

# UNIVERSIDAD DON BOSCO



## **DISEÑO DE CONTROLADORES DIGITALES POR MEDIO DE UNA IBM PC USANDO MODELADO MATEMATICO Y TECNICAS DE CONTROL DIGITAL**



**TRABAJO DE GRADUACION**

**PREPARADO PARA LA**

**FACULTAD DE INGENIERIA**

**PARA OPTAR AL GRADO DE**

**INGENIERO EN ELECTRONICA**

**POR**

**JUAN CARLOS NÚÑEZ  
RICARDO ZELAYA MARTÍNEZ**

**ABRIL-1997**

**SOYAPANGO**

**EL SALVADOR**

**CENTROAMERICA**

**UNIVERSIDAD  
DON BOSCO**

**RECTOR**

**ING. FEDERICO MIGUEL HUGUET RIVERA**



**SECRETARIO GENERAL**

**PRBO. PEDRO JOSE GARCIA CASTRO**

**DECANO DE LA FACULTAD DE INGENIERIA**

**ING. CARLOS ALBERTO GUTIERREZ**

**ASESOR DE TRABAJO DE GRADUACION**

**MSc. RIGOBERTO CHINCHILLA**

**JURADO EXAMINADOR**

**ING. NELSON QUINTANILLA**

**ING. JAIME VARELA**

**UNIVERSIDAD DON BOSCO**

**FACULTAD DE INGENIERIA**

**DEPARTAMENTO DE INGENIERIA ELECTRONICA**

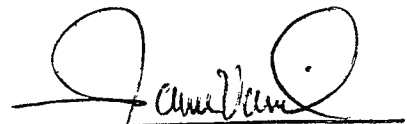
**JURADO EVALUADOR DEL TRABAJO DE GRADUACION**

**“ DISEÑO DE CONTROLADORES DIGITALES POR MEDIO DE UNA IBM PC USANDO  
MODELADO MATEMATICO Y TECNICAS DE CONTROL DIGITAL ”**



**ING. NELSON QUINTANILLA**

**JURADO**



**ING. JAIME VARELA**

**JURADO**



**MSc. RIGOBERTO CHINCHILLA**

**ASESOR**

Este trabajo lo dedico a mis padres Juan y Alba; y a Sayira, por su amor y apoyo para el logro de este triunfo.

JUAN CARLOS NUÑEZ

Este trabajo lo dedico a DIOS todopoderoso por darme voluntad y fuerzas para concluir esta meta tan ansiada; a mis padres Carlos y Lelys por todos los sacrificios y esfuerzos que hicieron para poder forjarme como profesional; a mis hermanas, familiares y demás amigos por brindarme todo su apoyo.

RICARDO ZELAYA MARTINEZ

## LISTA DE CONTENIDO

### Capítulo I : INTRODUCCIÓN

1.1 Motivación	1
1.2 Objetivos	2
1.3 Ventajas	3

### Capítulo II : DISEÑO DEL CIRCUITO ADQUISIDOR DE DATOS

2.1 Introducción	5
2.2 Diagrama de bloques del circuito adquisidor de datos	5
2.3 Sistema de bus IBM PC	7
2.4 Diseño del puerto paralelo de I/O	10
2.5 Circuito divisor de frecuencia	17
2.6 Diseño del circuito temporizador	18
2.7 Diseño del circuito convertidor analogo-digital	33

### Capítulo III. USO DEL ALGORITMO ERA

3.1 Introducción	41
3.2 Definición de función de transferencia	41
3.3 Respuesta al pulso discreto	43
3.4 Relación entre la respuesta al pulso discreto y la respuesta al impulso continuo	47

3.5 Respuesta al escalón unitario	48
3.6 Definición de la ecuación de diferencias finitas	51
3.7 Representación de sistemas dinámicos en el espacio de estado	55
3.8 Controlabilidad	58
3.9 Observabilidad	61
3.10 La matriz de Hankel y la determinación del orden del modelo del sistema	62
3.11 Descripción del algoritmo ERA	67
3.12 Ejemplo del uso del ERA en el modelado de un sistema hidráulico	71

#### Capítulo IV. DISEÑO DE CONTROLADORES DIGITALES Y SIMULACIÓN.

4.1 Introducción	85
4.2 Uso del equipo	85
4.3 Ejemplo de diseño de controladores	87

CONCLUSIONES Y RECOMENDACIONES	105
--------------------------------	-----

#### APENDICES

Apéndice A: Diseño de controladores digitales por el método de respuesta en frecuencia	107
Apéndice B: Filtros diseñados para las distintas constantes de tiempo de los sistemas a modelar	124
Apéndice C: Programas en MATLAB que desarrollan el algoritmo ERA.	127
Apéndice D: Manual de uso	131

## LISTA DE FIGURAS

Figura 2.1 Diagrama de bloques del adquisidor de datos	6
Figura 2.2 Sistema de bus IBM	8
Figura 2.3 Operación básica de I/O	11
Figura 2.4 Handshaking usando una bandera de dato disponible	12
Figura 2.5 Circuito amplificador	15
Figura 2.6 Circuito decodificador	16
Figura 2.7 Circuito divisor de frecuencias	18
Figura 2.8 Pin-out del 8253	21
Figura 2.9 Circuito temporizador	25
Figura 2.10 Diagrama de flujo de la función timer	29
Figura 2.11 Circuito para generar una señal de reloj a la entrada START del ADC	33
Figura 2.12 Circuito convertidor análogo-digital	36
Figura 2.13 Registro de estados para sondear las señales TCLK y EOC.	37
Figura 2.14 Circuito convertidor digital-análogo	38
Figura 2.15 Diagrama de flujo del programa de muestreo	39
Figura 3.1: Definición de función de transferencia	42
Figura 3.2: Respuesta impulso	43
Figura 3.3: Respuesta al pulso de un sistema lineal invariante en el tiempo	43
Figura 3.4: Generación de la respuesta discreta al pulso	44
Figura 3.5: Secuencia de pulsos de entrada discretos	46
Figura 3.6: Sistema hidráulico a modelar	74
Figura 3.7: Relación entrada/salida del sistema hidráulico	75



Figura 3.8: Respuesta al escalón experimental	76
Figura 3.9: Respuesta pulso del sistema hidráulico	77
Figura 3.10: Valores singulares de la matriz de Hankel	78
Figura 3.11: Comparación del modelo experimental con el modelo de 2º orden	79
Figura 3.12: Lugar de las raíces del modelo ERA de 2º orden	80
Figura 3.13: Comparación del modelo experimental con el modelo ERA de 4º orden	81
Figura 3.14: Lugar de las raíces del modelo ERA de 4º orden	82
Figura 3.15: Comparación del modelo experimental con el modelo de 6º orden	83
Figura 3.16: Lugar de las raíces del modelo ERA de 6º orden	84
Figura 4.1 Esquema para la adquisición de datos	85
Figura 4.2 Conexión de la señal de salida a la PC	86
Figura 4.3 Respuesta escalón experimental del motor DC	94
Figura 4.4 Magnitud de los valores singulares de Hankel	95
Figura 4.5 Comparación de la respuesta escalón y el modelo ERA	96
Figura 4.6 Respuesta en frecuencia del sistema de lazo abierto	97
Figura 4.7 Respuesta escalón unitario del sistema de lazo cerrado	98
Figura 4.8 Respuesta al escalón unitario del sistema compensado con PI análogo	99
Figura 4.9 Respuesta al escalón unitario del sistema compensado con PID análogo	100
Figura 4.10 Respuesta en frecuencia de lazo abierto con PI análogo	101
Figura 4.11 Respuesta en frecuencia de lazo abierto con PID análogo	102
Figura 4.12 Diagrama de bloques de un sistema en lazo cerrado	92
Figura 4.13 Respuesta en frecuencia de lazo abierto con PI digital	103

Figura 4.14 Respuesta en frecuencia de lazo abierto con PID digital

104

# CAPITULO I

## INTRODUCCION

### 1.1 MOTIVACION.

Los retos de la industria salvadoreña, que es el pilar de la economía, son grandes hacia el nuevo siglo. Todo va encaminado a obtener la mejor calidad posible en los productos que tendrán que competir en un mercado global.

Cuando se habla de calidad entran en juego los sistemas de control automático. Son estos sistemas los que permiten que variables como presión, temperatura, caudal, posición, velocidad, etc. sean controladas con exactitud para obtener los mejores resultados en los procesos industriales.

Comúnmente los controladores digitales industriales, tienen varias opciones para el ajuste de sus parámetros como son el método de Ziegler-Nichols y el de Autoajuste. En el método de Ziegler-Nichols ya existen fórmulas desarrolladas para el ajuste de controladores. Con el método de Autoajuste, el controlador en base a una medición de la salida y la referencia automáticamente selecciona los parámetros del controlador. Los dos métodos son bastante buenos y proporcionan resultados con relativa exactitud para los estándares actuales, pero no dejan de ser métodos empíricos de prueba y error que no son muy claros en su concepción. Estos métodos son los que se usan con más frecuencia en la

industria salvadoreña (según encuesta realizada) en empresas tales como: Embotelladora Salvadoreña, Destilería Salvadoreña, Nestlé de el Salvador y La Cooperativa Ganadera de Sonsonate ( La Salud ). Por lo antes expuesto es necesaria una alternativa innovadora a estos métodos.

## **1.2 OBJETIVOS.**

El objetivo principal de este proyecto es el de modelar sistemas industriales reales tales que, los modelos permitan un diseño de controladores digitales y aplicación de técnicas de control para mejorar las de la industria salvadoreña.

Para este objetivo, las siguientes actividades serán realizadas:

1. Diseño del circuito adquisidor de datos multifrecuencia.
2. Diseño del programa de modelado matemático.
3. Modelado de sistemas industriales reales.
4. Diseño de controladores digitales de los modelos obtenidos y simulación en computadora.

En el numeral 1 se diseñara el circuito adquisidor de datos. Este será capaz de muestrear sistemas dinámicos que posean constantes de tiempo desde 1 mseg hasta 10 seg.

En el numeral 2, se desarrollará el programa que permitirá modelar los sistemas dinámicos. El programa se basará en el algoritmo ERA (Eingensystem Realization Algorithm) del cual obtendremos una función de transferencia en el dominio digital.

En el numeral 3, se modelarán sistemas industriales típicos con variables como: temperatura, nivel, flujo, etc, con el objeto de diseñar un banco de modelos industriales con su respectiva descripción.

En el numeral 4, se harán simulaciones de los sistemas a controlar, realizando análisis tales como: respuesta en el tiempo y respuesta en frecuencia.

El uso de estas herramientas servirá para que tanto profesionales y estudiantes en el ramo de la ingeniería de control, conozcan como implementar un sistema de control de manera no empírica, tal como lo demandan las habilidades de un Ingeniero, especialmente en un futuro cercano.

### **2.3 VENTAJAS.**

Las ventajas y beneficios que traerá a la industria como a la enseñanza del control automático el desarrollo de este proyecto son:

1. Mayor exactitud en el diseño de controladores digitales. Con el modelado matemático, se define un juego de ecuaciones que representan la dinámica

del sistema con exactitud, o al menos, razonablemente bien, por tal razón se espera una mayor exactitud al diseñar los controladores.

2. Capacidad de conectar el dispositivo a diseñar a sensores normalizados. Ya que la mayoría de sensores industriales proporcionan una señal de salida de 4-20 mA, se construirá un circuito de acondicionamiento de señal para poder conectar al convertidor análogo-digital.
3. Lo innovador del uso del modelado matemático como también de las técnicas de control digital. A nivel de proyectos universitarios, es la primera vez que se utilizan estas herramientas.
4. Este proyecto podrá ser utilizado para propósitos académicos. Se podrá contar con un banco de modelos matemáticos, para poder ser analizados posteriormente

Cabe mencionar que el algoritmo de regulación, así como las simulaciones se harán en el programa MATLAB™, el cual es una herramienta muy poderosa para el análisis y diseño de sistemas.

## **CAPITULO II**

### **DISEÑO DEL CIRCUITO ADQUISIDOR DE DATOS.**

#### **2.1 INTRODUCCION**

En este capítulo se mostrará el diseño del circuito adquisidor de datos. Este diseño provee circuitos para interfazar la IBM PC a una diversidad de sensores. El circuito podrá ser conectado al conector del sistema de bus IBM PC.

#### **2.2 DIAGRAMA DE BLOQUES DEL CIRCUITO ADQUISIDOR DE DATOS.**

En la Figura 2.1 se observa el diagrama de bloques del circuito adquisidor de datos a diseñar, el cual nos muestra la operación general del circuito.

- Las líneas de direcciones, datos y control se toman del bus IBM PC que se explica en la siguiente sección.
- Las líneas de direcciones y control son amplificadas por medio de buffers de tres estados para que puedan ser manejadas por los distintos dispositivos.
- El comparador de 8 bits utiliza el bus de direcciones y una palabra de control cargada para habilitar al decodificador de 4-16 líneas el cual a la vez, se encarga de habilitar a los restantes dispositivos, dependiendo de la dirección de puerto de salida seleccionada.

