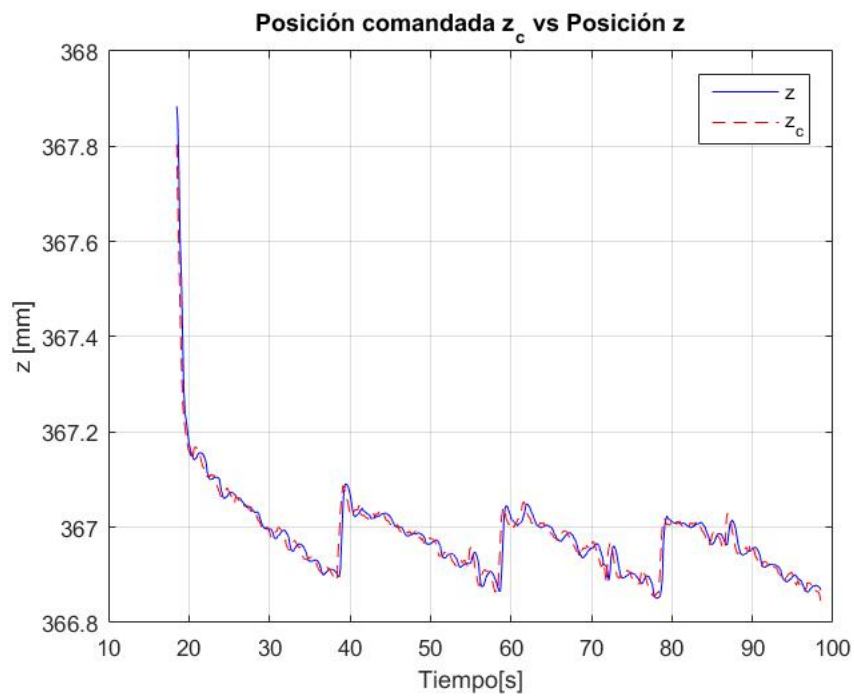


Figura 5.38: Comparación de la posición comandada con la posición actual para el caso 3 siguiendo una trayectoria diente de sierra en fuerza.

Como el diente de sierra presenta cambios en un ciclo de bajada vertical y subida oblicua, permite ver si ese cambio súbito lo puede seguir el robot, y como se observa en la Figura 5.37, el controlador permite seguir esta trayectoria durante este cambio, sin embargo le genera un pequeño desfase en la señal.

En la Figura 5.35 se puede observar el comportamiento de la posición z , la cual presenta algunos picos debido a que en esos instantes el seguimiento de fuerza presenta cambios súbitos en la trayectoria, al pasar de una subida con inclinación a una bajada vertical, que es la forma de la señal diente de sierra.

El último posible caso que podría estudiarse, es donde se tiene como objetivo de control el seguimiento de trayectoria, tanto en fuerza como en posición al mismo tiempo, ya que no se obtuvieron buenos resultados estos no fueron incluidos, debido a que el control de fuerza no cuenta con la parte derivativa que ayuda al seguimiento, y como se requiere también, que el control interno de posición siga una trayectoria, que viene de una posición comandada generada por el control de fuerza, el desempeño del control híbrido se ve disminuido.

5.4. Conclusiones

En este capítulo se presentaron resultados experimentales para 3 casos de estudio, en los cuales se analizó regulación y seguimiento tanto para posición como para fuerza. Se presentó la aplicación del esquema de control híbrido posición/fuerza en un robot industrial, para realizar tareas donde existe un contacto normal con una superficie, con un cierto valor de rigidez que depende de su material, este valor de rigidez fue compensado con una buena sintonización de las ganancias del control PI de fuerza.

Por otro lado los errores en el caso de regulación para fuerza y posición que se obtuvieron fueron buenos resultados, el control híbrido se mostró menos efectivo en el caso de seguimiento de una trayectoria de fuerza, ya que esto afectó generar una buena señal de referencia para el control de posición interno del robot, además en general todo el esquema de control se vio afectado por el periodo de muestro, ya que a su vez, es afectado por el tiempo de procesamiento que permite el controlador del robot industrial.

Capítulo 6

Conclusiones y trabajo futuro

En este trabajo se propone un esquema de control híbrido que consiste en conmutar y retroalimentar entre dos controles, un control PI de fuerza el cual se programó y un control PID interno de posición articular suministrado por el fabricante y que está disponible para realizar movimiento en el espacio cartesiano. Se mostró que es posible implementar una técnica de control híbrido para realizar tareas donde se controle la fuerza, sin necesidad de utilizar el equipamiento de control de fuerza que venden de forma externa los fabricantes de robots industriales. Aquí se implementó específicamente un control híbrido posición/fuerza en un robot industrial IRB 2400 de la marca ABB. Es importante destacar que el rendimiento del control se ve afectado por el tiempo de muestreo de la señal de fuerza debido al tiempo en que tarda en procesar el control interno PID del robot, sin embargo para tareas donde se busca regular solamente la fuerza, este control presenta un desempeño muy aceptable en estado estacionario, considerando que la señal de fuerza contiene ruido y que no fue implementado algún tipo de filtro para mejorar la señal.

Se mostró que el esquema de control híbrido posición/fuerza implementado permite control de fuerza en una tarea general de contacto y desplazamiento, ya que esto fue corroborado al obtener buenos resultados en los casos de estudio donde se realizó el seguimiento de posición y seguimiento de fuerza, se obtuvieron mejores resultados en posición que en fuerza en cuanto a su error y al tiempo de convergencia, debido a las elevadas prestaciones del control interno de posición del robot industrial. Esta tarea en general mostró la aplicación del esquema de control híbrido y su desempeño para ser extendida a tareas de aplicación como el maquinado, devastado y pulido de una pieza de metal. Los resultados experimentales obtenidos ponen en evidencia las cuestiones que más afectan el rendimiento del esquema de control híbrido, las cuales son que la superficie de contacto resulta no tener una rigidez uniforme en toda su superficie, además de presentar irregularidades por el montaje del acrílico y las condiciones de ambiente en que se encuentre, como la temperatura.

Las irregularidades que presentan las superficies de contacto, así como la variación de su rigidez dificultan la regulación correcta del valor de fuerza en todo tiempo en que se sigue una trayectoria de posición manteniendo una fuerza deseada, sin embargo se

mostró que una selección correcta de ganancias del control PI de fuerza permite mantener una regulación correcta de la fuerza, incluso cuando el efector final pasa por pequeñas irregularidades o desniveles como fue el caso del seguimiento de una línea recta, mientras el seguimiento de posición no se ve afectado en gran medida, ya que los resultados obtenidos fueron errores de posición muy pequeños.

El control híbrido posición/fuerza implementado, presenta ser reactivo ya que tiene una gran sensibilidad al cambio de fuerza de reacción durante la interacción del efector final con la superficie de contacto, por ejemplo al tener grandes variaciones de fuerza provocara oscilaciones en la posición comandada a seguir por el robot industrial.

Trabajo futuro

A continuación se mencionan las siguientes propuestas como trabajo futuro, dando seguimiento a la línea de investigación planteada en esta tesis:

- Realizar prueba de estabilidad del control híbrido posición/fuerza en robots industriales para estudiar más a fondo su desempeño.
- Probar el esquema de control híbrido posición/fuerza en tareas de aplicación como el maquinado de una pieza o tareas de ensamble donde se requiere control de fuerza.
- Se sugiere adicionar al control híbrido posición/fuerza una planeación de agarre con algún diseño óptimo del efector final para realizar tareas de multiagarre con robots industriales.
- Se propone implementar una comunicación basada en redes para controlar y monitorear un conjunto de robots industriales vía remota, con la aplicación de protocolos de comunicación TCP/IP basados en modelos OSI y TCP.
- Realizar tareas colaborativas con robots industriales extendiendo la comunicación a un conjunto de robots.
- Extender el control híbrido posición/fuerza a un control óptimo para mejorar el desempeño en el control de fuerza ya que existen tareas donde se necesita mayor precisión en el control de fuerza.

Apéndice A

Especificaciones del robot industrial IRB 2400 ABB

En este apéndice se presentan las especificaciones, medidas y datos del Robot IRB 2400 de la marca ABB, que fue con el cual se obtuvieron los resultados experimentales.

IRB 2400 Industrial Robot

Main Applications

Arc welding
Cutting/Deburring
Glueing/Sealing
Grinding/Polishing
Machine tending
Material handling



Most popular industrial robot

IRB 2400 is the world's most popular industrial robot in its class. It comprises a complete family of application optimized robots that maximize the efficiency of your arc welding, process and tending applications.

The IRB 2400 is a real hard worker. It can take additionally 35 kg load on axis 1 and up to 12 kg additional load on the upper arm - still keeping 100 % duty cycle.

The IRB 2400L model has 1.8 meters reach, 7 kg load capacity, large working range and slim arm and wrist. Other models offer handling capacity of up to 20 kg, excellent motion control, large load offset and unlimited motion in axis 6.

This means there's an IRB 2400 robot to give you excellent performance in your material handling, tending and process applications. All models offer you inverted mounting capability. The compact design of the IRB 2400 ensures ease of installation. The robust construction and use of minimum parts contribute to high reliability and long intervals between maintenance.

The Foundry Plus version is washable with high pressure steam and it's supplied with increased environment protection meeting IP 67 standard.

