



UNIVERSIDAD NACIONAL AUTONOMA DE
MEXICO

Escuela Nacional de Estudios Profesionales
Campus Aragón



Adaptación de dos robots industriales
para su utilización en el desarrollo de
nuevas técnicas y algoritmos de control

T E S I S

QUE PARA OBTENER EL TÍTULO DE
Ingeniero Mecánico Electricista

AREA: ELÉCTRICA ELECTRÓNICA

P R E S E N T A:

Adrian Miguel } Castillo Sánchez

ASESOR: DR. MARCO ANTONIO ARTEAGA PÉREZ

México, 2002

TESIS CON
FALLA DE ORIGEN



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

PAGINACION DESCONTINUA

Agradecimientos

A Dios por darme vida para realizar mis metas.

A mi Madre Lucía Adriana Sánchez Razo que me apoya y me ha enseñado el valor de la tolerancia. Gracias ¡mami! por el amor y cuidados hacia mí.

A mi Padre Felipe Castillo Valencia que me enseñó el valor del trabajo y de la responsabilidad. gracias por tu eterno amor y tu apoyo.

A mi Hermana Daniela que es el vivo ejemplo de la rebelión y la justicia. Eres una gran amiga ¡Sigue adelante!

A mi Hermano Rodrigo por su amistad y confianza. Que nunca se te quite el ímpetu de luchar por cosas que te gustan.

A mi Novia Jocelyn por el amor y apoyo que siempre me ha dado. Esto es uno de nuestros éxitos juntos ¡Te amo!

A mis Abuelos Ma. de la Luz Razo y Leobardo Sánchez por su eterno e incondicional apoyo.

A mi Abuela Inés Valencia por su cariño.

A mi Abuelo Froylán Castillo(†) que le hubiera gustado estar aquí.

A mis Tíos Paty, Víctor (Mayo), Mary, Héctor, Carmen, Gaby, Rubén, Alex, Vero, Pepe, Julieta, Celia, Genaro, Martha, Rafa, Tere y Susy.

A mis Primos Michelle, Mariana, Nancy, Nelly, Héctor, Bryan, Karla, Valeria, Ana, Vanessa, Johanna, Alan, Marlene, Marco, Michael, Frida, Germán, Karina y Gerardo.

A mis Amigos Jorge, Juan, Enrique, Angélica y Eduardo del Laboratorio de Robótica, que sin su ayuda hubiera sido más difícil lograr esto.

TESIS CON
FALLA DE ORIGEN

A mis Maestros que han sido parte importante en mi formación académica y personal, en especial al Dr. Marco Arteaga que me dio la oportunidad de desarrollarme en el campo que me gusta la "Robótica".

A la Universidad Nacional por el orgullo de pertenecer a esta institución. A la ENEP Aragón por mi formación como ingeniero, un especial reconocimiento a esta escuela.

Al Consejo Nacional de Ciencia y Tecnología por el apoyo para la realización de este trabajo.

Espero no ofender a nadie por omitir su nombre, agradezco a todas las personas que he conocido en mi vida porque de todas ellas he aprendido algo ¡Gracias a todas ellas!

Índice General

Agradecimientos	iii
1 Introducción	1
1.1 Estado del arte	4
2 Problemática de adaptar un robot industrial a un robot experimental	7
2.1 Rediseño e ingeniería inversa	7
2.2 Expectativas al adaptar un robot industrial a un robot experimental	11
2.3 Problemas de adaptación y metodología para solucionarlos	14
3 Adaptación de dos robots industriales a robots experimentales	17
3.1 Identificación y caracterización de los sistemas A465 y A255	17
3.2 Análisis del controlador C500 e inhabilitación de la tapa de control	22
3.2.1 Creación de buses internos	30
3.3 Descripción de la tarjeta de adquisición de datos	31
3.4 Accesorios para la adaptación	34
3.4.1 UMI-Flex6	34
3.4.2 Chasis UMI	35
3.5 Acople de señales y el sistema adaptado	38
3.6 Solución para el problema del control de hasta 6 articulaciones	42
3.6.1 Configuración para controlar 6 ejes del robot A465	43
3.6.2 Configuración para controlar 5 ejes del robot A255	46
4 Validación y resultados experimentales	49
4.1 Lenguaje de programación y programa básico de E/S	49
4.1.1 LabWindows/CVI y Software FlexMotion	49
4.1.2 Programa básico de entrada salida	51
4.2 Control PID programado	53
4.2.1 Implementación digital del controlador PID	54
4.3 Sintonización de las ganancias del control PID	55
4.3.1 Método de ajuste de Ziegler-Nichols (Z-N)	55

TESIS CON
FALLA DE ORIGEN

4.4	Creación de rutinas	57
4.5	Planeación de trayectorias	60
4.6	Obtención de la cinemática inversa de los robots A465 y A255	64
4.6.1	Cinemática directa	65
4.6.2	Cinemática inversa	67
4.7	Resultados experimentales	74
5	Conclusiones	77
5.1	Ventajas y desventajas	77
5.2	Trabajo futuro y perspectivas	79
A	Especificaciones de los robots A465 y A255	81
B	Especificaciones de la tarjeta FlexMotion 6C	89
C	Especificaciones de la UMI-Flex 6	97
D	Conectores del sistema adaptado	99
E	Galería de fotos	111
F	Guía rápida de conexión	113
F.1	Configuración del controlador C500	113
F.2	Conexión del Chasis UMI y el controlador C500 adaptado	116
F.3	Configuración del Chasis UMI	117
F.4	Conexión del Chasis UMI y las tarjetas FlexMotion	118
G	Glosario	119

Índice de Figuras

1.1	Robots en operación en el año 1999 comparado con la predicción para el año 2003 - Fuentes: UN/ECE e IFR, octubre 2000.	2
1.2	Diagrama del sistema robot que se encuentra en el Centro de Investigación Científica y de Educación Superior de Ensenada. <i>Cortesía del CICESE</i>	4
1.3	Robot manipulador SMART 3-S y controlador C3G-9000 de COMAU.	6
2.1	Método de Reikoff para ingeniería inversa.	9
2.2	Arquitectura de REV-ENGE (Kim y Bekey, 1994).	10
2.3	Disposición general de las partes de un controlador industrial	12
2.4	Disposición propuesta de las partes del un controlador experimental	13
3.1	Componentes básicos de un sistema A255	18
3.2	Componentes básicos de un sistema A465	18
3.3	Controlador C500	19
3.4	Diagrama de bloques del sistema servo del controlador C500	20
3.5	Esquema del encoder óptico incremental	21
3.6	Diagrama de fase del encoder en cuadratura	21
3.7	Identificación de los compartimentos del controlador C500.	23
3.8	Tarjeta controladora del controlador C500.	25
3.9	Tarjeta auxiliar del controlador C500.	26
3.10	Diagrama eléctrico de los buses BP-50 y BP-60.	27
3.11	Esquema del bus BP-60.	27
3.12	Disposición original de los conectores J6 y J7 del controlador C500.	30
3.13	Disposición de los cables del controlador C500 adaptado.	31
3.14	Localización de las partes de la tarjeta PCI-FlexMotion 6C	32
3.15	Recursos de un servocje en la tarjeta PCI-FlexMotion-6C.	33
3.16	Diagrama de partes de la UMI.	35
3.17	Chasis UMI	36
3.18	Diagrama interno del Chasis UMI.	37
3.19	Diagrama del conector UMIDB15.	38

TESIS CON
FALLA DE ORIGEN

3.20 a) Conexión estándar de pines del conector UMIDB15 para controlar las tres primeras articulaciones del robot b) Conector llave (DB - 15 hembra) que configura el UMIDB15 de la forma estándar.	39
3.21 Esquema del sistema adaptado para los robots A255 y A465.	41
3.22 interfaz unidireccional de 6 vías - UDI6	42
3.23 Módulo de conexiones de la interfaz unidireccional - UDI6	43
3.24 Esquema del sistema adaptado para controlar las seis articulaciones del robot A465.	45
3.25 Conexiones entre los conectores UMIDB15 de los Chasis UMI cuando se utiliza la interfaz UDI6 para el robot A465.	45
3.26 Esquema del sistema adaptado para controlar las cinco articulaciones del robot A255.	47
3.27 Conexiones entre los conectores UMIDB15 de los Chasis UMI cuando se utiliza la interfaz UDI6 para el robot A255.	47
4.1 Relación entre los elementos de un programa en LabWindows/CVI.	50
4.2 Programa básico de entrada / salida.	51
4.3 Lazo de control en retroalimentación.	53
4.4 Diagrama de un PID analógico.	54
4.5 Diagrama de un PID discreto.	55
4.6 Lazo cerrado solo con ganancia proporcional.	56
4.7 Diagrama de período crítico.	56
4.8 Rutina - Home A465—.	57
4.9 Rutina - Ready A465—.	58
4.10 Rutina - Home y Ready A255—.	59
4.11 Rutina - Mantener en posición actual—.	60
4.12 Trayectoria deseada en coordenadas rectangulares X (—), Y (- -), Z (- ·) para el robot A465.	63
4.13 Trayectoria deseada en coordenadas rectangulares X (—), Y (- -), Z (- ·) para el robot A255.	63
4.14 Isométrico de las trayectorias deseadas para los robots A465 y A255.	64
4.15 Esquema y diagrama de cuerpo libre para el robot A255.	66
4.16 Esquema y diagrama de cuerpo libre para el robot A465.	67
4.17 Plano $X_0 - Y_0$	68
4.18 Diagrama vectorial de las articulaciones 2 y 3 de los robots.	69
4.19 Ángulos de las articulaciones del robot A255.	72
4.20 Ángulos de las articulaciones del robot A465.	73
4.21 Fotografías del sistema adaptado en acción.	74
4.22 Trayectoria deseada (—) vs trayectoria real (- -) de las articulaciones del robot A255 y su correspondiente error de seguimiento.	75

4.23 Trayectoria deseada (—) vs trayectoria real (- - -) de las articulaciones del robot A465 y su correspondiente error de seguimiento.	76
A.1 Área de trabajo del robot A465 (vista lateral)	82
A.2 Área de trabajo del robot A465 (vista superior)	82
A.3 Área de trabajo del robot A255 (vista lateral)	83
A.4 Área de trabajo del robot A255 (vista superior)	83
B.1 Asignación de los 100 pines del conector de E/S de movimiento.	93
B.2 Asignación de los 50 pines del conector auxiliar de E/S digital de 24 bits.	95
D.1 Conector DB-25[I] del controlador C500.	100
D.2 Conector DB-25[II] del controlador C500.	101
D.3 Conector DB-25[1] del controlador C500.	102
D.4 Conector DB-25[2] del controlador C500.	103
D.5 Conector DB-15 del Chasis UMI.	104
D.6 Conector DB-9 del Chasis UMI.	105
D.7 Conector DB-25[1] del Chasis UMI.	106
D.8 Conector DB-25[2] del Chasis UMI.	107
D.9 Conector DB-25[I] del Chasis UMI.	108
D.10 Conector DB-25[II] del Chasis UMI.	109
E.1 Robots: a) A465, b) A255.	111
E.2 Panel frontal del controlador C500: a) A255, b) A465.	112
E.3 Controlador C500: a) original, b) adaptado.	112
E.4 Panel trasero del controlador C500 adaptado.	112
F.1 Conectores del controlador C500 original.	113
F.2 Buses BJ6 y BJ7.	114
F.3 Conectores del controlador C500 original.	115
F.4 Buses BJ6 y BJ7.	115
F.5 Chasis UMI	116
F.6 Conector llave 1	117
F.7 Conector llave 2	118

**TESIS CON
FALLA DE ORIGEN**



Índice de Tablas

1.1	Robots en operación en el año 1999 y predicción para el año 2000 y 2003.	3
1.2	Características del robot del CICESE.	5
1.3	Características del robot SMART 3-S.	6
3.1	Configuración y especificaciones de los robots A465 y A255	19
3.2	Características de voltaje y corriente de los amplificadores de los robots A465 y A255	20
3.3	Relación entre el desplazamiento angular y el número de cuentas en los encoders de las articulaciones de los robots A465 y A255.	22
3.4	Módulos electrónicos del controlador C500	22
3.5	Descripción de los conectores de la tarjeta CB	24
3.6	Distribución de los pines en el conector J6 del controlador C500 y su importancia en el sistema adaptado.	28
3.7	Distribución de los pines en el conector J7 del controlador C500 y su importancia en el sistema adaptado.	29
3.8	Función de cada conector del Chasis UMI.	36
3.9	Correspondencia de las señales entre el C500 y el Chasis UMI	39
3.10	Configuración de la tarjeta FlexMotion 1 y el conector UMIDB15 del Chasis UMI 1 para controlar las tres primeras articulaciones del robot A465.	40
3.11	Configuración de la tarjeta FlexMotion 2 y el conector UMIDB15 del Chasis UMI 2 para controlar las tres primeras articulaciones del robot A255.	40
3.12	Correspondencia entre los pines de los conectores A y B de la Interfaz Unidireccional de 6 vías.	43
3.13	Configuración de la tarjeta FlexMotion 1 y el conector UMIDB15 del Chasis UMI 1 para controlar las seis articulaciones del robot A465.	44
3.14	Configuración de la tarjeta FlexMotion 2 y el conector UMIDB15 del Chasis UMI 2 para controlar las seis articulaciones del robot A465.	44
3.15	Configuración de la tarjeta FlexMotion 1 y el conector UMIDB15 del Chasis UMI 1 para controlar las cinco articulaciones del robot A255.	46

3.16 Configuración de la tarjeta FlexMotion 2 y el conector UMIDB15 del Chasis UMI 2 para controlar las cinco articulaciones del robot A255. . .	46
4.1 Configuración de las tarjetas FlexMotion 1 y FlexMotion 2 para controlar las tres primeras articulaciones de los robots A465 y A255. . .	52
4.2 Parámetros de ajuste (método de oscilación).	56
4.3 Parámetros de los robot A255 y A465.	65
A.1 Rango de trabajo y velocidad de los robot A465 y A255	81
A.2 Asignación de pines del conector J6 de la tarjeta CB del controlador C500.	84
A.3 Asignación de pines del conector J6 de la tarjeta CB del controlador C500, continuación.	85
A.4 Asignación de pines del conector J7 de la tarjeta CB del controlador C500.	86
A.5 Características del controlador C500.	87
B.1 Asignación de los 100 pines del conector de E/S de movimiento. . . .	94
D.1 Asignación de pines del conector DB-25[I] del controlador C500 - C500DB25[I].	100
D.2 Asignación de pines del conector DB-25[II] del controlador C500 - C500DB25[II].	101
D.3 Asignación de pines del conector DB-25[1] del controlador C500 - C500DB25[1].	102
D.4 Asignación de pines del conector DB-25[2] del controlador C500 - C500DB25[2].	103
D.5 Asignación de pines del conector DB-15 del Chasis UMI - UMIDB15. . .	104
D.6 Asignación de pines del conector DB-9 del Chasis UMI - UMIDB09. . .	105
D.7 Asignación de pines del conector DB-25[1] del Chasis UMI - UMIDB25[1]. .	106
D.8 Asignación de pines del conector DB-25[2] del Chasis UMI - UMIDB25[2]. .	107
D.9 Asignación de pines del conector DB-25[I] del Chasis UMI - UMIDB25[I]. .	108
D.10 Asignación de pines del conector DB-25[II] del Chasis UMI - UMIDB25[II]. .	109

Capítulo 1

Introducción

La Robótica es una tecnología que surgió como tal aproximadamente hacia el año 1960; desde entonces han transcurrido muchos años y el interés que ha despertado es superior a cualquier previsión que en su nacimiento se pudiera formular. Siguiendo un proceso paralelo a la introducción de las computadoras en las actividades cotidianas de la vida del hombre, la robótica ha visto un intenso desarrollo, en concreto, de la denominada robótica industrial, de tal forma que los robots, que llegaron a ser considerados el paradigma de la automatización industrial, se han convertido en nuestros días en un elemento más, aunque importante, de dicha automatización.

Al igual que otras muchas ramas de la ciencia y la tecnología, la robótica nació llena de promesas de futuro de un desarrollo tan rápido e intenso que, se suponía, en pocos años alcanzaría metas que en aquellos momentos correspondían al ámbito de la ciencia-ficción. Las aportaciones de una informática en continuo desarrollo, junto a las novedosas metodologías de control e inteligencia artificial, permitían prever la disponibilidad de robots dotados de una gran flexibilidad y capacidad de adaptación al entorno, que invadirían todos los sectores productivos de forma imparable. Esto se ha vuelto realidad solo parcialmente y, en algunos aspectos, muy por debajo de las previsiones. Pasadas las primeras etapas de un desarrollo realmente vertiginoso y en muy diversos frentes, los problemas prácticos han frenado considerablemente las expectativas y obligado a reconocer que el avance va a ser mucho más lento de lo esperado.

No obstante, la robótica industrial ha alcanzado un elevado grado de madurez y la compra e instalación de robots industriales en los entornos productivos ha dejado de ser una aventura para convertirse en una opción razonable en muchos contextos de la automatización.

Aunque la apariencia de los robots industriales no ha cambiado significativa-

TESIS CON
FALLA DE ORIGEN

mente y muchos modelos actuales tienen una estructura y funcionamiento similares a los primeros prototipos, lo cierto es que tanto en los aspectos mecánicos como en el control y la programación los avances han sido importantes. Los robots actuales son más robustos, rápidos y fiables. Su capacidad de carga y repetitividad es comparativamente superior y su programación se ha facilitado considerablemente. En particular los robots industriales realizan tareas repetitivas y se emplean en gran escala en la industria automotriz, en la electrónica, y otras donde se utilizan para armar o ensamblar automáticamente una gran variedad de productos. Los robots pueden taladrar, poner componentes, soldar, pintar, transportar piezas, etc.

Las ventas de robots industriales se han incrementado en los últimos años, como se puede ver en la Tabla 1.1 y en la Figura 1.1. De acuerdo con las estadísticas de la *International Federation of Robotics* para el final del año 1999 había 742,500 robots operando en la industria alrededor del mundo, y se estima que para el año 2003 habrá alrededor de 862.000 robots operando en el mundo (IFR UN/ECE 1999, IFR UN/ECE 2000).

Por lo anterior, se puede contemplar la robótica como una ciencia que, a pesar de los grandes avances conseguidos, todavía ofrece un amplio campo para el desarrollo y la innovación. Es precisamente este aspecto el que motiva a muchos investigadores y aficionados a los robots a seguir adelante planteando robots cada vez más evolucionados.

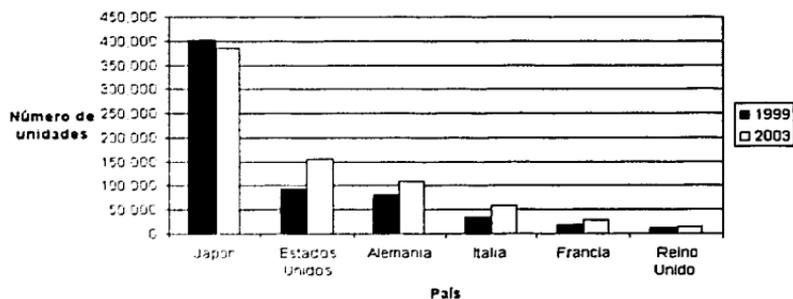


Figura 1.1: Robots en operación en el año 1999 comparado con la predicción para el año 2003 - Fuentes: UN/ECE e IFR, octubre 2000.

Una de estas formas de evolución se puede observar en las técnicas y algoritmos de control, los cuales han tenido un gran desarrollo en los últimos años. En

País	Número de unidades al final del año.					
	1990	1994	1998	1999	2000	2003
Japón	274,200	377,000	411,800	402,200	381,900	384,700
Estados Unidos	39,000	57,100	81,700	92,900	105,100	155,400
Alemania	27,300	45,300	73,200	81,200	89,400	109,500
Italia	12,200	20,600	31,500	35,000	39,300	57,600
Francia	8,400	12,300	16,200	18,200	20,000	28,200
Reino Unido	5,900	8,100	10,800	11,500	12,200	14,900
Rep. de Corea	3,000	12,700	31,400	33,700		
España	2,100	4,200	8,600	10,500		
Federación Rusa		20,000	10,000	10,000		
Otros países	87,300	8,400	45,200	47,300	109,700	111,700
Total	459,400	586,100	720,400	742,500	757,600	862,000

Fuentes: United Nations Economic Commission for Europe (UN/ECE)

+ International Federation of Robotics (IFR), octubre, 2000

Tabla 1.1: Robots en operación en el año 1999 y predicción para el año 2000 y 2003.

este sentido, cabe mencionar el control adaptable, el control por modos deslizantes, las técnicas de pasividad, el control difuso y el control neuronal, entre otros. Sin embargo, muchos de los sistemas desarrollados han sido probados únicamente en simulación y aún no han sido sometidos a una verificación experimental que permita su validación real. Por supuesto, la comprobación mediante simulación da un resultado cercano a la realidad; pero no es totalmente confiable. Esto es debido a que fenómenos como perturbaciones, incertidumbre paramétrica, o bien dinámica no modelada, entre otros, son difíciles de incluir de manera apropiada en la simulación (Gudiño y Arteaga 2000).

Por otra parte, tener laboratorios experimentales de robótica permite impulsar la investigación sobre el desempeño de nuevos algoritmos de control y probarlos en condiciones reales; además, si estos laboratorios cuentan con robots industriales proporcionan una oportunidad de probar nuevos algoritmos de control para una línea de producción antes de implementarlos en el trabajo industrial. Cabe mencionar que el alto costo en la creación de estos laboratorios es una desventaja. Esta es la principal razón por la que la mayoría de los laboratorios en México que experimentan con algoritmos de control cuentan con robots educativos y/o prototipo, los cuales no tienen las características y capacidades de un robot industrial.

TESES CON
FALLA DE ORIGEN

1.1 Estado del arte

Dentro de la literatura hay muy pocas referencias acerca de la adaptación de robots industriales de algún tipo para experimentar algoritmos de control. Como se mencionó anteriormente, existen laboratorios que prueban nuevas técnicas de control pero solamente en robots prototipo, como es el caso del Laboratorio de Robótica de la División de Física Aplicada del Centro de Investigación Científica y de Educación Superior de Ensenada (CICESE). Este laboratorio cuenta con un robot de dos grados de libertad (ver Figura 1.2). Por la información obtenida de CICESE (2000) y Kelly (2001), las dimensiones mecánicas y físicas del robot así como las especificaciones técnicas de los motores, hacen de este un dispositivo experimental de desempeño comparable a nivel de precisión, repetitividad y velocidad, a ciertos robots industriales, con la salvedad de contar sólo con dos ejes. Las principales características técnicas del robot sin carga se muestran en la Tabla 1.2.

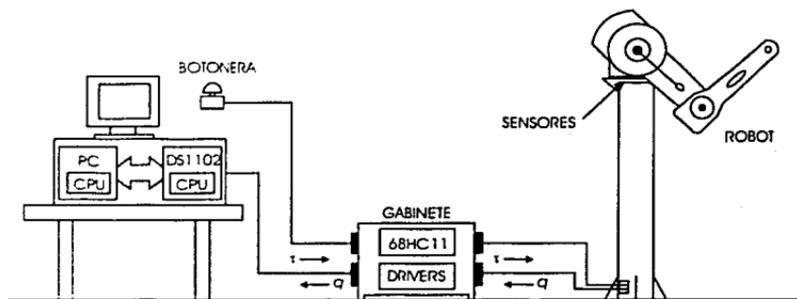


Figura 1.2: Diagrama del sistema robot que se encuentra en el Centro de Investigación Científica y de Educación Superior de Ensenada. *Cortesía del CICESE*

De las referencias encontradas más significativas están dos sistemas robóticos experimentales basados en robots industriales que se encuentran en Italia. El primero es el sistema robótico del laboratorio de Automatización y Robótica de la Universidad de Bologna, este sistema está basado en un robot industrial SMART 3-S de COMAU, un controlador C3G-9000, una *pinza* y una cámara de video. El sistema fue adaptado originalmente para probar algoritmos de control con retroalimentación por visión utilizando una plataforma de Linux en tiempo real. Posee la capacidad de comprobar diferentes algoritmos de control, en Macchelli *et al.* (2002) se describe el sistema robótico.

Número de ejes	2
Configuración de ejes	RR
Tamaño	0.45 m para brazo y antebrazo
Actuadores	eléctricos (transmisión directa)
Precisión	0.16 mm
Repetitividad	0.013 mm
Resolución	0.01 mm
Velocidad máxima	2.5 m/seg
Aceleración máxima	20 m/seg ²
Carga máxima	1.5 Kg

Tabla 1.2: Características del robot del CICESE.

Por otra parte, el Laboratorio PRISMA de la Universidad de Nápoles cuenta con un sistema de robots cooperativos, dicho sistema se basa en dos robots industriales SMART 3-S de COMAU de seis grados de libertad cada uno y dos controladores C3G-9000. Este sistema admite la programación de algoritmos de control para el sistema coordinado o para cada robot independientemente. Fue creado con la finalidad de probar algoritmos de control para robots coordinados y algoritmos de control de fuerza: por esta razón cada robot tiene un sensor de fuerza en el *efector final*. Para más información acerca del sistema robótico del laboratorio PRISMA consulte Caccavale y Chiacchio (2000) y Natale (2001).

Se observa que los dos sistemas en Italia utilizan el robot SMART 3-S y el controlador C3G-9000. Esto es debido a dos razones principales: la primera es que la empresa COMAU que fabrica los robots es italiana, y la segunda es que el controlador de fábrica C3G-9000 tiene la versatilidad de controlar los robot con algoritmos programados desde una PC con la ayuda de una tarjeta de adquisición de datos que provee COMAU. La Figura 1.3 muestra un esquema del robot SMART-3 S y el controlador C3G-9000 de COMAU mientras que la Tabla 1.3 describe las características principales del robot manipulador.

De acuerdo a lo expuesto antes, no existen muchos laboratorios donde se verifiquen algoritmos de control con robots de características industriales. En esta tesis se propone un sistema robótico basado en la adaptación de dos robots industriales para la experimentación y comprobación de algoritmos de control programados desde una computadora personal. El presente trabajo está organizado como sigue: en el Capítulo 2 se discute la problemática general de adaptar un robot industrial para experimentar con algoritmos de control, tocando puntos como: la ingeniería inversa y el rediseño de sistemas, los objetivos deseados de un sistema adaptado, además de proponer una metodología para llevar a cabo la adaptación de un sistema robot

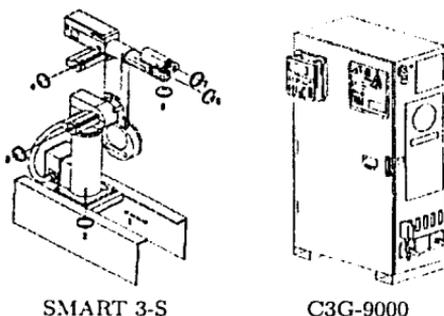


Figura 1.3: Robot manipulador SMART 3-S y controlador C3G-9000 de COMAU.

Número de ejes	6
Configuración de ejes	Rotacional
Actuadores	Eléctricos de DC
Precisión	± 0.10 mm
Repetitividad	± 0.10 mm
Resolución	0.01 mm
Velocidad máxima	2.5 m/seg
Aceleración máxima	20 m/seg ²
Carga máxima	8 Kg

Tabla 1.3: Características del robot SMART 3-S.

industrial a un sistema experimental. En el Capítulo 3 se describen los elementos del equipo con el cual se realizó la adaptación, además de detallar la experiencia de crear un sistema robótico experimental basado en dos robots industriales. En el Capítulo 4 se presenta la validación y los resultados experimentales obtenidos de programar un control Proporcional-Integral-Derivativo (PID) en el sistema adaptado para llevar a cabo la manipulación cooperativa de un objeto. Finalmente, en el Capítulo 5 se expresan las conclusiones además de incluir algunas perspectivas a futuro.

TESIS COMAU
FALLA DE ORIGEN

Capítulo 2

Problemática de adaptar un robot industrial a un robot experimental

El problema de adaptar un robot industrial a un robot experimental es básicamente que las especificaciones internas del sistema original no están disponibles para el usuario, estas especificaciones son considerados los "secretos" de diseño y construcción por parte del fabricante. Esta situación deriva en tener que aplicar técnicas de ingeniería inversa y re-ingeniería para alcanzar un entendimiento del diseño del sistema debido a que los secretos del fabricante no están disponibles para realizar el proceso de rediseño. En esencia, una ingeniería inversa con éxito da lugar a la obtención de una o más especificaciones de diseño y construcción del sistema en examinación. Por esta razón es importante revisar algunos conceptos y metodologías de ingeniería inversa para aplicarlas al problema de adaptación.

2.1 Rediseño e ingeniería inversa

El término *ingeniería inversa* tiene sus orígenes en el mundo del hardware y es un intento de descubrir el diseño a partir de la máquina, en contraste con la ingeniería hacia adelante o clásica que es el intento de producir la máquina a partir del diseño. En otras palabras la ingeniería inversa es un proceso de recuperación del diseño. Esto es un concepto interesante, pues admite generalización para aplicárselo a toda la ciencia, la cual podría ser considerada como el proyecto de ingeniería inversa que busca discernir el diseño del universo en su conjunto (las leyes naturales que lo rigen).

Las herramientas de la ingeniería inversa extraen información acerca de la arquitectura y diseño del sistema o subsistema existente. Los resultados obtenidos pueden tener diferentes usos, como el diseño de nuevos productos o servicios, recuperación del diseño del sistema analizado, modificación de diseños existentes, sustitución de

TESIS CON
FALLA DE ORIGEN

partes averiadas o documentación de diseños. Las siguientes definiciones clarifican los conceptos básicos al modificar o crear un sistema.

Ingeniería hacia adelante (Forward engineering): Proceso tradicional que lleva desde abstracciones de alto nivel y diseños lógicos a la implementación física de un sistema.

Ingeniería inversa (Reverse engineering): Proceso de analizar un sistema con dos metas en mente:

- analizar los componentes del sistema y sus interrelación; y
- crear representaciones del sistema en otra forma o en un nivel más alto de abstracción.

Re-ingeniería (Reengineering): Examinación de un sistema para reconstruirlo en una nueva forma con su subsecuente puesta en práctica.

Re-estructuración (Restructuring): Transformación de una forma de representación a otra en el mismo nivel relativo de abstracción. La nueva representación está destinada a preservar el comportamiento externo de la original.

Recuperación del diseño (Design recovery): Subconjunto de la ingeniería inversa en el cual el dominio del conocimiento, la información externa y la deducción se agregan a la observación del sistema a analizar. El objetivo de la recuperación del diseño es identificar abstracciones significativas de alto nivel más allá de las obtenidas directamente examinando el sistema.

Existen varios métodos y técnicas interesantes de ingeniería inversa aplicables a hardware y a software: la mayoría de los métodos que son aplicados a hardware están enfocados a la manufactura. Revisando la literatura encontramos dos enfoques interesantes: el primero de Rekoff (1984) acerca de la ingeniería inversa aplicada a diferentes sistemas, y el segundo de Kim y Bekey (1994) acerca del rediseño de sistemas. Estos enfoques se muestran a continuación.

Para Rekoff (1984), "*ingeniería inversa* es el acto de crear un conjunto de especificaciones para sistemas hardware por medio del análisis y dimensionamiento de un espécimen". En esta definición, un sistema hardware puede ser un sistema mecánico, eléctrico o electrónico.

En su método (Figura 2.1), Rekoff sugiere elaborar una descomposición jerárquica de un producto para después analizar dos de sus elementos en forma simultánea:

artículo y elemento. El método puede ser aplicado a cualquier nivel de la descomposición.

Dos especificaciones son definidas para describir un artículo: a) la funcional, que detalla cómo un artículo y sus elementos trabajan e interactúan entre ellos; y b) la dimensional, que establece las dimensiones de la parte, su material y cómo es transformado durante la manufactura, los valores de parámetros y sus tolerancias, además de cómo es ensamblado el artículo.

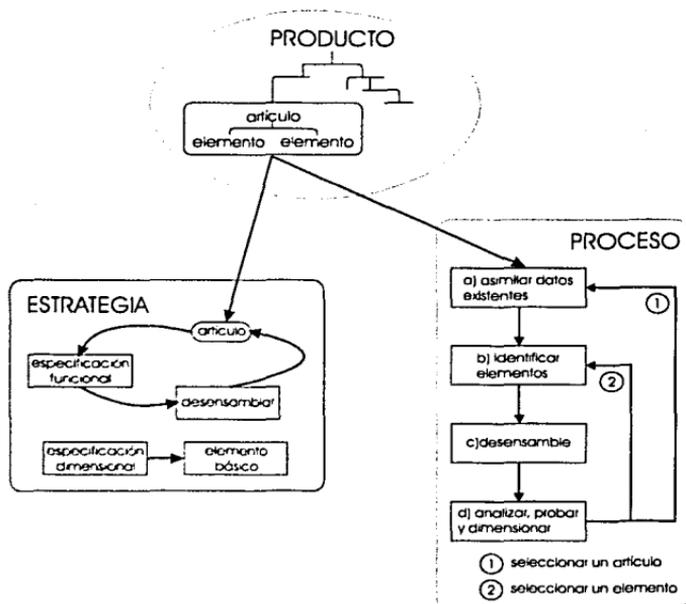


Figura 2.1: Método de Rekoff para ingeniería inversa.