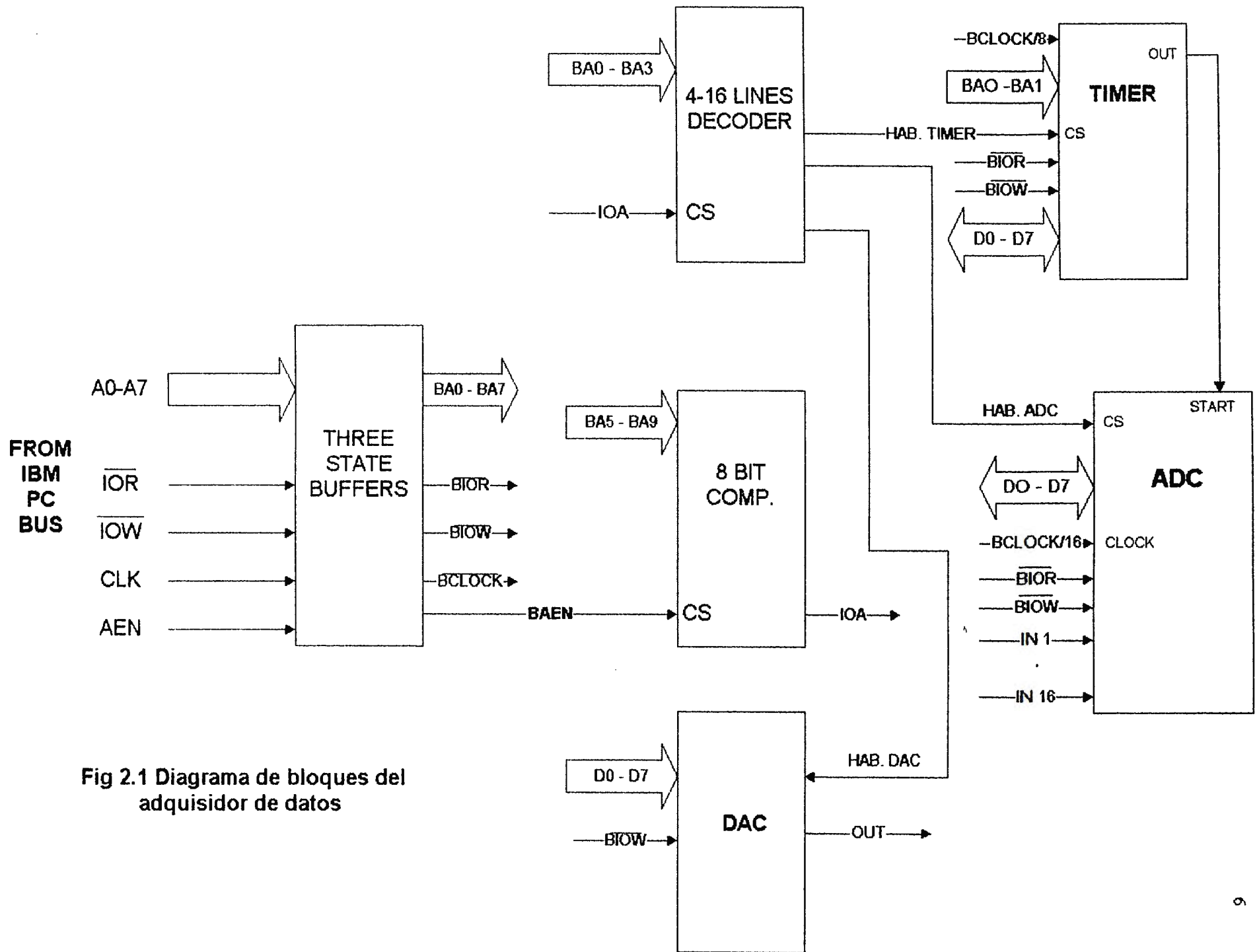


Fig 2.1 Diagrama de bloques del adquirente de datos



- El circuito timer, genera los pulsos necesarios para la entrada START del convertidor análogo-digital, por lo que, este circuito determina la frecuencia de muestreo a utilizar.
- El convertidor análogo-digital consta de 16 entradas analógicas lo cual lo hace muy versátil.
- El sistema también consta de un convertidor digital-análogo para poder generar una señal de salida en un momento dado.
- Nótese la división del reloj del sistema en los circuitos timer y convertidor análogo-digital, lo cual es ejecutado por un circuito divisor de frecuencia, con el fin de proporcionar la frecuencia específica de operación a cada circuito integrado.

## **2.3 SISTEMA DE BUS IBM PC.**

Este es la extensión del bus del microprocesador interno en la microcomputadora. Sin embargo, el bus es demultiplexado y ampliado para apoyar el acceso directo a memoria (DMA), interrupciones y otras características. Todas las señales son TTL. La Figura 2.2 muestra los 62 pines del sistema de bus IBM PC.

A continuación se explica en breve los pines del bus de expansión de una IBM PC:

- A0-A9. Este es el bus de direcciones del computador y son usados para direccionar memoria y dispositivos de entrada-salida dentro del sistema.

- **D0-D7.** Es el bus de direccionamiento de datos. Durante el ciclo de escritura del microprocesador, este proporciona datos en el bus de datos antes de que ocurra la transición positiva en las señales de IOW o MEMW. Durante el ciclo de lectura, el puerto de entrada o la memoria, coloca el dato en el bus de datos antes de que ocurra la transición positiva de las señales IOR o MEMR.

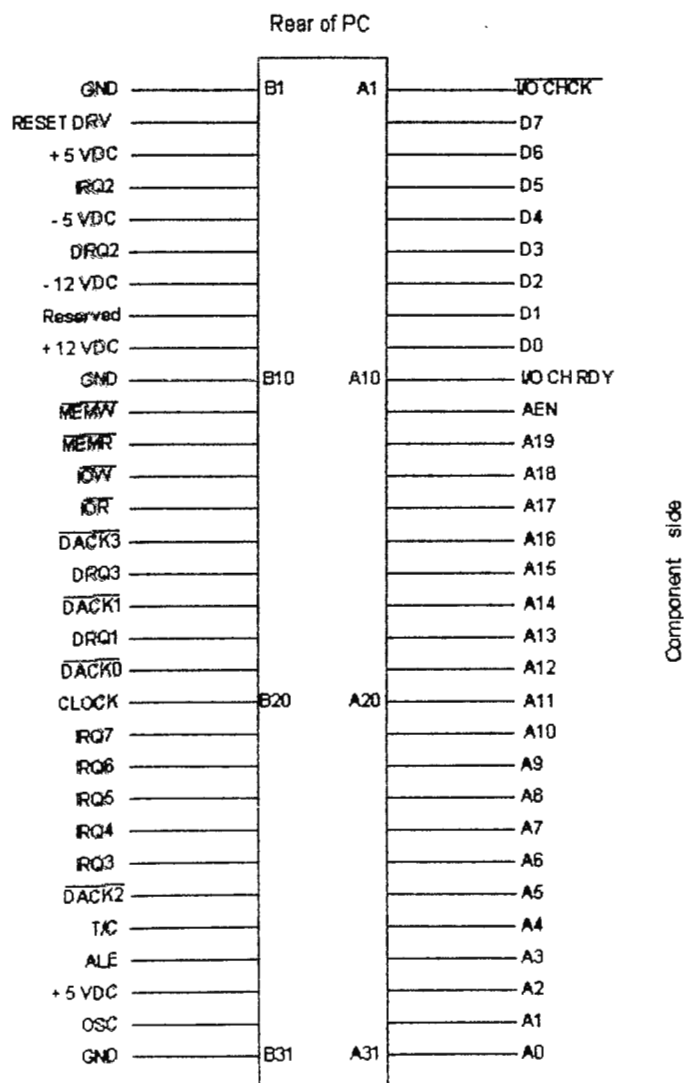


Fig. 2.2. Sistema de bus IBM

- **MEMR, MEMW, IOR, IOW.** Estas señales controlan las operaciones de lectura y escritura mencionadas arriba. Estas pueden ser generadas por el microprocesador o el controlador DMA.
  
- **ALE (Address Latch Enable).** Esta señal indica el comienzo de un ciclo de bus del microprocesador y es utilizada como indicador de direcciones válidas del DMA o del microprocesador.
  
- **AEN (Address Enable).** Esta señal es generada por el controlador DMA para indicar que un ciclo del DMA está en progreso. Es usado normalmente para deshabilitar la lógica de decodificación del puerto I/O durante un ciclo de DMA, de modo que las direcciones de memoria del DMA no sean usadas como direcciones del puerto de I/O.
  
- **OSC, CLOCK.** OSC es el reloj de alta velocidad del sistema con un período de 70 ns (14.31818 MHz) y 50% de ciclo de trabajo. CLOCK tiene una frecuencia de 8 MHz (esta es la frecuencia del reloj de la computadora usada en esta tesis).
  
- **I/O CH RDY.** Esta es una señal de entrada usada para generar estados de espera. Cuando esta señal está con un cero lógico, indica que el dispositivo periférico no está listo, permitiendo prolongar los ciclos de I/O. Es utilizada para manejar dispositivos periféricos lentos.

- *I/O CH CK*. Esta señal es usada para informar al procesador que hay un error de paridad en memoria o dispositivos de I/O.

- *RESET DRV*. Esta señal es usada para reinicializar la lógica del sistema para encender la máquina o en caso de cualquier bajo nivel de voltaje fuera de rango de operación. Esta señal es sincronizada con la transición negativa del oscilador.

## **2.4 DISEÑO DEL PUERTO PARALELO DE I/O.**

Las IBM PC aunque son muy eficientes para procesamiento de datos, necesitan una manera para comunicarse con el mundo exterior. El hardware de I/O y su correspondiente software proporcionan el interface para transferir los datos entre el computador y los dispositivos periféricos.

### **Conceptos básicos de I/O.**

Existen varias maneras para inicializar y controlar una transferencia de datos:

1. I/O controlado por programa
2. I/O controlado por interrupciones
3. I/O controlado por hardware (DMA)

Nos concentraremos únicamente en I/O controlado por programa, ya que rara vez las otras técnicas son usadas para interfaces con sensores.

En la Figura 2.3 se muestra un diagrama en bloques de la operación básica de I/O.

El decodificador genera un pulso de chip-select por medio del bus de direcciones y las señales de control IOR y IOW. El pulso chip-select para el puerto de entrada habilita los datos dentro del puerto desde el dispositivo de entrada. El pulso de chip-select para el puerto de salida captura los datos que proporciona el procesador al puerto. El puerto de entrada incluye buffers tri-estados para aislar los datos de entrada del bus excepto en el ciclo de lectura del bus. El puerto de salida utiliza latches para retener los datos hasta que los dispositivos de salida relativamente lentos acepten los bits de datos.

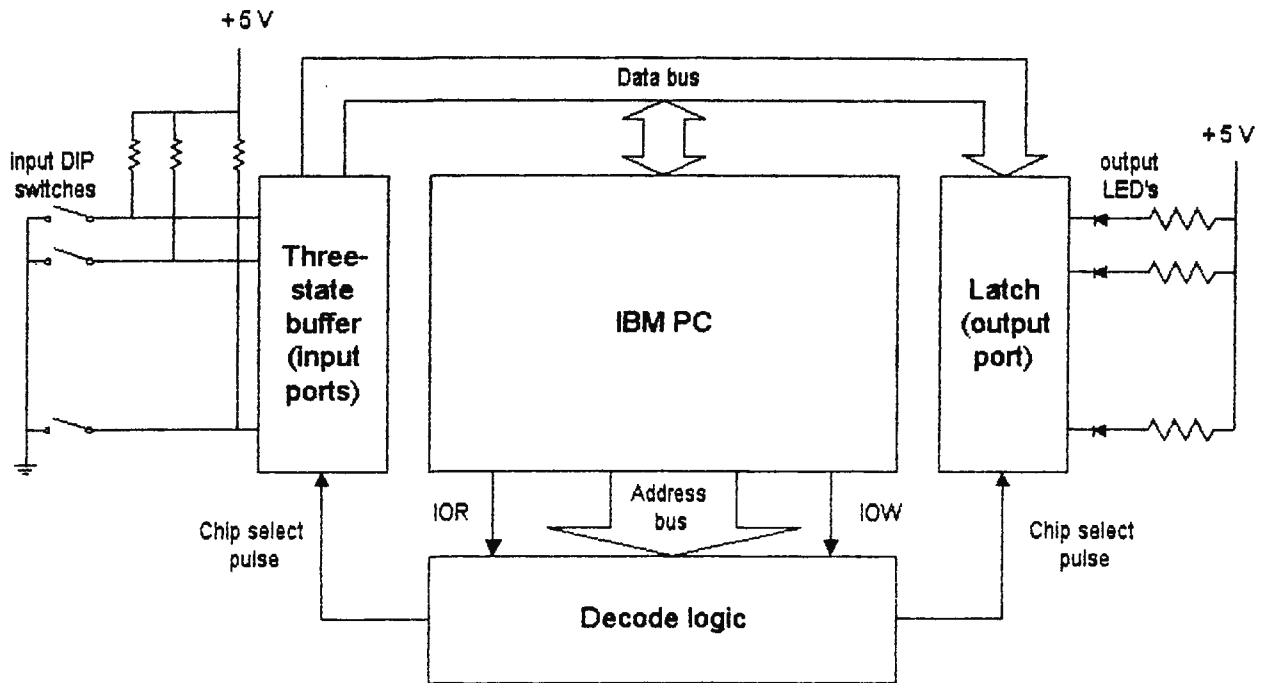


Fig. 2.3. Operación básica de I/O

### Transferencia I/O controlada por programa.

Los dos tipos de transferencia de I/O controlada por programa son *condicional e incondicional*. Una transferencia incondicional transporta datos desde un puerto de I/O sin

importar si el puerto está listo para recibir o transmitir el dato. Para esta transferencia, el procesador asume que los dispositivos de I/O están listos para recibir o transmitir datos. Errores en la transferencia de datos pueden ocurrir si al programar no se es cuidadoso. Si el procesador envía los datos mas rápido de lo que pueda recibir el puerto de salida, estos datos podrían perderse. También, un puerto de entrada podría proporcionar datos redundantes al procesador si este es accesado demasiado rápido.

