

INSTITUTO TECNOLÓGICO Y DE ESTUDIOS SUPERIORES DE MONTERREY

DIVISION DE GRADUADOS EN INGENIERIA



T E S I S

"OPTIMIZACION DE UNA CELDA FLEXIBLE DE
MANUFACTURA CON AYUDA DE UN SISTEMA
DE VISION"

QUE PARA OBTENER EL GRADO DE MAESTRIA
EN INGENIERIA CON ESPECIALIDAD EN
INGENIERIA DE CONTROL ES

PRESENTADA POR:

VICTOR SEGURA FLORES

045.62
TEC.43
1990
c.2

MONTERREY, N. L.

MAYO DE 1990

INSTITUTO TECNOLOGICO Y DE ESTUDIOS SUPERIORES DE MONTERREY

DIVISION DE GRADUADOS EN INGENIERIA



T E S I S

"OPTIMIZACION DE UNA CELDA FLEXIBLE DE
MANUFACTURA CON AYUDA DE UN SISTEMA
DE VISION"

QUE PARA OBTENER EL GRADO DE MAESTRIA
EN INGENIERIA CON ESPECIALIDAD EN
INGENIERIA DE CONTROL ES

PRESENTADA POR:
VICTOR SEGURA FLORES

MONTERREY, N. L.

MAYO DE 1990



**INSTITUTO TECNOLÓGICO Y DE ESTUDIOS
SUPERIORES DE MONTERREY**

DIVISION DE GRADUADOS EN INGENIERIA

TESIS

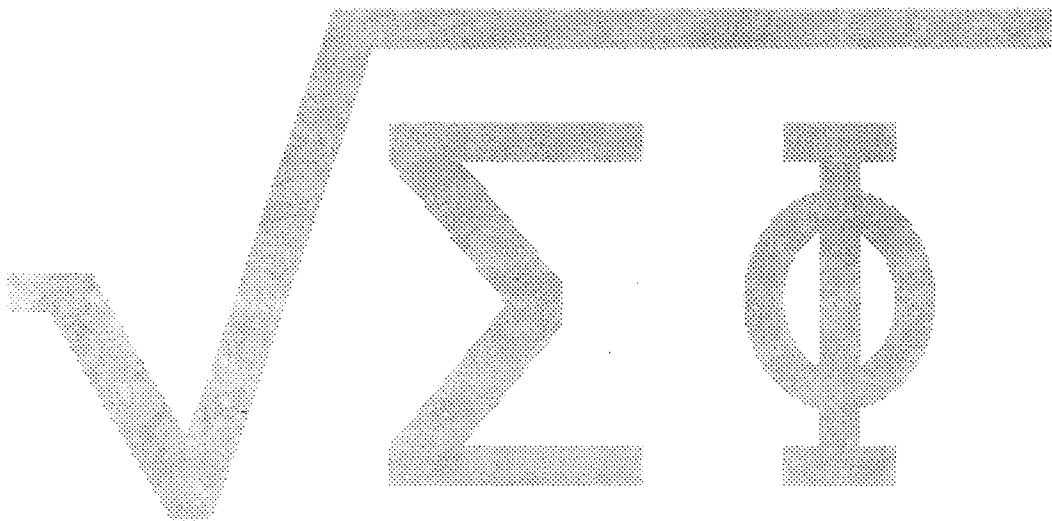
**"OPTIMIZACION DE UNA CELDA FLEXIBLE DE
MANUFACTURA CON AYUDA DE UN SISTEMA
DE VISION"**

**QUE PARA OBTENER EL GRADO DE MAESTRIA EN
INGENIERIA CON ESPECIALIDAD EN INGENIERIA DE
CONTROL ES PRESENTADA POR:**

VICTOR SEGURA FLORES

MAYO DE 1990

*Para mi Esposa Laura y para mi Hijo a
quien esperamos con amor.*



*Para orgullo de mi Madre y
gloria de mi Padre.*

AGRADECIMIENTOS :

Esta tesis es la finalización de mis estudios de maestría, la cual he logrado realizar gracias al apoyo incondicional que recibí de mi maestro y amigo el Lic. Juan Manuel Silva Ochoa. Por ello, quiero agradecerle de una manera muy especial que, sólo él sabrá comprender; los consejos y la ayuda que me brindó.

También deseo agradecer la valiosa ayuda y colaboración de mi asesor en este trabajo, el Dr. José de Jesús Rodríguez.

Finalmente, agradezco mucho la participación del joven Eduardo Franco para la terminación de este trabajo.

INDICE

CAPITULO 1 "OBJETIVOS e INTRODUCCION"

Objetivos e Introducción	1
--------------------------------	---

CAPITULO 2 "LA CELDA FLEXIBLE DE MANUFACTURA"

2.1 ¿Qué es un sistema flexible de manufactura?	4
2.2 Componentes de un sistema flexible de manufactura.....	5
2.3 Configuraciones de un sistema flexible.....	6
2.4 Descripción de la celda flexible de manufactura.....	9
2.4.1 Descripción técnica del robot.....	12
2.4.2 Especificaciones para torno y fresadora.....	14
2.4.3 Mesa intermedia.....	16
2.4.4 Bandas transportadoras.....	18
2.4.5 Controlador de los motores.....	18

CAPITULO 3 "PROGRAMACION DE LA CELDA FLEXIBLE DE MANUFACTURA"

3.1 ¿Qué es el control numérico?.....	19
3.1.1 Sistema de coordenadas. Posicionamiento absoluto e incremental.....	20
3.1.2 Tipos de sistemas de control numérico.....	22
3.2 Instrucciones y códigos para la programación de torno y fresadora	22
3.3 Programación del robot	36

CAPITULO 4 "EL SISTEMA DE VISION"

4.1 Los sistemas de visión.....	46
4.1.1 Adquisición y digitalización de la imagen	47
4.1.2 Procesamiento y análisis de la imagen	51
4.1.3 Interpretación	52
4.2 Aplicaciones de los sistemas de visión	52
4.3 Especificaciones del sistema de visión de la celda flexible	53

CAPITULO 5 "PROGRAMACION DEL SISTEMA DE VISION"

5.1	Módulo de control para la visión.....	57
5.2	Procedimiento para la sincronización entre las coordenadas del robot y el sistema de visión.	62
5.3	Procedimiento general para la operación de la celda flexible de manufactura, usando el sistema de visión	65

CAPITULO 6 "PROGRAMAS PARA LA OPTIMIZACION DEL SISTEMA"

6.1	¿Por qué es necesario el sistema de visión?.....	69
6.2	Descripción de los programas para la optimización del sistema	71
6.2.1	Programa para piezas maquinables en torno y fresadora.....	72
6.2.2	Programa para piezas maquinables en torno o fresadora.....	83

CAPITULO 7 "TECNICAS PARA LA DETECCION DE OBJETOS POR SUS BORDES"

7.1	Segmentación y descripción	88
7.2	Método de regiones.....	89
7.2.1	Método del umbral	89
7.2.1.1	Proceso interactivo por intensidad de pixeles	90
7.2.1.2	Operación óptima del umbral.....	96
7.2.2	Crecimiento de regiones.....	98
7.3	Métodos de detección de bordes.....	100
7.3.1	Operadores de Roberts y Sobel.....	101
7.3.2	La transformada de Hough.....	104

CAPITULO 8 "CONCLUSIONES"

Conclusiones.....	109
-------------------	-----

APENDICE A

Programas del robot para la integración de los elementos de la celda y el sistema de visión	111
---	-----

BIBLIOGRAFIA.....	123
--------------------------	------------

CAPITULO 1

OBJETIVOS E INTRODUCCION

El presente trabajo tiene dos objetivos principales. El primero de ellos, se enfoca al uso de un sistema de visión de una celda flexible de manufactura, para optimizar el funcionamiento de la misma. Se determina en forma precisa la posición de las piezas, se identifican y se realizan los movimientos necesarios para llevar las piezas al lugar o máquina que la solicite de acuerdo a las decisiones tomadas. Los algoritmos que se implementaron se presentan en el trabajo, logrando que el sistema funcione totalmente automático. Solo es necesario colocar las piezas en la banda de entrada y el sistema dará en la banda de salida piezas totalmente terminadas. En este sistema, el robot es la máquina encargada de la operación del proceso. Atiende a cada una de las máquinas entregándoles y recogiendo piezas una vez que las máquinas lo soliciten.

El segundo objetivo centra la atención a la presentación de las técnicas más comunes para la identificación de objetos donde la característica de mayor importancia son los bordes internos y externos de los objetos. Este material se presenta en el capítulo 7.

El capítulo 2 se enfoca hacia la descripción de los sistemas flexibles de manufactura y describe el sistema particular sobre el cual se trabajó. En el capítulo 3, se presenta una breve explicación sobre el control numérico y las instrucciones para la programación de la celda flexible. Se presentan algunos programas como ejemplos de la programación.

Los capítulos 4 y 5 tratan sobre el sistema de visión y la forma en que este se conecta con el robot para sincronizar los sistemas de coordenadas de la celda flexible y del sistema de visión. Se presenta un algoritmo para el uso del sistema de visión. Además, en el capítulo 6 se presentan los algoritmos realizados para la optimización del funcionamiento de la celda.

En el capítulo 8 se presentan los resultados y conclusiones del presente trabajo.

El uso y las aplicaciones de la computadora ha cambiado en forma significativa la tecnología aplicada a los procesos de manufactura. Hoy en día, un sistema integrado de manufactura está constituido por un conjunto de máquinas donde cada máquina debe realizar una tarea específica dentro del proceso de

producción. La tecnología está basada en la automatización de los elementos que están presentes en el proceso productivo, el cuál puede incluir:

- *Máquinas herramientas para el procesamiento de las piezas*
- *Máquinas ensambladoras*
- *Brazos mecánicos (robots) para mover y almacenar las piezas*
- *Sistemas de visión para el reconocimiento de las piezas*
- *Sistemas de inspección de calidad*
- *Computadoras para el control y retroalimentación del sistema*
- *Computadoras para simulación del proceso*

La comunicación entre los elementos del sistema, comúnmente llamado Celda Flexible de Manufactura, puede ser mediante códigos de control numérico o programas cíclicos, donde en la mayoría de los sistemas existe un elemento del proceso cuya programación es la encargada de englobar la programación del resto de las máquinas. De esta forma, dicho elemento puede ser el encargado de dar inicio a la secuencia de producción, efectuar pausas para la atención de las máquinas y dar fin al proceso.

Como resultado de una constante demanda de productos cuya calidad satisfaga las necesidades del cliente y además, debido a la gran competencia tanto nacional como internacional, las grandes firmas industriales han desarrollado nuevos métodos y técnicas de producción para lograr las características deseadas del producto final, introduciendo la "automatización" a las fábricas en base a sistemas flexibles de manufactura. Las razones principales para la automatización de los elementos de un proceso de producción son:

1. *Incremento en la Productividad al disminuir el tiempo de procesamiento de cada una de las piezas*
2. *Alto costo de la mano de obra*
3. *Seguridad en el trabajo*
4. *Alto costo de la materia prima desperdiciada*
5. *Obtener productos de calidad competitiva y*
6. *Reducción de inventarios*

Conforme aumenta la automatización de los elementos, los robots deben de aumentar sus capacidades sensitivas para interactuar con su medio ambiente de una mejor manera, logrando una gran capacidad de flexibilidad con su medio. De las formas en que un robot puede interactuar con su medio ambiente, la visión puede ser la más importante de todas. El proceso de la visión tiene como objetivo primordial derivar una descripción de la escena o medio ambiente, analizando una o más imágenes de dicha escena.

Para que el sistema de visión reconozca las piezas de trabajo y pueda tomar alguna decisión, primero debe distinguir los objetos de interés destacando partes de la imagen que corresponden a dichos objetos. Las escenas son digitalizadas de acuerdo a los niveles de gris que presentan y pueden ser almacenadas en memoria para un análisis posterior. Así, cada pieza de trabajo tiene una única representación por lo que se puede determinar los tipos de maquinado que se le han realizado y con ello, se determina el lugar o la máquina a la que puede continuar o bien, si la pieza está totalmente terminada, es llevada hacia la banda de salida o a la estación correspondiente de almacenaje.

CAPITULO 2

LA CELDA FLEXIBLE DE MANUFACTURA

2.1 ¿QUE ES UN SISTEMA FLEXIBLE DE MANUFACTURA? [1]

Un Sistema Flexible de Manufactura consiste de un grupo de estaciones de procesamiento interconectadas a sistemas automáticos para tomar y almacenar el material donde todo está controlado por un sistema integrado de computadora. Lo que dá el nombre de Sistema Flexible es la capacidad de procesar una gran variedad de diferentes tipos de piezas simultáneamente bajo programas de control numérico (ver capítulo 3) en las estaciones de trabajo.

El corazón de un sistema flexible es el control numérico computarizado (CNC). La clave principal de las máquinas de CNC es el hecho de que varios pasos pueden ser desarrollados en secuencia sin intervención manual y diferentes operaciones pueden ser almacenadas de tal manera que para iniciar una secuencia en particular, el operador solo requiere llamar al programa que contiene la secuencia. Las operaciones para retirar o colocar diferentes herramientas pueden ser realizadas sin interrumpir el proceso.

Los sistemas flexibles de manufactura pueden ser definidos de dos tipos: **SISTEMAS DEDICADOS** y **SISTEMAS ALEATORIOS**. Un *sistema dedicado* es usado para producir piezas con una variedad limitada en su diseño. Las diferencias geométricas son pequeñas por lo que el diseño del producto es considerado estable. Esto significa que la secuencia es casi idéntica para todas las partes procesadas. En lugar de usar máquinas de propósitos generales, éstas pueden ser diseñadas para el proceso específico. Los *sistemas aleatorios* son los más apropiados para trabajar bajo las siguientes condiciones: Los tipos de piezas pueden ser variados, existen variaciones sustanciales en la configuración de las mismas, pueden existir cambios en el tipo de pieza así como cambios en el procesamiento y la agenda de producción puede variar de un día a otro día. Para considerar todas estas variaciones, este tipo de sistemas tienen que ser más flexibles que los sistemas dedicados. Normalmente, el equipo de trabajo es un equipo de propósitos generales y es capaz de procesar partes en varias secuencias (orden aleatorio). Finalmente, el sistema de control computarizado debe ser más sofisticado.

Los sistemas flexibles son aplicables a una amplia variedad de operaciones de manufactura y sus principales ventajas y desventajas son:

VENTAJAS DE LOS SISTEMAS FLEXIBLES:

1. *Mejor aprovechamiento de los recursos capital / equipo*
2. *Reducción de tiempo de procesamiento*
3. *Reducción de inventarios*
4. *Reducción de mano de obra*
5. *Flexibilidad para realizar cambios rápidamente*
6. *Calidad consistente*
7. *Reducción de espacio*
8. *Mejor planeación*
9. *Mejor precisión*

DESVENTAJAS DE LOS SISTEMAS FLEXIBLES:

1. *Tecnología relativamente nueva*
2. *Sistemas complejos para operación y mantenimiento*
3. *Sistemas caros*
4. *La implementación puede durar varios años*
5. *Requieren una gran preparación de la mano de obra para ser manejados*
6. *Difíciles de integrar a la maquinaria ya existente en una empresa*

2.2 COMPONENTES DE UN SISTEMA FLEXIBLE DE MANUFACTURA

Básicamente, un sistema flexible de manufactura está constituido por tres componentes:

1. *Estaciones de Proceso.* Estas estaciones de trabajo son típicamente máquinas herramientas de control numérico que desarrollan el maquinado sobre las piezas. Además de este tipo de máquinas, los sistemas flexibles incluyen estaciones de inspección y ensamblado.

2. *Equipo para tomar y almacenar el material.* Se utilizan varios tipos de equipo automatizado para llevar las piezas entre las estaciones de trabajo y para almacenarlas.

3. *Sistema de control computacional.* Una computadora es usada para coordinar las actividades del procesamiento de las piezas así como el movimiento de las mismas entre dichas estaciones. Existen además controladores que son los encargados de efectuar los movimientos de los elementos del sistema. Las funciones del sistema de control computacional están agrupadas en las siguientes categorías.

1. *Control de cada estación de trabajo*
2. *Distribución de las instrucciones a cada estación*
3. *Control de producción*
4. *Control de tráfico*
5. *Monitoreo de las estaciones para tomar y almacenar material*
6. *Control de herramienta y*
7. *Monitoreo y reporte del comportamiento del sistema.*

2.3 CONFIGURACIONES DE UN SISTEMA FLEXIBLE

Lo que define la configuración de un sistema flexible es la forma en que el material puede ser llevado de un lugar a otro. En los casos en que los robots son los encargados de la transferencia de las piezas, la configuración del sistema se ve afectada por el alcance que éstos tienen. Los tipos de configuraciones que son más usados son divididos en cinco categorías a saber:

1. *En línea*
2. *En lazo*
3. *Escalera*
4. *Campo abierto*
5. *Robot centrado*

La *configuración en línea* es mostrada en la figura 2.1. Es la configuración más apropiada para sistemas donde la secuencia de operación es de una estación de trabajo a la siguiente. En esta configuración, el flujo de producción siempre es en un solo sentido.

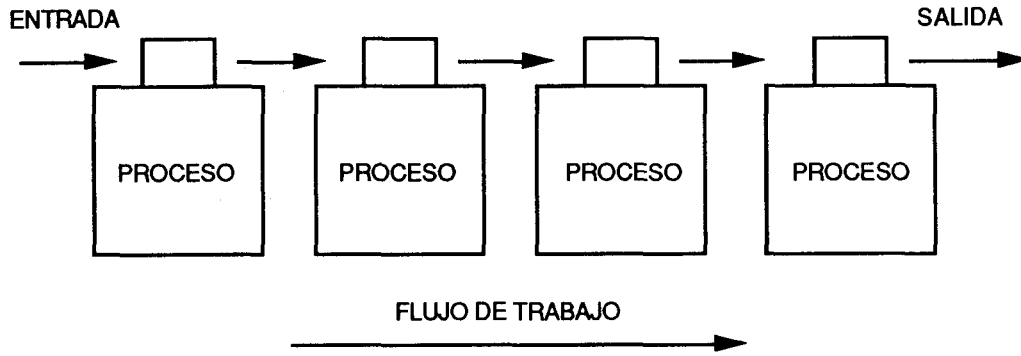


Fig. 2.1 Configuración en línea.

La configuración *en lazo* se muestra en la figura 2.2. Las partes fluyen en una dirección alrededor de un lazo con la capacidad de pararse en cualquier estación. La estación de carga y descarga está localizada al final del lazo. Un sistema secundario para tomar las piezas se encuentra en cada estación para no obstruir el paso alrededor del lazo.

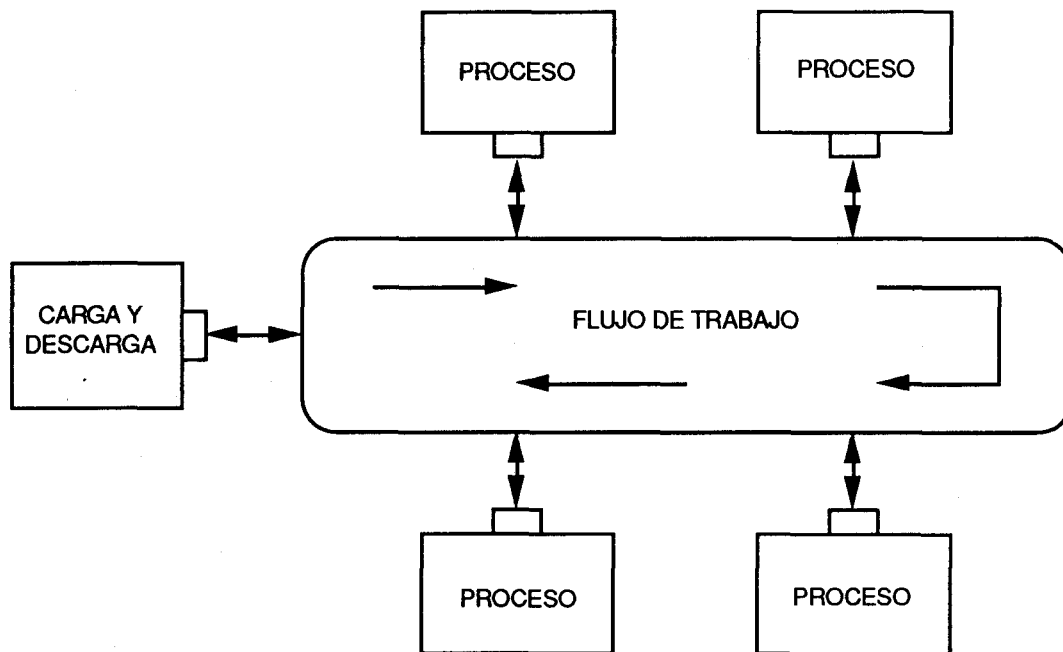


Fig. 2.2 Configuración en lazo.

La configuración *en escalera* es una adaptación de la configuración en lazo. Contiene travesaños que dan hacia las estaciones de trabajo, incrementando las opciones para que una pieza pase a la siguiente operación. Esto reduce la distancia promedio de viaje por lo que el tiempo de transferencia entre las estaciones se reduce considerablemente. Se muestra en la figura 2.3

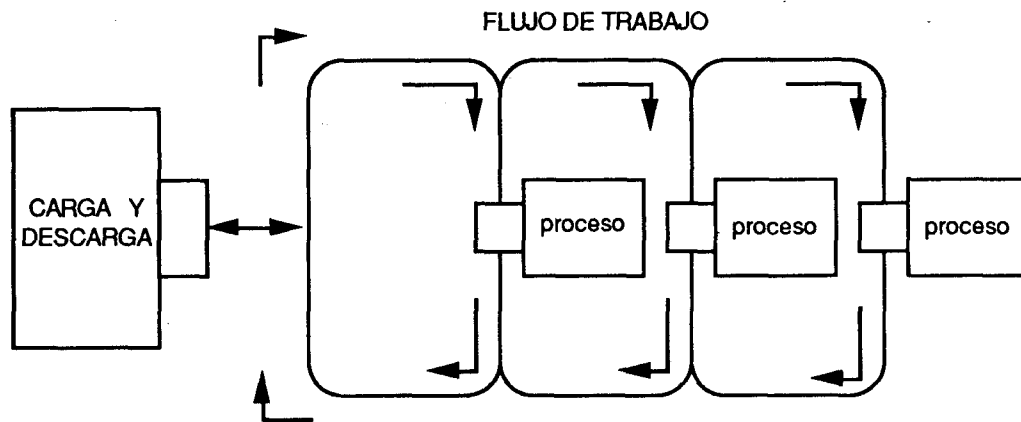


Fig. 2.3 Configuración en escalera.

La configuración de *campo abierto* también es una adaptación de la configuración en lazo. Consiste de lazos, escaleras y desviadores debidamente colocados para que las partes sean enrutadas a diferentes estaciones de trabajo dependiendo de cual esté disponible en ese momento.

Finalmente, la configuración del *robot centrado* es aquella donde uno o más robots son usados para transportar las piezas en el sistema. Se les puede adaptar tenazas configuradas a la forma específica de las piezas de tal manera que la operación de tomar y rotar las piezas sea más sencilla.

Una vez que se han definido los conceptos básicos sobre los sistemas flexibles de manufactura, en la siguiente sección describiremos la celda flexible de manufactura que fué utilizada para optimizar su operación con ayuda de un sistema de visión.

El problema que se presentaba en la operación de la celda flexible era el siguiente:

Cuando sea necesario que el tipo de maquinado se realice en el torno y la fresadora, la pieza de trabajo tiene que ser llevada a una mesa intermedia de almacenamiento una vez que se le ha realizado trabajo en una de las máquinas, para posteriormente pasarla a la segunda máquina cuando ésta se encuentre desocupada. La lógica para la programación cuando interaccionan las tres máquinas (torno, fresadora y robot), se complica, pudiendo tener el programa del robot más de diez subrutinas. Este problema se ha podido resolver gracias a la

ayuda del sistema de visión, ya que las piezas son digitalizadas de acuerdo a los niveles de gris que presentan (ver capítulo 4) y pueden ser almacenadas en memoria. Así, cada una de las piezas trabajadas o no trabajadas, tienen una única representación y de esta manera se puede saber si la pieza ha sido torneada o fresada. Cuando la celda flexible empieza a operar y las máquinas soliciten pieza, se les dá una pieza de la banda de entrada y las futuras ocasiones en que soliciten pieza, se busca en las posiciones ocupadas de la mesa intermedia, mediante el sistema de visión, una pieza que no haya sido trabajada en dicha máquina, haciendo más eficiente la programación. En este caso, el programa integrador para la operación del sistema consta únicamente de tres subrutinas, de las cuales dos de ellas son necesarias.

Con el programa que se realizó para la operación de la celda, solo es necesario colocar piezas en la banda de entrada, el robot detecta la posición en que se encuentran mediante el sistema de visión y las recoge para llevarlas a las máquinas que soliciten. Además, las recoge de las máquinas una vez que han sido trabajadas y, las lleva a la mesa o la banda de salida. De esta forma, la celda flexible de manufactura trabaja continuamente. Se recogen las piezas nuevas y se entregan en la salida piezas terminadas donde todas las operaciones se realizan en forma automática.

En el capítulo 3 hablaremos sobre la programación de dicha celda y en los capítulos posteriores veremos como manejar el sistema de visión.

2.4 DESCRIPCION DE LA CELDA FLEXIBLE DE MANUFACTURA

Hasta este punto, ya hemos visto cuales son los componentes más importantes de un sistema flexible de manufactura y a continuación describiremos la celda flexible de manufactura que se utilizó para el presente trabajo.

En primer lugar, la celda utilizada es una celda educacional; no es una celda de tamaño industrial. La configuración de esta celda es del tipo de robot centrado y se muestra en la figura 2.4.

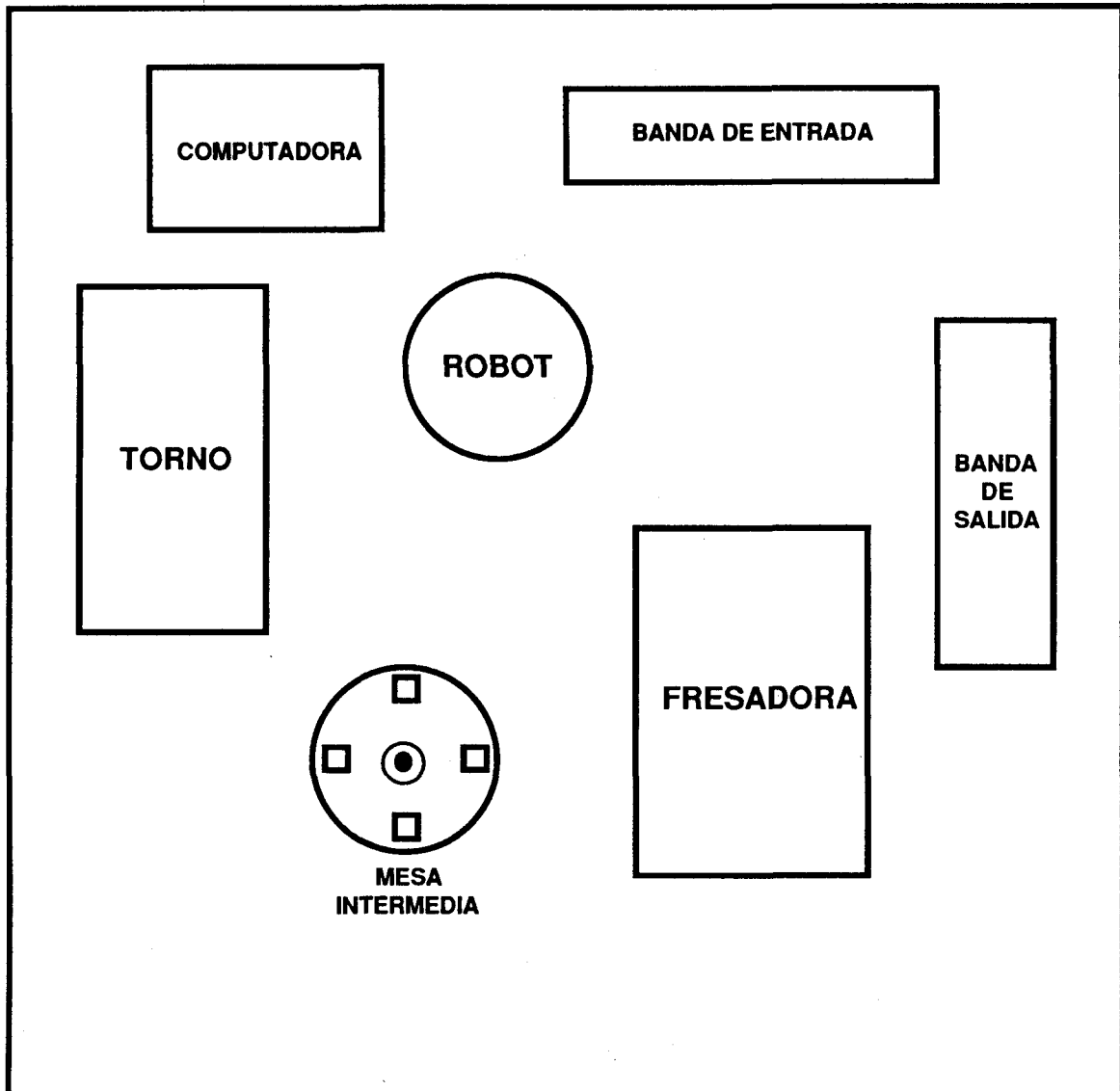


Fig. 2.4 Configuración de la celda flexible de manufactura utilizada.

Como se observa en la figura anterior, la celda flexible está constituida por los siguientes elementos:

- *Computadora IBM-XT de 640K RAM con disco duro de 20 MB y una unidad de disco flexible*
- *Banda de entrada del material*
- *Banda de salida del material*
- *Mesa intermedia para almacenamiento de piezas*

- *Torno de control numérico*
- *Fresadora de control numérico*
- *Robot de cinco grados de libertad con controlador*
- *Controladores CNC*

Además, posee un sistema de visión cuya operación y descripción se encuentra en los capítulos 4 y 5 de este material.

Las dos máquinas herramientas poseen un controlador propio que se encargan de efectuar los movimientos de la herramienta, movimientos de la mesa de trabajo, encender o apagar los motores, efectuar llamadas al robot para pedir que se recoja alguna pieza así como solicitarla. Estas operaciones son programadas desde la computadora como un código de control numérico. La computadora transfiere este código a los controladores y éstos a su vez los interpretan para realizar dichas operaciones. Los códigos e instrucciones para la programación de las máquinas herramientas se presentan en el siguiente capítulo.

El robot también tiene su propio controlador que ejecuta los movimientos programados en la microcomputadora, así como los movimientos de la banda de salida y de la mesa de almacenamiento.

El elemento maestro del sistema es el programa residente en la microcomputadora. Se puede ligar a los programas de las máquinas para realizar la sincronización en la operación y funcionamiento de la celda. Mientras que los programas de las máquinas contienen instrucciones para el maquinado e instrucciones para pedir que se les retire o se les dé alguna pieza de trabajo, los programas del robot jerarquizan la operación de las máquinas en el momento en que el robot tiene oportunidad de atenderlas. Cuando la máquinas requieren atención, el controlador de la máquina envía una señal al controlador del robot. Esta señal puede ser para indicarle al robot que la máquina se encuentra disponible para recibir una pieza nueva, o bien, para indicarle que ha terminado la operación de maquinado sobre la pieza por lo que solicita que se le retire dicha pieza.

Aún y cuando el controlador del robot es el maestro del sistema, los demás controladores son esclavos en la operación de cada máquina, incluido el robot.

Las características de operación del controlador centroid CNC son las siguientes:

- *Provee la lógica y energía suficiente para manejar los motores de paso que mueven a cada uno de los ejes de las máquinas CNC. Torno y fresadora.*

- *Contiene un interpretador de comandos para el código de control numérico. Se puede utilizar una microcomputadora o una terminal para comunicarse con él. Si se usa una microcomputadora, se tiene un editor donde se programan las instrucciones como código de control numérico, mientras que en una terminal, no existe un programa editor.*
- *Es manejado por medio de menús. Un menú principal, un menú editor y un menú de definición de parámetros con salidas que contienen mensajes descriptivos para la selección por parte del operador.*
- *Utiliza los códigos industriales estándar CNC. Esto provee la existencia de programas CNC que pueden ser cargados directamente, además de poder modificarlos y actualizarlos de una manera muy rápida.*
- *Complementa los códigos industriales estándar con varios códigos CNC extras, estos códigos son utilizados para operar relevadores, solenoides, interruptores y otros dispositivos los cuales son utilizados para realizar el control lógico de la secuencia de operación de un programa.*
- *Puede dar más de 5 Amp/bobina en el modelo estándar, el rango de par tiene un máximo de 100 oz-in.*
- *Puede manejar motores de 4 fases.*

Estas y otras características de operación que se definen en el siguiente capítulo muestran la razón del porqué el controlador CENTROID CNC es el maestro de la celda flexible de manufactura.

2.4.1 DESCRIPCION TECNICA DEL ROBOT

El robot o brazo mecánico está construido por elementos unidos mecánicamente y tiene 5 grados de libertad X, Y, Z, θ , γ y una tenaza. La figura 2.5 muestra las secciones de que está compuesto así como los movimientos que puede realizar.

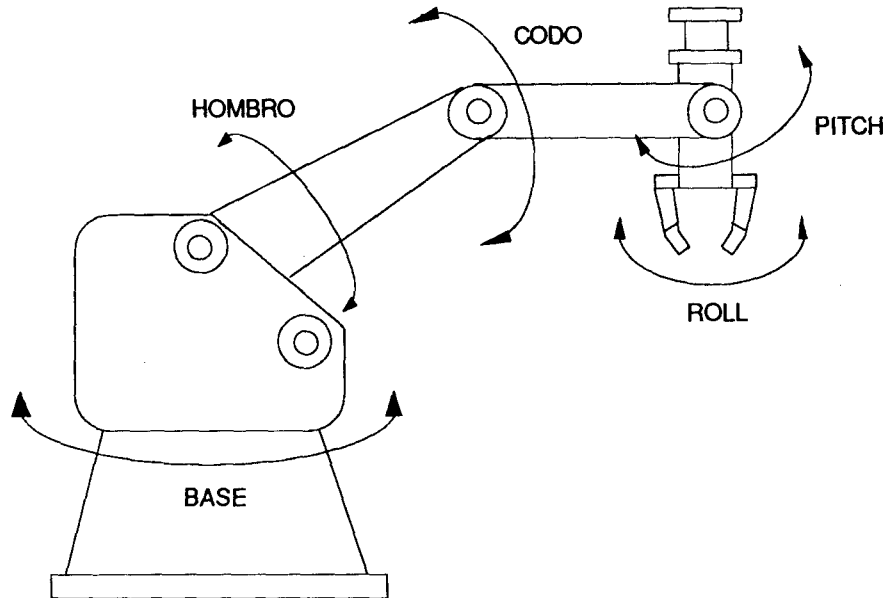


Fig. 2.5 Descripción básica del robot.

El número de elementos del brazo y el de las articulaciones que los relacionan, determinan los grados de libertad del robot, que en los robots industriales suele ser de 6, que coinciden con los movimientos independientes que posicionan las partes del brazo en el espacio. Tres de ellos definen la posición en el espacio y los otros tres definen la orientación del elemento terminal.

Para el robot utilizado, los cinco grados de libertad y la tenaza son operados por Servo motores DC de lazo cerrado. El correcto funcionamiento de un motor de este tipo exige la incorporación de un detector de posicionamiento del eje y una realimentación al sistema de control. Los detectores de posición para los motores que se utilizan, son codificadores incrementales ópticos. Su sistema de posicionamiento es incremental sin embargo, los elementos que componen el brazo mecánico tienen un micro interruptor para definir una posición física absoluta conocida como HOME de tal forma que esta posición puede ser usada para definir el sistema de referencia del robot.

ESPECIFICACIONES TECNICAS:

Número de ejes:	5 más la tenaza (el controlador puede controlar más de 8 ejes simultáneamente)
Capacidad de carga:	1 kg (2.2 lbs.)
Repetibilidad:	+/- 0.5 mm (+/- 0.02 in.)

Máxima velocidad rotacional:

Base	14.7 grados/seg.
Hombro	22 grados/seg.
Codo	30 grados/seg.
Elemento para ángulo de inclinación (Pitch)	105 grados/seg.
Elemento para ángulo de giro (Roll)	85 grados/seg.

Máxima abertura de tenaza: 75 mm (3 ins.)

Envolvente de área de trabajo:

Unión de base	340 grados
Unión de hombro	170 grados
Unión de codo	300 grados
Unión para el ángulo de inclinación	300 grados
Unión para el ángulo de giro	ilimitado

Dimensiones:

Altura máxima	930 mm (36.6 ins.)
Máximo radio de trabajo	610 mm (24 ins.)