De acuerdo con Rekoff, la estrategia general de la ingeniería inversa es tomar un producto, establecer una especificación funcional inicial, desensamblar el componente cuya función ha sido claramente entendida, realizar la correspondiente especificación funcional, desensamblar de nuevo y repetir el proceso. Esta secuencia de pasos es repetida hasta lograr un buen entendimiento de todos los componentes.

TESTEON
FALLA DE ...

Después de esto, cada elemento es aislado y medido para completar su especificación dimensional. De este modo, y en resumen, el método de ingeniería inversa consiste en una secuencia de cinco pasos: 1) asimilar datos existentes; 2) identificar elementos; 3) desensamblar; 4) analizar; y 5) probar y dimensionar. Estos pasos son ejecutados en forma cíclica como se ilustra en la Figura 2.1.

Por otra parte, Kim y Bekey (1994) presentan un proceso de rediseño basado en la secuencia de diseño. Para ellos, ingeniería inversa es la "actividad de: a) adquirir o inferir el proceso y la secuencia de diseño usados para crear un producto dado; y b) emplear el conocimiento inferido para recrear el diseño o rediseñar un producto".

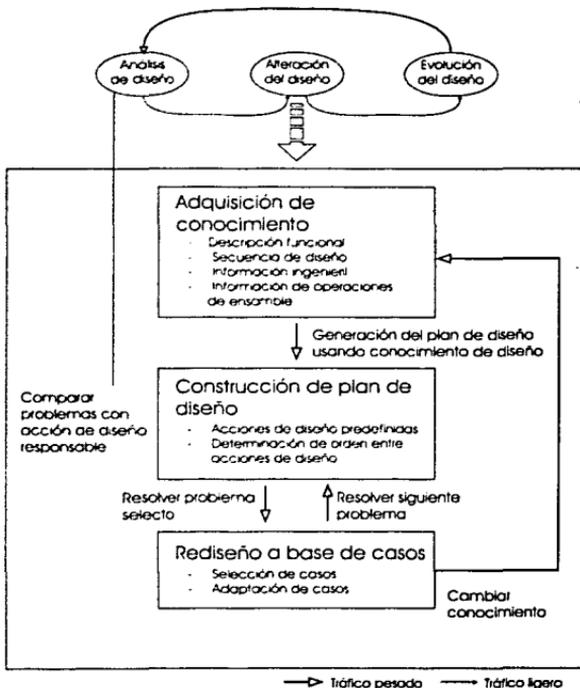


Figura 2.2: Arquitectura de REV-ENGE (Kim y Bekey, 1994).

Estos autores sostienen que si un sistema o producto ha de ser rediseñado pero no se tiene acceso a su proceso de diseño, entonces, la ingeniería inversa puede ser usada para asumir uno. El proceso de diseño obtenido es empleado para rediseñar el sistema.

Kim y Bekey proponen un sistema para asistir el rediseño para ensamble, capaz de aplicar ingeniería inversa a componentes, creando un proceso de diseño basado en una secuencia de diseño generada. El sistema REV-ENGE, que se muestra en la Figura 2.2, opera bajo el supuesto de un modelo de diseño de tres etapas: análisis, alteración y evaluación. Primeramente, REV-ENGE modela la alteración del diseño como un ciclo de tres actividades principales: adquisición de conocimiento, construcción de un proceso de diseño y modificación del diseño.

Después de revisar los puntos de vista de Rekoﬀ, Kim y Bekey, se puede concluir que la funcionalidad general de todo el sistema debe de ser algo perfectamente comprendido antes de que se produzca un trabajo de ingeniería inversa más detallado. De este modo se establece un contexto para su posterior análisis y se proporcionan ideas generales acerca de los problemas de interoperabilidad entre aplicaciones dentro del sistema. Cuando ya hay especificaciones del sistema se revisan para apreciar si se ajustan al sistema existente. Tomando en consideración los métodos de ingeniería inversa descritos anteriormente, a continuación se hace un análisis del funcionamiento general de los sistemas de robots industriales y se analizan las expectativas de un sistema experimental.

2.2 Expectativas al adaptar un robot industrial a un robot experimental

La mayoría de los robots operando en la industria son controlados por el algoritmo de control Proporcional - Integral - Derivativo (PID). Esto es debido a que se ha encontrado en el PID una confianza y un grado de precisión aceptable. Sin embargo, el uso de nuevos algoritmos o nuevas técnicas de control podrían incrementar el desempeño y la precisión de los robots a un bajo costo, especialmente cuando el robot tenga que interactuar con el medio ambiente. Otra de las características de los robots industriales es que su controlador se puede ver como una caja negra por el usuario, donde el operador sólo envía comandos a través de una interfaz como una computadora o un *teach pendal*, y obtiene una respuesta del robot, sin importar el proceso intermedio. Así, esta caja negra varía en configuración dependiendo del tipo de robot y de la empresa que lo fabrica. En general, todos los controladores cuentan básicamente con cuatro etapas o subsistemas que son:

- **Etapa de control:** en esta etapa se encuentra el algoritmo de control y la información sobre la dinámica y cinemática del robot.
- **Etapa de potencia:** en esta se localizan los manejadores/amplificadores que acondicionan la señal de salida con la suficiente potencia y forma para operar los actuadores del robot.
- **Etapa de protección:** en esta etapa se concentran las protecciones eléctricas como: fusibles, elementos térmicos, relays, etc., que protegen al controlador y al robot contra sobrecargas.
- **Etapa de comunicaciones:** en general esta etapa se encarga de comunicar al módulo industrial con los elementos de entrada como computadoras, teach pendals, joysticks, etc. Cabe mencionar que los dispositivos de entrada sólo pueden comandar trayectorias o ejecutar rutinas, más no permiten programar el algoritmo de control.

En la Figura 2.3 se muestra la interacción entre las etapas que conforman en general un controlador industrial y su relación con el robot.

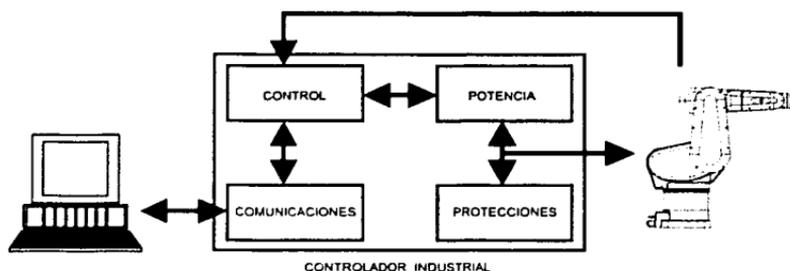


Figura 2.3: Disposición general de las partes de un controlador industrial

Para explicar la problemática de adaptación debemos analizar los objetivos y metas de convertir un robot industrial a uno experimental, poniendo atención en las necesidades e infraestructura que se tiene. A continuación se muestran las expectativas y requerimientos de un sistema robótico experimental:

TESIS CON
FALLA DE ORIGEN

- Un sistema experimental debe mantener el estándar industrial, con la capacidad y las características de un robot industrial. Una forma para alcanzar este objetivo es basar el sistema experimental en robots industriales. Otra es encargarse del diseño y fabricación de un robot con las características necesarias a alguna empresa o institución, lo cual implicaría un costo muy elevado.
- El sistema debe ser de fácil operación y comprensión para el usuario, con ideas y conceptos afines a ingenieros, técnicos y especialistas en control y robótica. Debe permitir programar de manera sencilla algoritmos de control desde una PC, utilizando programación estructurada basada en un lenguaje abierto y popular (ej. C++).
- Debe ser un sistema versátil que permita realizar experimentos con varias configuraciones del robot haciendo uso de sensores y herramientas, además de que pueda interactuar con otros robots. Dentro de esta versatilidad se puede incluir la posibilidad de regresar a su estado original (configuración de fábrica) con facilidad.
- Debe ser un sistema económico. Esto se puede lograr aprovechando la mayoría del hardware original: haciendo cambios mínimos pero significativos en el robot y su controlador para evitar el diseño o re-diseño de los subsistemas involucrados.

En la Figura 2.4 se muestra la estructura propuesta del sistema robótico experimental. Es importante señalar que la creación de un sistema robótico experimental para la implementación de algoritmos de control se complica si se desea un sistema que pueda llamarse universal, esto es, donde se puedan realizar distintos y variados experimentos, y permita que varias personas o instituciones puedan disfrutar de los beneficios del sistema. Los problemas de crear un sistema con las características descritas anteriormente se tratan a continuación.

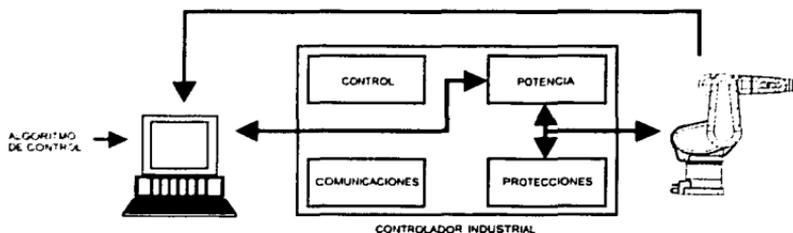


Figura 2.4: Disposición propuesta de las partes de un controlador experimental

TESIS CON
FALLA DE ORIGEN

2.3 Problemas de adaptación y metodología para solucionarlos

La problemática particular de adaptar un robot industrial a uno experimental depende de la arquitectura y configuración del robot y su controlador. Sin embargo, existen algunos problemas en común que hay que resolver en el proceso. Estos problemas se enumeran a continuación:

1. Inhibición del control de fábrica. En la mayoría de los casos el control de fábrica es un PID. Todos los sistemas de control de posición, velocidad y seguimiento de trayectorias están regidos por este controlador y es difícil su inhibición o aislamiento.
2. Arquitectura cerrada. Las partes principales del controlador (potencia, control, protecciones y comunicaciones) difícilmente pueden ser identificadas y aisladas para un análisis o uso independiente. Esto complica la sustitución del control original por uno propio.
3. No se tiene acceso al modelo dinámico del robot por parte de la empresa. Toda la información está integrada en algún lugar desconocido del controlador.
4. Insuficiente información sobre los componentes del sistema. En la mayoría de los casos la información proporcionada por el fabricante consta sólo de manuales de operación y mantenimiento, los cuales no aportan mucha información sobre la arquitectura y el funcionamiento interno del robot y su controlador.
5. Intercambio de datos entre la computadora y el robot. El acondicionamiento de las señales entre el robot y la computadora se debe hacer bajo un estándar industrial para posteriormente llevar los algoritmos desarrollados a la industria. El intercambio de datos se debe realizar con tarjetas computacionales de adquisición de datos accesibles y confiables.
6. El diseño de las etapas de potencia y protecciones. Cada empresa determina la forma de acondicionar las señales del manejador/amplificador. Si se decide no ocupar parte del controlador original, entonces se tiene que crear módulos de potencia y protección considerando las características eléctricas, mecánicas y los protocolos de comunicación del robot, los cuales no están disponibles al usuario.
7. Crear una interfaz gráfica hombre-máquina igual o mejor que la original para la programación y operación del sistema.

Las complicaciones al crear un sistema robótico experimental deben ser solucionadas mediante una metodología que considere la mayoría de los inconvenientes que se puedan presentar, sin importar el modelo o configuración del robot y su controlador. De este modo, la adaptación de un robot industrial basado en servomotores de CD para implementar algoritmos de control programados en una computadora personal, nos lleva a:

1. Identificar y caracterizar el robot industrial, estableciendo sus características físicas, mecánicas, eléctricas, etc.
2. Reconocer los subsistemas del controlador, identificando las partes principales del mismo y su funcionamiento, aunque sea de forma parcial. Esto depende en gran medida de que el controlador de fábrica sea de arquitectura abierta o cerrada.
3. Medir el nivel y rango de las señales que se presentan en los conectores externos e internos del sistema original. En el caso de contar con estos datos en los manuales es importante verificarlo.
4. Inhibir o eliminar la etapa de control original del robot, determinando si se puede desacoplar la parte de control del resto del sistema. En esta etapa es posible que sea necesaria la creación y diseño de hardware.
5. Crear o desensamblar el manejador/amplificador, el cual recibirá las señales de control de la computadora y las acondicionará para hacer funcionar los actuadores del robot (ver Figura 2.4).
6. Seleccionar una tarjeta de adquisición de datos con la capacidad de obtener la información de retroalimentación de cada una de las articulaciones del robot, además de comandar una señal adecuada a la entrada del manejador/amplificador.
7. Acoplar los nuevos sistemas a nivel hardware, garantizando la concordancia de las señales y tratando de mantener el estándar industrial.
8. Seleccionar un lenguaje de programación compatible con el hardware, en particular con las tarjetas de adquisición de datos. Dicho lenguaje debe ser accesible, versátil y con la posibilidad de crear programas basados en gráficos para facilitar la creación de una interfaz hombre-máquina.
9. Crear un programa piloto de sólo entrada/salida para probar las funciones básicas del sistema acoplado, como son compatibilidad de las señales y la respuesta del robot. En esta etapa se hace importante el uso de los resultados obtenidos en el punto 3 para no sobrepasar los límites del robot.

10. Crear una interfaz gráfica hombre-máquina que sea amigable, versátil y fácil de programar. Esta debe tener fácil acceso a los puertos de entrada/salida y monitoreo del robot.
11. Crear rutinas para inicializar el sistema como: inicialización de las tarjetas de adquisición de datos, configuración de las tarjetas, home, ready, etc.
12. Crear rutinas de protección por software. Estas rutinas o programas protegerán al robot y al sistema en general. Algunas de las rutinas son: saturación de los comandos de voltaje a un nivel adecuado, detección del límite mecánico para evitar que las partes móviles se deterioren por golpes, etc.
13. Validar el sistema mediante la implementación de algún algoritmo de control. Cumpliendo con el objetivo del sistema se deberá poder implementar de forma fácil un algoritmo de control que regule al robot.

Mediante los pasos descritos anteriormente debe ser posible adaptar un robot industrial a robot experimental. Las complicaciones propias de cada robot en particular deberán ser superadas utilizando diferentes métodos, y su complejidad dependerá de las características del sistema y de la infraestructura con la que se cuente.

Capítulo 3

Adaptación de dos robots industriales a robots experimentales

En este capítulo se describe la conversión de los robots industriales A465 y A255 de la compañía canadiense *CRS Robotics* a robots experimentales. La metodología seguida para la conversión se basa en el procedimiento enunciado en la Sección 2.3. Cabe mencionar que el equipo descrito a continuación es propiedad del Laboratorio de Robótica de la División de Estudios de Posgrado de la Facultad de Ingeniería (DEPFI) de la UNAM.

3.1 Identificación y caracterización de los sistemas A465 y A255

Los sistemas robots A465 y A255 de la marca canadiense *CRS Robotics* constan de un brazo robot y de un controlador cada uno. Los brazos robot son articulados con 6 y 5 grados de libertad, respectivamente. Las Figuras 3.1 y 3.2 muestran los componentes básicos de los sistemas A255 y A465, mientras que la Tabla 3.1 muestra las especificaciones mecánicas. Otros datos sobresalientes como el rango de trabajo y velocidad de las articulaciones se muestran en el Apéndice A.

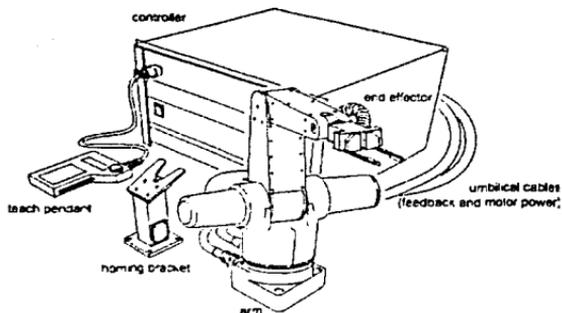


Figura 3.1: Componentes básicos de un sistema A255

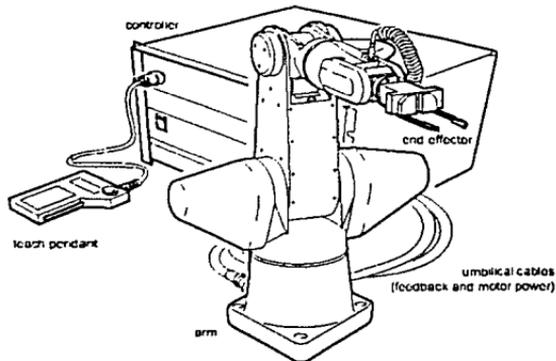


Figura 3.2: Componentes básicos de un sistema A465

Los actuadores de las articulaciones de los robots son servomotores de CD con retroalimentación por encoder óptico en cuadratura; este tipo de motores responden a la entrada de energía eléctrica como sigue: la velocidad es proporcional al voltaje y el torque es proporcional a la corriente. Con el fin de controlar los actuadores de los robots, ambos sistemas cuentan con el controlador de arquitectura abierta C500 (Figura 3.3).

Característica	ROBOT A465	ROBOT A255
Tipo	Articulado	Articulado
Grados de libertad	6	5
Montaje	De pie o invertido ¹	De pie o invertido ²
Impulsión	Servo motores de CD Encoders ópticos	Servo motores de CD Encoders ópticos
Transmisión	Impulsores armónicos	Impulsores armónicos y engranes rectos / cónicos con cadena.
Terminales del brazo	Solenoides neumáticas de 4 vías, conector de servo gripper.	Solenoides neumáticas de 4 vías, conector de servo gripper.
Especificación	Carga 2kg / 4.4lbs nom 3kg / 6.6lbs max	1kg / 2.2lbs nom 2kg / 4.4lbs max
Alcance	711mm/27.992" sin pinza 864mm/34.016" con pinza	560mm/22.047" sin pinza 660mm/25.984" con pinza
Repetitividad	-/- 0.05mm -/- 0.002"	-/- 0.05mm +/- 0.002"
Peso	31Kg/68.2lb	17kg/37.4lb

¹Base con pistas de longitud estándar para ambos montajes.

²Base con pista de longitud estándar (invertible).

Tabla 3.1: Configuración y especificaciones de los robots A465 y A255

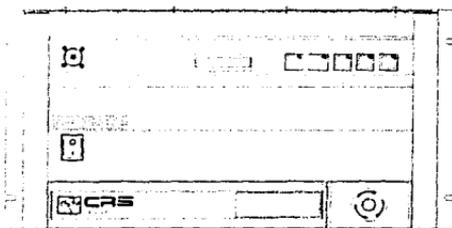


Figura 3.3: Controlador C500

El controlador C500 es la unidad electrónica computarizada con memoria que proporciona las señales de control necesarias para la operación del brazo robot. Configurado como parte de un sistema robot de CRS, el controlador contiene amplificadores para los motores servo de los brazos robot de las series A255 y A465. Las características del controlador C500 están listadas en la Tabla A.5 del Apéndice A. A grandes rasgos el sistema servo del controlador C500 opera como se ilustra en la Figura 3.4.

TESIS CON
FALLA DE ORIGEN

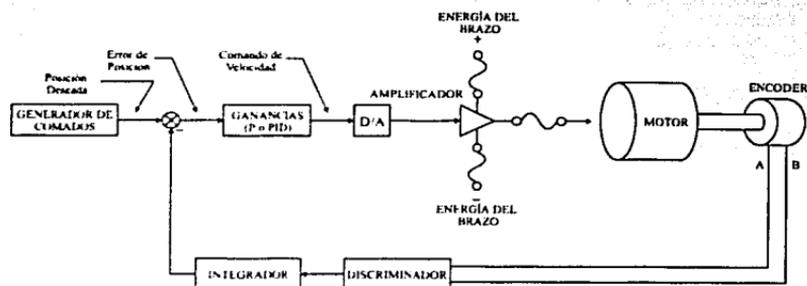


Figura 3.4: Diagrama de bloques del sistema servo del controlador C500

Del diagrama de bloques se observa que el controlador PID envía una señal de control al amplificador para regular al motor que a su vez entrega información sobre su posición mediante un encoder incremental. De forma breve se describirán las características de estas señales. La señal de control entregada por el controlador es enviada a un DAC, cuya salida es enviada al amplificador de potencia donde la señal es amplificada al nivel requerido para manejar el motor. La señal de entrada máxima hacia el amplificador es 10 VCD, esto generalmente es un estándar industrial. Cabe notar que el voltaje entregado por los amplificadores tiene una relación lineal a la señal comandada por el controlador. La Tabla 3.1 muestra la máxima salida de los amplificadores para los robots.

E_p	Comando de entrada, V_e (volts)	A465			Relación V_s/V_e	A255		
		Salida del amplificador		Relación V_s/V_e		Salida del amplificador		Relación V_s/V_e
Motor		Voltaje, V_s (volts)	Corriente (amperes)			Voltaje V_s (volts)	Corriente (amperes)	
1	± 10	± 70	12	7	± 10	± 25	2	2.5
2	± 10	± 70	12	7	± 10	± 25	2	2.5
3	± 10	± 70	12	7	± 10	± 25	2	2.5
4	± 10	± 30	3	3	± 10	± 25	2	2.5
5	± 10	± 30	3	3	± 10	± 25	2	2.5
6	± 10	± 30	3	3	---	---	---	---

Tabla 3.2: Características de voltaje y corriente de los amplificadores del robots A465 y A255

Por otra parte los codificadores ópticos (encoders) son captadores de posición angulares que suministran (en un código binario) la posición angular del eje sobre el que están montados. Los encoders de los robots A465 y A255 son del tipo incremental montados directamente en el eje del motor. Los encoders usan discos de vidrio

de calidad óptica con 1000 líneas por revolución o un disco metálico de 500 líneas por revolución, para la retroalimentación de posición. Cada disco tiene una señal de referencia (pulso) por revolución la cual es alineada durante el *procedimiento de honing*.

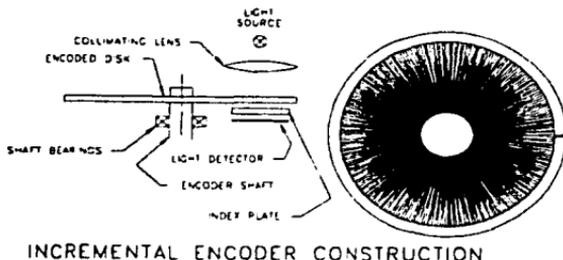


Figura 3.5: Esquema del encoder óptico incremental

Los codificadores incrementales constan de una célula fotoeléctrica o un fototransistor que detecta las variaciones de iluminación, provocadas por el paso, delante de una fuente luminosa, de trazos negros regularmente espaciados (ver Figura 3.5). Esta alternancia de fases iluminadas y sombreadas se traduce en tres trenes de pulsos con una forma de onda mostrada en la Figura 3.6. Los encoders poseen dos salidas principales (A, B), generando cada una de ellas cierto número de pulsos o cuentas por vuelta, y una tercera (index) que se utiliza para evitar error en la detección.

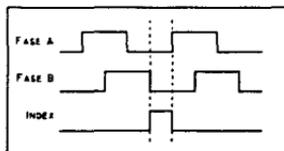


Figura 3.6: Diagrama de fase del encoder en cuadratura

El número de cuentas por revolución determina la resolución del encoder; además existe una relación entre el número de líneas contadas y el desplazamiento angular de las articulaciones de los robots (diferente al número de líneas en el encoder por el tipo de transmisión). La Tabla 3.1 muestra la relación *cuentas/desplazamiento angular* de cada una de las articulaciones de los robots A255 y A465. Cabe mencionar que los encoders de los robots también entregan el complemento de las señales mencionadas, de tal forma que se tiene $A, B, Index, \bar{A}, \bar{B}, \bar{Index}$.

Articulación	Robot A465		Robot A255	
	rad/cuentas	grados/cuentas	rad/cuentas	grados/cuentas
1	$5\pi \times 10^{-6}$	9×10^{-4}	$\pi/144000$	1.25×10^{-3}
2	$5\pi \times 10^{-6}$	9×10^{-4}	$\pi/144000$	1.25×10^{-3}
3	$5\pi \times 10^{-6}$	9×10^{-4}	$\pi/144000$	1.25×10^{-3}
4	$\pi/110000$	$90/55000$	$3.125\pi \times 10^{-5}$	5.625×10^{-3}
5	$\pi \times 10^{-5}$	1.8×10^{-3}	$6.25\pi \times 10^{-5}$	11.25×10^{-3}
6	$\pi \times 10^{-5}$	1.8×10^{-3}	—	—

Tabla 3.3: Relación entre el desplazamiento angular y el número de cuentas en los encoders de las articulaciones de los robots A465 y A255.

3.2 Análisis del controlador C500 e inhabilitación de la etapa de control

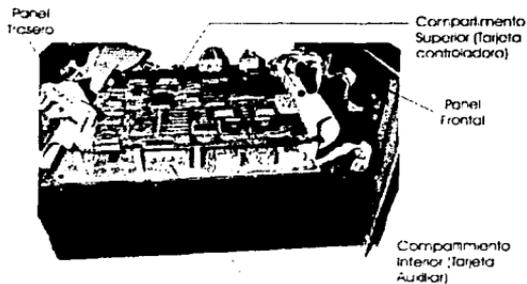
Después de haber explicado de forma breve la naturaleza de las señales del lazo de control industrial, describiremos las partes principales a nivel hardware del controlador C500 con el fin de deshabilitar la etapa de control. El hardware electrónico del controlador C500 se divide en 3 compartimentos o secciones principales (ver Figura 3.7). El compartimiento superior contiene la electrónica del control del robot. El compartimiento inferior contiene la fuente de poder de CA, la circuitería principal de encendido y la tarjeta del filtro de potencia. En el espacio sobrante, al lado de los compartimentos superior e inferior, están los amplificadores de los motores de los robots. La Tabla 3.4 lista todos los módulos electrónicos del controlador C500. Estos módulos son descritos en detalle en el Capítulo 3 de CRS (1997a).

Cantidad		Descripción	Localización
A465	A255		
1	1	Tarjeta controladora	Compartimiento superior (CB)
4	6	Amplificador lineal	Sección de amplificadores (AS)
1	1	Fuente de poder de la computadora	Compartimiento inferior (AB)
1	1	Banco de capacitores	Compartimiento inferior (AB)
1	1	Montura del transformador	Compartimiento inferior (AB)
1	1	Tarjeta del conector de encoders	Panel trasero (PT)
1	1	Montura de los fusibles del panel	Panel frontal (PF)
1	1	Tarjeta del display del panel frontal	Panel frontal (PF)
1	1	Tarjeta de regulación de CA	Compartimiento inferior (AB)
3	0	Amplificadores PWM	Sección de amplificadores (AS)

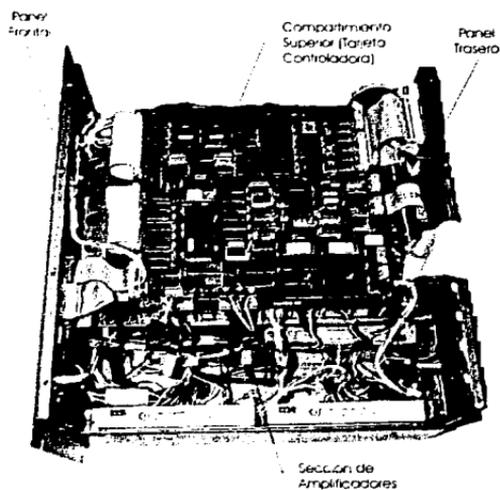
Tabla 3.4: Módulos electrónicos del controlador C500

Es importante señalar que la tarjeta controladora es responsable de las comunicaciones, procesamiento de datos y funciones de memoria del sistema robótico. Por comodidad, la tarjeta del compartimiento superior se designará *tarjeta controladora* y se identificará por las letras "CB"; de forma análoga, la tarjeta del





a. Vista lateral.



b. Vista superior

Figura 3.7: Identificación de los compartimentos del controlador C500.

TESIS CON
FALLA DE ORIGEN

compartimiento inferior se designará *tarjeta auxiliar* y se identificará por las letras "AB". Recordando el esquema de la Figura 2.4, el objetivo es deshabilitar la etapa de control y comunicaciones del controlador industrial y sólo utilizar la etapa de potencia y de protecciones eléctricas. Por este motivo esta sección se enfocará al estudio y deshabilitación de la tarjeta controladora. Las Figuras 3.8 y 3.9 muestran los esquemas de las tarjetas CB y AB respectivamente. La Tabla 3.2 muestra la descripción y etiqueta de cada uno de los conectores de la tarjeta CB.

Etiqueta	Descripción
J1	Conector No. 1 de interconexión con la tarjeta auxiliar
J2	Puerto de comunicación
J3	Conector del Teach Pendant
J4	Transputer Network UP
J5	Transputer Network DOWN
J6	Conector No. 2 de interconexión con la tarjeta auxiliar
J7	Conector de encoders
J8	Conector de la energía de los encoders
J9	Expansión de los amplificadores
J10	Fuente de poder
J13	Conector de energía de la tarjeta auxiliar

Tabla 3.5: Descripción de los conectores de la tarjeta CB

TESTE O
FALLA DE ORIGEN

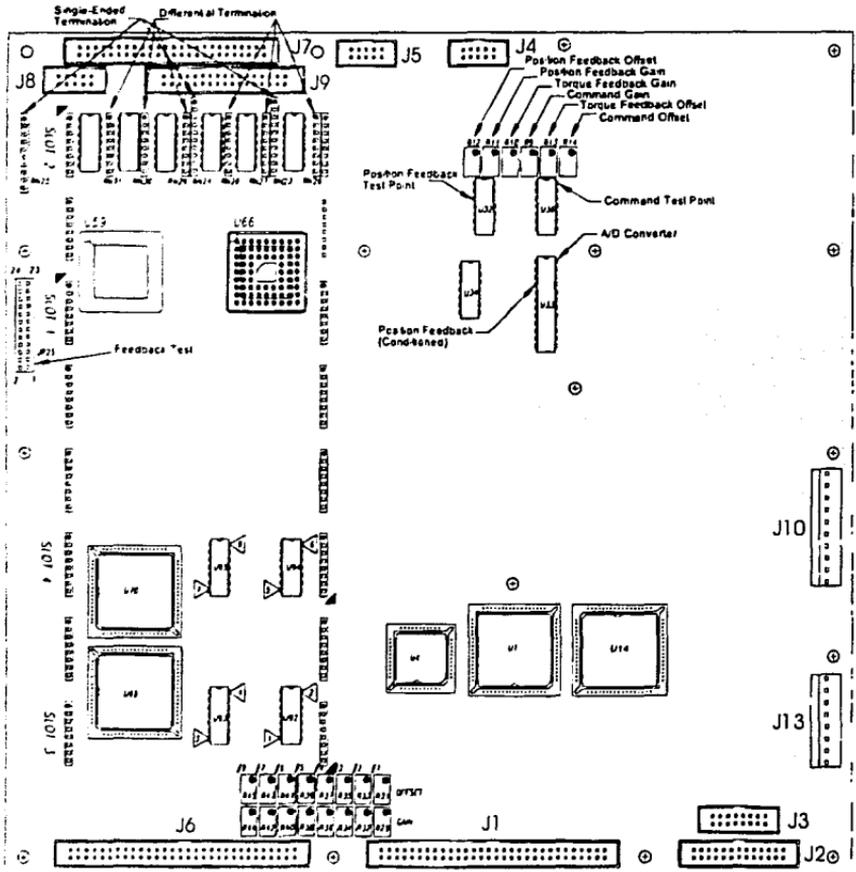


Figura 3.8: Tarjeta controladora del controlador C500.