IRB 2400

Main applications

Arc welding, cutting/deburring, glueing/sealing, grinding/polishing, machine tending, material handling

Specification

Variants	Reach	Payload	Protection
IRB 2400L	1.8 m	7 kg	Foundry
IRB 2400-10	1.55 m	12 kg	FoundryPlus
IRB 2400-16	1.55 m	20 kg	FoundryPlus

Number of axes	6
Mounting	Floor and inverted all versions. Wall IRB 2400-10

Supplementary load	IRB 2400L	IRB 2400-10	IRB 2400-16
Upper arm,			
-wrist end	1 kg	2 kg	2 kg
-rear end	10 kg	10 kg	10 kg
Base unit	35 kg	35 kg	35 kg

Physical

Dimensions	IRB 2400L	IRB 2400-10	IRB 2400-16
Total height	1731 mm	1564 mm	1564 mm
Robot base	723x600 mm	723x600 mm	723x600 mm
Robot weight	380 kg	380 kg	380 kg

Performance (according to ISO 9283)

Position repeatability (RP)	0.03 - 0.07 mm depending on variant
Path repeatability	0.11-0.15 mm depending on variant

Movements

Axis	IRB 2400L	IRB 2400-10	IRB 2400-16
Working range			
Axis 1	360°	360°*	360°
Axis 2	210°	210°	210°
Axis 3	125°	125°	125°
Axis 4	370°	400°	400°
Axis 4, Option	-	Unlimited	Unlimited
Axis 5	230°	240°	240°
Axis 6	800°	800°	800°
Axis 6, Option	Unlimited	Unlimited	Unlimited

Max. speed

Positioning	IRB 2400L	IRB 2400-10	IRB 2400-16
Axis 1	150°/s	150°/s 90° **	150°/s
Axis 2	150°/s	150°/s 90° **	150°/s
Axis 3	150°/s	150°/s 90° **	150°/s
Axis 4	360°/s	360°/s	360°/s
Axis 5	360°/s	360°/s	360°/s
Axis 6	450°/s	450°/s	450°/s

**) For wall mounted version

*) ± 30° for wall mounted robot

Electrical Connections

Supply voltage	200–600 V, 50/60 Hz
Power consumption	ISO-Cube at max speed 0.67 kW

Environment

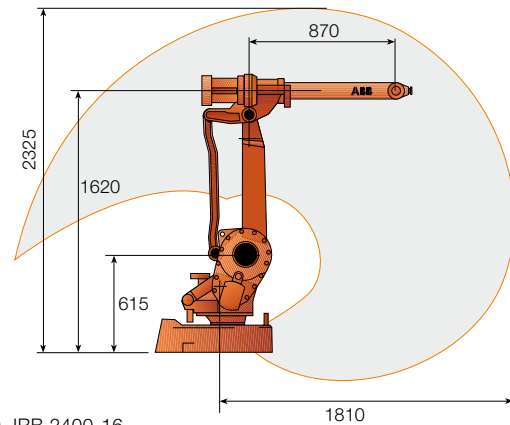
Ambient temperature for mechanical unit:	
During operation	+5°C (41°F) to + 45°C (113°F)
Relative humidity	Max. 95%

Degree of protection	IRB 2400L	IRB 2400-10	IRB 2400-16
Standard and Clean Room versions	IP54		IP54
Foundry or Foundry Plus versions		IP55/67	IP67IP67
Noise level	Max. 70 dB (A)		
Emission	EMC/EMI-shielded		

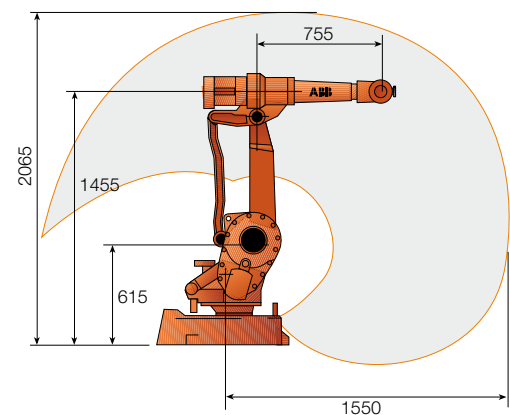
Data and dimensions may be changed without notice

Working range

IRB 2400L



IRB 2400-10, IRB 2400-16



2 Description

2.1 Structure

The robot is made up of two main parts: a manipulator and a controller.

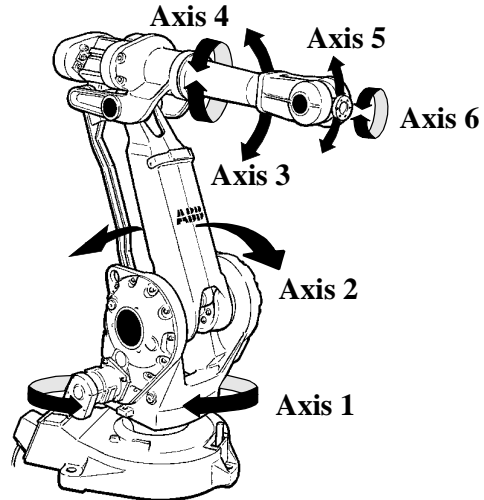


Figure 1 The IRB 2400 manipulator has 6 axes.

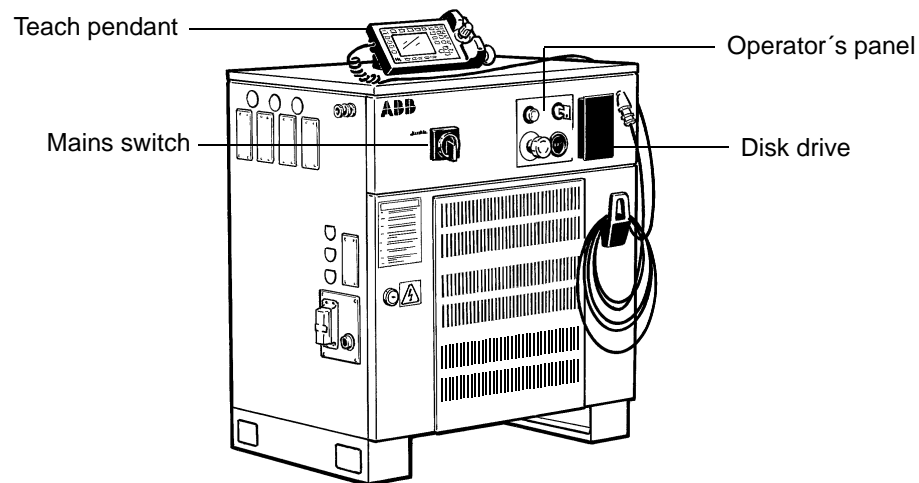
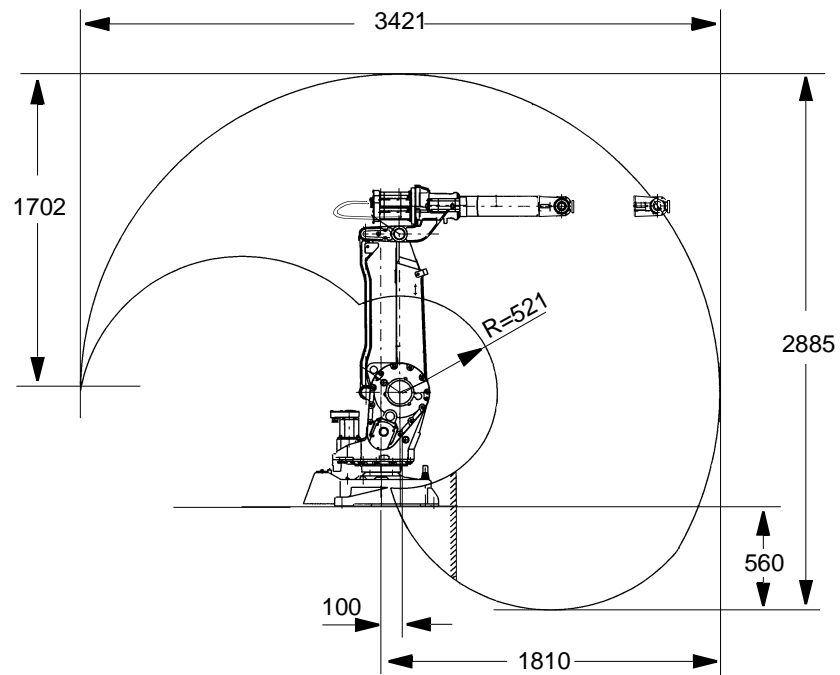


Figure 2 The controller is specifically designed to control robots, which means that optimal performance and functionality is achieved.

The controller contains the electronics required to control the manipulator, external axes and peripheral equipment.

2.8 Robot Motion

IRB 2400L



IRB 2400/10
IRB 2400/16

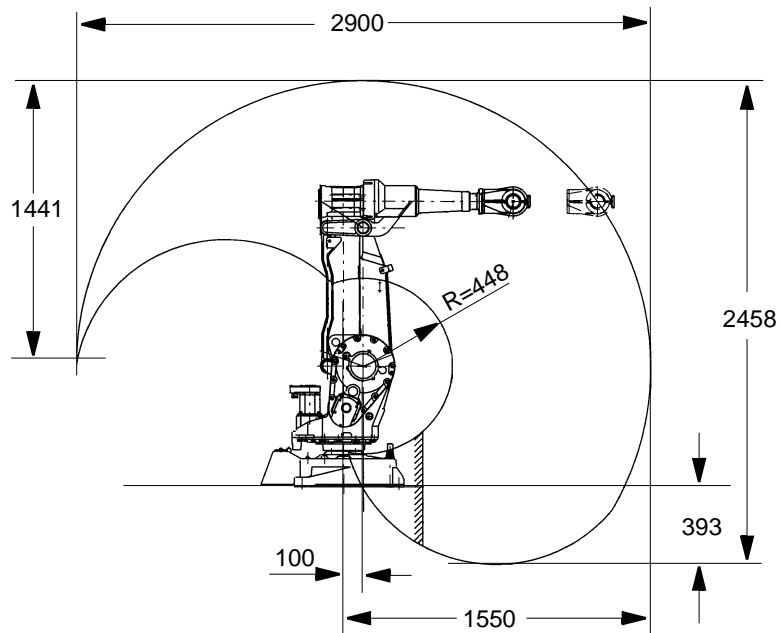


Figure 8 Working space of IRB 2400 (dimensions in mm).

Description

Motion performance

The QuickMove™ concept means that a self-optimizing motion control is used. The robot automatically optimizes the servo parameters to achieve the best possible performance throughout the cycle – based on load properties, location in working area, velocity and direction of movement.

- No parameters have to be adjusted to achieve correct path, orientation and velocity.
- Maximum acceleration is always obtained (acceleration can be reduced, e.g. when handling fragile parts).
- The number of adjustments that have to be made to achieve the shortest possible cycle time is minimized.

The TrueMove™ concept means that the programmed path is followed – regardless of the speed or operating mode – even after an emergency stop, a safeguarded stop, a process stop, a program stop or a power failure.

The robot can, in a controlled way, pass through singular points, i.e. points where two axes coincide.