Peso:

Robot	11 kgs (24.2 lbs.)
Controlador	5 kgs (11 lbs.)

2.4.2 ESPECIFICACIONES PARA TORNO Y FRESADORA

Las especificaciones referentes al torno y la fresadora, son concernientes a la definición de los parámetros de los motores que efectúan el movimiento de los ejes. Los parámetros son definidos en un menú de especificación de parámetros (SETUP MENU) que se encuentra en el menú de control (ver capítulo 3).

Cuando esta opción es seleccionada, el siguiente menú suplementario aparece en la pantalla:

SETUP MENU

1. *Display Setup Parameters*
2. *Display Tool/motor Parameters*
3. *Positioning/Units of Measure*
4. *Delete Mode/Optional Stop*
5. *Circular/Feedrate/Transmission Parameters*
6. *Tool Parameters*
7. *Motor Parameters*
8. *Zero Axis Position*
9. *Move Axis*
10. *Go To Axis Zero*
11. *Go To Axis Limit*
12. *Set Z Axis Home*
13. *Control Menu*

2- Display Tool/Motor Parameters

Cuando la opción 2 es seleccionada, se obtiene como salida los valores de los parámetros de los motores y herramientas. Esta información es presentada de la siguiente forma:

TOOL/MOTOR PARAMETERS

<i>Tool</i>	<i>Length</i>	<i>Diameter</i>	<i>Tool</i>	<i>Length</i>	<i>Diameter</i>	<i>Tool</i>	<i>Length</i>	<i>Diameter</i>
1	1.6000	0.0000	2	1.4500	0.0000	3	2.0000	0.0000
4	1.0500	0.0000	5	0.0000	0.0000	6	0.0000	0.0000
7	0.0000	0.0000	8	0.0000	0.0000	9	0.0000	0.0000
10	0.0000	0.0000	11	0.0000	0.0000	12	0.0000	0.0000
13	0.0000	0.0000	14	0.0000	0.0000	15	0.0000	0.0000
16	1.5450	0.0000	17	0.0000	0.0000	18	0.0000	0.0000
19	0.0000	0.0000	20	0.0000	0.0000	21	0.0000	0.0000
22	0.0000	0.0000	23	0.0000	0.0000	24	0.0000	0.0000

<i>Axis</i>	<i>Turns</i>	<i>Steps</i>	<i>Inst</i>	<i>Max</i>	<i>Acc</i>	<i>Limits</i>		<i>Motor Parameters</i>							
	<i>/Unit</i>	<i>/Turn</i>	<i>Rate</i>	<i>Rate</i>	<i>Rate</i>	-	+	1	2	3	4	5	6	7	8
X	10.00	200	300	800	600	1	2	5	9	10	6				
Y	10.00	200	300	600	400	3	4	5	9	10	6				
Z	10.00	200	100	500	200	5	6	5	9	10	6				

Cada uno de los parámetros es definido a continuación.

Turns/Unit. Es el número de vueltas por unidad de longitud del tornillo de dirección del eje. Por ejemplo 10 vueltas/pulgada

Steps/Turn. Es el número de pasos por revolución del motor. Típicamente es de 200 para pasos completos y 400 para medios pasos.

Inst Rate. Es la velocidad instantánea o de inicio del motor y está dada en pasos por segundo.

Max Rate. Es la máxima velocidad del motor dada en pasos por segundo. Esta es la velocidad a la que giran los motores cuando se ejecuta una instrucción del tipo G0.

Acc Rate. Es la aceleración del motor en pasos por segundo². Esta es la razón a la cual el motor puede pasar de la velocidad instantánea a la máxima velocidad.

Los movimientos transversales son realizados a la máxima velocidad, mientras que los movimientos lineales y circulares son realizados en velocidad instantánea.

3- Positioning/Units of Measure

Al seleccionar esta opción, el sistema ofrece la oportunidad al operador de seleccionar el tipo de unidades de longitud en que desee trabajar, así como el sistema de posicionamiento.

La salida de esta opción es la siguiente:

" Absolute or Incremental Positioning (A or I) ? "

" Inch or Metric Units of Measure (I or M) ? "

El resto de las opciones son usadas para: definir posiciones de referencia y tolerancias, efectuar movimientos en cada eje al origen establecido y efectuar movimientos a la coordenada máxima en cada eje.

2.4.3 MESA INTERMEDIA

La mesa rotatoria intermedia, es un ejemplo de la expansión del sistema del robot para simular una estación de trabajo automática. Sirve para posicionar las piezas de trabajo cuando éstas tienen que ser transferidas de una máquina a otra después de haber sido maquinadas. Por ejemplo, si las piezas tienen que ser torneadas y fresadas, primero son llevadas a una máquina para realizarles un tipo de trabajo y posteriormente son llevadas a la mesa intermedia. Cuando la segunda máquina se encuentre desocupada, se le dá una de las piezas que se encuentren en la mesa intermedia y que no ha pasado por esta segunda máquina. De esta forma, la pieza será terminada y luego será llevada hacia la banda de salida. La figura 2.6 muestra la mesa intermedia.

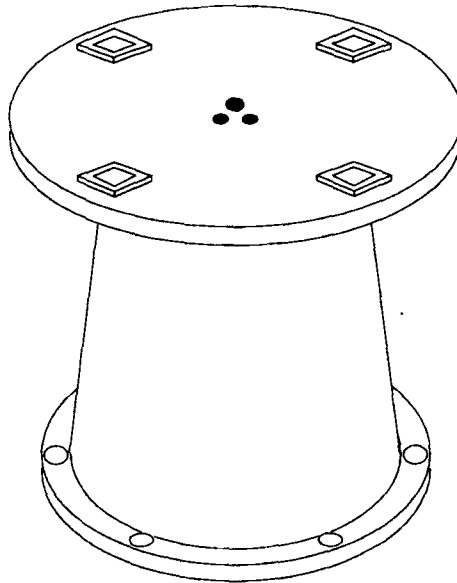


Fig. 2.6 La mesa rotatoria Intermedia.

Se compone de dos partes:

1. *Una base estacionaria*
2. *Un plato redondo que tiene movimiento en contra y a favor de las manecillas del reloj*

Un motor está fijado en la base y está conectado al controlador del robot como la entrada 6 para motores. La razón de disminución de la velocidad de la caja reductora es de 426: 1 y la velocidad de rotación del plato es de 12 RPM.

El plato redondo tiene 4 apartados fijos donde son colocadas las piezas de trabajo durante la operación del sistema. El diámetro del plato es de 305 mm y los apartados son cuadrados de 40 mm de lado.

La mesa rotatoria agrega un grado de libertad al sistema del robot permitiendo que el robot sea el que posiciona las piezas en la mesa.

El controlador CNC trata a la mesa rotatoria como el eje 6 del sistema.

2.4.4 BANDAS TRANSPORTADORAS

Las bandas transportadoras son otro ejemplo para la simulación de un sistema industrial. Se tienen dos bandas transportadoras en el sistema. Una de ellas es para transportar las piezas nuevas hacia un lugar en el cual las pueda recoger el robot y llevarlas hacia las máquinas. Esta banda, es la banda de entrada. La segunda banda es la de descarga y sirve para llevar las piezas hacia la salida de la celda flexible. El robot se encarga de llevar las piezas de las máquinas hasta esta banda. Es operada como una posición del robot. Se muestra en la figura 2.7

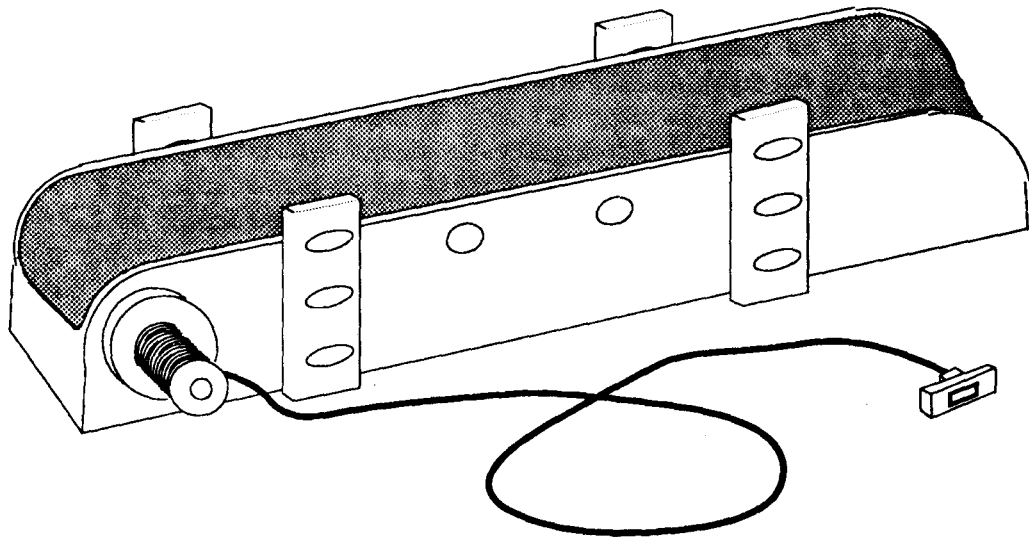


Fig. 2.7 Banda transportadora.

La longitud de las bandas es de 810 mm y su ancho es de 100 mm. Están montadas sobre una base de metal y son controladas por motores cuya caja de velocidad tiene una razón de 65.5: 1.

2.4.5 CONTROLADOR DE LOS MOTORES

El controlador de los motores es una unidad que se conecta al controlador del robot. El controlador del robot es capaz de controlar 8 servo motores DC. Los motores 1 a 5 controlan los movimientos del robot y el motor 8 está conectado a las tenazas. Los motores 6 y 7 son para la mesa intermedia y la banda de salida. El uso y operación de esta unidad se hace por medio del software del robot (SCORBASE SOFTWARE).

CAPITULO 3

PROGRAMACION DE LA CELDA FLEXIBLE DE MANUFACTURA

3.1 ¿QUE ES EL CONTROL NUMERICO?

La electrónica y la tecnología computacional han tenido una gran influencia sobre la función de control en procesos de manufactura. Un ejemplo importante sobre la aplicación de esta tecnología en sistemas de producción es el Control Numérico. El control numérico usa un programa de instrucciones que es transmitido electrónicamente al equipo de producción para regular su función y operación [1].

La parte de la programación del control numérico, es concerniente a la planeación y documentación de la secuencia de los pasos de procesamiento que serán desarrollados sobre la máquina de control numérico. El control numérico es una forma de automatización programable en la cual el equipo es controlado por medio de números, letras y otros símbolos. Los números, letras y símbolos son codificados en un formato apropiado que define un programa de instrucciones para un trabajo en particular. Si el tipo de trabajo cambia, el programa de instrucciones debe cambiar. La capacidad de cambiar el programa es lo que hace que el control numérico sea fácil de usar ya que es mucho más fácil escribir un nuevo programa que hacer alteraciones en el equipo.

Las aplicaciones del control numérico son muy amplias pero, vamos a dividir las en dos categorías:

1. *Aplicado a máquinas herramientas tales como tornos, fresadoras, cepillos.*
2. *Aplicado a máquinas ensambladoras y de inspección.*

El principio de operación del control numérico es: el control de la posición relativa de la herramienta con respecto al objeto que será procesado. Los componentes básicos de un sistema de control numérico son los siguientes:

- 1.- *Programa de Instrucciones*
- 2.- *Unidades de Control*
- 3.- *Equipo de manufactura (máquinas herramientas)*

La relación de estos tres componentes está ilustrado en la figura 3.1

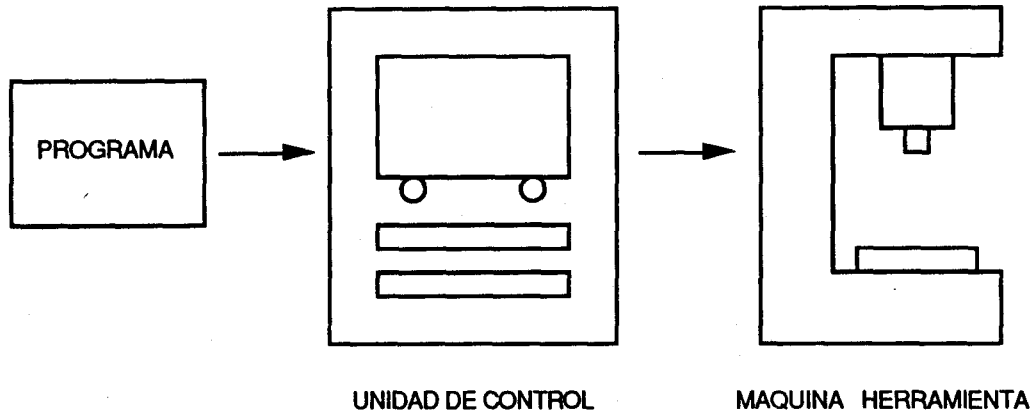


Fig 3.1 Componentes básicos de un sistema de control numérico.

El Programa de Instrucciones son comandos que definen paso a paso la operación de maquinado. Estos comandos se refieren a posiciones de la herramienta, orientaciones de los ejes, direcciones de corte, velocidad de corte y otras funciones.

La Unidad de Control consiste de la electrónica y el hardware de control para leer e interpretar el programa de instrucciones y lo convierte en acciones mecánicas de la máquina herramienta.

Las Máquinas Herramientas son los componentes que efectúan el trabajo mediante el movimiento de: mesa de trabajo, porta herramientas y control de los motores.

3.1.1 SISTEMA DE COORDENADAS. POSICIONAMIENTO ABSOLUTO E INCREMENTAL.

Usando como ejemplo la fresadora, la broca se encuentra sobre un eje. Para hacer más fácil la programación, se puede considerar que la pieza de trabajo está estacionaria mientras que la herramienta es la que se mueve. El sistema de coordenadas está definido con respecto a la mesa de trabajo.

Los ejes x y y son definidos sobre el plano de la mesa de trabajo y el eje z es perpendicular a este plano y el movimiento en el eje z es controlado por el movimiento vertical de la broca. Las direcciones positiva y negativa del movimiento son mostradas en la figura 3.2

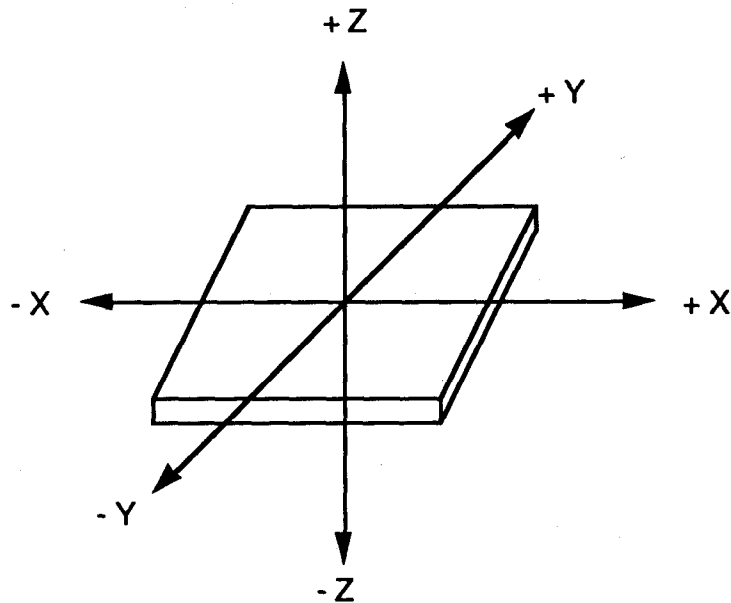


Fig. 3.2 Sistema de coordenadas de la fresadora.

Para el torno, solo es necesario considerar dos ejes, ya que solo se requiere el movimiento de la herramienta sobre la pieza de trabajo. Como un ejemplo, el eje z es el eje de rotación de la pieza de trabajo y el eje x define la localización de la herramienta. Esto se muestra en la figura 3.3

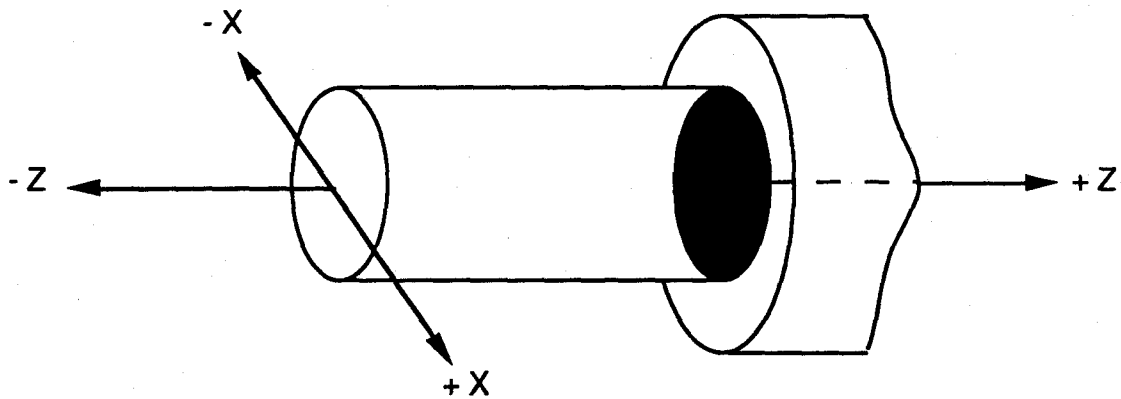


Fig. 3.3 Ejes X y Z para el torno.

El propósito del sistema de coordenadas es dar la localización de la herramienta en relación a la pieza de trabajo. En algunas máquinas de control numérico, el origen o posición cero está localizado siempre en una misma posición sin embargo, otras máquinas tienen la opción de definir el origen en cualquier posición. Este es el caso de las máquinas que se han usado.

Otra opción disponible es el uso del Sistema Absoluto o del Sistema Incremental para el posicionamiento de la herramienta. En el Sistema Absoluto, la localización de la herramienta siempre se define en relación al punto cero u origen. En contraste, en el Sistema Incremental la posición de la herramienta se define en base a la localización anterior de la herramienta.

3.1.2 TIPOS DE SISTEMAS DE CONTROL NUMERICO [2].

Existen tres tipos de sistemas para el control del movimiento de la herramienta con respecto a la pieza. Estos tres tipos son:

SISTEMAS PUNTO A PUNTO. Este sistema también es llamado Sistema Posicional. El objetivo de este sistema es mover la herramienta a una localidad definida. La velocidad y la trayectoria del movimiento no es importante en estos sistemas. Una vez que la herramienta se encuentra en la posición deseada, la operación de maquinado es realizada en dicha posición. Este tipo de sistemas son las máquinas más simples por lo que son los menos caros de los tres tipos de sistemas.

SISTEMAS DE CORTE RECTO. Este tipo de Sistemas son capaces de mover la herramienta paralelamente a uno de los ejes. En estos sistemas no es posible combinar movimientos en más de una dirección. Un sistema de este tipo también es capaz de desarrollar los movimientos de un sistema punto a punto.

SISTEMAS DE CONTORNO. Estos sistemas con los más complejos, los más flexibles y los más caros. Son capaces de desarrollar los movimientos de los sistemas anteriores. Su principal característica es la capacidad para controlar simultáneamente más de un eje de movimiento. La trayectoria de corte es controlada continuamente para controlar la geometría del maquinado y por esta razón, estos sistemas también son llamados Sistemas de Trayectoria Continua. El torno y la fresadora usadas en la celda flexible son ejemplos de este tipo de Sistemas.

3.2 INSTRUCCIONES Y CODIGOS PARA LA PROGRAMACION DEL TORNO Y FRESADORA [3]

Como se ha visto, la unidad de control es la encargada de transformar los símbolos, números y letras en operaciones definidas de maquinado. En el sistema o celda flexible que se ha utilizado, esta unidad de control es el *CENTROID CNC* (ver capítulo 2). *El CENTROID CNC* es programado y supervisado desde una microcomputadora o una terminal a través de un puerto serie RS232. Su memoria de 16 K RAM permite entre 800 y 1000 líneas de programación. La información de los programas no se pierde aún después de

apagar el sistema y los parámetros iniciales del modo SETUP no es necesario definirlos cada vez que se inicia la operación.

Los interruptores límite de los ejes y los interruptores de entrada son totalmente programables. Son siete los interruptores de entrada que están disponibles. El interruptor de entrada número 8 está reservado para paros de emergencia.

El operador puede gobernar la aceleración y desaceleración de los movimientos. También puede definir el origen (HOME) del sistema de coordenadas. Durante la operación de los programas, el número de líneas, posición de los ejes y velocidad de corte son actualizados en la pantalla.

Para cancelar la operación del programa, el *CENTROID CNC* nos permite utilizar la operación < Control c >

MENU DE CONTROL

Cuando el controlador CNC es encendido, el siguiente menú aparece en la pantalla:

CONTROL MENU

1. RUN/AUTO
2. RUN/BLOCK
3. SEARCH
4. SETUP
5. EDIT
6. RELEASE
7. RESET
8. MOTOR TEST

OPTION NUMBER?

La respuesta a esta pregunta determina el proceso seleccionado.

CARRIAGE RETURN

Esta respuesta redespliega el menú de control para inspección del operador. Esto es útil porque el menú de control no es desplegado automáticamente después de completar cada proceso. El operador puede usar esta respuesta para inspeccionar todas las opciones disponibles.

1. RUN/AUTO

Esta opción ejecuta el programa CNC creado con el Editor, además carga las condiciones iniciales establecidas y entrega como salida el siguiente mensaje:

"HIT RETURN to CONTINUE"

Cuando el operador presione la tecla RETURN, comienza la ejecución del programa.

2. RUN/BLOCK

Esta opción ejecuta el programa CNC, línea por línea. Después de cada línea ejecutada aparece el siguiente mensaje:

"HIT RETURN TO CONTINUE".

3. SEARCH

Busca en el programa un número de línea. El mensaje "Number to search for?N" es la salida. Cuando la línea es dada, la búsqueda comienza, el programa rastrea línea por línea para el número especificado. Cada línea precedida por "." es ejecutada, todas las demás son ignoradas. Cuando la línea es localizada, el mensaje:

"SEARCH SUCCESSFUL"

es la salida. Cuando la tecla de RETURN es presionada, el programa continúa desde ese punto.

4. SETUP

Transfiere las operaciones al modo setup. Es usado para establecer las condiciones iniciales de operación del programa.

5. EDIT

Se activa el editor para insertar, modificar o simplemente inspeccionar el programa CNC. Cuando esta opción es elegida, aparece un menú del editor.

6. RELEASE

Cancela la potencia a los motores. Los motores también pueden ser apagados mediante instrucciones M. Después de que esta opción se seleccionó, la salida es el siguiente mensaje:

"X, Y, Z Axis or All (X,Y,Z or A)"

El motor del eje que se desea apagar puede ser escogido al seleccionar la opción

7. RESET

Al escoger esta opción, todos los parámetros son reinicializados. Esta opción borra cualquier programa que se encuentre en la memoria. Da posicionamiento absoluto y coloca el origen de los ejes X, Y y Z en la posición presente de dichos ejes. Elimina todos los interruptores límite que se encuentren encendidos. Reinicializa los parámetros de los motores. Aparece el siguiente mensaje después de que se seleccionó:

Are you sure (Y or N) ?

8. MOTOR TEST

Prueba los parámetros de los motores y su montaje. Los tres motores de los ejes inician un movimiento de atrás hacia adelante para una distancia de 1 pulgada, con una velocidad de 3 plg/min. Este movimiento continúa hasta que el operador presiona < Control c >.

EDIT MENU

Cuando se selecciona la opción EDIT del menú de control del CNC se despliegan las siguientes opciones:

EDIT MENU

- T - Top
- B - Bottom
- I - Insert/Download
- D - Delete
- L - List/Upload
- M - Move
- C - Copy
- F - Display Free Space
- S - Squeeze Space
- K - Kill Program
- X - Exit to Control Menú

COMMAND?

La respuesta a COMMAND determina la opción seleccionada.

T-TOP Despliega la primera línea del programa CNC.

ejemplo: "COMMAND?" T

1: N5 G90 G75

B-BOTTOM Despliega la última línea del programa CNC

ejemplo: "COMMAND?" B

207: N900 M2

I-INSERT/DOWNLOAD Cuando se tecldea (In), donde I es la opción de insert y n es el número del línea, se prepara al editor para insertar las líneas al programa, en la pantalla se despliega el número y (:), cuando el operador termina de teclear la línea y oprime RETURN el número siguiente y (:) aparece. Para salir de este comando se tecldea RETURN.

Este comando también se utiliza para cargar el programa directamente de la computadora.

Ejemplo: Si existe el siguiente programa CNC:

1: N100 G90 G0 X4.0 Y3.0

2: N120 X5.2

3: N140 X6.7

Y queremos agregar más líneas entonces:

"COMMAND?" I3

3: N130 G91 Y4.2

4: N135 G90

5: (RETURN)

El programa que resulta será:

1: N100 G90 G0 X4.0 Y3.0

2: N120 X5.2

3: N130 G91 Y4.2

4: N135 G90

5: N140 X6.7

D-DELETE. Como parámetros se le dan los números de líneas que delimitan el block que se desea borrar:

Ejemplo: utilizando el ejemplo anterior tenemos:

"COMMAND? D1,2

Entonces se obtendrá lo siguiente:

1: N130 G91 Y4.2
2: N135 G90
3: N140 X6.7

L-LIST/UPLOAD La forma completa del comando es Ln1,n2 y despliega en pantalla el bloque comprendido entre las líneas inicial (n1) y la final (n2)

Si los números son omitidos, el editor actualiza el programa del CNC centroid controller cargándolo de la computadora al controlador.

Ejemplo: "COMMAND?" L1,3

Entonces se obtendrá lo siguiente

1: N130 G91 Y4.2
2: N135 G90
3: N140 X6.7

M-MOVE El comando completo es Mn1,n2,n3 donde n1 es el número inicial, n2 es el número final y n3 es el número de inserción. El comando mueve el bloque comprendido entre n1 y n2 a un lugar después de n3.

C-COPY Es similar a Move, la diferencia es que Copy genera dos versiones del mismo segmento.

F-FREE Despliega el número de líneas que se han introducido y el espacio que queda disponible.

S-SQUEEZE SPACE Elimina la fragmentación ocurrida al realizar funciones de borrado de segmentos.

K-KILL PROGRAM Borra cualquier programa que exista en la memoria del CNC, pide la confirmación con el siguiente mensaje:

Are you sure (Y or N) ?

X-EXIT TO CONTROL MENU Salida del menú del editor.

CODIGOS DE PROGRAMACION

Los principales sistemas que se implementaron para la programación de máquinas de control numérico, nacieron en la década de los 60's. En aquella época, la programación se realizaba mediante tarjetas perforadas donde cada renglón de la tarjeta era usado para definir una instrucción de operación. En los años recientes, el uso de las cintas magnéticas y discos flexibles son los medios más comunes para el almacenamiento de los programas.

Uno de los primeros códigos de programación que apareció en los Estados Unidos de Norteamérica, es el código EIA. Dicho código es el que presenta más estándares de programación en la industria de las máquinas de control numérico. Fué elaborado por The Electronics Industries Association (EIA). El código de programación que usa la celda flexible de manufactura, es un código de este tipo. Las instrucciones son presentadas en esta misma sección.

Además de el código EIA, existen muchos otro códigos que son definidos como códigos más estructurados. Entre estos tipos de códigos podemos mencionar los siguientes:

APT (Automatically Programmed Tools). Este lenguaje fue desarrollado en el año de 1956. Su uso se centra en la programación para sistemas con más de cinco ejes.

AUTOSPOT (Automatic System for Positioning Tools). Fue desarrollado por la IBM en el año de 1962.

SPLIT (Sundstrand Processing Language Internally Translated)

COMPACT II

ADAPT (Adaptation of APT)

EXAPT (Extended Subset of APT)

A continuación se listan las instrucciones del código de programación para la celda flexible de manufactura.

FUNCIONES G

G0. Trasladar la herramienta sin cortar, movimiento rápido.

Ejemplo: G0 Z0

La velocidad máxima en el movimiento de traslación puede ser definida por el operador en el menú de control en la opción de SETUP MENU.

G1. Movimiento para cortar, se le da el avance
Ejemplo: G1 Z-19 F100, F100 representa la velocidad de avance de la herramienta en décimas de pulg./minuto donde en este ejemplo, la velocidad de avance de la herramienta es de 10 pulgadas/minuto.

G2. Movimiento circular en contra de las manecillas del reloj.

G3. Movimiento circular a favor de las manecillas del reloj.

G4. Define las pausas en la ejecución del programa
Ejemplo: G4/200 pausa de 200 seg/100 = 2 segundos

G17. Selecciona el plano XY cómo el plano de interpolación circular.

G18. Selecciona el plano XZ cómo el plano de interpolación circular.

G19. Selecciona el plano YZ cómo el plano de interpolación circular.

G30: Cancela el punto simétrico, se usa en conjunción con G31 y G32

G31. Crea un punto de simetría con respecto al eje Y.

G32. Crea un punto de simetría con respecto al eje X.

G70. Indica que el sistema de medición que se va a usar es el inglés (pulgadas)

G71. Indica que el sistema de medición que se va a usar es el decimal.

G74. Indica Interpolación circular en un solo cuadrante (arcos de hasta 90 grados)

Ejemplo: G90 G74 X0. Y2

G2 X2. Y0. J2.

construye un arco de 90 grados con radio de 2 unidades. El arco comienza en el punto X=0, Y=2 y termina en el punto X=2, Y=0.

G75. Indica interpolación circular en varios cuadrantes.

Ejemplo: G90 G75 X0. Y2.

se mueve al punto de inicio

G3 X0. Y2. I0. J0

construye un círculo completo con centro X=0, Y=0. El punto inicial está localizado en la posición X=0, Y=2 y termina en este mismo punto haciendo un movimiento en sentido contrario a las manecillas del reloj.

G77. Ciclo de careado

Ejemplo: G77 X2. Y.75 Y.25 F100

careo una pieza moviendo la cabeza de la fresa 2 unidades en la dirección de

X y 0.75 unidades en la dirección de Y, a la velocidad de 10 unidades/min. Utiliza un incremento de Y de 0.25 unidades por paso.

G.78 Ciclo de fresado interior con avance definido

Ejemplo: G78 X.80 X.25 X.05 Y.50 Y.15 F400

Internamente hace un cuadro moviendo la fresadora un total de 0.8 unidades en el eje X y 0.5 unidades en el eje Y.

Utiliza un incremento inicial de 0.25 unidades en X, de 0.15 en Y y una velocidad de 40 unidades/min. Continúa hasta que la fresadora ha procesado 0.75 unidades en X y 0.45 en Y. Adicionalmente se mueve 0.15 unidades en cada dirección y realiza una pasada final a 60 unidades/min. Al terminar vuelve a la posición inicial.

G79. Esta función es utilizada para hacer círculos mayores que el diámetro de la fresadora

Ejemplo: G79 J1.0 F100

Internamente hace un hoyo el cual tiene un radio de 1 unidad (más el radio de corte) a una velocidad de 10 unidades/min.

G80. Cancela las instrucciones G81 a G89.

G81. Realiza una serie de perforaciones en forma cíclica

Ejemplo: G81 X4.0 Y5.0 Z1.2

X5.0

X6.0

hace un hoyo de 1.2 unidades de profundidad en X=4, Y=5; después hace un hoyo de X=5, Y=5; finalmente hace un hoyo en X=6, Y=5.

G82. Similar al G81, pero se le especifica el tiempo que permanecerá en el fondo del agujero (con G4)

G83. Realiza un orificio a intervalos.

Mientras está cortando el material, la velocidad de corte es especificada con la instrucción "feed" de alimentación y la salida es a velocidad rápida.

Primero a partir de la posición inicial, penetra lento (de acuerdo al feed) hasta una profundidad inicial, para después regresar. De ahora en adelante penetrará un incremento cada vez, hasta alcanzar la posición final indicada.

Ejemplo: G83 Z1 Z2 Z3

donde Z1 es profundidad total

Z2 es profundidad inicial

Z3 es el incremento

G84. Es similar a G81 con la diferencia de que las velocidades de entrada y de salida de la fresadora son controladas por la especificación de alimentación (feed).

G85. Ejecuta ciclos de perforado con una velocidad de entrada de alimentación F, para una misma velocidad de salida. El valor de la coordenada Z, se toma de manera incremental durante la ejecución, regresando a su modo original una vez terminada dicha operación.

G86. Realiza ciclos de perforado con un movimiento rápido a la posición de trabajo, para una perforación una velocidad de alimentación. Después realiza una espera de parte del operador para terminar el ciclo con una salida rápida. La espera es con el fin de realizar ajustes durante la ejecución del ciclo.

G87. Esta instrucción permite realizar ciclos de perforado realizando una entrada inicial a velocidad de alimentación, para después hacer varios ciclos de perforado para un determinado incremento de profundidad. La profundidad total se le proporciona previamente, La coordenada en el eje Z es tomada de manera incremental. La salida entre incrementos de perforados se realiza rápidamente y para un valor de 0.05 pulgadas.

G89. Ejecuta ciclos de perforado pero con un tiempo de pausa. Primero se posiciona rápido en el punto, después entra a velocidad de alimentación y espera un tiempo previo asignado por el operador; finalmente sale con velocidad de alimentación.

G90. Posicionamiento absoluto, todos los movimientos se refieren al origen X=0, Y=0, Z=0

G91. Posicionamiento incremental, los movimientos se refieren al punto anterior.

G92. La posición absoluta del eje especificado es asignada igual al valor definido por G92.

Ejemplo: G92 X0. Y0. Z0.

la posición presente es asignada como el origen.

FUNCIONES M

M0. Paradas de programa para inspección.

Se ejecuta cualquier instrucción al terminar aparece el mensaje "PARADA DEL PROGRAMA POR INSPECCION. OPRIMIR <RET> PARA CONTINUAR".

M1. Parada del programa por inspección.

Es la misma función que M0, sólo que M1 se ejecuta si el parámetro de parada opcional está en ON.

M2. Parada del programa para volver a empezar. Primero el eje Z es llevado a HOME con velocidad rápida, y luego se ejecuta cualquier función X,Y. Después de este movimiento aparece el mensaje "PARADA DE PROGRAMA OPRIMA <RET> PARA VOLVER A EMPEZAR".

M6. Cambio de herramienta.

Debe tener como parámetros el número de herramienta y la velocidad.

Primero el eje Z se mueve hacia HOME con movimiento rápido, después cualquier movimiento X,Y se ejecuta. Al completarse, se toman como referencias nuevas el número de herramienta y la velocidad del husillo. A continuación aparece el mensaje: 'INSERTAR HERRAMIENTA ADECUADA, OPRIMIR <RET> PARA CONTINUAR".

El eje Z se altera por la nueva longitud de la herramienta y el programa continúa.

M11, M12, M13. Salida directa de los ejes X,Y,Z.

Cada eje tiene 4 bits para hacer combinaciones de 0-15. El 15 envía corriente a todas las salidas. El 0 quita la corriente a las salidas, por ejemplo: M11/0 para apagarlas.

TORNO

M11/0 desactiva motor eje X.

M12/0 desactiva motor eje Y.

M12/1 chuck cerrado

M12/2 chuck abierto

M12/4 empujador hacia adelante.

M12/8 empujador hacia atrás.

M13/0 desactiva motor eje Z.

M14. Salida directa al eje auxiliar.

Funciona igual que las anteriores, sólo que en este caso las salidas están conectadas al motor del husillo, al sujetador y al refrigerante. Si se quieren todas encendidas utilizar M14/7.

FRESADORA

M14/0 ningún auxiliar encendido

M14/1 encender motor

M14/2 encender sujetador

M14/3 encender sujetador y motor

M14/4 encender refrigerante

M14/5 encender refrigerante y motor

M14/6 encender refrigerante y sujetador

M14/7 encender motor, sujetador y refrigerante.

TORNO

M14/0 parar el motor del chuck

M14/1 arrancar el motor del chuck

M15. Modo de continuación de salida directa

M15/0 Continuar normalmente después de M11, M12, M13, y M14 (default).

M15/1 Da corriente a los ejes después del modo de continuación.

M16. Método de continuación e interrupción del programa. Las instrucciones M16/1 hasta M16/8 especifican continuar después de recibir la señal de switch 1-8.

M16/9 Se usa en conjunción con la función G4 para ejecutar el tiempo de pausa definido en esta instrucción.

M16/10 Se usa para continuar las operaciones después de dar <RET>.

M16/0 Se usa después de cualquier función M15 para regresar al modo original.

M25. Eje Z a posición de HOME.

Antes de cualquier operación, el eje Z se lleva a HOME con movimiento rápido. Después cualquier movimiento X, Y es efectuado y el programa continúa.

Ejemplo: El siguiente programa muestra la utilización de los comandos G18 y G17

G71 G90 Coord. milimétricas y coord. absolutas.

G0 Z0 Mov. rápido sin cortar hasta Z=20

M14/2 Activar sujetador

G0 X-10 Y0 Mov. rápido sin corte hasta X=-10, Y=0

G0 Z-4 Mov. rápido hasta Z=-4, sin corte.