TERMINAL
FALLA DE CONTROL

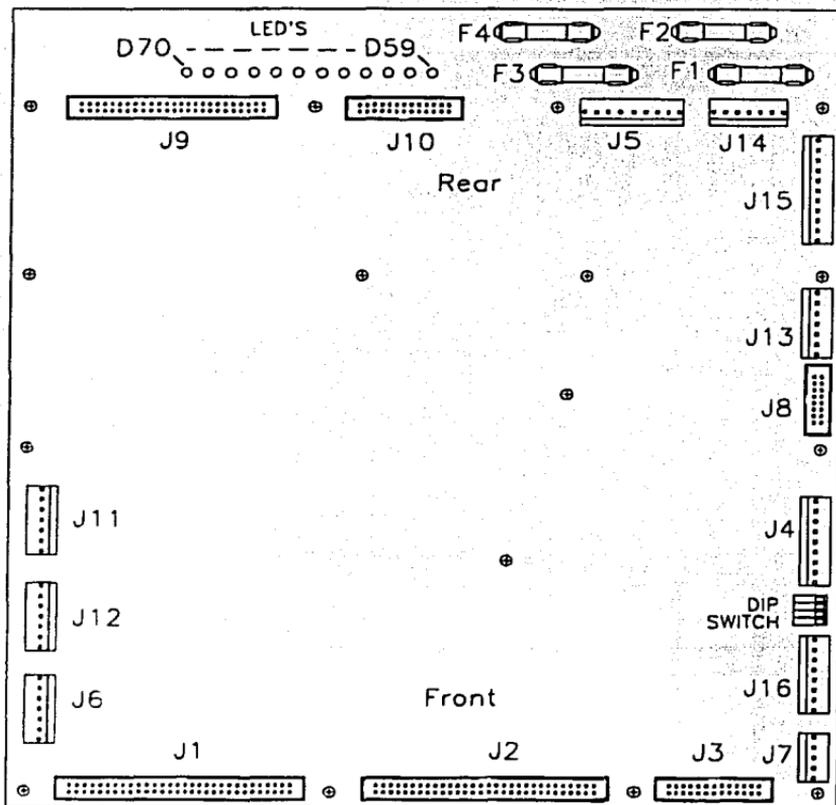


Figura 3.9: Tarjeta auxiliar del controlador C500.

Como se recordará de la Figura 2.3, existe dependencia entre las etapas de un controlador industrial. Si a esto se agrega la falta de información acerca del funcionamiento del controlador, se puede afirmar que no es tan fácil la desconexión de la tarjeta CB. Por esta razón se midieron y analizaron las señales de los conectores de la tarjeta CB con el fin de sólo dejar conectadas las señales necesarias para que el

TESIS CON
FALLA DE ORIGEN

controlador C500 sirva al objetivo planteado y el PID de fábrica quede totalmente inhabilitado. Para la medición de las señales se fabricaron 2 buses de prueba, el bus BP-50 y el BP-60 de 50 y 60 vías respectivamente. El diagrama eléctrico de los buses es mostrado en la Figura 3.10, mientras que la Figura 3.11 muestra un esquema del bus BP-60.

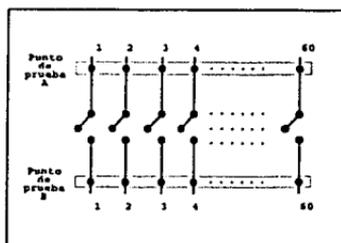


Figura 3.10: Diagrama eléctrico de los buses BP-50 y BP-60.

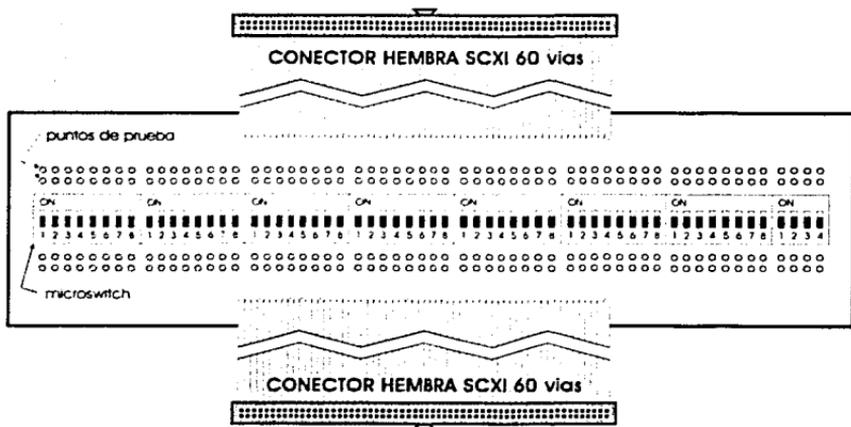


Figura 3.11: Esquema del bus BP-60.

Los buses BP-50 y BP-60 auxiliaron en las mediciones en tres aspectos importantes: a) en poder medir de forma rápida y fácil la magnitud de las señales en los conectores sin tener que desconectarlos, b) en saber la dirección de flujo de la señal y,

TESIS CON
FALLA DE ORIGEN

c) en poder desconectar una por una las vías del bus. Basándose en los manuales de servicio del controlador C500 y los resultados obtenidos de las mediciones se obtuvo información sobre los conectores de la tarjeta CB mostrada en la Figura 3.8.

- El conector *J1* de la tarjeta CB interconecta a esta con la tarjeta AB. Las señales que en este bus se presentan son necesarias para que el controlador encienda y reciba las señales de paradas de emergencia de dispositivos externos, por ejemplo del *teach pendal*.
- El conector *J6* recibe las señales de identificación de los switches de límite y de Home, además envía los comandos de control a los amplificadores a través de la tarjeta AB. Es por aquí donde se comandarán las señales de salida del controlador propio. El conector *J6* se comunica con la tarjeta AB a través de un bus de 60 pines que se conecta al conector *J1* de la tarjeta auxiliar. Las vías 41, 42, 54 y 55 de dicho bus envían señales de verificación de buena conexión entre las tarjetas CB y AB y es necesario que permanezcan conectadas para que el controlador C500 se active. Por lo que respecta a las otras vías, la Tabla 3.6 muestra una descripción de los pines necesarios para el sistema adaptado. La asignación de pines del conector *J6* se describe en la Tabla A.2 del Apéndice A.

Pines	Descripción
1 - 5, 7	Comandos de voltaje para los ejes 1-6 { $\pm 10V$ continuo}
6, 8	Comandos de voltaje para ejes opcionales 7 y 8 { $\pm 10V$ continuo}
9 - 32	Señal de identificación de los switches de home, switches de límite de viaje positivo, switches de límite de viaje negativo y switches térmicos para los ejes 1 - 6 {0 - 5V binario}
33 - 40	Señal de identificación de los: switches de home, switches de límite de viaje positivo, switches de límite de viaje negativo y switches térmicos para los ejes opcionales 7 y 8 {0 - 5V binario}
41 - 42	Envían señales de verificación de buena conexión entre las tarjetas AB y CB {0 - 5V binario}, es necesario que permanezcan conectados estos pines entre el conector CB-J6 y AB-J1
43 - 53	No es necesario que sean conectados para el sistema adaptado
54 - 55	Envían señales de verificación de buena conexión entre las tarjetas AB y CB {0 - 5V binario}, es necesario que permanezcan conectados estos pines entre el conector CB-J6 y AB-J1
56 - 60	No es necesario que sean conectados para el sistema adaptado

Tabla 3.6: Distribución de los pines en el conector *J6* del controlador C500 y su importancia en el sistema adaptado.

- El conector *J7* recibe las señales de encoder de las articulaciones del robot que proporciona la tarjeta RF-J4 del panel trasero. Al mismo tiempo, los pines 47, 48, 49 y 50 proporcionan la energía para los switches de home y los switches de límite. Por tal motivo es necesario que no sean desconectados si se desea la identificación de dichos switches. La Tabla 3.7 muestra una descripción de los pines necesarios para el sistema adaptado. La asignación de pines del conector *J7* se describe en la Tabla A.4 del Apéndice A.

Pines	Descripción
1 - 6	Señales A, A, B, B, Z, Z del encoder del eje 1 {0 - 5V binario}
7 - 12	Señales A, A, B, B, Z, Z del encoder del eje 2 {0 - 5V binario}
13 - 18	Señales A, A, B, B, Z, Z del encoder del eje 3 {0 - 5V binario}
19 - 24	Señales A, A, B, B, Z, Z del encoder del eje 4 {0 - 5V binario}
25 - 30	Señales A, A, B, B, Z, Z del encoder del eje 5 {0 - 5V binario}
31 - 36	Señales A, A, B, B, Z, Z del encoder del eje 6 {0 - 5V binario}
37 - 46	No es necesario que sean conectados para el sistema adaptado
47 - 50	Energía para el servo gripper, los switches de home, switches de límite de viaje positivo y switches de límite de viaje negativo, es necesario que permanezcan conectados estos pines entre el conector CB-J7 y RF-J4 {20V CD}

Tabla 3.7: Distribución de los pines en el conector *J7* del controlador C500 y su importancia en el sistema adaptado.

- El conector *J8* suministra la energía necesaria para que los encoders funcionen. Por tal motivo es indispensable que sea conectado.
- Los demás conectores listados en la Tabla 3.2 son utilizados para las comunicaciones entre el C500 y una PC o un *teach pendal*, y la alimentación de la misma tarjeta. El hecho de que estos conectores estén conectados no afecta en el sistema adaptado.

Del análisis anterior, se observa que los conectores *J6* y *J7* son de interés por contener las señales de control y de retroalimentación, para controlar el robot, respectivamente. La Figura 3.12 muestra un esquema de estos dos conectores. En las Tablas A.2 y A.4 se muestra la asignación de pines de estos dos conectores, para conocer la asignación de los demás conectores se puede consultar CRS (1995).

TESIS CON
FALLA DE ORIGEN

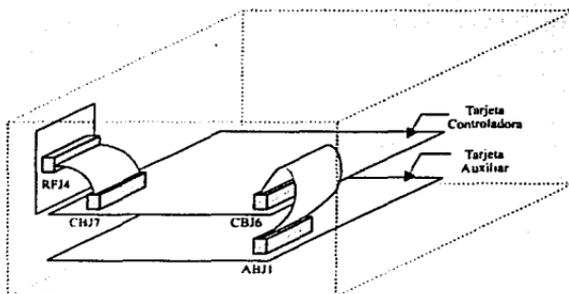
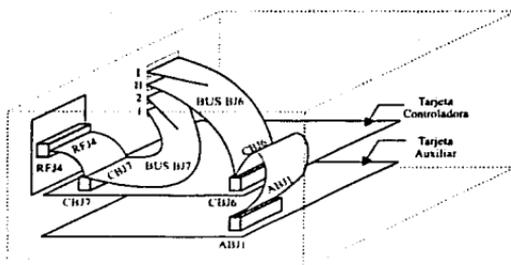


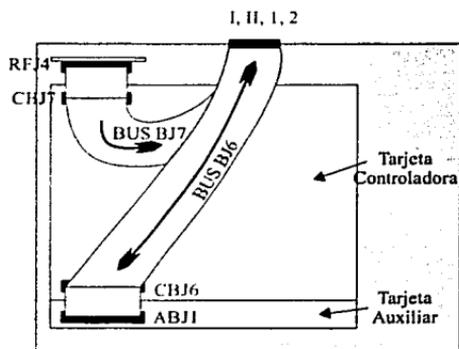
Figura 3.12: Disposición original de los conectores J6 y J7 del controlador C500.

3.2.1 Creación de buses internos

Ya identificadas las señales necesarias para el sistema adaptado, se procede a la creación de buses internos que deshabiliten las señales innecesarias y permitan el acceso a las señales "útiles" desde el exterior de la consola del C500 para su interconexión con la tarjeta de adquisición de datos. Para tener acceso a las señales seleccionadas del controlador C500, se fabricaron 2 buses llamados BJ6 y BJ7, los cuales tienen 2 conectores DB-25 cada uno, estos conectores se colocaron en la parte trasera del controlador C500. En el bus BJ6 están las señales de control hacia el robot y la identificación de los switches de límite y home, mientras que en el bus BJ7 están las señales de los encoders de los actuadores del robot. Para su identificación, a los conectores del bus BJ6 se les designó como C500DB25[I] y C500DB25[II], y a los conectores del bus BJ7 como C500DB25[1] y C500DB25[2]. La Figura 3.13 muestra la disposición de los buses y las Tablas D.1, D.2, D.3 y D.4 del Apéndice D enlistan la asignación de pines de los conectores C500DB25[I], C500DB25[II], C500DB25[1] y C500DB25[2] respectivamente. Es importante resaltar que los cambios realizados en el interior del controlador C500 no son significativos desde nivel hardware debido a un correcto análisis en el funcionamiento del mismo, esto denota una ventaja al poder regresar el controlador C500 a su estado original solamente regresando a su posición los buses de fábrica.



Vista isométrico



Vista superior

Figura 3.13: Disposición de los cables del controlador C500 adaptado.

3.3 Descripción de la tarjeta de adquisición de datos

Ahora que se ha aislado la etapa de control del sistema original y se han creado buses para tener fácil acceso a las señales necesarias para el control, hace falta una tarjeta que sea capaz de admitir las señales identificadas del controlador C500. Para este fin el laboratorio de robótica tiene dos tarjetas PCI-FlexMotion-6C de *National Instruments*. La tarjeta PCI-FlexMotion-6C es una combinación de controladores

para servomotores o motores de pasos para computadoras con bus PCI. Estos controladores proveen control completo de movimiento programable hasta para seis ejes de movimiento completamente independientes o simultáneamente coordinados. Además, tiene funciones de E/S dedicadas para detectar los switches de límite y home. También cuenta con funciones adicionales de E/S de propósito general. Se pueden usar estos controladores para toda aplicación de control de posición y velocidad. La Figura 3.14 muestra la localización de las diferentes partes de la tarjeta.

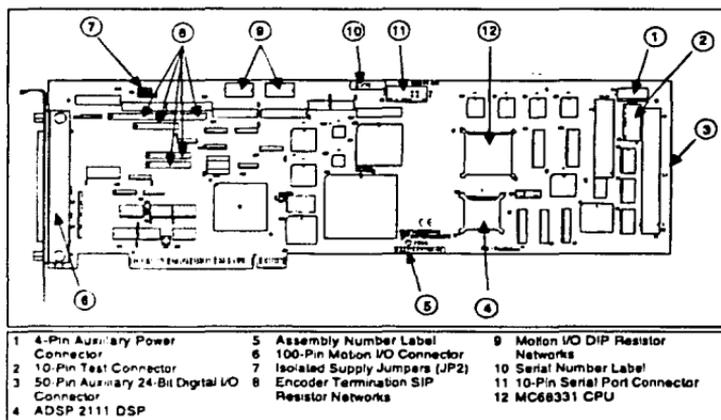


Figura 3.14: Localización de las partes de la tarjeta PCI-FlexMotion 6C

Los servoejes de la tarjeta pueden controlar servomotores, servovalvulas y otros servo dispositivos. Es importante mencionar que los servoejes **siempre operan en modo a lazo cerrado**. Estos ejes usan encoders en cuadratura o entradas analógicas para la retroalimentación de posición y velocidad y proveen comandos analógicos de salida con un rango industrial-estándar de ± 10 V. A grandes rasgos, un servoeje consiste de un generador de trayectoria, un controlador PID, un recurso de salida (DAC's) y un recurso de retroalimentación (encoders o un ADC's). La configuración de un servoeje se muestra en la Figura 3.15. En su configuración por default, la FlexMotion viene preconfigurada como seis servoejes con el encoder 1 y el DAC 1 mapeados para el eje 1; el encoder 2 y el DAC 2 mapeados para el eje 2 y así hasta el eje 6. Sin embargo, es simple para usuarios avanzados mapear cualquier recurso de retroalimentación y cualquier recurso de salida para cualquier

TESIS CON
FALLA DE ORIGEN

eje. Esta flexibilidad permite adaptar cada eje para acomodarlos a los requerimientos específicos del sistema de movimiento.

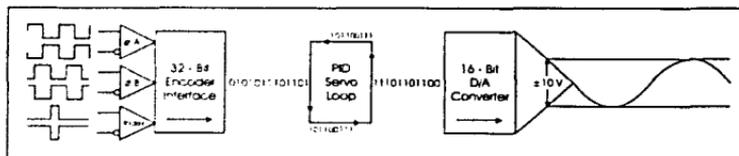


Figura 3.15: Recursos de un servoeje en la tarjeta PCI-FlexMotion-6C.

Los DAC's y ADC's que no son usados por un eje están disponibles para su uso en aplicaciones específicas que no requieren del servoeje. Por ejemplo, se puede controlar directamente un DAC, que no está siendo mapeado como una salida analógica, como una salida analógica de propósito general ($\pm 10V$). Así mismo, se puede usar cualquier canal del ADC para medir potenciómetros u otro sensor analógico. Es importante mencionar que, una vez mapeado para un eje, todas las características y funciones de un recurso de entrada o salida están disponibles como parte del eje. En el caso de que algún recurso utilizado por un eje de la tarjeta se desee emplear para alguna otra aplicación, siempre tendrá la prioridad el servoeje. Por lo tanto, el PID interno de la tarjeta rige los recursos de entrada y salida. Por esta razón sólo se pueden utilizar $6 - E$ recursos de entrada o salida independientes, donde E es el número de servoejes habilitados. Esto es un punto clave en el funcionamiento de la tarjeta porque las señales de encoder sólo pueden ser leídas cuando los servoejes están habilitados. Por ejemplo, si se requiere leer 6 entradas de encoder ($E = 6$) no queda ningún recurso de salida posible para utilizar ($6 - E = 0$). Si se divide los recursos de E/S se tiene que una tarjeta puede leer 3 entradas de encoder y comandar señales de control por 3 DAC's simultáneamente. De este modo, sólo se puede controlar 3 ejes completos de movimiento por tarjeta en el sistema adaptado.

Las conexiones externas de la tarjeta PCI-FlexMotion-6C (ver Figura 3.14) constan de dos conectores que se encargan de todas las señales hacia y del sistema externo de movimiento. Estos conectores son:

- Un conector de 100 pines de E/S de movimiento: este conector contiene todas las señales requeridas para controlar hasta seis ejes en lazo cerrado incluyendo salidas de comandos analógicos hacia el servomotor, entradas de retroalimentación por encoder, entradas de identificación de los switches de límite de viaje negativo y positivo, entradas de identificación de los switches de home, entradas de disparo. Este conector también contiene ocho canales de

entrada Analógico-Digital (A/D) de 12 bits para retroalimentación analógica o entradas analógicas de propósito general. La Figura B.1 en el Apéndice B muestra la asignación de pines para el conector de E/S de movimiento en la tarjeta PCI-FlexMotion-6C.

- Un conector auxiliar de 50 pines de E/S digital: este conector se encuentra en la parte posterior de la tarjeta. Es un puerto digital de 24 bits y en el hardware está configurado como tres puertos de 8 bits cada uno. La asignación de pines de este conector se expone en el Apéndice B.

La tarjeta FlexMotion se manipula usando alguno de los siguientes softwares: LabVIEW™, LabWindows/CVI™, BridgeVIEW™ o el programa FlexMotion™ siempre que incluya FlexCommander. En el Apéndice B se muestran más características de la tarjeta PCI-FlexMotion-6C.

Después de haber descrito a grandes rasgos las características de la tarjeta FlexMotion, es importante destacar que la tarjeta tiene un controlador interno PID que condiciona el número de puertos de entrada y salida disponibles para el sistema adaptado. Este inconveniente se puede resolver cambiando la tarjeta de adquisición de datos, sin necesidad de cambiar el resto del sistema a excepción del alambrado de la tarjeta y el direccionamiento de los puertos de E/S en los controladores programados. En el momento de escribir la presente tesis no se ha encontrado alguna tarjeta de adquisición de datos que acepte todos las señales necesarias para el sistema adaptado. Un método alternativo para contrarrestar la limitación de la tarjeta está descrito en la Sección 3.6.

3.4 Accesorios para la adaptación

Ahora es necesario acoplar los dos sistemas independientes: el controlador C500 adaptado y la tarjeta de adquisición de datos. Para este fin existen accesorios de conectividad en el mercado. Además es necesaria la creación de algunos accesorios más especializados para hacer el acople más sencillo.

3.4.1 UMI-Flex6

Como se ve de la descripción de la tarjeta, son demasiadas las señales que deben ser alambradas en la tarjeta PCI-FlexMotion-6C; para facilitar esta labor *National Instruments* ofrece una variedad de productos para usar con las tarjetas FlexMotion, incluyendo cables e Interfaces de Movimiento Universal -UMI- (por sus siglas en inglés). La UMI-Flex6 es un accesorio de conectividad que conecta las fuentes de poder, los servo amplificadores, motores, encoders y los switches de límite y home a

la tarjeta de control PCI-FlexMotion-6C. La UMI simplifica el alambrado con bloques de terminales separados por ejes (encoders, switches, amplificadores). Cuenta además con un bus de monitoreo que automáticamente deshabilita la salida a los amplificadores si la computadora maestra es apagada o el cable interfaz se desconecta.

El laboratorio de robótica posee dos UMI-Flex6, una para cada tarjeta, con su respectivo cable de interfaz para las tarjetas FlexMotion. La Figura 3.16 muestra un esquema de las partes de la UMI. En el Apéndice C se listan las especificaciones de esta interfaz.

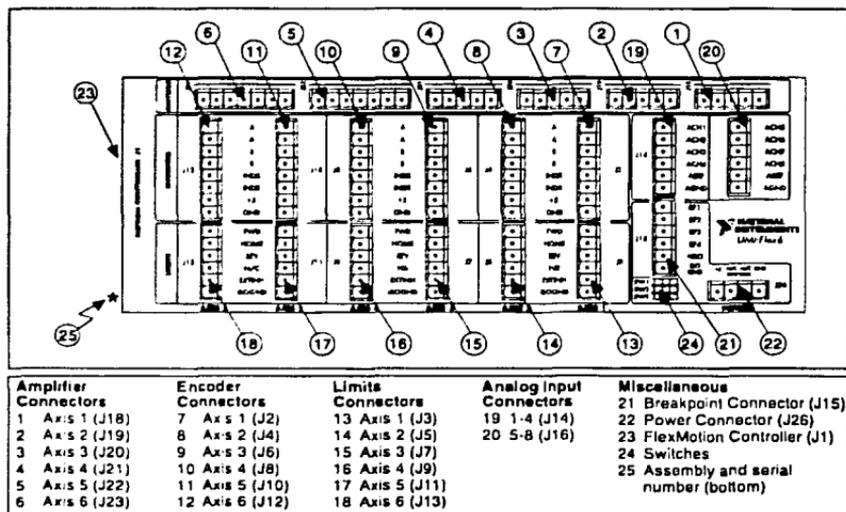


Figura 3.16: Diagrama de partes de la UMI.

3.4.2 Chasis UMI

Por cuestiones de robustez en el manejo, se crearon dos chasis idénticos para encapsular cada una de las UMIFlex6. Este chasis llamado "Chasis UMI" tiene 7 conectores en los cuales se organizan y separan las señales de la UMI. La Figura

TESIS CON
FALLA DE ORIGEN

3.17 muestra el Chasis UMI, y la Tabla 3.8 describe la función de cada conector.

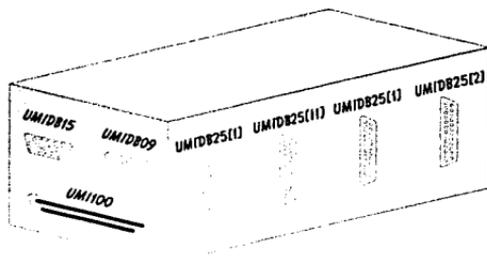


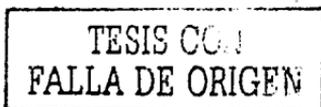
Figura 3.17: Chasis UMI

Conector	Descripción
UMIDB15	Conector tipo DB de 15 pines que contiene: - Comandos de voltaje de los ejes 1 - 6 de la tarjeta PCI-FlexMotion-6C - Terminales de entrada al amplificador de las articulaciones 1 - 6 del robot
UMIDB09	Conector tipo DB de 9 pines que contiene las entradas de los ADC's
UMIDB25_1 y 2;	Conector tipo DB de 25 pines que contiene las entradas tipo DB de las señales de encoder
UMIDB25_1 y 11}	Conector tipo DB de 25 pines que contiene: - Las entradas de identificación de los switches de límite y los switches de home - Salida de comandos de voltaje analógico hacia los amplificadores del robot
UM100	Conector tipo SCXI de 100 pines que conecta la tarjeta PCI-FlexMotion-6C con la UMIFlex6

Tabla 3.8: Función de cada conector del Chasis UMI.

Las Tablas D.5, D.6, D.7, D.8, D.9 y D.10 en el Apéndice D muestran la asignación de pines de los conectores UMIDB15, UMIDB09, UMIDB25[1], UMIDB25[2], UMIDB25[1] y UMIDB25[11] respectivamente.

Para clarificar la función y utilidad de los conectores, la Figura 3.18 muestra el



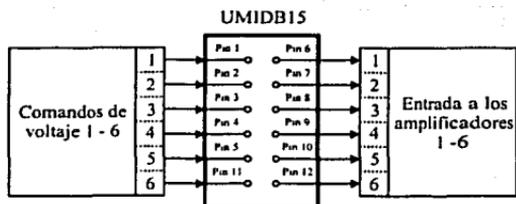


Figura 3.19: Diagrama del conector UMIDB15.

3.5 Acople de señales y el sistema adaptado

Los subsistemas están listos para ser acoplados; solo se necesita conocer la correspondencia que existen entre las señales del controlador C500 adaptado y el Chasis UMI. La Tabla 3.9 muestra dicha correspondencia. La forma en la que se agruparon las señales en el Chasis UMI permite que las interfaces externas que conectan los dos subsistemas sean simples cables de 25 vías con conectores DB25 macho en los extremos con correspondencia uno a uno. Por convención se asignó el número "1" a todos los accesorios del sistema adaptado para el robot A465 (tarjeta FlexMotion 1, Chasis UMI 1, Módulo C500 1, etc.); análogamente se asignó el número "2" a todos los accesorios del sistema adaptado para el robot A255.

Es relevante mencionar que todas las señales de encoder de las articulaciones del robot están alambreadas a su respectivo eje en la tarjeta y, como se mencionó en la Sección 3.3, es necesario que se dividan los recursos de E/S de la tarjeta. Por este motivo una tarjeta sólo puede controlar tres articulaciones del robot, de tal forma que se optó por configurar los ejes 1, 2 y 3 de la tarjeta para leer las señales de encoder, y los ejes 4, 5 y 6 como salidas de comandos de voltaje que serán enrutadas, con la ayuda del conector UMIDB15, hacia las articulaciones 1, 2 y 3 del robot, respectivamente. Las Tablas 3.10 y 3.11 muestran la configuración correspondiente para las tarjetas de adquisición de datos y el conector UMIDB25.

De forma gráfica, la configuración del conector UMIDB15 se muestra en la Figura 3.20(a). La Figura 3.20(b) muestra el conector llave que configura el conector UMIDB15.



Chasis UMI	C500	Chasis UMI	C500	Chasis UMI	C500	Chasis UMI	C500
Phase 1A	A1+	Phase 1 \bar{A}	A1-	Forward limit 1	PSW1	Reverse limit 1	NSW1
Phase 2A	A2+	Phase 2 \bar{A}	A2-	Forward limit 2	PSW2	Reverse limit 2	NSW2
Phase 3A	A3+	Phase 3 \bar{A}	A3-	Forward limit 3	PSW3	Reverse limit 3	NSW3
Phase 4A	A4+	Phase 4 \bar{A}	A4-	Forward limit 4	PSW4	Reverse limit 4	NSW4
Phase 5A	A5+	Phase 5 \bar{A}	A5-	Forward limit 5	PSW5	Reverse limit 5	NSW5
Phase 6A	A6+	Phase 6 \bar{A}	A6-	Forward limit 6	PSW6	Reverse limit 6	NSW6
Phase 1B	B1-	Phase 1 \bar{B}	B1-	Analog Output 1	Vcom1	HOME 1	HSW1
Phase 2B	B2-	Phase 2 \bar{B}	B2-	Analog Output 2	Vcom2	HOME 2	HSW2
Phase 3B	B3-	Phase 3 \bar{B}	B3-	Analog Output 3	Vcom3	HOME 3	HSW3
Phase 4B	B4-	Phase 4 \bar{B}	B4-	Analog Output 4	Vcom4	HOME 4	HSW4
Phase 5B	B5-	Phase 5 \bar{B}	B5-	Analog Output 5	Vcom5	HOME 5	HSW5
Phase 6B	B6-	Phase 6 \bar{B}	B6-	Analog Output 6	Vcom6	HOME 6	HSW6
Index 1	Z1-	Index 4	Z4-	Index 1	Z1-	Index 4	Z4-
Index 2	Z2-	Index 5	Z5-	Index 2	Z2-	Index 5	Z5-
Index 3	Z3+	Index 6	Z6+	Index 3	Z3-	Index 6	Z6-

Tabla 3.9: Correspondencia de las señales entre el C500 y el Chasis UMI

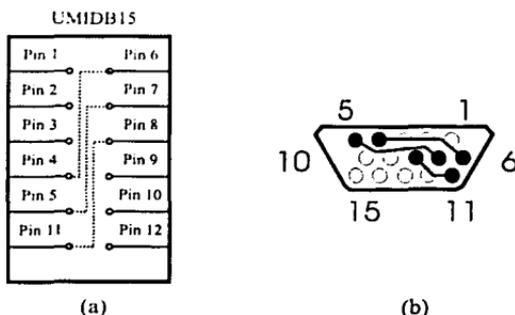


Figura 3.20: a) Conexión estándar de pines del conector UMIDB15 para controlar las tres primeras articulaciones del robot b) Conector llave (DB - 15 hembra) que configura el UMIDB15 de la forma estándar.

Con esto el sistema queda acoplado a nivel hardware para controlar 3 articulaciones de cada uno de los robots industriales. La Figura 3.21 muestra una representación del sistema adaptado para los robots A465 y A255.

TESIS CON
FALLA DE ORIGEN

Tarjeta FlexMotion 1			Chasis UMI 1	
Configuración de los servomotores de la tarjeta			Configuración del conector UMIDB15	
EJE	Estado del servomotor	Configurado para:	Unir pines	Configurado para:
1	Habilitado	Entrada de encoders de la articulación 1 del robot A465		---
2	Habilitado	Entrada de encoders de la articulación 2 del robot A465		---
3	Habilitado	Entrada de encoders de la articulación 3 del robot A465		---
4	Deshabilitado	Salida de comandos de voltaje	4 - 6	Enviar el comando de voltaje 4 de la tarjeta a la articulación 1 del robot
5	Deshabilitado	Salida de comandos de voltaje	5 - 7	Enviar el comando de voltaje 5 de la tarjeta a la articulación 2 del robot
6	Deshabilitado	Salida de comandos de voltaje	8 - 11	Enviar el comando de voltaje 6 de la tarjeta a la articulación 3 del robot

Tabla 3.10: Configuración de la tarjeta FlexMotion 1 y el conector UMIDB15 del Chasis UMI 1 para controlar las tres primeras articulaciones del robot A465.

Tarjeta FlexMotion 2			Chasis UMI 2	
Configuración de los servomotores de la tarjeta			Configuración del conector UMIDB15	
EJE	Estado del servomotor	Configurado para:	Unir pines	Configurado para:
1	Habilitado	Entrada de encoders de la articulación 1 del robot A255		---
2	Habilitado	Entrada de encoders de la articulación 2 del robot A255		---
3	Habilitado	Entrada de encoders de la articulación 3 del robot A255		---
4	Deshabilitado	Salida de comandos de voltaje	4 - 6	Enviar el comando de voltaje 4 de la tarjeta a la articulación 1 del robot
5	Deshabilitado	Salida de comandos de voltaje	5 - 7	Enviar el comando de voltaje 5 de la tarjeta a la articulación 2 del robot
6	Deshabilitado	Salida de comandos de voltaje	8 - 11	Enviar el comando de voltaje 6 de la tarjeta a la articulación 3 del robot

Tabla 3.11: Configuración de la tarjeta FlexMotion 2 y el conector UMIDB15 del Chasis UMI 2 para controlar las tres primeras articulaciones del robot A255.

TESIS CON
FALLA DE ORIGEN

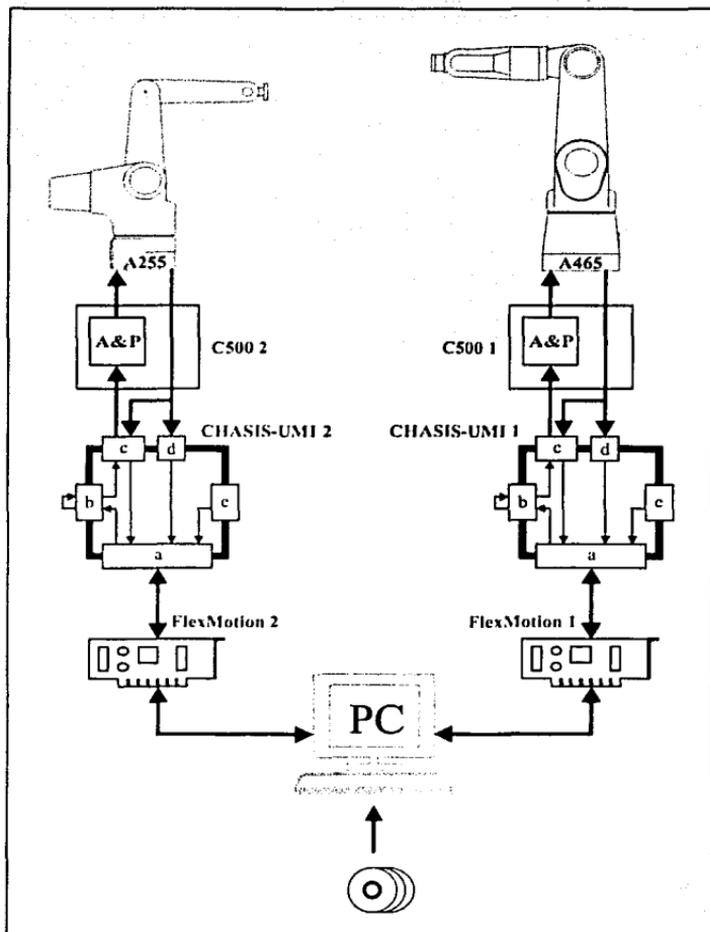


Figura 3.21: Esquema del sistema adaptado para los robots A255 y A465.