Para evitar estos problemas, los puertos de I/O son configurados para transferencia de datos condicional usando handshaking. La Figura 2.4 muestra un ejemplo del uso del handshaking para una operación de I/O.

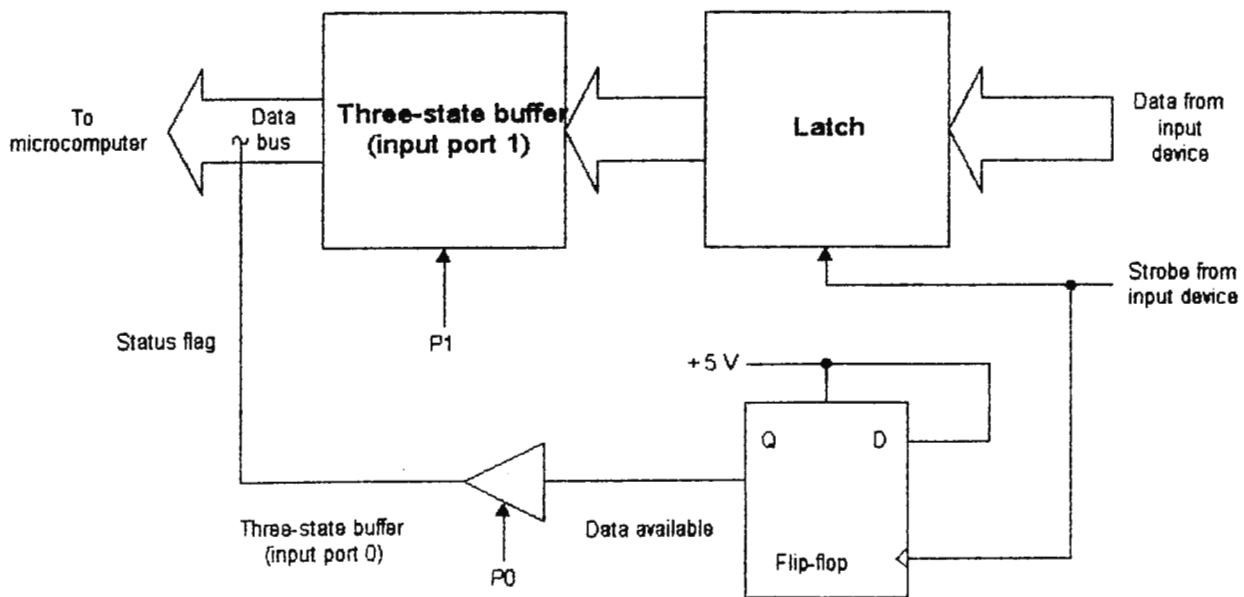


Fig. 2.4. Handshaking usando una bandera de dato disponible.

Un dispositivo de entrada tiene datos disponibles en el puerto de entrada 1 (P1) para transmitirlos a la computadora. Para indicar la disponibilidad de los datos, el dispositivo de

entrada coloca una bandera en el puerto de entrada 0 (P0). La computadora accesa periódicamente el estado de la bandera en P0 y la lee. Si ésta es "1", el dato esta disponible, y la computadora acepta el dato desde el P1. La señal chip-select para P1 habilita el dato desde el P1 y también limpia la bandera de disponibilidad del dato.

### Mapeo de I/O en la IBM PC.

Para generar correctamente las señales de chip-select, se necesita conocer el direccionamiento del puerto de I/O y la asignación de puertos. El diseño de la PC provee 10 bits para direccionamiento de los puertos, estos son los bits A0-A9, para un total de 1024 direcciones de puerto. Las primeras 512 direcciones desde 0000h hasta 01FFh son asignadas para el motherboard del sistema. Las restantes 512 direcciones, desde 0200h hasta 03FFh, están disponibles para tarjetas en cinco slots. La Tabla 2.1 muestra las direcciones de los puertos de I/O para una IBM PC. Los puertos utilizados para el adquisidor de datos comprenden desde 0300h hasta 030Ch.

Rango hexadecimal	Uso
000-00F	DMA chip 8237 A-5
020-021	Interrupción del 8259 A
040-043	timer 8253
060-063	PPI 8255 A
080-083	registro de la pagina DMA
0Ax	registro de la mascara NMI

0Cx	reservado
0Ex	reservado
100-1FF	sin uso
200-20F	control de juegos
210-217	unidad de expansión
220-24F	reservado
278-27F	reservado
2F0-2F7	reservado
2F8-2FF	comunicaciones asíncronas (2)
300-31F	tarjeta adquisidora
320-32F	disco fijo
378-37F	impresor
380-38C	comunicaciones SDLC
3B0-3BF	display/impresor monocromático IBM
3C0-3CF	reservado
3D0-3DF	color/gráficos
3E0-3F7	reservado
3F8-3FF	comunicaciones asíncronas (1)

Tabla. 2.1. Mapa de I/O en una IBM PC.



## Amplificación del bus y decodificación en el adquireedor de datos.

Las Figuras 2.5 y 2.6 ilustran el diseño que provee el interface al bus IBM PC para todos los otros circuitos. Estos circuitos amplifican el bus de direcciones, BA0 a BA9 y las señales de control BIOR, BLOW, BRESET DRV, BCLOCK. Nótese que la primera letra B indica que esta es una señal amplificada.

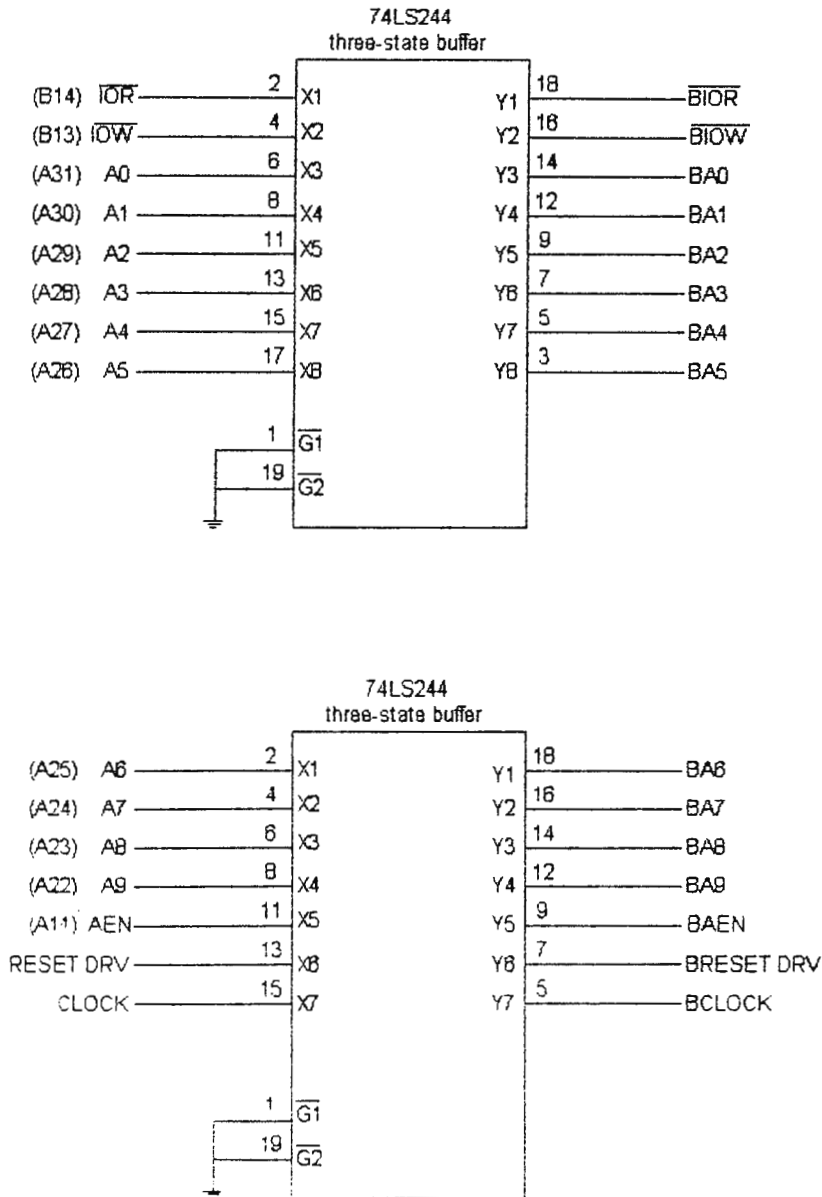


Fig. 2.5. Circuito amplificador .

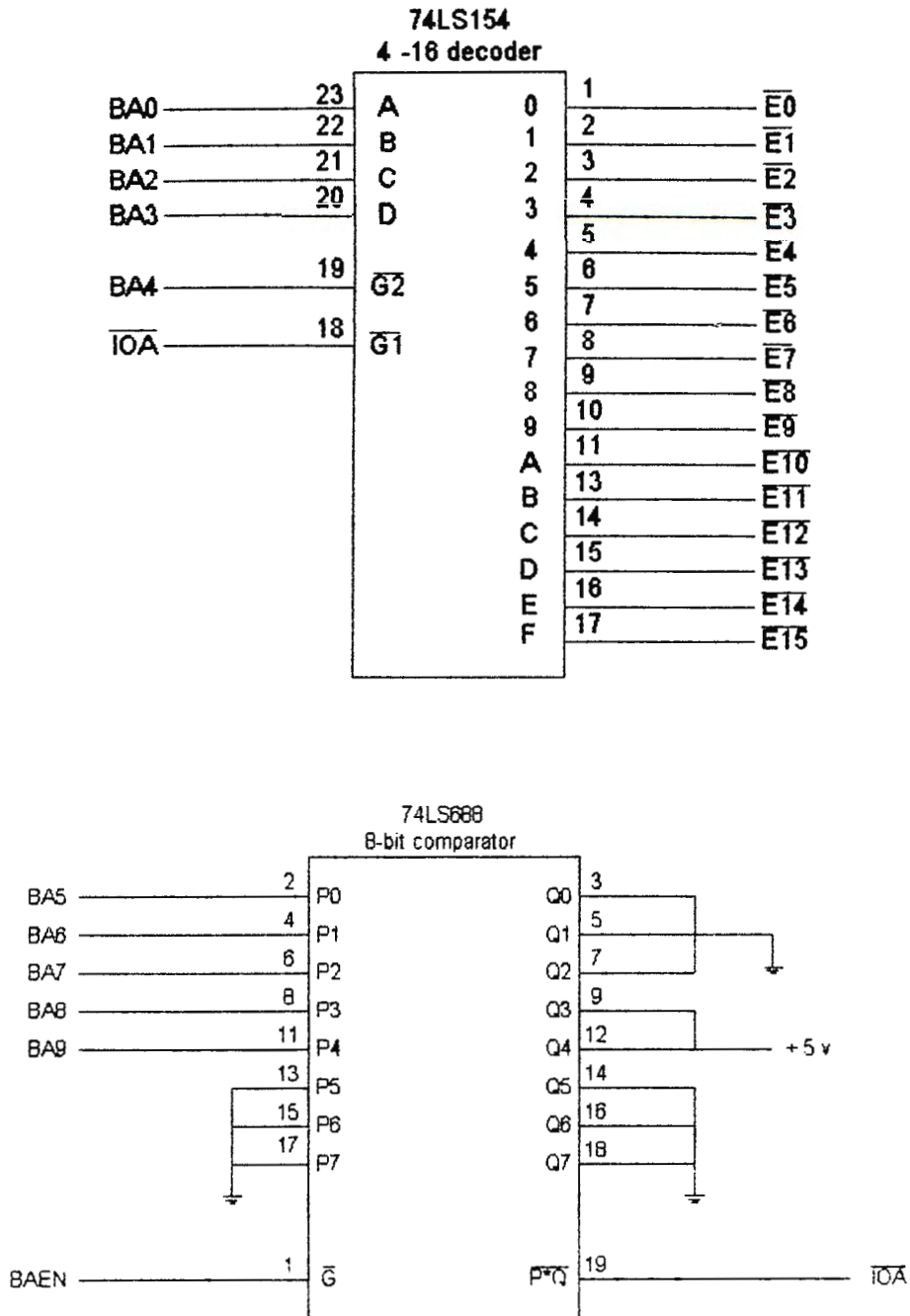


Fig. 2.6. Circuito decodificador.

El circuito decodificador genera 32 señales de selección de puerto, de E0 hasta E31 (0300h a 031Fh).

Dos buffers de tres estados (74LS244) son usados para amplificar los bytes A0-A9 del bus de direcciones y las señales de control requeridas. Para colocar las 32 direcciones del puerto en el espacio deseado desde 0300h hasta 03FFh, se usa un 74LS688 para comparar el valor colocado de 03xxh con los bytes de dirección (A5-A9) en el bus y generar una señal de IOA. La señal de IOA habilita el decodificador de 4 a 16 líneas (74LS154). Una señal de chip-select, habilitación de dirección (AEN) es usada para deshabilitar el comparador de modo que los puertos de I/O no puedan ser accesados durante el ciclo de DMA.