M14/7 Encender motor, sujetador y refrigerante

G1 Z-5 F1000 Movimiento de corte, hasta Z=-5 a vel = 100 in/min.

G18 Escoge el plano XZ para círculo.

G75 G2 X10 I0 K3 F400 Interpolación circular en dos cuadrantes en contra de las manecillas del reloj.

X=10 Punto final del radio.

I=0, K=3 Centro del círculo.

Velocidad = 40 in/min.

G0 Z0 Mov. rápido a Z=0.

G17 Escoger el plano XY para círculo

X0 Y0 Se posiciona en X=0, Y=0.
 M14/0 Apaga Chuck.
 M11/0 Desactiva el motor del eje X.
 M12/0 Desactiva el motor del eje Y.
 M13/0 Desactiva el motor del eje Z.

Ejemplo: El siguiente programa muestra como es posible utilizar los comandos G19 y G17.

```
G17 G90
G0 Z0
M14/2
G0 X0 Y-10
G0 Z-4
M14/7
G1 Z-10 F1000
G19 Escoger plano YZ para círculo.
G75 G2 Y10 J0 K-18 F400
G0 Z0
G17
X0 Y0
M14/0
M11/0
M12/0
M13/0
```

Ejemplo: El siguiente programa muestra como es posible utilizar los comandos G30, G31 y G32.

```
G71
G90
M14/2
G0 X-10 Y-10
Z-4
M14/7
G1 Z-6 F1000
G32 Crea un punto de simetría con respecto al eje X.
G1 X0 Y-20 F600
G30
G0 Z-4
X10 Y-10
G1 Z-6 F1000
G31 Crea un punto de simetría con respecto al eje Y.
G1 X20 Y0 F600
G30 Cancela el punto de simetría
G0 Z0
X0 Y0
```

M14/0
M11/0
M12/0
M13/0

Ejemplo para torno: El siguiente programa muestra como es posible utilizar los comandos M12/1, M12/2, M12/4 M12/8, M14/0 y M14/1

```
G71 G90
G0 Z0
X0
M11/0
M13/0
M12/2 Abrir Chuck
M12/4 Contrapunto hacia adelante
M12/1 Cerrar chuck
M12/8 Contrapunto hacia atrás.
G0 Z-8
X0
M14/1 Encender el motor del chuck
G1 Z-9
X-9
G0 X0
G18
G75 G3 X-9 Z-5 I-5 K-5 F800
G0 X0
M14/0 Parar el motor del chuck
M12/2
M11/0
M13/0
```

Ejemplo de macros: Los siguientes Macros definen el uso de códigos para generar un conjunto de caracteres para el CNC controller para realizarse en la fresadora. Cada letra o número en este conjunto de caracteres tendrá 0.25 ln. de ancho, 0.3 de alto y .03 1n. de profundidad.

```
#1; '1'
G1 G91 Z-.03 Mov. de corte con posicionamiento
incremental hasta Z=-.03.
X.2 Mov. de corte hasta X=.2.
X-.1 Mov. de corte hasta X=-.1.
Y.3 Mov. de corte hasta Y=.3.
X-.05 Y-.075 Mov. de corte hasta X=-.05, Y=-.075.
G0 X.2 Y-.225 Z.03 Mov. rápido sin corte hasta X=.2, Y=-.225 y Z=.03.
$
```

```
#11; 'A'
G1 G91 Z-.03
X.1 Y.3
X.1 Y-.3
G0 X-.15 Y.15 Z.03
G1 Z-.03
X.1
G0 X.1 Y-.15 Z.03
```

La manera correcta de poder utilizar las Macros al ejecutar un programa es la siguiente:

```
G0 G75 F200; INITIAL SETUP
G91 X-1.255 Y-.35; POSITION FOR 1
=#1
G91 X-2.875 Y-.35; POSITION FOR A
=#11
```

3.3 PROGRAMACION DEL ROBOT

El lenguaje usado para operar el robot educacional es el SCORBASE. Con propósitos educativos, este lenguaje puede está comprendido de tres niveles, los cuales han sido diseñados para un progreso continuo en el aprendizaje para la operación del robot.

El nivel del SCORBASE que se tiene en el laboratorio, corresponde a una de las nuevas versiones del nivel tres. Además de comprender las instrucciones de este nivel, contiene otras instrucciones que se usan en el sistema de visión.

La programación del robot se realiza desde un sistema más general que comprende: manejo del sistema de visión, programación del robot y sincronización del sistema de visión con el sistema de coordenadas del robot. Las siguientes páginas tratan únicamente de la programación del robot, el capítulo 5 trata del manejo del sistema de visión y la sincronización entre la visión y el sistema de coordenadas del robot.

El siguiente menú es la pantalla principal del módulo de control del robot. SE selecciona mediante la tecla f2 en el módulo de control del sistema (ver capítulo 5). Todas y cada una de las opciones que se presentan en los diferentes menús de este módulo, pueden ser seleccionadas presionando el número correspondiente a la instrucción o bien, la primera letra de la instrucción. En pocas ocasiones es necesario presionar dos letras. Las letras de selección de cada instrucción están al lado izquierdo de la instrucción y de un color más brillante.

ROBOT CONTROL MAIN SCREEN

ESHED ROBOTEC (1982) LIMITED

ROBOT MAIN MENU ROBOTVISION

1. TEACH POSITIONS
2. EDIT PROGRAM (OFF - LINE)
3. PROGRAM HANDLING
4. RUN PROGRAM
5. HOME
6. MAIN ROBOTVISION MENU

La opción que se desee puede ser seleccionada presionando el número de dicha opción o bien, presionando la primera tecla de la instrucción deseada. Por ejemplo, presionando la tecla "1" o bien la tecla " T ", se transfiere la operación al menú de TEACH POSITIONS. De la misma manera, el resto de la opciones pueden ser seleccionadas. A continuación se da una explicación de cada una de las opciones que se presentan en el menú y sucesivamente, se explicarán las opciones en los menús disponibles.

Explicación:

TEACH POSITIONS: Esta opción nos lleva a la pantalla del programa donde se nos permite enseñar posiciones al robot, las cuales pueden ser grabadas para un manejo posterior en la operación del sistema.

EDIT PROGRAM (OFF - LINE): Esta opción nos lleva a la pantalla donde se pueden editar y/o modificar los programas del robot.

PROGRAM HANDLING: Esta opción nos lleva a la pantalla donde se nos permite salvar, cargar y borrar los programas del robot. Los programas pueden ser guardados en disco flexible o disco duro y son cargados de estas unidades al controlador del robot para la ejecución de los movimientos.

RUN PROGRAM: Esta opción nos lleva a la pantalla donde se permite ejecutar los programas.

HOME: En esta opción, se permite llevar al robot a la posición de referencia de home para que quede definida como la posición de referencia absoluta y a partir de esta posición, sean definidas las demás posiciones.

MAIN ROBOTVISION MENU: Seleccionando esta opción se regresa al menú general del sistema.

TEACH POSITIONS SCREEN

<- ACTIVE ->

1/Q	Move Base Right/Left	RP	Record Position...
2/W	Move Shoulder Up/Down	RR	Record Relative Position...
3/E	Move Elbow Up/Down	TP	Teach Position... (xyz)
4/R	Move Wrist- Pitch Up/Down	TR	Teach Relative Position... (xyz)
5/T	Move Wrist-Roll Right/Left	TO	Teach Pos. Relative to Object
6/Y	Move Axis 6 +/-	SF	Scan Frame
7/U	Move Axis 7 +/-	SO	Scan for Object #. in Frame #.
OG/CG	Open/Close Gripper	N	Next Object #...
F/S	Fast/Speed 0 Movement	GO	Go to Object #....
ON/OFF	Turn Output ON/OFF	GP/GH	Go to Position.../Go Home

<-- SPACE --> change active side L List / Delete Position <ESC> MAIN MENU

X	Y	Z	PITCH	ROLL	AX-6	AX-7
163.42	0.00	499.97	-64	52	0.00	0.00

El menú de teach positions está compuesto de dos ventanas que contienen diez instrucciones cada una de ellas. En un momento determinado, solo una de las ventanas está activada, la cual se indica mediante un rectángulo iluminado que rodea a dicha ventana. Además, aparece un letrero en la parte superior de la ventana que se encuentra activa. La selección de la ventana se hace mediante las flechas izquierda y derecha respectivamente y la barra espaciadora actúa como un intercambiador de las ventanas. Esto es, si la ventana izquierda se encuentra activa y se presiona la barra espaciadora, se activará la ventana derecha. De igual forma, la ventana izquierda se activará si se presiona la barra espaciadora.

Explicación:

1/Q	Teclas para movimiento der. o izq de la base del robot
2/W	Teclas para el movimiento hacia arriba y abajo del hombro
3/E	Teclas para el movimiento hacia arriba y abajo del codo
4/R	Teclas para el movimiento hacia arriba o abajo de la canilla
5/T	Teclas para el movimiento de giro de la canilla
6/Y	Teclas para el movimiento hacia adelante o atrás del motor 6
7/U	Teclas para el movimiento hacia adelante o atrás del motor 7
OG/CG	Teclas para abrir o cerrar las tenazas
F/S	Teclas para movimiento rápido o definir una velocidad en los movimientos del robot.
ON/OF	Sirve para operar una salida específica. Encenderla o apagarla

RP (RECORD POSITION) Grabar una posición. Se lleva al robot hasta la posición deseada y en ese estado se ejecuta esta instrucción. Las coordenadas de la posición son grabadas a partir de la posición de referencia absoluta (Home). Se pueden grabar hasta 100 posiciones para la operación del robot.

RR (RECORD RELATIVE POSITION) Grabar una posición en forma relativa a otra posición previamente grabada. Se tiene que dar como información en que coordenadas será el movimiento relativo, en que magnitud y con respecto a que posición.

TEACH POSITION RELATIVE TO POSITION

X (MM) = 0.00 0.00
 Y (MM) = 0.00 0.00
 Z (MM) = 0.00 40
 PITCH (DEG) = 0.00 0.00
 ROLL (DEG) = 0.00 -90
 RECORD POSITION (1-100) 12 RELATIVE TO POSITION 11

La posición número 12 estará definida 40mm arriba de la posición número 11 y con un ángulo de giro de -90 grados.

TP (TEACH POSITION ...(XYZ)) Enseñar una posición al robot mediante las coordenadas X, Y, Z las cuales tienen su sistema de referencia en la posición de referencia absoluta (Home). Los valores de las coordenadas son dadas por el usuario.

TR (TEACH RELATIVE POSITION ...(XYZ)) Enseñar una posición relativa a otra mediante el sistema de coordenadas absolutas. En la instrucción RR, la posición relativa queda definida por los incrementos o decrementos en las coordenadas. En esta instrucción, la posición relativa queda definida a partir del sistema de coordenadas absoluto de referencia, es decir, con respecto al home del robot.

TO (TEACH POSITION RELATIVE TO OBJECT) Enseñar una posición al robot. Este tipo de posición es definida a partir de la localización de algún objeto específico. Por ejemplo:

TEACH POSITION RELATIVE TO OBJECT

X (MM) = 0.00 0.00
 Y (MM) = 0.00 0.00
 Z (MM) = 0.00 40
 PITCH (DEG) = 0.00 0.00
 ROLL (DEG) = 0.00 -90
 RECORD POSITION (1-100) 12 RELATIVE TO OBJECT 2 Regular / Fixed

Esta posición será grabada como la posición número 12 y estará 40mm arriba del objeto 2 con un ángulo de giro de -90 grados a partir del ángulo de giro definido como referencia. Existen dos tipos de estas posiciones: Regulares y Fijas. Las Regulares estan definidas como posiciones tales que el ángulo de giro de la tenaza será perpendicular a la orientación del objeto. Las posiciones Fijas son definidas con un ángulo de giro de 0 grados.

Las instrucciones SF (Scan Frame), SO (Scan for Object), N (Next Object) y GO (Go to Object) serán tratadas en la sección de Edit Program.

LIST/DELETE POSITIONS

1. LIST POSITION
2. LIST FROM POSITION
3. DELETE POSITION
4. DELETE FROM POS... TO POS...
5. DELETE ALL
6. PRINT POSITIONS (TO PRINTER)

<ESC> - RETURN TO TEACH POSITIONS MENU

AX-1 AX-2 AX-3 AX-4 AX-5 AX-6 AX-7

(En esta parte, los valores de las coordenadas están dados en medidas angulares)

ROBOT PRESENT POSITION IS:

PRESS ANY KEY TO CONTINUE

Explicación:

LIST POSITION: Despliega las coordenadas, en todos los ejes, de alguna posición del robot

LIST FROM POSITION: Despliega las coordenadas de todas las posiciones a partir de una de ellas

DELETE POSITION: Borra una posición específica

DELETE FROM... TO...: Borra todas las posiciones especificadas

DELETE ALL: Borra todas las posiciones grabadas

PRINT POSITION: Imprime las coordenadas de alguna posición

EDIT PROGRAM (OFF-LINE)

OG/CG	Open / Close Gripper	SF	Scan Frame
GP	Go to Position... Fast / Slow	SO	Scan for Object #.. in frame #.
W	Wait... seconds	OV	Overview frame #..
II	If Input #. jump to ...	IV	If Visuccess jump to ...
ON/OF	turn Output #. oN / oFf	IO	If Object #... exists jump to ...
J	Jump to line #..	Go	Go to object #... fast / slow
SS	Set subroutine #..	N	Next Object
RS	Return from subroutine	SM	Set memory #.. to viscount
CS	Call Subroutine #..	SM	Set memory #.. to...
RMP	Remark: Print	SM	Set memory #.. to sensor
OM	On motor #. error, jump..c/s/o	IM	If memory #.<=>memory #.. jump...
SH	Set present position as home	SC	Set counter #.. to..
IL	If limit switch #. jump to ...	DC	Decrement counter
SA	Set axis #. to zero	IC	If counter #.. > 0 jump to..

Explicación:

O/C: Comandos para abrir o cerrar tenazas

1: Comando para mandar al robot a una posición grabada

I: Comando para insertar líneas al programa

X: Comando para remplazar una línea ya existente

L: Comando para transferir a la pantalla donde se listan o borran las líneas del programa

SF (SCAN FRAME #.) Cuando este comando es ejecutado, se realiza la segmentación de la imagen del marco definido. El análisis de la imagen es realizado de acuerdo a la aplicación de visión que se genera en el MODULO DE CONTROL DE VISION. El resultado de este análisis puede ser usado posteriormente en comandos como los siguientes.

'Go to object #...', 'If object #... exists...', 'If Visuccess ...'

Los parámetros del marco de la escena son dados en el módulo de visión en la pantalla de definición de parámetros CAMERA SETUP (Ver capítulo 5). Si esta instrucción se ejecuta sin haber definido el marco, se recibe un mensaje de error como el siguiente: ' FRAME #... NOT SET PROPERLY '. El número total de marcos que se pueden definir es 4. La instrucción siguiente

SCAN FRAME # 1

Efectúa la segmentación de la imagen del marco 1 para la identificación de los objetos en dicho marco.

SO(SCAN FOR OBJECT #. IN FRAME #.) Este comando es similar al comando anterior, solo que termina de ejecutar el análisis cuando el objeto del tipo especificado es identificado. Si el objeto mencionado no está presente en la escena, el análisis termina hasta después de analizar todos los objetos presentes en el marco definido. La secuencia del análisis es de arriba a abajo y de izquierda a derecha.

OV(OVERVIEW FRAME #) Cuando este comando es ejecutado, se efectúa la digitalización del marco definido, además, cuenta el número de objetos que se encuentren en la imagen. El conteo de los objetos será realizado de acuerdo a los valores máximo y mínimo de los parámetros definidos.

IV(IF VISUCCESS JUMP TO #)Este es un comando para brincar. Si visuccess es verdadero entonces se ejecutará el brinco. El valor asociado a visuccess será determinado de acuerdo a la última digitalización realizada. Por ejemplo, si el comando previo fué SCAN FRAME #..., entonces el valor de visuccess es verdadero solo si un objeto fué identificado durante la digitalización. Si la instrucción anterior fué SCAN FOR OBJECT #..., el valor de visuccess será verdadero si el objeto seleccionado fué encontrado durante la digitalización.

IO(IF OBJECT #... EXISTS JUMP TO #)Esta también es una instrucción para saltar de línea. El salto se ejecutará solo si el objeto mencionado fué detectado en la búsqueda o digitalización de la imagen.

GO(GO TO OBJECT #... FAST/SLOW)Este comando le indicará al robot que vaya por el objeto previamente definido. Esta instrucción solo será ejecutada cuando el objeto se encuentre en la imagen de la cámara de sincronización. El robot se moverá hasta el centro de gravedad del objeto con un pitch de -90 grados y un roll de 90 grados de la orientación del objeto. La localización en el eje Z del brazo del robot es el parámetro insertado en la sincronización entre el robot y el sistema de visión.

N(NEXT OBJECT #...)Este comando habilita al sistema a acceder más de un objeto del mismo tipo. Cuando la instrucción GO TO OBJECT #3 es ejecutada, el robot se mueve al primer objeto encontrado en la digitalización.

SM(SET MEMORY #... TO VISCOUNT)Viscount es el número de objetos encontrados en la imagen el cuál, será almacenado en memoria. Este valor será determinado por la última búsqueda realizada. Las dos instrucciones siguientes son análogas a esta instrucción. Se almacena en memoria un valor definido o la medida de abertura de la mano del robot. Este valor de abertura es determinado a través de un sensor.

IM(IF MEMORY #...<=> MEMORY #... JUMP ...)Este comando es para efectuar comparaciones entre los valores almacenados en memoria. Los operadores de comparación son todos aquellos que se usan con los símbolos mayor, menor e igual.

SC(SET COUNTER #... TO...)Asignar un valor un valor a un contador.

DC(DECREMENT COUNTER)Decrementar en uno el valor del contador.

IC(IF COUNTER #... > 0 JUMP TO)Comando de salto a otra instrucción por comparación entre el valor de un contador y un valor cualquiera.

LIST/DELETE PROGRAM

1. LIST LINE
2. LIST FROM LINE
3. DELETE LINE
4. DELETE FROM LINE. . . TO LINE
5. DELETE ALL
6. PRINT PROGRAM (WITH PRINTER ONLY)

<ESC> - RETURN TO EDIT PROGRAM MENU

PRESS ANY KEY TO CONTINUE

Explicación:

LIST LINE: Despliega una línea específica

LIST FROM LINE: Despliega todas las líneas a partir de una línea específica

DELETE LINE: Borra una línea específica

DELETE FROM TO: Borra todas las líneas comprendidas entre una línea inicial y una línea final

DELETE ALL: Borra todas las líneas del programa

PRINT PROGRAM: Imprimir en una impresora el programa

PROGRAM HANDLING MENU

1. SAVE PROGRAM
2. LOAD PROGRAM
3. DELETE PROGRAM
4. CATALOG

<ESC> RETURN TO MAIN MENU

Explicación:

SAVE PROGRAM: Guardar el programa de la memoria de la computadora a un disco externo

LOAD PROGRAM: Cargar un programa de disco a la memoria de la computadora

DELETE PROGRAM: Borra un programa

CATALOG: Lista todos los programas que existen en el sistema de la computadora y que son del robot

RUN PROGRAM MENU

1. RUN SINGLE LINE
2. RUN SINGLE CYCLE
3. RUN CONTINUOUSLY
4. JUMP TO LINE...

PRESS "G" KEY TO LET THE ROBOT RUN
PRESS "B" FOR IMMEDIATE BRAKE
ANY OTHER KEY FOR REGULAR STOP

<ESC> - RETURN TO MAIN MENU

Explicación:

RUN SINGLE LINE: Ejecuta el programa línea por línea. La ejecución de cada línea es realizada después de presionar la tecla de return.

RUN SINGLE CYCLE: Ejecuta una sola vez todo el programa.

RUN CONTINUOUSLY: Ejecuta continuamente todo el programa.

JUMP TO LINE: Salta a una línea del programa y a partir de esta línea continúa la operación del programa.

PRESS " G " TO: Presionando esta tecla se pueden ejecutar los programas.

" B " FOR IMMEDIATE BRAKE: Tecla para paros de emergencia sin perder la información. Se puede continuar presionando la tecla C o regresar al menú RUN PROGRAM MENU presionando la tecla M.

ANY OTHER KEY...: Presionando cualquier otra tecla de G y B, se brinca a la última línea del programa y el robot se para.

HOME MENU

En este menú se lleva al robot a su posición de home. Presionando la tecla G el robot ejecuta su subrutina para posicionarse en home. El resto de las opciones que aparecen en este menú son para mover al robot a lo más cerca posible de su posición de home.

1/Q	Move Base Right/Left
2/W	Move Shoulder Up/Down
3/E	Move Elbow Up/Down
4/R	Move Wrist- Pitch Up/Down
5/T	Move Wrist-Roll Right/Left
6/Y	Move Axis 6 +/-
7/U	Move Axis 7 +/-
OG/CG	Open /Close Gripper
F/S	Fast/Speed 0 Movement
ON/OFF	Turn Output oN/oFf
H	Set present position as Home
Z	Set gripper to zero

PRESS "G" KEY TO SYNCHRONIZE THE ROBOT
ANY OTHER KEY WILL STOP THE ROBOT

<ESC> - RETURN TO ROBOT MAIN MENU

CAPITULO 4

EL SISTEMA DE VISION

4.1 LOS SISTEMAS DE VISION [4].

Un sistema de visión es definido como un sistema que es capaz de analizar imágenes de escenas para el conocimiento del medio ambiente. Son clasificados en sistemas bidimensionales o tridimensionales. La operación de un sistema de visión puede ser dividida en las tres siguientes funciones:

1. *Adquisición de la imagen y digitalización*
2. *Procesamiento y análisis de la imagen*
3. *Interpretación*

Las funciones básicas de un sistema de visión están representadas en la siguiente figura (fig. 4.1).

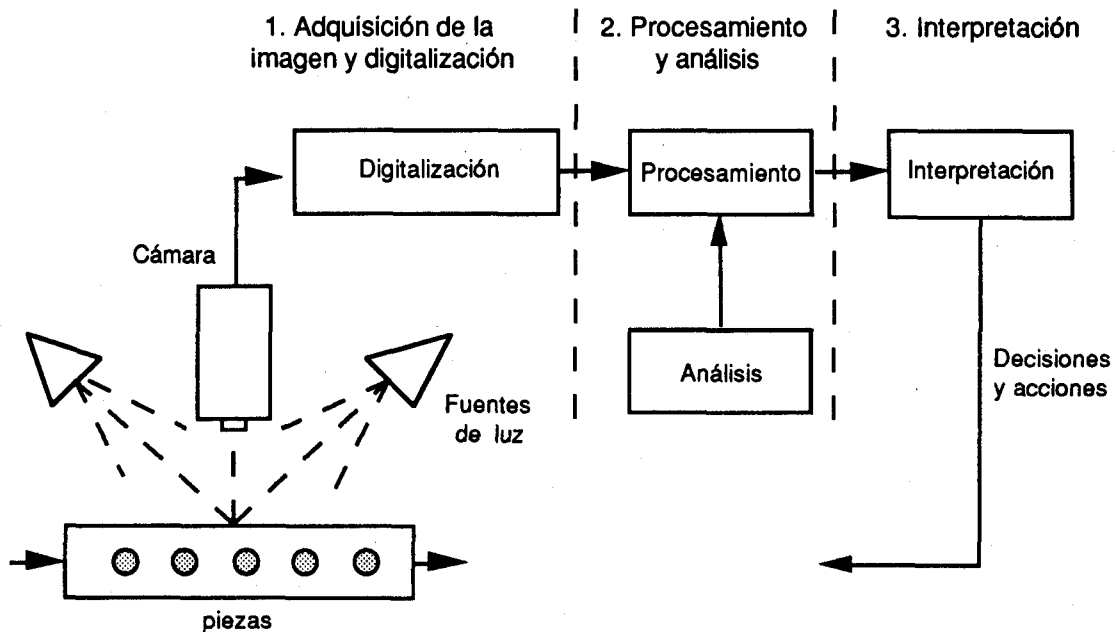


Fig. 4.1 Funciones básicas de un sistema de visión.

El análisis e interpretación de una imagen debe llegar hasta inferir los objetos que están presentes en la escena, incluyendo sus formas, tamaños, posiciones y orientaciones y en algunos casos, el color. Las imágenes son consideradas como una función que nos proporciona el valor de la intensidad de luz en cualquier punto de la imagen.

El valor mínimo de intensidad corresponde a una imagen oscura y el valor máximo corresponde a una imagen muy brillante. Naturalmente, la brillantez de las imágenes dependerá considerablemente de la iluminación de la escena por lo que la iluminación del medio ambiente es de una gran importancia en los sistemas de visión.

4.1.1 ADQUISICION Y DIGITALIZACION DE LA IMAGEN

La adquisición y digitalización de la imagen es realizada mediante el uso de cámaras de video y sistemas digitalizadores que almacenan la imagen para un análisis subsecuente.

La cámara es enfocada sobre el objetivo de interés y la imagen es obtenida dividiendo el área de vista de la cámara en una matriz de elementos discretos llamados píxeles, en la cual cada elemento tiene un valor que es proporcional a la intensidad de luz de la escena.

Los valores de intensidad de luz también son referidos como niveles de gris de la imagen. Los valores de intensidad para cada píxel son convertidos a un valor digital equivalente mediante un convertidor análogo digital.

La operación de adquirir la imagen de la escena resulta más sencilla si los objetos de la escena contrastan sustancialmente con el medio ambiente en que se encuentren. La figura 4.2 muestra la imagen obtenida de una escena mediante un sistema de visión, conocido como VISION BINARIA.

En la visión binaria, la intensidad de luz de cada uno de los píxeles es finalmente reducida a dos valores, blanco o negro, pudiendo depender del hecho de que la intensidad de luz de cada píxel sobrepase un valor dado de intensidad de luz conocido como umbral.

Los sistemas más sofisticados son capaces de distinguir y almacenar diferentes sombras o niveles de gris de una imagen. Estos sistemas son llamados Sistemas con Escala de Gris.

Estos sistemas no solo pueden detectar la parte exterior de los objetos, también pueden detectar textura y en muchas ocasiones, el color de los objetos. Típicamente usan 4, 6 u 8 bits de memoria para los niveles de gris. Ocho bits corresponden a $2^8 = 256$ niveles de intensidad.

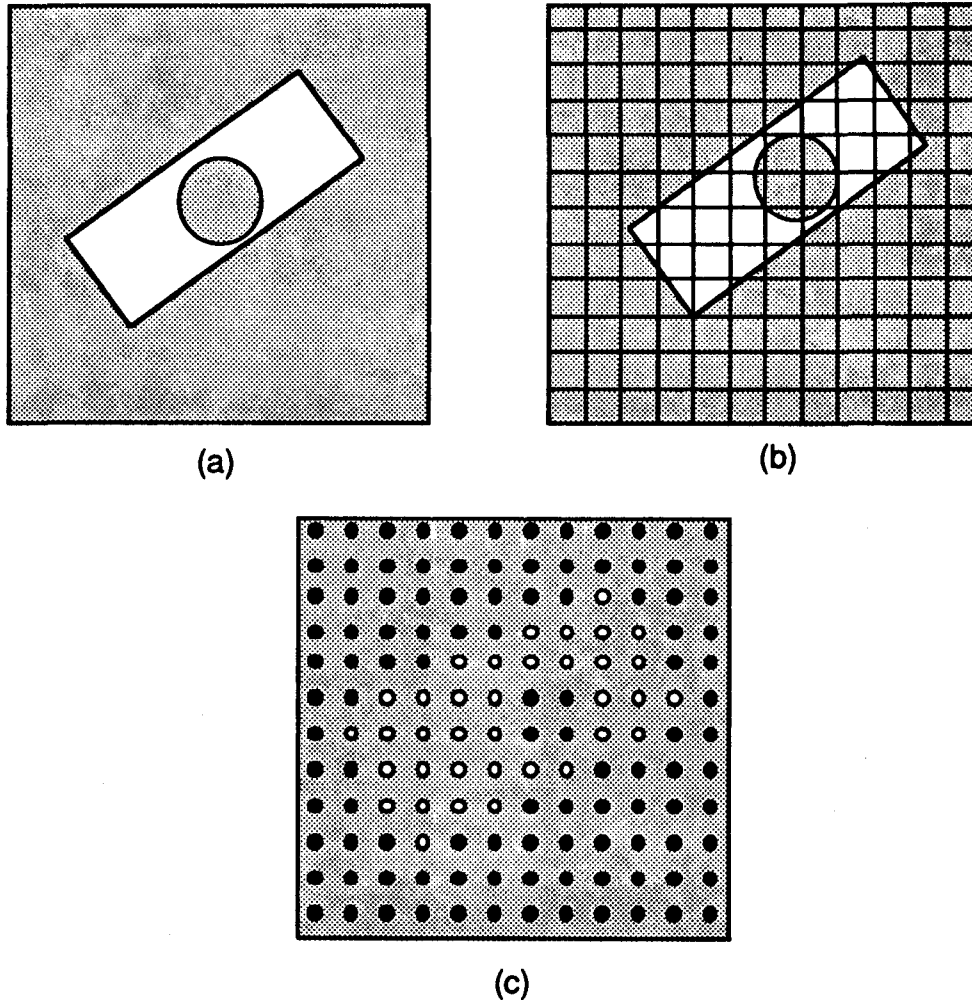


Fig. 4.2 División de la imagen en una matriz de elementos donde cada elemento tiene un valor de intensidad correspondiente a la porción de la imagen. (a) La escena; (b) Matriz de 12x12 elementos; (c) Valores discretos para cada uno de los niveles de intensidad de los elementos de la imagen.

Cada conjunto de píxeles es referido como un marco (frame). Cada marco, consiste de un conjunto de valores digitalizados y son almacenados en la memoria de la computadora.

Dos tipos de cámaras son usadas en las aplicaciones de los sistemas de visión: Cámaras Vidicón (del tipo usadas en televisión) y Cámaras de Estado Sólido.

Las cámaras vidicón operan mediante un haz de electrones y un circuito de deflexión que enfoca la imagen en una superficie fotoconductive y barriendo la superficie con el haz electrónico para obtener los valores relativos de los diferentes píxeles. Diferentes áreas en la superficie fotoconductive tienen

diferentes niveles de voltaje correspondiendo a la intensidad de luz de las áreas. El haz de electrones sigue un barrido definido y divide la superficie en un gran número de líneas horizontales.

Posteriormente, lee las líneas de arriba a abajo. Cada una de las líneas es dividida en una serie de puntos. El número de líneas multiplicado por el número de puntos en cada línea da las dimensiones de la matriz de elementos o pixeles.

Las cámaras de estado sólido operan enfocando la imagen en un arreglo bidimensional de elementos fotosensitivos finamente espaciado. Los elementos fotosensitivos forman la matriz de pixeles como en el caso anterior.

Una carga eléctrica es generada por cada uno de los elementos de acuerdo a la intensidad de luz que poseen. Esta carga es acumulada en un dispositivo de almacenaje consistente en un arreglo de elementos de almacenaje correspondiendo uno para cada uno de los elementos fotosensitivos. Los valores de las cargas son leídas secuencialmente en el procesamiento y análisis de la señal de la imagen.

Las cámaras de estado sólido tienen grandes ventajas sobre las cámaras de vidicón en las aplicaciones industriales. Físicamente son más pequeñas y más resistentes, y la imagen producida es más estable. Las cámaras vidicón sufren distorsión porque el tiempo de barrido y lectura de los elementos es mayor que en las mismas operaciones en los elementos fotosensitivos. Sin embargo, las cámaras de vidicón poseen un gran número de elementos en la matriz de la imagen, y la resolución de los sistemas de visión es dependiente del número de pixeles.

Entre mayor sea el número de pixeles, mayor es el grado de resolución del sistema de visión. Sin embargo, el costo de las cámaras está en función del número de pixeles y el tiempo de procesamiento y lectura de los elementos de la matriz se incrementa con el número de pixeles. Típicamente, los arreglos o matrices de pixeles son de 128x128, 256x256 y 512x512.

Otro aspecto importante de los sistemas de visión es la iluminación. La escena vista por la cámara debe estar bien iluminada y la iluminación debe ser siempre constante. Estos sistemas casi siempre requieren de una iluminación especial y diferente a la del medio ambiente.

Existen numerosas formas de iluminar la escena de la imagen. Una forma de clasificar las posibilidades para la iluminación puede ser en base a la localización de la fuente de luz, éstas formas pueden ser: Iluminación frontal, Iluminación posterior e Iluminación lateral. Las tres formas para la posición de la fuente de luz son mostradas en la figura 4.3

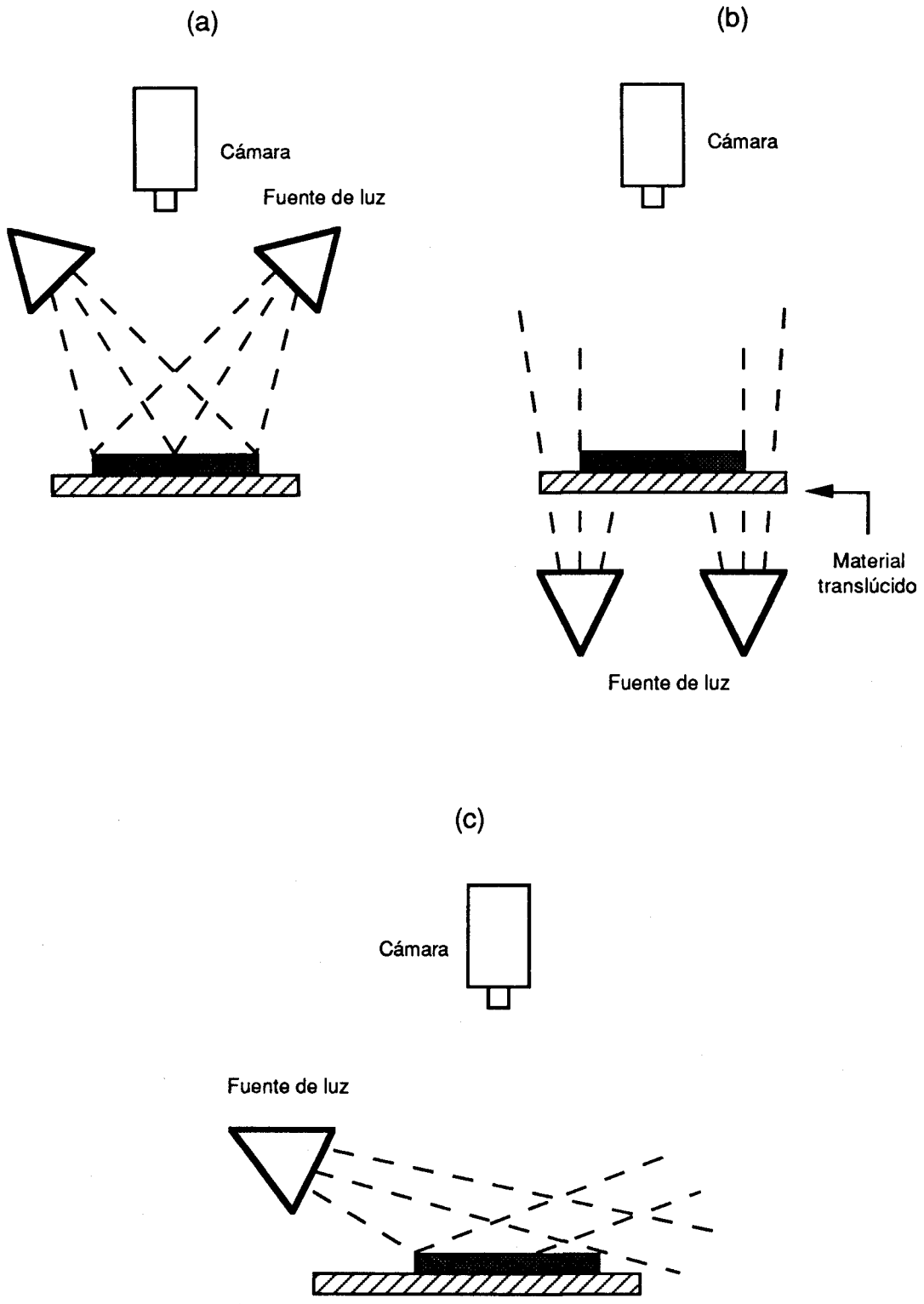


Fig. 4.3 Tipos de iluminación (a) Frontal, (b) Posterior y (c) Lateral

En la iluminación frontal, la fuente de luz está localizada en el mismo lado que el objeto y la cámara. Esto produce una luz reflejada del objeto que permite la inspección de características de la superficie tales como etiquetado e impresión, continuidad de líneas en tarjetas impresas para circuitos electrónicos.

En la iluminación posterior, la fuente de luz está localizada atrás del objeto visto por la cámara. Esto crea una silueta oscura del objeto que debe contrastar fuertemente con el fondo o mesa de trabajo. Este tipo de iluminación puede ser usado para distinguir la forma exterior de los objetos.