TESIS CON
FALLA DE ORIGEN

3.6 Solución para el problema del control de hasta 6 articulaciones

En la Sección 3.3 se mencionó que los encoders sólo pueden ser leídos por la tarjeta de adquisición de datos cuando los servo ejes están habilitados. Por lo tanto una tarjeta FlexMotion-6C sólo pueden utilizar $6 - E$ recursos de salida independientes del servoeje ($E =$ número de servo ejes habilitados en la tarjeta). Esto implica que, si se desea controlar de 4 hasta 6 ejes de movimiento en el sistema adaptado, se necesitan al menos dos tarjetas para hacerlo. El diseño del Chasis UMI facilita el uso de las dos tarjetas FlexMotion para controlar hasta seis ejes de movimiento para el sistema adaptado. Para este fin se creó la interfaz unidireccional de seis vías "UDI6" que interconecta las Chasis UMI y evita el re-alambramiento de las señales del sistema. La Figura 3.22 muestra un esquema de la interfaz UDI6.



Figura 3.22: interfaz unidireccional de 6 vías - UDI6

El flujo de datos en la interfaz UDI6 es del conector A hacia el conector B. Recordando que las señales de encoders están conectadas con su correspondiente eje en la tarjeta FlexMotion, se configura una de las tarjetas como entrada de encoders y la otra como salida de comandos de voltaje. La tarjeta que recibe las señales de encoder es siempre la tarjeta FlexMotion 1 para el robot A465 y la tarjeta FlexMotion 2 para el robot A255. De este modo, dependiendo del robot que se desee controlar se deben configurar las tarjetas adecuadamente.

Por lo tanto, el conector A se enchufa al conector UMIDB15 del Chasis UMI que está configurado para la salida de los comandos de voltaje y el conector B se enchufa al conector UMIDB15 del Chasis UMI que recibirá las señales de encoder. La configuración por default de las conexiones entre el conectores A y B se muestra en la Tabla 3.12.

El módulo de conexiones de la interfaz UDI6 permite hacerla más versátil. Por medio de este módulo las señales se pueden enrutar de diferentes maneras para varias aplicaciones. La disposición mostrada en la Tabla 3.12 se cumple si y sólo si

CONECTOR				
A			B	
PIN	DESCRIPCIÓN		DESCRIPCIÓN	PIN
6	Señal "Analog Output1"	→	EJE 1 (Motor 1)	1
7	Señal "Analog Output2"	→	EJE 2 (Motor 2)	2
8	Señal "Analog Output3"	→	EJE 3 (Motor 3)	3
9	Señal "Analog Output4"	→	EJE 4 (Motor 4)	4
10	Señal "Analog Output5"	→	EJE 5 (Motor 5)	5
12	Señal "Analog Output6"	→	EJE 6 (Motor 6)	11
15	GND	→	GND	15

Tabla 3.12: Correspondencia entre los pines de los conectores A y B de la Interfaz Unidireccional de 6 vías.

los jumpers en el bloque de conexiones están conectados como se ve en la Figura 3.23.

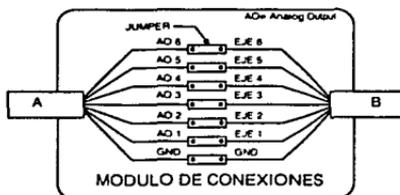


Figura 3.23: Módulo de conexiones de la interfaz unidireccional - UDI6

3.6.1 Configuración para controlar 6 ejes del robot A465

Ahora se mostrará la configuración necesaria en las tarjetas de adquisición de datos y los conectores UMIDB15 para controlar los seis ejes del robot A465. Las Tablas 3.13 y 3.14 muestran la configuración de las tarjetas FlexMotion y los Chasis UMI para controlar el robot A465. La Figura 3.24 muestra el esquema del sistema con la interfaz UDI6 para el robot A465, mientras que la Figura 3.25 muestra las conexiones entre los conectores UMIDB15 de los Chasis UMI.

TESIS CON
FALLA DE ORIGEN

Tarjeta FlexMotion 1			Chasis UMI 1	
Configuración de los servomotores de la tarjeta			Configuración del conector UMIDB15	
EJE	Estado del servomotor	Configurado para:	Unir pines	Configurado para:
1	Habilitado	Entrada de encoders de la articulación 1 del robot A465	UD16	Recibir el comando de voltaje 1 proveniente del CHASIS UMI 2 a la articulación 1 del robot
2	Habilitado	Entrada de encoders de la articulación 2 del robot A465	UD16	Recibir el comando de voltaje 2 proveniente del CHASIS UMI 2 a la articulación 2 del robot
3	Habilitado	Entrada de encoders de la articulación 3 del robot A465	UD16	Recibir el comando de voltaje 3 proveniente del CHASIS UMI 2 a la articulación 3 del robot
4	Habilitado	Entrada de encoders de la articulación 4 del robot A465	UD16	Recibir el comando de voltaje 4 proveniente del CHASIS UMI 2 a la articulación 4 del robot
5	Habilitado	Entrada de encoders de la articulación 5 del robot A465	UD16	Recibir el comando de voltaje 5 proveniente del CHASIS UMI 2 a la articulación 5 del robot
6	Habilitado	Entrada de encoders de la articulación 6 del robot A465	UD16	Recibir el comando de voltaje 6 proveniente del CHASIS UMI 2 a la articulación 6 del robot

Tabla 3.13: Configuración de la tarjeta FlexMotion 1 y el conector UMIDB15 del Chasis UMI 1 para controlar las seis articulaciones del robot A465.

Tarjeta FlexMotion 2			Chasis UMI 2	
Configuración de los servomotores de la tarjeta			Configuración del conector UMIDB15	
EJE	Estado del servomotor	Configurado para	Unir pines	Configurado para:
1	Deshabilitado	Salida de comandos de voltaje	UD16	Enviar el comando de voltaje 1 de la tarjeta al CHASIS UMI 1
2	Deshabilitado	Salida de comandos de voltaje	UD16	Enviar el comando de voltaje 2 de la tarjeta al CHASIS UMI 1
3	Deshabilitado	Salida de comandos de voltaje	UD16	Enviar el comando de voltaje 3 de la tarjeta al CHASIS UMI 1
4	Deshabilitado	Salida de comandos de voltaje	UD16	Enviar el comando de voltaje 4 de la tarjeta al CHASIS UMI 1
5	Deshabilitado	Salida de comandos de voltaje	UD16	Enviar el comando de voltaje 5 de la tarjeta al CHASIS UMI 1
6	Deshabilitado	Salida de comandos de voltaje	UD16	Enviar el comando de voltaje 6 de la tarjeta al CHASIS UMI 1

Tabla 3.14: Configuración de la tarjeta FlexMotion 2 y el conector UMIDB15 del Chasis UMI 2 para controlar las seis articulaciones del robot A465.

TESIS CON
FALLA DE ORIGEN

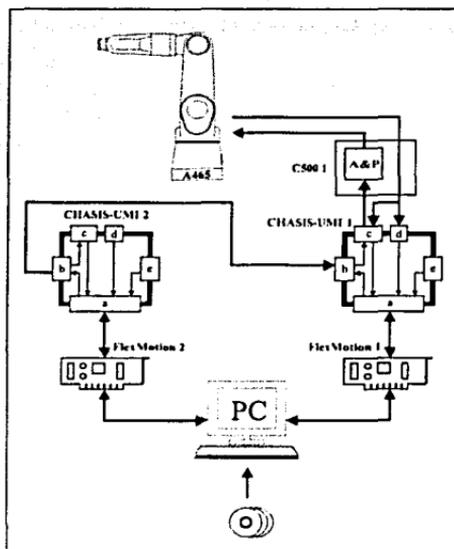


Figura 3.24: Esquema del sistema adaptado para controlar las seis articulaciones del robot A465.

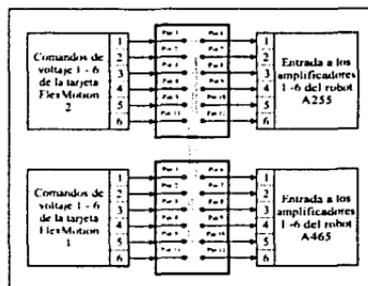


Figura 3.25: Conexiones entre los conectores UMIDB15 de los Chasis UMI cuando se utiliza la interfaz UD16 para el robot A465.

TESIS CON
FALLA DE ORIGEN

3.6.2 Configuración para controlar 5 ejes del robot A255

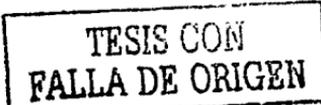
Análogamente al Apartado 3.6.1 se mostrará la configuración necesaria en las tarjetas de adquisición y los conectores UMIDB15 para controlar los cinco ejes del robot A255. Las Tablas 3.15 y 3.16 muestran la configuración de las tarjetas FlexMotion y los Chasis UMI para controlar el robot A255. La Figura 3.26 muestra el esquema del sistema con la interfaz UDI6 para el robot A255. La Figura 3.27 muestra las conexiones entre los conectores UMIDB15 de los Chasis UMI.

Tarjeta FlexMotion 1			Chasis UMI 1	
Configuración de los servos de la tarjeta			Configuración del conector UMIDB15	
EJE	Estado del servos	Configurado para:	Unir pines	Configurado para:
1	Deshabilitado	Salida de comandos de voltaje	UDI6	Enviar el comando de voltaje 1 de la tarjeta al CHASIS UMI 2
2	Deshabilitado	Salida de comandos de voltaje	UDI6	Enviar el comando de voltaje 2 de la tarjeta al CHASIS UMI 2
3	Deshabilitado	Salida de comandos de voltaje	UDI6	Enviar el comando de voltaje 3 de la tarjeta al CHASIS UMI 2
4	Deshabilitado	Salida de comandos de voltaje	UDI6	Enviar el comando de voltaje 4 de la tarjeta al CHASIS UMI 2
5	Deshabilitado	Salida de comandos de voltaje	UDI6	Enviar el comando de voltaje 5 de la tarjeta al CHASIS UMI 2
6	Deshabilitado	Salida de comandos de voltaje	UDI6	Enviar el comando de voltaje 6 de la tarjeta al CHASIS UMI 2

Tabla 3.15: Configuración de la tarjeta FlexMotion 1 y el conector UMIDB15 del Chasis UMI 1 para controlar las cinco articulaciones del robot A255.

Tarjeta FlexMotion 2			Chasis UMI 2	
Configuración de los servos de la tarjeta			Configuración del conector UMIDB15	
EJE	Estado del servos	Configurado para:	Unir pines	Configurado para:
1	Habilitado	Entrada de encoders de la articulación 1 del robot A255	UDI6	Recibir el comando de voltaje 1 proveniente del CHASIS UMI 1 a la articulación 1 del robot
2	Habilitado	Entrada de encoders de la articulación 2 del robot A255	UDI6	Recibir el comando de voltaje 2 proveniente del CHASIS UMI 1 a la articulación 2 del robot
3	Habilitado	Entrada de encoders de la articulación 3 del robot A255	UDI6	Recibir el comando de voltaje 3 proveniente del CHASIS UMI 1 a la articulación 3 del robot
4	Habilitado	Entrada de encoders de la articulación 3 del robot A255	UDI6	Recibir el comando de voltaje 4 proveniente del CHASIS UMI 1 a la articulación 4 del robot
5	Habilitado	Entrada de encoders de la articulación 3 del robot A255	UDI6	Recibir el comando de voltaje 5 proveniente del CHASIS UMI 1 a la articulación 5 del robot
6	No importa	---	UDI6	---

Tabla 3.16: Configuración de la tarjeta FlexMotion 2 y el conector UMIDB15 del Chasis UMI 2 para controlar las cinco articulaciones del robot A255.



 TESIS CON
 FALLA DE ORIGEN

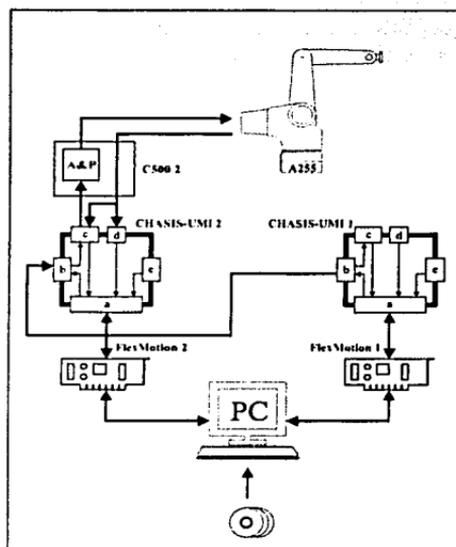


Figura 3.26: Esquema del sistema adaptado para controlar las cinco articulaciones del robot A255.

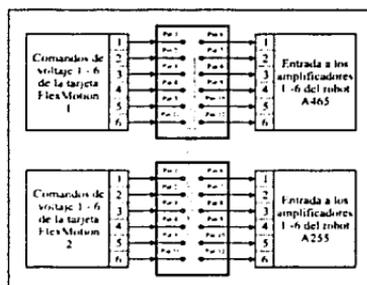


Figura 3.27: Conexiones entre los conectores UMIDB15 de los Chasis UMI cuando se utiliza la interfaz UD16 para el robot A255.

TESIS CON
FALLA DE ORIGEN

Con esto queda completamente adaptado el sistema robótico experimental a nivel hardware. En el Capítulo 4 se validará el nuevo sistema programando un algoritmo de control propio.

Capítulo 4

Validación y resultados experimentales

En este capítulo se presenta la validación del sistema adaptado descrito en el Capítulo 3. La validación consistió en programar un algoritmo de control para el nuevo sistema con los robots A465 y A255 trabajando simultáneamente.

4.1 Lenguaje de programación y programa básico de E/S

Después de acoplar el sistema a nivel hardware, es necesario seleccionar el software con el cual se programen los algoritmos de control. Como se vio en la Sección 2.3, se necesita de un software compatible con el hardware que admita programación estructurada basada en un lenguaje abierto, accesible, popular y versátil. Por lo tanto la selección del software está restringida a uno que pueda manejar la tarjeta de adquisición de datos. Se eligió el software LabWindows/CVI debido a la versatilidad y sencillez que ofrece el trabajar en lenguaje C.

4.1.1 LabWindows/CVI y Software FlexMotion

LabWindows/CVI es un ambiente de desarrollo de software para programadores de C que se puede utilizar para: a) desarrollar programas interactivamente, b) acceder a poderosas librerías de funciones para crear aplicaciones de control de instrumentos y adquisición de datos y c) tomar ventaja de un amplio conjunto de herramientas para adquisición, análisis y presentación de datos.

Los programas escritos en el ambiente interactivo de LabWindows/CVI deben respetar las especificaciones del ANSI C. Además, se pueden utilizar libremente los módulos de objeto compilados de C, las librerías de enlace dinámico (DLLs), las librerías de C, y drivers de instrumentos conjuntamente con archivos fuente del ANSI C cuando se desarrollan los programas.

El poder de LabWindows/CVI radica en sus librerías. Estas tienen funciones para desarrollar todas las fases del sistema de adquisición de datos y de control de instrumentos. Debido a que LabWindows/CVI es flexible, se puede hacer casi cualquier planteamiento para construir un programa. La mayoría de los proyectos que se construyen incluyen probablemente los elementos siguientes: una interfaz de usuario, un programa de control, la adquisición de datos y el análisis de datos. La Figura 4.1 muestra de forma general los elementos del programa. El programa de control recibe la entrada de la interfaz de usuario, de los elementos de adquisición y de los análisis de datos.

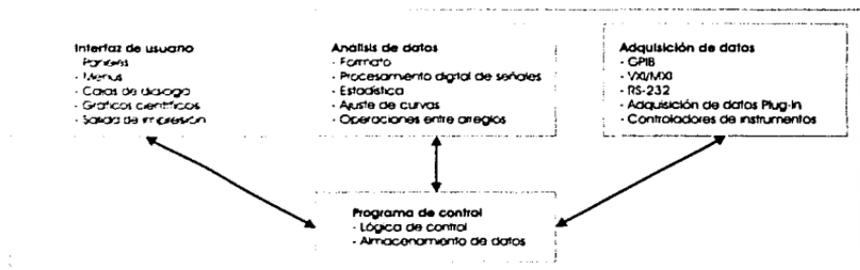


Figura 4.1: Relación entre los elementos de un programa en LabWindows/CVI.

Por otra parte, el software FlexMotion es una poderosa interfaz de programación entre la aplicación de control de movimiento y las tarjetas FlexMotion de *National Instruments* para computadoras con buses ISA y PCI. El software FlexMotion, combinado con las tarjetas correspondientes, proporciona funcionalidad y poder para los sistemas integrados de movimiento para su uso en laboratorios y ambientes de producción.

El software incluye una utilidad de configuración para la tarjeta de movimiento. Esta utilidad asocia direcciones físicas del bus con las identificaciones de la tarjeta, la cual es usada en programas para distinguir entre las tarjetas. También es utilizado para verificar que la tarjeta FlexMotion esté instalada correctamente y si se está comunicando con la computadora maestra. Las funciones que constituyen la librería

TESIS CON
FALLA DE ORIGEN

del software FlexMotion se pueden llamar desde LabWindows/CVI.

Para aprender más acerca de las librerías de LabWindows/CVI, drivers para instrumentos, adquisición de datos, la utilización de librerías de ANSI C y el software FlexMotion consulte Khalid (2000), National Instruments (1998) y National Instruments (1999a).

4.1.2 Programa básico de entrada salida

Para verificar la conexión del hardware del sistema adaptado se creó un programa de E/S básico que permite comandar salidas de voltaje por los DAC's de la tarjetas FlexMotion y que lee las señales de encoder de los robots. El diagrama de flujo del programa se muestra en la Figura 4.2

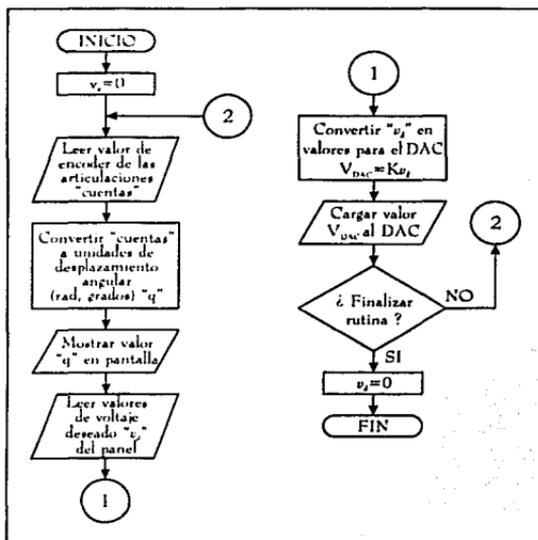


Figura 4.2: Programa básico de entrada / salida.

A continuación se describen de forma breve las instrucciones básicas para configurar la tarjeta y desarrollar el programa de E/S. En el disco de $3\frac{1}{2}$ " adjunto están los programas fuente desarrollados en el software LabWindows/CVI.

TESIS CON
FALLA DE ORIGEN

Configuración de la tarjeta

Las instrucciones básicas para configurar e inicializar la tarjeta FlexMotion se listan a continuación. Para más información acerca de la configuración consulte National Instruments (1999b). En el programa "INICIALIZACION" del disco de 3 $\frac{1}{2}$ " se encuentra la rutina de configuración para las tarjetas de acuerdo a la Tabla 4.1.

Tarjeta FlexMotion 1			Tarjeta FlexMotion 2	
Configuración de los servoejes de la tarjeta			Configuración de los servoejes de la tarjeta	
EJE	Estado del servoeje	Configurado para:	Estado del servoeje	Configurado para:
1	Habilitado	Entrada de encoders de la articulación 1 del robot A465	Habilitado	Entrada de encoders de la articulación 1 del robot A255
2	Habilitado	Entrada de encoders de la articulación 2 del robot A465	Habilitado	Entrada de encoders de la articulación 2 del robot A255
3	Habilitado	Entrada de encoders de la articulación 3 del robot A465	Habilitado	Entrada de encoders de la articulación 3 del robot A255
4	Deshabilitado	Salida de comandos de voltaje	Deshabilitado	Salida de comandos de voltaje
5	Deshabilitado	Salida de comandos de voltaje	Deshabilitado	Salida de comandos de voltaje
6	Deshabilitado	Salida de comandos de voltaje	Deshabilitado	Salida de comandos de voltaje

Tabla 4.1: Configuración de las tarjetas FlexMotion 1 y FlexMotion 2 para controlar las tres primeras articulaciones de los robots A465 y A255.

Comando:	flex_config_axis	
Formato:	status = flex_config_axis (boardID, axis, primaryFeedback, secondaryFeedback, primaryOutput, secondaryOutput)	
Propósito:	Configura un eje definiendo sus recursos de retroalimentación y salida	
Nombre	Tipo	Descripción
boardID	u8	Asignado por la computadora, depende del slot donde esté instalada la tarjeta
axis	u8	Eje que será configurado
primaryFeedback	u8	Recurso de retroalimentación primario (encoder o ADC)
secondaryFeedback	u8	Recurso de retroalimentación secundario (encoder o ADC)
primaryOutput	u8	Recurso de salida primario (DAC)
secondaryOutput	u8	Recurso de salida secundario (DAC)

Comando:	flex_enable_axes	
Formato:	status = flex_enable_axes (boardID, reserved, PIDrate, axisMap)	
Propósito:	Habilita el eje a operar y define la velocidad de muestreo del PID	
Nombre	Tipo	Descripción
boardID	u8	Asignado por la computadora, depende del slot donde esté instalada la tarjeta
reserved	u8	No se usa
PIDrate	u8	Velocidad de muestreo del PID
axisMap	u8	Mapa de bits de los ejes a habilitar

TESIS CON
FALLA DE ORIGEN

Instrucciones de entrada/salida

A continuación se presentan las instrucciones básicas del software FlexMotion para leer las señales de encoder y comandar voltajes a los DAC's.

Comando: flex_load_dac
Formato: status = flex_load_dac (boardID, DAC, outputValue, inputVector)
Propósito: Carga un valor de salida a un DAC no-mapeado.

Nombre	Tipo	Descripción
boardID	us	Asignado por la computadora, depende del slot donde esté instalada la tarjeta
DAC	us	DAC que será controlado
outputValue	i16	Valor enviado al DAC
inputVector	us	Fuente de datos para esta función

Comando: flex_read_encoder_rtn
Formato: status = flex_read_encoder_rtn (boardID, axisOrEncoder, encoderCounts)
Propósito: Lee la posición de un encoder

Nombre	Tipo	Descripción
boardID	us	Asignado por la computadora, depende del slot donde esté instalada la tarjeta
axisOrEncoder	us	Eje o encoder que será leído
inputVector	us	Fuente de datos para esta función

Comando: flex_reset_encoder
Formato: status = flex_reset_encoder (boardID, encoder, position, inputVector)
Propósito: Reseta la posición de un encoder no-mapeado al valor especificado.

Nombre	Tipo	Descripción
boardID	us	Asignado por la computadora, depende del slot donde esté instalada la tarjeta
encoder	us	Encoder que será reseteado
position	i32	Valor de reset para el encoder
inputVector	us	Fuente de datos para esta función

4.2 Control PID programado

Recordando el objetivo para lo que se creó el sistema experimental se debe poder programar algoritmos de control para los robots. Para la validación del sistema se programó un controlador Proporcional - Integral - Derivativo (PID), debido a la sencillez y al hecho de que sólo se necesita comprobar que los robots se pueden controlar con algoritmos de *control propios*. La Figura 4.3 muestra un lazo de control en retroalimentación.

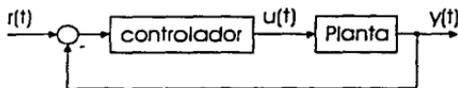


Figura 4.3: Lazo de control en retroalimentación.

TESIS CON
FALLA DE ORIGEN

Para este caso el bloque "controlador" del lazo de control es un control PID descrito por (4.1). La Figura 4.4 muestra el diagrama de un controlador PID analógico.

$$\begin{aligned} u(t) &= K_P \left[e(t) + \frac{1}{T_n} \int_0^t e(\tau) d\tau + T_v \dot{e}(t) \right] \\ &= K_P e(t) + K_I \int_0^t e(\tau) d\tau + K_D \dot{e}(t) \end{aligned} \quad (4.1)$$

con $K_I = \frac{K_P}{T_n}$; $K_D = K_P T_v$

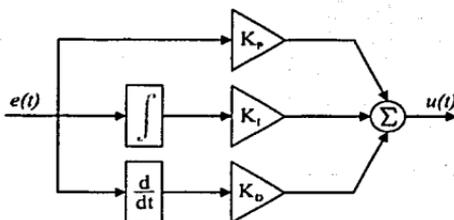


Figura 4.4: Diagrama de un PID analógico.

4.2.1 Implementación digital del controlador PID

Debido a que el sistema es digital, el control debe ser programado de forma discreta. La ecuación (4.2) describe un control PID de forma discreta con integración trapezoidal (Kuo 1996). La Figura 4.5 muestra la implementación del controlador PID digital.

$$PID_{\text{discreto}} = f_P(kT) + f_I(kT) + f_D(kT) \quad (4.2)$$

donde:

$$f_P(kT) = K_P \{e(kT)\}$$

$$f_I(kT) = K_I \left\{ \left(f_I((k-1)T) \right) + \frac{T}{2} \{e(kT) + e((k-1)T)\} \right\}$$

$$f_D(kT) = K_D \left\{ \frac{1}{T} (e(kT) - e((k-1)T)) \right\}$$

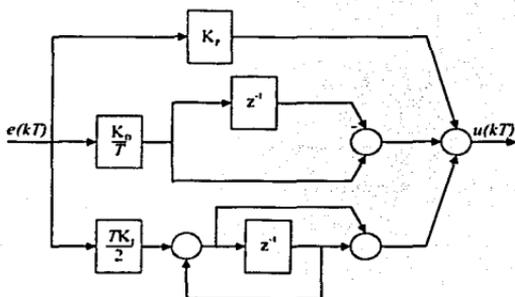


Figura 4.5: Diagrama de un PID discreto.

4.3 Sintonización de las ganancias del control PID

Sintonizar un controlador PID significa determinar los valores apropiados de K_P , T_n y T_v en la ecuación (4.1). La buena sintonización de un controlador PID es un tanto subjetiva ya que el criterio de evaluación depende de la aplicación, de la experiencia del ingeniero y de las preferencias personales del operador, pero en general el criterio que debe cumplir un sistema controlado es:

- Capaz de responder rápidamente ante cambios del valor de referencia con un mínimo sobrepaso y un tiempo rápido de estabilización
- Capaz de recuperarse rápidamente ante perturbaciones.

4.3.1 Método de ajuste de Ziegler-Nichols (Z-N)

En esta sección se verá el método de oscilación o método de respuesta en frecuencia para el ajuste de las ganancias de un controlador PID. El método se basa en un lazo de control solo con ganancia proporcional (ver Figura 4.6). De acuerdo con la ganancia necesaria para que el sistema empiece a oscilar y al periodo de esas oscilaciones, se pueden establecer las ganancias del controlador PID. Este método fue propuesto por Ziegler y Nichols (1942), quienes se basaron en la práctica para desarrollarlo.

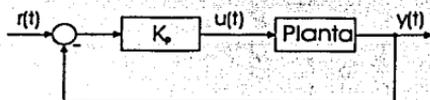


Figura 4.6: Lazo cerrado solo con ganancia proporcional.

Metodología.

Este procedimiento es válido solo para plantas estables en lazo abierto y se lleva a cabo siguiendo los siguientes pasos:

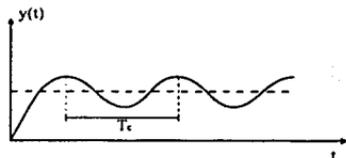


Figura 4.7: Diagrama de período crítico.

1. Utilizando sólo control proporcional y comenzando con un valor de ganancia pequeño, incrementarla hasta que el lazo comience a oscilar. Notar que se requieren oscilaciones lineales y que éstas deben ser observadas en la salida del controlador (ver Figura 4.7).
2. Registrar la ganancia crítica del controlador $K_P = K_C$ y el período de oscilación de la salida del controlador, T_C .
3. Ajustar los parámetros del controlador según la Tabla 4.2 (Astrom y Wittenmark 1990):

Controlador	K_P	T_n	T_v
<i>P</i>	$0.50K_C$		
<i>PI</i>	$0.45K_C$	$\frac{T_C}{1.2}$	
<i>PID</i>	$0.60K_C$	$0.50T_C$	$\frac{T_C}{8}$

Tabla 4.2: Parámetros de ajuste (método de oscilación).

TESIS CON
 FALLA DE ORIGEN

Los resultados obtenidos se presentan a continuación. Es importante mencionar que las ganancias tuvieron que ser ajustadas en línea para mejorar el desempeño del controlador; especialmente las constantes integrales, cuyos valores son altos y causan inestabilidad en la respuesta del sistema. Cabe notar que las ganancias obtenidas de Ziegler-Nichols marcaron un buen parámetro para comenzar a sintonizar.

	A465			A255		
	ARTICULACIONES	1	2	3	1	2
K_P	42	90	72	108	108	102
K_I	1.41	2.59	2.29	2.2	1.75	1.84
K_D	1.1	1.7	1.7	1.0	2.0	2.0

4.4 Creación de rutinas

Ahora que ya se tiene un control programado se pueden crear rutinas de inicialización de posición del robot. En este caso específico se crearon rutinas de *home* y de *ready* para los robots.

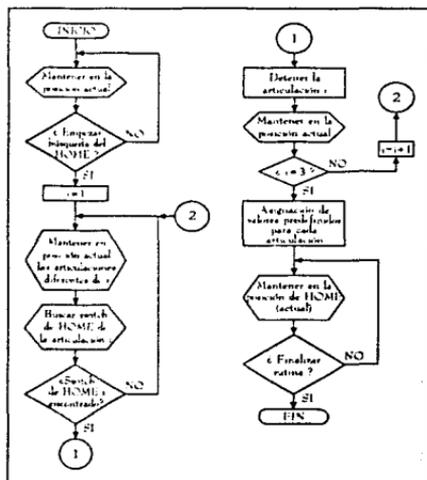


Figura 4.8: Rutina —Home A465—.

La Figura 4.8 muestra la rutina de *home* para el robot A465. Esta rutina busca los switches de home de las tres primeras articulaciones del robot. La búsqueda se

realiza de la manera siguiente:

Articulación 1: Comienza un giro positivo (CCW) hasta detectar la señal de referencia de home (cambio de 5 a 0 volts).

Articulación 2: Si la señal de referencia de home es igual a cero volts: comenzar un giro positivo (CCW) hasta detectar un cambio de estado (de 0 a 5 volts). Si la señal de referencia de home es igual a cinco volts: comenzar un giro negativo (CW) hasta detectar un cambio de estado (de 5 a 0 volts).

Articulación 3: Si la señal de referencia de home es igual a cero volts: comenzar un giro positivo (CCW) hasta detectar un cambio de estado (de 0 a 5 volts). Si la señal de referencia de home es igual a cinco volts: comenzar un giro negativo (CW) hasta detectar un cambio de estado (de 5 a 0 volts).

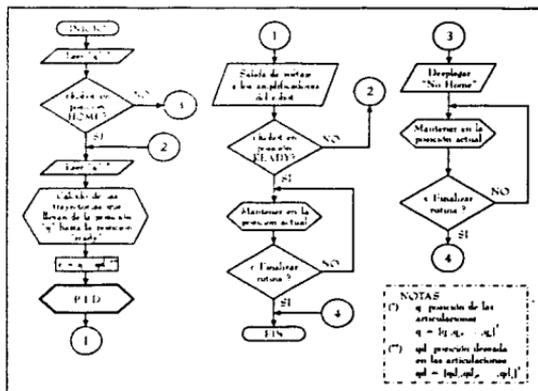


Figura 4.9: Rutina —Ready A465—.

Cuando todas las señales de referencia del home han sido encontradas, se asignan valores predefinidos de posición a cada una de las articulaciones. Esta posición es llamada "posición home". Estos valores son con respecto a la posición cero que se desea para el robot. Los valores predefinidos se muestran a continuación.

Articulación 1: -169.5816°

Articulación 2: 1.8936°

Articulación 3: 30.8331°

TESIS CON
FALLA DE ORIGEN

La rutina ready para el robot A465 se muestra en la Figura 4.9. Esta rutina lleva al robot de la posición home a la posición cero del sistema ("posición ready").