Coordinate systems

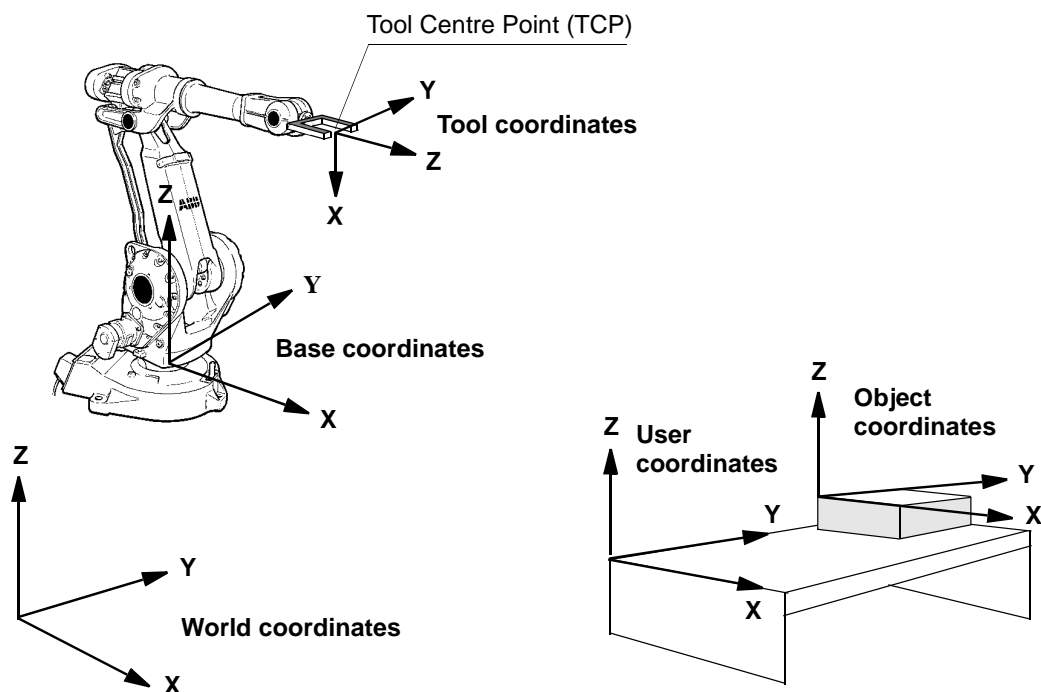


Figure 9 The coordinate systems, used to make jogging and off-line programming easier.

The world coordinate system defines a reference to the floor, which is the starting point for the other coordinate systems. Using this coordinate system, it is possible to relate the robot position to a fixed point in the workshop. The world coordinate system is also very useful when two robots work together or when using a robot carrier.

IRB 2400/10
IRB 2400/16

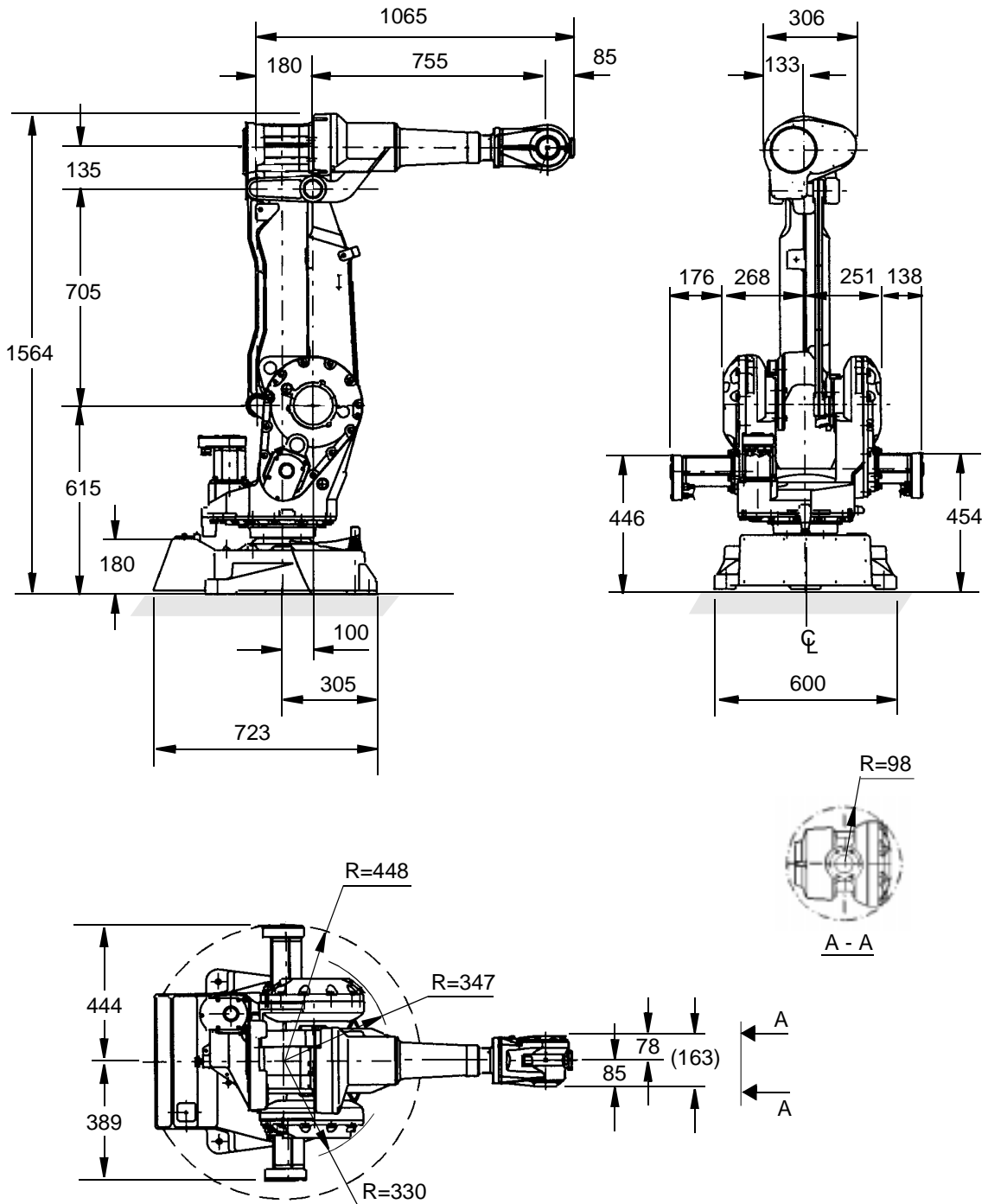


Figure 12 View of the manipulator from the side, rear and above (dimensions in mm).

IRB 2400/10
IRB 2400/16

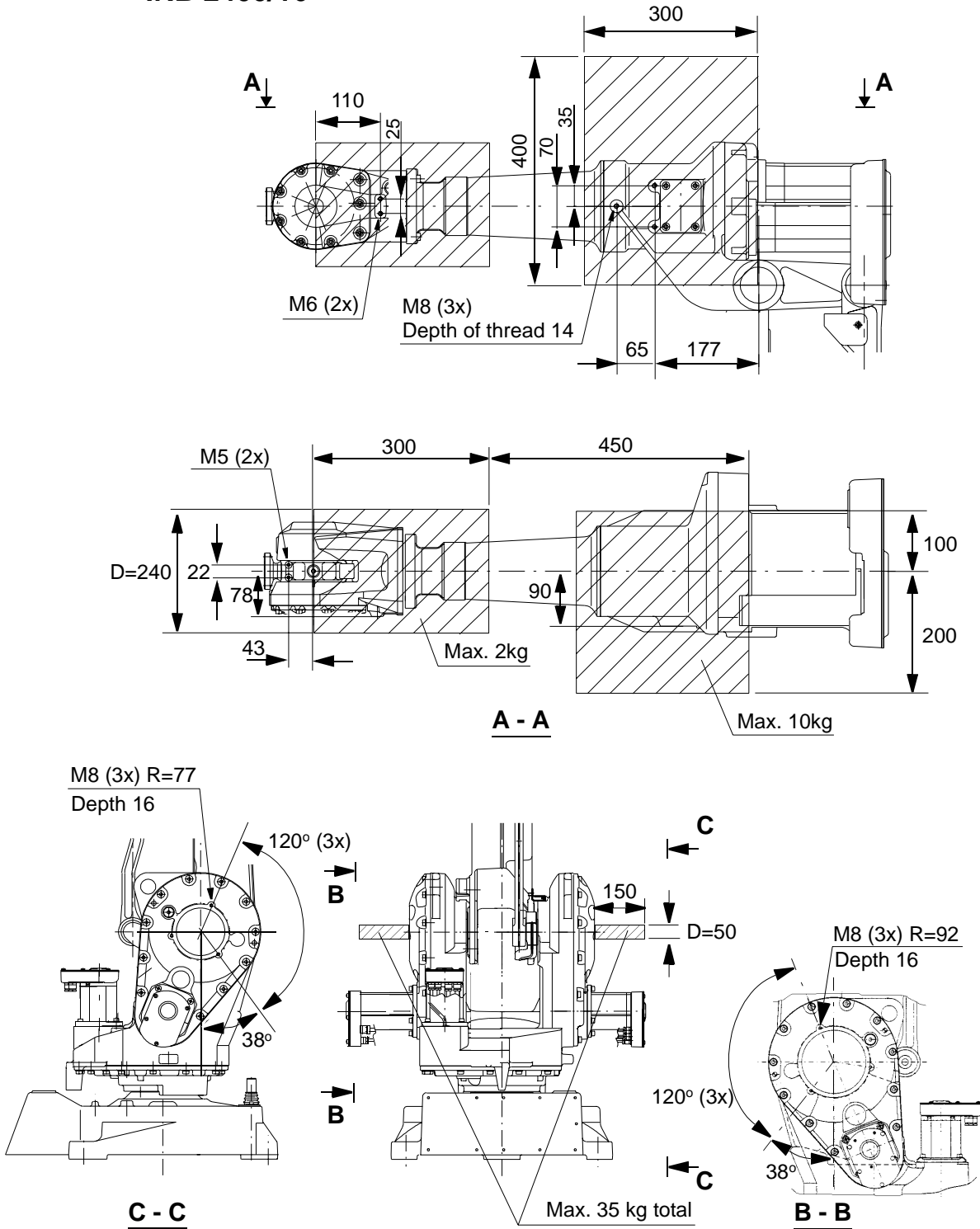
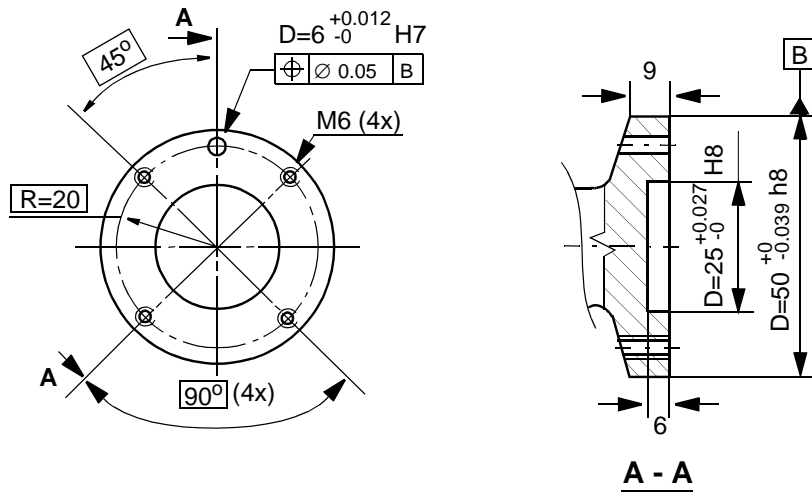


Figure 21 The shaded area indicates the permitted positions (centre of gravity) for any extra equipment mounted in the holes (dimensions in mm).