La iluminación lateral es una técnica utilizada para la identificación de los objetos por medio de la sombra que proyectan. Se usa para la inspección de defectos y fallas en la superficie de los objetos.

4.1.2 PROCESAMIENTO Y ANALISIS DE LA IMAGEN

La segunda función en la operación de los sistemas de visión es el procesamiento y análisis de la imagen. Un gran número de técnicas han sido desarrolladas para analizar los datos que provienen de la imagen. Una categoría de técnicas en el procesamiento y análisis de imágenes es llamada segmentación (ver capítulo 7). Esta técnica tiene como objetivo definir y separar en regiones de interés la imagen. Dos de las técnicas más comunes de segmentación son: Método de Umbral y Detección de Bordes.

El método de umbral involucra la conversión del valor de cada pixel en un valor binario, representando únicamente dos colores, blanco o negro. Esto se hace comparando el valor de intensidad de luz correspondiente a cada uno de los pixeles contra un valor de umbral definido. Si la intensidad del pixel es mayor que el valor de umbral definido, se le da el valor binario de 1 representando de esta manera a un color blanco y si la intensidad es menor que el umbral, se le da el valor de 0 representando el color negro. De esta forma, la imagen queda reducida o representada por una matriz de valores binarios lo cual simplifica el problema de definir e identificar los objetos de la imagen.

Las diversas técnicas de detección de bordes centran su atención a la determinación y localización de las fronteras entre los objetos y la superficie en que éstos se encuentran en la imagen. Esto se realiza identificando el contraste en la intensidad de luz que existe en los pixeles adyacentes a los bordes o fronteras de los objetos. El capítulo 7 describe la técnica del umbral así como los algoritmos más utilizados en la técnica de detección de bordes.

Otro conjunto de técnicas desarrolladas para la segmentación es conocido como Extracción de Características. Estas características pueden ser área de los objetos, longitud, ancho, perímetro, centro de gravedad y otras mediciones. Los métodos de extracción de características están diseñados para determinar

características basadas en el área de los objetos y sus fronteras usando umbrales, detección de bordes y otras técnicas de segmentación. Por ejemplo, el área puede ser determinada por medio del número total de píxeles que están contenidos dentro de los objetos, la longitud es determinada por medio de la medición en píxeles entre dos puntos o extremos.

4.1.3 INTERPRETACION

Para una aplicación cualquiera, la interpretación de la imagen está basada en la extracción de las características de los objetos, sin importar el método o técnica utilizado para la segmentación. La función de interpretación es concerniente al reconocimiento de los objetos y es referida como Reconocimiento de Objetos o Reconocimiento de Patrones. El objetivo es identificar los objetos de la imagen por comparación con modelos definidos. Algunas técnicas usan máscaras para efectuar las comparaciones con la imagen a un nivel de pixel a pixel. Una de las principales dificultades que se presentan en este método es el alineamiento que tienen que tener en forma constante las piezas frente a la cámara.

4.2 APLICACIONES DE LOS SISTEMAS DE VISION.

Las aplicaciones de los sistemas de visión en los sistemas de manufactura están divididas en las siguientes cuatro categorías:

- 1. Inspección*
- 2. Identificación de partes*
- 3. Control visual*
- 4. Monitoreo de seguridad*

Por mucho, la inspección en el control de calidad es la categoría más grande. Se estima que la inspección constituye el 90% de las aplicaciones de los sistemas de visión. Los sistemas de visión instalados en la industria desarrollan una gran variedad de tareas de inspección. Las aplicaciones casi siempre son para procesos donde existe una gran producción en masa. Las tareas de inspección típicas incluyen las siguientes operaciones: Medición de dimensiones, Verificación de la presencia de componentes en productos ensamblados, Verificación del número y localización de agujeros en piezas, Identificación de defectos en la superficie de las piezas, Identificación de defectos en las etiquetas o impresión de los artículos.

4.3 ESPECIFICACIONES DEL SISTEMA DE VISION DE LA CELDA FLEXIBLE DE MANUFACTURA

El software del sistema de visión que se tiene para la celda flexible de manufactura puede operar conjuntamente con el robot para el manejo de la celda. Como sabemos, el robot puede interactuar con el resto de las máquinas y si agregamos el sistema de visión, se tiene un sistema que puede funcionar de una manera totalmente automática y además, puede reconocer las piezas de trabajo así como las operaciones que se les hayan realizado. El siguiente capítulo trata sobre la programación de todo el sistema.

ESPECIFICACIONES:

- Posee un controlador con una tarjeta digitalizadora. Esta tarjeta es el hardware que permite la operación de considerar la imagen como un arreglo bidimensional de elementos de intensidad de luz, conocidos como pixeles.
- El sistema es compatible con cualquier computadora IBM PC/XT/AT. Para la celda flexible, se tiene una computadora IBM XT con 640 K RAM, un disco duro de 20 Megas y una unidad de disco flexible de 720 K.
- La resolución de la tarjeta digitalizadora es 64,000 pixeles, considerados como una matriz de 320x200
- El número de niveles de intensidad o niveles de gris es de 64
- Consta de 2 cámaras RS-170 del tipo vidicon y 2 monitores monocromáticos. Un monitor es para observar las escenas y el otro es para observar la digitalización de las imágenes. Cada una de las cámaras puede ser seleccionada mediante el software para realizar una aplicación específica. También se puede trabajar con las dos cámaras a la vez.
- El software del sistema de visión está diseñado para almacenar en memoria hasta 255 imágenes. Cada imagen puede ser almacenada en disco duro o disco flexible para un análisis posterior.
- Tiene una interfase RS232 para la comunicación con el robot.
- La aplicación del software es por medio de menús amigables así como pantallas de acción para las operaciones que se deseen realizar.

EL SOFTWARE TIENE CAPACIDAD PARA DESARROLLAR LAS OPERACIONES SIGUIENTES:

- Preprocesamiento de las imágenes. Este preprocesamiento consiste en la aplicación de filtros en los archivos de las imágenes para eliminar el posible ruido de la señal.
- Análisis de histogramas para los niveles de intensidad. Se puede considerar que la señal recibida en el proceso de digitalización es una función que muestra los valores de intensidad de la imagen en cada punto. Estos valores pueden ser organizados en un histograma que represente la frecuencia de ocurrencia de cada nivel de intensidad de la imagen. Para una mejor comprensión al respecto, ver la primera sección del capítulo 7.
- Definición de umbral para la segmentación. En visión binaria, un valor de umbral debe ser definido para efectuar la comparación entre los niveles de intensidad de la imagen con este valor. Este tema es tratado en la primera sección del capítulo 7.
- Procesamiento de imágenes de objetos por medio de características como:
 - Cálculo de áreas. El área de un objeto se determina como el número de pixeles dentro del contorno del objeto. Puede ser usado como una característica para el reconocimiento del objeto.
 - Análisis de agujeros. El número de agujeros en los objetos y el área total de los mismos también puede ser usado como una característica de importancia para el reconocimiento de los objetos.
 - Cálculo del perímetro de los objetos. En el proceso de segmentación, puede ser determinado el valor del perímetro de los objetos. Al igual que el valor del área, también representa una característica importante.
- Permite al usuario definir y seleccionar los parámetros para el reconocimiento de los objetos. El sistema de visión que se tiene en el laboratorio, permite definir los parámetros de interés para efectuar la identificación de los objetos. Esto es, presenta la oportunidad de definir si el área, perímetro o el número mismo de agujeros son las características para la identificación de los objetos. En el siguiente capítulo se explica brevemente la forma en que se puede definir los parámetros mencionados y sus respectivas tolerancias.
- Permite la enseñanza de posiciones relativas a los objetos. Este punto es tratado en el siguiente capítulo.
- El número mínimo de líneas de los programas es de 3 y el número máximo es de 400 líneas

La figura 4.4 muestra todos los componentes del sistema.

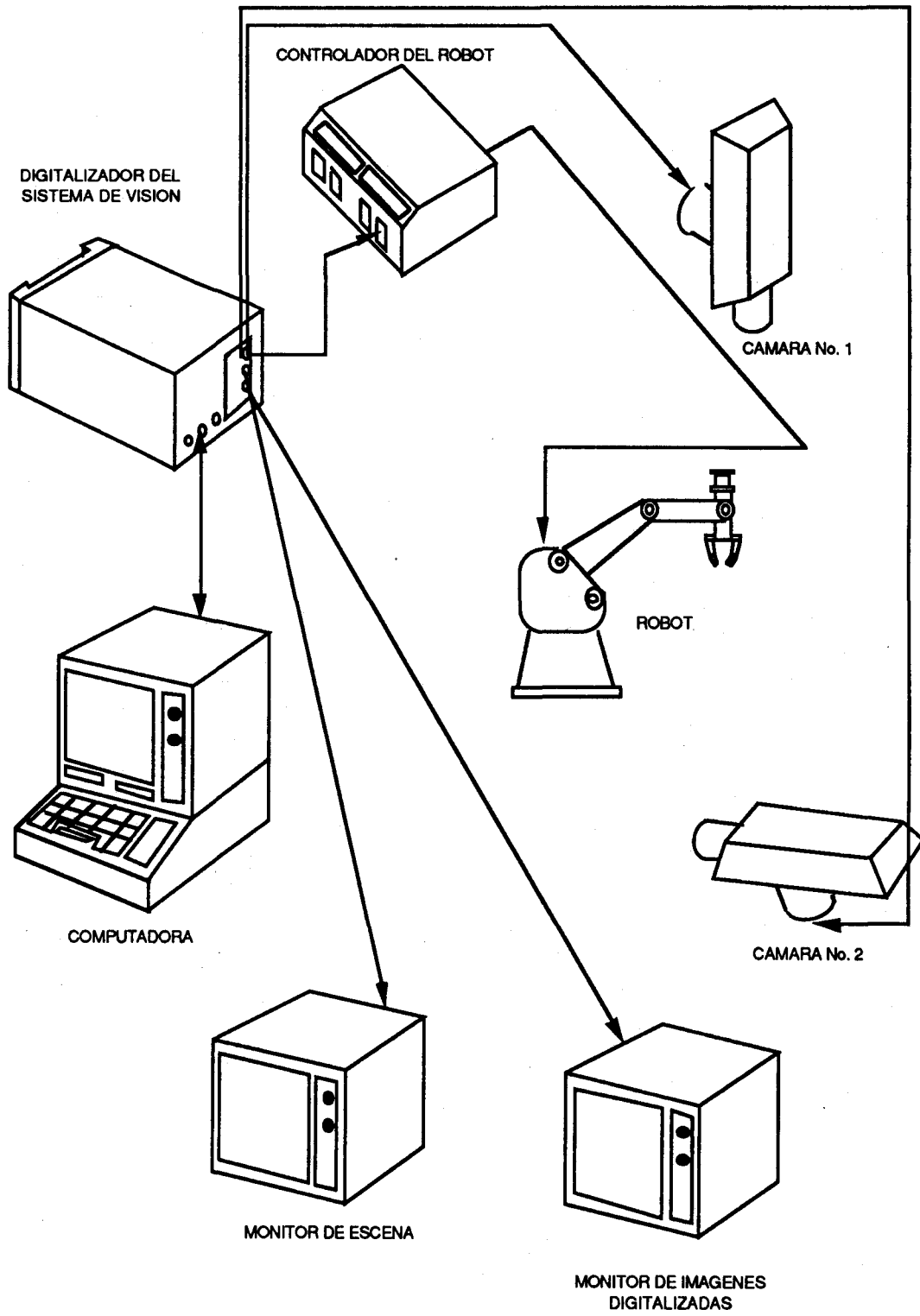


Fig. 4.4 Componentes del sistema de visión.

CAPITULO 5

PROGRAMACION DEL SISTEMA DE VISION

En el capítulo 3 se mencionó que el sistema de robotvisión está comprendido de tres grandes partes, las cuales son: Control del Robot, Control de Visión y Sincronización entre el robot y el sistema de visión. En la sección 3 de dicho capítulo se explicó la parte correspondiente al Control y Programación del Robot. Sin duda alguna, el programa del robot es el cerebro en la operación de la celda flexible de manufactura sin embargo, el programa del robot hace uso del sistema de visión, el cual tiene: parámetros que deben ser definidos antes de la operación, ajustes de las cámaras, definición de marcos para tomas de imágenes, definición de valores de umbral y otras variables de interés que se presentarán en este capítulo.

Además de las tres secciones anteriores, el sistema de robotvisión también tiene una sección referente a los archivos donde pueden ser almacenadas: las imágenes digitalizadas, la definición de parámetros, información de los parámetros de los objetos de la escena y otras variables. Esta parte es conocida como Archivo de Aplicaciones.

En este capítulo se mostrarán las secciones que componen al sistema, de una manera somera. Para obtener mayor información al respecto, ver el manual del sistema [3].

La pantalla principal del sistema de robotvisión es presentada en la figura 5.1

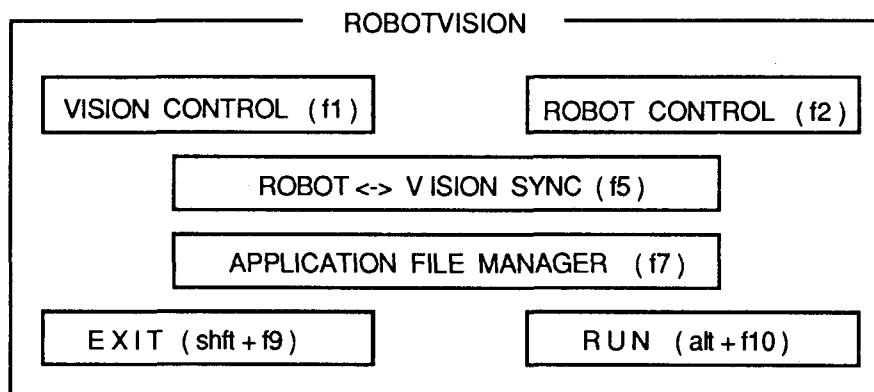


Fig. 5.1 Pantalla principal del sistema de robotvisión.

Como se puede ver desde este momento, el sistema de robotvisión es manejado mediante ventanas de fácil selección. Seleccionando la tecla correspondiente se puede entrar a cualquier aplicación. Para el caso del Control del Robot, la tecla correspondiente es f2.

Seleccionando la tecla f1, se puede entrar al control del sistema de visión para generar las aplicaciones de visión. Seleccionando la tecla f5 se entra al módulo para sincronizar las coordenadas del robot con las coordenadas del sistema de visión. Con la tecla f7, entramos al módulo de archivos para salvar o cargar aplicaciones o partes de aplicaciones. Presionando al mismo tiempo las teclas alt y f10, se transfiere la operación al módulo correspondiente para la ejecución de los programas integrando la visión al sistema. Con las teclas shift y f9, se puede salir del sistema y terminar la operación.

5.1 MODULO DE CONTROL PARA LA VISION

Con la selección de la tecla f1, entramos a la pantalla del módulo de control del sistema de visión. La figura 5.2 muestra la pantalla principal del módulo de control para la visión.

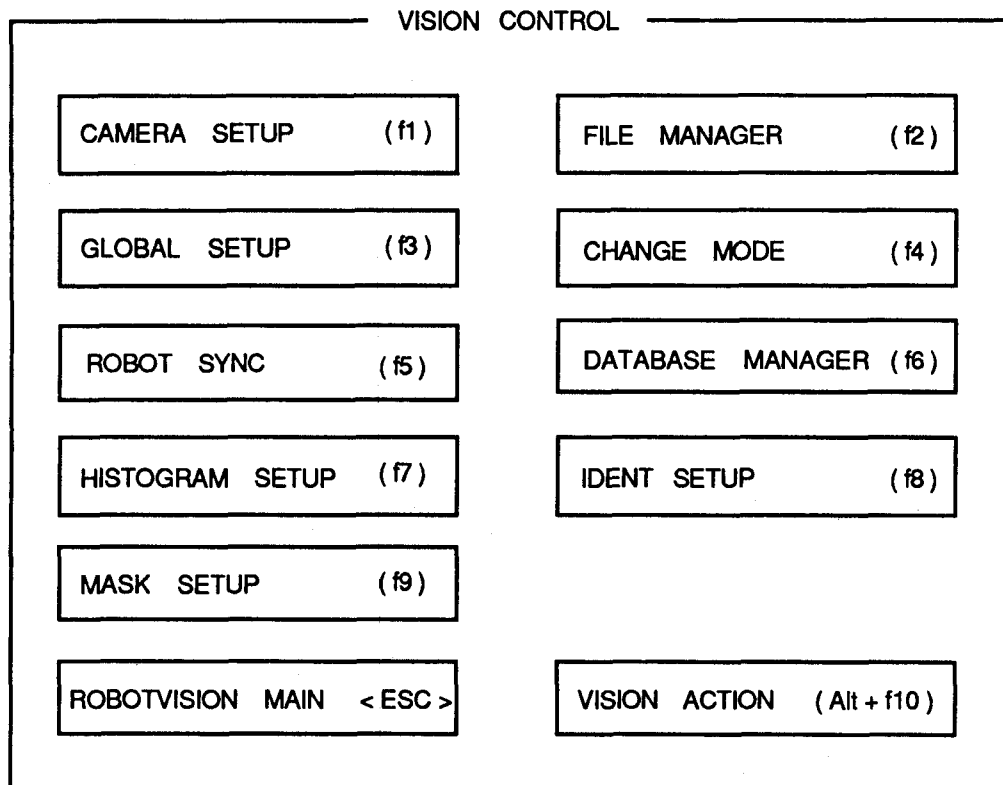


Fig. 5.2 Pantalla principal del módulo de control de la visión.

En este módulo, se definen todos los parámetros y variables de interés para el sistema de visión. La explicación de cada una de las ventanas es dada a continuación.

CAMERA SETUP (f1)

En esta ventana del módulo de visión se definen:

Las cámaras que serán usadas en el sistema. El sistema soporta hasta 4 cámaras diferentes. La definición de las cámaras queda establecida al asignarles un número de identificación.

Definición de los marcos para las imágenes. Se pueden definir hasta 4 marcos diferentes. Las medidas de los marcos, largo y ancho, también pueden ser definidas. La magnitud máxima para el marco son: longitud máxima = 320 y ancho máximo = 200. Además de las medidas del marco, se define el punto en que inicia el marco.

Definición de los niveles de negro y blanco. El sistema permite que estos niveles puedan ser variados observándose en el monitor como se presentan los cambios en los niveles para una mejor operación. Los niveles de negro y blanco pueden ser definidos en cada uno de los marcos. El rango para los valores es de 0 a 63 donde el nivel de blanco es mayor que el nivel de negro.

Además de los parámetros anteriores, en esta sección se puede digitalizar la imagen y almacenarla en memoria en algún archivo.

FILE MANAGER (f2)

En esta sección se puede salvar y cargar los archivos de aplicaciones realizadas o partes de aplicaciones. Cada aplicación consta de cuatro archivos (files) de interés, los cuales son: Setup File, Database File, Picture File y Robxyset File.

En el Setup File se almacenan todos los parámetros referentes a la operación del sistema de visión, tales como los niveles de gris, marcos, filtros para la segmentación de la imagen, formas y parámetros para la identificación de los objetos, nivel de umbral, etc.

El Database File es usado para almacenar los valores de las variables referentes a los objetos que son analizados. Estas variables son área, perímetro, orientación en el sistema de coordenadas de la celda, número de agujeros, área de los agujeros y otras variables de forma de los objetos.

Para almacenar una imagen digitalizada, se usa el Picture File. La imagen es almacenada como un conjunto de valores discretos. Se puede hacer uso de este archivo para graficar en un histograma de intensidad las frecuencias de ocurrencia de los diferentes valores de gris de la imagen.

Finalmente, el Robxysset File es usado para guardar las coordenadas y los parámetros de la sincronización entre el robot y el sistema de visión.

Los archivos pueden ser cargados y/o almacenados en un disco flexible o bien, en el disco duro del sistema.

GLOBAL SETUP (f3)

Esta parte del sistema es usada para indicar las variables que deben ser determinadas en el análisis de los objetos. Estas variables son:

Brillantez de los Objetos de trabajo. Se le indica al sistema si los objetos son más brillantez o más oscuros que el fondo o mesa de trabajo.

Tamaño de los Objetos. Las medidas máxima y mínima para los objetos, que permite el sistema, son: 0 y 63 999 pixeles. Se pueden definir estos valores de acuerdo a los objetos de trabajo y el sistema solamente considerará aquellos objetos que detecte entre tales valores.

Determinación del Centro de Gravedad de los Objetos y del Perímetro. Se presenta la opción de indicar al sistema si se desea o nó, que se encuentre el centro de gravedad de los objetos y su perímetro.

Analizar los Objetos en base a Parámetros de Forma y Agujeros. Estos parámetros de forma son definidos como razones entre área y perímetro, área total entre área de los agujeros y número de agujeros.

También se presenta la opción de cargar el Archivo Setup de alguna aplicación para realizada.

CHANGE MODE (f4)

En esta parte, se define el modo de operación del sistema, el cual puede ser:

LEARN: En este modo, el sistema desarrolla un análisis completo de los objetos que se presentan en la imagen. Después de analizar cada objeto, el sistema pide un número de identificación de cada objeto (ID NUMBER).

IDENTIFY: En esta caso, la identificación de los objetos se realiza de acuerdo a las especificaciones del usuario.

OVERVIEW: En este modo de operación el sistema solamente localiza los objetos y los cuenta sin efectuar ninguna identificación.

También se le indica al sistema de donde va a tomar las características de los objetos para la identificación. El archivo de las características puede ser cargado de un disco flexible o bien, pueden ser consideradas de los objetos que detecten las cámaras. Es decir, se puede tener como fuente de información a un archivo ya grabado o a la imagen que presenten las cámaras.

Se presenta la opción de mostrar una tabla de descripción mientras se realiza la digitalización de la imagen.

ROBOT SYNC (f5)

Esta sección del sistema nos lleva a la pantalla donde se realiza la sincronización entre las coordenadas del robot y el sistema de visión. Esta parte es tratada en la sección 5.2

DATABASE MANAGER (f6)

Como se mencionó anteriormente, el Archivo Database contiene toda la información relevante a las características de los objetos identificados. En esta pantalla del sistema se puede realizar lo siguiente:

Cargar el archivo database de algún disco.

Salvar, ver, modificar, borrar e imprimir la información contenida en el archivo.

HISTOGRAM SETUP (f7)

Esta parte del sistema es usada para: Mostrar el histograma de frecuencias absolutas y absolutas acumuladas de la imagen.

Se puede definir un valor de umbral para una buena segmentación de la imagen. El valor de umbral puede ser sugerido por el mismo sistema de acuerdo a los niveles de gris de la imagen y a la iluminación de la escena. Para mayor información referente al histograma de frecuencias y al valor de umbral, ver el capítulo 7.

IDENTIFICATION SETUP (f8)

En esta sección se le indica al sistema cómo debe efectuar la identificación de los objetos durante la activación del modo IDENTIFY.

La identificación puede ser en base a: área, perímetro, agujeros y parámetros de forma como la redondez. Se pueden definir los valores máximo y mínimo de cada variable así como las tolerancias para cada uno de los parámetros y de esta forma validar las características.

MASK SETUP (f9)

La sección de filtros para el preprocesamiento de la señal se encuentra en esta sección. Se tienen seis filtros para el preprocesamiento, los cuales son:

LOW PASS FILTER (grey and binary display). El filtro paso bajo es un filtro que actúa sobre el archivo que contiene la información de la imagen digitalizada, localizando los niveles irregulares (ruido) y corrigiéndolos de la siguiente manera: Cada pixel es comparado con el valor promedio de sus vecinos y si la diferencia es muy grande, se reemplaza el valor de dicho pixel por el valor promedio de los vecinos. Este hecho, trae como consecuencia que la imagen sea un poco borrosa. Para evitar esto, el filtro asigna un valor máximo de diferencia para que los valores de los pixeles sean reemplazados por el promedio de sus vecinos. Este valor máximo es llamado " altura" del filtro paso bajo.

LOW PASS SMOOTH. Este filtro actúa de la misma manera que el filtro paso bajo con una altura de filtro de 0.

BINARY SMOOTH. Este filtro actúa sobre el archivo de la imagen binaria. Después de que los valores de los pixeles de la imagen fueron comparados con el valor de umbral (imagen binaria), este filtro busca pixeles que contrasten fuertemente en una región. Por ejemplo, un pixel blanco en una área totalmente negra. Es más rápido que los otros filtros pero, es menos poderoso debido a que en el proceso de binarización de la imagen se pierden muchos datos.

ERODE FILTER. Este tipo de filtro también actúa sobre el archivo de la imagen binaria. Busca irregularidades en los objetos. Por ejemplo, si se tienen dos objetos separados que se encuentran " unidos " por un pixel, este filtro los separa.

DILATE FILTER. Este filtro es complementario del filtro erode. Busca las irregularidades de los objetos y los une. Por ejemplo, si se tienen dos partes de un objeto separadas tal vez por las condiciones de luz, este filtro unirá dichas partes.

VISION ACTION (Alt + f10)

En esta sección es donde se le enseñan al sistema los diferentes objetos de interés para las aplicaciones. El modo de operación es el modo de aprendizaje o "learning".

Cuando se efectúa la identificación de los objetos, se presenta un reporte estático para indicar al usuario en que sección de la segmentación se encuentra. Esta sección puede ser: análisis de perímetro, agujeros, forma y localización.

La identificación de los objetos se realiza por comparación entre los valores de los parámetros definidos (área, perímetro... etc.) con los valores obtenidos en la segmentación. Cuando los objetos son identificados, el sistema pregunta el nombre del objeto y el número de identificación del mismo. El número de identificación puede estar entre 1 y 255, que es el número máximo de objetos que se permite.

5.2 PROCEDIMIENTO PARA LA SINCRONIZACION ENTRE LAS COORDENADAS DEL ROBOT Y EL SISTEMA DE VISION

De la pantalla principal del sistema de robotvisión, presionando la tecla f5 se transfiere la operación del sistema al módulo para la sincronización de las coordenadas del robot y el sistema de visión. Ambos sistemas tienen su propio sistema de coordenadas por lo que deben ser sincronizados.

La sincronización puede ser realizada una sola vez y almacenada en el Archivo Robxysset. Este archivo puede ser cargado para futuras aplicaciones siempre que el robot y la cámara de sincronización no sean movidas de sus respectivos lugares. El sistema está diseñado para trabajar hasta con 4 cámaras, pero solo una de ellas será con la que se realice la sincronización. Aún y cuando se pueden tener varios marcos definidos, la sincronización se realiza sobre el marco definido como el número 1. Después de realizada la sincronización, todos los marcos definidos en la cámara de sincronización automáticamente se ajustarán con el robot.

El proceso de sincronización es realizado colocando un objeto en un punto dado de la celda flexible de manufactura, de tal manera que pueda ser visto por la cámara de sincronización y además pueda ser tomado por el robot. De esta forma, se consideran las coordenadas del objeto para el sistema de visión y para el sistema de coordenadas del robot.

CUIDADOS PARA LA SINCRONIZACION:

1. La cámara de sincronización y el robot son colocados de tal forma que la cámara esté viendo el área de trabajo del robot.
2. La cámara de sincronización es colocada verticalmente, de tal forma que su vista sea hacia abajo. Una buena exactitud de la cámara incrementará la exactitud con que el robot llegue a las piezas.

3. El objeto utilizado para llevar a cabo la sincronización debe ser uno que ya fué analizado por el módulo de visión. Es decir, debe ser un objeto que ya ha sido identificado y por lo tanto, tiene un nombre en el sistema y un número identificador.
4. El robot debe partir de su posición de home.

La pantalla principal de este módulo se presenta en la figura 5.3

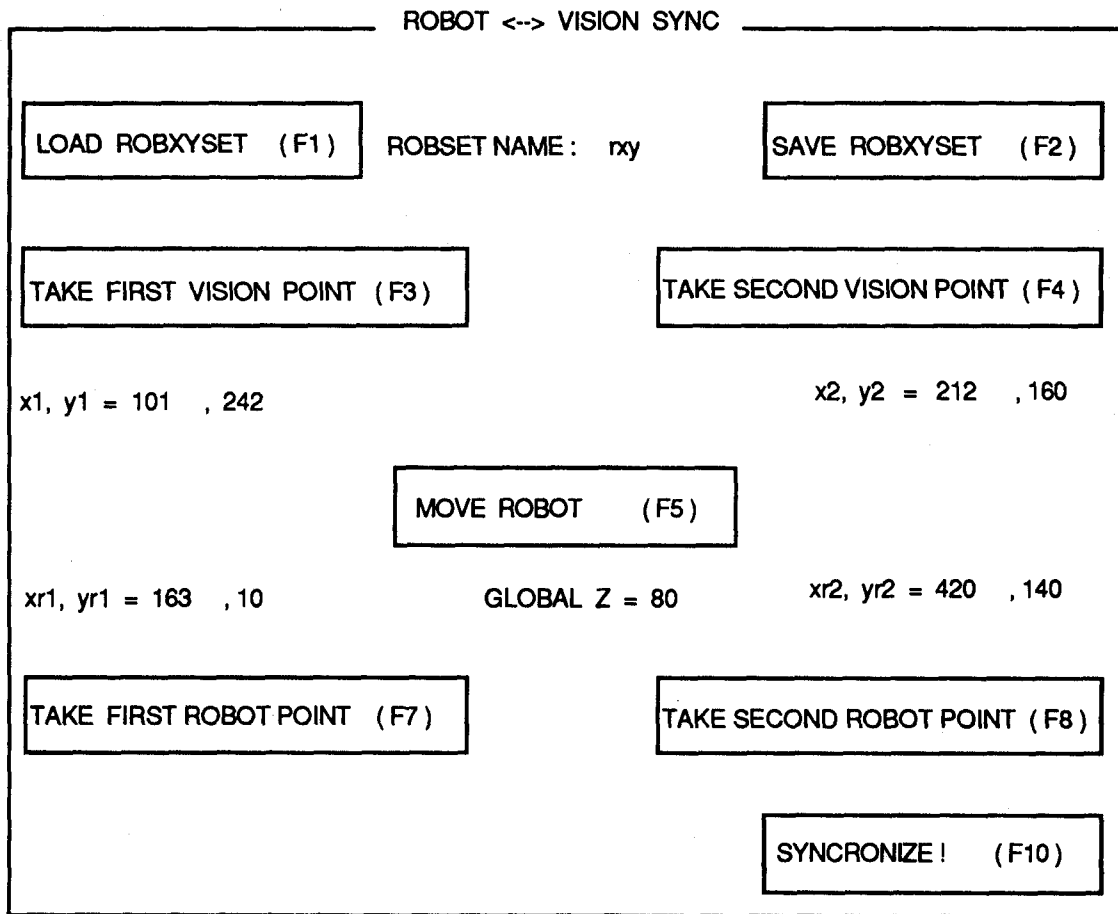


Fig. 5.3 Pantalla principal del módulo de sincronización.

Observe como la primer ventana que se tiene en la pantalla corresponde a una aplicación para cargar el Archivo Robxyset de algún disco. Esto se realiza presionando la tecla f1. Una vez que el sistema se sincroniza, las coordenadas de sincronización pueden ser almacenadas en un archivo y posteriormente llamarlas, evitando tener que realizar la operación de sincronización. Si las coordenadas de sincronización son cargadas de algún archivo, es importante verificar que tanto el robot como la cámara de sincronización no han sido

movidos del lugar en que se realizó la sincronización, ya que las coordenadas de sincronización dependen del punto en que se encuentre el robot y la cámara.

Presionando la tecla f2 se graba el Archivo Robxset. Se puede grabar en un disco flexible o bien en el disco duro. Considera al disco duro como la unidad base de almacenamiento si no se indica otra unidad de almacenamiento. El nombre esta determinado en la parte de ROBSET NAME: NOMBRE.

Si se desea que se grave en un disco flexible, se le tiene que indicar antes del nombre la unidad, de la siguiente forma: A: NOMBRE DEL ARCHIVO.

La operación de sincronización es llevada a cabo mediante la ejecución de dos pasos en los cuales se debe mover el robot a una posición donde se encuentre el objeto que se utilice. El robot debe estar en su posición de home y el objeto debe estar en una posición donde pueda ser alcanzado por el robot y además, la cámara de sincronización debe estar viendo al objeto de manera perpendicular. Bajo estas condiciones, se puede realizar el primer paso que consiste en:

PASO 1. Presionando la tecla f5 aparece en la pantalla una subpantalla para mover el robot, tal como se muestra en la figura 5.4.

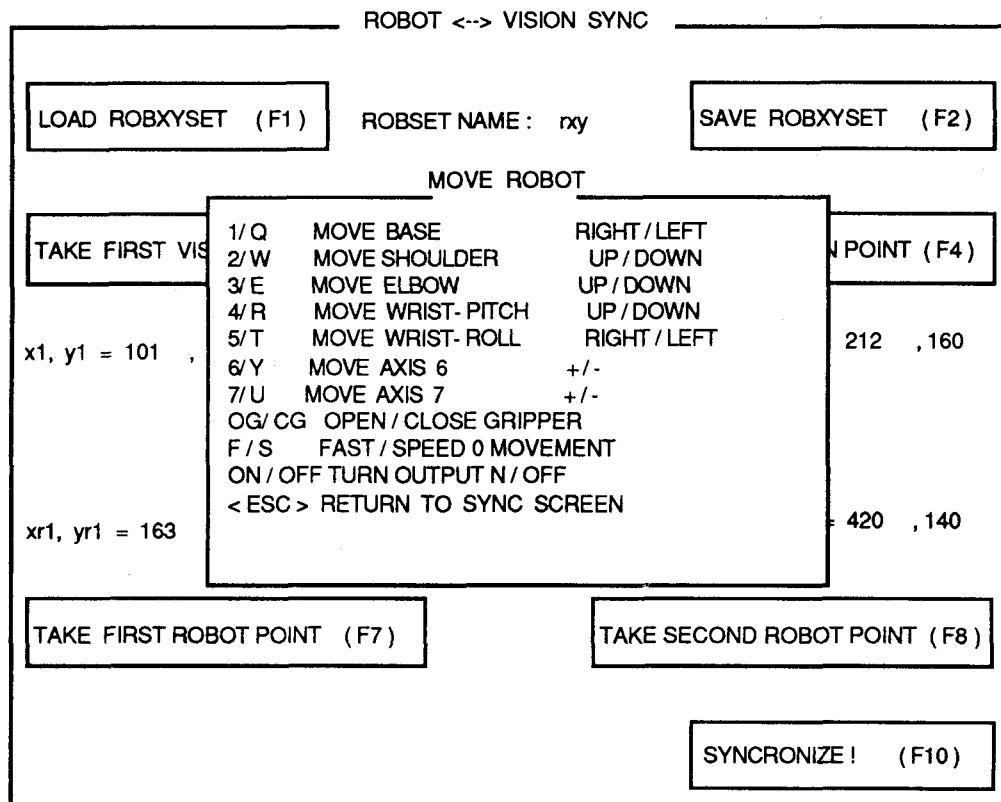


Fig. 5.4 Subpantalla para mover el robot a la posición de la pieza.

En este momento, el robot debe ser llevado hasta la localización exacta del objeto, incluso puede tomarlo para verificar que se encuentra exactamente en las coordenadas del objeto.

Una vez que el robot se encuentra en la posición del objeto, presionando la tecla <ESC> se regresa a la pantalla principal de sincronización y presionando la tecla f7 (Take First Robot Point) se graban las coordenadas del objeto en el sistema de coordenadas del robot. Teniendo cuidado de no mover el objeto, se retira el robot de la posición del objeto, utilizando la tecla f5, a un lugar que no obstruya la vista de la cámara. Así, la cámara se encuentra "viendo" al objeto sin alguna obstrucción. En este momento y en la pantalla de sincronización, se presiona la tecla f3 para grabar las coordenadas del objeto en el sistema de visión. En la pantalla se observa que el proceso de identificación del objeto se realiza correctamente.

PASO 2. Para realizar el segundo paso, se debe mover el objeto a otra posición. El segundo paso consiste en la realización de la misma operación del paso anterior. En este caso, se considerarán las teclas f8 y f4 para grabar las coordenadas del objeto en el sistema del robot y de visión respectivamente.

La operación final del proceso, una vez que se han tomado las coordenadas del objeto para cada sistema, es efectuar la sincronización entre los dos sistemas presionando la tecla f10. Si el proceso se realizó con éxito, se recibe el mensaje: SYNC OK ! pero si existió algún error, se recibe el mensaje: ERR IN SYNC ! , indicándonos que el proceso debe ser repetido.

Una de las desventajas del sistema de visión es el hecho de que la sincronización entre las coordenadas del robot y el sistema de visión solo se puede realizar con una cámara, aún y cuando el sistema es capaz de soportar hasta cuatro cámaras a la vez. Una posible mejora del sistema puede consistir en desarrollar el software necesario para efectuar la sincronización con cualquier cámara o bien, con dos de ellas donde una sincronización sea soporte de la otra.