## **2.5 CIRCUITO DIVISOR DE FRECUENCIA.**

La Figura 2.7 muestra el circuito utilizado en el adquisidor de datos para dividir la frecuencia del reloj por 2, 4, 8 y 16. Hay que recordar que el reloj del sistema utilizado es de 8 MHz. El objetivo de la división de frecuencia es el de suministrar la frecuencia adecuada a los distintos dispositivos utilizados en el adquisidor de datos, por ejemplo: el convertidor análogo-digital necesita una frecuencia de 1 MHz y el circuito temporizador requiere una frecuencia de entrada de 2 MHz.

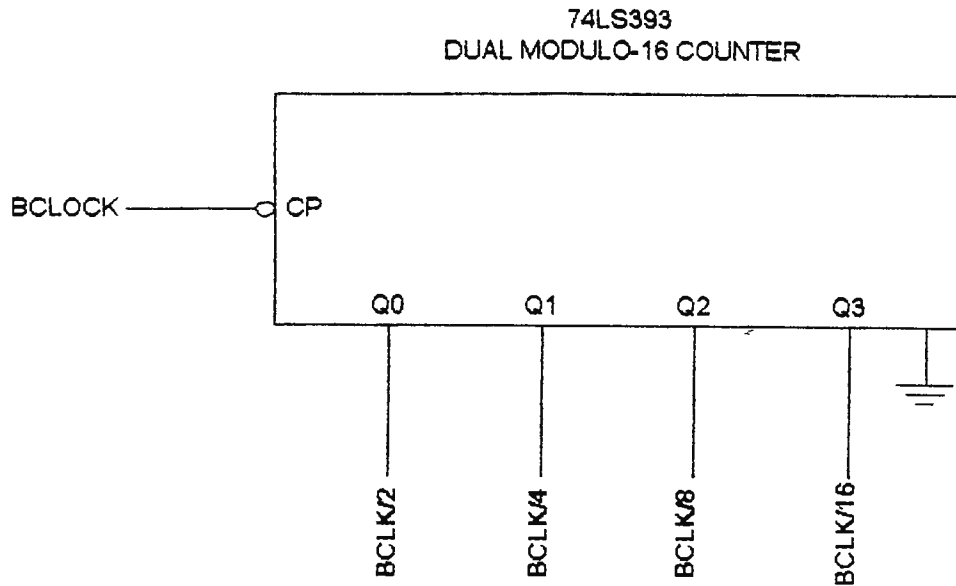


Fig. 2.7. Circuito divisor de frecuencias.

## 2.6 DISEÑO DEL CIRCUITO TEMPORIZADOR.

El circuito temporizador es necesario para realizar un muestreo multifrecuencia. Esto se logra proporcionando pulsos de frecuencia variable a la entrada START del convertidor análogo-digital (la función de la entrada START se explicará más adelante), utilizando un circuito temporizador programable integrado 8253 que se explica a continuación.

### Circuito Integrado 8253.

El temporizador programable de intervalos 8253, consta de tres contadores programables de 16 bits. Cada contador puede contar en binario o en decimal codificado en binario (BCD). Así mismo, cada contador consta de un reloj, una salida, y una línea de habilitación (GATE). Hay seis modos diferentes de operación:

**-Modo 0.** Permite utilizar el contador 8253 como contador de eventos. En este modo, la salida se vuelve un 0 lógico cuando se escribe la palabra de control y permanece ahí hasta que hay N más el número de conteo programado. Por ejemplo, si se programa un conteo de 5, la salida seguirá siendo 0 durante seis conteos a partir de N. Se debe tener en cuenta que la entrada de la compuerta (GATE) debe ser un 1 lógico a fin de permitir el conteo.

**-Modo 1.** El modo1 hace que el contador funcione como multivibrador monoestable. En este modo, la entrada GATE dispara el contador de modo que produzca un pulso en la salida OUT, que se convierte en 0 lógico mientras dura el conteo. Si el conteo es de 10, entonces la salida va a bajo durante 10 períodos de reloj cuando se le dispara. Si la entrada GATE ocurre mientras dura el pulso de salida, se vuelve a cargar el contador con el total de la cuenta y la conexión en la salida continúa mientras dure el conteo.

**-Modo 2.** Este modo permite que el contador genere una serie de pulsos continuos que tienen un ancho de un pulso de reloj. La separación entre los pulsos la determina el conteo. Por ejemplo, para un conteo de 10, la salida es un 1 lógico durante 9 períodos de reloj y está en un nivel bajo durante un período de reloj. La entrada GATE debe ser un 1 lógico para que este modo genere una serie continua de impulsos.

**-Modo 3.** Este modo es similar al modo 2, pero una onda cuadrada es producida a la salida. Si la cuenta  $N$  es par, la salida es 1 lógico para  $N/2$  ciclos de reloj y es 0 lógico para  $N/2$  ciclos de reloj. Si la cuenta es impar, la salida es 1 lógico para  $(N+1)/2$  cuentas y 0 lógico para  $(N-1)/2$  cuentas. La función de la GATE es la misma que en el modo 2.

**-Modo 4.** Este modo permite que el contador produzca un solo pulso en la salida. Si se programa un conteo de 10, la salida está en alto durante 10 períodos de reloj y luego, en bajo en uno. En este modo, igual que en los modos 2 y 3, se emplea la entrada GATE para habilitar el contador. La entrada GATE debe ser un 1 lógico para que el contador funcione en estos tres modos.

**-Modo 5.** La salida es la misma que en el modo 4 excepto que el conteo es habilitado por una transición positiva en la GATE.

Los relojes deben ser de cualquier frecuencia hasta 2 MHz. Las salidas pueden ser usadas para causar interrupciones, ser sondeadas o proporcionar reloj a otros dispositivos. En la Figura 2.8 se observa el pin-out del 8253.

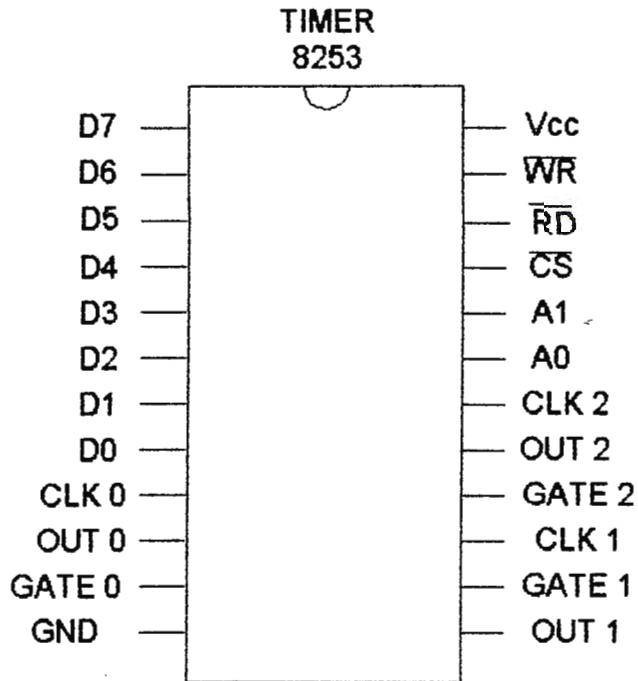


Fig. 2.8. Pin-out del 8253.

La Tabla 2.2 (a) muestra la asignación de bits de la palabra de control para inicializar un contador específico. Los dos bits más significativos (SC1, SC0) designan el contador a usar. La Tabla 2.2 (b) ilustra esto.

D7	D6	D5	D4	D3	D2	D1	D0
SC1	SC0	RL1	RL0	M2	M1	M0	BCD

Tabla 2.2 (a)

D7	D6	
SC1	SC0	
0	0	Selecciona contador 0
0	1	Selecciona contador 1
1	0	Selecciona contador 2
1	1	llegal

Tabla 2.2 (b)

D5	D4	
RL1	RL0	
0	0	Operación de retención del contador
0	1	Lee o carga el byte menos significativo
1	0	Lee o carga el byte mas significativo
1	1	Lee o carga el primer byte menos significativo

Tabla 2.2 (c)



D3	D2	D1	
M2	M1	M0	
0	0	0	Modo 0
0	0	1	Modo 1
X	1	0	Modo 2
X	1	1	Modo 3
1	0	0	Modo 4
1	0	1	Modo 5

Tabla 2.2 (d)

D0	
BCD	
0	Binario
1	BCD

Tabla 2.2 (e)

Tabla 2.2. Asignación de bits en el 8253 para la palabra de control

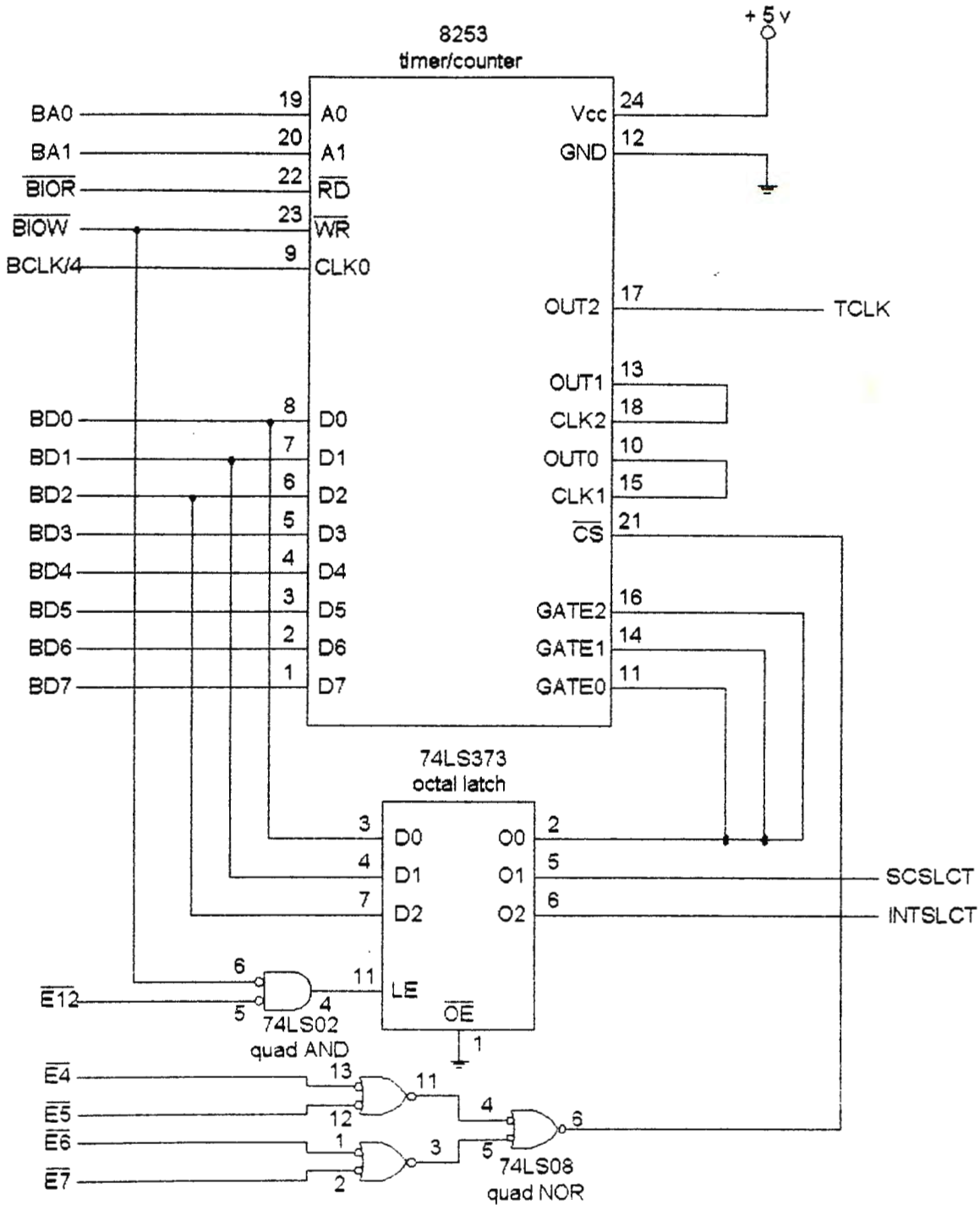
Los próximos dos bits (RL1, RL2) dicen al 8253 ya sea si el usuario quiere leer información de el, o cargar una cuenta dentro de el. Ya que cada contador consta de 16 bits, hay tres opciones para cargar la cuenta. El usuario puede cargar (1) únicamente el

byte menos significativo, (2) únicamente el byte más significativo, o (3) el byte menos significativo primero, y luego el byte más significativo. La Tabla 2.2 (c) muestra la asignación exacta.

Los próximos tres bits (M2, M1, M0) seleccionan uno de los seis modos de operación. La Tabla 2.2 (d) muestra la asignación de los bits y su significado. El bit menos significativo direcciona al timer a contar en binario de 16 bits o BCD de 4 décadas. La Tabla 2.2 (e) muestra ésta asignación.

En la Figura 2.9 se muestra el circuito esquemático que es implementado en el adquirente de datos. El voltaje de 5V es conectado entre Vcc y GND. Las líneas D0 a D7 se conectan al bus de datos. Las líneas RD y WR conectan a las señales BIOR y BIOW. Las líneas A0 a A1 conectan al bus de direcciones amplificado.

La Tabla 2.3 muestra como se puede direccionar la palabra de estados y los tres relojes. La línea chip-select (CS) debe ser activados por cuatro puertos separados ya que la palabra de control y los contadores son direccionados separadamente. Las líneas E4 (puerto 304h) hasta E7 (puerto 307h) accesan esos puertos. La línea E12 (puerto 30Ch) controlan las GATES del 8253 a través de un latch 74LS373 para permitir que la señal salga por el pin OUT2. El reloj del contador es 1 MHz que es un octavo de la frecuencia del reloj del sistema. Los tres contadores están conectados en cascada para proveer una posible cuenta de 48 bits, permitiendo la división de la frecuencia de entrada del reloj hasta  $2^{48}-1$ . Este arreglo permite un período máximo de reloj de 7 años.