IRB 2400L



IRB 2400/10
IRB 2400/16

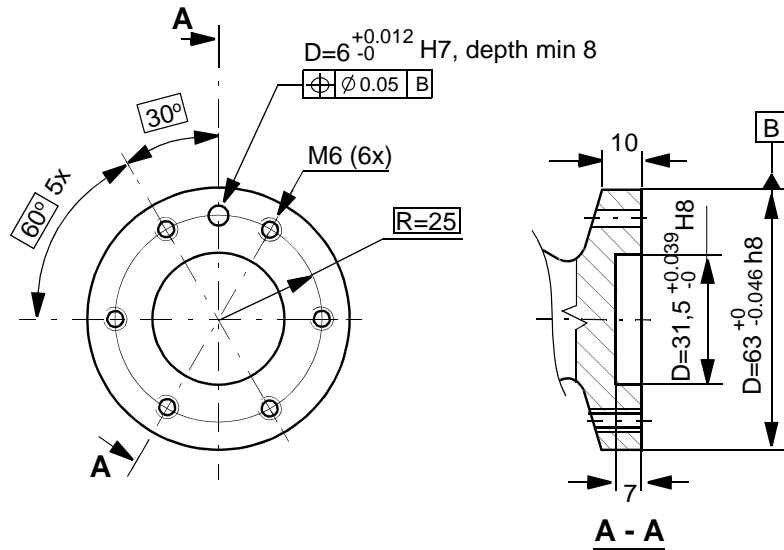


Figure 22 The mechanical interface, mounting flange (dimensions in mm).

IRB 2400/10, IRB 2400/16

The working area is the same for both floor and inverted mounting

Type of motion	Range of movement
Axis 1 Rotation motion	+180° to -180°
Axis 2 Arm motion	+110° to -100°
Axis 3 Arm motion	+65° to -60°
Axis 4 Wrist motion	+200° to -200° (Unlimited as optional)
Axis 5 Bend motion	+120° to -120°
Axis 6 Turn motion	+400° to -400° (Unlimited as optional)

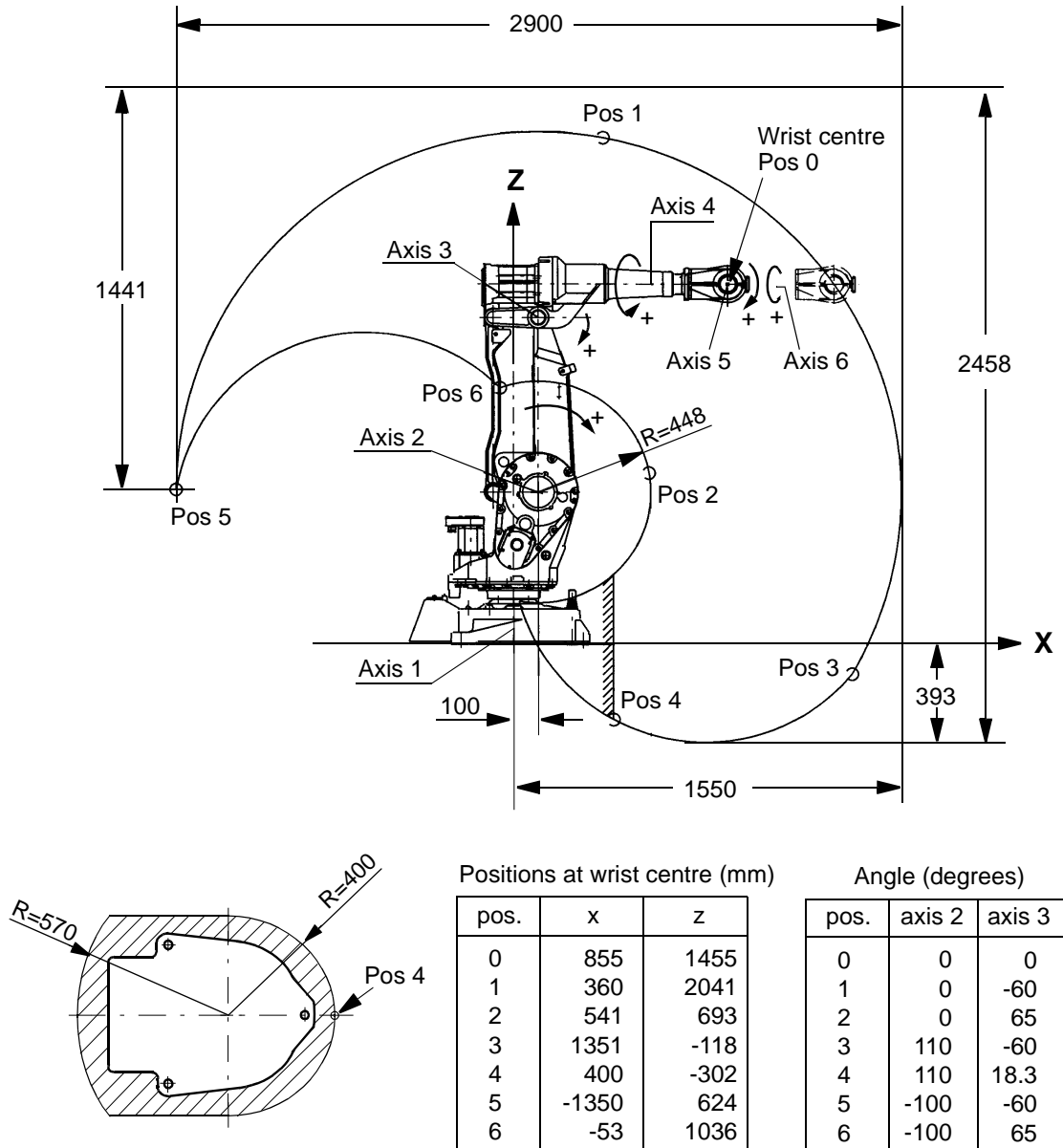


Figure 24 The extreme positions of the robot arm (dimensions in mm).

Technical specification

Performance according to ISO 9283

At rated load and 1 m/s velocity on the inclined ISO test plane with all six robot axes in motion.

Unidirectional pose repeatability:

RP = 0.06 mm

Linear path accuracy:

AT = 0.45 - 1.0 mm

Linear path repeatability:

RT = 0.14 - 0.25 mm

Minimum positioning time, to within 0.2 mm of the position:

0.2 - 0.35 sec. (on 35 mm linear path)

0.4 - 0.6 sec. (on 350 mm linear path)

The above values are the range of average test-results from a number of robots. If guaranteed values are required, please contact your nearest ABB Flexible Automation Centre.

Velocity

Versions:	IRB 2400L	IRB 2400/10	IRB 2400/16
Axis no. 1	150°/s	150°/s	150°/s
2	150°/s	150°/s	150°/s
3	150°/s	150°/s	150°/s
4	360°/s	360°/s	360°/s
5	360°/s	360°/s	360°/s
6	450°/s	450°/s	450°/s

There is a supervision to prevent overheating in applications with intensive and frequent movements.

Resolution

Approx. 0.01° on each axis.

3.10 Inputs and Outputs

Types of connection

The following types of connection are available:

- “Screw terminals” on the I/O units
- Serial interface for distributed I/O units
- Air and signal connections to upper arm

For more detailed information, see Chapter 4: *Specification of Variants and Options*.

I/O units

Several I/O units can be used. The following table shows the maximum number of physical signals that can be used on each unit.

Type of unit	Option no.	Digital		Analog			Power supply
		In	Out	Voltage inputs	Voltage output	Current output	
Digital I/O 24 VDC	20x	16	16				Internal/External ¹
Digital I/O 120 VAC	25x	16	16				Internal/External
Analog I/O	22x			4	3	1	Internal
AD Combi I/O	23x	16	16		2		Internal/External ¹
Relay I/O	26x	16	16				Internal/External ¹
Allen-Bradley Remote I/O Slave	281	128 ²	128				
Interbus-S Slave	284-285	64 ²	64				
Profibus DP Slave	286-287	128 ²	128				
Simulated I/O ³		100	100	30	30		
Encoder interface unit ⁴	288-289	1					

1. The digital signals are supplied in groups, each group having 8 inputs or outputs.
2. To calculate the number of logical signals, add 2 status signals for RIO unit and 1 for Interbus-S and Profibus DP.
3. A non physical I/O unit can be used to form cross connections and logical conditions without physical wiring. No. of signals are to be configured. Some ProcessWares include SIM unit.
4. Dedicated for conveyor tracking only.

Distributed I/O

The total number of logical signals is 512 (inputs or outputs, group I/O, analog and digital including field buses)

Max. total no of units*	20 (including SIM units)
Max. total cable length	100 m
Cable type (not included)	According to DeviceNet specification release 1.2
Data rate (fixed)	500 Kbit/s

* Max. four units can be mounted inside the cabinet.

Signal data

Permitted customer 24 V DC load		max. 6 A
Digital inputs 24 V DC	(options 201-208) Optically-isolated	
	Rated voltage:	24 V DC
	Logical voltage levels: "1"	15 to 35 V
	"0"	-35 to 5 V
	Input current at rated input voltage:	6 mA
	Potential difference:	max. 500 V
	Time delays: hardware	5–15 ms
	software	≤ 3 ms
	Time variations:	± 2 ms
Digital outputs 24 V DC	(options 201-208) Optically-isolated, short-circuit protected, supply polarity protection	
	Voltage supply	19 to 35 V
	Rated voltage	24 V DC
	Output current:	max. 0.5 A
	Potential difference:	max. 500 V
	Time delays: hardware	≤ 1 ms
	software	≤ 2 ms
	Time variations:	± 2 ms
Relay outputs	(options 261-268)	
	Single pole relays with one male contact (normally open)	
	Rated voltage:	24 V DC, 120 V AC
	Voltage range:	19 to 35 V DC 24 to 140 V AC
	Output current:	max. 2 A
	Potential difference:	max. 500V
	Time intervals: hardware (set signal)	typical 13 ms
	hardware (reset signal)	typical 8 ms
	software	≤ 4 ms
Digital inputs 120 V AC	(options 251-258) Optically isolated	
	Rated voltage	120 V AC
	Input voltage range: "1"	90 to 140 V AC
	Input voltage range: "0"	0 to 45 V AC
	Input current (typical):	7.5 mA
	Time intervals: hardware	≤ 20 ms
	software	≤ 4 ms

3 Technical specification

Applies to standard and Foundry versions unless otherwise stated.