La rutina de *home* y *ready* para el robot A255 es un caso particular debido a que el robot no tiene switches de home. Por esta razón, el robot se debe colocar en una posición de referencia fija invariante para realizar la rutina que se muestra en la Figura 4.10.

Para este caso los valores predefinidos de posición para cada una de las articulaciones está referida a la posición cero que se desea para el robot. Los valores predefinidos se muestran a continuación.

Articulación 1: 0°
 Articulación 2: 77.27°
 Articulación 3: -40.75°

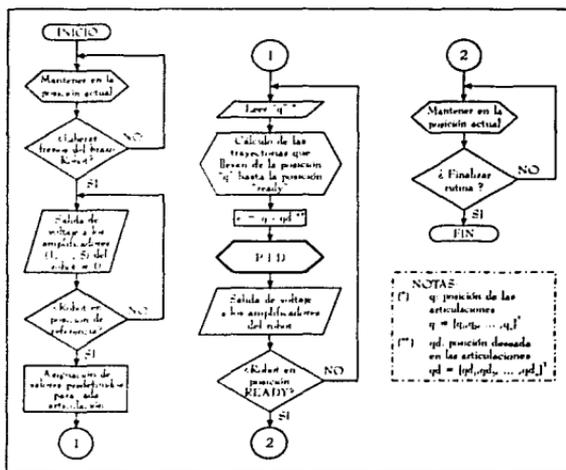


Figura 4.10: Rutina —Home y Ready A255—.

Finalmente, la Figura 4.11 muestra el diagrama de flujo de la rutina "Mantener en la posición actual", la cual es usada en las rutinas antes descritas. En el disco de

TESIS CON
FALLA DE ORIGEN

3 $\frac{1}{2}$ " está el código fuente de las rutinas descritas en esta sección.

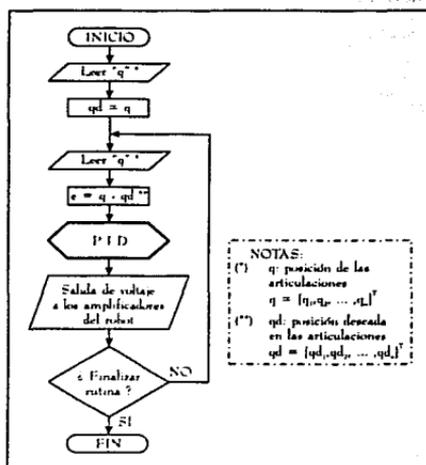


Figura 4.11: Rutina —Mantener en posición actual—.

4.5 Planeación de trayectorias

La rutina para manipulación de un objeto por los dos robots necesita la planeación de los movimientos de los manipuladores en el espacio de tres dimensiones (X, Y, Z). El cálculo de las trayectorias presentadas aquí se puede revisar en Bentrup (2002), y están dadas por:

Robot A465

$$\text{Articulación 1: } x_{d1} = x_{01} + a_{x31}t^3 + a_{x41}t^4 + a_{x51}t^5$$

$$\text{Articulación 2: } y_{d1} = y_{01} + a_{y31}t^3 + a_{y41}t^4 + a_{y51}t^5$$

$$\text{Articulación 3: } z_{d1} = z_{01} + a_{z31}t^3 + a_{z41}t^4 + a_{z51}t^5$$

TESIS CON
FALLA DE ORIGEN

Robot A255

Articulación 1: $x_{d2} = x_{02} + a_{x32}t^3 + a_{x42}t^4 + a_{x52}t^5$

Articulación 2: $y_{d2} = y_{02} + a_{y32}t^3 + a_{y42}t^4 + a_{y52}t^5$

Articulación 3: $z_{d2} = z_{02} + a_{z32}t^3 + a_{z42}t^4 + a_{z52}t^5$

Los coeficientes de estas trayectorias son los siguientes:

Intervalo: $(0, 5] \rightarrow 0\text{seg} < t \leq 5\text{seg}$

$x_{01} = 0.506$	$a_{x31} = 0.008688$	$a_{x41} = -0.0026064$	$a_{x51} = 0.000208512$
$y_{01} = 0$	$a_{y31} = 0$	$a_{y41} = 0$	$a_{y51} = 0$
$z_{01} = 0.635$	$a_{z31} = -0.022$	$a_{z41} = 0.0066$	$a_{z51} = -0.000528$
$x_{02} = 1.0432$	$a_{x32} = -0.008688$	$a_{x42} = 0.0026064$	$a_{x52} = -0.000208512$
$y_{02} = 0$	$a_{y32} = 0$	$a_{y42} = 0$	$a_{y52} = 0$
$z_{02} = 0.635$	$a_{z32} = -0.022$	$a_{z42} = 0.0066$	$a_{z52} = -0.000528$

Intervalo: $(5, 10] \rightarrow 5\text{seg} < t \leq 10\text{seg}$

$x_{01} = 0.6146$	$a_{x31} = 0.0008$	$a_{x41} = -0.00024$	$a_{x51} = 0.0000192$
$y_{01} = 0$	$a_{y31} = 0$	$a_{y41} = 0$	$a_{y51} = 0$
$z_{01} = 0.36$	$a_{z31} = 0$	$a_{z41} = 0$	$a_{z51} = 0$
$x_{02} = 0.9346$	$a_{x32} = -0.0008$	$a_{x42} = 0.00024$	$a_{x52} = -0.0000192$
$y_{02} = 0$	$a_{y32} = 0$	$a_{y42} = 0$	$a_{y52} = 0$
$z_{02} = 0.36$	$a_{z32} = 0$	$a_{z42} = 0$	$a_{z52} = 0$

Intervalo: $(10, 15] \rightarrow 10\text{seg} < t \leq 15\text{seg}$

$x_{01} = 0.6246$	$a_{x31} = 0.004$	$a_{x41} = -0.0012$	$a_{x51} = 0.000096$
$y_{01} = 0$	$a_{y31} = 0$	$a_{y41} = 0$	$a_{y51} = 0$
$z_{01} = 0.36$	$a_{z31} = 0.022$	$a_{z41} = -0.0066$	$a_{z51} = 0.000528$
$x_{02} = 0.9246$	$a_{x32} = 0.004$	$a_{x42} = -0.0012$	$a_{x52} = 0.000096$
$y_{02} = 0$	$a_{y32} = 0$	$a_{y42} = 0$	$a_{y52} = 0$
$z_{02} = 0.36$	$a_{z32} = 0.022$	$a_{z42} = -0.0066$	$a_{z52} = 0.000528$

Intervalo: $(25, 30] \rightarrow 25\text{seg} < t \leq 30\text{seg}$

$x_{01} = 0.6746$	$a_{x31} = -0.004$	$a_{x41} = 0.0012$	$a_{x51} = -0.000096$
$y_{01} = 0$	$a_{y31} = 0$	$a_{y41} = 0$	$a_{y51} = 0$
$z_{01} = 0.635$	$a_{z31} = -0.022$	$a_{z41} = 0.0066$	$a_{z51} = -0.000528$
$x_{02} = 0.9746$	$a_{x32} = -0.004$	$a_{x42} = 0.0012$	$a_{x52} = -0.000096$
$y_{02} = 0$	$a_{y32} = 0$	$a_{y42} = 0$	$a_{y52} = 0$
$z_{02} = 0.635$	$a_{z32} = -0.022$	$a_{z42} = 0.0066$	$a_{z52} = -0.000528$

TESIS CON
FALLA DE ORIGEN

Intervalo: (30, 35] $\rightarrow 30\text{seg} < t \leq 35\text{seg}$

$x_{01} = 0.6246$	$a_{x31} = -0.0008$	$a_{x41} = 0.00024$	$a_{x51} = -0.0000192$
$y_{01} = 0$	$a_{y31} = 0$	$a_{y41} = 0$	$a_{y51} = 0$
$z_{01} = 0.36$	$a_{z31} = 0$	$a_{z41} = 0$	$a_{z51} = 0$
$x_{02} = 0.9246$	$a_{x32} = 0.0008$	$a_{x42} = -0.00024$	$a_{x52} = 0.0000192$
$y_{02} = 0$	$a_{y32} = 0$	$a_{y42} = 0$	$a_{y52} = 0$
$z_{02} = 0.36$	$a_{z32} = 0$	$a_{z42} = 0$	$a_{z52} = 0$

Intervalo: (35, 40] $\rightarrow 35\text{seg} < t \leq 40\text{seg}$

$x_{01} = 0.6146$	$a_{x31} = -0.008688$	$a_{x41} = 0.0026064$	$a_{x51} = -0.000208512$
$y_{01} = 0$	$a_{y31} = 0$	$a_{y41} = 0$	$a_{y51} = 0$
$z_{01} = 0.36$	$a_{z31} = 0.022$	$a_{z41} = -0.0066$	$a_{z51} = 0.000528$
$x_{02} = 0.9346$	$a_{x32} = 0.008688$	$a_{x42} = -0.0026064$	$a_{x52} = 0.000208512$
$y_{02} = 0$	$a_{y32} = 0$	$a_{y42} = 0$	$a_{y52} = 0$
$z_{02} = 0.36$	$a_{z32} = 0.022$	$a_{z42} = -0.0066$	$a_{z52} = 0.000528$

Para el caso donde $15\text{seg} < t \leq 25\text{seg}$, se tiene los siguientes polinomios

Intervalo: (15, 25] $\rightarrow 15\text{seg} < t \leq 25\text{seg}$

$$w = w_0 + a_{x31}t^3 + a_{x41}t^4 + a_{x51}t^5$$

$$x_{d1} = x_r + r \cos(w)$$

$$y_{d1} = y_r + r \sin(w)$$

$$z_{d1} = z_r + r \cos(w) - r$$

$$x_{d2} = x_{d1} + \text{largo}$$

$$y_{d2} = y_{d1}$$

$$z_{d2} = z_{d1}$$

$w_0 = 0$	$a_{x31} = 0.062832$	$a_{x41} = -0.0094248$	$a_{x51} = 0.000376992$
$x_r = 0.6246$	$y_r = 0$	$z_r = 0.635$	$r = 0.05$
$\text{largo} = 0.3$			

Las Figuras 4.12, 4.13 y 4.14 muestran la trayectoria que deben seguir los robots.

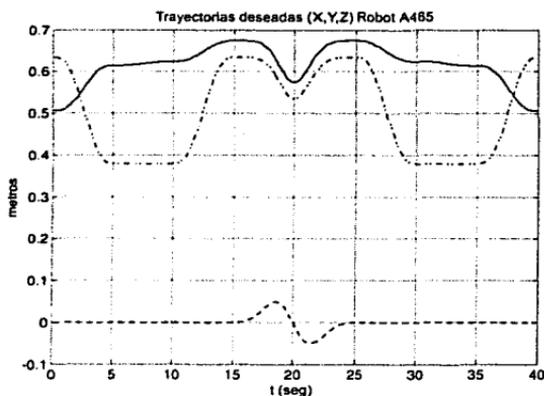


Figura 4.12: Trayectoria deseada en coordenadas rectangulares X (—), Y (- -), Z (- · ·) para el robot A485.

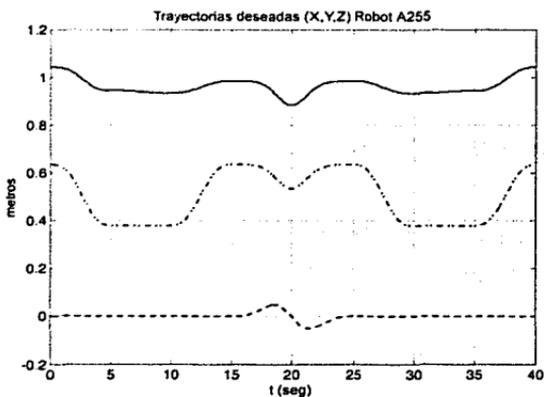


Figura 4.13: Trayectoria deseada en coordenadas rectangulares X (—), Y (- -), Z (- · ·) para el robot A255.

TESIS CON
FALLA DE ORIGEN

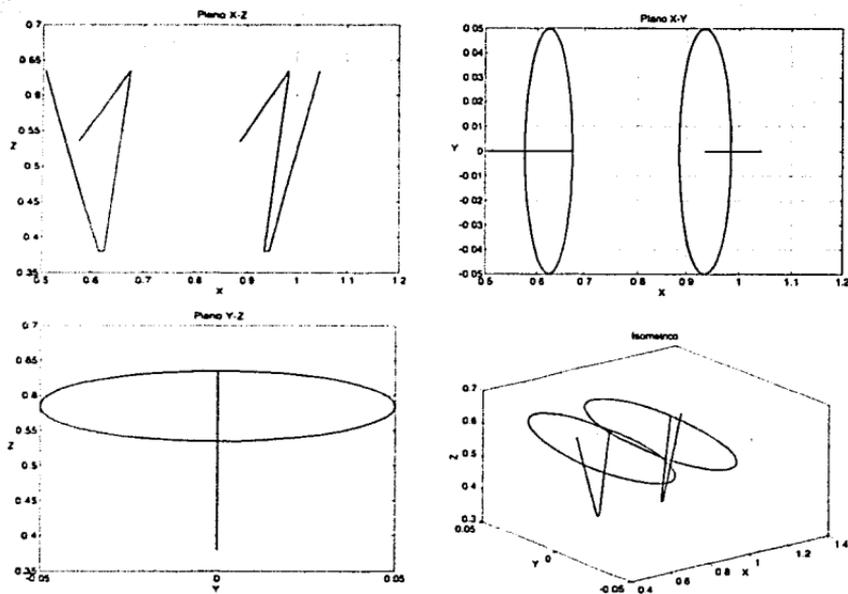


Figura 4.14: Isométrico de las trayectorias deseadas para los robots A465 y A255.

4.6 Obtención de la cinemática inversa de los robots A465 y A255

Las trayectorias obtenidas en la sección anterior están dadas en coordenadas rectangulares (x, y, z) ; pero es necesario conocer el valor de cada una de las tres primeras articulaciones de los robots para llevar a cabo el movimiento deseado. Para lograrlo hace falta conocer la cinemática inversa de los robots. La metodología para obtenerla se puede revisar en Spong y Vidyasagar (1989). Aquí se presenta sin detalles la obtención de la cinemática inversa.

TESIS CON
FALLA DE ORIGEN

4.6.1 Cinemática directa

Antes de calcular la cinemática inversa debemos conocer la cinemática directa de los robots. El problema de la cinemática directa es determinar la posición del *efector final* de acuerdo a la posición angular que tiene cada una de las articulaciones de los robots. La forma más común de representar la cinemática directa es a través de una matriz de transformación homogénea ${}^0T_n \in \mathbb{R}^{4 \times 4}$, donde n es el número de articulaciones del robot. La matriz 0T_n está definida como

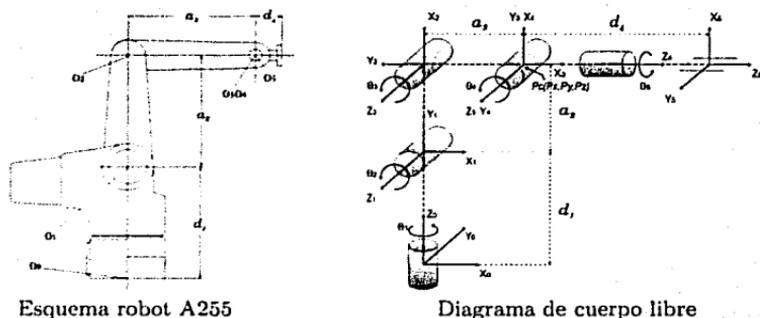
$${}^0T_n = {}^0A_1 {}^1A_2 \dots {}^{i-1}A_i$$

donde ${}^{i-1}A_i$ con $i = 1, 2, \dots, n$, es la matriz homogénea que transforma las coordenadas de un punto del sistema de coordenadas i al sistema $i - 1$. Las Figuras 4.15 y 4.16 muestran un esquema de los robots y sus correspondientes diagramas de cuerpo libre. La Tabla 4.3 muestra los parámetros del diagrama de cuerpo libre.

Para el caso del robot A465 se utilizó el algoritmo de Denavit-Hartenberg para obtener las matrices de transformación (Spong y Vidyasagar 1989). Para el robot A255 las matrices de transformación se obtienen por inspección debido a la configuración mecánica del robot. Las matrices de transformación homogénea quedan como sigue

A	Robot A255		Robot A465	
	a_i (metros)	d_i (metros)	a_i (metros)	d_i (metros)
1	0	0.254	0	0.330
2	0.254	0	0.305	0
3	0.254	0	0	0
4	0	0	0	0.330
5	0	0.051	0	0
6	—	—	0	0.076

Tabla 4.3: Parámetros de los robot A255 y A465.



Esquema robot A255

Diagrama de cuerpo libre

Figura 4.15: Esquema y diagrama de cuerpo libre para el robot A255.

Robot A255

$${}^0A_1 = \begin{bmatrix} C_1 & 0 & S_1 & 0 \\ S_1 & 0 & -C_1 & 0 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^1A_2 = \begin{bmatrix} 1 & 0 & 0 & a_2 C_2 \\ 0 & 1 & 0 & a_2 S_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2A_3 = \begin{bmatrix} 1 & 0 & 0 & a_3 C_3 \\ 0 & 1 & 0 & a_3 S_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^3A_4 = \begin{bmatrix} -S_4 & 0 & C_4 & 0 \\ C_4 & 0 & S_4 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^4A_5 = \begin{bmatrix} C_5 & -S_5 & 0 & 0 \\ S_5 & C_5 & 0 & 0 \\ 0 & 0 & 1 & d_5 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

TESIS CON
FALLA DE ORIGEN

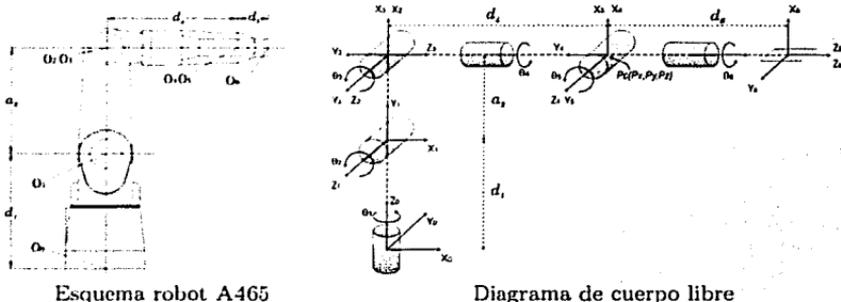


Figura 4.16: Esquema y diagrama de cuerpo libre para el robot A465.

Robot A465

$${}^0A_1 = \begin{bmatrix} C_1 & 0 & S_1 & 0 \\ S_1 & 0 & -C_1 & 0 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^1A_2 = \begin{bmatrix} C_2 & -S_2 & 0 & a_2 C_2 \\ S_2 & C_2 & 0 & a_2 S_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2A_3 = \begin{bmatrix} C_3 & 0 & S_3 & 0 \\ S_3 & 0 & -C_3 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^3A_4 = \begin{bmatrix} C_4 & 0 & -S_4 & 0 \\ S_4 & 0 & C_4 & 0 \\ 0 & -1 & 0 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^4A_5 = \begin{bmatrix} C_5 & 0 & S_5 & 0 \\ S_5 & 0 & -C_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^5A_6 = \begin{bmatrix} C_6 & -S_6 & 0 & 0 \\ S_6 & C_6 & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

con: $S_i = \sin(\theta_i)$ y $C_i = \cos(\theta_i)$

Como se mencionó al principio de la sección, sólo se utilizarán las tres primeras articulaciones del robot, por lo tanto $\theta_4 = \theta_5 = \theta_6 \equiv 0$ para el robot A465 y $\theta_4 = \theta_5 \equiv 0$ para el robot A255.

4.6.2 Cinemática inversa

A continuación se calculará la cinemática inversa de los robots. El problema de la cinemática inversa es determinar los ángulos de cada una de las articulaciones

TESIS CON
FALLA DE ORIGEN

de los robots a partir de la posición del *efector final*, en coordenadas rectangulares (X, Y, Z). La cinemática inversa para los robots A465 y A255 puede ser dividida en dos problemas independientes: posición inversa y orientación inversa. El procedimiento es similar para los dos robots y es descrito a continuación.

Posición Inversa

La posición del *efector final* en los robots está determinada por las tres primeras articulaciones de los mismos. Recordando las Figuras 4.16 y 4.15 el valor de P_c se calcula como:

Robot A255		Robot A465
$P_c = d - d_5 Rk$		$P_c = d - d_6 Rk$

donde:

d : posición deseada del *efector final* medido desde la base del robot.

d_6 : distancia entre el *efector final* y el centro de la muñeca del robot A465.

d_5 : distancia entre el *efector final* y el centro de la muñeca del robot A255.

R : matriz que denota la orientación deseada del *efector final* con respecto al sistema de coordenadas $0X_0Y_0Z_0$.

$$k = [0 \ 0 \ 1]^T$$

Cálculo de θ_1 . El cálculo de θ_1 es idéntico para los dos robots. La Figura 4.17 muestra el plano $X_0 - Y_0$ donde se puede ver la relación de θ_1 con P_x y P_y .

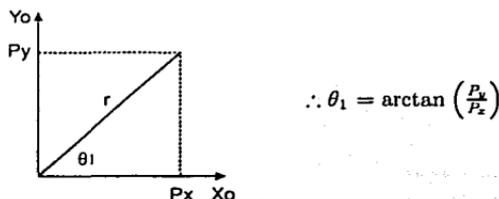
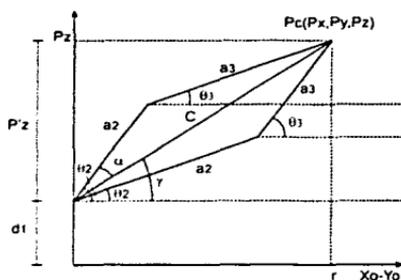


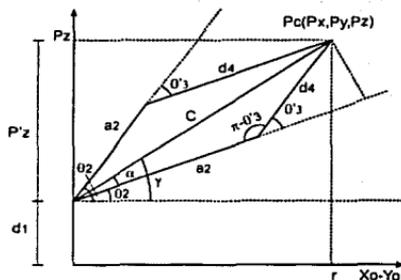
Figura 4.17: Plano $X_0 - Y_0$.

Cálculo de θ_2 y θ_3 . El cálculo de θ_2 y θ_3 se apoya en la Figura 4.18

TESIS CON
 FALLA DE ORIGEN



Esquema para el robot A255



Esquema para el robot A465

Figura 4.18: Diagrama vectorial de las articulaciones 2 y 3 de los robots.

Robot A255	$\theta_2 = \arctan\left(\frac{\pm\sqrt{1-\sigma^2}}{\sigma}\right) + \arctan\left(\frac{P_z-d_1}{\sqrt{P_x^2+P_y^2}}\right)$ $\theta_3 = \arctan\left(\frac{\psi}{\pm\sqrt{a_3^2-\psi^2}}\right)$
Robot A465	$\theta_2 = \arctan\left(\frac{P_z-d_1}{\sqrt{P_x^2+P_y^2}}\right) - \arctan\left(\frac{d_4 \sin \theta'_3}{a_2+d_4 \cos \theta'_3}\right)$ $\theta_3 = \theta'_3 + \frac{\pi}{2}$

con:

$$\sigma \triangleq \frac{(a_2)^2 + (P_x)^2 + (P_y)^2 + (P_z)^2 - (a_3)^2}{2a_2\sqrt{(P_x)^2 + (P_y)^2 + (P_z)^2}}$$

$$\psi \triangleq P_z - (a_2 S_2 + d_1)$$

$$\theta'_3 = \arctan\left(\frac{\pm\sqrt{1-D^2}}{D}\right)$$

$$D \triangleq \frac{(P_x-d_1)^2 + P_z^2 + P_y^2 - a_2^2 - d_4^2}{2a_2d_4}$$

TESIS CON
 FALLA DE ORIGEN

Orientación Inversa

La orientación del *efector final* está determinada por las tres últimas articulaciones en el robot A465 y por las dos últimas para el robot A255. Para encontrar θ_4 , θ_5 y θ_6 (sólo para A465) se toma en cuenta que:

$$\begin{aligned} \text{Robot A465:} \quad {}^0R_6 &= {}^0R_1 {}^1R_2 {}^2R_3 {}^3R_4 {}^4R_5 {}^5R_6 = R = {}^0R_3 {}^3R_6 \\ \text{Robot A255:} \quad {}^0R_5 &= {}^0R_1 {}^1R_2 {}^2R_3 {}^3R_4 {}^4R_5 = R = {}^0R_3 {}^3R_5 \end{aligned} \quad (4.3)$$

donde:

$${}^0R_3 = {}^0R_1 {}^1R_2 {}^2R_3$$

$${}^3R_6 = {}^3R_4 {}^4R_5 {}^5R_6$$

$${}^3R_5 = {}^3R_4 {}^4R_5$$

Despejando 3R_6 y 3R_5 de (4.3) se tiene

$$\text{Robot A465:} \quad U \triangleq {}^3R_6(\theta_4, \theta_5, \theta_6) = {}^0R_3^T R = \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ u_{21} & u_{22} & u_{23} \\ u_{31} & u_{32} & u_{33} \end{bmatrix} \quad (4.4)$$

$$\text{Robot A255:} \quad V \triangleq {}^3R_5(\theta_4, \theta_5) = {}^0R_3^T R = \begin{bmatrix} v_{11} & v_{12} & v_{13} \\ v_{21} & v_{22} & v_{23} \\ v_{31} & v_{32} & v_{33} \end{bmatrix}$$

Las matrices 3R_6 y 3R_5 son calculadas con la ayuda de los resultados obtenidos de la cinemática directa, por lo tanto se obtiene

$$\text{A465:} \quad U = \begin{bmatrix} C_4 C_5 C_6 - S_4 S_6 & -C_4 C_5 S_6 - S_4 C_6 & C_4 S_5 \\ S_4 C_5 C_6 + C_4 S_6 & -S_4 C_5 S_6 + C_4 C_6 & S_4 S_5 \\ -S_5 C_6 & S_5 S_6 & C_5 \end{bmatrix} = \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ u_{21} & u_{22} & u_{23} \\ u_{31} & u_{32} & u_{33} \end{bmatrix} \quad (4.5)$$

$$\text{A255:} \quad V = \begin{bmatrix} -S_4 C_5 & S_4 S_5 & C_4 \\ C_4 C_5 & -C_4 S_5 & S_4 \\ S_5 & C_5 & 0 \end{bmatrix} = \begin{bmatrix} v_{11} & v_{12} & v_{13} \\ v_{21} & v_{22} & v_{23} \\ v_{31} & v_{32} & v_{33} \end{bmatrix} \quad (4.6)$$

A partir de las ecuaciones (4.5) y (4.6) se calculará los ángulos para los robots

Robot A465

- Si u_{13} y $u_{23} \neq 0 \Rightarrow u_{33} \neq \pm 1$

$$\theta_5 = \arctan\left(\frac{\pm\sqrt{1-u_{33}^2}}{u_{33}}\right)$$

Se observa que θ_4 y θ_6 dependen del signo que se elija para el radical en θ_5

Si se elige (+) tenemos:

$$\theta_4 = \arctan\left(\frac{u_{21}}{u_{13}}\right)$$

$$\theta_6 = \arctan\left(\frac{u_{32}}{-u_{31}}\right)$$

Si se elige (-) tenemos:

$$\theta_4 = \arctan\left(\frac{-u_{21}}{-u_{13}}\right)$$

$$\theta_6 = \arctan\left(\frac{-u_{32}}{u_{31}}\right)$$

- Si $u_{33} = 1 \Rightarrow \theta_5 = 0 \quad \therefore \sin(\theta_5) = 0 ; \cos(\theta_5) = 1$

$$\Rightarrow \theta_4 + \theta_6 = \arctan\left(\frac{u_{21}}{u_{22}}\right) = \arctan\left(\frac{-u_{12}}{u_{11}}\right) = \arctan\left(\frac{u_{21}}{u_{11}}\right) = \arctan\left(\frac{-u_{12}}{u_{22}}\right)$$

Por convención tenemos $\theta_4 \hat{=} 0$, entonces

$$\theta_6 = \arctan\left(\frac{u_{21}}{u_{22}}\right) = \arctan\left(\frac{-u_{12}}{u_{11}}\right) = \arctan\left(\frac{u_{21}}{u_{11}}\right) = \arctan\left(\frac{-u_{12}}{u_{22}}\right)$$

- Si $u_{33} = -1 \Rightarrow \theta_5 = \pi \quad \therefore \sin(\theta_5) = 0 ; \cos(\theta_5) = -1$

$$\Rightarrow \theta_4 - \theta_6 = \arctan\left(\frac{-u_{21}}{u_{22}}\right) = \arctan\left(\frac{-u_{12}}{-u_{11}}\right) = \arctan\left(\frac{-u_{21}}{-u_{11}}\right) = \arctan\left(\frac{-u_{12}}{u_{22}}\right)$$

Por convención tenemos $\theta_4 \hat{=} 0$, entonces

$$\theta_6 = \arctan\left(\frac{-u_{21}}{u_{22}}\right) = \arctan\left(\frac{-u_{12}}{-u_{11}}\right) = \arctan\left(\frac{-u_{21}}{-u_{11}}\right) = \arctan\left(\frac{-u_{12}}{u_{22}}\right)$$

Robot A255

De la ecuación (4.6) se observa que $u_{33} \equiv 0$ para que exista una solución posible. Entonces, θ_4 y θ_5 se pueden calcular como:

$$\theta_4 = \arctan\left(\frac{v_{24}}{v_{13}}\right) = \arctan\left(\frac{-v_{11}}{v_{21}}\right) = \arctan\left(\frac{v_{21}}{-v_{22}}\right)$$

$$\theta_5 = \arctan\left(\frac{v_{11}}{v_{32}}\right) = \arctan\left(\frac{v_{12}}{-v_{11}}\right) = \arctan\left(\frac{-v_{21}}{v_{21}}\right)$$

Ahora que la cinemática inversa ha sido obtenida se pueden calcular los ángulos de cada una de las articulaciones de los robots. Las Figuras 4.19 y 4.20 muestran las curvas obtenidas de aplicar la cinemática inversa a las trayectorias deseadas.

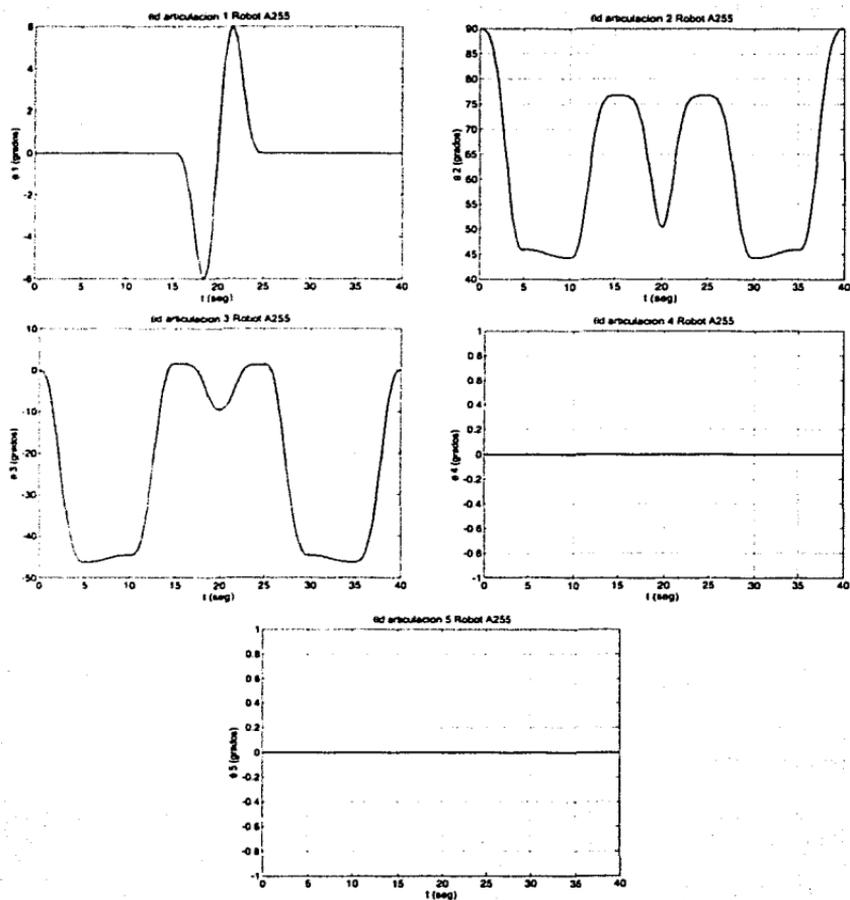


Figura 4.19: Ángulos de las articulaciones del robot A255.