5.3 PROCEDIMIENTO GENERAL PARA LA OPERACION DE LA CELDA FLEXIBLE DE MANUFACTURA, USANDO EL SISTEMA DE VISION

La celda flexible de manufactura tiene dos cámaras de vídeo y dos monitores blanco y negro. Una de las cámaras se encuentra sobre la banda de entrada y la otra se encuentra sobre la mesa intermedia de almacenamiento. En uno de los monitores se observa la escena y en el otro se presentan las imágenes digitalizadas de acuerdo al nivel de gris y al valor de umbral seleccionados.

En esta sección, describiremos en orden secuencial los pasos necesarios para la operación del sistema de la celda flexible de manufactura. El paso 1

forma parte del control del robot mientras que los pasos 2 al 6 y 8 corresponden a la parte de visión. El paso 7 corresponde a la sincronización entre los sistemas de coordenadas y el paso 9 se realiza en la parte del sistema donde se ejecutan los programas (RUN).

PASO 1. Generar el programa del robot definiendo el número de marcos que se usarán en la operación, así como el número de objetos diferentes con que se trabajará. Además, se deben grabar las posiciones necesarias para la aplicación. El programa del robot se realiza en la parte de ROBOT CONTROL (tecla f2) de la pantalla principal del sistema de robotvisión.

PASO 2. En la pantalla principal del sistema de robotvisión, presionando la tecla f1, se entra a la sección correspondiente a la parte de visión. En esta sección, presionando nuevamente la tecla f1, entramos a la parte correspondiente conocida como CAMARA SETUP. Aquí se tienen que definir los parámetros siguientes.

1. Cámara de sincronización. Normalmente se consideró que esta cámara debía ser la que se encuentra sobre la banda transportadora de entrada, por el hecho de que por esta banda son introducidos los objetos de trabajo a la celda flexible.
2. Definir el valor del nivel de gris con que se desea trabajar. Para una óptima operación, se puede cambiar en forma dinámica este valor. Esto significa que los cambios que sufrirá la imagen digitalizada por cambiar dicho valor, se observarán en pantalla del monitor correspondiente. El ajuste de este valor debe ser tal que la imagen del objeto sea lo más nítida posible y sin distorsión. Naturalmente, se presenta la opción de indicarle al sistema que digitalice la imagen para su observación, la cual aparecerá en el monitor.
3. Definir las medidas del marco donde se localiza la región de interés.

PASO 3. Regresando a la pantalla principal del sistema presionando la tecla <ESC>, presionamos la tecla f7 y entramos a la parte de HISTOGRAM SETUP. Aquí se tiene que definir el valor de umbral adecuado a las características de iluminación de la escena. El sistema de visión presenta la opción de determinar en forma automática este valor, de acuerdo al histograma de intensidad de la imagen que se presenta. Para más información respecto al umbral, ver la primera sección del capítulo 7.

Una vez definido el valor de umbral, regrese a la sección de CAMARA SETUP, con <ESC> y f1, e indique al sistema que digitalice la imagen para verificar que el valor de umbral seleccionado es el más adecuado para la imagen en consideración.

PASO 4. Con la tecla <ESC> regrese a la pantalla principal del sistema. En este momento, presione la tecla f9 y entre al módulo de MASK SETUP. En esta

parte del sistema de visión le indicará al sistema si se usará algún filtro. Los resultados de la aplicación de los diferentes filtros sobre la imagen se pueden ver al digitalizar la imagen y observarla en la pantalla.

PASO 5. Este paso consiste en indicarle al sistema los parámetros y tolerancias de importancia en la identificación de los objetos. La parte correspondiente del sistema para esta aplicación es la de IDENT SETUP, la cual se accesa mediante la tecla f8. Puede indicarle al sistema cual será el valor mínimo y máximo de cada característica, así como la tolerancia máxima entre los valores de las mismas, para identificar los objetos.

PASO 6. Presione las teclas ALT + F10 y entre al módulo de VISION ACTION. En esta parte del sistema de visión enseñará al sistema los diferentes objetos de trabajo. En la pantalla correspondiente a esta sección se presenta la opción de ejecutar la operación de aprendizaje, una vez, n veces o bien en forma continua hasta que se detecte un objeto con las características previamente definidas en los pasos 4 y 5 anteriores. Si se detectaron objetos y/o sombras, éstos son presentados en forma individual en la pantalla para el reconocimiento propio del usuario, de tal forma que es el usuario quien define que objetos serán grabados en memoria. Los objetos que seleccione el usuario recibirán un nombre y un número de identificación.

Al momento de ejecutar la sesión de aprendizaje, puede ser que el sistema no detecte objeto alguno. Esto situación se puede deber a una mala elección en el valor de umbral del sistema, por lo que es necesario corregirlo y efectuar nuevas pruebas hasta que los objetos sean detectados por el sistema. Esta situación se presenta en el caso en que se desee enseñar al sistema objetos que se encuentran en varias posiciones, por ejemplo en la banda de entrada y en la mesa intermedia de trabajo. En estos casos, el valor de umbral debe ser seleccionado de tal manera que se ajuste correctamente a las dos escenas: En la banda de entrada y en la mesa intermedia.

Cuando los objetos son detectados y grabados en el sistema, es el momento de grabar los archivos de parámetros y base de datos de los objetos (Setup File y Database File), así como el archivo de imagen (Picture File).

PASO 7. Saliendo de la sección de visión, con la tecla <ESC>, regresamos a la pantalla general del sistema de visión y presionamos la tecla f5 para entrar al módulo de sincronización entre las coordenadas del sistema del robot y el sistema de visión. Esta parte se encuentra explicada en la sección anterior de este capítulo.

PASO 8. Después de efectuar la sincronización entre los sistemas, es el momento de definir todos los marcos necesarios para la aplicación del programa del sistema. Estos marcos son definidos en la parte de visión, en la sección de CAMARA SETUP.

PASO 9. Por último, estando en la pantalla principal del sistema, presionando las teclas ALT + f10 se transfiere el control del sistema a la sección donde se ejecutan los programas. En la pantalla de esta sección se presenta la opción de ejecutar una sola vez el programa, n veces o continuamente. También se presenta en la pantalla la línea del programa que se está ejecutando.

CAPITULO 6**PROGRAMAS PARA LA OPTIMIZACION DEL SISTEMA****6.1 ¿POR QUE ES NECESARIO EL SISTEMA DE VISION?**

Como el sistema flexible tiene dos máquinas herramientas, torno y fresadora, es posible trabajar con piezas cuya forma sea la más adecuada a cada una de las máquinas o bien, con una pieza cuya operación a realizar se tenga que hacer en las dos máquinas. Pueden existir un gran cantidad de formas de las piezas para la operación pero en nuestro caso, se trabajó con las piezas mostradas en la figura 6.1 siguiente.

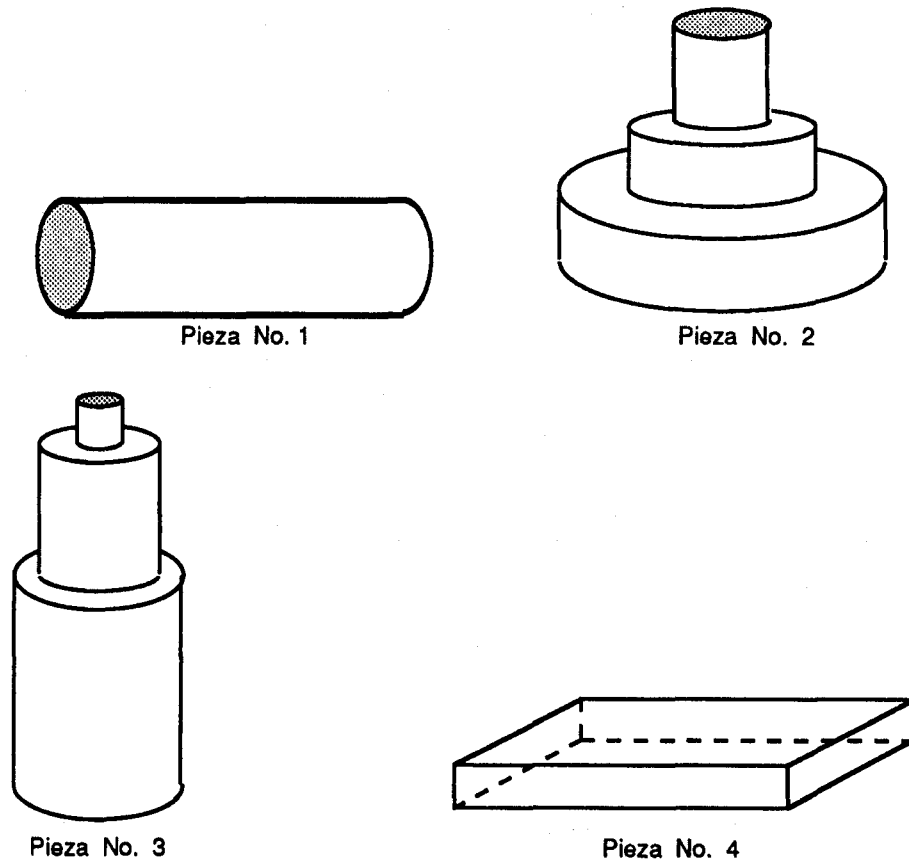


Fig. 6.1 Piezas utilizadas para la operación de la celda.

Las piezas 1 y 3 son piezas que sólo pueden ser maquinadas en el torno, mientras que la pieza 4 únicamente puede ser trabajada en la fresadora. Esto es debido a la configuración de las mordazas que se tiene en la mesa de trabajo de cada una de las máquinas. El torno tiene mordazas que se ajustan perfectamente bien a cuerpos de forma cilíndrica tal como las piezas mencionadas. Las mordazas de la fresadora tienen la forma de un cuadrado. La pieza 2 es la única que se puede maquinar tanto en el torno como en la fresadora a pesar de tener una forma cilíndrica. Se puede tomar en la fresadora debido a que su diámetro externo es de la misma medida que los lados del cuadrado que definen las mordazas.

Cuando una pieza de trabajo requiere una operación de maquinado tanto en el torno como en la fresadora, tal como la pieza 2, la lógica para desarrollar la programación es complicada y laboriosa. Para un trabajo de este tipo y sin el sistema de visión de la celda, los programas del robot requieren un gran número de contadores así como una gran cantidad de subrutinas. Esto se debe a que la pieza de trabajo tiene que ser llevada a la mesa intermedia una vez que se le realizó el trabajo en alguna de la máquinas para posteriormente pasarla a la otra. La mesa intermedia tiene cuatro posiciones disponibles para colocar las piezas.

La cosa parece sencilla, pero en realidad, se tienen que manejar varios contadores para determinar el estado de maquinado de la pieza en la celda. Dicho estado está definido como: Definir si la pieza acaba de entrar al ciclo de trabajo, definir si ya tiene algún tipo de maquinado (Torneado y/o Fresado) y definir su posición en la mesa intermedia.

Para eliminar estos problemas y determinar de una forma más rápida si la pieza presenta algún tipo de maquinado, se pensó que lo mejor sería usar un sistema de visión en la celda flexible. El sistema de visión que se integró a la celda, vino a resolver los problemas de programación y además, puede identificar las piezas de trabajo maquinadas o no, así como también puede recoger las piezas de entrada en la posición en que éstas se encuentren, sin la necesidad de colocar guías. Estas guías determinaban una posición única para las piezas de entrada y dicha posición era considerada como una posición más del robot. Así pues, con la ayuda del sistema de visión se puede determinar la posición de las piezas aún y cuando esta posición no permanezca fija entre una pieza y otra.

Mediante la ayuda del software del sistema de visión, se realizaron dos programas para el robot. Estos programas se presentan en el apéndice A de este trabajo y están basados en los diagramas de flujo que se presentan en la siguiente sección. El primer programa se realizó para piezas que tuviesen que ser trabajadas tanto en el torno como en la fresadora, tal como la pieza No. 2 que se presenta en la figura 6.1. El segundo programa fué hecho para piezas que sólo requieren maquinado en una de las máquinas, tal como las piezas 3 y 4 mostradas en la figura 6.1.

Ambos programas optimizan la operación de la celda por las siguientes razones:

- *El sistema de visión identifica las piezas con que se trabaja tanto en la banda de carga como en la mesa rotatoria. Esto evita el uso excesivo de contadores y subrutinas para determinar el estado de maquinado de la pieza en la celda. El programa realizado para piezas como la No. 2, tiene únicamente tres subrutinas donde, dos de ellas son necesarias y la otra es para buscar una posición vacía en la mesa intermedia.*
- *Mediante el sistema de visión se puede determinar la posición de cada pieza de entrada aún y cuando éstas no tengan una posición fija en la banda de entrada. El sistema de visión transfiere las coordenadas del objeto al robot para que sea éste el que recoja las piezas. Sin el sistema de visión, la posición de las piezas no debe de variar porque dicha posición es grabada como una posición del robot, la cuál debería permanecer fija.*
- *Las piezas que se coloquen en la mesa intermedia tienen prioridad sobre las piezas de entrada para ser terminadas. Esto es con el fin de tener la mayor producción posible de piezas totalmente terminadas. El sistema de visión detecta si las piezas han sido torneadas o fresadas, para llevarlas a la máquina por la cuál no han pasado.*
- *Si se está trabajando con la pieza No. 2 y éstas se llegasen a terminar, se indica mediante una señal al operador. Esta señal es desconectada cuando vuelve a entrar la pieza No. 2 y se continúa con la operación. Si alguna pieza diferente tal como la No. 3 y 4 se introdujo una vez que se terminó la pieza No.2, se avisa al operador de este hecho para que cambie los programas que se tienen por los programas para la pieza que entró. Estos avisos al operador se realizan encendiendo un foco.*

Si se trabaja con piezas diferentes no se presenta problema alguno, porque la pieza que ha sido trabajada en la máquina correspondiente puede salir inmediatamente de la celda a través de la banda de salida, sin necesidad de utilizar la mesa intermedia.

6.2 DESCRIPCION DE LOS PROGRAMAS PARA LA OPTIMIZACION DEL SISTEMA

Se realizaron dos programas para optimizar el uso de la celda flexible de manufactura con la ayuda del sistema de visión. Uno de los programas permite manejar piezas que tienen que ser maquinadas en torno y fresadora sin importar el orden en que esto ocurra. El otro programa maneja piezas que se deben maquinar sólo en el torno o sólo en la fresadora. En lo que sigue, se describe lo que estos programas realizan y sus diagramas de flujos.

6.2.1 PROGRAMA PARA PIEZAS MAQUINABLES EN TORNO Y FRESADORA (PIEZA No. 2).

El programa siguiente hace uso de 9 contadores definidos para determinar lo siguiente: Determinar si las máquinas productivas están maquinando alguna pieza; Determinar si la pieza que se encuentre en alguna máquina ya fué maquinada por la otra; Determinar si existe pieza en una posición intermedia y finalmente, para determinar que posición de la mesa intermedia está vacía u ocupada con alguna pieza, esto es posible determinarlo debido a que cada una de las posiciones de la mesa tienen asociado un contador. Por ello, el programa inicia asignando un valor de 0 a todos los contadores ya que no hay piezas en las posiciones. Cuando se deje una pieza en una posición, a su respectivo contador se le asigna el valor de 1. Los contadores asociados al torno son el 1 y 2 y los correspondientes a la fresadora son el 3 y 4, la posición intermedia tiene asociado el contador 5 y a las cuatro posiciones de la mesa rotatoria les corresponden los contadores 6, 7, 8 y 9.

Cuando alguna de las máquinas productivas solicita atención al manipulador, se determina a través de los contadores 1 y 3, respectivamente para torno y fresadora, si la llamada de atención es para llevarle una pieza o bien para retirar la pieza que tiene. Por ejemplo; si el torno solicita atención y el valor del contador 1 es mayor que 0, indica que la llamada es para que se le retire la pieza que tiene y, si el valor del contador es 0, indica que el torno no tiene pieza y por lo tanto la llamada de atención es para que el manipulador le lleve una pieza para trabajar.

Una vez que las máquinas solicitan atención y se determina que requieren una pieza, primero se busca en la mesa intermedia con auxilio del sistema de visión si hay piezas que no hayan sido trabajadas en dicha máquina; si existe alguna pieza que cumple con el requisito, es tomada por el manipulador y es llevada a la máquina. Si no existen piezas en la mesa intermedia para dicha máquina, se busca en la banda de entrada, con la otra cámara del sistema de visión, una pieza nueva o en bruto y si hay alguna pieza, el manipulador la toma y efectúa los movimientos necesarios para llevarla a la máquina herramienta.

Así pues, las piezas de la mesa intermedia tienen prioridad para ser terminadas. Los contadores 6, 7, 8 y 9 son utilizados únicamente para saber en que posición de la mesa hay una pieza y el sistema de visión efectúa el reconocimiento de dicha pieza. Cuando en la mesa intermedia y en la banda de entrada ya no existen piezas para cualquiera de las dos máquinas, se le indica el operador que el lote de piezas ya se terminó y, en caso de que posteriormente entrara una pieza igual, el programa continúa la operación. Si en la banda de entrada llegasen a existir piezas diferentes a las que se están maquinando, también se le avisa al operador que entró un lote de piezas diferentes para que cambie el programa del sistema y los de las máquinas herramientas una vez que todas las piezas del lote anterior hayan salido de la celda ya terminadas.

Si la llamada de atención que solicitan las máquinas herramientas es para retirar la pieza que tiene, verificando el valor del contador 2 para el torno o el 4 para la fresadora, se determina si la pieza debe ir a la mesa intermedia o la banda de salida. Un valor de 1 en estos contadores indica que la pieza ya fué trabajada en la otra máquina y por lo tanto irá a la banda de salida para su descarga de la celda.

Si los contadores tienen un valor de 0, la pieza deberá ser colocada en la mesa intermedia siempre que alguna posición esté vacía. Si todas las posiciones de la mesa estuvieran ocupadas, la pieza es colocada en una posición intermedia teniendo prioridad esta pieza de ser colocada en la mesa una vez que se desocupe alguna posición.

El diagrama de flujo mostrado en la figuras 6.2 (a) a 6.2 (g), se realizó para la operación del sistema donde se trabaje con piezas que tengan que ser maquinadas en ambas máquinas, tal como la pieza 2 presentada en la figura 6.1 anterior. Para esta misma pieza se definió:

- Objeto número 1 (ID. NUMBER 1). *Pieza maquinada en el torno y solicitada por la fresadora.*
- Objeto número 2 (ID. NUMBER 2). *Pieza maquinada en fresadora y por lo tanto, debe ser solicitada por el torno.*
- Objeto número 3 (ID. NUMBER 3). *Pieza nueva y que no tiene ninguna operación de maquinado. Se encuentra en la banda de entrada sin posición fija.*
- Objeto número 4 (ID. NUMBER 4). *Pieza que exclusivamente puede ser trabajada en el torno. Corresponde a la pieza No. 3 que se muestra en la figura 6.1.*
- Objeto número 5 (ID. NUMBER 5). *Pieza que únicamente puede ser trabajada en la fresadora y corresponde a la pieza No. 4 de la figura 6.1.*

Los objetos números 1 y 2 se definieron para un tipo de maquinado en particular, de acuerdo a los programas que se tenga en el torno y la fresadora. Estos programas pueden ser cambiados, de tal manera que la forma de estos objetos cambiará y si esto sucede, basta con entrar en el sistema de visión y enseñarle al sistema cuáles serán los objetos definidos como 1 y 2.

Estos objetos siempre estarán en la mesa intermedia, debido a que ya presentan algún maquinado. Los objetos 4 y 5 definidos corresponden a los objetos que únicamente pueden ser maquinados en torno o fresadora pero no en ambas máquinas. La entrada de estos objetos es considerada como la entrada

de un lote de piezas nuevas y por lo tanto, es necesario cambiar los programas del sistema una vez que todas las piezas que se encuentren en la celda hayan sido terminadas.

Se utilizan nueve contadores definidos de la siguiente manera:

Contador 1. Definido para determinar si el torno tiene alguna pieza, su valor es 0 si no existe pieza y es 1 si tiene pieza.

Contador 2. Definido para determinar si la pieza que tiene el torno ya fué maquinada en la fresadora. Vale 0 si la pieza no ha sido maquinada en la fresadora y tiene el valor de 1 si la pieza ya pasó por la fresadora.

Contador 3. Definido para determinar si la fresadora tiene alguna pieza, su valor es 0 si no existe pieza y es 1 si tiene pieza.

Contador 4. Definido para determinar si la pieza que tiene la fresadora ya fué maquinada en el torno. Vale 0 si la pieza no ha sido maquinada en el torno y tiene el valor de 1 si la pieza ya pasó por el torno.

Contador 5. Definido para indicar que existe una pieza en la posición intermedia. Las piezas pueden ser llevadas a esta posición cuando las posiciones de la mesa intermedia se encuentren ocupadas.

Contadores 6, 7, 8 y 9. Estos contadores se usan para determinar que existe alguna pieza en las posiciones de la mesa intermedia. El contador 6 está asociado a la primera posición de la mesa, el 7 a la segunda posición y así sucesivamente. Toman el valor de 0 si la posición no está ocupada y toman el valor de 1 si la posición está ocupada. Estos contadores son utilizados debido a que el sistema de visión no presenta la opción de tomar la imagen de toda la mesa intermedia y efectuar el análisis sobre todas las posiciones, por lo que la segmentación es realizada en cada una de las posiciones.

La notación utilizada en el diagrama de flujo es la siguiente:

BUSCAR OBJETO... Se refiere a la operación de buscar los objetos de la escena mediante el sistema de visión, digitalizando la imagen de la escena. Se aplican los diferentes filtros seleccionados y se comparan los parámetros de los objetos detectados con los parámetros grabados en el sistema, para efectuar la identificación de los objetos.

EXISTE EL OBJETO... Pregunta realizada para verificar la existencia de los objetos de interés.

¿C. 6 > 0?... Esta pregunta se realiza para determinar si el valor del contador (C.) 6 es mayor que cero. De la misma manera, se realizan preguntas para todos

los contadores que se utilizan en el programa. En este tipo de preguntas lo único que cambia es el número del contador.

C. $X = 0$ o $C. X = 1$... Asignación de valor a un contador. El programa utiliza los contadores 6, 7, 8, y 9 para determinar la existencia de alguna pieza en las posiciones de la mesa intermedia. El contador 6 está asociado a la posición uno de la mesa, el 7 a la posición dos y así sucesivamente. Un valor de 1 de los contadores nos indica que existe pieza en la posición mientras que un valor de 0 nos indica que no hay pieza en esa posición. Por ejemplo, la subrutina δ mostrada en la figura 6.2 (d) tiene este tipo de asignación. En ella, se le da el valor de cero al contador asociado a la posición de la cuál se haya retirado la pieza. En la subrutina ε mostrada en la figura 6.2 (e) se les asigna el valor de 1 al contador asociado a la posición donde se coloque una pieza.

APAGAR SEÑAL DE TORNO o FRESADORA... Esta es una instrucción que nos permite desconectar la señal de atención que solicitan el torno y la fresadora al robot, naturalmente, se debe desconectar una vez que se le haya atendido a la máquina que solicitó el servicio.

ENCENDER SEÑAL... Esta instrucción es ejecutada para indicar al operador que ya no existen objetos para la operación o bien, para indicar que ha entrado un lote de piezas diferentes y por lo tanto, es necesario cambiar los programas que se tienen para torno, fresadora y robot por los programas correspondientes que se utilizarán en las nuevas piezas.

δ , ε , θ y ρ ... Subrutinas para dar o retirar piezas de las máquinas. Son mostradas en las figuras 6.2 (d), 6.2 (e), 6.2 (f) y 6.2 (g).

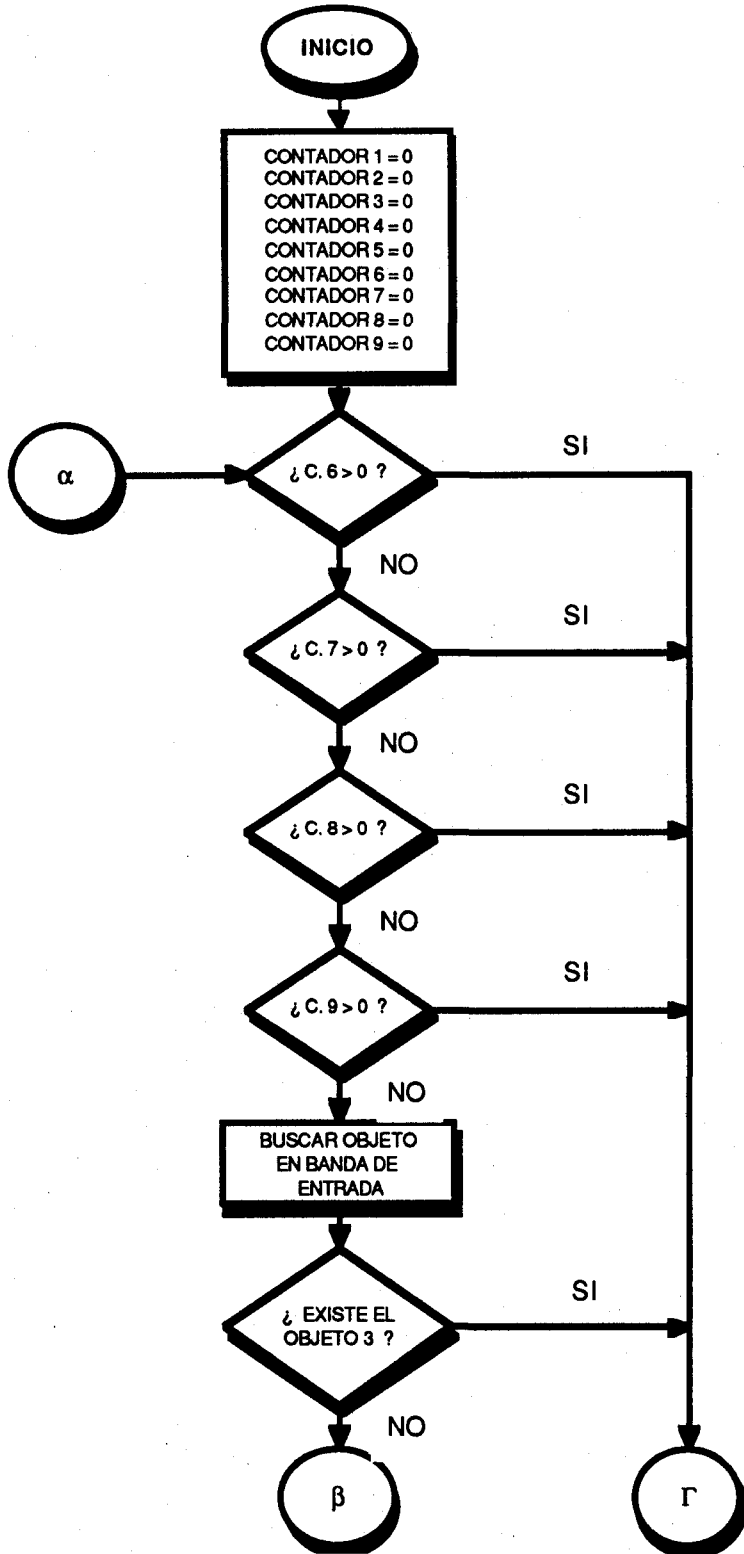


Figura 6.2 (a)

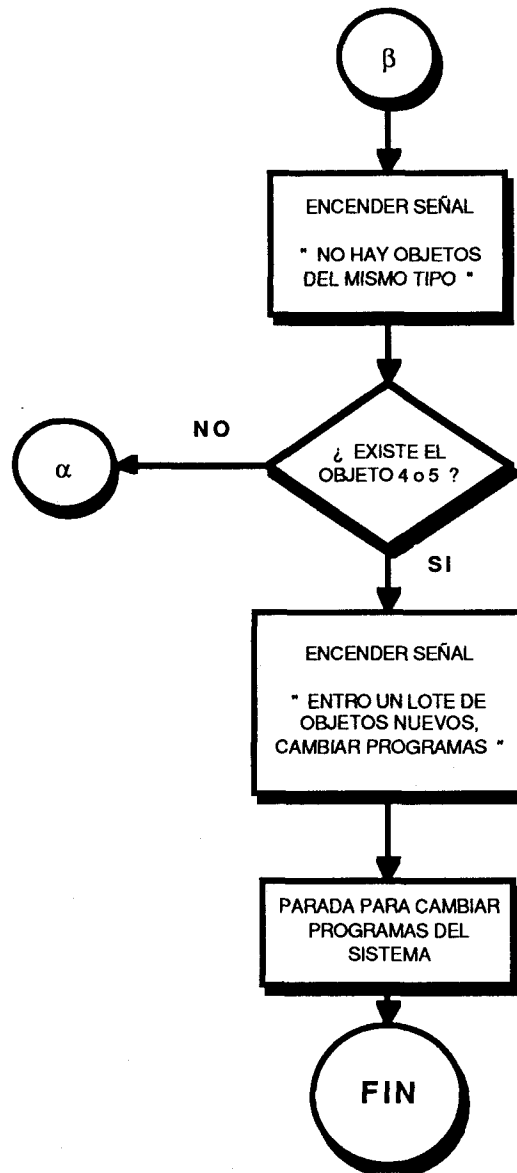


Figura 6.2 (b)

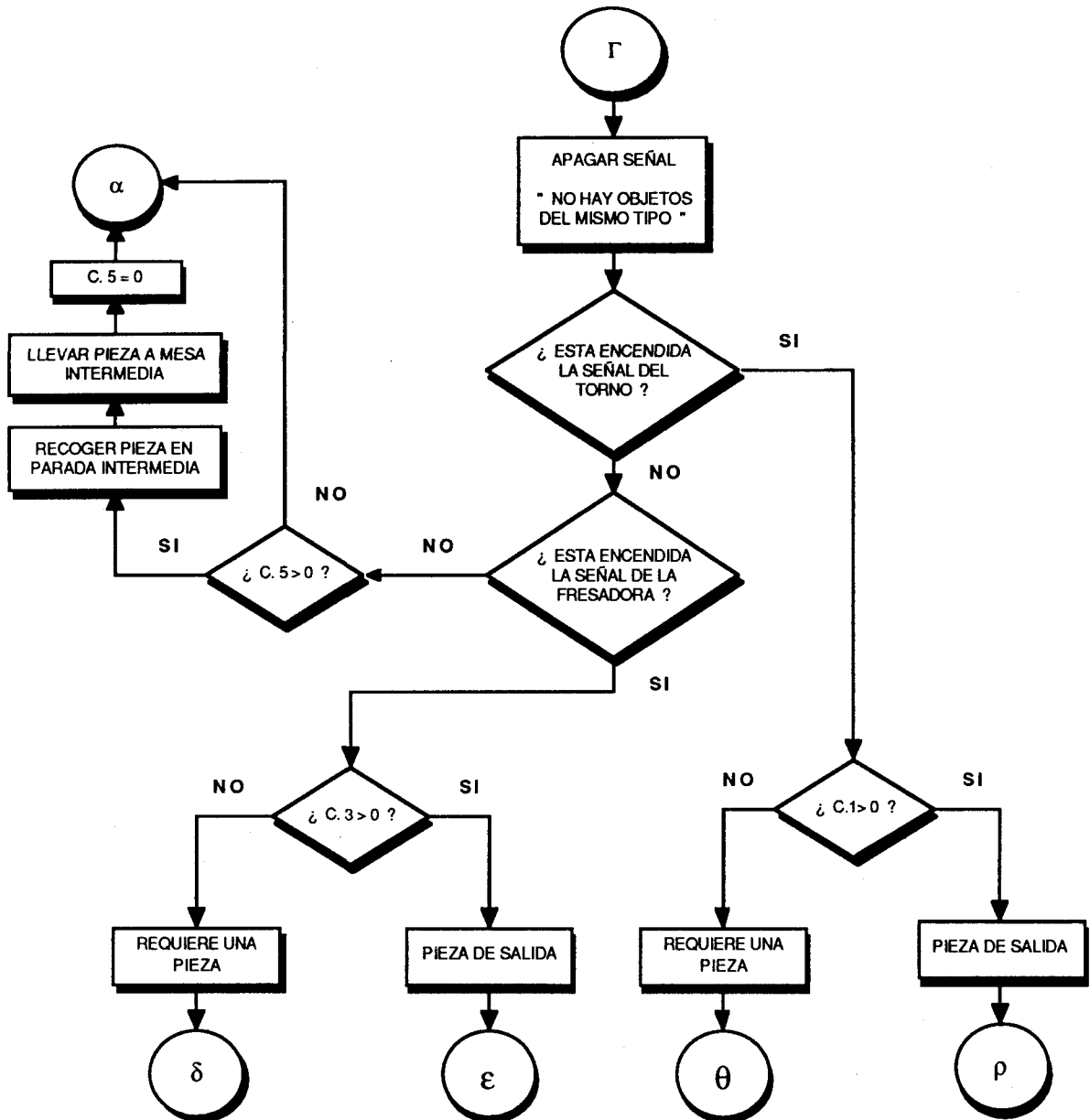


Figura 6.2 (c)

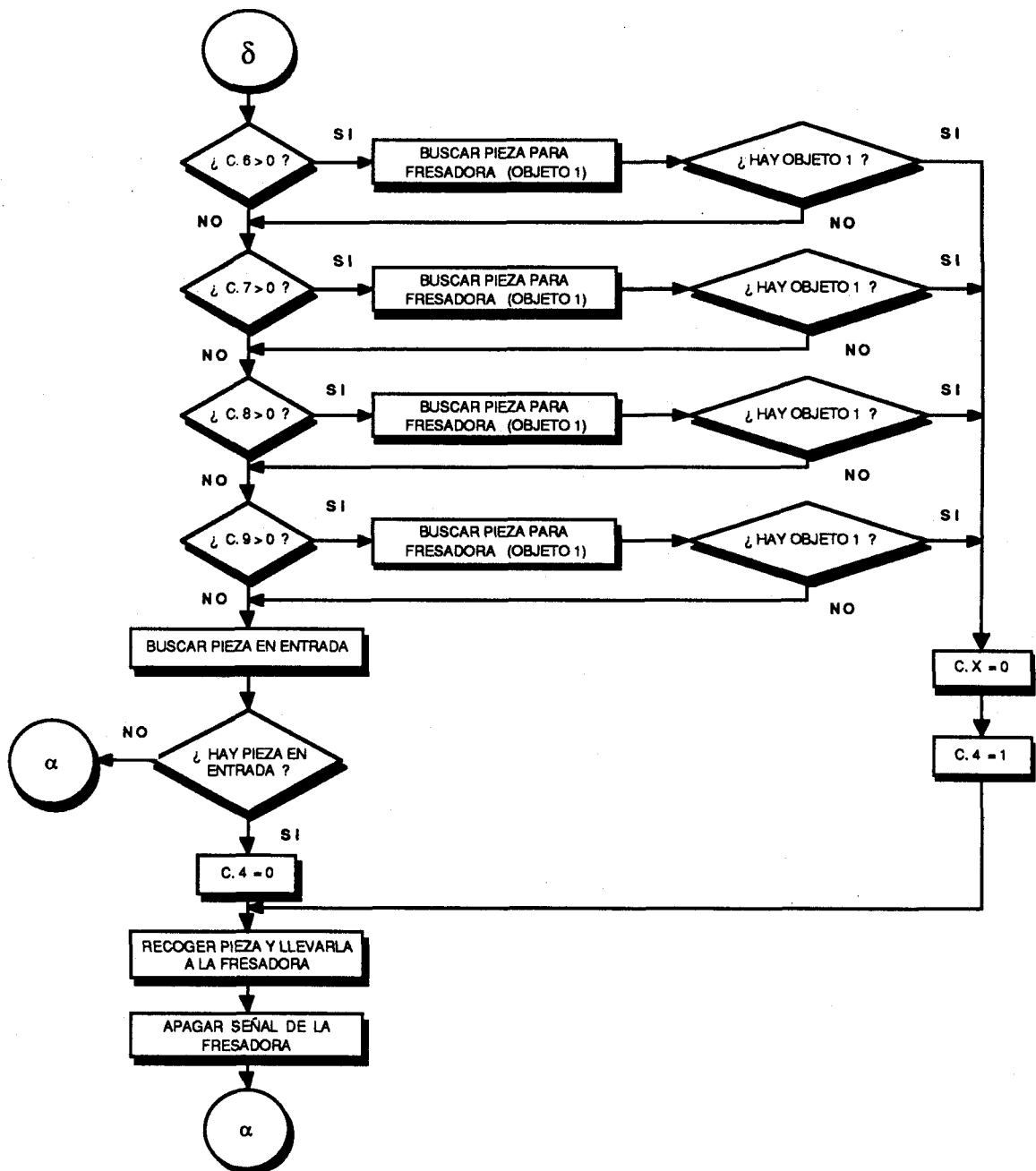


Figura 6.2 (d)