3.1 Structure

Weight:	Manipulator	380 kg
	Controller	240 kg
Volume:	Controller	950 x 800 x 540 mm
Airborne noise level:	The sound pressure level outside the working space	< 70 dB (A) Leq (acc. to Machinery directive 89/392 EEC)

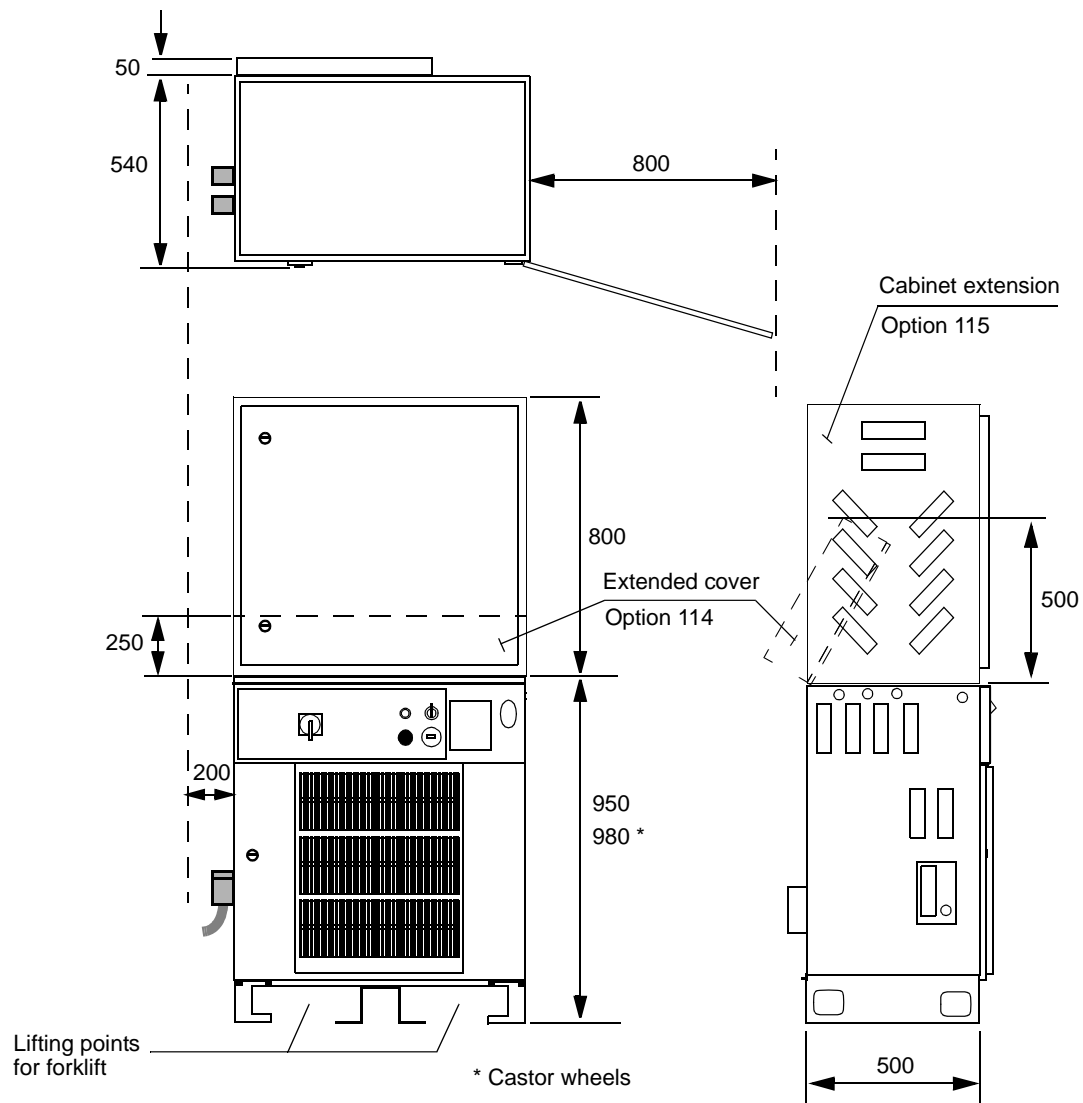


Figure 10 View of the controller from the front and from above (dimensions in mm).

Apéndice B

Programas de Arduino y RAPID

Programas desarrollados en el microcontrolador ATmega2560 por medio de una tarjeta embebida Arduino y programas desarrollados en el lenguaje RAPID compilados en RobotStudio.

Programa en Arduido del convertidor analógico-digital

```
//Autor: Azael
//Institución Cinvestav-Mecatronica
//Programa en Arduino para un convertidor analogico-digital de 10bits
#include <math.h>
//Declaración de variables-----
    int t=0;
    int z;
int AN0_pin=0;
int valor_sensor;
int d1;
int d2;
int d3;
const int Pin_B0=53;
const int Pin_B1=52;
//-----
//Configuración inicial del programa-----
void setup() {
DDRA = B11111111;
//DDRB = B11111111;
    pinMode(Pin_B0, OUTPUT);
    pinMode(Pin_B1, OUTPUT);
    //analogReadResolution(12);
    Serial.begin(9600);
}
//-----
```

```
//Programa principal-----
void loop() {
valor_sensor=analogRead(AN0_pin);
Serial.println(valor_sensor);
if (valor_sensor<=255)
{
PORTA=valor_sensor;
digitalWrite(Pin_B0,LOW);
digitalWrite(Pin_B1,LOW);
}
if (valor_sensor>255 && valor_sensor<=511)
{
d1=valor_sensor-255;
PORTA=d1;
digitalWrite(Pin_B0, HIGH);
}
if (valor_sensor>511 && valor_sensor<=767)
{
d2=valor_sensor-511;
PORTA=d2;
digitalWrite(Pin_B0,LOW);
digitalWrite(Pin_B1,HIGH);
}
if (valor_sensor>767)
{
d3=valor_sensor-767;
PORTA=d3;
digitalWrite(Pin_B0,HIGH);
digitalWrite(Pin_B1,HIGH);
}
delay(40);
}
```

Programa en RAPID para el caso de estudio 1

```
MODULE MainModule
!Configuración de efectores finales-----
PERS tooldata efector_final:=[TRUE,[[3.37831,0.159115,59.6913],[1,0,0,0]],
[0.09,[0,0,1],[1,0,0,0],0,0,0]];
PERS tooldata efector2:=[TRUE,[[0.62465,0.8516,59.9038],[1,0,0,0]],
[0.09,[0,0,1],[1,0,0,0],0,0,0]];
!-----
!Variables de adquisición de datos-----
VAR num D0;
```

```

VAR num D1;
VAR num D2;
VAR num D3;
VAR num D4;
VAR num D5;
VAR num D6;
VAR num D7;
VAR num D8;
VAR num D9;
VAR num k;
!-----
!-----
!Ctes del control PI-----
CONST num kp:=0.05*0.00125;
CONST num ki:=0.3*0.00125;
!CONST num kd:=0.00000001;
CONST num h:=0.01;
!-----
!Condiciones iniciales-----
    VAR num FR:=0;
    CONST num K0:=10200;
    VAR num PF:=0;
    CONST num Prefz:=368.4;
    CONST num Prefx:=468.14;
!-----
!Variables y constantes auxiliares-----
    CONST num Fd:=10;
    VAR num e:=0;
    VAR num ie:=0;
    VAR num de:=0;
    VAR num Pref;
    VAR num ia:=0;
    VAR num i:=1;
    !VAR num ea:=0;
!-----
!Vectores de datos-----
    PERS num e_vector{38000};
    PERS num Penetracion_vector{3800};
    PERS num FR_vector{3800};
    PERS num e_posicion{3800};
    PERS num t_vector{3800};
    PERS num t_vector2{3800};

```

```

PERS num x_vector{3800};
PERS num y_vector{3800};
PERS num z_vector{3800};
PERS num PF_vector{3800};
VAR num PO_vector{3800};
VAR num Datos_vector{3800};
!-----
        VAR num x;
        VAR num z;
        VAR num y;
        VAR pos pos1;
        VAR num POz;
        VAR dnum SF:=1;
        VAR num sensor_F;
!Variables de reloj-----
        VAR clock time;
        VAR clock time2;
        VAR num tt;
        VAR num tt2;
!-----
!Programa principal-----
PROC main()
        k:=k+1;
        ClkStart time;
        ClkStart time2;
        tt2:=ClkRead(time2);
        t_vector2{k}:=tt2;

        Datos1;
        sensor_F:=DnumToNum(SF);
                pos1:=CPos();
                        x:=pos1.x;
                        y:=pos1.y;
                        z:=pos1.z;
        x_vector{k}:=x;
        y_vector{k}:=y;
        z_vector{k}:=z;
        IF z<=368.5 THEN
                ControladorPID;
        ELSE
MoveJ [[468.14,-1205.94,368.4],[0.00171871,-0.779389,-0.626535,
-0.00193248],

```

```

[-1,0,0,0],[0.0402135,9E+09,9E+09,9E+09,9E+09,9E+09]], v300, fine,
  efector_final;
      ENDIF
      ENDPROC
!-----
!-----
!Rutina de posición Home
PROC azacalibracion()
MoveAbsJ JHome\NoEOffs, v300, z50, efector_final;
ENDPROC
!-----
!-----
!Rutina que implementa el controlador PI de Fuerza-----
      PROC ControladorPID()
        tt:=ClkRead(time);
        t_vector{i}:=tt;

        Datos1;
        sensor_F:=DnumToNum(SF);
        FR:=(sensor_F*5)/1024)*24-0.1;
        FR_vector{i}:=FR;
        e:=Fd-FR;
        e_vector{i}:=e;
!Integral del error
        ie:=ia+(h/2)*(e+ea);
!Derivada del error
        !de:=(e-ea)/h);
!Salida del controlador
        Pref:=kp*e+ki*ie;
        IF Pref*1000>=2.5 THEN
          Pref:=2/1000;
        ENDIF
        Penetracion_vector{i}:=Pref;
        PF_vector{i}:=PF;
!Actualizacion de variables
        i:=i+1;
        ia:=ie;
        !da:=de;
        ea:=e;
        PF:=Prefz-Pref*1000;
        PF_vector{i}:=PF;
        pos1:=CPos();

```

```

        x:=pos1.x;
        y:=pos1.y;
        z:=pos1.z;
        e_posicion{i}:=PF-z;
MoveJ [[Prefx,-1205.94,PF],[0.00171871,-0.779389,-0.626535,
-0.00193248],
[-1,0,0,0],[0.0402135,9E+09,9E+09,9E+09,9E+09,9E+09]], v100,
fine, efector_final;
        IF tt>=65 THEN
        Stop;
        ENDIF

```

```

ENDPROC

```

```

!-----
!--Datos 1-----

```

```

PROC Datos1()

```

```

!-----Entrada D0

```

```

IF DI10_9=1 THEN

```

```

    D0:=1;

```

```

    BitSet SF,1;

```

```

ELSE

```

```

    D0:=0;

```

```

    BitClear SF,1;

```

```

ENDIF

```

```

!-----

```

```

!-----Entrada D1

```

```

IF DI10_10=1 THEN

```

```

    D1:=1;

```

```

    BitSet SF,2;

```

```

ELSE

```

```

    D1:=0;