**Timer/counter circuit**

Fig. 2.9. Circuito temporizador.

RD	WR	A1	A0	
1	0	0	0	Carga contador 0
1	0	0	1	Carga contador 1
1	0	1	0	Carga contador 2
1	0	1	1	Escribe palabra de Control
0	1	0	0	Lee contador 0
0	1	0	1	Lee contador 1
0	1	1	0	Lee contador 2

Tabla 2.3. Direccionamiento de contadores y palabra de control en el 8253.

Anteriormente, se menciona que la salida OUT2 puede ser conectada con software ya sea a una interrupción o a un convertidor análogo-digital. La Tabla 2.4 muestra los puertos asignados para el 8253. Ya que se ha configurado este timer como un simple contador, todos los tres contadores en cascada del 8253 deben ser habilitados al mismo tiempo y tener cuentas cargadas dentro de ellos.

Direcciones del puerto (HEX)	
304	Contador 0
305	Contador 1
306	Contador 2
307	Registro de control
30C	puerto GATE (solamente bit 0 )

Tabla 2.4. Asignación de puertos en el 8253.

Ahora se realizará una función en lenguaje C que inicializa el 8253 para producir una secuencia de pulsos a un rango dado por parámetros de entrada especificados por el usuario. La función `timer( )` coloca el 8253 para marcar a una razón dada. Esta función `timer ( )` es pasar el número de períodos de tiempo entre marcas y una unidad de tiempo. Un valor de unidad igual a "s", "m" o "u" selecciona un periodo en segundos, milisegundos o microsegundos respectivamente. Los microsegundos son redondeados a múltiplos cerrados de 3.3543  $\mu$ s. Como ejemplo, el comando

`timer ( 5. 'm' ) ;`

hace que el contador produzca un pulso continuo con un periodo de 5 ms entre los pulsos. Para la propia inicialización de el registro de estado en la tarjeta adquisidora, se puede seleccionar entre usar los pulsos desde cada contador para el inicio de la conversión analogo-digital o para provocar interrupciones.

La función `timer ( )` habilita los tres contadores en el 8253 para operar en modo 2, produciendo un tren continuo de pulsos repetitivos. Es importante para realizar esto que los registros de los contadores deben ser cargados con una cuenta de por lo menos 2 para que produzcan una salida. El efecto de esto es dividir la cuenta de entrada por 2.

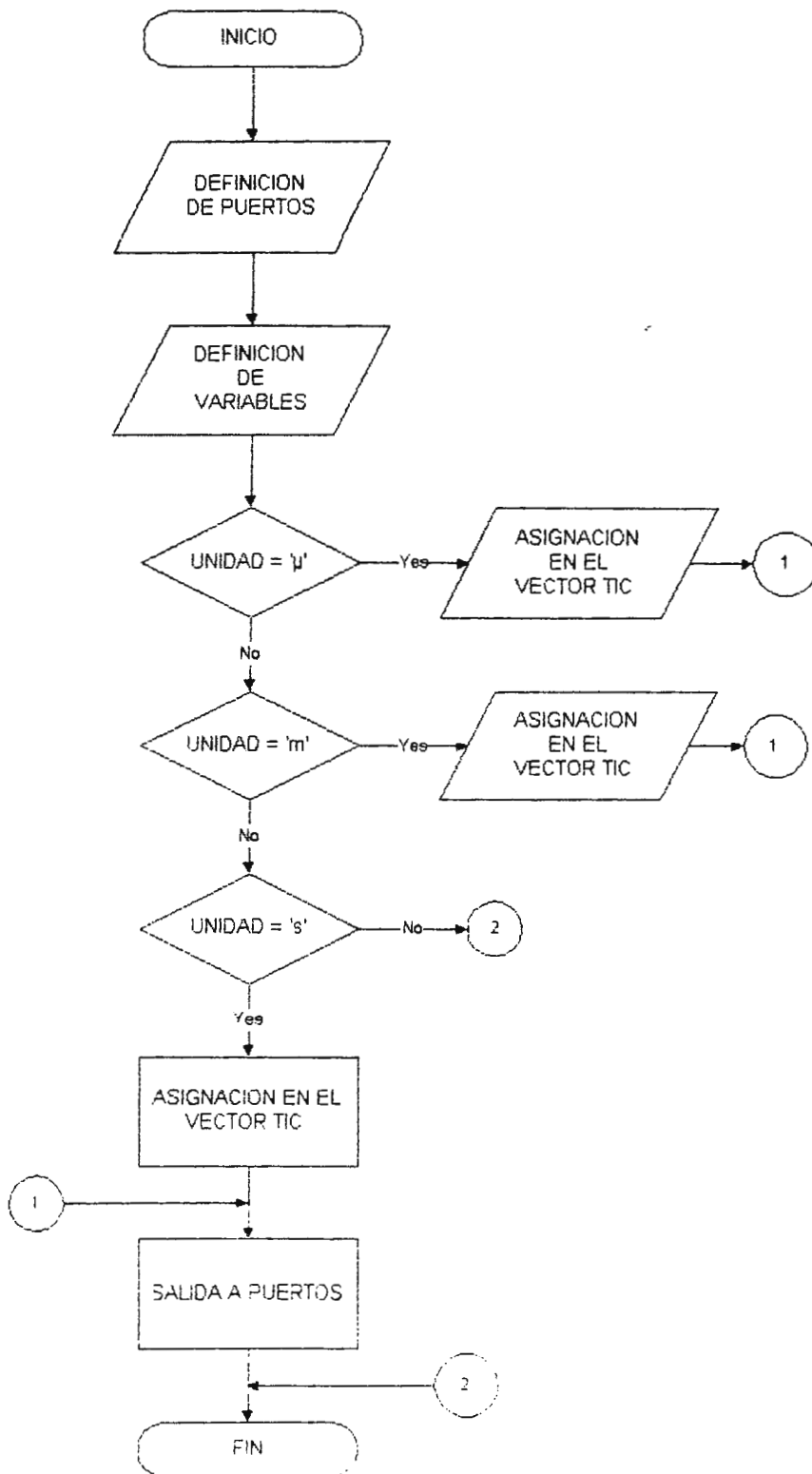


Figura 2.10 Diagrama de flujo de la función timer.

**Programa que inicializa los contadores en el 8253 y genera una secuencia de pulsos.**

```
#define CNT0 0x304      /* first counter register */
#define CNT1 0x305      /* second counter register */
#define CNT2 0x306      /* third counter register */
#define TCNTRL 0x307    /* timer control register */
```

**timer (period,unit)**

```
int period ;
char unit ;
{
char tlc[6] ;
unsigned int tottics ;
if (unit == "u")          /*microseconds */
{
tottics = (float) period / 3.3543 ;
tlc[5] = tottics / 256 ;
tlc[4] = tottics % 256 ;
tlc[3] = 0 ;
tlc[2] = 2 ;
tlc[1] = 0 ;
tlc[0] = 2 ;
}
```



```

if (unit == "m")           /*milliseconds */
    {
        tlc[5] = (periodx2) / 256 ;
        tlc[4] = (periodx2) % 256 ;
        tlc[3] = 1 ;
        tlc[2] = 42 ;
        tlc[1] = 0 ;
        tlc[0] = 2 ;
    }

if (unit == "s")           /* seconds */
    {
        tlc[5] = (periodx2) / 256 ;
        tlc[4] = (periodx2) % 256 ;
        tlc[3] = 0x17 ;
        tlc[2] = 0x4B ;
        tlc[1] = 0 ;
        tlc[0] = 100 ;
    }

outp (TCNTRL, 0x34) ;     /* counter 0, load LSB and MSB,
                               mode 2 */

outp (TCNTRL, 0x74) ;     /* counter 1, load LSB and MSB,
                               mode 2 */

outp (TCNTRL, 0xB4) ;     /* counter 2, load LSB and MSB,

```

**mode 2 \*/**

```
outp (CNT0, tlc[0]) ;  
outp (CNT0, tlc[1]) ;  
outp (CNT1, tlc[2]) ;  
outp (CNT1, tlc[3]) ;  
outp (CNT2, tlc[4]) ;  
outp (CNT2, tlc[5]) ;  
}
```

En general, la función **timer ( )** es diseñada para cargar los dos primeros contadores del 8253 para producir una entrada de reloj en el último contador que podría ser la mitad de la unidad de tiempo especificada. Esta unidad es usada para que una razón de un pulso por unidad de tiempo podría ser generada cargando el contador 2 con el valor 2. El número de unidades de tiempo por marca es multiplicado por 2 y cargado dentro del registro de reloj 2. Esto provoca una duración del pulso igual a la mitad de la unidad de tiempo que aparecerá en la salida CLK2 a la razón especificada. Por la forma en que está alambrado el 8253 en el adquisidor, el período mínimo que pueden ser producidos en la salida TCLK es 3.354  $\mu$ s. Esto es logrado por cargar cada uno de los tres registros de reloj con el número

## 2.7 DISEÑO DEL CIRCUITO CONVERTIDOR ANALOGO-DIGITAL.

En esta sección se explicará el diseño de un simple adquisidor de datos para una IBM PC. El sistema es diseñado alrededor de un ADC0817, un integrado muy popular de 8 bits. Se selecciono el ADC0817 por que es un chip de bajo costo y además tiene un multiplexor análogo (AMUX), cuya salida está disponible para poder conectar un circuito de muestreo y retención (S/H) el cual es conectado entre el AMUX y el ADC.

### Diseño del hardware.

Una aplicación común de un ADC es el de muestrear una señal a una razón constante y repetitiva. Esto requiere una señal de reloj para manejar la línea de inicio de conversión (START) del ADC. La Figura 2.11 muestra el circuito para obtener esta señal de reloj en el adquisidor de datos.

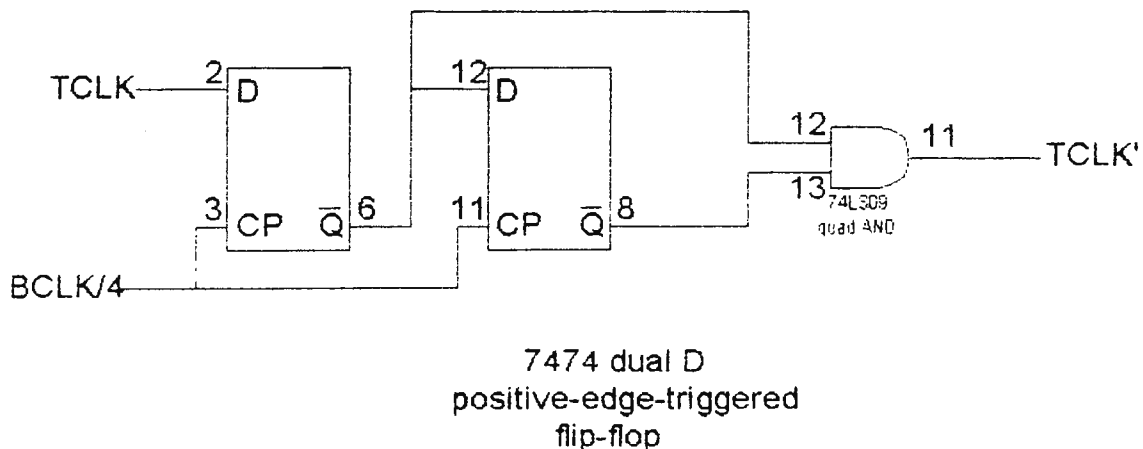


Fig. 2.11 Circuito para generar una señal de reloj conveniente a la entrada  
START del ADC.

La salida TCLK del timer 8253 no tiene un pulso bastante corto para manejar una señal típica de START o una línea de interrupción porque la señal original de reloj pasa a través de tres contadores dentro del chip, lo cual tiene el efecto de hacer una señal de 50% de ciclo de trabajo. El circuito de dos flip-flop simplemente hace un tren de pulsos de TCLK con pulsos de duración igual a la mitad de BCLK/4. De esta manera TCLK' tiene pulsos bastante angostos para controlar la línea START y la razón de muestreo puede ser programada por medio del circuito integrado 8253.

La Figura 2.12 muestra el diagrama del circuito ADC. Este es alambrado al sistema de bus y a los circuitos decodificadores de I/O descritos anteriormente. Al sistema pueden conectarse hasta 16 señales de entrada con voltaje de 0 a 5 voltios. El LM329B, es un zener de referencia amplificado por el LM301 (OP-AMP), el cual genera una referencia muy precisa de 5 V.

Para operar este convertidor, se coloca el número de canal en los 4 LSBs de un byte de datos y se emite un comando de salida para enviar el byte al puerto 308h, el cual mantiene seleccionada las líneas E8 y B10W. Esta selección de canal se produce a causa de que los 4 bits son retenidos dentro del registro de dirección de canal del ADC. La línea SCSLCT (Start Conversion seLeCT) retenida por medio del puerto 30Ch, controla si la conversión es inicializada por TCLK' o por el puerto 30Ah ( línea E10 ). Si SCSLCT es 1 lógico, un comando de entrada o salida con cualquier dato que active la selección del puerto

30Ah ( correspondiente a la línea E10 ) inicia la conversión, o si SCSLCT es 0 lógico, la señal TCLK' inicia la conversión.