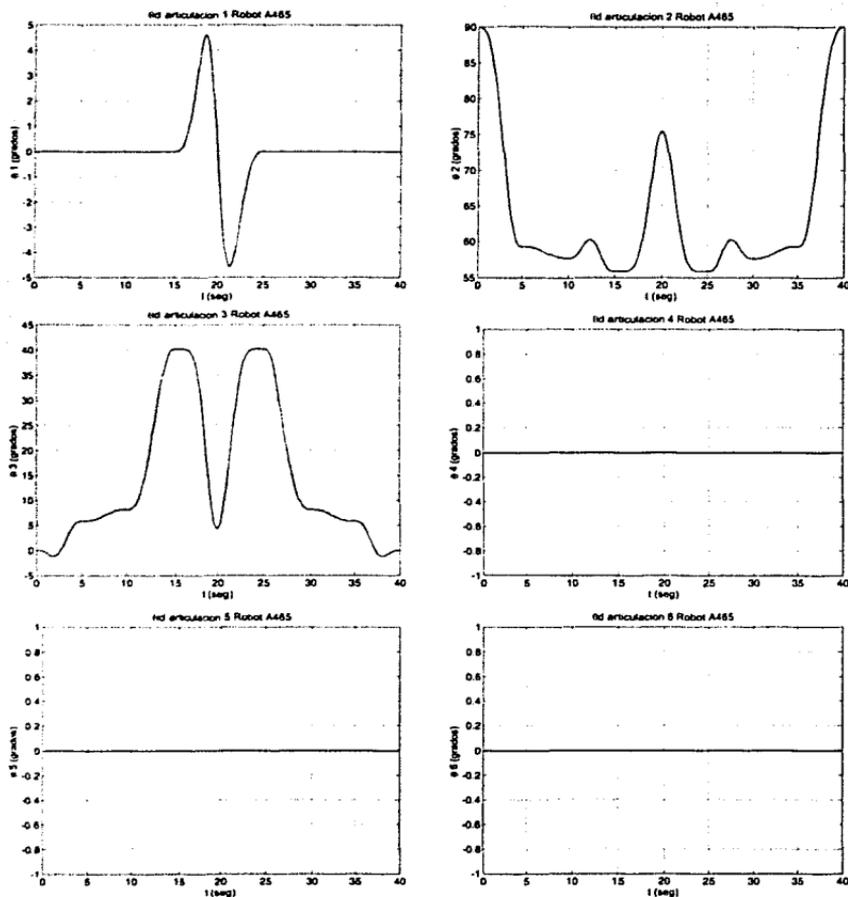


Figura 4.20: Ángulos de las articulaciones del robot A465.

TESIS CON
FALLA DE ORIGEN

4.7 Resultados experimentales

En esta sección se presentan los resultados obtenidos de implementar las trayectorias propuestas en la Sección 4.5. Las Figuras 4.22 y 4.23 muestran las trayectorias deseadas vs las trayectorias reales de las articulaciones de los robots, además de los errores de seguimiento. Es importante hacer notar que, aunque el error de seguimiento no es aproximadamente cero, los resultados son excelentes puesto que se logró programar la cinemática inversa del robot en línea y utilizarla con propósitos de control. Por último, la Figura 4.21 muestra el sistema adaptado en acción. En el Apéndice E se muestran fotografías del sistema.

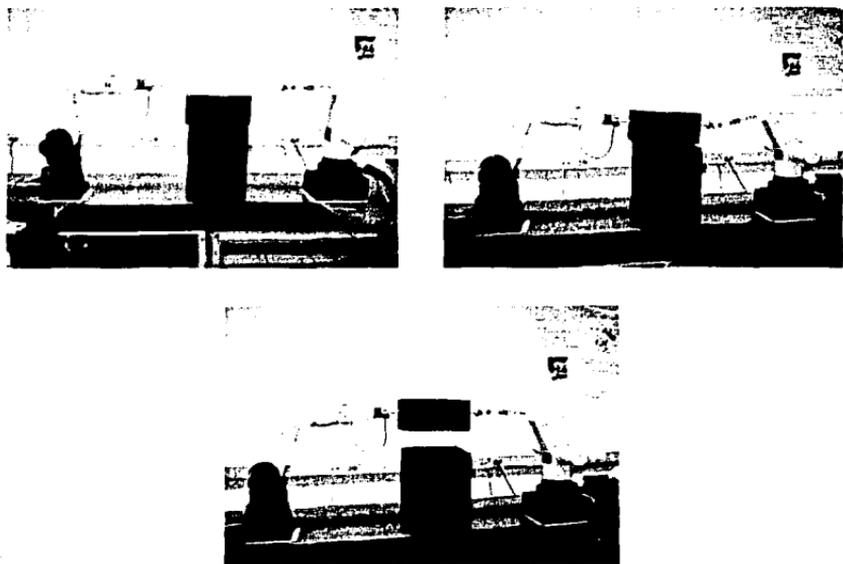


Figura 4.21: Fotografías del sistema adaptado en acción.

TESIS CON
FALLA DE ORIGEN

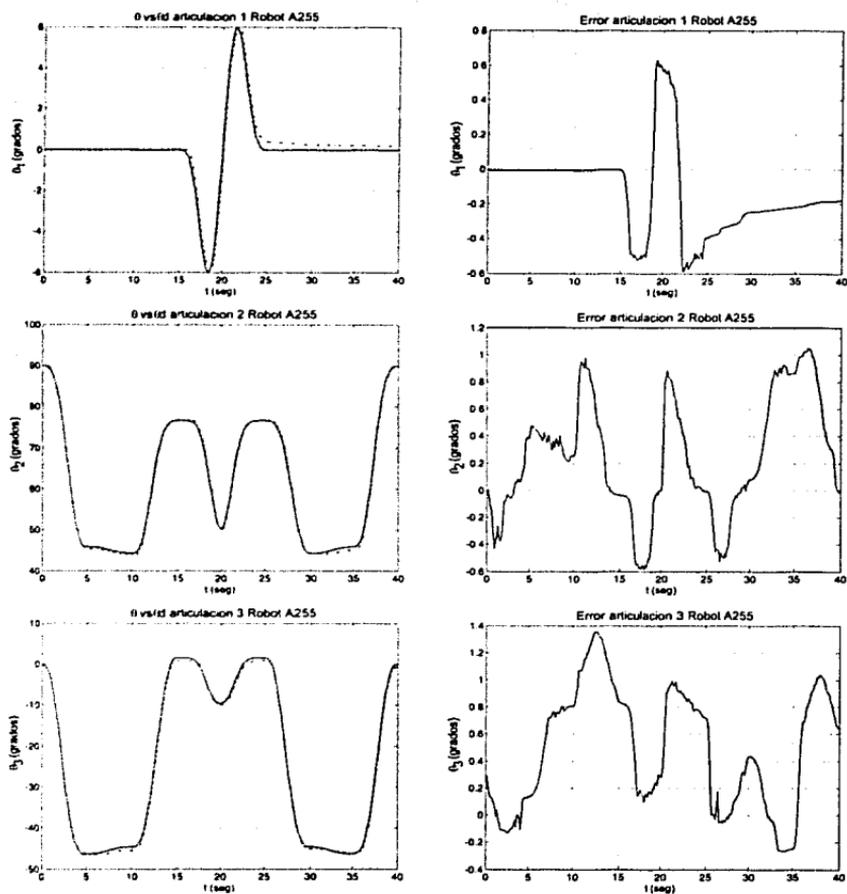


Figura 4.22: Trayectoria deseada (—) vs trayectoria real (- - -) de las articulaciones del robot A255 y su correspondiente error de seguimiento.

TESIS CON
FALLA DE ORIGEN

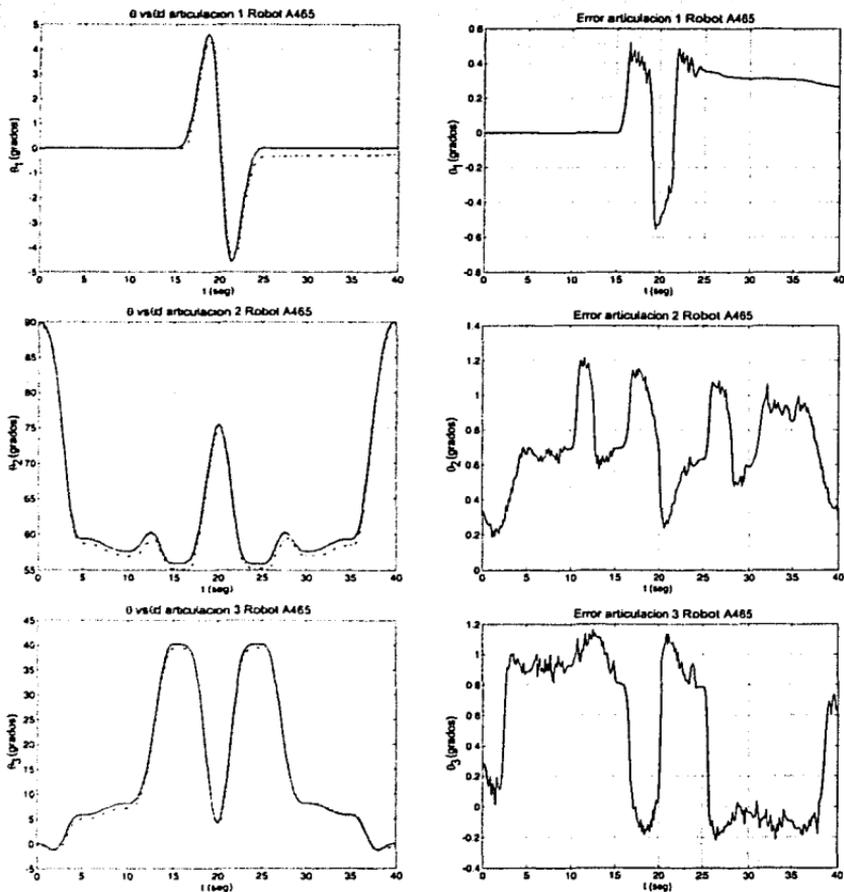


Figura 4.23: Trayectoria deseada (—) vs trayectoria real (- - -) de las articulaciones del robot A465 y su correspondiente error de seguimiento.

TESIS CON
FALLA DE ORIGEN

Capítulo 5

Conclusiones

En la presente tesis se ha descrito la experiencia de crear un sistema robótico experimental para comprobación de algoritmos de control. El principal resultado de esta experiencia muestra que es posible realizar un sistema robótico basado en los robots industriales A465 y A255 de *CRS Robotics*, no prototipos, sin re-diseñar por completo el hardware del controlador original. Se propuso una metodología para la adaptación de sistemas industriales a sistemas experimentales. El problema de coordinar la PC con el controlador original de los robots se analizó y se propuso una solución. Finalmente se validó el sistema programando un algoritmo de control PID para que los dos robots manipulen un objeto de forma conjunta.

5.1 Ventajas y desventajas

El sistema creado presenta características muy interesantes para la comprobación de algoritmos de control en el sentido de capacidades mecánicas y eléctricas, debido a que está conformado por robots industriales. De las ventajas con las que cuenta este sistema se encuentran las siguientes:

- El sistema cuenta con características de carga, repetitividad y precisión industriales que aumentan las perspectivas de usar los algoritmos comprobados en robots ya instalados en la industria.
- El sistema puede ser regresado fácilmente a su configuración de fábrica para utilizar los robots en aplicaciones donde la comprobación de algoritmos de control no es necesario, por ejemplo la planeación de celdas de trabajo.
- Los robots tiene 5 y 6 grados de libertad, lo que permite alcanzar posición arbitraria en el espacio de tres dimensiones *OXYZ*. En el caso del robot de 6

GDL también se obtiene orientación arbitraria. Cabe resaltar que la mayoría de los robots experimentales prototipo cuentan con sólo dos grados de libertad.

- El sistema puede manipular los dos robots completamente independientes o simultáneamente coordinados vía una computadora personal.
- Para el usuario la operación del sistema adaptado se lleva a cabo desde la computadora maestra, sin tener que conocer en detalle las especificaciones del sistema.
- La programación se basa en lenguaje ANSI C, lo cual hace simple la programación para el común de técnicos, ingenieros y especialistas en control y robótica.
- La programación de los algoritmos de control se puede realizar fuera de línea; esto permite que otras instituciones puedan programar algoritmos de control para su posterior comprobación en el sistema creado.
- En el sistema se pueden desarrollar y experimentar con diversos algoritmos de control. Las características de la PC y del sistema adaptado permiten anexar diferentes sensores y herramientas, como por ejemplo: sensores de fuerza, cámaras de video y/o pinzas.
- El sistema permite realizar experimentos con varias configuraciones mecánicas del robot. Llevando a cabo una combinación de las articulaciones utilizadas se puede configurar un robot con 1, 2, . . . , n grados de libertad.
- Por último, si los robots son desconectados el hardware residente en la PC puede ser usado para controlar muy diversos servo sistemas.

Por otra parte el sistema tiene también desventajas que limitan su rendimiento, a continuación se listan los inconvenientes

- El tiempo de muestreo de los controladores programados está limitado por la velocidad de adquisición de las tarjetas, que para el caso de una computadora Pentium 4 es de hasta 6ms, para controlar 6 ejes de movimiento. Este se considera un tiempo de muestreo grande para aplicaciones que necesitan trayectorias de alta frecuencia o cuando se aplican técnicas de control adaptable que requieren de gran velocidad en el procesamiento.
- El sistema tiene limitaciones en el uso de los ejes que reducen el rendimiento del sistema, lo que impide, por el momento, controlar todas las articulaciones de los robots al mismo tiempo. Esto es debido a las tarjetas de adquisición de datos.

5.2 Trabajo futuro y perspectivas

El siguiente paso lógico es mejorar el sistema resolviendo las desventajas del mismo. El trabajo futuro básicamente es:

- Desarrollar un programa estándar en el cual las funciones de entrada/salida y de protección del sistema (saturación, límites de viaje mecánico) estén establecidas, de tal forma que el usuario sólo incluya el algoritmo de control bajo las condiciones asentadas.
- Establecer rutinas en tiempo real (real-time) para el sistema adaptado garantizando un tiempo de muestreo uniforme.
- Seguir buscando la forma de reducir el tiempo de muestreo, ya sea con otras tarjetas o aumentando la velocidad de la computadora maestra.
- Buscar la forma superar completamente la limitación de ejes en el sistema adaptado.

Por el momento el sistema creado juega un rol central en las actividades de investigación desarrolladas en el Laboratorio de Robótica de la DEPEI y las perspectivas son grandes para el sistema, si se toma en cuenta que no existe uno parecido en México. La idea es implementar diversos algoritmos de control (visión, fuerza, etc.) usando diferentes técnicas de control como difuso, adaptable, robusto, modos deslizantes, etc., siendo extensivo a estrategias avanzadas de control para robots cooperativos, técnicas que la mayoría de las veces sólo han sido comprobadas en simulación.

Apéndice A

Especificaciones de los robots A465 y A255

Este apéndice describe a grandes rasgos las características más importantes de los sistemas robot A465 y A255. La Tabla A.1 muestra el rango de trabajo de cada robot. Las Figuras A.1, A.2, A.3 y A.4 muestran el área de trabajo de los robots.

RANGO DE TRABAJO Y VELOCIDAD			
ROBOT	EJES	RANGO	VELOCIDAD MÁXIMA
A465	J1 (cintura)	$\pm 175^\circ$	180 °/seg.
	J2 (hombro)	$\pm 90^\circ$	180 °/seg.
	J3 (codo)	$\pm 110^\circ$	180 °/seg.
	J4 (rotación de muñeca)	$\pm 180^\circ$	171 °/seg.
	J5 (inclinación de muñeca)	$\pm 105^\circ$	173 °/seg.
	J6 (rotación de la herramienta)	$\pm 180^\circ$	171 °/seg.
A255	J1 (cintura)	350° (totales)	210 °/seg.
	J2 (hombro)	110° (totales)	210 °/seg.
	J3 (codo)	125° (totales)	210 °/seg.
	J4 (inclinación de muñeca)	220° (totales)	675 °/seg.
	J5 (rotación de muñeca)	360° (totales)	1350 °/seg.

Tabla A.1: Rango de trabajo y velocidad de los robot A465 y A255

TESIS CON
FALLA DE ORIGEN

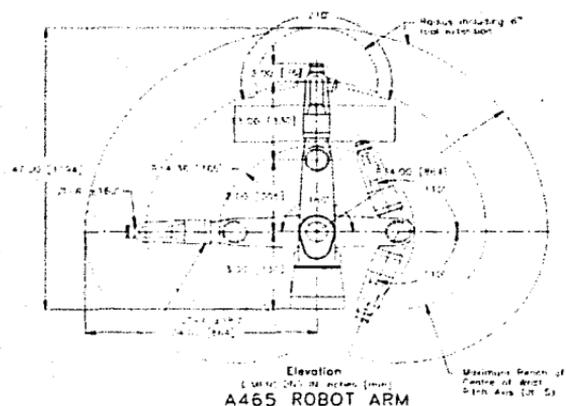


Figura A.1: Área de trabajo del robot A465 (vista lateral)

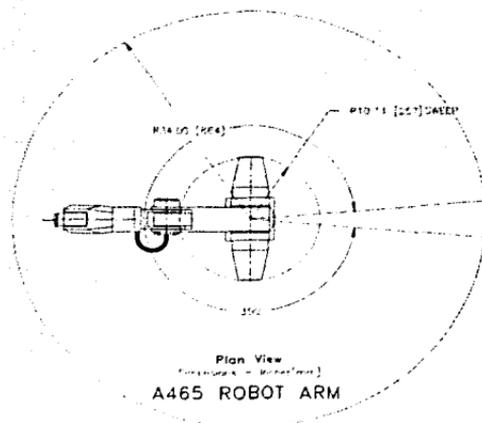


Figura A.2: Área de trabajo del robot A465 (vista superior)

TESIS CON
FALLA DE ORIGEN

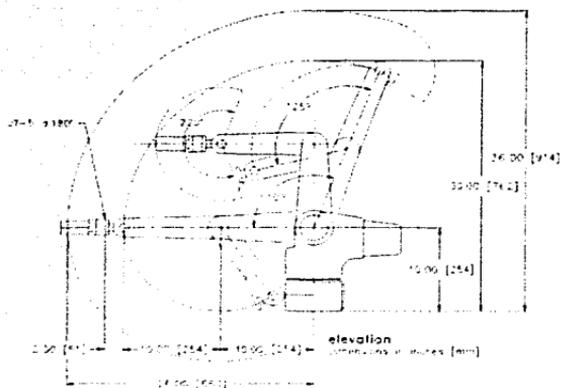


Figura A.3: Área de trabajo del robot A255 (vista lateral)

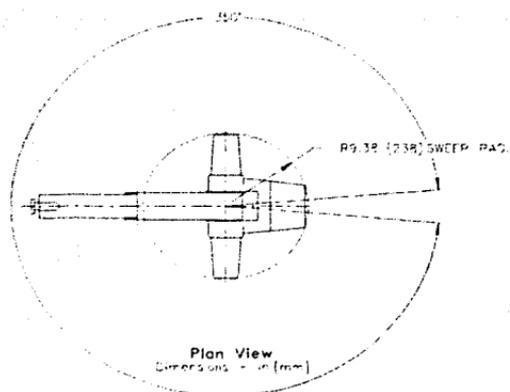


Figura A.4: Área de trabajo del robot A255 (vista superior)

TESIS CON
FALLA DE ORIGEN

Pin#	Function	Signature	Description
1	Vcom1	Analog	Command voltage for axis 1
2	Vcom5	Analog	Command voltage for axis 5
3	Vcom2	Analog	Command voltage for axis 2
4	Vcom6	Analog	Command voltage for axis 6
5	Vcom3	Analog	Command voltage for axis 3
6	Vcom7	Analog	Command voltage for axis 7
7	Vcom4	Analog	Command voltage for axis 4
8	Vcom8	Analog	Command voltage for axis 8
9	HSW1	OC	Homing switch for axis 1
10	PSW1	OC	Positive Travel switch for axis 1
11	NSW1	OC	Negative travel switch for axis 1
12	TSW1	OC	Thermal switch for axis 1
13	HSW2	OC	Homing switch for axis 2
14	PSW2	OC	Positive Travel switch for axis 2
15	NSW2	OC	Negative travel switch for axis 2
16	TSW2	OC	Thermal switch for axis 2
17	HSW3	OC	Homing switch for axis 3
18	PSW3	OC	Positive Travel switch for axis 3
19	NSW3	OC	Negative travel switch for axis 3
20	TSW3	OC	Thermal switch for axis 3
21	HSW4	OC	Homing switch for axis 4
22	PSW4	OC	Positive Travel switch for axis 4
23	NSW4	OC	Negative travel switch for axis 4
24	TSW4	OC	Thermal switch for axis 4
25	HSW5	OC	Homing switch for axis 5
26	PSW5	OC	Positive Travel switch for axis 5
27	NSW5	OC	Negative travel switch for axis 5
28	TSW5	OC	Thermal switch for axis 5
29	HSW6	OC	Homing switch for axis 6
30	PSW6	OC	Positive Travel switch for axis 6

Tabla A.2: Asignación de pines del conector J6 de la tarjeta CB del controlador C500.

TESIS CON
FALLA DE ORIGEN

Pin#	Function	Signature	Description
31	NSW6	OC	Negative travel switch for axis 6
32	TSW6	OC	Thermal switch for axis 6
33	HSW7	OC	Homing switch for axis 7
34	PSW7	OC	Positive Travel switch for axis 7
35	NSW7	OC	Negative travel switch for axis 7
36	TSW7	OC	Thermal switch for axis 7
37	HSW8	OC	Homing switch for axis 8
38	PSW8	OC	Positive Travel switch for axis 8
39	NSW8	OC	Negative travel switch for axis 8
40	TSW8	OC	Thermal switch for axis 8
41	TPESop -	Swith	Teach pendant E-Stop pair
42	TPESop-		Teach pendant E-Stop pair
43	LiveMan -	Swith	Teach pendant liveman switch pair
44	LiveMan-		Teach pendant E-Stop pair
45	Brake -	Swith	Brake relay source
46	Brake-		Switch Brake relay return (normally open)
47	ArmOn -	Swith	Arm Power relay source
48	ArmOn-	Swith	Arm Power relay return (normally open)
49	AnalogIn1		
50	AnalogIn2		
51	AEEStop -	Switch	Auxiliary E-Stop switch pair. Used for exp Amp connector. Can be by passed on the auxiliary board with a switch.
52	AEEStop-		
53	N/C		
54	N/C		
55	N/C		
56	N/C		
57	N/C		
58	N/C		
59	N/C		
60	N/C		

Tabla A.3: Asignación de pines del conector J6 de la tarjeta CB del controlador C500, continuación.

Para más información acerca del controlador C500 consulte CRS (1995). Para obtener información más detallada acerca del robot A465 puede consultar CRS (1997a) y CRS (2001b). En el caso del robot A255 puede consultar CRS (1997b) y CRS (2001a).

TESIS CON
FALLA DE ORIGEN

Pin#	Function	Signature	Description
1	A1+	RS-422	Axis 1 Channel A input
2	A1-	RS-422	Axis 1 Channel A input (complementary)
3	B1+	RS-422	Axis 1 Channel B input
4	B1-	RS-422	Axis 1 Channel B input (complementary)
5	Z1-	RS-422	Axis 1 Channel Z input
6	Z1-	RS-422	Axis 1 Channel Z input (complementary)
7	A2-	RS-422	Axis 2 Channel A input
8	A2-	RS-422	Axis 2 Channel A input (complementary)
9	B2-	RS-422	Axis 2 Channel B input
10	B2-	RS-422	Axis 2 Channel B input (complementary)
11	Z2+	RS-422	Axis 2 Channel Z input
12	Z2-	RS-422	Axis 2 Channel Z input (complementary)
13	A3+	RS-422	Axis 3 Channel A input
14	A3-	RS-422	Axis 3 Channel A input (complementary)
15	B3+	RS-422	Axis 3 Channel B input
16	B3-	RS-422	Axis 3 Channel B input (complementary)
17	Z3+	RS-422	Axis 3 Channel Z input
18	Z3-	RS-422	Axis 3 Channel Z input (complementary)
19	A4+	RS-422	Axis 4 Channel A input
20	A4-	RS-422	Axis 4 Channel A input (complementary)
21	B4+	RS-422	Axis 4 Channel B input
22	B4-	RS-422	Axis 4 Channel B input (complementary)
23	Z4+	RS-422	Axis 4 Channel Z input
24	Z4-	RS-422	Axis 4 Channel Z input (complementary)
25	A5-	RS-422	Axis 5 Channel A input
26	A5-	RS-422	Axis 5 Channel A Input (complementary)
27	B5-	RS-422	Axis 5 Channel B input
28	B5-	RS-422	Axis 5 Channel B input (complementary)
29	Z5-	RS-422	Axis 5 Channel Z input
30	Z5-	RS-422	Axis 5 Channel Z input (complementary)
31	A6+	RS-422	Axis 6 Channel A input
32	A6-	RS-422	Axis 6 Channel A Input (complementary)
33	B6+	RS-422	Axis 6 Channel B input
34	B6-	RS-422	Axis 6 Channel B input (complementary)
35	Z6-	RS-422	Axis 6 Channel Z input
36	Z6-	RS-422	Axis 6 Channel Z input (complementary)
37	NC	N/A	No Connect
38	Shield		No Connect
39			
40			
41	SGPos	Analog	Servo gripper position
42	SGTor	Analog	Servo gripper torque (not used)
43	AirGrip-	Power	Solenoid return
44	Gnd	Power	
45			
46	Gnd	Power	
47	+12 V	Power	+12 Volt supply to the servo gripper / Air Gripper Solenoid
48	Gnd	Power	
49	+12 V	Power	+12 Volt supply to the servo gripper / Air Gripper Solenoid
50	Gnd	Power	

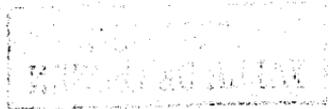
Tabla A.4: Asignación de pines del conector J7 de la tarjeta CB del controlador C500.

TESIS CON
FALLA DE ORIGEN

Rendimiento	
<ul style="list-style-type: none"> - Una tasa de actualización de 70 µs por eje - Tasa de actualización de instrucciones cada 10 ms - Búsqueda de entradas y salidas cada 20 ms - Actualización del servo a 1 kHz - Alcance de posición de 2²⁴ (número de pulsos por revolución contados por el contador) - Alcance de la velocidad a 2²¹ (pulsos por segundo) - Control del amplitudismo del motor de ±10 VDC - Resolución de 12 bits para comando de los motores - Entradas diferenciales o analógicas - Realiza hasta 30 tareas concurrentes 	
Hardware	
Procesador	- Intel 80286 80287
Transmisor	- Frecuencia 20 MHz T400 - Frecuencia 30 MHz T405 (opcional) - 4 lugares sobre el tablero de circuitos para TRAM - Acceso externo para TRAM net
Eje	- Soporte completo de 4 ejes - Soporte de servo manipulador
Entradas y Salidas I/O	- 16 entradas ópticamente aisladas - 4 salidas relé de contacto (24 V at 3 A) - 12 salidas ópticamente aisladas (150 mA at 24 V) - Entrada/Salida I/O expansibles a través de Interfaces PLC
Memoria disponible para el usuario	- 256 kilobytes (kbytes) de memoria de acceso al azar (RAM) respaldada por batería - 512 kilobytes (kbytes) de memoria FLASH para reserva no volátil (opcional)
Comunicación	- Puertos RS232C de datos estándar 38.4 k-baud (baudios) con un máximo de 78.4 k-baudios - Protocolos de comunicaciones con detección de errores ACI
Temperatura	- De +10 a +40 centígrados
Energía	- 100-115-230 VAC 60/50 Hz (seleccionable), 350 VA (A255 y controlador de movimiento), 900 VA (A405 y robot de función correcta (gantry))
Estándares	- Diseñado para cumplir con los requisitos estándares de seguridad para robots de I LITTO ANSI/RIA 15.06 - Inmutabilidad EN50062-2 - Emisiones EN50061-2
Software	
Programación	- Lenguaje de programación RAPT-II y sistema operativo de multitarea - Robocon-II - Herramientas de desarrollo para redes de transmisorres
Tipo de robot	- Articulado y de función correcta (gantry)
Coordenadas	- Motor, articulación, minuto y marcas de referencia remota
Tipo de control	- PID (proporcional-integral-derivativo)
Método de velocidad	- Trapezoidal o parabólica
Programación por aprendizaje	- fuerza de línea - comando de operación
Tipo de trayectoria	- spline - spline interpolada (de punto a punto) - línea recta - vía constante - movimiento relativo - movimiento combinado - interpolación cúbica
Consola de operación	
Configuración	- pantalla color, tamaño de 3 metros de largo
Panel de operación	- 4 líneas a 20 caracteres (LCD)
Señales	- diseñado para cumplir con los requisitos estándar para consolas de operación de ANSI/RIA 15.02 - interruptor de activación - parada de emergencia (asegurada)
Características físicas	
Peso	- 60 libras (31 kilogramos)
Tamaño del controlador	- 19 pulgadas (483 mm) de ancho x 10.5 pulgadas (267 mm) de largo - 1.75 pulgadas (44.8 mm) de espesor - 19 pulgadas instalable en armazón

Tabla A.5: Características del controlador C500.

**TESIS CON
FALLA DE ORIGEN**



Apéndice B

Especificaciones de la tarjeta FlexMotion 6C

Este apéndice lista las especificaciones de funcionamiento del hardware y software para los controladores FlexMotion 6C.