```

```

    BitClear SF,2;

```

```

ENDIF

```

```

!-----

```

```

!-----Entrada D2

```

```

IF DI10_11=1 THEN

```

```

    D2:=1;

```

```

    BitSet SF,3;

```

```

ELSE

```

```

    D2:=0;

```

```

    BitClear SF,3;

```

```

ENDIF
!-----
!-----Entrada D3
IF DI10_12=1 THEN
    D3:=1;
    BitSet SF,4;
ELSE
    D3:=0;
    BitClear SF,4;
ENDIF
!-----
!-----Entrada D4
IF DI10_13=1 THEN
    D4:=1;
    BitSet SF,5;
ELSE
    D4:=0;
    BitClear SF,5;
ENDIF
!-----
!-----Entrada D5
IF DI10_14=1 THEN
    D5:=1;
    BitSet SF,6;
ELSE
    D5:=0;
    BitClear SF,6;
ENDIF
!-----
!-----Entrada D6
IF DI10_15=1 THEN
    D6:=1;
    BitSet SF,7;
ELSE
    D6:=0;
    BitClear SF,7;
ENDIF
!-----
!-----Entrada D7
IF DI10_16=1 THEN
    D7:=1;
    BitSet SF,8;

```

```

ELSE
    D7:=0;
    BitClear SF,8;
ENDIF
!-----Entrada D8
IF DI10_1=1 THEN
    D8:=1;
    BitSet SF,9;
ELSE
    D8:=0;
    BitClear SF,9;
ENDIF
!-----Entrada D9
IF DI10_2=1 THEN
    D9:=1;
    BitSet SF,10;
ELSE
    D9:=0;
    BitClear SF,10;
ENDIF
ENDPROC
!-----
ENDMODULE

```

Programa en RAPID para el caso de estudio 2

```

MODULE MainModule
!Configuración de efectores finales-----
PERS tooldata efector_final:=[TRUE,[[3.37831,0.159115,59.6913],[1,0,0,0]],
[0.09,[0,0,1],[1,0,0,0],0,0,0]];
PERS tooldata efector2:=[TRUE,[[0.62465,0.8516,59.9038],[1,0,0,0]], [0.09,
[0,0,1],[1,0,0,0],0,0,0]];
!-----
!Variables de adquisición de datos-----
    VAR num D0;
    VAR num D1;
    VAR num D2;
    VAR num D3;
    VAR num D4;
    VAR num D5;
    VAR num D6;
    VAR num D7;
    VAR num D8;
    VAR num D9;

```



```

    VAR num k;
!-----
!Ctes del controlador PI de fuerza-----
    CONST num    kp:=0.004*0.00125;
    CONST num    ki:=1.1*0.00125;
!CONST num    kd:=0.00000001;
    CONST num h:=0.01;
!-----
!Condiciones iniciales-----
    VAR num FR:=0.0;
    CONST num K0:=10200;
    VAR num PF:=0;
    CONST num Prefz:=367.9;
!-----
!Variables auxiliares-----
    CONST num Fd:=10.0;
    VAR num e:=0;
    VAR num ie:=0;
    !VAR num de:=0;
    VAR num Pref;
    VAR num ia:=0;
    VAR num i:=1;
!-----
!Vectores de datos
    PERS num e_vector{11000};
    PERS num Penetracion_vector{11000};
    PERS num FR_vector{11000};
    PERS num e_posicion{11000};
    PERS num t_vector{11000};
    PERS num t_vector2{11000};
    PERS num x_vector{11000};
    PERS num y_vector{11000};
    PERS num z_vector{1100};
    PERS num PF_vector{1100};
!-----
    VAR num PO_vector{12000};
    VAR num Datos_vector{12000};
!-----
    VAR num x;
    VAR num y;
    VAR num z;
    VAR pos pos1;

```

```

        VAR dnum SF:=1;
        VAR num sensor_F;
!-----
!Variables para generar Trayectoria circunferencia-----
        VAR num yc:=-1039.9;
        VAR num xc:=271.3;
        CONST num r:=50;
        CONST num a:=221.3;
        CONST num b:=-1039.9;
        VAR num theta:=0;
!-----
        VAR clock time;
        VAR clock time2;
        VAR num tt;
        VAR num tt2;
!-----
!Programa principal
PROC main()
    k:=k+1;
        ClkStart time2;
        tt2:=ClkRead(time2);
        t_vector2{k}:=tt2;

        Datos1;
        sensor_F:=DnumToNum(SF);
            pos1:=CPos();
                x:=pos1.x;
                y:=pos1.y;
                z:=pos1.z;
            IF z<=368.2 THEN
                ControladorPID;
            ELSE
MoveJ [[xc,yc,367.9],[0.00171871,-0.779389,-0.626535,-0.00193248],
[-1,0,0,0],[0.0402135,9E+09,9E+09,9E+09,9E+09,9E+09]], v100, fine,
efector_final;
                ENDIF
        ENDPROC
!-----
!-----
!Rutina de posición Home
PROC azacalibracion()
MoveAbsJ JHome\NoEOffs, v300, z50, efector_final;

```

```

ENDPROC
!-----
!-----
!Rutina que implementa el controlador PI de Fuerza-----
  PROC ControladorPID()
    ClkStart time;
    tt:=ClkRead(time);
    t_vector{i}:=tt;
  Datos1;
  sensor_F:=DnumToNum(SF);
    x:=pos1.x;
    y:=pos1.y;
    z:=pos1.z;
    x_vector{i}:=x;
    y_vector{i}:=y;
    z_vector{i}:=z;
    FR:=((sensor_F*5)/1024)*24-0.1;
!-----

    IF FR>=50.0 THEN
  FR:=50.0;
    ENDIF
    FR_vector{i}:=FR;
    e:=Fd-FR;
    e_vector{i}:=e;
!Integral del error
    ie:=ia+(h/2)*(e+ea);
!Derivada del error
    !de:=(e-ea)/h;
!Salida del controlador
    Pref:=kp*e+ki*ie;
    IF Pref*1000>=2.5 THEN
    Pref:=2/1000;
    ENDIF
    Penetracion_vector{i}:=Pref;
!Actualizacion de variables
    i:=i+1;
    ia:=ie;
    ea:=e;
    PF:=Prefz-Pref*1000;
    PF_vector{i}:=PF;
    pos1:=CPos();
    x:=pos1.x;

```

```

        y:=pos1.y;
        z:=pos1.z;
        e_posicion{i}:=PF-z;
!Generación de trayectoria circular-----
        theta:=theta+1;
        xc:=a+r*cos(theta);
        yc:=b+r*sin(theta);
MoveJ [[xc,yc,PF],[0.00171871,-0.779389,-0.626535,-0.00193248],
[-1,0,0,0],[0.0402135,9E+09,9E+09,9E+09,9E+09,9E+09]], v100, fine,
efector_final;
!-----
        IF theta>=1080 THEN !Stop del programa
                Stop;
        ENDIF
    ENDPROC
!-----
!--Datos 1-----
PROC Datos1()

!-----Entrada D0
IF DI10_9=1 THEN
    D0:=1;
    BitSet SF,1;
ELSE
    D0:=0;
    BitClear SF,1;
ENDIF
!-----
!-----Entrada D1
IF DI10_10=1 THEN
    D1:=1;
    BitSet SF,2;
ELSE
    D1:=0;
    BitClear SF,2;
ENDIF
!-----
!-----Entrada D2
IF DI10_11=1 THEN
    D2:=1;
    BitSet SF,3;
ELSE

```

```

        D2:=0;
        BitClear SF,3;
ENDIF
!-----
!-----Entrada D3
IF DI10_12=1 THEN
    D3:=1;
    BitSet SF,4;
ELSE
    D3:=0;
    BitClear SF,4;
ENDIF
!-----
!-----Entrada D4
IF DI10_13=1 THEN
    D4:=1;
    BitSet SF,5;
ELSE
    D4:=0;
    BitClear SF,5;
ENDIF
!-----
!-----Entrada D5
IF DI10_14=1 THEN
    D5:=1;
    BitSet SF,6;
ELSE
    D5:=0;
    BitClear SF,6;
ENDIF
!-----
!-----Entrada D6
IF DI10_15=1 THEN
    D6:=1;
    BitSet SF,7;
ELSE
    D6:=0;
    BitClear SF,7;
ENDIF
!-----
!-----Entrada D7
IF DI10_16=1 THEN

```

```

        D7:=1;
        BitSet SF,8;
ELSE
        D7:=0;
        BitClear SF,8;
ENDIF
!-----Entrada D8
IF DI10_1=1 THEN
        D8:=1;
        BitSet SF,9;
ELSE
        D8:=0;
        BitClear SF,9;
ENDIF
!-----Entrada D9
IF DI10_2=1 THEN
        D9:=1;
        BitSet SF,10;
ELSE
        D9:=0;
        BitClear SF,10;
ENDIF
ENDPROC
!-----
ENDMODULE

```

Programa en RAPID para el caso de estudio 3

```

MODULE MainModule
!Configuración de efectores finales-----
PERS tooldata efector_final:=[TRUE, [[3.37831,0.159115,59.6913],[1,0,0,0]],
[0.09,[0,0,1],[1,0,0,0],0,0,0]];
PERS tooldata efector2:=[TRUE, [[0.62465,0.8516,59.9038],[1,0,0,0]],
[0.09,[0,0,1],[1,0,0,0],0,0,0]];
!-----
!Variables de adquisición de datos-----
VAR num D0;
VAR num D1;
VAR num D2;
VAR num D3;
VAR num D4;
VAR num D5;
VAR num D6;
VAR num D7;

```

```

VAR num D8;
VAR num D9;
VAR num k;
!-----
!Ctes del Controlador PI-----
  CONST num    kp:=0.004*0.00125;
  CONST num    ki:=1.1*0.00125;
  !CONST num    kd:=0.00000001;
  CONST num    h:=0.01;
!-----
!Condiciones iniciales-----
  VAR num FR:=0.0;
  CONST num K0:=10200;
  VAR num PF:=0;
  CONST num Prefz:=367.9;
!-----
!Variables auxiliares-----
  VAR num Fd:=10.0;
  VAR num e:=0;
  VAR num ie:=0;
  VAR num de:=0;
  VAR num Pref;
  VAR num ia:=0;
  VAR num i:=1;
  VAR num ea:=0;
  !VAR num da:=0;
!-----
!Vectores de datos-----
  PERS num e_vector{6000};
  PERS num Penetracion_vector{6000};
  PERS num FR_vector{6000};
  PERS num Fd_vector{6000};
  PERS num e_posicion{6000};
  PERS num t_vector{6000};
  PERS num t_vector2{6000};
  PERS num x_vector{6000};
  PERS num y_vector{6000};
  PERS num z_vector{6000};
  PERS num PF_vector{6000};
!-----
  VAR num x;
  VAR num y;