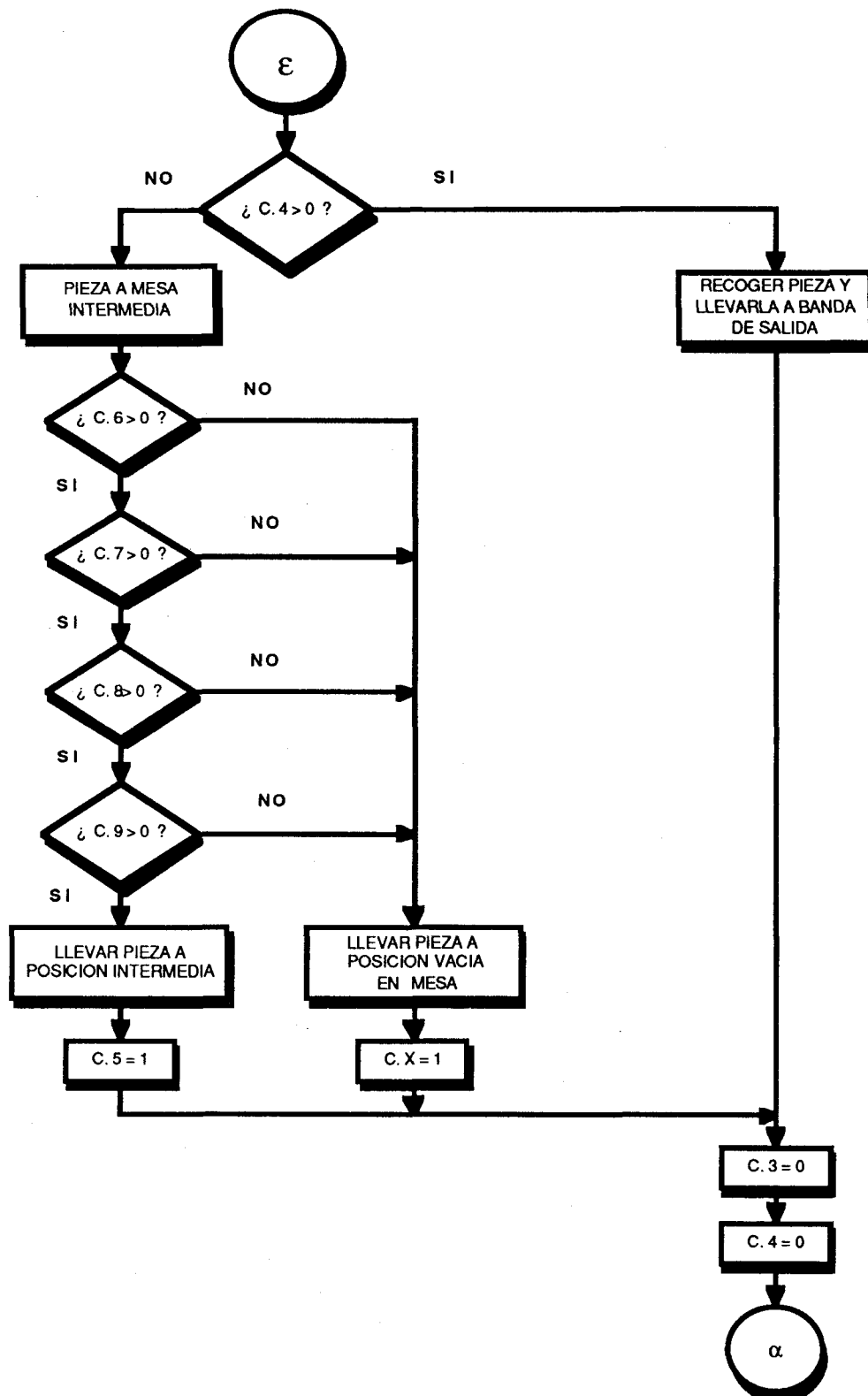


Figura 6.2 (e)

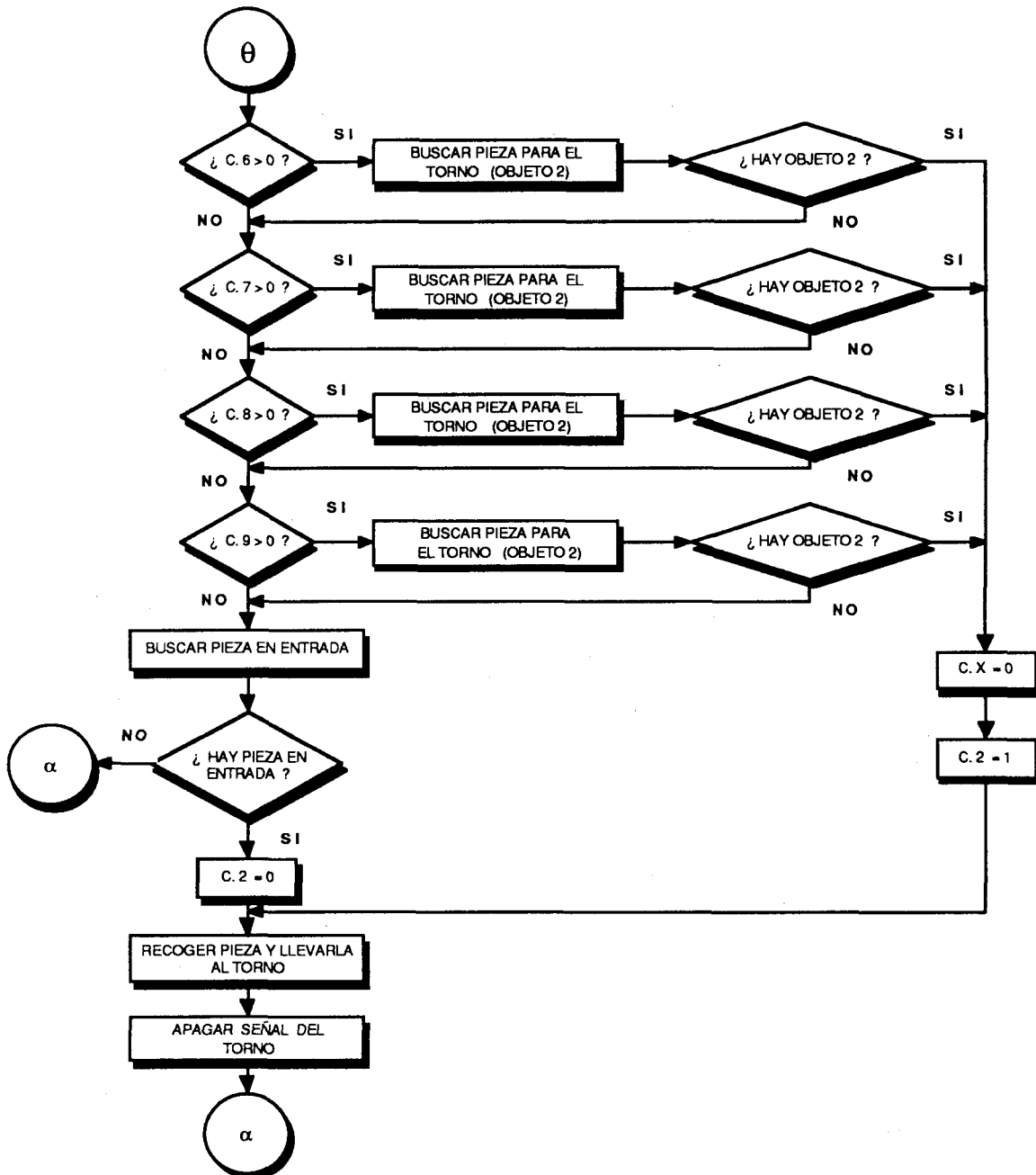


Figura 6.2 (f)

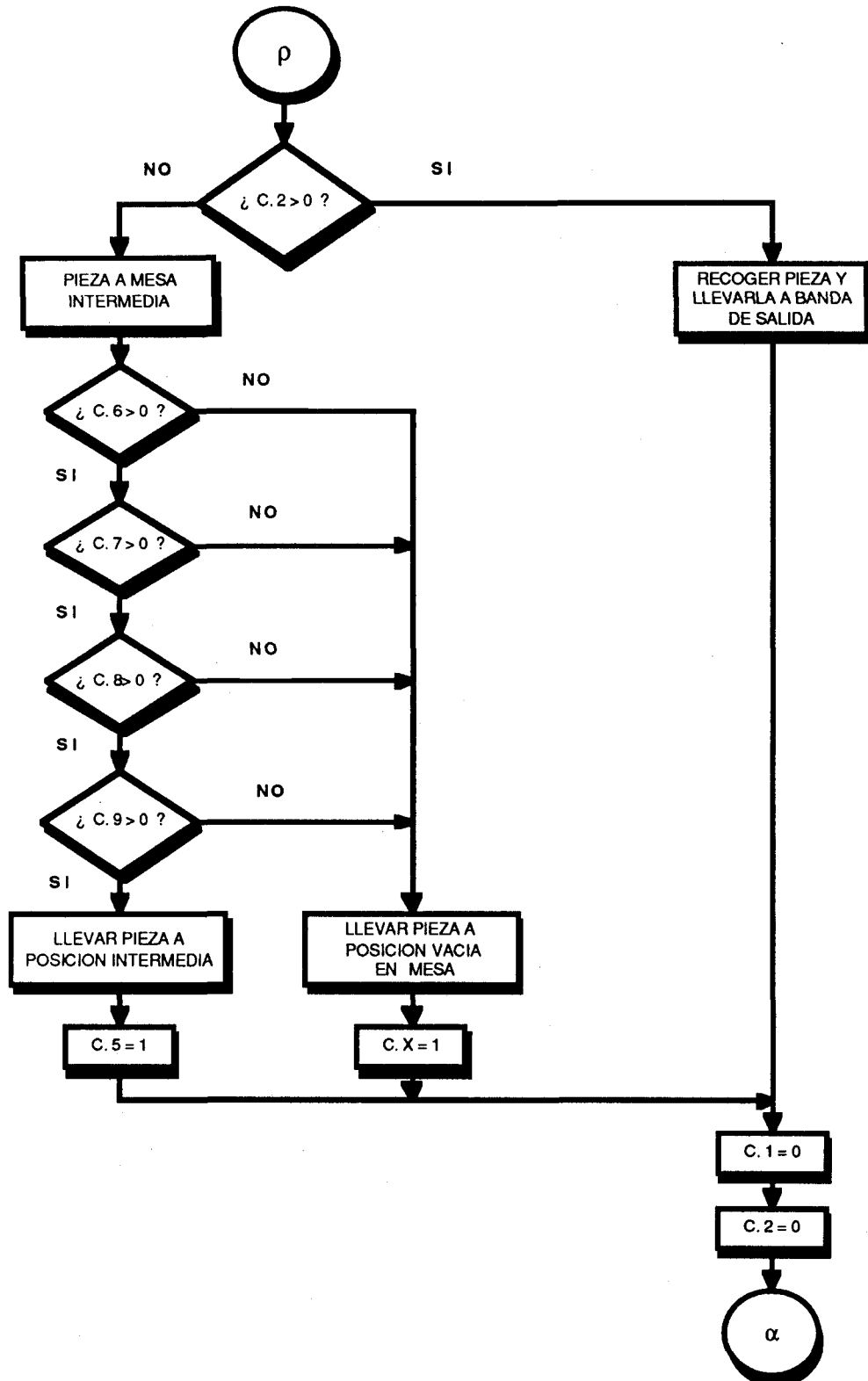


Figura 6.2 (g)

6.2.2 PROGRAMA PARA PIEZAS MAQUINABLES EN TORNO O FRESADORA (PIEZAS No. 3 y 4).

Para la operación de este programa solamente se utilizan piezas que una vez que han sido maquinadas van a la banda de salida para su descarga por ello, este programa no es tan complejo como el anterior ya que solamente utiliza dos contadores definidos para determinar si cada una de las máquinas tiene alguna pieza. El contador 1 está asociado al torno mientras que el contador 2 corresponde a la fresadora. Cuando los contadores tienen un valor de 1, significa que tienen pieza trabajando y un valor de 0 indica que no tienen pieza. Así, cuando las máquinas solicitan atención al manipulador únicamente se verifica si el valor de su respectivo contador es 0 o mayor que 0 para determinar si la llamada solicitada es para llevarle o retirarle la pieza. Si las máquinas requieren una pieza, mediante el sistema de visión se identifica la pieza que se encuentre en la banda de entrada determinando de esta manera si la pieza debe ir al torno o fresadora. Una vez que la pieza es identificada, el sistema de visión transfiere las coordenadas de la posición de la pieza al controlador del manipulador para que el manipulador recoja la pieza y la lleve a la máquina correspondiente.

Este diagrama de flujo se muestra en las figuras 6.3 (a) a 6.3 (c). Fué realizado para la operación del sistema donde únicamente se usarán piezas maquinables en torno o fresadora. El sistema de visión ayudará a reconocer estas piezas y determinará hacia que máquina debe ir cada pieza. En este caso, no se hace uso de la mesa intermedia de almacenamiento, porque cada pieza que haya sido maquinada va directamente a la banda de salida. Se definieron los siguientes objetos para la operación.

- Objeto número 3 (ID. NUMBER 3). *Pieza No. 2 de la figura 6.1. La entrada de estas piezas al sistema indican que las piezas 3 y 4 se han terminado por lo que será necesario cambiar los programas del sistema.*
- Objeto número 4 (ID. NUMBER 4). *Pieza que exclusivamente puede ser trabajada en el torno. Corresponde a la pieza No. 3 que se muestra en la figura 6.1.*
- Objeto número 5 (ID. NUMBER 5). *Pieza que únicamente puede ser trabajada en la fresadora y corresponde a la pieza No. 4 de la figura 6.1.*

En este diagrama, únicamente se utilizaron dos contadores definidos de la siguiente manera:

Contador 1. Definido para determinar si el torno tiene alguna pieza de trabajo. Vale 0 si no hay pieza y vale 1 si el torno tiene pieza.

Contador 2. Definido para determinar si la fresadora tiene alguna pieza de trabajo. Vale 0 si no hay pieza y vale 1 si la fresadora tiene pieza.

La notación en este diagrama de flujo es la misma que para el diagrama anterior.

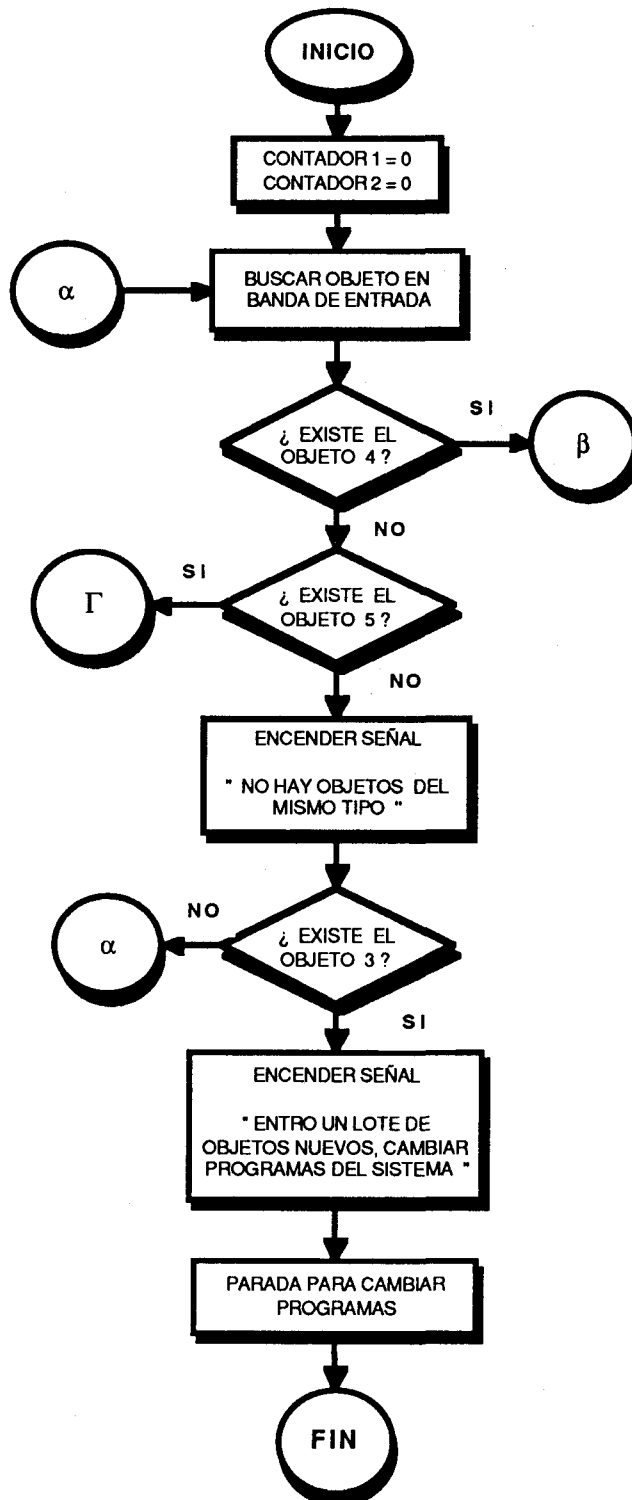


Figura 6.3 (a)

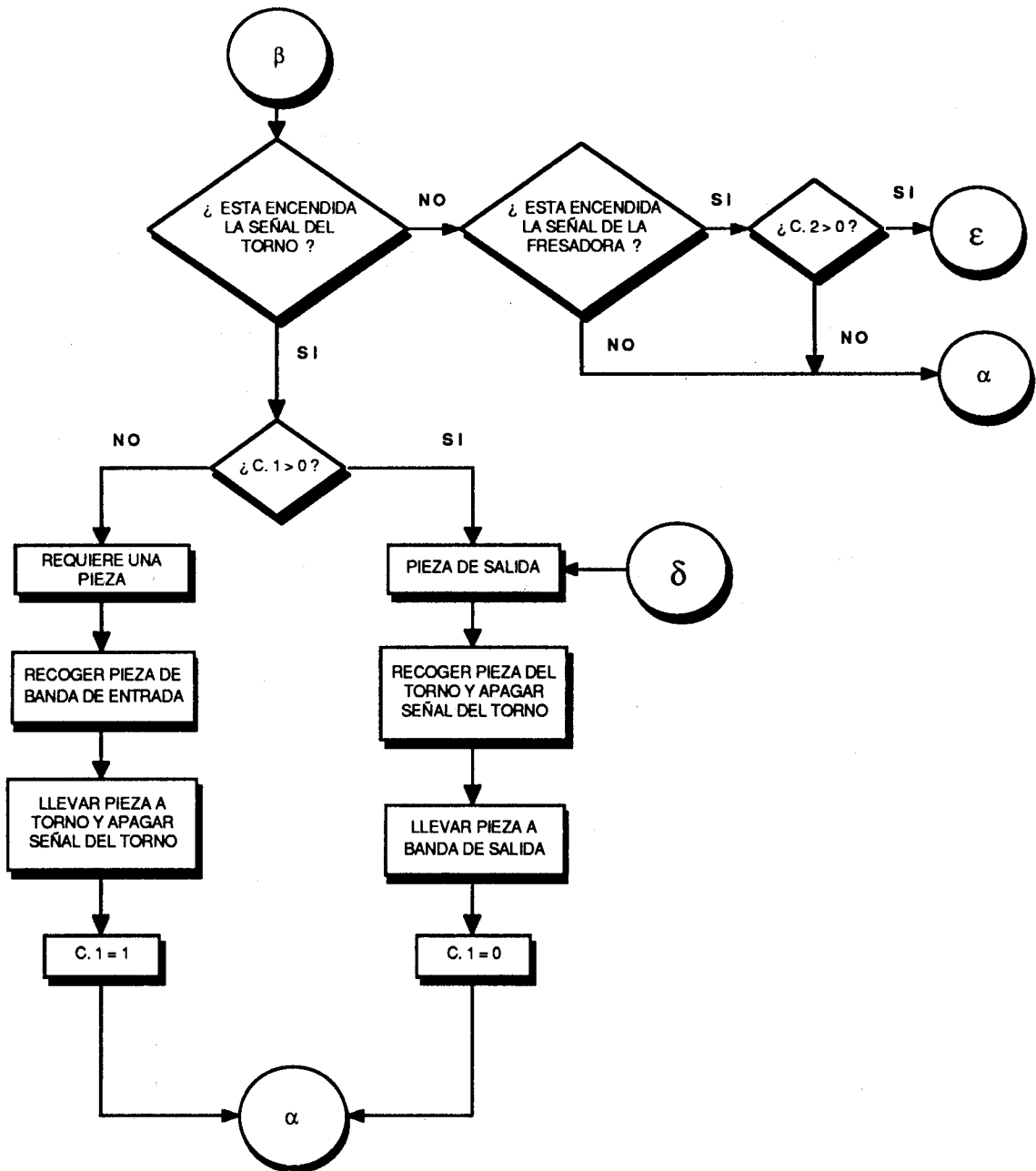


Figura 6.3 (b)

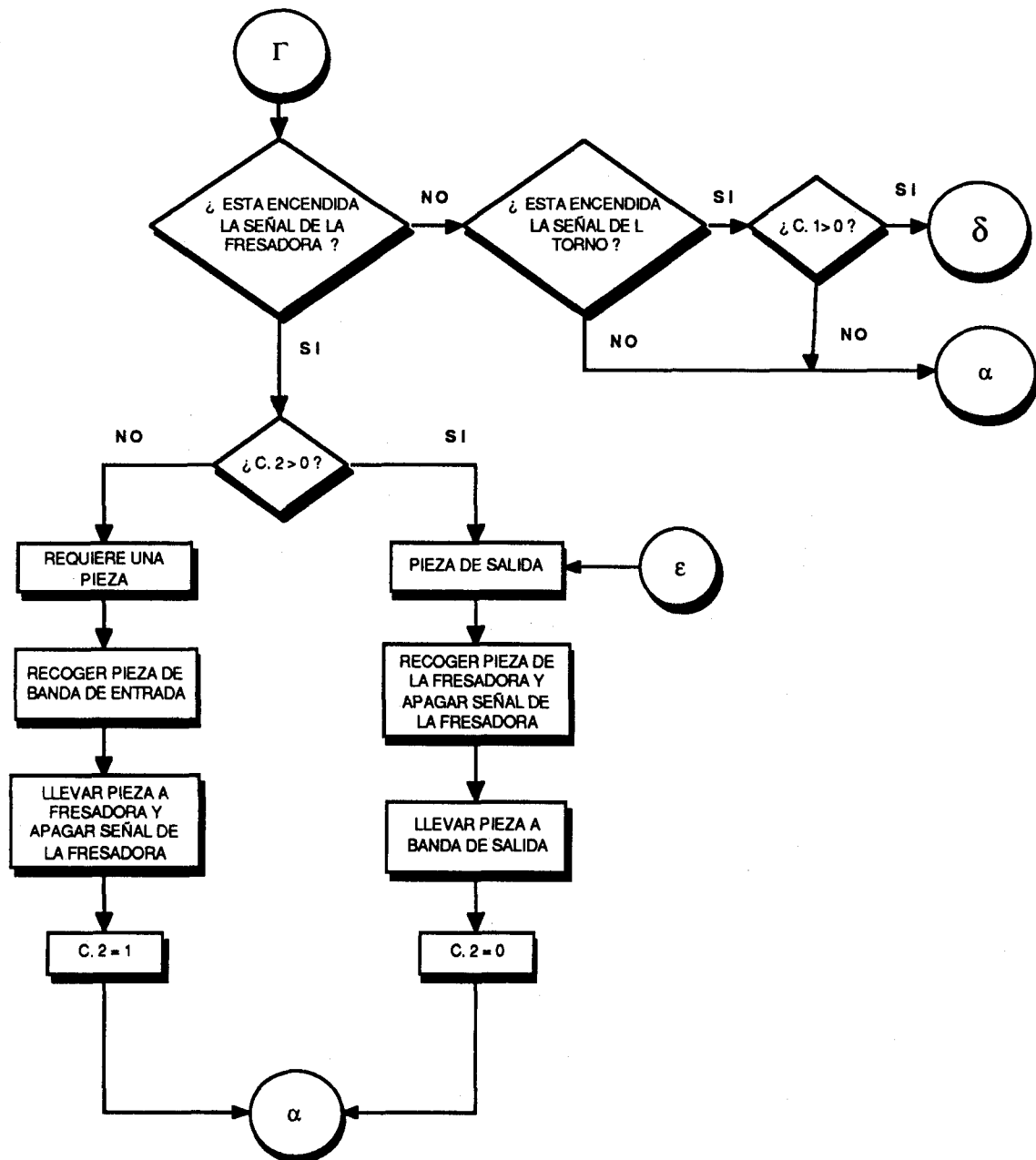


Figura 6.3 (c)

CAPITULO 7

TECNICAS PARA LA DETECCION DE OBJETOS POR MEDIO DE SUS BORDES

7.1 SEGMENTACION Y DESCRIPCION

Todos los sistemas de visión tienen como objetivo primordial derivar una descripción de una escena, analizando una o más imágenes de dicha escena. Para que el sistema reconozca piezas, partes, perforaciones y en general objetos de una superficie, primero debe distinguir los objetos de interés del resto de la superficie destacando partes de la imagen que corresponden a dichos objetos [4].

El proceso de extraer subconjuntos de una imagen, dividiéndola en las partes y objetos que la constituyen, es llamado **SEGMENTACION**. El proceso de segmentación es uno de los pasos más importantes en la operación de un sistema de visión, ya que es aquí donde los objetos son extraídos de la escena para un análisis futuro donde se puede determinar si existen objetos que se superponen o se tocan e incluso, dicha escena segmentada puede servir como base para una posterior segmentación destacando de esta forma los detalles de las piezas de la escena.

El reconocimiento de los objetos por medio del análisis para la extracción de las características constituye la etapa de **DESCRIPCION**. Esta etapa de descripción puede presentar mucha dificultad, dependiendo de la complejidad de los objetos. Si los objetos de la escena difieren mucho entre sí, se puede realizar una comparación sencilla contra patrones o plantillas. Si los objetos a reconocer son complejos, puede ser necesario descomponer el proceso de reconocimiento para: *Detectar primero subobjetos y reconocer sus propiedades y luego, reconocer los objetos como combinaciones de esos subobjetos en base a relaciones específicas previamente definidas.* Este procedimiento es conocido como **COMPARACION DE ESTRUCTURAS**.

Para el sistema de visión utilizado, las características o descriptores que definen a los objetos están dados en base a las siguientes características:

AREA DE LA SUPERFICIE TOTAL: Es el área de la región contenida en el contorno del objeto, incluyendo el área de los agujeros. Los agujeros de un objeto son considerados como una región del mismo objeto donde la intensidad del objeto es considerablemente pequeña comparándola con la intensidad del

resto del objeto. Su valor está dado en pixeles. Este valor es de gran utilidad por el hecho de que la geometría de los objetos no cambia para un proceso de maquinado en particular.

AREA DE LA SUPERFICIE: Es el área contenida en el contorno del objeto sin incluir el área de los agujeros.

NUMERO DE AGUJEROS: Corresponde al número de agujeros que fueron detectados dentro del contorno del objeto.

LONGITUD DEL PERIMETRO: Es la longitud exterior o longitud del contorno de los objetos. En este sistema, este valor de longitud está dado en pixeles pero, en muchos otros sistemas de visión se puede determinar la longitud real.

A partir de los descriptores anteriores, existen otros que están dados como función de ellos, tales descriptores corresponden a razones como PERIMETRO/AREA DEL OBJETO, AREA DE AGUJEROS/AREA DEL OBJETO, etc.

Es importante reconocer que el proceso de descripción ayuda a identificar y/o reconocer los objetos de la escena segmentada por lo que a continuación se explicarán las técnicas más usuales para dicho reconocimiento de los objetos.

7.2 METODO DE REGIONES [5]

7.2.1 METODO DEL UMBRAL

Cuando en la escena de operación existen objetos cuya brillantez difiere significativamente de la mesa de trabajo, la digitalización de dicha imagen da como resultado un conjunto de pixeles con niveles de gris diferentes a los niveles de gris de los vecinos circundantes. Generalmente, esta operación es controlada por medio de la forma en que incide la luz sobre los objetos de trabajo, logrando de esta manera una gran diferencia en la brillantez de los objetos.

Para los marcos definidos de una imagen, los subconjuntos de la imagen pueden ser extraídos estableciendo un UMBRAL para los niveles de gris. Cada uno de los marcos de la imagen representa una porción de la misma, los cuales son definidos por medio de coordenadas. Se puede definir el ancho y largo para cada uno de ellos así como el punto o coordenada de inicio. De esta manera, cada pixel será clasificado como "claro" u "oscuro" dependiendo de si su nivel de gris es inferior o superior al umbral definido.

Si en el sistema de visión utilizado la iluminación está bajo control, es posible establecer un valor de umbral permanente para segmentar las escenas de determinada clase, pero, en general, será necesario determinar el valor de umbral para cada región individual. Tal es el caso que se presentó para las escenas de la mesa de trabajo y de la banda de entrada para la celda flexible de manufactura PREP- INC.

Cuando los objetos ocupan gran parte de la escena, se pueden determinar analizando el HISTOGRAMA DE LA IMAGEN, el cual es una gráfica que muestra la frecuencia de ocurrencia de cada nivel de gris de la imagen.

El histograma debe de presentar dos picos, uno que representa el nivel de gris del fondo o mesa de trabajo y el otro pico representa los niveles de gris del objeto. Los niveles intermedios entre los picos (Valles) son relativamente infrecuentes. En estos casos, un buen nivel en el cual establecer el valor de umbral es el más infrecuente entre los picos, correspondiendo al valle, ya que casi todos los pixeles del objeto caen a un lado de dicho umbral y los del fondo caen del otro lado.

Si la iluminación de la escena no es uniforme, de tal manera que los objetos no puedan ser separados del fondo o mesa de trabajo por comparación con el nivel de umbral, esta situación se puede atacar dividiendo la imagen digitalizada en marcos y escogiendo un valor de umbral adecuado para cada marco para analizar en cada una de ellos su propio histograma.

7.2.1.1 PROCESO INTERACTIVO POR INTENSIDAD DE PÍXELES

La señal f' de la imagen asociada a un objeto puede ser definida por la siguiente transformación de intensidad:

$$f' = \begin{cases} f & f_l \leq f \leq f_u \\ 0 & \text{Resto} \end{cases} \quad (7.1)$$

La gráfica de la ecuación anterior está representada en la figura 7.1

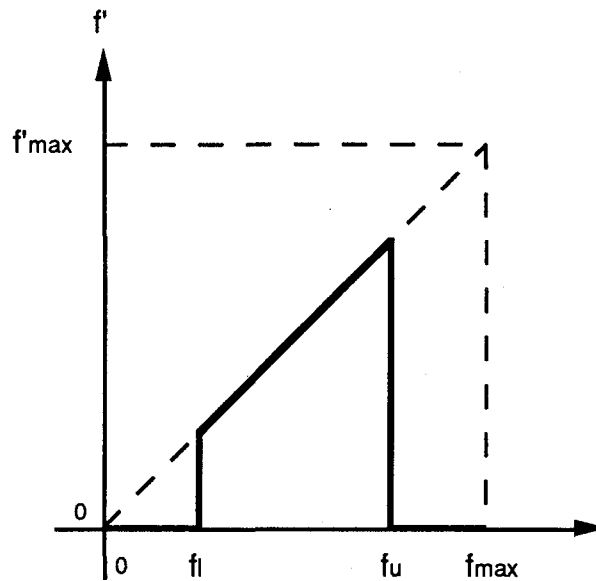


Fig. 7.1 Escala de intensidad de la señal de una imagen.

Los valores de los parámetros f_i y f_u inicialmente son considerados como el valor mínimo y el valor máximo posible de los valores de intensidad de la imagen, los cuales son cero y f_{max} respectivamente. Como la salida f' de la imagen puede ser vista en el monitor de imágenes digitalizadas, el valor de f_i puede ser incrementado y/o el valor de f_u ser decrementado gradualmente hasta que el valor de la señal f' mantiene su valor de intensidad original, es decir, hasta que no existan distorsiones en la imagen. Los valores resultantes de f_i y f_u corresponden a las fronteras inferior y superior del rango de intensidad de los pixeles del objeto de la escena. El rango de intensidad del objeto de la figura 7.2 ha sido determinado por los valores de $f_i=32$, $f_u=63$ y $f_{max}=255$.

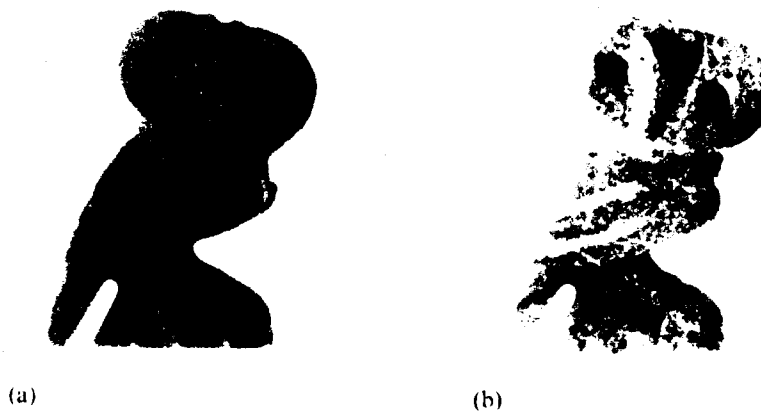


Fig. 7.2

Es evidente de la figura 7.2 que el objeto presenta un gran contraste cuando se mueve el rango de intensidad. Una función de intensidad acotada, con un mínimo número de parámetros puede ser considerada sin perder de vista los efectos de los parámetros individuales sobre la manipulación de los datos de la imagen. Un ejemplo de esto es la siguiente transformación:

$$f' = \begin{cases} \epsilon f & 0 \leq f \leq f_l \\ \lambda \epsilon (f - f_l) + f_l' & f_l \leq f \leq f_u \\ \epsilon (f - f_u) + f_u' & f_u \leq f \leq f_{\max} \end{cases} \quad (7.2)$$

donde:

$$\epsilon = \frac{f'_{\max}}{f_{\max} + (f_u - f_l)(\lambda - 1)} \quad f_l' = \epsilon f_l \quad f_u' = \epsilon \{ \lambda (f_u - f_l) + f_l \}$$

En la cual solo un parámetro es agregado, λ , para la medición del rango de intensidad con límites de f_l y f_u del objeto. Este parámetro determina la pendiente de la parte lineal de la función acotada dentro y fuera del intervalo de intensidad $\{f_l \leq f \leq f_u\}$.

La figura 7.3 muestra la gráfica de la función de intensidad acotada de la ecuación (7.2) para $\lambda=2$ y $\lambda=\infty$

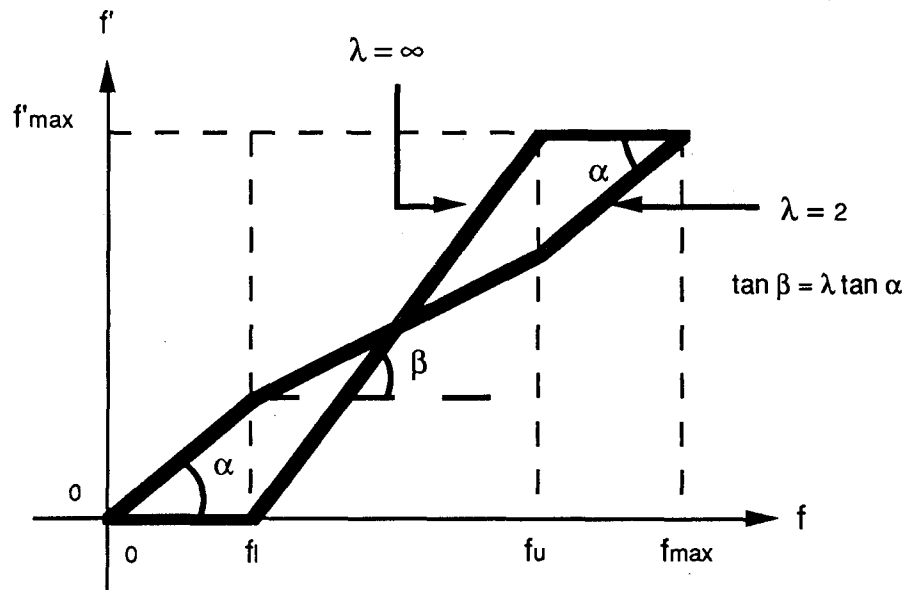


Fig. 7.3 Gráfica de la función acotada.

Como $\lambda \rightarrow \infty$ el rango de intensidad $\{ f_l \leq f \leq f_u \}$ es mapeado en el máximo rango de intensidad de salida $\{ 0 \leq f' \leq f_{\max} \}$. Este parámetro fué usado en la figura 7.2 en la cual se demuestra que la imagen resulta considerablemente mejor.