Funcionamiento Servo

Rango de velocidad de actualización del PID	62.5 a 500 μ s/muestra
Máxima velocidad de actualización del PID	62.5 μ s / eje
Velocidad de actualización del PID en los 6 ejes	375 μ s total
Velocidad de actualización de trayectoria	62.5 μ /eje
Sincronización multiejes	<1 muestra
Exactitud en posición	
Retroalimentación del encoder	± 1 cuenta de cuadratura
Retroalimentación analógica	± 1 LSB
Parámetros de trayectoria	
Rango de posición absoluta	$\pm 2^{31}$ cuentas
Máximo tamaño de movimiento relativo	$\pm 2^{31}$ cuentas
Rango de velocidad	1 a $\pm 16,000,000$ cuentas / s
Rango de RPM	10^{-5} a 10^6 revoluciones / min
Aceleración/desaceleración	4,000 a 128,000,000 cuentas / s ²
Rango de RPS/s	10^{-1} a 10^8 revoluciones / s ²
Rango de seguimiento de error	0 a 32,767 cuentas
Modos del lazo de servocontrol	PID, PIVff, S-Curve, doble lazo
Ganancias del PID (Kp, Ki y Kd)	0 a 32,767
Límite de integración (Ilim)	0 a 32,767
Período de muestreo derivativo(Td)	1 a 63 muestras
Ganancias de la alimentación adelantada (Aff, Vff)	0 a 32,767

TESIS CON
FALLA DE ORIGEN

Ganancia de retroalimentación de velocidad (Kv)	0 a 32,767
Salidas analógicas de los servo comandos	
Rango de voltaje	± 10 V
Resolución	16 bits (0.000305 V / LSB)
Límites del torque programable	
Límite positivo	± 10 V (-32,768 a +32,768)
Límite negativo	± 10 V (-32,768 a +32,768)
Offset programable	± 10 V (-32,768 a +32,768)
Funcionamiento a pasos	
Rango de velocidad de actualización de trayectoria	125 a 500 μ s/muestra
Máxima velocidad de actualización	125 μ s/eje
Sincronización multieje	<1 muestra
Exactitud de posición	
Escalonamiento en lazo abierto	1 completo, medio o micro paso
Retroalimentación del encoder	± 1 cuenta de cuadratura
Retroalimentación analógica	± 1 LSB
Parámetros de trayectoria	
Rango de posición	$\pm 2^{31}$ pasos
Máximo tamaño de movimiento relativo	$\pm 2^{31}$ cuentas
Rango de velocidad	1 a $\pm 1,500,000$ paso / s
Rango de RPM	10^{-5} a 10^6 revoluciones / min
Aceleración / desaceleración	4,000 a 128,000,000 pasos / s ²
Rango de RPS / s	10^{-1} a 10^8 revoluciones / s ²
Rango de seguimiento de error	0 a 32,767 cuentas
Salidas de escalonamiento	
Velocidad máxima del pulso	1.5 MHz (completo, medio, micro paso)
Ancho de pulso mínimo	300 ns a 1.5 MHz
Modo de salida del paso	paso y dirección o CW/CCW
Salidas de escalonamiento	TTL con colector abierto
Rango de voltaje	0 a 5 V
Salida de voltaje en bajo	<0.6 V a 64mA
Polaridad	Programable, activo en alto o activo en bajo

Sistema de seguridad

Función del releador	
Tiempo de respuesta	Restablece el controlador al estado de inicio 63 ms
Entrada E-Stop (paro de emergencia)	
Rango de voltaje	Optoacoplada 0 a 24 V
Optoacoplador	TLP626 o equivalente

Corriente máxima de entrada
Control

10 mA
Deshabilita todos los ejes y comandos de salida

Movimiento de entrada/salida

Entrada de encoders

Máxima velocidad de cuenta

Encoder 1 y 2

Encoder 3 y 4

Encoder 5 y 6

Opciones de las señales de encoder

Voltaje de umbral de la entrada

Rango de voltaje

Ancho mínimo del pulso de Index

Entradas adelante, reversa y home

Número de entradas

Rango de voltaje

Optoacoplador

Corriente máxima de entrada

Polaridad

Entradas de señal de disparo

Número de entradas

Rango de voltaje

Optoacoplador

Corriente máxima de entrada

Polaridad

Ancho de pulso mínimo

Salidas breakpoint

Número de salidas

Rango de voltaje

Optoacoplador

Polaridad

Salidas habilitar/deshabilitar

Número de salidas

Rango de voltaje

Polaridad

Entradas analógicas

Número de entradas

Rango de voltaje

Resistencia de entrada

Resolución

Salidas analógicas

Cuadratura

16 MHz

1 MHz

2 MHz

Simple o complementada: A, \bar{A} , B,

\bar{B} , Index, $\bar{\text{Index}}$ por eje

± 0.3 V (típico)

0 a 5 V

83 ns

Optoacopladas

18 (3 por eje)

0 a 24 V

TLP626 o equivalente

10 mA

Programable, activo en alto o activo en bajo

Optoacopladas

4 (del encoder 1 al 4)

0 a 24 V

TLP2631 o equivalente

10 mA

Programable, activo en alto o activo en bajo

83 ns

Optoacopladas

4 (del encoder 1 al 4)

0 a 24 V

TLP627 o equivalente

Programable, activo en alto o activo en bajo

Colector abierto

6 (1 por eje)

0 a 12 V

Programable, activo en alto o activo en bajo

8 (multiplexadas)

± 10 V

20 KW min

12 bit (0.0049 V/LSB)

Número de salidas	6
Rango de voltaje	± 10 V
Corriente de salida	± 10 mA
Resolución	16 bits (0.000305 V/LSB)

E/S digitales de 24 bits

Puertos de 8 bits

Máximo número de puertos de entrada	2
Máximo número de puertos de salida	3

Entradas

Rango de voltaje	0 a 5 V
Voltaje bajo de entrada	0.8 V
Voltaje alto de entrada	2.0 V
Polaridad	Programable, activo en alto o activo en bajo

Salidas

Rango de voltaje	0 a 5 V
Voltaje bajo de entrada	<0.5 V a 24 mA
Voltaje alto de entrada	>2.4 V a 3 mA
Polaridad	Programable, activo en alto o activo en bajo

Salidas PWM

Número de salidas PWM	2
Frecuencia máxima PWM	32 KHz
Resolución	8 bits
Rango del ciclo de trabajo	0 a (255/256)%
Fuentes de reloj	Entrada de contador externa o interna

Requerimientos de energía (Máximos)

-5V ($\pm 3\%$)	2.0 A
-12V ($\pm 3\%$)	150 mA
-12V ($\pm 3\%$)	200 mA
Consumo de energía	14.2 W

Dimensiones (sin incluir conector)

PCI-FlexMotion 6C	33.8 x 9.9 cm (13.3 x 3.9 in)
-------------------	-------------------------------

Ambiente

Temperatura de operación	0 a 55 °C
Temperatura de almacenamiento	-20 a 70°C
Rango de humedad relativa	10 a 90% (sin condensación)

La Figura B.1 muestra la asignación de pines para el conector de 100 pines de E/S de movimiento para la tarjeta PCI-FlexMotion-6C. La línea arriba del nombre del pin indica que la señal se activa en bajo.

La Figura B.2 muestra la asignación de pines del conector auxiliar. Por default el puerto de 24 bits está dividido en 3 puertos digitales de E/S de 8 bits cada uno. Cada circuito de salida puede recibir 24mA y proporcionar 3 mA. Para más información sobre la configuración de la tarjeta FlexMotion consulte National Instruments (1999b)

(M)	(M)	(M)
Axis 1 Forward Limit Input	2	32
Axis 1 Home Input	3	33
Axis 1 Reverse Limit Input	4	34
Axis 1 Trigger Input 1	5	35
Axis 1 Home Input	6	36
Axis 1 Home Input	7	37
Axis 2 Forward Limit Input	8	38
Axis 2 Trigger Input 1	9	39
Axis 2 Reverse Limit Input	10	40
Axis 3 Forward Limit Input	11	41
Axis 3 Home Input	12	42
Axis 3 Reverse Limit Input	13	43
Axis 4 Forward Limit Input	14	44
Axis 4 Reverse Limit Input	15	45
Axis 6 Reverse Limit Input	16	46
Axis 6 Home Input	17	47
Encoder 1 Phase A	18	48
Encoder 1 Phase B	19	49
Encoder 1 Index	20	50
Encoder 2 Phase A	21	51
Encoder 2 Phase B	22	52
Encoder 2 Index	23	53
Encoder 3 Phase A	24	54
Encoder 3 Phase B	25	55
Encoder 3 Index	26	56
Encoder 4 Phase A	27	57
Encoder 4 Phase B	28	58
Encoder 4 Index	29	59
Encoder 5 Phase A	30	60
Encoder 5 Phase B	31	61
Encoder 5 Index	32	62
Encoder 6 Phase A	33	63
Encoder 6 Phase B	34	64
Encoder 6 Index	35	65
Encoder 7 Phase A	36	66
Encoder 7 Phase B	37	67
Encoder 7 Index	38	68
Encoder 8 Phase A	39	69
Encoder 8 Phase B	40	70
Encoder 8 Index	41	71
Encoder 9 Phase A	42	72
Encoder 9 Phase B	43	73
Encoder 9 Index	44	74
Encoder 10 Phase A	45	75
Encoder 10 Phase B	46	76
Encoder 10 Index	47	77
Encoder 11 Phase A	48	78
Encoder 11 Phase B	49	79
Encoder 11 Index	50	80
Encoder 12 Phase A	51	81
Encoder 12 Phase B	52	82
Encoder 12 Index	53	83
Encoder 13 Phase A	54	84
Encoder 13 Phase B	55	85
Encoder 13 Index	56	86
Encoder 14 Phase A	57	87
Encoder 14 Phase B	58	88
Encoder 14 Index	59	89
Encoder 15 Phase A	60	90
Encoder 15 Phase B	61	91
Encoder 15 Index	62	92
Encoder 16 Phase A	63	93
Encoder 16 Phase B	64	94
Encoder 16 Index	65	95
Encoder 17 Phase A	66	96
Encoder 17 Phase B	67	97
Encoder 17 Index	68	98
Encoder 18 Phase A	69	99
Encoder 18 Phase B	70	100
Encoder 18 Index	71	101
Encoder 19 Phase A	72	102
Encoder 19 Phase B	73	103
Encoder 19 Index	74	104
Encoder 20 Phase A	75	105
Encoder 20 Phase B	76	106
Encoder 20 Index	77	107
Encoder 21 Phase A	78	108
Encoder 21 Phase B	79	109
Encoder 21 Index	80	110
Encoder 22 Phase A	81	111
Encoder 22 Phase B	82	112
Encoder 22 Index	83	113
Encoder 23 Phase A	84	114
Encoder 23 Phase B	85	115
Encoder 23 Index	86	116
Encoder 24 Phase A	87	117
Encoder 24 Phase B	88	118
Encoder 24 Index	89	119
Encoder 25 Phase A	90	120
Encoder 25 Phase B	91	121
Encoder 25 Index	92	122
Encoder 26 Phase A	93	123
Encoder 26 Phase B	94	124
Encoder 26 Index	95	125
Encoder 27 Phase A	96	126
Encoder 27 Phase B	97	127
Encoder 27 Index	98	128
Encoder 28 Phase A	99	129
Encoder 28 Phase B	100	130
Encoder 28 Index	101	131
Encoder 29 Phase A	102	132
Encoder 29 Phase B	103	133
Encoder 29 Index	104	134
Encoder 30 Phase A	105	135
Encoder 30 Phase B	106	136
Encoder 30 Index	107	137
Encoder 31 Phase A	108	138
Encoder 31 Phase B	109	139
Encoder 31 Index	110	140
Encoder 32 Phase A	111	141
Encoder 32 Phase B	112	142
Encoder 32 Index	113	143
Encoder 33 Phase A	114	144
Encoder 33 Phase B	115	145
Encoder 33 Index	116	146
Encoder 34 Phase A	117	147
Encoder 34 Phase B	118	148
Encoder 34 Index	119	149
Encoder 35 Phase A	120	150
Encoder 35 Phase B	121	151
Encoder 35 Index	122	152
Encoder 36 Phase A	123	153
Encoder 36 Phase B	124	154
Encoder 36 Index	125	155
Encoder 37 Phase A	126	156
Encoder 37 Phase B	127	157
Encoder 37 Index	128	158
Encoder 38 Phase A	129	159
Encoder 38 Phase B	130	160
Encoder 38 Index	131	161
Encoder 39 Phase A	132	162
Encoder 39 Phase B	133	163
Encoder 39 Index	134	164
Encoder 40 Phase A	135	165
Encoder 40 Phase B	136	166
Encoder 40 Index	137	167
Encoder 41 Phase A	138	168
Encoder 41 Phase B	139	169
Encoder 41 Index	140	170
Encoder 42 Phase A	141	171
Encoder 42 Phase B	142	172
Encoder 42 Index	143	173
Encoder 43 Phase A	144	174
Encoder 43 Phase B	145	175
Encoder 43 Index	146	176
Encoder 44 Phase A	147	177
Encoder 44 Phase B	148	178
Encoder 44 Index	149	179
Encoder 45 Phase A	150	180
Encoder 45 Phase B	151	181
Encoder 45 Index	152	182
Encoder 46 Phase A	153	183
Encoder 46 Phase B	154	184
Encoder 46 Index	155	185
Encoder 47 Phase A	156	186
Encoder 47 Phase B	157	187
Encoder 47 Index	158	188
Encoder 48 Phase A	159	189
Encoder 48 Phase B	160	190
Encoder 48 Index	161	191
Encoder 49 Phase A	162	192
Encoder 49 Phase B	163	193
Encoder 49 Index	164	194
Encoder 50 Phase A	165	195
Encoder 50 Phase B	166	196
Encoder 50 Index	167	197
Encoder 51 Phase A	168	198
Encoder 51 Phase B	169	199
Encoder 51 Index	170	200
Encoder 52 Phase A	171	201
Encoder 52 Phase B	172	202
Encoder 52 Index	173	203
Encoder 53 Phase A	174	204
Encoder 53 Phase B	175	205
Encoder 53 Index	176	206
Encoder 54 Phase A	177	207
Encoder 54 Phase B	178	208
Encoder 54 Index	179	209
Encoder 55 Phase A	180	210
Encoder 55 Phase B	181	211
Encoder 55 Index	182	212
Encoder 56 Phase A	183	213
Encoder 56 Phase B	184	214
Encoder 56 Index	185	215
Encoder 57 Phase A	186	216
Encoder 57 Phase B	187	217
Encoder 57 Index	188	218
Encoder 58 Phase A	189	219
Encoder 58 Phase B	190	220
Encoder 58 Index	191	221
Encoder 59 Phase A	192	222
Encoder 59 Phase B	193	223
Encoder 59 Index	194	224
Encoder 60 Phase A	195	225
Encoder 60 Phase B	196	226
Encoder 60 Index	197	227
Encoder 61 Phase A	198	228
Encoder 61 Phase B	199	229
Encoder 61 Index	200	230
Encoder 62 Phase A	201	231
Encoder 62 Phase B	202	232
Encoder 62 Index	203	233
Encoder 63 Phase A	204	234
Encoder 63 Phase B	205	235
Encoder 63 Index	206	236
Encoder 64 Phase A	207	237
Encoder 64 Phase B	208	238
Encoder 64 Index	209	239
Encoder 65 Phase A	210	240
Encoder 65 Phase B	211	241
Encoder 65 Index	212	242
Encoder 66 Phase A	213	243
Encoder 66 Phase B	214	244
Encoder 66 Index	215	245
Encoder 67 Phase A	216	246
Encoder 67 Phase B	217	247
Encoder 67 Index	218	248
Encoder 68 Phase A	219	249
Encoder 68 Phase B	220	250
Encoder 68 Index	221	251
Encoder 69 Phase A	222	252
Encoder 69 Phase B	223	253
Encoder 69 Index	224	254
Encoder 70 Phase A	225	255
Encoder 70 Phase B	226	256
Encoder 70 Index	227	257
Encoder 71 Phase A	228	258
Encoder 71 Phase B	229	259
Encoder 71 Index	230	260
Encoder 72 Phase A	231	261
Encoder 72 Phase B	232	262
Encoder 72 Index	233	263
Encoder 73 Phase A	234	264
Encoder 73 Phase B	235	265
Encoder 73 Index	236	266
Encoder 74 Phase A	237	267
Encoder 74 Phase B	238	268
Encoder 74 Index	239	269
Encoder 75 Phase A	240	270
Encoder 75 Phase B	241	271
Encoder 75 Index	242	272
Encoder 76 Phase A	243	273
Encoder 76 Phase B	244	274
Encoder 76 Index	245	275
Encoder 77 Phase A	246	276
Encoder 77 Phase B	247	277
Encoder 77 Index	248	278
Encoder 78 Phase A	249	279
Encoder 78 Phase B	250	280
Encoder 78 Index	251	281
Encoder 79 Phase A	252	282
Encoder 79 Phase B	253	283
Encoder 79 Index	254	284
Encoder 80 Phase A	255	285
Encoder 80 Phase B	256	286
Encoder 80 Index	257	287
Encoder 81 Phase A	258	288
Encoder 81 Phase B	259	289
Encoder 81 Index	260	290
Encoder 82 Phase A	261	291
Encoder 82 Phase B	262	292
Encoder 82 Index	263	293
Encoder 83 Phase A	264	294
Encoder 83 Phase B	265	295
Encoder 83 Index	266	296
Encoder 84 Phase A	267	297
Encoder 84 Phase B	268	298
Encoder 84 Index	269	299
Encoder 85 Phase A	270	300
Encoder 85 Phase B	271	301
Encoder 85 Index	272	302
Encoder 86 Phase A	273	303
Encoder 86 Phase B	274	304
Encoder 86 Index	275	305
Encoder 87 Phase A	276	306
Encoder 87 Phase B	277	307
Encoder 87 Index	278	308
Encoder 88 Phase A	279	309
Encoder 88 Phase B	280	310
Encoder 88 Index	281	311
Encoder 89 Phase A	282	312
Encoder 89 Phase B	283	313
Encoder 89 Index	284	314
Encoder 90 Phase A	285	315
Encoder 90 Phase B	286	316
Encoder 90 Index	287	317
Encoder 91 Phase A	288	318
Encoder 91 Phase B	289	319
Encoder 91 Index	290	320
Encoder 92 Phase A	291	321
Encoder 92 Phase B	292	322
Encoder 92 Index	293	323
Encoder 93 Phase A	294	324
Encoder 93 Phase B	295	325
Encoder 93 Index	296	326
Encoder 94 Phase A	297	327
Encoder 94 Phase B	298	328
Encoder 94 Index	299	329
Encoder 95 Phase A	300	330
Encoder 95 Phase B	301	331
Encoder 95 Index	302	332
Encoder 96 Phase A	303	333
Encoder 96 Phase B	304	334
Encoder 96 Index	305	335
Encoder 97 Phase A	306	336
Encoder 97 Phase B	307	337
Encoder 97 Index	308	338
Encoder 98 Phase A	309	339
Encoder 98 Phase B	310	340
Encoder 98 Index	311	341
Encoder 99 Phase A	312	342
Encoder 99 Phase B	313	343
Encoder 99 Index	314	344
Encoder 100 Phase A	315	345
Encoder 100 Phase B	316	346
Encoder 100 Index	317	347
Encoder 101 Phase A	318	348
Encoder 101 Phase B	319	349
Encoder 101 Index	320	350
Encoder 102 Phase A	321	351
Encoder 102 Phase B	322	352
Encoder 102 Index	323	353
Encoder 103 Phase A	324	354
Encoder 103 Phase B	325	355
Encoder 103 Index	326	356
Encoder 104 Phase A	327	357
Encoder 104 Phase B	328	358
Encoder 104 Index	329	359
Encoder 105 Phase A	330	360
Encoder 105 Phase B	331	361
Encoder 105 Index	332	362
Encoder 106 Phase A	333	363
Encoder 106 Phase B	334	364
Encoder 106 Index	335	365
Encoder 107 Phase A	336	366
Encoder 107 Phase B	337	367
Encoder 107 Index	338	368
Encoder 108 Phase A	339	369
Encoder 108 Phase B	340	370
Encoder 108 Index	341	371
Encoder 109 Phase A	342	372
Encoder 109 Phase B	343	373
Encoder 109 Index	344	374
Encoder 110 Phase A	345	375
Encoder 110 Phase B	346	376
Encoder 110 Index	347	377
Encoder 111 Phase A	348	378
Encoder 111 Phase B	349	379
Encoder 111 Index	350	380
Encoder 112 Phase A	351	381
Encoder 112 Phase B	352	382
Encoder 112 Index	353	383
Encoder 113 Phase A	354	384
Encoder 113 Phase B	355	385
Encoder 113 Index	356	386
Encoder 114 Phase A	357	387
Encoder 114 Phase B	358	388
Encoder 114 Index	359	389
Encoder 115 Phase A	360	390
Encoder 115 Phase B	361	391
Encoder 115 Index	362	392
Encoder 116 Phase A	363	393
Encoder 116 Phase B	364	394
Encoder 116 Index	365	395
Encoder 117 Phase A	366	396
Encoder 117 Phase B	367	397
Encoder 117 Index	368	398
Encoder 118 Phase A	369	399
Encoder 118 Phase B	370	400
Encoder 118 Index	371	401
Encoder 119 Phase A	372	402
Encoder 119 Phase B	373	403
Encoder 119 Index	374	404
Encoder 120 Phase A	375	405
Encoder 120 Phase B	376	406
Encoder 120 Index	377	407
Encoder 121 Phase A	378	408
Encoder 121 Phase B	379	409
Encoder 121 Index	380	410
Encoder 122 Phase A	381	411
Encoder 122 Phase B	382	412
Encoder 122 Index	383	413
Encoder 123 Phase A	384	414
Encoder 123 Phase B	385	415
Encoder 123 Index	386	416

Nombre de la señal	Referencia	Dirección	Descripción
Analog Output < 1..6 >	Tierra de salida analógica	Salida	Comando analógico del motor
Analog Output Ground	-	-	Referencia para Vout
Stepper < 5..6 > step (CW)	Tierra digital	Salida	Escalonador solamente - paso del motor o control en dirección de las manecillas del reloj
Stepper < 5..6 > dir (CCW)	Tierra digital	Salida	Escalonador solamente - dirección del motor o control en dirección contrario de las manecillas del reloj
Axis < 1..6 > Inhibit	Tierra digital	Salida	Inhabilitación del amplificador/impulsor
Encoder < 1..6 > phase A	Tierra digital	Entrada	Entrada de la fase A del encoder
Encoder < 1..6 > phase A	Tierra digital	Entrada	Entrada de la fase A complementada del encoder
Encoder < 1..6 > phase B	Tierra digital	Entrada	Entrada de la fase B del encoder
Encoder < 1..6 > phase B	Tierra digital	Entrada	Entrada de la fase B complementada del encoder
Encoder < 1..6 > Index	Tierra digital	Entrada	Entrada de índice o marcador
Encoder < 1..6 > Index	Tierra digital	Entrada	Entrada de índice o marcador complementado
Digital Ground	-	-	Referencia para las entradas/salidas digitales
Host +5V	Tierra digital	Salida	Bus de la computadora de 5 Volts
-ISO	ISORTN	Entrada	Fuente de voltaje aislada
Trigger Input < 1..4 >	ISORTN	Entrada	Entrada de la señal de disparo
Breakpoint Output < 1..4 >	ISORTN	Salida	Salida del punto de ruptura
Axis < 1..6 > forward limit input	ISORTN	Entrada	Señal de entrada del límite hacia adelante de los ejes
Axis < 1..6 > home input	ISORTN	Entrada	Señal de entrada de home de los ejes
Axis < 1..6 > reverse limit input	ISORTN	Entrada	Señal de entrada del límite hacia atrás de los ejes
E stop	ISORTN	Entrada	Entrada del paro de emergencia
ISORTN	-	-	Referencia para señales aisladas
Analog Reference	Analog Input Ground	Salida	7.5 V (nom) de salida
Analog Input < 1..8 >	Analog Input Ground	Entrada	Entradas del ADC de 12 bits
Analog Input Ground	-	-	Referencia para entradas analógicas

Tabla B.1: Asignación de los 100 pines del conector de E/S de movimiento.

TESIS CON
FALLA DE ORIGEN

Port 3 bit 7	1	2	Digital Ground
Port 3 bit 6	3	4	Digital Ground
Port 3 bit 5	5	6	Digital Ground
Port 3 bit 4	7	8	Digital Ground
Port 3 bit 3	9	10	Digital Ground
Port 3 bit 2	11	12	Digital Ground
Port 3 bit 1	13	14	Digital Ground
Port 3 bit 0	15	16	Digital Ground
Port 2 bit 7	17	18	Digital Ground
Port 2 bit 6	19	20	Digital Ground
Port 2 bit 5	21	22	Digital Ground
Port 2 bit 4	23	24	Digital Ground
Port 2 bit 3	25	26	Digital Ground
Port 2 bit 2	27	28	Digital Ground
Port 2 bit 1	29	30	Digital Ground
Port 2 bit 0	31	32	Digital Ground
Port 1 bit 7	33	34	Digital Ground
Port 1 bit 6	35	36	Digital Ground
Port 1 bit 5	37	38	Digital Ground
Port 1 bit 4	39	40	Digital Ground
Port 1 bit 3	41	42	Digital Ground
Port 1 bit 2	43	44	Digital Ground
Port 1 bit 1	45	46	Digital Ground
Port 1 bit 0	47	48	Digital Ground
-5 V	49	50	Digital Ground

Figura B.2: Asignación de los 50 pines del conector auxiliar de E/S digital de 24 bits.

TESIS CON
FALLA DE ORIGEN



Apéndice C

Especificaciones de la UMI-Flex 6

Las siguientes especificaciones son para la UMI-Flex 6, a una temperatura de 25°C. Para más información consulte National Instruments (1999c).

Inhabilitación de ejes

Rango de voltaje	0 a 5 VDC
Corriente de salida	16 mA

Ambiente de operación

Temperatura	0 a 55 °C
Temperatura de almacenamiento	-20 a 70 °C
Humedad relativa	10 a 90% (sin condensación)

Requerimientos de energía

+5 VDC	0.2 A
+ISO	
Voltaje	5 a 24 VDC
Corriente	Definida por las señales conectadas

Físicas

Dimensiones	26.67 x 11.18 cm (10.5 x 4.4 in)
-------------	----------------------------------

TESIS CON
FALLA DE ORIGEN



Apéndice D

Conectores del sistema adaptado

Este apéndice muestra los conectores del sistema adaptado. Las Tablas D.1, D.2, D.3 y D.4 enlistan la asignación de pines para los conectores del controlador C500 adaptado. Por otro lado, las Tablas D.5, D.6, D.7, D.8, D.9 y D.10 muestran la asignación de pines para los conectores del Chasis UMI.



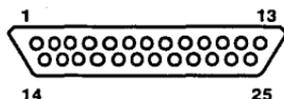


Figura D.1: Conector DB-25[I] del controlador C500.

CONECTOR C500DB25[I]			
No. de PIN	Función	Dirección	Descripción
1	Vcom1	Entrada	Command voltage for axis 1
2	Vcom2	Entrada	Command voltage for axis 2
3	Vcom3	Entrada	Command voltage for axis 3
4	Vcom4	Entrada	Command voltage for axis 4
5	HSW1	Salida	Homing switch for axis 1
6	NSW1	Salida	Negative travel switch for axis 1
7	HSW2	Salida	Homing switch for axis 2
8	NSW2	Salida	Negative travel switch for axis 2
9	HSW3	Salida	Homing switch for axis 3
10	NSW3	Salida	Negative travel switch for axis 3
11	HSW4	Salida	Homing switch for axis 4
12	NSW4	Salida	Negative travel switch for axis 4
13	HSW5	Salida	Homing switch for axis 5
14	Vcom5	Entrada	Command voltage for axis 5
15	Vcom6	Entrada	Command voltage for axis 6
16	Vcom7	Entrada	Command voltage for axis 7
17	Vcom8	Entrada	Command voltage for axis 8
18	PSW1	Salida	Positive Travel switch for axis 1
19	TSW1	Salida	Thermal switch for axis 1
20	PSW2	Salida	Positive Travel switch for axis 2
21	TSW2	Salida	Thermal switch for axis 2
22	PSW3	Salida	Positive Travel switch for axis 3
23	TSW3	Salida	Thermal switch for axis 3
24	PSW4	Salida	Positive Travel switch for axis 4
25	TSW4	Salida	Thermal switch for axis 4

Tabla D.1: Asignación de pines del conector DB-25[I] del controlador C500 - C500DB25[I].

TESIS CON
FALLA DE ORIGEN

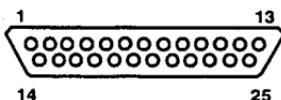


Figura D.2: Conector DB-25[II] del controlador C500.

CONECTOR C500DB25[II]			
No. de PIN	Función	Dirección	Descripción
1	PSW5	Salida	Positive Travel switch for axis 5
2	TSW5	Salida	Thermal switch for axis 5
3	PSW6	Salida	Positive Travel switch for axis 6
4	TSW6	Salida	Thermal switch for axis 6
5	TPEStop-		Teach pendant E-Stop pair
6	LiveMan-		Teach pendant E-Stop pair
7	Brake-		Switch brake relay return (normally open)
8	ArmOn-		Arm Power relay return (normally open)
9	AnalogIn2		
10	AEEStop+		
11	N.C.		
12	N.C.		
13	N.C.		
14	NSW5	Salida	Negative travel switch for axis 5
15	HSW6	Salida	Homing switch for axis 6
16	NSW6	Salida	Negative travel switch for axis 6
17	TPEStop+		Teach pendant E-Stop pair
18	LiveMan+		Teach pendant LiveMan switch pair
19	Break+	Salida	Break relay source
20	ArmOn+	Salida	Arm Power relay source
21	AnalogIn1		
22	AEEStop+		Auxiliary E-Stop Switch pair. Used for exp Amp connector. Can be by passed on the auxiliary board with a switch
23	N.C.		
24	N.C.		
25	N.C.		

Tabla D.2: Asignación de pines del conector DB-25[II] del controlador C500 - C500DB25[II].

TESIS CON
FALLA DE ORIGEN

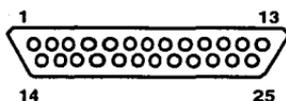


Figura D.3: Conector DB-25[1] del controlador C500.

CONECTOR C500DB25[1]			
No. de PIN	Función	Dirección	Descripción
1	A5-	Salida	Axis 5 Channel A input
2	Z4-	Salida	Axis 4 Channel Z input
3	B4-	Salida	Axis 4 Channel B input
4	A4-	Salida	Axis 4 Channel A input
5	Z3-	Salida	Axis 3 Channel Z Input
6	B3-	Salida	Axis 3 Channel B Input
7	A3-	Salida	Axis 3 Channel A Input
8	Z2-	Salida	Axis 2 Channel Z Input
9	B2-	Salida	Axis 2 Channel B Input
10	A2-	Salida	Axis 2 Channel A Input
11	Z1-	Salida	Axis 1 Channel Z Input
12	B1-	Salida	Axis 1 Channel B Input
13	A1-	Salida	Axis 1 Channel A Input
14	Z4-	Salida	Axis 4 Channel Z Input (complementary)
15	B4-	Salida	Axis 4 Channel B Input (complementary)
16	A4-	Salida	Axis 4 Channel A Input (complementary)
17	Z3-	Salida	Axis 3 Channel Z Input (complementary)
18	B3-	Salida	Axis 3 Channel B Input (complementary)
19	A3-	Salida	Axis 3 Channel A Input (complementary)
20	Z2-	Salida	Axis 2 Channel Z Input (complementary)
21	B2-	Salida	Axis 2 Channel B Input (complementary)
22	A2-	Salida	Axis 2 Channel A Input (complementary)
23	Z1-	Salida	Axis 1 Channel Z Input (complementary)
24	B1-	Salida	Axis 1 Channel B Input (complementary)
25	A1-	Salida	Axis 1 Channel A Input (complementary)

Tabla D.3: Asignación de pines del conector DB-25[1] del controlador C500 - C500DB25[1].

TESIS CON
FALLA DE ORIGEN

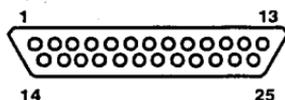


Figura D.4: Conector DB-25[2] del controlador C500.

CONECTOR C500DB25[2]			
No. de PIN	Función	Dirección	Descripción
1	GND		
2	GND		
3	GND		
4	GND		
5	SGTOR	Salida	Servo gripper torque (not used)
6			
7	Shield		No Connect
8	Z6-	Salida	Axis 6 Channel Z Input (complementary)
9	B6-	Salida	Axis 6 Channel B Input (complementary)
10	A6-	Salida	Axis 6 Channel A Input (complementary)
11	Z5-	Salida	Axis 5 Channel Z Input (complementary)
12	B5-	Salida	Axis 5 Channel B Input (complementary)
13	A5-	Salida	Axis 5 Channel A Input (complementary)
14	+12 V	Salida	+12 Volts supply to the servo gripper / Air gripper solenoid
15	-12 V	Salida	-12 Volts supply to the servo gripper / Air gripper solenoid
16			
17	AirGrip-		Solenoid return
18	SGPos	Salida	Servo gripper position
19			
20	N.C		No Connect
21	Z6+	Salida	Axis 6 Channel Z Input
22	B6+	Salida	Axis 6 Channel B Input
23	A6+	Salida	Axis 6 Channel A Input
24	Z5+	Salida	Axis 5 Channel Z Input
25	B5+	Salida	Axis 5 Channel B Input

Tabla D.4: Asignación de pines del conector DB-25[2] del controlador C500 - C500DB25[2].

EMIS CON
FALLA DE ORIGEN

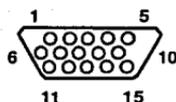


Figura D.5: Conector DB-15 del Chasis UMI.

CONECTOR UMIDB15		
PIN	Función	Descripción
1	Entrada al amplificador 1	Terminales de entrada del amplificador de la articulación 1 del robot
2	Entrada al amplificador 2	Terminales de entrada del amplificador de la articulación 2 del robot
3	Entrada al amplificador 3	Terminales de entrada del amplificador de la articulación 3 del robot
4	Entrada al amplificador 4	Terminales de entrada del amplificador de la articulación 4 del robot
5	Entrada al amplificador 5	Terminales de entrada del amplificador de la articulación 5 del robot
6	Salida Analógica 1	Comando de control para el eje 1 de la tarjeta PCI FlexMotion 6C
7	Salida Analógica 2	Comando de control para el eje 2 de la tarjeta PCI FlexMotion 6C
8	Salida Analógica 3	Comando de control para el eje 3 de la tarjeta PCI FlexMotion 6C
9	Salida Analógica 4	Comando de control para el eje 4 de la tarjeta PCI FlexMotion 6C
10	Salida Analógica 5	Comando de control para el eje 5 de la tarjeta PCI FlexMotion 6C
11	Entrada al amplificador 6	Terminales de entrada del amplificador de la articulación 6 del robot
12	Salida Analógica 6	Comando de control para el eje 6 de la tarjeta PCI FlexMotion 6C
13	NC	No se conecta
14	NC	No se conecta
15	GND	Tierra analógica

Tabla D.5: Asignación de pines del conector DB-15 del Chasis UMI – UMIDB15.

TESIS CON
FALLA DE ORIGEN

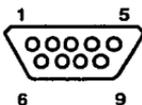


Figura D.6: Conector DB-9 del Chasis UMI.

CONECTOR UMIDB09		
PIN	Función	Descripción
1	Analog Input 1	Entrada de voltaje, terminal Analog Input 1 de la UMI
2	Analog Input 2	Entrada de voltaje, terminal Analog Input 2 de la UMI
3	Analog Input 3	Entrada de voltaje, terminal Analog Input 3 de la UMI
4	Analog Input 4	Entrada de voltaje, terminal Analog Input 4 de la UMI
5	Analog Input 5	Entrada de voltaje, terminal Analog Input 5 de la UMI
6	Analog Input 6	Entrada de voltaje, terminal Analog Input 6 de la UMI
7	Analog Input 7	Entrada de voltaje, terminal Analog Input 7 de la UMI
8	Analog Input 8	Entrada de voltaje, terminal Analog Input 8 de la UMI
9	GND	Tierra

Tabla D.6: Asignación de pines del conector DB-9 del Chasis UMI – UMIDB09.