El LF398 es un circuito de S/H, el cual localiza el dato en la entrada hasta que la línea EOC ( End Of Conversion ) del ADC sea 1 lógico. Después el ADC inicia el proceso de conversión, en este momento la línea EOC es 0 lógico. Esto provoca que el LF398 pare de localizar y retiene el dato que está en la entrada.

La línea INTSLCT ( INTerrupt seLeCT ) activada por el puerto 30Ch, controla si la línea IRQ2 es activada por EOC del ADC o por TCLK'. Una vez que el ADC a completado la conversión en un punto de muestreo, la línea EOC será 1 lógico. Si INTSLCT es 1 lógico, EOC causa una interrupción en IRQ2.

Si INTSLCT es 0 lógico, el procesador no es alertado que el fin de conversión ha ocurrido. Por esta razón, se implementa el circuito de la Figura 2.13 de modo que el microprocesador pueda leer si EOC es alto o bajo por medio del puerto 309h ( línea E9 ). Este proceso, donde el microprocesador es responsable de determinar si un evento ha ocurrido, es llamado sondeo. Este circuito también permite el sondeo de la señal TCLK para determinar si en un instante ésta es alta o baja.

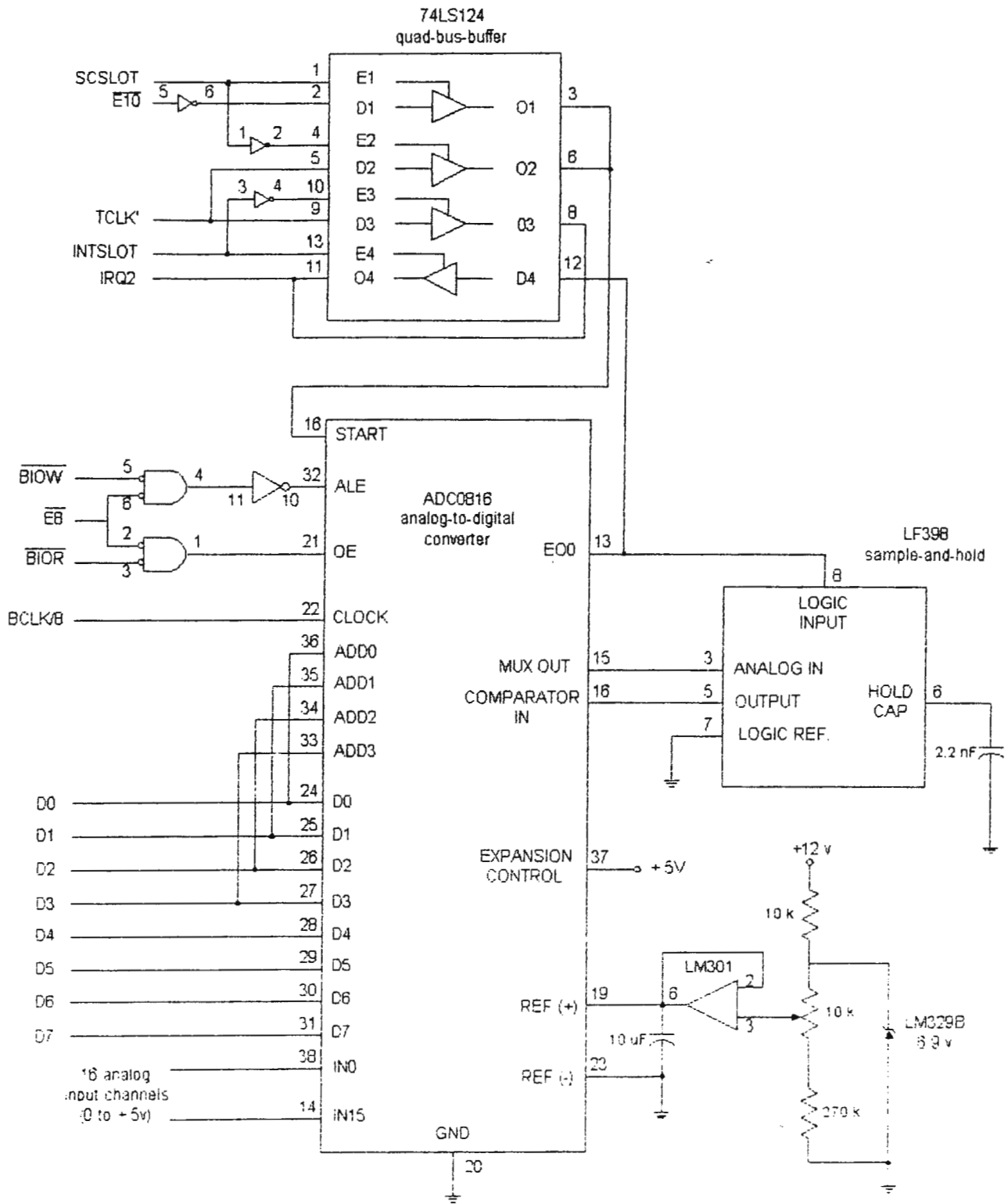


Fig. 2.12. Circuito convertidor análogo-digital.

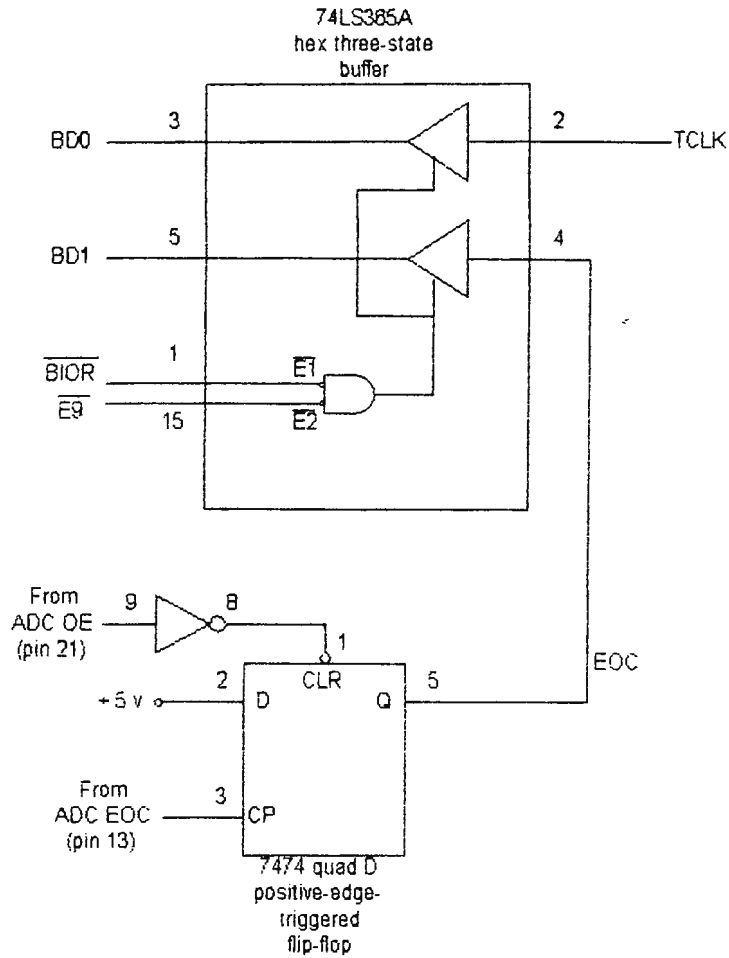


Fig. 2.13. Registro de estados para sondear las señales TCLK y EOC.

La Figura 2.14 muestra el diagrama del circuito de un convertidor digital-análogo (DAC) implementado en el adquisidor de datos. La línea de selección de puerto E11( puerto 30Bh ) controla la selección del DAC como un puerto de salida. Las líneas de control de salidas son configuradas para producir un rango de voltaje entre 0 y 2.5 Voltios.

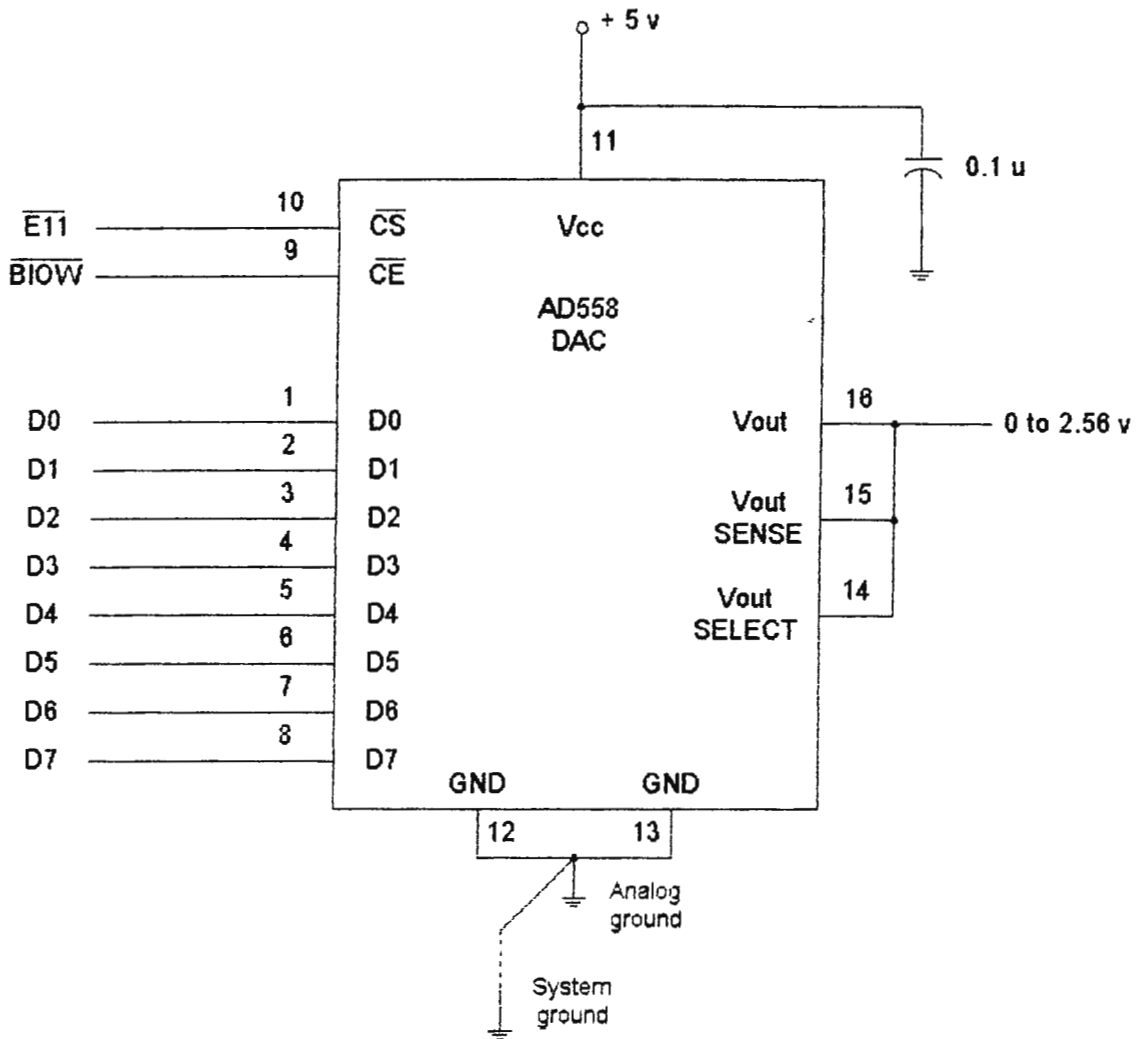


Fig. 2.14. Circuito convertidor digital-análogo.

### Diseño del software.

Ahora se muestra un programa en Lenguaje C con su respectivo diagrama de flujo, para muestrear datos desde el ADC bajo control de interrupción y enviando hacia el exterior los datos por medio del DAC. Notese que el intervalo de muestreo es de 5 ms.