Como solamente un número finito Q de valores discretos de intensidad pueden ser representados en una imagen digitalizada, los valores de la ecuación (7.2) tienen que ser vistos como una cantidad. Por razones de eficiencia, primero debemos calcular una función de mapeo de intensidad $T(i)$ para $i=0, 1, 2, \dots, Q-1$ y almacenarla como un vector Q -dimensional. Posteriormente, el acotamiento es llevado a cabo indexando este vector y usando los valores de intensidad de los pixeles de la imagen de entrada como índices. El efecto del mapeo de la intensidad puede ser demostrado con la ayuda del HISTOGRAMA DE INTENSIDAD. Dando los valores de intensidad de la imagen en pixeles, f , pueden ser representados por q bits, donde la longitud es $Q=2^q$. El vector Q -dimensional $h(i)$ donde $i=0, 1, 2, \dots, Q-1$ cuyos elementos pueden ser calculados por la siguiente ecuación:

$$h(i) = \sum_{M=0}^{M-1} \sum_{N=0}^{N-1} \delta\{f(m, n) - i\} \quad (7.3)$$

donde:

$$\delta(\epsilon) = \begin{cases} 1 & \epsilon = 0 \\ 0 & \text{Resto} \end{cases}$$

Es conocido como el HISTOGRAMA DE INTENSIDAD. Observe como para un valor definido de intensidad i , la ecuación para el histograma define un valor que se incrementa cuando un pixel de la imagen tiene el mismo valor de intensidad que el valor de intensidad i definido por lo que se puede decir que esta ecuación acumula el número total de pixeles de la imagen que tienen el mismo valor de intensidad.

El i -ésimo elemento de este vector representa el número de pixeles en la imagen con un valor de intensidad i . Dividiendo los elementos del vector h por MN , que es el total de pixeles, obtenemos la FRECUENCIA RELATIVA de ocurrencia de los valores de intensidad de la imagen. Por ejemplo, la figura 7.4 muestra los histogramas correspondientes a cada una de las imágenes de la figura 7.2. El segundo histograma ha sido acotado en la intensidad por lo que los pixeles de la imagen del fondo son concentrados en un valor de intensidad muy grande, mientras que los valores de gris que representan al objeto son distribuidos sobre todo el rango posible de valores.

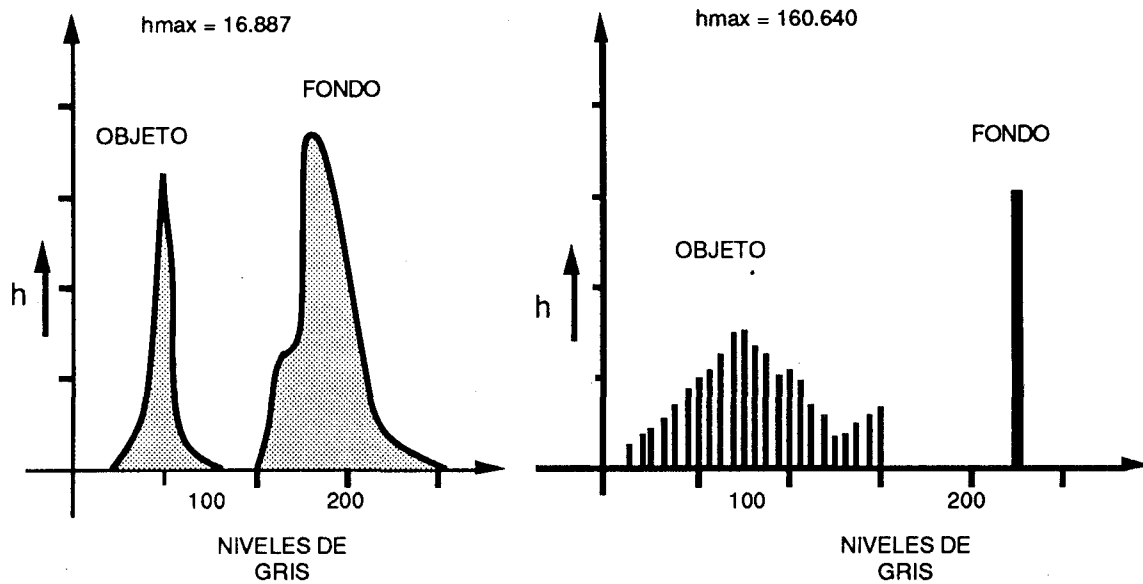


Fig. 7.4 Histogramas de los objetos de la figura 7.2

La relación existente entre la señal de entrada $f(m,n)$ y la señal de salida $g(m,n)$ para un sistema de visión cuya operación de descripción está basada en el método del umbral, está definida por la ecuación

$$g(m,n) = \begin{cases} l_1 & 0 \leq f(m,n) \leq S \\ l_2 & S \leq f(m,n) \leq f_{\max} \end{cases} \quad (7.4)$$

donde l_1 e l_2 son dos valores arbitrarios con $l_1 \neq l_2$ y normalmente se consideran como $l_1=0$ e $l_2=1$, S es la intensidad de umbral que es usada. Para una adecuada selección del valor de S , los pixeles con un valor de l_1 en la imagen de salida representan los objetos y aquellos de valor de l_2 representan el fondo o viceversa. En tal caso, el valor de umbral S debe ser determinado a partir del histograma correspondiente y dicho valor debe estar entre los picos que representan al objeto y al fondo respectivamente en el histograma.

Para imágenes que contienen P objetos con diferente intensidad en su región que los caracteriza, la salida está definida como:

$$g(m, n) = l_i \quad \text{para } S_{i-1} \leq f(m, n) < S_i \quad (7.5)$$

donde $i=1, 2, \dots, P$, $S_0=0$ y $S_P= f_{\max} + 1$. En este caso, la confiabilidad de la operación de umbral depende significativamente del valor más adecuado de los valores de umbral que aquí se tengan. Un valor de umbral erróneo llevará a una segmentación y descripción de la imagen totalmente errónea. Como en el caso

anterior, los valores apropiados del umbral para la segmentación pueden ser determinados del análisis del histograma de intensidad global. Estos valores de umbral estarán dados en los P-1 valles que se encuentran entre los P picos en el histograma P-modal. Si los valles no están bien definidos, los valores apropiados del umbral pueden ser evaluados mediante un histograma de pesos. En lugar de obtener el histograma de acuerdo a la ecuación 7.3, la contribución de cada pixel, $f(m,n)$, en forma individual acumulada por $g\{e(m,n)\}$, de acuerdo a las características, $e(m,n)$, de la imagen en la coordenada (m,n) estará dada por:

$$h(i) = \sum_{M=0}^{M-1} \sum_{N=0}^{N-1} g\{e(m,n)\} \delta\{f(m,n) - i\} \quad (7.6)$$

donde:

$$i=0,1,\dots, f_{\max}.$$

Por ejemplo, si la diferencia de intensidad absoluta entre el pixel $f(m,n)$ con cada uno de los cuatro vecinos adyacentes es seleccionada por $e(m,n)$ como:

$$e(m,n) = \left| f(m,n) - 1/4 \{ f(m,n-1) + f(m,n+1) + f(m-1,n) + f(m+1,n) \} \right| \quad (7.7)$$

y con respecto a $e(m,n)$ la función monóticamente decreciente

$$g(e(m,n)) = \{1 + e(m,n)\}^{-1} \quad (7.8)$$

Observe como a partir del valor definido por $e(m,n)$ en la ecuación 7.7, $g(e(m,n))$ siempre tomará un valor menor o igual a 1 ya que $e(m,n)$ podrá ser 0 o un número mayor que 1. El valor de $e(m,n)$ será 0 cuando los pixeles vecinos a (m,n) tengan el mismo valor de intensidad que dicho pixel y, será muy grande cuando los pixeles vecinos tengan un valor de intensidad diferente al valor de intensidad del pixel (m,n) . Tales casos se pueden presentar cuando el pixel (m,n) se encuentre dentro del objeto y los pixeles vecinos también se encuentren dentro. Si el pixel (m,n) se encuentra dentro del objeto y los vecinos se encuentran fuera, tal como en un borde, el valor de $e(m,n)$ se incrementará debido a la diferencia entre las intensidades. Esto trae como consecuencia que el valor de $g(e(m,n))$ permanezca como 1 cuando los vecinos y el pixel (m,n) tengan la misma intensidad pero, el valor de $g(e(m,n))$ disminuirá cuando los pixeles vecinos estén fuera del objeto. Así, $g(e(m,n))$ es una función de pesos la cual disminuirá la frecuencia en el histograma cuando los pixeles se encuentren cercanos a los bordes del objeto, esto se representará en el histograma como un valle más profundo.

La figura 7.5 muestra dos histogramas de una escena donde existen dos imágenes, por ello, se presentan dos valores de umbral S_1 y S_2 . El primer

histograma fué determinado mediante la ecuación (7.3) mientras que el segundo, fué obtenido mediante la ecuación (7.6). Note que en el segundo histograma se presenta claramente cuales deben ser los valores de umbral S_1 y S_2 más adecuados para la segmentación de la imagen.

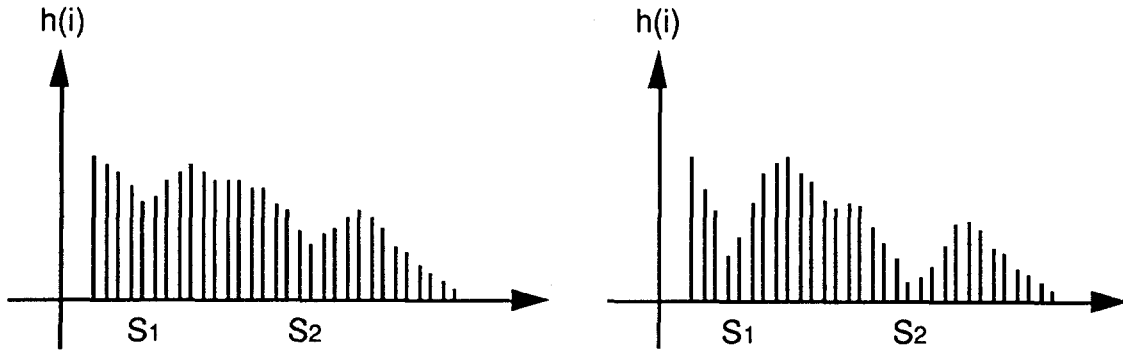


Fig. 7.5 Histogramas obtenidos con las ecuaciones (7.3) y (7.6) para una imagen

7.2.1.2 OPERACION OPTIMA DEL UMBRAL [6]

Los valores de la intensidad dentro de la región que define a un objeto, no siempre son constantes debido a fluctuaciones aleatorias de la señal. Estas variaciones pueden ser descritas como una aproximación a una función de densidad de probabilidades. La implicación de este argumento es de que si una región es descrita por un histograma multimodal, deberá ser unida de tal manera que la región resultante tenga un histograma unimodal.

En esta sección, una imagen consiste de $M \times N$ pixeles la cual contiene un objeto claro O sobre un fondo oscuro H . la probabilidad de que la intensidad i ocurra dentro de dos áreas de imágenes f_ϵ con $\epsilon=O$ o $\epsilon=H$ puede ser aproximada por la función de densidad de probabilidad

$$P_\epsilon(i) = \frac{1}{(2\pi)^{1/2} \sigma_\epsilon} \exp \left\{ -\frac{(i - \mu_\epsilon)^2}{2\sigma_\epsilon^2} \right\} \quad (7.9)$$

Con una media μ_ϵ y una desviación estándar σ_ϵ . Si asumimos que f_O consiste de P_O pixeles y f_H consiste de P_H pixeles y de que

$$\frac{P_O + P_H}{MN} = 1 \quad (7.10)$$

Tenemos que la distribución de intensidad en toda la imagen es

$$P(i) = \frac{P_O p_O(i) + P_H p_H(i)}{MN} \quad (7.11)$$

En base a lo que asumimos anteriormente, es razonable clasificar los pixeles con una intensidad menor que S como el fondo y la parte remanente como el objeto. La distribución de probabilidad de las dos áreas parcialmente superpuestas nos llevará a un error E_ϵ en la segmentación resultante, debido a que pixeles del área ϵ pueden ser erróneamente asignados al área η donde $\eta \neq \epsilon$ por lo que

$$E_\epsilon(s) = \int_A^B P_\epsilon(i) di \quad (7.12)$$

donde:

$A = -\infty$ y $B = S$ para $\epsilon = O$, y $A = S$ y $B = \infty$ para $\epsilon = H$

Entonces el error total E es

$$E(s) = \frac{P_O E_O(i) + P_H E_H(i)}{MN} \quad (7.13)$$

Sustituyendo las ecuaciones (7.9) en (7.12) y después en la ecuación (7.13), diferenciando con respecto a S y resolviendo la ecuación igual a cero, obtenemos la siguiente ecuación para S:

$$\left(\frac{1}{\sigma_H^2} - \frac{1}{\sigma_O^2} \right) S^2 - 2 \left(\frac{\mu_O}{\sigma_O^2} - \frac{\mu_H}{\sigma_H^2} \right) S + 2 \ln \left(\frac{\sigma_H p_O}{\sigma_O p_H} \right) + \frac{\mu_H^2}{\sigma_H^2} - \frac{\mu_O^2}{\sigma_O^2} = 0 \quad (7.14)$$

Como la ecuación de arriba es cuadrática para S, se obtendrán dos valores necesarios de umbral para la segmentación de la imagen en las regiones f_o y f_h . Para $\sigma_o = \sigma_h = \sigma$ la ecuación (7.14) define un valor de umbral como:

$$S = \frac{\mu_O + \mu_H}{2} - \frac{\sigma^2}{\mu_O - \mu_H} \ln \left(\frac{p_O}{p_H} \right) \quad (7.15)$$

Para $\sigma_0 = \sigma_H$ y $P_0 = P_H$, S es la media aritmética de μ_0 y μ_H . De las ecuaciones (7.14) y (7.15), los valores óptimos del umbral son definidos por los parámetros de la función de densidad de probabilística $p(i)$. En base a la ecuación (7.3), el mínimo error obtenido por medio de mínimos cuadrados es:

$$E_H = \sum_{i=0}^{f_{\max}} \left\{ P(i) - \frac{h(i)}{MN} \right\}^2 \quad (7.16)$$

7.2.2 CRECIMIENTO DE REGIONES

Las precondiciones necesarias para la aplicación del método del umbral descrito anteriormente, son:

- 1 El número de objetos contenidos en la imagen es conocido *a priori*
- 2 Los objetos son representados dentro de rangos de intensidad los cuales se traslapan unos a otros en una parte muy pequeña y además,
- 3 Un valor de umbral adecuado puede ser calculado

Un procedimiento alternativo que toma en consideración la conectividad de las imágenes durante la segmentación, y por lo tanto es posible dividir f en P subregiones orientadas, f_ϵ , es el método de CRECIMIENTO DE REGIONES.

De un pixel escogido inicialmente (idealmente un pixel por cada región f_ϵ), se inicia un proceso iterativo el cual asigna paso por paso a los pixeles vecinos propiedades similares a la subregión f_ϵ . El crecimiento de la región continúa hasta que todos los pixeles de la imagen son asignados a una de las subregiones f_ϵ .

Inicialmente, de cada objeto con un valor de nivel de gris debe de encontrarse un punto inicial sobre el cual el proceso de crecimiento de la región pueda ser llevado a cabo en ese nivel. Si más de un punto inicial es encontrado para una región, la región es dividida en varias regiones; este tipo de error puede ser corregido por la combinación de las regiones con propiedades similares. Si una región que define a un objeto no contiene un punto inicial, es combinada con una región vecina. De aquí, la localización del punto inicial es el paso crítico en este método y determina la calidad de la segmentación. Este proceso es ilustrado en la figura 7.6.

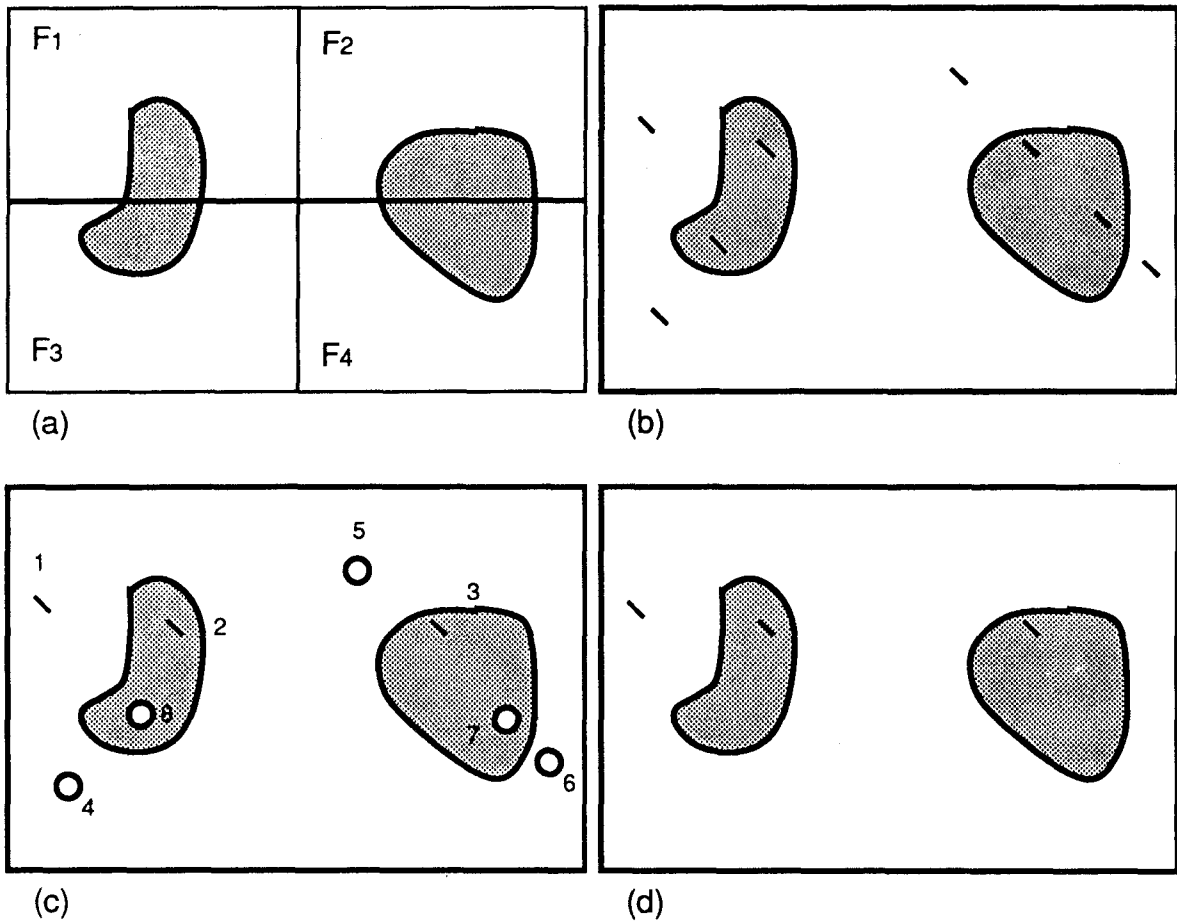


Fig. 7.6 Evaluación del punto inicial para el crecimiento de regiones.

Como un primer paso para realizar este procedimiento, se realiza un análisis de patrones dentro de una ventana F_i de la imagen para determinar la intensidad mínima I_{li} así como la intensidad máxima I_{ui} dentro de la ventana. Los valores locales de los umbrales S_i los cuales serán necesarios, pueden ser determinados de la ecuación siguiente

$$S_i = \frac{I_{ui} - I_{li}}{K} \quad (7.17)$$

donde K es una constante definida empíricamente la cual en muchos casos toma el valor de 3. Para evitar que se escoja un punto inicial que se encuentre en el borde de una región, se realiza un segundo paso en el cual todos aquellos puntos cuya intensidad de gradiente no exceda al valor ΔS son marcados. Ocho puntos son marcados de esta forma en la figura 7.6 (b); en la práctica el número de puntos debe ser de cientos o miles de píxeles. La determinación del valor del

ΔS no es muy crítico en este método, ya que normalmente se considera entre un 10% y un 20% del valor máximo de la intensidad. Comenzando a partir de los pixeles P_j (definidos por números del 1 al 8 en este ejemplo), aquellos pixeles P_i que van siguiendo una trayectoria arbitraria con P_j (donde la diferencia de intensidad entre P_j y todos los puntos a lo largo de la trayectoria no exceden al valor S_i) son removidos consecutivamente si P_i cae en F_i . En el ejemplo mostrado en la figura 7.6 (c) los puntos 4, 5 y 6 son removidos por el punto 1, el punto 8 es removido por el punto 2, y el punto 7 es removido por el punto 3 (estos están indicados por círculos). El valor de intensidad umbral S_i está excedido a lo largo de las trayectorias 1-2, 1-8, 2-3, 2-7, 3-2, 3-1, 1-3 y 1-7 (indicado por los cuadros). La trayectoria 3-6 no fué considerada porque el punto 6 fue eliminado por el punto 1 antes de iniciar el análisis. Los pixeles que quedan son mostrados en la figura 7.6 (d) donde puede verse que solamente un punto inicial se obtiene para cada región.

En la práctica no todas las trayectorias entre dos puntos pueden ser examinadas, por lo que solamente se limita a líneas rectas entre dos puntos y trayectorias paralelas a los dos ejes de coordenadas.

Este método ofrece mucho más ventajas que el método del umbral, ya que

- (a) No se requiere tener un conocimiento *a priori* de la imagen
- (b) El método es extremadamente insensitivo a distorsiones cuando los puntos iniciales han sido considerados correctamente

Este método es considerado como un método De Abajo-Arriba, porque el proceso progresa a partir de pixeles individuales incrementándose las regiones de la imagen hasta que finalmente toda la imagen es procesada.

7.3 METODOS DE DETECCION DE BORDES

El método del umbral no es confiable cuando existen objetos pequeños en la escena, debido a que los picos que producen en el histograma de la imagen son pequeños. De igual manera, si se tienen muchos objetos de diferente brillo, el proceso de segmentación resulta complicado porque los picos que representan a los objetos en el histograma se superponen. Así pues, lo deseable es que los objetos posean un brillo relativamente uniforme y que contrasten fuertemente con su contorno inmediato. Esto implica que la variación de nivel de gris ocurre gradualmente dentro de cada objeto y la variación en los bordes de los objetos es muy grande. En tales casos, los objetos se pueden extraer por Detección de Bordes, donde se considera la detección de los pixeles en los cuales la razón de cambio de los niveles de gris respecto al espacio es muy grande [7]

El método clásico de detectar los bordes en una imagen es aplicar un operador derivada tal como el Gradiente. Dicho operador dará valores altos en los bordes, sin importar las orientaciones. En este material, se utilizarán aproximaciones discretas del gradiente tales como el operador de Roberts y el operador de Sobel.

Otro método consiste en conocer la forma del objeto deseado, para que sea reconocido por medio de un proceso de comparación entre patrones o plantillas. La forma más primitiva es comparar el patrón contra la imagen, pixel a pixel, en todas las orientaciones y posiciones posibles. La comparación se puede realizar a través de diferencias a nivel entre pixel y pixel o determinando alguna función de correlación. Este método resulta computacionalmente costoso ya que se usa para detectar partes pequeñas de objetos.

Un método menos costoso computacionalmente, conocido como la Transformada de Hough, se puede utilizar para objetos extensos con pocos cambios, tales como una curva delgada o el borde de un objeto. La idea de estas transformaciones es mapear hacia un espacio transformando definido de tal manera que los pixeles de las curvas o bordes de interés sean proyectados sobre un mismo punto en ese espacio.

7.3.1 OPERADORES DE ROBERTS Y SOBEL [8]

Un borde en una imagen puede ser definido como una discontinuidad o un cambio abrupto en el nivel de gris. Los bordes pueden ser de diferente tamaño sin embargo, lo más significativo es que contrastan fuertemente sin importar su orientación. Idealizando, se puede modelar matemáticamente un borde como una función escalón por lo que la técnica de convolución gaussiana también es útil para la descripción de imágenes digitalizadas.

El operador gradiente aplicado a funciones continuas produce un vector en cada punto cuya dirección da la dirección del máximo cambio de la función en este punto, y cuya magnitud es la magnitud del máximo cambio. Su definición para una función continua de dos variables $f(x,y)$ está dada por:

$$\nabla f = \frac{\partial f}{\partial x} i + \frac{\partial f}{\partial y} j \quad (7.18)$$

donde i y j son vectores unitarios en el eje x y y respectivamente.

Un gradiente en forma digital puede ser calculado convolucionando dos ventanas con una imagen, una ventana da la componente en x , g_x , del gradiente y la otra da la componente en y , g_y . Los operadores de Roberts y Sobel, son aproximaciones discretas del gradiente y actúan como una máscara

convolucionando sobre la función de la imagen digitalizada. En forma general el proceso de convolucionar la máscara con la imagen se muestra en la figura 7.7

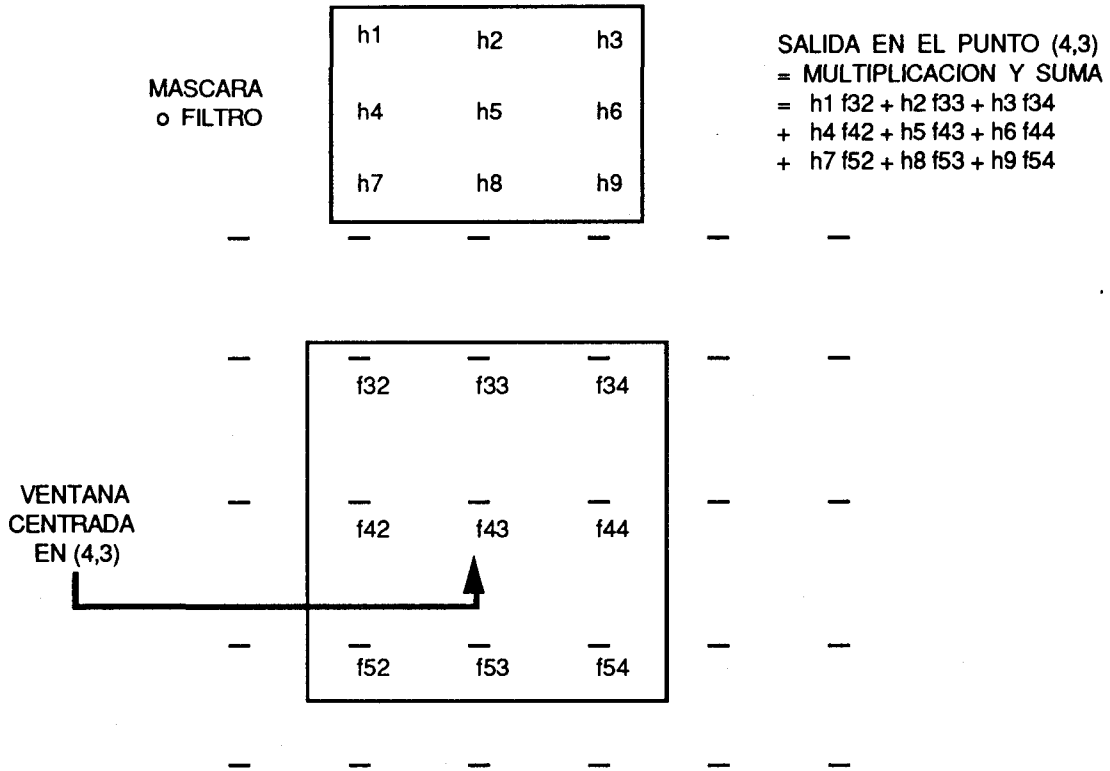


Fig. 7.7 Operación de convolución de una ventana con una imagen digitalizada.

El operador cruzado de Roberts está definido como una ventana 2x2 actuando sobre los píxeles de una imagen. Considerando la figura 7.8

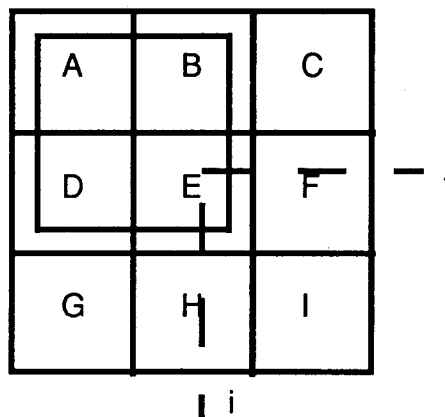


Fig. 7.8

Se define el OPERADOR CRUZADO DE ROBERTS como

$$g(i, j) = \text{Max}(|A-E|, |B-D|) \quad (7.19)$$

La ecuación anterior proviene de considerar al operador como una ventana tal que:

$$g_x(i, j) = \text{Masc}_x * p(i, j) \quad g_y(i, j) = \text{Masc}_y * p(i, j) \quad (7.20)$$

donde $p(i, j)$ representa a los vecinos del pixel $P(i, j)$ y

$$\text{Masc}_x = \begin{matrix} 1 & 0 \\ 0 & -1 \end{matrix} \quad \text{Masc}_y = \begin{matrix} 0 & 1 \\ -1 & 0 \end{matrix}$$

El operador gradiente produce dos vectores (g_x y g_y) para cada pixel además, el operador de Roberts genera éstos vectores con un corrimiento de tal forma que se encuentran centrados en $(i, j + 1/2)$ para el vector g_x y en $(i-1/2, j)$ para el vector g_y . Para obtener vectores centrados, se hace uso de máscaras de 3×3 . Las máscaras definidas de 2×2 no funcionan muy bien cuando existe ruido en la imagen, mientras que las de 3×3 son más sensitivas cuando hay ruido.

Entre los operadores definidos como aproximaciones del gradiente con máscaras de 3×3 se encuentran el Operador de Sobel, El Laplaciano y el Bi-Laplaciano entre otros[9].

El Operador de Sobel fué diseñado como una aproximación discreta del gradiente sobre las componentes horizontal y vertical sobre la máscara. La figura 7.9 muestra dos máscaras, donde no debe dejarse de notar, en este y en todos los casos, que la suma de los pesos es siempre cero. De la misma forma, la suma de los pesos de los vecinos reflejados sobre el pixel (i, j) es siempre cero. En la figura 7.9 el Operador de Sobel ha sido aplicado en una inspección visual de ensamblado de válvulas.

Los Operadores Laplaciano y Bi-Laplaciano para funciones continuas están definidos por las ecuaciones (7.21) y (7.22). La aplicación de estos operadores se muestra en la figura 7.9.

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \quad (7.21)$$

$$\nabla^4 = \frac{\partial^4}{\partial x^2 \partial y^2} \quad (7.22)$$

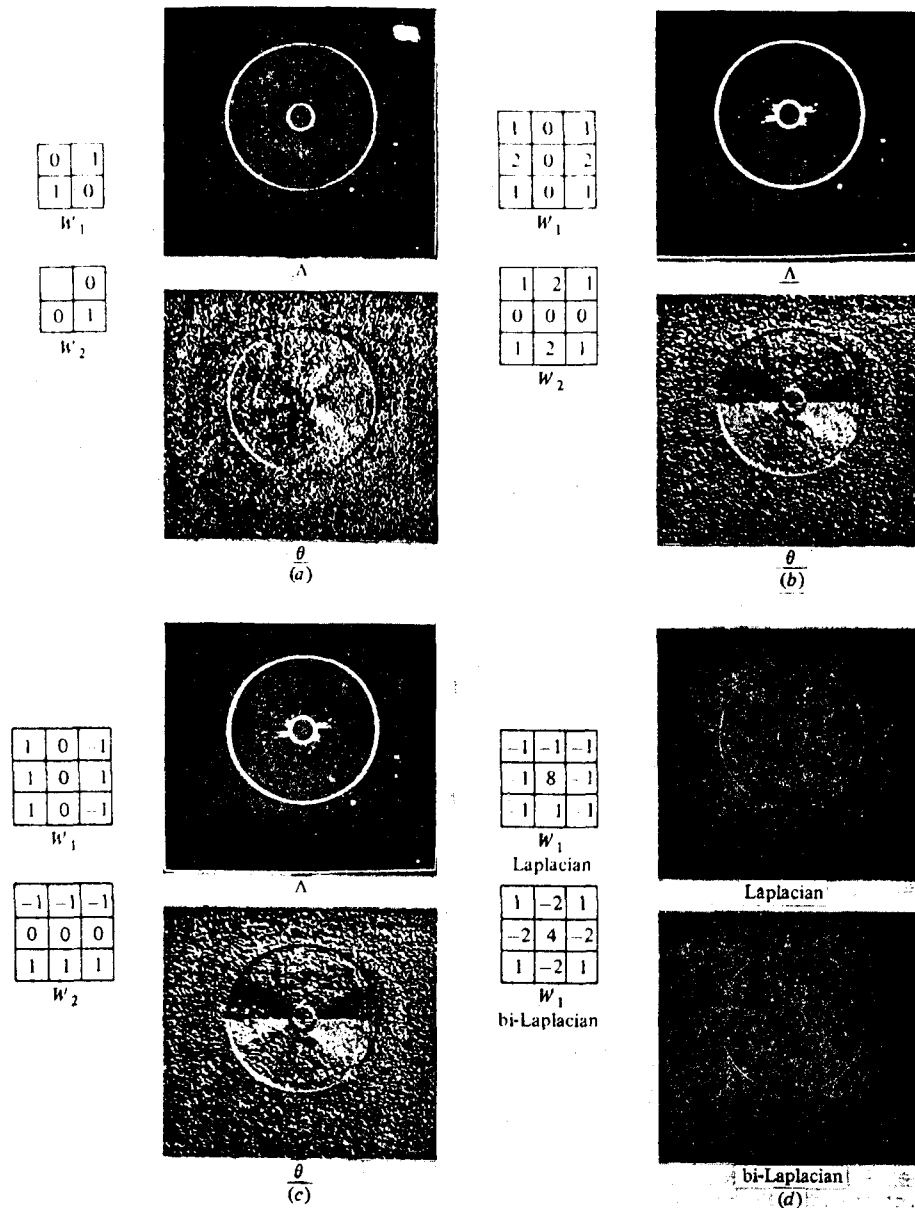


Fig. 7.9 Operadores de Sobel, Laplaciano y Bi-Laplaciano

7.3.2 LA TRANSFORMADA DE HOUGH [9]

La técnica de la Transformada de Hough es aplicable si se conoce algo de la localización de la frontera, o si la forma de frontera puede ser descrita por una curva paramétrica tal como una línea recta. Su principal ventaja es de que no se ve afectada por espacios entre curvas o por ruido. El concepto básico de esta técnica, es de que proyecta los elementos de la frontera del objeto, dando una evidencia local, a un dominio diferente donde las características globales son más claramente definidas. Primero se proyecta al espacio de la transformada y luego se realiza un análisis en ese espacio.

Para mostrar la técnica, consideremos los cuatro puntos colineales que se encuentran sobre un elemento del borde L' , tal como se muestra en la figura 7.10 (a). Considerando cualquiera de esos puntos, digamos el punto (x', y') , un número infinito de líneas pueden ser trazadas a través de dicho punto. Suponga que la línea L es una línea típica. Como se puede ver, esta línea está caracterizada por dos parámetros. Uno es el ángulo θ de la normal a la línea pasando por el origen $(0, 0)$. El otro es la longitud ρ de esta normal. Entonces, observamos que cualquier punto arbitrario (x, y) sobre la línea L está dado por la siguiente ecuación:

$$x \cos \theta + y \sin \theta = \rho \quad (7.23)$$

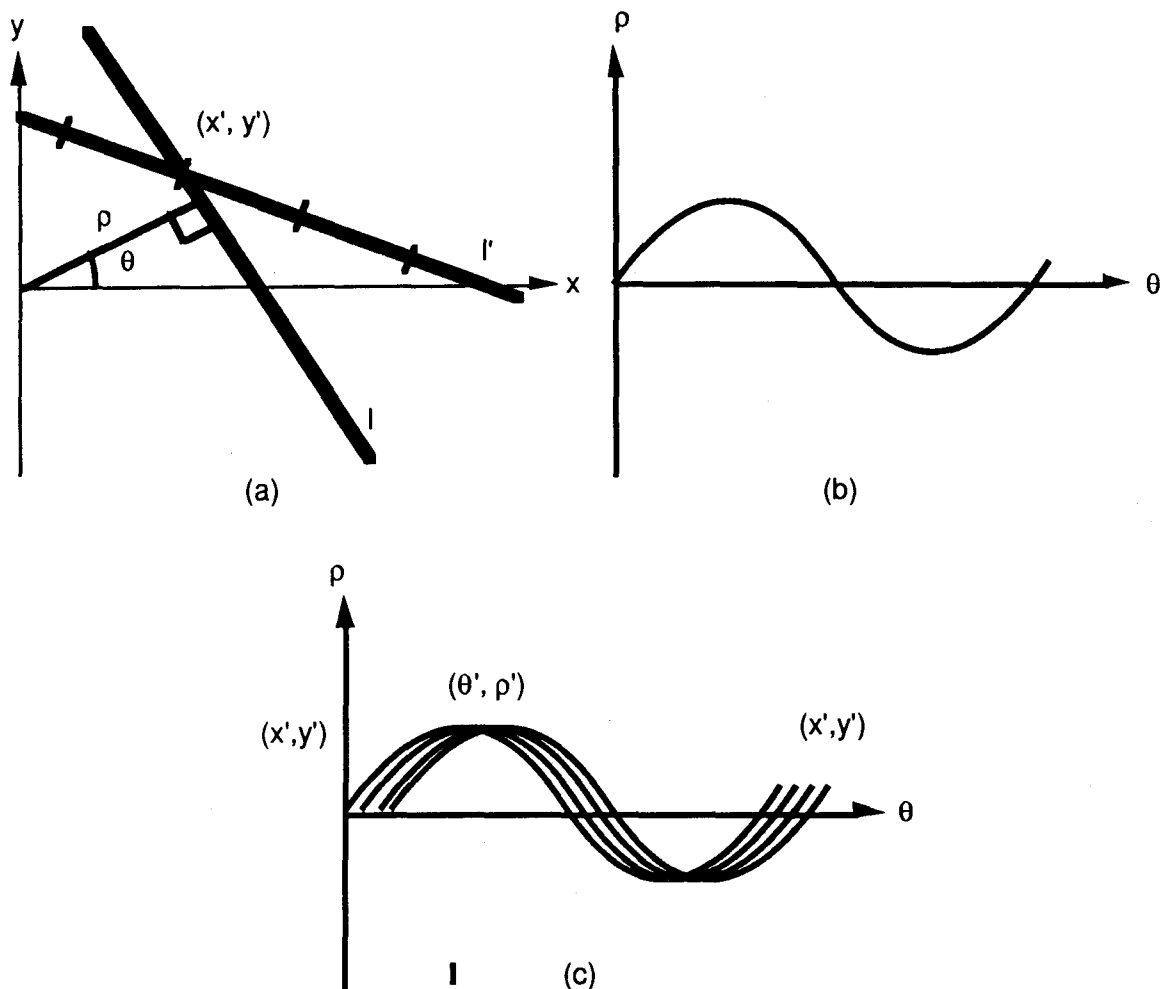


Fig. 7.10 La Transformada de Hough para puntos sobre una línea.(a) Parametrización de la línea L , donde θ y ρ son parámetros fijos (b) ρ como una función de θ , donde $\rho = x' \cos \theta + y' \sin \theta$ para parámetros fijos x' y y' . (c) Representación de los puntos de (a) En el espacio de la Transformada de Hough (θ', ρ') da los parámetros de la línea que pasa por los puntos.