```

```

        VAR num z;
        VAR pos pos1;
        VAR num POz;
        VAR num t_muestreo;
        VAR motsetdata muestreo;
        VAR dnum SF:=1;
        VAR num sensor_F;
!-----
!Variables para generar Trayectoria -----
        CONST num yc:=-1039.9;
        CONST num xc:=321.3;
        CONST num A:=10;
        CONST num w:=2;
!-----
        VAR clock time;
        VAR clock time2;
        VAR num tt;
        VAR num tt2;
!-----
!Programa principal-----
PROC main()
k:=k+1;
        ClkStart time2;
        tt2:=ClkRead(time2);
        t_vector2{k}:=tt2;
        Datos1;
        sensor_F:=DnumToNum(SF);
                pos1:=CPos();
                x:=pos1.x;
                y:=pos1.y;
                z:=pos1.z;
        IF z<=368.2 THEN
        ControladorPID;
        ELSE
MoveJ [[xc,yc,367.9],[0.00171871,-0.779389,-0.626535,-0.00193248],
[-1,0,0,0],[0.0402135,9E+09,9E+09,9E+09,9E+09,9E+09]], v100, fine,
efector_final;
                ENDIF
        ENDPROC
!-----
!-----
!Rutina de posición Home-----

```



```

PROC azacalibracion()
MoveAbsJ JHome\NoEOffs, v300, z50, efector_final;
ENDPROC
!-----
!-----
!Rutina que implementa el controlador PI de Fuerza
  PROC ControladorPID()
    ClkStart time;
    tt:=ClkRead(time);
    t_vector{i}:=tt;

    Datos1;
    sensor_F:=DnumToNum(SF);
    x:=pos1.x;
    y:=pos1.y;
    z:=pos1.z;
    x_vector{i}:=x;
    y_vector{i}:=y;
    z_vector{i}:=z;
    FR:=(sensor_F*5)/1024)*24-0.1;
    FR_vector{i}:=FR;
!Trayectoria de fuerza :Función diente de sierra-----
    IF tt<=20 THEN
      Fd:=tt*0.2+8;
    ENDIF
      IF tt>=20 AND tt<=40 THEN
        Fd:=tt*0.2+4;
      ENDIF
        IF tt>=40 AND tt<=60THEN
          Fd:=tt*0.2;
        ENDIF
          IF tt>=60 AND tt<=80THEN
            Fd:=tt*0.2-4;
          ENDIF
!-----
    Fd_vector{i}:=Fd;
    e:=Fd-FR;!Error de fuerza
    e_vector{i}:=e;
!Integral del error
    ie:=ia+(h/2)*(e+ea);
!Derivada del error
    !de:=(e-ea)/h);

```

```

!Salida del controlador
  Pref:=kp*e+ki*ie;
  IF Pref*1000>=2.5 THEN
    Pref:=2/1000;
  ENDIF
  Penetracion_vector{i}:=Pref;
!Diferencia e la salida;
  dtau:=Pref-taup;
!Actualizacion de variables
  i:=i+1;
  ia:=ie;
  !da:=de;
  ea:=e;
  PF:=Prefz-Pref*1000;
  PF_vector{i}:=PF;
  pos1:=CPos();
      x:=pos1.x;
      y:=pos1.y;
      z:=pos1.z;
      e_posicion{i}:=PF-z;
MoveJ [[xc,yc,PF],[0.00171871,-0.779389,-0.626535,-0.00193248],
[-1,0,0,0],[0.0402135,9E+09,9E+09,9E+09,9E+09,9E+09]], v100,
fine, efector_final;
!-----
      IF tt>=80 THEN
        Stop;
      ENDIF
    ENDPROC
!-----
!--Datos 1-----
PROC Datos1()
!-----Entrada D0
IF DI10_9=1 THEN
  D0:=1;
  BitSet SF,1;
ELSE
  D0:=0;
  BitClear SF,1;
ENDIF
!-----
!-----Entrada D1
IF DI10_10=1 THEN

```

```

        D1:=1;
        BitSet SF,2;
ELSE
        D1:=0;
        BitClear SF,2;
ENDIF
!-----
!-----Entrada D2
IF DI10_11=1 THEN
        D2:=1;
        BitSet SF,3;
ELSE
        D2:=0;
        BitClear SF,3;
ENDIF
!-----
!-----Entrada D3
IF DI10_12=1 THEN
        D3:=1;
        BitSet SF,4;
ELSE
        D3:=0;
        BitClear SF,4;
ENDIF
!-----
!-----Entrada D4
IF DI10_13=1 THEN
        D4:=1;
        BitSet SF,5;
ELSE
        D4:=0;
        BitClear SF,5;
ENDIF
!-----
!-----Entrada D5
IF DI10_14=1 THEN
        D5:=1;
        BitSet SF,6;
ELSE
        D5:=0;
        BitClear SF,6;
ENDIF

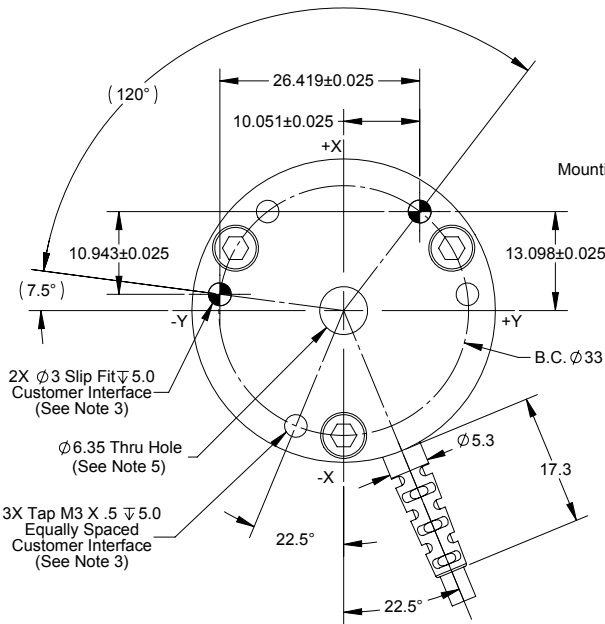
```

```
!-----
!-----Entrada D6
IF DI10_15=1 THEN
    D6:=1;
    BitSet SF,7;
ELSE
    D6:=0;
    BitClear SF,7;
ENDIF
!-----
!-----Entrada D7
IF DI10_16=1 THEN
    D7:=1;
    BitSet SF,8;
ELSE
    D7:=0;
    BitClear SF,8;
ENDIF
!-----Entrada D8
IF DI10_1=1 THEN
    D8:=1;
    BitSet SF,9;
ELSE
    D8:=0;
    BitClear SF,9;
ENDIF
!-----Entrada D9
IF DI10_2=1 THEN
    D9:=1;
    BitSet SF,10;
ELSE
    D9:=0;
    BitClear SF,10;
ENDIF
ENDPROC
!-----
ENDMODULE
```

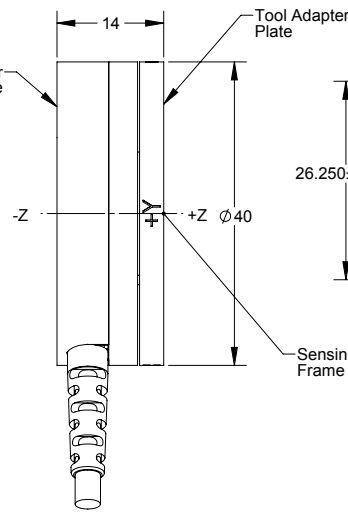
Apéndice C

Especificaciones del sensor de fuerza y componentes electrónicos

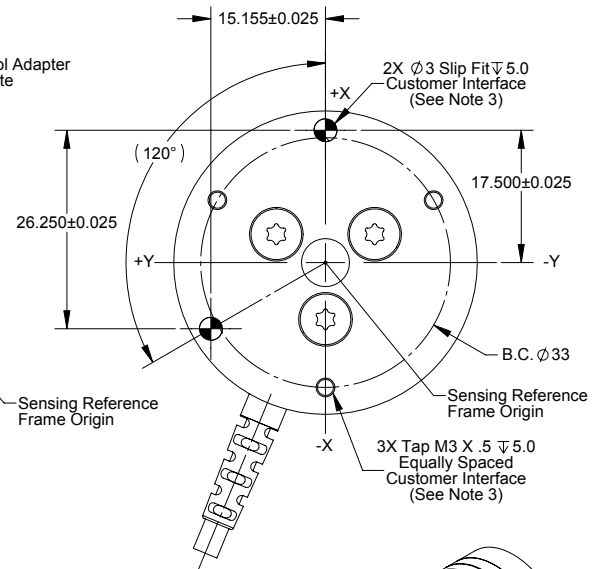
Rev.	Description	Initiator	Date
08	ECO 14323: Increased tolerance on dowel hole positions from ± 0.013 to ± 0.025	SZJ	3/28/2016



MOUNTING SIDE

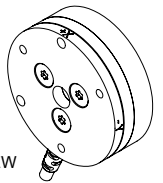


SIDE VIEW



TOOL SIDE

ISOMETRIC VIEW



Notes:

1. Material: Mounting and Tool Adapter made of Aluminum, Transducer made of hardened stainless steel.
2. **WARNING: DO NOT LOOSEN OR REMOVE INTERFACE PLATES OR CABLE DUE TO POTENTIAL DAMAGE.**
3. **DO NOT EXCEED INTERFACE DEPTH. MAY CAUSE DAMAGE**
4. Standard connector (not shown) has 17mm diameter & is 67.5mm long.
5. Do not allow items passing through center hole to touch both the Mounting and Tool Adapter Plates or accuracy will be compromised.

NOTES: UNLESS OTHERWISE SPECIFIED.

DO NOT SCALE DRAWING. ALL DIMENSIONS ARE IN MILLIMETERS.



1031 Goodworth Drive, Apex, NC 27539, USA
 Tel: +1.919.772.0115 Email: info@ati-ia.com
 Fax: +1.919.772.8259 www.ati-ia.com
 ISO 9001 Registered Company

PROPERTY OF ATI INDUSTRIAL AUTOMATION, INC. NOT TO BE REPRODUCED IN ANY MANNER EXCEPT ON ORDER OR WITH PRIOR WRITTEN AUTHORIZATION OF ATI.