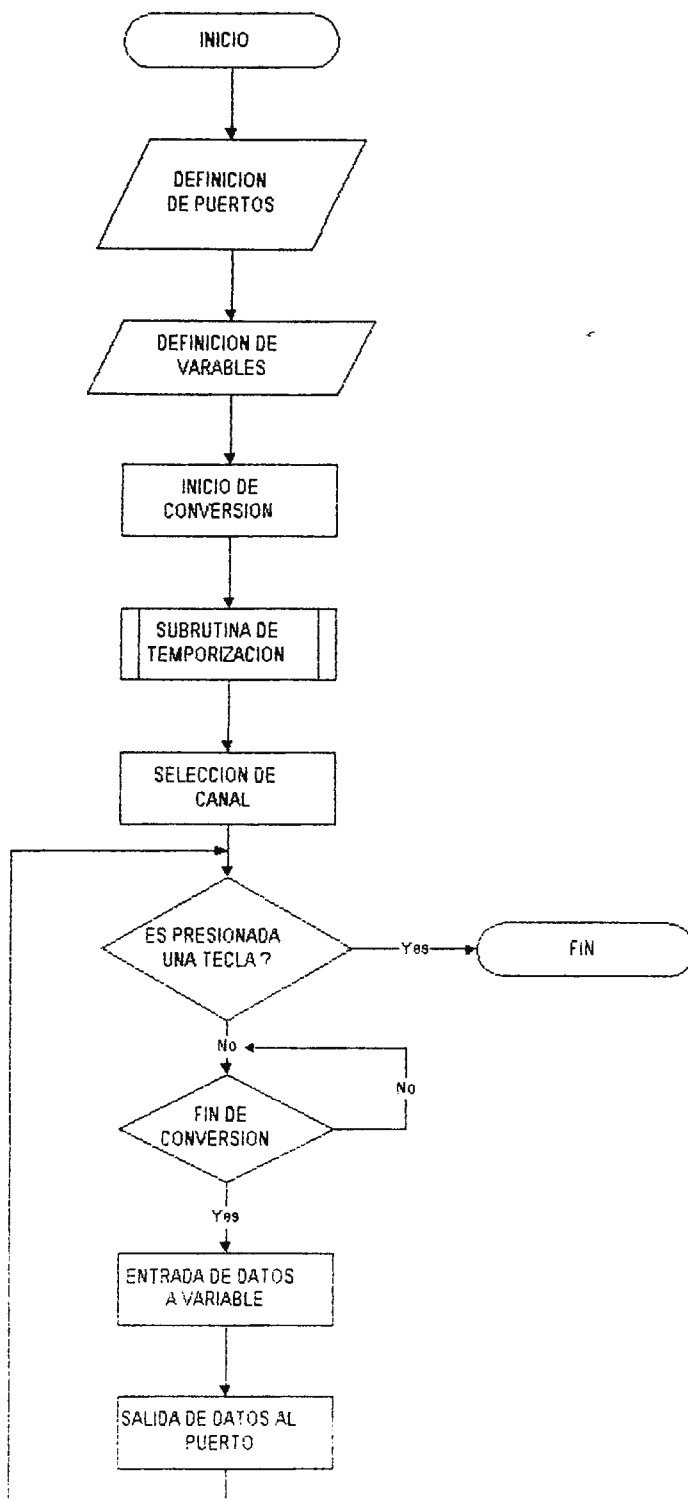


Figura 2.15. Diagrama de flujo del programa de muestreo.

## Programa para muestrear datos con el ADC y enviar los datos por el DAC

```

#define STAT 0x309    /* Protoboard status register */
#define CNTRL 0x30C  /* Protoboard control register */
#define ADC 0x308    /* A-to-D address and data. */
#define DAC 0x30B    /* D-to-A port address. */

main ()
{
    char datum ;

    outp ( CNTRL, 5 ) ;    /* Start conversión */
    timer ( 5, 'm' ) ;    /* Sample every 5ms. */
    outp ( ADC, 1 ) ;     /* Select channel 1 */

    while ( ! kbhit () )    /* Continue execution until a key is
                               hit */
        if ( Inp ( STAT ) & 2 )    /* Test for end-of-conversión */
            {
                datum = Inp ( ADC ) ;    /* Sample data */
                outp ( DAC, datum ) ;    /* Output sample data */
            }
}

```

## CAPITULO III

# USO DEL ALGORITMO ERA (EINGENSYSTEM REALIZATION ALGORITHM) PARA EL MODELADO DE SISTEMAS.

### 3.1 INTRODUCCION

En éste capítulo se estudiará el algoritmo ERA para el modelado de sistemas, con el cual se obtiene una función de transferencia de un sistema dado. Antes de revisar el algoritmo, es necesario revisar algunos conceptos de la teoría de sistemas lineales que son muy importantes para el entendimiento de éste algoritmo.

### 3.2 DEFINICION DE FUNCION DE TRANSFERENCIA [1].

La función de transferencia,  $H(s)$ , para un sistema lineal invariante en el tiempo de una entrada y una salida, es la razón de la transformada de Laplace de la salida,  $Y(s)$ , y la transformada de Laplace de la entrada,  $R(s)$ . Ya que se está hablando de sistemas generales se usa la notación  $r(t)$  para la entrada y  $y(t)$  para la salida. Como la transformada de Laplace de un impulso es la unidad, la función de transferencia es la transformada de Laplace de la respuesta impulso del sistema. La Figura 3.1 ilustra un sistema general e introduce la notación descrita anteriormente.

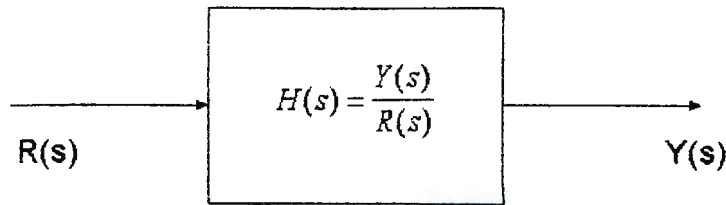


Figura 3.1 Definición de función de transferencia

La salida y la entrada para un sistema lineal invariable en el tiempo están relacionadas por la siguiente ecuación diferencial:

$$\frac{d_n y}{dt^n} + a_{n-1} \frac{d_{n-1} y}{dt^{n-1}} + \dots + a_1 \frac{dy}{dt} + a_0 y = \frac{d_m r}{dt^m} + b_{m-1} \frac{d_{m-1} r}{dt^{m-1}} + \dots + b_0 r \quad (3.1)$$

La correspondiente ecuación de transferencia es entonces dada por la ecuación 3.2.

$$H(s) = \frac{b_m s^m + b_{m-1} s^{m-1} + \dots + b_1 s + b_0}{s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0} \quad (3.2)$$

La función de transferencia puede ser obtenida de la respuesta impulso, como es mostrado en la Figura 3.2. Si una función impulso es aplicada a la entrada de un sistema, la transformada de Laplace de la salida es igual a la función de transferencia del sistema. La respuesta al impulso del sistema, es por lo tanto la transformada inversa de Laplace de la función de transferencia.

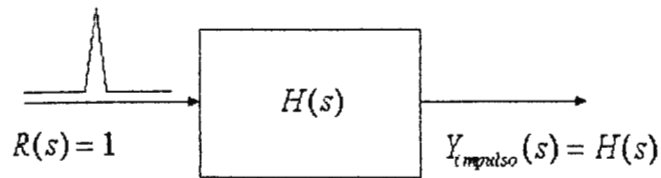


Figura 3.2 Respuesta impulso.

### 3.3 RESPUESTA AL PULSO DISCRETO [2].

Si un sistema es lineal e invariante en el tiempo, entonces su respuesta a una entrada de pulso unitario de duración  $T$  es siempre de la misma forma, pero desplazada en el tiempo sin importar el tiempo inicial de la aplicación del pulso. Esto es ilustrado en la Figura 3.3.

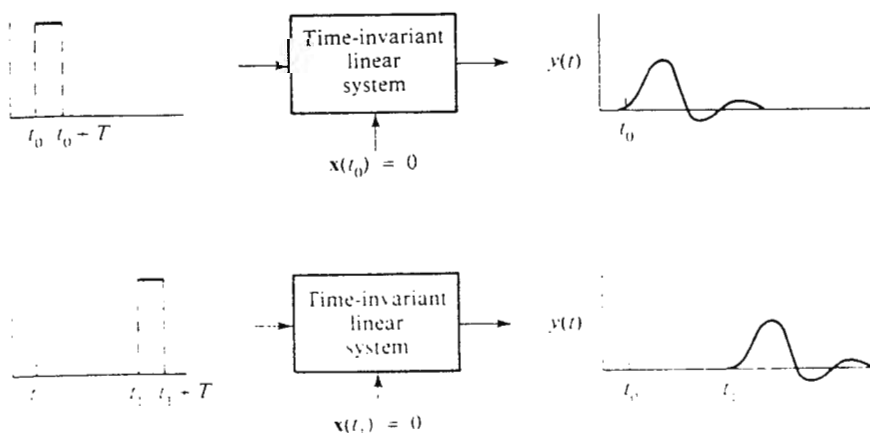


Figura 3.3. Respuesta al pulso de un sistema lineal invariante en el tiempo.

Si se aplica un pulso de amplitud unitario de duración  $T$  a un sistema lineal invariante en el tiempo y muestreando la salida cada  $T$  unidades de tiempo, obtendremos la salida:

$$h_T(k) \equiv y_{zS}(kT) \quad k = 1, 2, 3, \dots \quad (3.3)$$

La secuencia  $h_T(k)$  es llamada la *respuesta al pulso discreto* del sistema. Esto es descrito en la Figura 3.4.

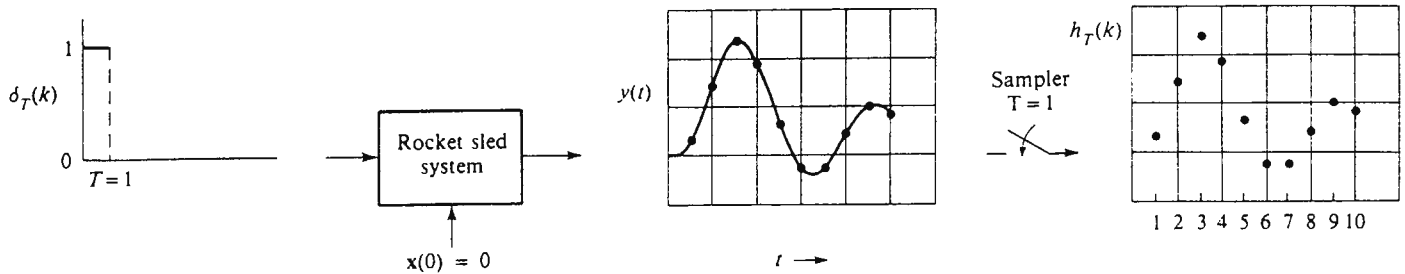


Figura 3.4. Generación de la respuesta discreta al pulso

Supóngase que la secuencia de pulsos de entrada como es mostrado en la Figura 3.5 son aplicados al sistema lineal descrito en la Figura 3.4, el cual tiene condiciones iniciales 0. Se denotan los valores de los pulsos de entrada como:

$$u_T(k) \equiv u(t), \quad t \in (kT, (k+1)T) \quad k = 0, 1, 2, \dots \quad (3.4)$$

Ahora, ya que la respuesta de salida a un simple pulso de amplitud unitario es conocida, el principio de superposición lineal puede ser aplicado para determinar la respuesta de salida a la secuencia entera de pulsos de entrada. En particular, la salida al tiempo  $t = T = 1$  es únicamente el resultado del primer pulso. Esto es:

$$y_{zs}(T) = h_T(1) * u_T(0) \quad (3.5)$$

Sin embargo, la salida al tiempo  $t = 2T = 2$  es el resultado del primer pulso y el segundo pulso. Esto es:

$$y_{zs}(2T) = h_T(2) * u_T(0) + h_T(1) * u_T(1) \quad (3.6)$$

El primer término en la ecuación (3.6) es la salida, al tiempo  $t = 2T$ , causada por el primer pulso, y el segundo término es la salida, al tiempo  $t = 2T$ , causada por el segundo pulso. Por el principio de superposición lineal, la respuesta total es la suma de las respuestas causadas por ambos pulsos.

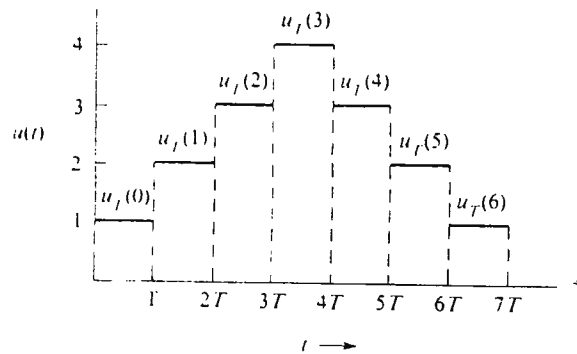


Figura 3.5. Secuencia de pulsos de entrada discretos.

La salida muestreada en  $t = kT$  es entonces la suma de las respuestas causadas por todas las entradas previas:

$$y_{zs}(kT) = [h_T(k) * u_T(0)] + [h_T(k-1) * u_T(1)] + \dots \\ + [h_T(2) * u_T(k-2)] + [h_T(1) * u_T(k-1)] \quad (3.7)$$

La ecuación (3.7) puede escribirse en notación compacta usando la *sumatoria de convolución*:

$$y_{zs}(kT) = \sum_{i=0}^{k-1} h_T(k-i) * u_T(i) \quad (3.8)$$



### 3.4 RELACION ENTRE LA RESPUESTA AL PULSO DISCRETO Y LA RESPUESTA AL IMPULSO CONTINUO [2].

La secuencia de la respuesta al pulso discreto  $h_T(k)$ ,  $k = 1, 2, 3, \dots$ , es definida como la respuesta muestreada de un sistema lineal invariante en el tiempo forzada por una entrada pulso unitario:

$$\delta_T(t) \equiv \begin{cases} 0 & t < 0 \\ 1 & 0 < t < T \\ 0 & t > T \end{cases} \quad (3.9)$$

La respuesta de salida para todo tiempo  $t$  puede ser encontrada por medio de la siguiente integral de convolución .

$$y(t) = \int_{-\infty}^t h(t - \tau) \delta_T(\tau) d\tau \quad (3.10)$$

Para una entrada  $t \geq T$  la ecuación (3.9) puede ser sustituida en la ecuación (3.10):

$$y(t) = \int_0^T h(t - \tau) d\tau \quad (3.11)$$

Ya que  $h_T(k)$  es la respuesta muestreada a una entrada pulso unitario, la ecuación (3.11) puede ser usada para relacionar  $h_T(k)$  y  $h(t)$ :

$$h_T(k) = \int_0^T h(kT - \tau) d\tau \quad (3.12)$$

### 3.5 RESPUESTA AL ESCALON UNITARIO [2].

Hay ciertas entradas que son básicas en el análisis de respuestas de sistemas lineales ya que éstas revelan el comportamiento del sistema. Por el principio de superposición lineal, éste comportamiento podría generalizarse a cualquier otra entrada. (Una de éstas se tiene encontrando la entrada impulso  $\delta(t)$ , produciendo una muy importante función de respuesta al impulso  $h(t)$ . Otra de éstas entradas especiales es la *función escalón unitario* :

$$u_{-1}(t) \equiv \begin{cases} 0 & t < 0 \\ 1 & t > 0 \end{cases} \quad (3.13)$$

Como la función impulso unitario. la discontinuidad de la función escalón unitario ocurre al tiempo cuando el argumento va a 0. Por ejemplo.

$$u_{-1}(t-7) = \begin{cases} 0 & t < 7 \\ 1 & t > 7 \end{cases} \quad (3.14)$$

La función escalón unitario podría relacionarse también con la función impulso unitario por medio de la relación

$$u_{-1}(t-7) = \int_{-\infty}^{\infty} \delta(\tau-7) d\tau \quad (3.15)$$

Esto es, la integral de  $\delta(\tau-7)$  es la unidad para todo  $t > 7$ .