TESIS CON
FALLA DE ORIGEN

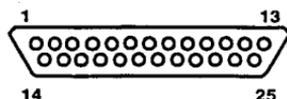


Figura D.7: Conector DB-25[1] del Chasis UMI.

CONECTOR UMIDB25[1]			
No. de PIN	Función	Dirección	Descripción
1	Phase 5A	Entrada	Encoder 5 Phase A
2	Index 4	Entrada	Encoder 4 Index
3	Phase 4B	Entrada	Encoder 4 Phase B
4	Phase 4A	Entrada	Encoder 4 Phase A
5	Index 3	Entrada	Encoder 3 Index
6	Phase 3B	Entrada	Encoder 3 Phase B
7	Phase 3A	Entrada	Encoder 3 Phase A
8	Index 2	Entrada	Encoder 2 Index
9	Phase 2B	Entrada	Encoder 2 Phase B
10	Phase 2A	Entrada	Encoder 2 Phase A
11	Index 1	Entrada	Encoder 1 Index
12	Phase 1B	Entrada	Encoder 1 Phase B
13	Phase 1A	Entrada	Encoder 1 Phase A
14	Index 4	Entrada	Encoder 4 Index
15	Phase 4B	Entrada	Encoder 4 Phase B
16	Phase 4A	Entrada	Encoder 4 Phase A
17	Index 3	Entrada	Encoder 3 Index
18	Phase 3B	Entrada	Encoder 3 Phase B
19	Phase 3A	Entrada	Encoder 3 Phase A
20	Index 2	Entrada	Encoder 2 Index
21	Phase 2B	Entrada	Encoder 2 Phase B
22	Phase 2A	Entrada	Encoder 2 Phase A
23	Index 1	Entrada	Encoder 1 Index
24	Phase 1B	Entrada	Encoder 1 Phase B
25	Phase 1A	Entrada	Encoder 1 Phase A

Tabla D.7: Asignación de pines del conector DB-25[1] del Chasis UMI - UMIDB25[1].

TESIS CON
FALLA DE ORIGEN

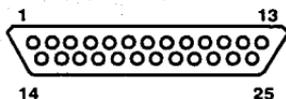


Figura D.8: Conector DB-25[2] del Chasis UMI.

CONECTOR UMIDB25[2]			
No. de PIN	Función	Dirección	Descripción
1	N/C		
2	N/C		
3	N/C		
4	N/C		
5	N/C		
6	N/C		
7	N/C		
8	Index 6	Entrada	Encoder 6 Index
9	Phase 6B	Entrada	Encoder 6 Phase B
10	Phase 6A	Entrada	Encoder 6 Phase A
11	Index 5	Entrada	Encoder 5 Index
12	Phase 5B	Entrada	Encoder 5 Phase B
13	Phase 5A	Entrada	Encoder 5 Phase A
14	N/C		
15	N/C		
16	N/C		
17	N/C		
18	N/C		
19	N/C		
20	N/C		
21	Index 6	Entrada	Encoder 6 Index
22	Phase 6B	Entrada	Encoder 6 Phase B
23	Phase 6A	Entrada	Encoder 6 Phase A
24	Index 5	Entrada	Encoder 5 Index
25	Phase 5B	Entrada	Encoder 5 Phase B

Tabla D.8: Asignación de pines del conector DB-25[2] del Chasis UMI - UMIDB25[2].

TESIS CON
FALLA DE ORIGEN

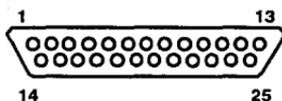


Figura D.9: Conector DB-25[I] del Chasis UMI.

CONECTOR UMIDB25[I]			
No. de PIN	Función	Dirección	Descripción
1	Analog Output 1	Salida	Analog motor command for axis 1
2	Analog Output 2	Salida	Analog motor command for axis 2
3	Analog Output 3	Salida	Analog motor command for axis 3
4	Analog Output 4	Salida	Analog motor command for axis 4
5	HOME1	Entrada	Home Input for axis 1
6	Reverse limit 1	Entrada	Reverse / CCW limit input for axis 1
7	HOME2	Entrada	Home Input for axis 2
8	Reverse limit 2	Entrada	Reverse / CCW limit input for axis 2
9	HOME3	Entrada	Home Input for axis 3
10	Reverse limit 3	Entrada	Reverse / CCW limit input for axis 3
11	HOME4	Entrada	Home Input for axis 4
12	Reverse limit 4	Entrada	Reverse / CCW limit input for axis 4
13	HOME5	Entrada	Home Input for axis 5
14	Analog Output 5	Salida	Analog motor command for axis 5
15	Analog Output 6	Salida	Analog motor command for axis 6
16	N/C		
17	N/C		
18	Forward limit 1	Entrada	Forward / CW limit input for axis 1
19	N/C		
20	Forward limit 2	Entrada	Forward / CW limit input for axis 2
21	N/C		
22	Forward limit 3	Entrada	Forward / CW limit input for axis 3
23	N/C		
24	Forward limit 4	Entrada	Forward / CW limit input for axis 4
25	N/C		

Tabla D.9: Asignación de pines del conector DB-25[I] del Chasis UMI - UMIDB25[I].

TESIS CON
 FALLA DE ORIGEN

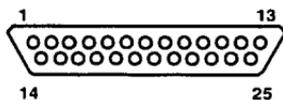


Figura D.10: Conector DB-25[II] del Chasis UMI.

CONECTOR UMIDB25[II]			
No. de PIN	Función	Dirección	Descripción
1	Forward limit 5	Entrada	Forward / CW limit input for axis 5
2	N.C.		
3	Forward limit 6	Entrada	Forward / CW limit input for axis 6
4	N.C.		
5	N.C.		
6	N.C.		
7	N.C.		
8	N.C.		
9	N.C.		
10	N.C.		
11	N.C.		
12	N.C.		
13	N.C.		
14	Reverse limit 5	Entrada	Reverse / CCW limit input for axis 5
15	HOME 6	Entrada	Home Input for axis 6
16	Reverse limit 6	Entrada	Reverse / CCW limit input for axis 6
17	N.C.		
18	N.C.		
19	N.C.		
20	N.C.		
21	N.C.		
22	N.C.		
23	N.C.		
24	N.C.		
25	N.C.		

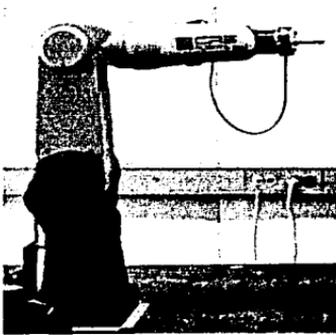
Tabla D.10: Asignación de pines del conector DB-25[II] del Chasis UMI - UMIDB25[II].

TESIS CON
FALLA DE ORIGEN

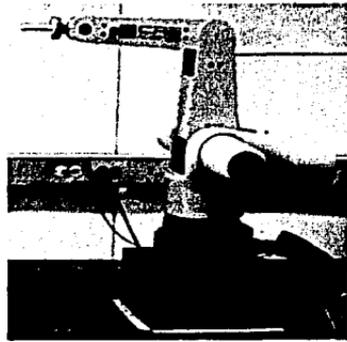
Apéndice E

Galería de fotos

En este apéndice se muestran fotos del sistema adaptado.



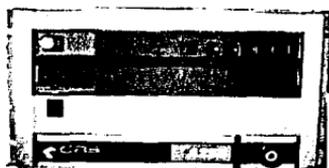
(a)



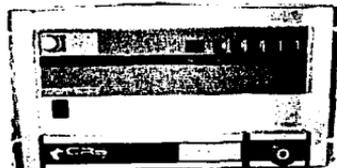
(b)

Figura E.1: Robots: a) A465, b) A255.

TESIS CON
FALLA DE ORIGEN



(a)



(b)

Figura E.2: Panel frontal del controlador C500: a) A255, b) A465.



(a)



(b)

Figura E.3: Controlador C500: a) original, b) adaptado.

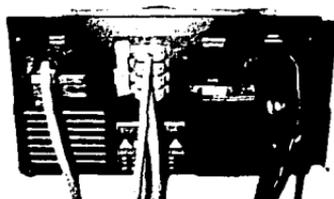


Figura E.4: Panel trasero del controlador C500 adaptado.

TESIS CON
FALLA DE ORIGEN

Apéndice F

Guía rápida de conexión

En este apéndice se describen las conexiones necesarias para cambiar, a nivel hardware, del sistema robótico original al sistema adaptado. Por convención se le asignó el número "1" a los accesorios del robot A465 y el número "2" para el robot A255.

F.1 Configuración del controlador C500

► *Para cambiar la configuración del controlador C500 del robot A465.*

1. Apague el equipo.
2. Retire la tapa superior del controlador C500 A465.
3. Identifique los conectores CBJ6, CBJ7, ABJ1 y RFJ4 dentro del C500 (Ver Figura F.1).

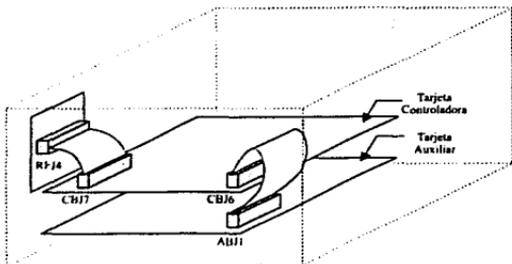


Figura F.1: Conectores del controlador C500 original.

4. Use una pulsera antiestática para realizar los pasos siguientes.
5. Desconecte el cable que une los conectores CBJ6 y ABJ1.
6. Desconecte el cable que une los conectores CBJ7 y RFJ4. Tenga cuidado de sujetar la tarjeta RFJ4.
7. Conecte el Bus BJ6 a los conectores CBJ6 y ABJ1 (Ver Figura F.2).
NOTA IMPORTANTE: Verifique que las etiquetas del Bus BJ6 y del C500 coincidan al momento de conectar.
8. Conecte el Bus BJ7 a los conectores CBJ7 y RFJ4 (Ver Figura F.2).
Tenga cuidado de sujetar la tarjeta RFJ4.
NOTA IMPORTANTE: Verifique que las etiquetas del Bus BJ7 y del C500 coincidan al momento de conectar.

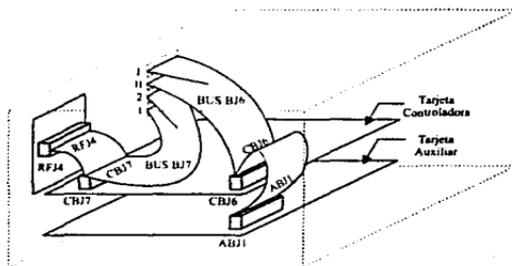


Figura F.2: Buses BJ6 y BJ7.

9. Verifique que todas las conexiones sean correctas (MUY IMPORTANTE).
10. Coloque la tapa del controlador C500.

El procedimiento antes descrito deshabilita el controlador de fábrica del controlador C500 para el robot A465 y permite disponer de las señales necesarias para el sistema adaptado desde los conectores C500DB25[I], C500DB25[II], C500DB25[1] y C500DB25[2].

► **Para cambiar la configuración del controlador C500 del robot A255.**

1. Apague el equipo.
2. Retire la tapa superior del controlador C500 A255.
3. Identifique los conectores CBJ6, CBJ7, ABJ1 y RFJ4 dentro del C500 (Ver Figura F.3).

TESIS CON
FALLA DE ORIGEN

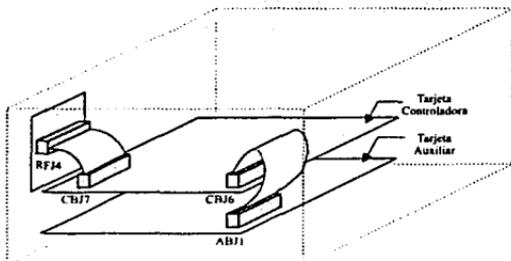


Figura F.3: Conectores del controlador C500 original.

4. Use una pulsera antiestática para realizar los pasos siguientes.
5. Desconecte el cable que une los conectores CBJ6 y ABJ1.
6. Desconecte el cable que une los conectores CBJ7 y RFJ4. Tenga cuidado de sujetar la tarjeta RFJ4.
7. Conecte el Bus BJ6 a los conectores CBJ6 y ABJ1 (Ver Figura F.4).
NOTA IMPORTANTE: Verifique que las etiquetas del Bus BJ6 y del C500 coincidan al momento de conectar.
8. Conecte el Bus BJ7 al conector RFJ4 (Ver Figura F.4). Tenga cuidado de sujetar la tarjeta RFJ4.
NOTA IMPORTANTE: Verifique que las etiquetas del Bus BJ7 y del C500 coincidan al momento de conectar.

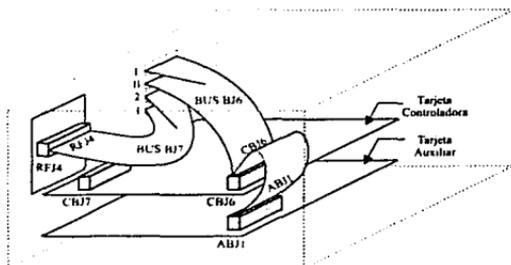


Figura F.4: Buses BJ6 y BJ7.

9. Verifique que todas las conexiones sean correctas (MUY IMPORTANTE).
10. Coloque la tapa del controlador C500.

El procedimiento antes descrito deshabilita el controlador de fábrica del controlador C500 para el robot A255 y permite disponer de las señales necesarias para el sistema adaptado desde los conectores C500DB25[I], C500DB25[II], C500DB25[1] y C500DB25[2].

F.2 Conexión del Chasis UMI y el controlador C500 adaptado

Para realizar la conexión del controlador C500 adaptado con el Chasis UMI para los robots A255 o A465 siga el siguiente procedimiento:

1. Identifique los conectores C500DB25[I], C500DB25[II], C500DB25[1] y C500DB25[2] en el panel trasero de la consola C500 A465 (para el robot A465) o C500 A255 (para el robot A255).
2. Identifique los conectores UMIDB25[I], UMIDB25[II], UMIDB25[1] y UMIDB25[2] en el Chasis UMI 1 (para el robot A465) o Chasis UMI 2 (para el robot A255). La Figura F.5 muestra los conectores del Chasis UMI.

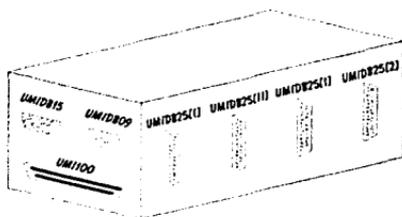
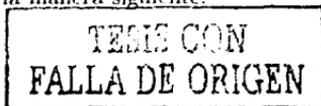


Figura F.5: Chasis UMI

3. Haga la interconexión de los conectores mencionados en 1. y 2. con la ayuda de los cables DB25 etiquetados como I, II, 1 y 2. La conexión se lleva a cabo de la manera siguiente:



Conector C500	Unir con cable DB25		Conector Chasis UMI
C500DB25[1]	← I	→	UMIDB25[1]
C500DB25[11]	← II	→	UMIDB25[11]
C500DB25[1]	← 1	→	UMIDB25[1]
C500DB25[2]	← 2	→	UMIDB25[2]

NOTAS:

a) Los cables DB25 I, II, 1 y 2 son idénticos y se pueden intercambiar indistintamente. Por convención cada cable se coloca con su respectivo número de conector.

b) Los Chasis UMI 1 y 2 pueden ser intercambiados indistintamente. Por convención el Chasis UMI 1 es para el robot A465 y el Chasis UMI 2 para el robot A255.

F.3 Configuración del Chasis UMI

- **Para utilizar las tres primeras articulaciones de los robots A255 y A465.**

1. Apague el controlador C500.
2. Cortocircuite los pines 4 - 6, 5 - 7, 8 - 11 del conector UMIDB15 en el Chasis UMI (1 o 2) con la ayuda del conector llave 1 (Ver Figura F.6).

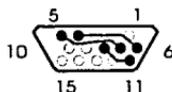


Figura F.6: Conector llave 1

3. Encienda el controlador C500.

- **Para utilizar las tres (dos) últimas articulaciones del robot A465 (A255).**

1. Apague el controlador C500.

2. Cortocircuite los pines 1 - 9, 2 - 10, 3 - 12 del conector UMIDB15 en el Chasis UMI (1 o 2) con la ayuda del conector llave 2 (Ver Figura F.7).

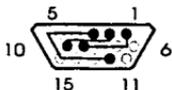


Figura F.7: Conector llave 2

3. Encienda el controlador C500.

► **Para utilizar las seis (cinco) articulaciones del robot A465 (A255).**

1. Apague el controlador C500.
2. Coloque el Bus Unidireccional UDI6 como sigue: El conector A de la UDI6 en el conector UMIDB15 del Chasis 2 (1) y el conector B de la UDI6 en el conector UMIDB15 del Chasis 1 (2).
3. Encienda el controlador C500.

NOTA ACLARATORIA: El procedimiento descrito arriba se cumple sólo si el Chasis UMI 1 es conectado al controlador A465 y el Chasis UMI 2 al controlador A255. En el caso de ser intercambiados, basta con remplazar el número 1 por 2 y viceversa en el procedimiento anterior.

F.4 Conexión del Chasis UMI y las tarjetas FlexMotion

Con la ayuda del cable de 100 vías, conecte las tarjetas FlexMotion con el conector UMI100 del Chasis UMI. Recuerde verificar que el número de identificación (ID) de la tarjeta, asignado por la computadora maestra, coincida con el número de identificación que se asigna en los programas creados en LabWindows/CVI.

TESIS CON
FALLA DE ORIGEN

Apéndice G

Glosario

A: amperes.

Actuador: Motor o transductor que convierte la energía eléctrica, hidráulica, o neumática en potencia para movimiento o reacción.

A/D: Analógico a digital.

Alcance (Reach): La distancia máxima que el brazo puede extender el efector final. Define la envolvente (límite) de trabajo.

Algoritmo: Conjunto de reglas específicas para la solución de un problema en un número finito de pasos.

Amplificador (amplifier): Controlador que envía la energía para operar al motor en respuesta al bajo nivel de las señales de control. En general, el amplificador está diseñado para operar con un tipo de motor en particular. No se puede utilizar un controlador de pasos para operar un motor de CD, por ejemplo.

Amplitud de movimiento (Range of Motion): Extensión del recorrido de un segmento o de un brazo, dependiente de los límites del movimiento rotacional de las articulaciones y del largo de los segmentos.

Analog input < 1..8 >: Entradas analógicas < 1..8 > Entradas analógicas del ADC de 12 bits.

Analog output < 1..6 >: Salidas analógicas < 1..6 > Salidas de voltaje del DAC.

Arquitectura abierta: Se refiere a una configuración del controlador donde cada uno de sus componentes principales pueden ser identificadas fácilmente.

Arquitectura cerrada: Que no es de arquitectura abierta.

Articulación (Joint): Parte de un brazo que conecta dos segmentos (eslabones), con un eje de rotación que provee un grado de libertad.

Axis < 1..6 > Inhibit: Inhabilitación de la salida de los ejes del 1 al 6.

Axis < 1..6 > Forward Limit Input: Interruptor de límite hacia adelante (CW) de los ejes 1 al 6.

Axis < 1..6 > Home input: Entrada de la señal home de los ejes 1 hasta el 6.

Axis < 1..6 > Reverse Limit Input: Interruptor de límite hacia atrás (CCW) de los ejes 1 al 6.

Brazo (arm): Parte mecánica de un sistema de robot, dirigida por el controlador, que realiza trabajos. Consiste de un conjunto de segmentos conectados secuencialmente por articulaciones. Cuenta con dos extremos, la base, sujeto a una plataforma, y con el otro extremo equipado con un actuador que realiza una tarea. Las articulaciones son capaces de mover los segmentos y el actuador a diversas ubicaciones y orientaciones.

TESIS CON
FALLA DE ORIGEN

- Bus:** Grupo de conductores que interconectan circuitos individuales en una computadora. Típicamente, un bus es el vehículo de expansión en el cual los dispositivos de entrada / salida son conectados.
- CA:** Corriente Alterna.
- Capacidad de carga (Payload):** Es el peso, generalmente en kilogramos, que el robot puede manipular. Peso soportado por el brazo y/o la magnitud de la fuerza que el brazo puede ejercer sobre un objeto. Carga útil o admisible.
- Computadora maestra: (host computer)** Es la computadora donde están alojadas las tarjetas de adquisición de datos. También conocido como ordenador huésped.
- Control:** El proceso de hacer que una variable o sistema de variables se ajusten a lo que se desea.
- Control derivativo:** Un esquema de control en el cual la señal del manejador para el actuador es proporcional a la derivada con respecto al tiempo de la diferencia entre la entrada (salida deseada) y la salida real medida.
- Controlador (controller):** Un dispositivo (hardware o software) que continuamente mide el valor de cantidad o condición de una variable y entonces automáticamente actúa sobre el equipo controlado para corregir alguna desviación en un valor deseado preestablecido. Dispositivo computarizado que controla al brazo, recibe información, procesa datos, envía señales de control al brazo y recibe retroalimentación del brazo.
- Controlador original:** Controlador (software o hardware) programado por el fabricante del sistema robot.
- Controlador propio:** Algoritmo de control programado desde la computadora maestra para el sistema adaptado.
- Cuentas en cuadratura (quadrature counts):** Cuatro líneas de resolución en el encoder.
- D/A:** Digital a analógico.
- DAC:** Convertidor digital a analógico (D/A Converter). Es parte de la circuitería en la tarjeta del controlador de los servo - ejes. Su función es convertir la señal digital de la computadora a un voltaje analógico.
- DC:** Corriente directa.
- Dedicado(a) (dedicated):** Asignada a una función particular.
- DGND:** Señal de tierra digital.
- Diagrama de bloques:** Descripción o representación gráfica de un programa o algoritmo.
- Dirección:** Código de caracteres que identifican una localidad específica o series de localidades en la memoria o en un sistema de PC anfitrión.
- DLL:** Librería de vínculos dinámicos (dynamic link library) - provee la API para los controladores de movimiento.
- Efector final (end effector):** Dispositivo sujeto al extremo del último elemento del robot que se utiliza para realizar las tareas del robot, tales como herramientas, tenazas, etc. A falta de herramienta se considera el extremo del elemento como el efector final.
- Eje:** Unidad que controla un motor o cualquier dispositivo de control o movimiento.
- Eje de movimiento (Axis of Motion):** Línea que pasa a través de una articulación y en torno a la cual gira un segmento o sección similar.
- Eje (X,Y o Z) (Axis X,Y,or Z):** Línea de referencia de un sistema de coordenadas. En un sistema cartesiano de coordenadas, los tres ejes (X, Y, Z) se cruzan en ángulos rectos; X y Y definen un plano horizontal y Z define la dimensión vertical. Cualquier punto en el espacio de trabajo del brazo, esto es, cualquier ubicación del manipulador o actuador, puede ser identificada con este sistema.
- Elemento (link):** Parte rígida del brazo robot entre dos articulaciones inmediatas.

TESIS CON
FALLA DE ORIGEN

- Encoder (codificador):** Dispositivo que transforma el movimiento mecánico en señales eléctricas; usado para monitorear la posición y velocidad en sistemas de lazo cerrado. Transductor usado para convertir posición lineal o rotativa a datos digitales. Dispositivo sujeto a cada eje motriz que convierte el movimiento de la articulación en señal para ser procesada por el controlador.
- E/S (I/O):** Entrada / salida - transferir datos a/de una computadora a/de canales de comunicación, dispositivos de interfaz, y/o interfaces de control movimiento.
- Espacio de trabajo (Workspace):** Espacio o volumen consistente de todos los puntos que el brazo robot puede alcanzar. El límite exterior del espacio de trabajo es la envoltura de trabajo.
- Error de seguimiento:** Diferencia entre la posición instantánea de la trayectoria comandada y la posición retroalimentada.
- GND:** Tierra (ground)
- Grados de libertad GDL(Degree of freedom DOF):** Movimientos básicos independientes que posicionan a los elementos de un robot en el espacio. Posibles movimientos básicos (giros y desplazamientos) independientes. Número de formas independientes que el efector final puede moverse. Es definido por el número de ejes rotacionales o traslacionales a través del cual el movimiento puede ser obtenido. Número de movimientos independientes en que el brazo puede mover su efector final (herramienta), definidos según los ejes de movimiento. Por ejemplo, el A465 tiene seis grados de libertad y puede mover a las coordenadas especificadas por X, Y, Z, giro de muñeca [giro en Z], Flexión de muñeca [giro en Y] y giro de actuador [giro en X].
- Home switch (input):** Una posición física determinada por el sistema mecánico o el diseñador como una posición para inicializar el sistema. Fresamente, la posición HOME es también referida como la posición cero en un marco de referencia de posición absoluta.
- Index:** Marcador entre revoluciones consecutivas del encoder.
- Lazo abierto:** Se refiere a un sistema de control de movimiento donde no son usados sensores externos para proveer corrección en las señales de velocidad y posición.
- Lazo cerrado:** Un sistema de movimiento que usa un dispositivo de retroalimentación para proveer datos de velocidad y posición, reportando su estado y poder controlar exactamente velocidad y posición. El uso de un lazo de retroalimentación para medir y comparar el desempeño real con el desempeño deseado. Esta estrategia permite el control del robot haga los ajustes necesarios.
- Limit switch / end-of-travel position (input):** Sensor que alerta al control electrónico que el fin físico de viaje se aproxima y que el movimiento debería detenerse.
- Manejador (driver):** La fuente o medio de suministro de energía de los actuadores del robot para producir movimiento.
- Modo absoluto:** Maneja la posición cargada como una posición relativa a cero mientras hace el movimiento.
- Parada de emergencia (e-stop):** Circuito que cortan el suministro de energía al brazo. Cuando se corta la energía del brazo, los frenos antifalla actúan para evitar que el robot se mueva por la gravedad o inercia. Método que neutraliza los controles del robot, desenergiza el brazo y detiene el movimiento del brazo robot.
- PID:** Lazo de control proporcional-integral-derivativo.
- Pinza:** Actuador diseñado para sujetar o tomar.
- Posición absoluta:** posición relativa a cero.
- Posición relativa:** Destino u objetivo del movimiento especificado con respecto a la posición actual a pesar de su valor.
- Procedimiento de homing:** Rutina mediante la cual se reconocen las referencias mecánicas de las articulaciones de los robots.

TESIS CON
FALLA DE ORIGEN

- Resolución (resolution):** Mínimo incremento de movimiento o distancia que puede ser detectado o controlado.
- Resolución del encoder:** Número de líneas del encoder entre señales index consecutivas (marcas o Z-bit). Si el encoder no tiene una señal index, la resolución del encoder puede ser referido como las líneas por revolución.
- Repetibilidad (repeatability):** Capacidad para repetir un mismo movimiento o llegar a los mismos puntos bajo las mismas señales de control; también se puede definir como el error de un ciclo a otro al intentar desempeñar una labor determinada.
- Ruido (noise):** Señal eléctrica indeseable - el ruido viene de fuentes externas tales como líneas de energía de AC, motores, generadores, transformadores, luces fluorescentes, soldadoras, CTR displays, computadoras, tormentas eléctricas, radio transmisores y fuentes internas tales como semiconductores, resistores y capacitores. El ruido altera las señales que tu intentas enviar o recibir.
- Servo motor CD:** Motores eléctricos controlados usando un mecanismo de retroalimentación. Un transductor en retroalimentación y un control de velocidad forman un servo loop (lazo servo). Los servo motores de CD son controlados a través de un cambio de voltaje; el motor corre más rápido si un voltaje más alto es aplicado. Consiste de estator, rotor, conmutador, cojinetes, armazón.
- Teach Pendal:** Terminal de control manual del robot, con un teclado y una pantalla de cuarzo líquido (LCD). Constituye un medio para movidizar al robot, indicarle ubicaciones, y ejecutar programas del robot.
- UMI:** Universal Motion Interface. Accesorio de conectividad para las tarjetas FlexMotion de National Instruments.
- VCD:** Voltaje de corriente directa.

TESIS CON
FALLA DE ORIGEN

Bibliografía

- Astrom, K. y B. Wittenmark. 1990. *Computer controlled system*. 2nd edn. Englewood clifts, New Jersey: Prentice Hall.
- Bentrup, C. M. 2002. *Manipulación estable de objetos rígidos utilizando dos robots industriales y un sistema de visión*. Tesis de Licenciatura. Facultad de Ingeniería, Universidad Nacional Autónoma de México.
- Caccavale, F. y P. Chiacchio. 2000. An experimental setup for cooperative manipulation. *Industrial Robot: An International Journal* 27(2). 120-130.
- CICESE 2000. Laboratorio de robótica. [on line]. Ensenada, B.C., México. Centro de Investigación Científica y de Educación Superior de Ensenada. Enero 2000 [cited 24 agosto 2001]. Available from <<http://electronica.cicse.mx/laboratorios/robotica.html>>.
- CRS 1995. *Controlador C500 - Guía de usuario*. CRS Robotics Corporation. Ontario, Canada. UMI-17-556-E.
- CRS 1997a. *A465 Robot Service Manual*. CRS Robotics Corporation. Ontario, Canada. UMS-17-504.
- CRS 1997b. *Brazo Robot A255 - Guía de usuario*. CRS Robotics Corporation. Ontario, Canada. UMI-14-504-E.
- CRS 2001a. *A255 Robot System User Guide*. CRS Robotics Corporation. Ontario, Canada. UMI-A255-400.
- CRS 2001b. *A465 Robot System User Guide*. CRS Robotics Corporation. Ontario, Canada. UMI-A465-400.
- Gudiño, J. y M. Arteaga. 2000. Adaptación de un robot industrial para su utilización en investigación. *Memorias del Segundo Congreso Nacional de Robótica*. AMRob. Toluca, México. 94-99.

TESIS CON
FALLA DE ORIGEN

- IFR UN/ECE 1999. Sales of industrial robots are booming in europe and north america but plummeting in japan and asia. *World Robotics 1999 - Statistics, Market Analysis, Forecasts, Case Studies and Profitability of Robot Investmen*. UN/ECE IFR. Geneva. ISBN No. 92-1-101007-1. Press Release. ECE/STAT/99/2. Available on line from <<http://www.unece.org/press/99stat2e.htm>>, 5 Octubre 1999 [cited 4 noviembre 2001].
- IFR UN/ECE 2000. The boom in robot investment continues - 900,000 industrial robots by 2003. *World Robotics 2000 - Statistics, Market Analysis, Forecasts, Case Studies and Profitability of Robot Investmen*. UN/ECE IFR. Geneva. ISBN No. 92-1-101029-2. Press Release ECE/STAT/00/10. Available on line from <<http://www.unece.org/press/00stat10e.htm>>, 17 October 2000 [cited 4 noviembre 2001].
- Kelly, R. 2001. Robot manipulador: Características y modelo. 13p.
- Khalid, S. 2000. *LabWindows / CVI: Programming for beginners*. U.S.A.: Prentice Hall. 651p.
- Kim, G. y G. Bekey. 1994. *Artificial Intelligence in Design '94*. Netherlands: Kluwer Academic Publisher.
- Kuo, B. 1996. *Sistemas de control automático*. 7th edn. México: Prentice Hall.
- Macchelli, A., C. Melchiorri y D. Pescoller. 2002. An experimental set-up for robotics and control systems research using real-time linux and comau smart 3-s robot. CHECAR Available on line from <<http://www.unece.org/press/00stat10e.htm>>, 17 October 2000 [cited 4 Octubre 2002].
- Natale, C. 2001. Welcome to the prisma lab. [on line]. Napoles, Italia. Università degli studi di Napoli Federico II. 24 octubre 1997. update 3 Agosto 2001 [cited 24 agosto 2001]. Available from <<http://www.prisma.unina.it/lab.htm>>.
- National Instruments 1998. *Motion Control - FlexMotion™ - Software Reference Manual*. 1998 edn. National Instruments. Austin, Texas, USA. Part Number: 321943A-01.
- National Instruments 1999a. *LabWindows/CVI - User Manual*. february 1998 edn. National Instruments. Austin, Texas, USA. Part Number: 320681D-01.
- National Instruments 1999b. *Motion Control - FlexMotion™ - 6C Hardware User Manual*. 1999 edn. National Instruments. Austin, Texas, USA. Part Number: 321944B-01.

TESIS CON
FALLA DE ORIGEN

- National Instruments 1999c. *Universal Motion Interface (UMI) Accessory - User Guide*. 1999 edn. National Instruments. Austin, Texas, USA. Part Number: 321941B-01.
- Rekoff, M. 1984. Method for reverse engineering. *Instrumentation in the Aerospace Industry* 30. 619-625.
- Spong, M. y M. Vidyasagar. 1989. *Robot Dynamics and control*. Singapore: John Wiley & Sons.
- Ziegler, J. G. y N. B. Nichols. 1942. Optimum settings for automatic controllers. *Transactions of the Amer. Soc. Mech. Engrs.* (64). 759-768.

TESIS CON
FALLA DE ORIGEN