En esta ecuación, tratando a θ y ρ como parámetros fijos, resulta una relación lineal entre las variables x y y . Por otro lado, si suponemos a (x, y) como un punto fijo en la ecuación (7.23), entonces esta ecuación nos definirá una relación entre los parámetros θ y ρ . Mas aún, esta relación es una función senoidal tal como se muestra en la figura 7.10(b). Cada par (θ, ρ) en la gráfica parametriza una de un número infinito de rectas que cruzan por el punto (x', y') . Esto se puede ver claramente si consideramos que la línea L rote alrededor del punto (x', y') ; θ cambiará hasta 360° mientras que ρ cambiará de valor entre dos valores límites. Observamos entonces que el punto (x', y') en la imagen original en el plano X-Y ha sido proyectado en el punto (θ, ρ) en el espacio de la Transformada de Hough en una curva senoidal. Aplicando la misma transformación para cada uno de los puntos restantes de figura 7.10(a), obtenemos las curvas que se muestran en la figura 7.10(c). Es importante hacer notar que cada uno de los cuatro puntos imagen están sobre la misma línea recta L' parametrizada por (θ', ρ') . Como la ecuación (7.23) debe ser cumplida para cada uno de los cuatro puntos, se sigue que cada una de las curvas transformadas pasan por el punto (θ', ρ') . Esta visto que la intersección caracteriza el valor de (θ', ρ') , lo cual define la línea que pasa por los cuatro puntos. Entonces el problema de detectar los puntos colineales de la imagen en el plano X-Y ha sido convertido a un problema simple en el cual la intersección de las curvas es buscada en el plano de la Transformada de Hough.

Un método muy simple de implementar que sirve para descubrir los puntos producidos por la intersección de las líneas, es llamado el METODO DEL ACUMULADOR. Consiste en cuantizar las variables θ y ρ en n y m niveles respectivamente, convirtiendo el espacio de la Transformada de Hough en un arreglo de $n \times m$ celdas (acumulador). Como la transformada para un valor (x', y') de cada punto de la imagen es graficado en el espacio θ - ρ , cada celda del acumulador es incrementada siempre que la curva senoidal pase a través de él. Después de que todos los puntos del plano X-Y han sido transformados, los elementos del arreglo con un alto valor en su contador pueden ser determinados por medio de un valor de umbral. Obviamente, estos picos en el arreglo corresponden a las intersecciones de muchas curvas en el espacio transformado, y los valores asociados de (θ, ρ) indican los parámetros de las líneas en el espacio imagen producidas por los puntos colineales. Por ejemplo, en el caso de figura 7.10(a), el arreglo de la celda (θ', ρ') deberá registrar en su contador un valor de 4; todas las otras celdas deberán ser iguales a 0.

La parametrización dada por la ecuación (7.23) puede ser cambiada por la representación para la línea dada por la ecuación

$$Y = mX + c \tag{7.24}$$

ya que los valores de θ y ρ están restringidos a los límites siguientes

$$0^\circ \leq \theta \leq 360^\circ$$

$$|p| \leq (M^2_{\max} + N^2_{\max})^{1/2} \quad (7.25)$$

donde M y N son las dimensiones de la imagen.

La figura 7.11 muestra el arreglo del acumulador para la transformada de Hough para una imagen que contiene cuatro curvas de puntos colineales.

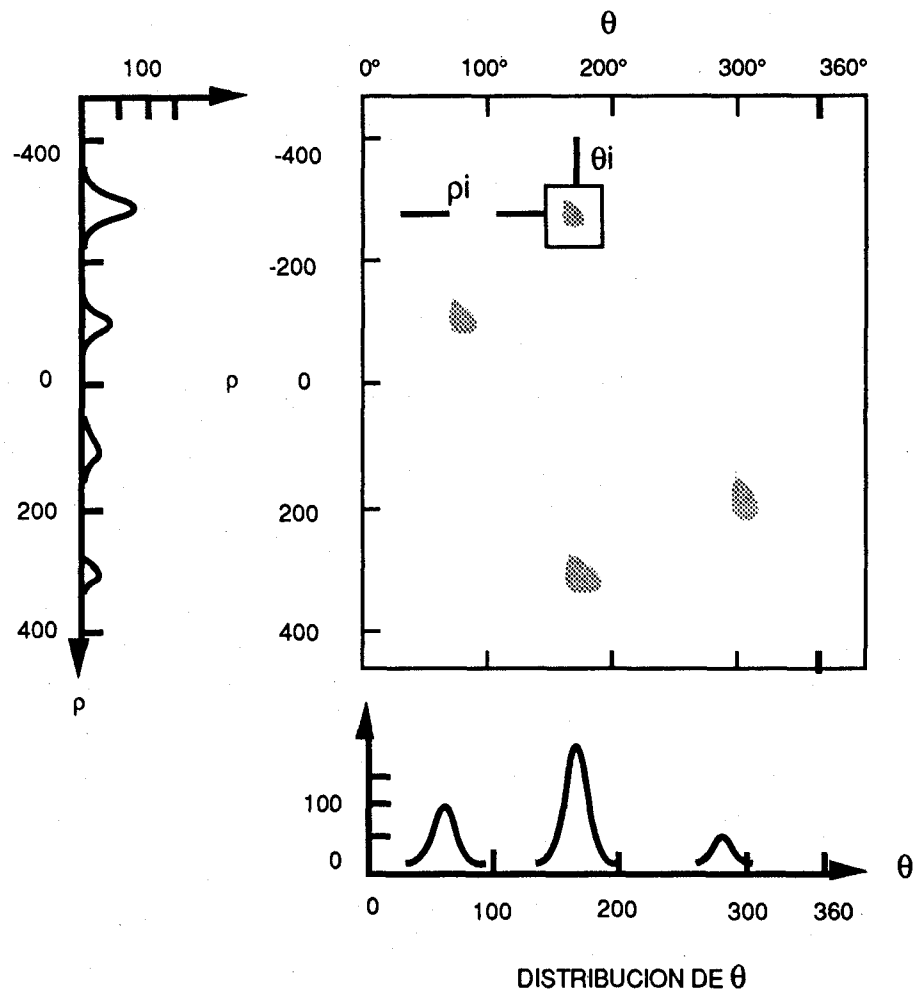


Fig. 7.11 Acumulador para la transformada de Hough para una imagen con cuatro curvas con puntos colineales.

La Transformada de Hough puede ser extendida para encontrar otras curvas típicas sin embargo, la complejidad se incrementa exponencialmente con el número de parámetros de la curva, lo cual gobierna la dimensión del espacio de la Transformada de Hough. Por ello, esta técnica solo es práctica para curvas con un número pequeño de parámetros.

Para el caso de líneas rectas, el ALGORITMO PARA LA DETECCION es el siguiente[10]:

1. Cuantizar los parámetros espaciales c y m entre valores apropiados para un máximo y un mínimo
2. Formar un arreglo (Acumulador), $A(c, m)$, cuyos elementos son inicialmente cero
3. Para cada punto (x,y) de la imagen tal que el valor de su gradiente exceda un valor de umbral, incrementar todos las celdas del acumulador

$$A(c,m) = A(c,m)+1$$

para m y c satisfaciendo la ecuación $c = -mX+Y$ dentro de los límites de la digitalización.

4. Los valores de las celdas del acumulador dan una medición del número de puntos sobre la línea.

CAPITULO 8

CONCLUSIONES

Se utilizó el sistema de visión como auxilio a la celda para facilitar su optimización. Esto permitió flexibilizar el uso de la celda de manufactura eliminando algunas de las restricciones que usualmente se tienen en este tipo de sistemas. En este caso se eliminó la necesidad de que la pieza al entrar quede ubicada en el mismo lugar ya que el sistema de visión determina la posición de la pieza. En la mesa de almacenaje intermedio se eliminó la necesidad de "saber" por programa, mediante el uso de contadores, si la pieza ya había sido maquinada por el torno o la fresadora, ya que el sistema de visión permite identificar este hecho. El sistema de visión identifica si la pieza que esta entrando a la celda corresponde a los programas con que están trabajando el torno y la fresadora en ese momento, si no es así detiene su trabajo y le avisa al operador para que este cargue los nuevos programas.

Después de haber efectuado un gran número de pruebas y corridas del sistema, los principales problemas que se presentaron fueron los siguientes:

1. En la parte correspondiente a la segmentación de la imagen se encontró como un problema grave el color de las piezas de trabajo, ya que dichas piezas son de color azul marino y el fondo, correspondiente a la banda de entrada y a la mesa intermedia, es de color negro por lo que se decidió pintar o marcar con color blanco las piezas de trabajo. Además de lo anterior, la segmentación de un objeto, que puede ser vista en el monitor de imágenes digitalizadas, cambiada cuando se realizaba sobre un mismo objeto por lo que el reconocimiento de los objetos se dificultaba.
2. En la ejecución del programa para piezas maquinables en torno y fresadora se presentaba el problema de la definición de un valor de umbral adecuado para la segmentación de las imágenes. Estas imágenes corresponden al momento en que la pieza de trabajo se encuentre sobre la banda de entrada o sobre la mesa intermedia de trabajo. Mientras que la banda de entrada es totalmente negra, la mesa intermedia es de color naranja excepto las cajas donde se posicionan las piezas de trabajo. Así, la banda de entrada puede tener un valor de umbral muy grande ya que existe una gran diferencia entre la intensidad de la pieza y la intensidad de la banda. Por otro lado, el valor de umbral adecuado para la imagen sobre la mesa intermedia era muy pequeño. Bajo condiciones de iluminación, se logró tener un valor de umbral adecuado para ambas imágenes.

3. En lo que respecta al equipo utilizado, usualmente el robot pierde la posición exacta de cada una de las posiciones previamente grabadas. Las cámaras, no funcionan correctamente ya que existe mucho ruido por parte de ellas.
4. El sistema de programación que se tiene, no contempla la opción de cambiar automáticamente los programas de las máquinas herramientas cuando una pieza diferente, a la con que se está trabajando, entra al sistema. Modificando el sistema de programación que proporciona el proveedor o desarrollando uno nuevo es posible ampliar el uso del sistema de visión para permitir que el sistema atienda automáticamente piezas diferentes, de una en una, haciendo que el cambio de los programas requeridos se realice en forma automática.

En la parte correspondiente a las técnicas para la detección de objetos por medio de sus bordes, se desarrollaron las ecuaciones necesarias para la operación óptima del umbral ya que el sistema de visión que se tiene hace uso de él. Tal como se muestra en la sección 7.2.1.2 donde el valor óptimo corresponde a la media entre la frecuencia de ocurrencia de los pixeles para el objeto y el fondo, el sistema de visión considera este valor promedio.

APENDICE A

"PROGRAMAS DEL ROBOT PARA LA INTEGRACION DE LOS ELEMENTOS DE LA CELDA Y EL SISTEMA DE VISION"

A1. PROGRAMA PARA LA INTEGRACION DEL SISTEMA DE MANUFACTURA UTILIZANDO PIEZAS MAQUINABLES EN EL TORNO Y LA FRESADORA

La descripción de este programa se encuentra en la sección 2.1 del capítulo 6 de este trabajo y se presenta además el diagrama de flujo correspondiente al programa.

```

1      TURN OFF OUTPUT #3
2      TURN OFF OUTPUT #4
3      GO POSITION 1 *FAST
4      SET COUNTER #1 TO 0
5      SET COUNTER #2 TO 0
6      SET COUNTER #3 TO 0
7      SET COUNTER #4 TO 0
8      SET COUNTER #5 TO 0
9      SET COUNTER #6 TO 0
10     SET COUNTER #7 TO 0
11     SET COUNTER #8 TO 0
12     SET COUNTER #9 TO 0
13     IF COUNTER #6 > 0 JUMP TO 25
14     IF COUNTER #7 > 0 JUMP TO 25
15     IF COUNTER #8 > 0 JUMP TO 25
16     IF COUNTER #9 > 0 JUMP TO 25
17     SCAN FRAME #1
18     IF OBJECT #3 EXISTS JUMP TO 25
19     TURN ON OUTPUT #3
20     IF OBJECT #4 EXISTS JUMP TO 23
21     IF OBJECT #5 EXISTS JUMP TO 23
22     JUMP TO LINE #13
23     TURN ON OUTPUT #4
24     JUMP TO LINE #26
25     TURN OFF OUTPUT #3
26     IF INPUT #2 ON JUMP TO 31
27     IF INPUT #1 ON JUMP TO 108
    
```

```

28     IF COUNTER #5 > 0 JUMP TO 376
29     JUMP TO LINE #13
30     REMARK: SUBROUTINA PARA ATENDER AL TORNO
31     IF COUNTER #1 > 0 JUMP TO 103
32     IF COUNTER #6 > 0 JUMP TO 37
33     IF COUNTER #7 > 0 JUMP TO 49
34     IF COUNTER #8 > 0 JUMP TO 61
35     IF COUNTER #9 > 0 JUMP TO 73
36     JUMP TO LINE #85
37     GO POSITION 60 *FAST
38     SCAN FRAME #2
39     IF OBJECT #2 EXISTS JUMP TO 41
40     JUMP TO LINE #33
41     GO POSITION 61 *FAST
42     GO POSITION 62 *FAST
43     GO POSITION 63 SPEED #2
44     CLOSE GRIPPER
45     GO POSITION 62 SPEED #2
46     GO POSITION #1 *FAST
47     SET COUNTER #6 TO 0
48     JUMP TO LINE #100
49     GO POSITION 70 *FAST
50     SCAN FRAME #2
51     IF OBJECT #2 EXISTS JUMP TO 53
52     JUMP TO LINE #34
53     GO POSITION 71 *FAST
54     GO POSITION 72 *FAST
55     GO POSITION 73 SPEED #2
56     CLOSE GRIPPER
57     GO POSITION 72 SPEED #2
58     GO POSITION 1 *FAST
59     SET COUNTER #7 TO 0
60     JUMP TO LINE #100
61     GO POSITION 80 *FAST
62     SCAN FRAME #2
63     IF OBJECT #2 EXISTS JUMP TO 65
64     JUMP TO LINE #35
65     GO POSITION 81 *FAST
66     GO POSITION 82 *FAST
67     GO POSITION 83 SPEED #2
68     CLOSE GRIPPER
69     GO POSITION 83 SPEED #2
70     GO POSITION 1 *FAST
71     SET COUNTER #8 TO 0
72     JUMP TO LINE #100
73     GO POSITION 90 *FAST
74     SCAN FRAME #2

```

```

75     IF OBJECT #2 EXISTS JUMP TO 77
76     JUMP TO LINE #85
77     GO POSITION 91 *FAST
78     GO POSITION 92 *FAST
79     GO POSITION 92 SPEED #2
80     CLOSE GRIPPER
81     GO POSITION 92 SPEED #2
82     GO POSITION 1 *FAST
83     SET COUNTER #9 TO 0
84     JUMP TO LINE #100
85     SCAN FRAME #1
86     IF OBJECT #3 EXISTS JUMP TO 88
87     JUMP TO LINE #13
88     GO TO OBJECT #3 SPEED #4
89     CLOSE GRIPPER
90     GO POSITION 85 *FAST
91     GO POSITION 86 *FAST
92     GO POSITION 87 SPEED #2
93     OPEN GRIPPER
94     GO POSITION 95 *FAST
95     GO POSITION 96 *FAST
96     GO POSITION 97 SPEED #2
97     CLOSE GRIPPER
98     GO POSITION 95 *FAST
99     JUMP TO LINE #102
100    SET COUNTER #2 TO 1
101    JUMP TO LINE #103
102    SET COUNTER #2 TO 0
103    IF COUNTER #1 > 0 JUMP TO 105
104    JUMP TO LINE #184
105    IF COUNTER #2 > 0 JUMP TO 216
106    JUMP TO LINE #263
107    REMARK: SUBROUTINA PARA ATENDER A LA FRESADORA
108    IF COUNTER #3 > 0 JUMP TO 180
109    IF COUNTER #6 > 0 JUMP TO 114
110    IF COUNTER #7 > 0 JUMP TO 126
111    IF COUNTER #8 > 0 JUMP TO 138
112    IF COUNTER #9 > 0 JUMP TO 150
113    JUMP TO LINE #162
114    GO POSITION 60 *FAST
115    SCAN FRAME #2
116    IF OBJECT #1 EXISTS JUMP TO 118
117    JUMP TO LINE #110
118    GO POSITION 61 *FAST
119    GO POSITION 62 *FAST
120    GO POSITION 63 SPEED #2
121    CLOSE GRIPPER

```



```

122 GO POSITION 62 SPEED #2
123 GO POSITION 1 *FAST
124 SET COUNTER #6 TO 0
125 JUMP TO LINE #177
126 GO POSITION 70 *FAST
127 SCAN FRAME #2
128 IF OBJECT #1 EXISTS JUMP TO 130
129 JUMP TO LINE 111
130 GO POSITION 71 *FAST
131 GO POSITION 72 *FAST
132 GO POSITION 73 SPEED #2
133 CLOSE GRIPPER
134 GO POSITION 72 SPEED #2
135 GO POSITION 1 *FAST
136 SET COUNTER #7 TO 0
137 JUMP TO LINE #177
138 GO POSITION 80 *FAST
139 SCAN FRAME #2
140 IF OBJECT #1 EXISTS JUMP TO 142
141 JUMP TO LINE #112
142 GO POSITION 81 *FAST
143 GO POSITION 82 *FAST
144 GO POSITION 83 SPEED #2
145 CLOSE GRIPPER
146 GO POSITION 82 SPEED #2
147 GO POSITION 1 *FAST
148 SET COUNTER #8 TO 0
149 JUMP TO LINE #177
150 GO POSITION 90 *FAST
151 SCAN FRAME #2
152 IF OBJECT #1 EXISTS JUMP TO 154
153 JUMP TO LINE #162
154 GO POSITION 91 *FAST
155 GO POSITION 92 *FAST
156 GO POSITION 93 SPEED #2
157 CLOSE GRIPPER
158 GO POSITION 92 SPEED #2
159 GO POSITION 1 *FAST
160 SET COUNTER #9 TO 0
161 JUMP TO LINE #177
162 SCAN FRAME #1
163 IF OBJECT #3 EXISTS JUMP TO 165
164 JUMP TO LINE #13
165 GO TO OBJECT #3 *FAST
166 CLOSE GRIPPER
167 GO POSITION 85 *FAST
168 GO POSITION 86 *FAST

```

```

169 GO POSITION 87 SPEDD #2
170 OPEN GRIPPER
171 GO POSITION 95 *FAST
172 GO POSITION 96 *FAST
173 GO POSITION 97 SPEED #2
174 CLOSE GRIPPER
175 GO POSITION 95 *FAST
176 JUMP TO LINE #179
177 SET COUNTER #4 TO 1
178 JUMP TO LINE $180
179 SET COUNTER #4 TO 0
180 IF COUNTER #3 > 0 JUMP TO 182
181 JUMP TO LINE 202
182 IF COUNTER #4 > 0 JUMP TO 241
183 JUMP TO LINE #281
184 REMARK: CARGAR PIEZA DE ENTRADA O DE MESA INTER.
ALTORNO
185 GO POSITION 1 *FAST
186 GO POSITION 40 *FAST
187 GO POSITION 44 *FAST
188 GO POSITION 45 SPEED #2
189 GO POSITION 46 SPEED #1
190 CALL SUBROUTINE #2
191 IF INPUT #2 ON JUMP TO 193
192 JUMP TO LINE #191
193 OPEN GRIPPER
194 CALL SUBROUTINE #2
195 SET COUNTER #1 TO 1
196 GO POSITION 45 SPEED #2
197 GO POSITION 44 SPEED #3
198 GO POSITION 40 *FAST
199 GO POSITION 1 *FAST
200 JUMP TO LINE #13
201 REMARK: CARGAR PIEZA DE ENTRADA O DE MESA INTER. A
FRESADORA
202 GO POSITION 1 *FAST
203 GO POSITION 50 *FAST
204 GO POSITION 51 *FAST
205 GO POSITION 52 SPEED #2
206 GO POSITION 53 SPEED #1
207 OPEN GRIPPER
208 CALL SUBROUTINE #1
209 SET COUNTER #3 TO 1
210 GO POSITION 52 SPEED #2
211 GO POSITION 51 *FAST
212 GO POSITION 50 *FAST
213 GO POSITION 1 *FAST

```

214 JUMP TO LINE #13
 215 REMARK: RETIRAR PIEZA DEL TORNO A LA SALIDA
 216 GO POSITION 1 *FAST
 217 GO POSITION 40 *FAST
 218 OPEN GRIPPER
 219 GO POSITION 41 *FAST
 220 GO POSITION 42 SPEED #2
 221 GO POSITION 43 SPEED #1
 222 CLOSE GRIPPER
 223 CALL SUBROUTINE #2
 224 WAIT 3 SECONDS
 225 GO POSITION 42 SPEED #1
 226 GO POSITION 41 SPEED #2
 227 GO POSITION 40 *FAST
 228 GO POSITION 1 *FAST
 229 GO POSITION 30 *FAST
 230 GO POSITION 31 *FAST
 231 GO POSITION 32 SPEED #1
 232 OPEN GRIPPER
 233 GO POSITION 33 *FAST
 234 SET COUNTER #1 TO 0
 235 SET COUNTER #2 TO 0
 236 GO POSITION 31 *FAST
 237 GO POSITION 30 *FAST
 238 GO POSITION 1 *FAST
 239 JUMP TO LINE #13
 240 REMARK: RETIRAR PIEZA DE LA FRESADORA A LA SALIDA
 241 GO POSITION 1 *FAST
 242 GO POSITION 50 *FAST
 243 OPEN GRIPPER
 244 GO POSITION 51 *FAST
 245 GO POSITION 52 SPEED #2
 246 GO POSITION 53 SPEED #1
 247 CLOSE GRIPPER
 248 CALL SUBROUTINE #1
 249 GO POSITION 52 SPEED #2
 250 GO POSITION 51 *FAST
 251 GO POSITION 30 *FAST
 252 GO POSITION 31 *FAST
 253 GO POSITION 32 SPEED #1
 254 OPEN GRIPPER
 255 GO POSITION 33 *FAST
 256 SET COUNTER #3 TO 0
 257 SET COUNTER #4 TO 0
 258 GO POSITION 31 *FAST
 259 GO POSITION 30 *FAST
 260 GO POSITION 1 *FAST

```

261 JUMP TO LINE #13
262 REMARK: RETIRAR PIEZA DEL TORNO A LA MESA INTERMEDIA
263 GO POSITION 1 *FAST
264 OPEN GRIPPER
265 GO POSITION 40 *FAST
266 GO POSITION 41 *FAST
267 GO POSITION 42 SPEED #2
268 GO POSITION 43 SPEED #1
269 CLOSE GRIPPER
270 CALL SUBROUTINE #2
271 WAIT 3 SECONDS
272 GO POSITION 42 SPEED #1
273 GO POSITION 41 SPEED #2
274 GO POSITION 40 *FAST
275 GO POSITION 1 *FAST
276 SET COUNTER #1 TO 0
277 SET COUNTER #2 TO 0
278 CALL SUBROUTINE #3
279 JUMP TO LINE #13
280 REMARK: RETIRAR PIEZA DE LA FRESADORA A LA MESA
INTERM.
281 GO POSITION 1 *FAST
282 GO POSITION 50 *FAST
283 OPEN GRIPPER
284 GO POSITION 51 *FAST
285 GO POSITION 52 SPEED #2
286 GO POSITION 53 SPEED #1
287 CLOSE GRIPPER
288 CALL SUBROUTINE #1
289 GO POSITION 52 SPEED #1
290 GO POSITION 51 *FAST
291 GO POSITION 1 *FAST
292 SET COUNTER #3 TO 0
293 SET COUNTER #4 TO 0
294 CALL SUBROUTINE #3
295 JUMP TO LINE #13
296 REMARK:****
297 REMARK:****
298 SET SUBROUTINE #1
299 REMARK: DESCONECTAR LA FRESADORA
300 TURN ON OUTPUT #1
301 IF INPUT #1 ON JUMP TO 300
302 TURN OFF OUTPUT #1
303 RETURN FROM SUBROUTINE
304 REMARK:****
305 SET SUBROUTINE #2
306 REMARK: DESCONECTAR EL TORNO

```

```

307   TURN ON OUTPUT #2
308   IF INPUT #2 ON JUMP TO 307
309   TURN OFF OUTPUT #2
310   RETURN FORM SUBROUTINE
311   REMARK:****
312   REMARK:****
313   SET SUBROUTINE #3
314   REMARK: BUSCAR POSICION VACIA EN LA MESA INTERM.
315   IF COUNTER #6 > 0 JUMP TO 317
316   JUMP TO LINE #323
317   IF COUNTER #7 > 0 JUMP TO 319
318   JUMP TO LINE #333
319   IF COUNTER #8 > 0 JUMP TO 321
320   JUMP TO LINE #343
321   IF COUNTER #9 > 0 JUMP TO 363
322   JUMP TO LINE #353
323   GO POSITION 60 *FAST
324   GO POSITION 61 *FAST
325   GO POSITION 62 *FAST
326   GO POSITION 63 SPEED #2
327   OPEN GRIPPER
328   GO POSITION 62 SPEED #2
329   SET COUNTER #6 TO 1
330   GO POSITION 1 *FAST
331   JUMP TO LINE #372
332   REMARK:****
333   GO POSITION 70 *FAST
334   GO POSITION 71 *FAST
335   GO POSITION 72 *FAST
336   GO POSITION 73 SPEED #2
337   OPEN GRIPPER
338   GO POSITION 72 SPEED #2
339   SET COUNTER #7 TO 1
340   GO POSITION 1 *FAST
341   JUMP TO LINE #372
342   REMARK:****
343   GO POSITION 80 *FAST
344   GO POSITION 81 *FAST
345   GO POSITION 82 *FAST
346   GO POSITION 83 SPEED #2
347   OPEN GRIPPER
348   GO POSITION 82 SPEED #2
349   SET COUNTER #8 TO 1
350   GO POSITION 1 *FAST
351   JUMP TO LINE #372
352   REMARK:****
353   GO POSITION 90 *FAST

```

```

354 GO POSITION 91 *FAST
355 GO POSITION 92 *FAST
356 GO POSITION 93 SPEED #2
357 OPEN GRIPPER
358 GO POSITION 92 SPEED #2
359 SET COUNTER #9 TO 1
360 GO POSITION 1 *FAST
361 JUMP TO LINE #372
362 REMARK:****
363 SET COUNTER #5 TO 1
364 GO POSITION 20 *FAST
365 GO POSITION 21 *FAST
366 GO POSITION 22 SPEED #2
367 GO POSITION 23 SPEED #1
368 OPEN GRIPPER
369 GO POSITION 22 SPEED #2
370 GO POSITION 1 *FAST
371 JUMP TO LINE #373
372 SET COUNTER #5 TO 0
373 RETURN FROM SUBROUTINE
374 REMARK:****
375 REMARK: RECOGER PIEZA EN PARADA INTERMEDIA
376 GO POSITION 1 *FAST
377 GO POSITION 20 *FAST
378 OPEN GRIPPER
379 GO POSITION 21 *FAST
380 GO POSITION 22 SPEED #2
381 GO POSITION 23 SPEED #1
382 CLOSE GRIPPER
383 GO POSITION 22 SPEED #2
384 GO POSITION 1 *FAST
385 CALL SUBROUTINE #3
386 JUMP TO LINE #13

```

A2. PROGRAMA PARA LA INTEGRACION DEL SISTEMA DE MANUFACTURA UTILIZANDO PIEZAS MAQUINABLES UNICAMENTE EN EL TORNO O EN LA FRESADORA

La descripción de este programa se encuentra en la sección 2.2 del capítulo 6 de este trabajo y se presenta además el diagrama de flujo correspondiente al programa.

```

1      GO POSITION 1 *FAST
2      SET COUNTER #1 TO 0
3      SET COUNTER #2 TO 0
4      TURN OFF OUTPUT #3
5      TURN OFF OUTPUT #4
6      SCAN FRAME #1
7      IF OBJECT #4 EXISTS JUMP TO 15
8      SCAN FRAME #1
9      IF OBJECT #5 EXISTS JUMP TO 19
10     TURN ON OUTPUT #3
11     IF OBJECT #3 EXISTS JUMP TO 13
12     JUMP TO LINE #6
13     TURN ON OUTPUT #4
14     JUMP TO LINE #20
15     TURN OFF OUTPUT #3
16     IF INPUT #2 ON JUMO TO 24
17     IF INPUT #1 ON JUMP TO 81
18     JUMP TO LINE #6
19     TURN OFF OUTPUT #3
20     IF INPUT #1 ON JUMP TO 81
21     IF INPUT #2 ON JUMP TO 24
22     JUMP TO LINE #6
23     REMARK: SUBROUTINA PARA ATENDER AL TORNO
24     IF COUNTER #1 > 0 JUMP TO 57
25     REMARK: DAR PIEZA AL TORNO
26     SCAN FRAME #1
27     IF OBJECT #4 EXISTS JUMP TO 29
28     JUMP TO LINE #6
29     GO TO OBJECT #4 *FAST
30     CLOSE GRIPPER
31     GO POSITION 10 *FAST
32     GO POSITION 11 *FAST
33     GO POSITION 12 SPEED #2
34     OPEN GRIPPER
35     GO POSITION 20 *FAST
36     GO POSITION 21 *FAST
37     GO POSITION 22 SPEED #2
38     CLOSE GRIPPER
39     GO POSITION 20 *FAST

```

```

40 GO POSITION 1 *FAST
41 GO POSITION 30 *FAST
42 GO POSITION 31 *FAST
43 GO POSITION 32 SPEED #1
44 GO POSITION 33 SPEED #1
45 CALL SUBROUTINE #2
46 IF INPUT #2 ON JUMP TO 48
47 JUMP TO LINE #46
48 OPEN GRIPPER
49 CALL SUBROUTINE #2
50 SET COUNTER #1 TO 1
51 GO POSITION 32 SPEED #2
52 GO POSITION 32 SPEED #2
53 GO POSITION 30 *FAST
54 GO POSITION 1 *FAST
55 JUMP TO LINE #6
56 REMARK: RETIRAR PIEZA DEL TORNO A SALIDA
57 GO POSITION 1 *FAST
58 GO POSITION 30 *FAST
59 OPEN GRIPPER
60 GO POSITION 31 *FAST
61 GO POSITION 32 SPEED #2
62 GO POSITION 33 SPEED #1
63 CLOSE GRIPPER
64 CALL SUBROUTINE #2
65 WAIT 3 SECONDS
66 GO POSITION 32 SPEED #1
67 GO POSITION 31 SPEED #2
68 GO POSITION 30 *FAST
69 GO POSITION 1 *FAST
70 GO POSITION 40 *FAST
71 GO POSITION 41 *FAST
72 GO POSITION 42 SPEED #1
73 OPEN GRIPPER
74 GO POSITION 43 *FAST
75 SET COUNTER #1 TO 0
76 GO POSITION 41 *FAST
77 GO POSITION 40 *FAST
78 GO POSITION 1 *FAST
79 JUMP TO LINE #6
80 REMARK: SUBROUTINA PARA ATENDER A LA FRESADORA
81 IF COUNTER #2 > 0 JUMP TO 103
82 REMARK: DAR PIEZA A LA FRESADORA
83 SCAN FRAME #1
84 IF OBJECT #5 EXISTS JUMP TO 86
85 JUMP TO LINE #6
86 GO TO OBJECT #5 *FAST

```



```

87      CLOSE GRIPPER
88      GO POSITION 50 *FAST
89      GO POSITION 51 *FAST
90      GO POSITION 52 *FAST
91      GO POSITION 1 *FAST
92      GO POSITION 53 *FAST
93      GO POSITION 54 SPEED #2
94      GO POSITION 55 SPEED #1
95      OPEN GRIPPER
96      CALL SUBROUTINE #1
97      SET COUNTER #2 TO 1
98      GO POSITION 54 SPEED #2
99      GO POSITION 53 *FAST
100     GO POSITION 1 *FAST
101     JUMP TO LINE #6
102     REMARK: RETIRAR PIEZA DE LA FRESADORA A SALIDA
103     GO POSITION 1 *FAST
104     GO POSITION 53 *FAST
105     OPEN GRIPPER
106     GO POSITION 54 SPEED #2
107     GO POSITION 53 SPEED #1
108     CLOSE GRIPPER
109     CALL SUBROUTINE #1
110     GO POSITION 54 SPEED #2
111     GO POSITION 53 *FAST
112     GO POSITION 40 *FAST
113     GO POSITION 41*FAST
114     GO POSITION 42 SPEED #2
115     OPEN GRIPPER
116     GO POSITION 43 *FAST
117     SET COUNTER #2 TO 0
118     GO POSITION 41 *FAST
119     GO POSITION 40 *FAST
120     GO POSITION 1 *FAST
121     JUMP TO LINE #6
122     SET SUBROUTINE #1
123     REMARK: APAGAR FRESADORA
124     TURN ON OUTPUT #1
125     IF INPUT #1 ON JUMP TO 124
126     TURN OFF OUTPUT #1
127     RETURN FROM SUBROUTINE
128     SET SUBROUTINE #2
129     REMARK: APAGAR TORNO
130     TURN ON OUTPUT #2
131     IF INPUT #2 ON JUMP TO 130
132     TURN OFF OUTPUT #2
133     RETURN FROM SUBROUTINE

```

BIBLIOGRAFIA

- [1] Mikell P. Groover, "Automation, Production Systems, and Computer-Integrated Manufacturing, Englewood Cliffs, NJ, Prentice Hall, 1987.
- [2] James A. Gupton, Jr., "Computer-Controlled Industrial Machines, Processes and Robots, Englewood Cliffs, NJ, Prentice Hall, 1986.
- [3] "Scorbot ER-III and Robotvision Users' Manual", Prep-Inc, Tel-Aviv 61282, Israel, 1984.
- [4] K. S. Fu, R. C. González, "Robotics, Control Sensing, Vision and Inteligence", New York, McGraw Hill, 1987.
- [5] Dana H. Ballard, Cristipher M Brown, "Computer Vision", Englewood Cliffs, NJ, Prentice Hall, 1982.
- [6] Friedrich M Wahl, "Digital Image Signal Processing", Norwood MA 02062, Artech House Inc.,1987.
- [7] I. Alexander, "Artificial Vision for Robots", New York, Chapman and Hall, 1984.
- [8] Wayne Niblack, "An Introduction to Digital Image Processing", Englewood Cliffs, NJ, Prentice Hall, 1986.
- [9] Martin D. Levine, "Vision in Man and Machine", New York, McGraw Hill, 1985.
- [10] Robert J. Schalkoff, "Digital Image Processing and Computer Vision", New York, John Wiley and Sons, Inc, 1989.

Centro de Información-Bibliotecas



30002005147570