DRAWN BY: J. Parker 3/14/07	TITLE: Mini40-E Transducer with Strain Relieved Cable
CHECKED BY: P.Fusco 4/18/07	SCALE: 2:1
PROJECT # -	SIZE: B
SHEET 1 OF 1	DRAWING NUMBER: 9230-05-1314
	REVISION: 08

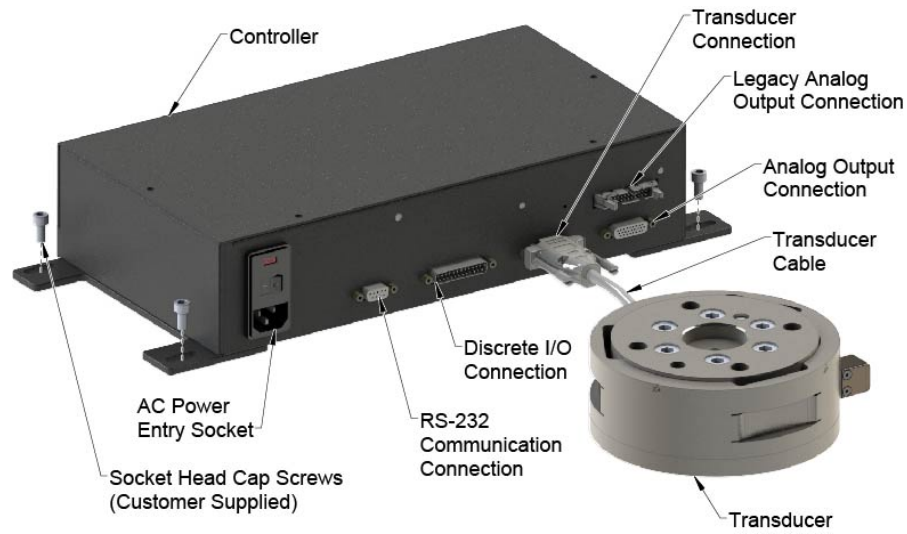


Figure 2.3—F/T Controller Connected to a Transducer without a Mux Box

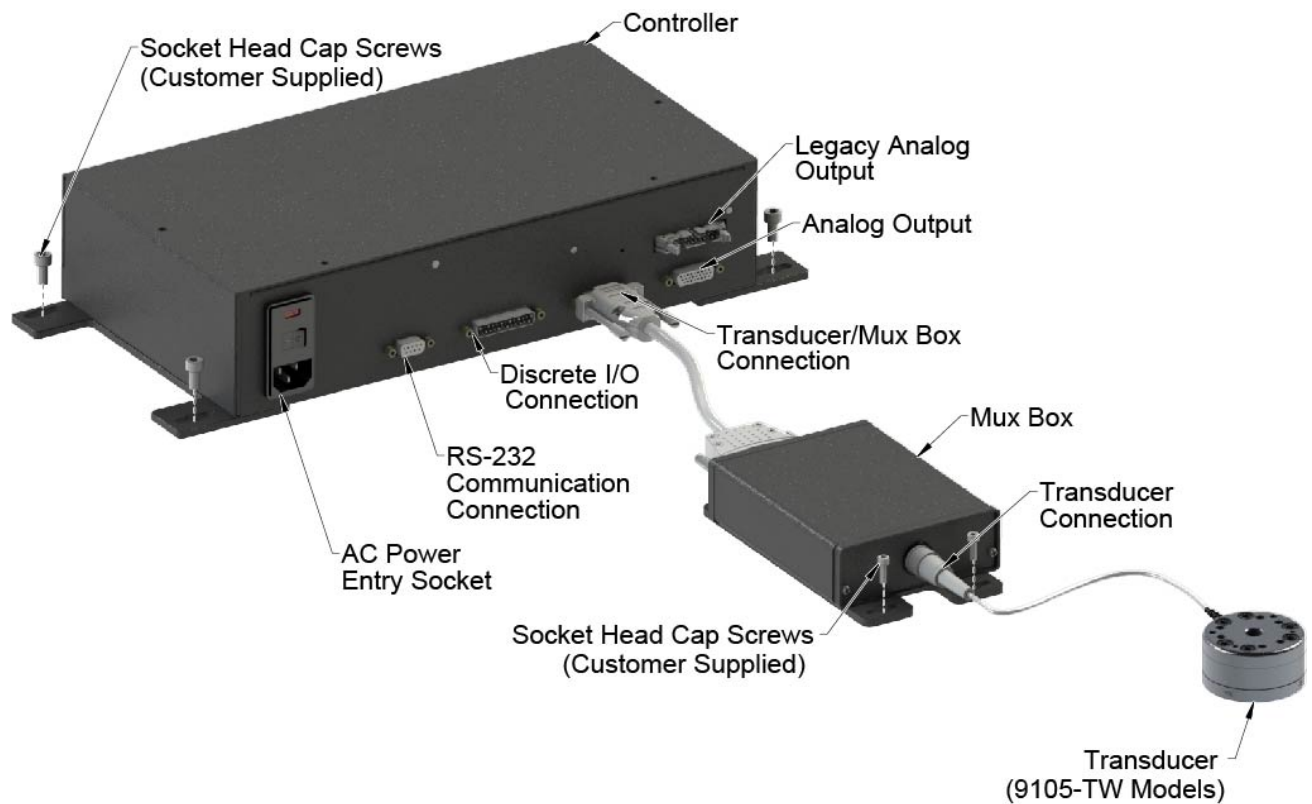


Figure 2.4—F/T Controller Connected to a 9105-TW Transducer and Mux Box

NOTE: The total length of cabling connected to a F/T Controller connector cannot exceed 30 m (98' 5").

Bibliografía

- [1] Rogelio Portillo-Vélez, Alejandro Rodríguez Ángeles & Carlos A. Cruz Villar (2010). An Optimal Admittance Approach for Safe Robot-Environment Interaction. Memorias del Congreso Anual de la Asociación Mexicana de Robótica (COMRob). Mazatlán, Sinaloa, México.
- [2] Hogan, N. (1984). Impedance control: An approach to manipulation. In American Control Conference, IEEE Transactions of the ASME, 107(1), (pp. 304-313).
- [3] Hogan, N. (1985). Impedance control: An approach to manipulation: Part II Implementation. Journal of dynamic systems, measurement and control, 107(1), (pp. 8-16).
- [4] Raibert, M. H., & Craig, J. J. (1981). Hybrid position/force control of manipulators. Journal of Dynamic Systems, Measurement, and Control, 103(2), (pp. 126-133).
- [5] Portillo-Vélez, R. D. J., Rodríguez-Ángeles, A., & Cruz-Villar, C. A. (2015). An optimization-based impedance approach for robot force regulation with prescribed force limits. Mathematical Problems in Engineering, (Vol. 2015, pp. 1-13).
- [6] Craig, J. J. (2005). Introduction to robotics: mechanics and control. Upper Saddle River: Pearson Prentice Hall, (Vol. 3, pp. 48-70).
- [7] Lewis, F. L., Dawson, D. M., & Abdallah, C. T. (1993). Control of robot manipulators. Prentice Hall PTR, Vol. 236.
- [8] Spong, M. W., & Vidyasagar, M. (2008). Robot dynamics and control. John Wiley & Sons, (Vol. 2004, pp. 1-303).
- [9] Yoshikawa, T. (2000). Force control of robot manipulators. In Robotics and Automation. Proceedings. ICRA. IEEE International Conference on (Vol. 1, pp. 220-226).
- [10] Siciliano, B., & Khatib, O. (Eds.). (2008). Springer handbook of robotics. Springer Science & Business Media.
- [11] Ogata, K. (2003). Ingeniería de control moderna. Pearson Educación.

- [12] Salisbury, J. K. (1980). Active stiffness control of a manipulator in Cartesian coordinates. *Decision and Control including the Symposium on Adaptive Processes, IEEE Conference on* (Vol. 19, pp. 95-100).
- [13] Chiaverini, S., Siciliano, B., & Villani, L. (1994). Force/position regulation of compliant robot manipulators. *IEEE Transactions on Automatic Control*, 39(3), pp. 647-652.
- [14] Chiaverini, S., Siciliano, B., & Villani, L. (1999). A survey of robot interaction control schemes with experimental comparison. *IEEE/ASME Transactions on mechatronics*, 4(3), pp. 273-285.
- [15] Lu, W. S., & Meng, Q. H. (1991). Impedance control with adaptation for robotic manipulations. *IEEE Transactions on Robotics and Automation*, 7(3), pp. 408-415.
- [16] Brumovsky, A. F., Liste, V. M., & Anigstein, M. (2006). Implementación de control de fuerzas en robots industriales: un caso. *IV Jornadas Argentinas de Robótica, JAR08. Córdoba*.
- [17] Robertsson, A., Olsson, T., Johansson, R., Blomdell, A., Nilsson, K., Haage, M., & Brantmark, H. (2006). Implementation of industrial robot force control case study: high power stub grinding and deburring. In *Intelligent Robots and Systems, IEEE/RSJ International Conference on* (Vol. 2006, pp. 2743-2748).
- [18] Nof, S. Y. (Ed.). (1999). *Handbook of industrial robotics* (Vol. 1). John Wiley & Sons.
- [19] Pires, J. N., & Godinho, T. (2008). Control difuso de fuerza en robótica industrial. *Ingenierías*, 11(41), pp 55.
- [20] Kamnik, R., Matko, D., & Bajd, T. (1998). Application of model reference adaptive control to industrial robot impedance control. *Journal of Intelligent and Robotic Systems*, 22(2), 153-163.
- [21] Blomdell, A., Bolmsjo, G., Brogardh, T., Cederberg, P., Isaksson, M., Johansson, R., & Robertsson, A. (2005). Extending an industrial robot controller: implementation and applications of a fast open sensor interface. *IEEE Robotics & Automation Magazine*, 12(3), pp. 85-94.
- [22] Chaudhary, H., Panwar, V., Sukavanam, N., & Prasad, R. (2014). ANFIS PD+I Based Hybrid Force/Position Control of an Industrial Robot Manipulator. *International Journal of Materials, Mechanics and Manufacturing*, 2(2).
- [23] Lopes, A., & Almeida, F. (2008). A force impedance controlled industrial robot using an active robotic auxiliary device. *Robotics and Computer-Integrated Manufacturing ISO 690*, 24(3), pp. 299-309.
- [24] Valera, A., Zotovic, R., & Vallés, M., Implementación del Control de Fuerza para un Robot Industrial mediante Matlab.

- [25] Hogan, N. (1984). Impedance control of industrial robots. *Robotics and Computer-Integrated Manufacturing*, 1(1), 97-113.
- [26] Drake, S. H. (1978). Using compliance in lieu of sensory feedback for automatic assembly (Doctoral dissertation, Massachusetts Institute of Technology).
- [27] <http://www.roboticastreet.com/robots-de-abb-ayudan-a-embotellar-agua-en-el-desierto/>
- [28] <http://new.abb.com/products/robotics/application-equipment-and-accessories/integrated-force-control>
- [29] http://www.ati-ia.com/es-MX/products/ft/ft_models.aspx?id=Mini40
- [30] https://library.e.abb.com/public/6aeb483836740e11c1257b4b0052375b/3HAC032104-005_revE_es.pdf
- [31] <https://www.arduino.cc/en/Guide/ArduinoMega2560>