Por la relación integral de la ecuación (3.15), la cual es matemáticamente correcta, el impulso unitario es referido como *la derivada del escalón unitario*.

Aplicando un escalón unitario en  $t=0$  a un sistema lineal invariante en el tiempo nos da una respuesta al escaló unitario, la cual es :

$$r(t) = \int_{-\infty}^{\infty} h(t-\tau)u_{-1}(\tau)d\tau \quad (3.16)$$

Ya que  $u_{-1}(t)$  es cero para  $t < 0$  y la unidad para  $t > 0$ . esto puede escribirse como:

$$r(t) = \int_0^t h(t - \tau) d\tau \quad t > 0 \quad (3.17)$$

Muestreando la respuesta escalón unitario cada  $T$  unidades de tiempo, una relación también puede ser encontrada para la secuencia de la respuesta al pulso discreto,  $h_T(k)$ ,  $k = 1, 2, 3, \dots$

$$r(kT) = \int_0^{kT} h(kT - \tau) d\tau \quad (3.18)$$

Aunque una entrada impulso ideal,  $\delta(t)$ , es imposible de realizar físicamente, igualmente la entrada pulso digital, es muy dificultoso de producir, particularmente si  $T$  es muy pequeño. Por otro lado, la entrada escalón unitario es típicamente una entrada que puede producirse con razonable fidelidad. Por ejemplo, en un sistema eléctrico, se puede producir una entrada escalón con solo cerrar un interruptor a una batería como fuente de entrada.

La respuesta al pulso discreto, puede encontrarse por medio de la ecuación:

$$h_T(k) = r(kT) - r(kT - T) \quad (3.19)$$

esta ecuación dice que el modelo de respuesta al pulso discreto, es simplemente la diferencia de las muestras al escalón unitario en el tiempo  $t = kT$  y el tiempo  $t = (k-1)T$ , siempre y cuando el tiempo de muestreo sea mucho menor del sistema.

### 3.6 DEFINICION DE LA ECUACION DE DIFERENCIAS FINITAS [2].

Como se menciona anteriormente, para un sistema SISO ( single input single output ) representado por la siguiente ecuación diferencial

$$\frac{d_n y}{dt^n} + a_{n-1} \frac{d_{n-1} y}{dt^{n-1}} + \dots + a_1 \frac{dy}{dt} + a_0 y = \frac{d_m r}{dt^m} + b_{m-1} \frac{d_{m-1} r}{dt^{m-1}} + \dots + b_0 r \quad (3.1)$$

tiene como función de transferencia

$$H(s) = \frac{b_m s^m + b_{m-1} s^{m-1} + \dots + b_1 s + b_0}{s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0} \quad (3.2)$$

La respuesta discreta al pulso es dada por la ecuación

$$h_T(k) = r(kT) - r(kT - T) \quad (3.19)$$

Otra alternativa para el modelo del pulso discreto es la ecuación discreta de diferencias finita:

$$\begin{aligned}
y_T(k+n) + a_{T_{n-1}} y_T(k+n-1) + a_{T_{n-2}} y_T(k+n-2) + \dots + a_{T_0} y_T(k) \\
= b_{T_{n-1}} u_T(k+n-1) + b_{T_{n-2}} u_T(k+n-2) + \dots + b_{T_0} u_T(k) \quad (3.20)
\end{aligned}$$

Los coeficientes  $2n \{a_{T_{n-1}}, \dots, a_{T_0}\}$  y  $\{b_{T_{n-1}}, \dots, b_{T_0}\}$  son constantes para un sistema lineal invariable en el tiempo, pero ellos son dependientes del período de muestreo  $T$ . Estos coeficientes tampoco son iguales a los coeficientes del modelo en tiempo continuo. El propósito de esta sección es derivar el modelo representado por la ecuación (3.20) para un modelo en tiempo continuo dado.

Apropiadamente computado, la representación de la ecuación (3.20) es exacta; ésta provee una salida muestreada exacta  $y(kT) = y_T(k)$  para una entrada muestreada  $u_T(k)$ . La ecuación (3.20) es una alternativa deseable para la simulación digital de sistemas continuos. Para la simulación digital es conveniente mover todas las variables por  $n$  unidades de tiempo y resolver para  $y_T(k)$ :

$$\begin{aligned}
y_T(k) = -a_{T_{n-1}} y_T(k-1) - a_{T_{n-2}} y_T(k-2) - \dots - a_{T_0} y_T(k-n) \\
+ b_{T_{n-1}} u_T(k-1) + b_{T_{n-2}} u_T(k-2) + \dots + b_{T_0} u_T(k-n) \quad (3.21)
\end{aligned}$$

La simulación de la ecuación (3.21) en computadora digital es bastante eficiente. Computando  $y(kT) = y_T(k)$  en cada tiempo  $k$  requiere salvar  $n$  valores pasados de la salida y  $n$  valores pasados de la entrada.

Hay muchas maneras de derivar una ecuación de diferencias aproximada, ecuación (3.21), para el modelo en tiempo continuo, ecuación (3.1), pero un método muy eficaz es derivado considerando la respuesta a una entrada pulso unitario:

$$\delta_T(t) \equiv \begin{cases} 1 & 0 < t \leq T \\ 0 & \text{otro caso} \end{cases} \quad (3.22)$$

Además, éste método es "exacto" a pesar del espaciamiento del muestreo,  $T$ .

Si la entrada pulso digital es aplicada al sistema, entonces  $u_T(0) = 1$  y  $u_T(k) = 0$ ,  $k > 0$ . También la salida a una entrada pulso unitario es la secuencia de la respuesta al pulso discreto. La secuencia de la respuesta al pulso discreto puede ser determinada por la ecuación (3.19) o por forzar al sistema físico con una entrada pulso-digital.

Usando la ecuación (3.21) y usando el hecho de que  $u_T(0) = 1$ ,  $u_T(k) = 0$  para  $k = 1, 2, \dots$ , y  $h_T(k) = y_T(k)$  obtenemos:

$$h_T(1) = b_{T_{n-1}} \quad (3.23)$$

$$h_T(2) = -a_{T_{n-1}} h_T(1) + b_{T_{n-2}} \quad (3.24)$$

$$h_T(3) = -a_{T_{n-1}} h_T(2) - a_{T_{n-2}} h_T(3) + b_{T_{n-3}} \quad (3.25)$$

$$\dots\dots\dots$$

$$h_T(n) = -a_{T_{n-1}} h_T(n-1) - \dots - a_{T_1} h_T(1) + b_{T_0} \quad (3.26)$$

$$h_T(n+1) = -a_{T_{n-1}} h_T(n) - \dots - a_{T_1} h_T(2) - a_{T_0} h_T(1) \quad (3.27)$$

$$\dots\dots\dots$$

$$h_T(2n) = -a_{T_{n-1}} h_T(2n-1) - \dots - a_{T_1} h_T(n+1) - a_{T_0} h_T(n) \quad (3.28)$$

Hay  $2n$  coeficientes desconocidos,  $\{a_{T_{n-1}}, \dots, a_{T_0}\}$  y  $\{b_{T_{n-1}}, \dots, b_{T_0}\}$ , y  $2n$  ecuaciones. Estas son representadas por las ecuaciones (3.23) a la (3.28). Las  $n$  ecuaciones representadas por (3.27) y (3.28) son colocadas convenientemente como un juego de ecuaciones lineales:

$$\begin{bmatrix} h_T(1) & h_T(2) & \dots & h_T(n) \\ h_T(2) & h_T(3) & \dots & h_T(n+1) \\ \vdots & \vdots & & \vdots \\ h_T(n) & h_T(n+1) & \dots & h_T(2n-1) \end{bmatrix} \begin{bmatrix} a_{T_0} \\ a_{T_1} \\ \vdots \\ a_{T_{n-1}} \end{bmatrix} = \begin{bmatrix} -h_T(n+1) \\ -h_T(n+2) \\ \vdots \\ -h_T(2n) \end{bmatrix} \quad (3.29)$$

La matriz con los coeficientes  $n \times n$  de la ecuación (3.29):



$$H_T \equiv \begin{bmatrix} h_T(1) & h_T(2) & \cdots & h_T(n) \\ h_T(2) & h_T(3) & \cdots & h_T(n+1) \\ \cdots & \cdots & \cdots & \cdots \\ h_T(n) & h_T(n+1) & \cdots & h_T(2n-1) \end{bmatrix} \quad (3.30)$$

es una matriz muy especial con propiedades teóricas de los sistemas muy especiales. Esta es llamada *matriz de Hankel* del sistema. Los elementos de dicha matriz se le conocen como *parámetros de Markov*

### 3.7 REPRESENTACION DE SISTEMAS DINAMICOS EN EL ESPACIO DE ESTADO [3].

Un sistema dinámico que consiste en un número finito de parámetros concentrados, se puede describir mediante ecuaciones diferenciales ordinarias en las que el tiempo es la variable independiente. Mediante notación vectorial-matricial, se puede expresar una ecuación diferencial de orden  $n$  por una ecuación diferencial vectorial-matricial de primer orden. Si  $n$  elementos del vector son un conjunto de variables de estado, a la ecuación diferencial vectorial-matricial se le llama *ecuación de estado*.

#### Representación en el espacio de estado de sistemas de ecuaciones diferenciales lineales de orden $n$

Sea el siguiente sistema de orden  $n$  :

$$y^{(n)} + a_1 y^{(n-1)} + \dots + a_{n-1} \dot{y} + a_n y = u \quad (3.31)$$

Considerando que el conocimiento de  $y(0)$ ,  $y'(0)$ , ... junto con la entrada  $u(t)$  para  $t \geq 0$ , determina totalmente el comportamiento futuro del sistema, se puede tomar  $y(t)$ ,  $y'(t)$ , ... como un juego de  $n$  variables de estado.

Se define:

$$\begin{aligned} x_1 &= y \\ x_2 &= \dot{y} \\ &\vdots \\ x_n &= y^{(n-1)} \end{aligned}$$

Entonces la ecuación (3.31) se puede escribir como:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 \\ &\vdots \\ \dot{x}_{n-1} &= x_n \\ \dot{x}_n &= -a_n x_1 - a_1 x_n + u \end{aligned}$$

6

$$\mathbf{x} = \mathbf{Ax} + \mathbf{Bu} \quad (3.32)$$

donde:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ -a_n & -a_{n-1} & -a_{n-2} & \dots & -a_1 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}$$

La que se puede expresar como:

$$y = \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

$$y = \mathbf{C}\mathbf{x} \quad (3.33)$$

La ecuación diferencial de primer orden, ecuación (3.32), es la ecuación de estado, y la ecuación (3.33), es la ecuación de salida.

### 3.8 CONTROLABILIDAD [3].

Se dice que un sistema es controlable en el tiempo  $t_0$ , si por medio de un vector de control no restringido, es posible transferir el sistema desde cualquier estado inicial  $\mathbf{X}(t_0)$  a cualquier otro estado, en un tiempo finito.

#### Independencia lineal de vectores.

Se dice que los vectores  $\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_n$  son linealmente independientes si:

$$C_1\mathbf{X}_1 + C_2\mathbf{X}_2 + \dots + C_n\mathbf{X}_n = 0$$

donde  $C_1, C_2, \dots, C_n$  son constantes, implica que:

$$C_1 = C_2 = \dots = C_n = 0$$

por el contrario, se dice que los vectores  $X_1, X_2, \dots, X_n$ , son linealmente dependientes si y solo si  $X_i$  se puede representar como una combinación lineal  $X_j$  ( $j = 1, 2, \dots, n; j \neq i$ ); ó

$$X_i = \sum_{\substack{j=1 \\ j \neq i}}^n C_j X_j$$

para algún conjunto de constantes  $C_j$ .

Nótese que si una matriz de  $n \times n$  es no singular (es decir, la matriz es de rango  $n$  o el determinante es no nulo), entonces  $n$  vectores columna (o renglón), son linealmente independientes. Si la matriz de  $n \times n$ , es singular (es decir, el rango de la matriz es menor que  $n$  o el determinante es cero), entonces  $n$  vectores columna (o renglón), son linealmente dependientes.

## Controlabilidad de sistemas en tiempo continuo.

Considere el sistema en tiempo continuo:

$$\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{Bu} \quad (3.34)$$

El sistema descrito por la ecuación (3.34) se dice que es de estado controlable en  $t = t_0$  si es posible construir una señal de control no restringida, que pueda transferir un estado inicial en cualquier estado final en un intervalo de tiempo finito  $t_0 \leq t \leq t_1$ . Si todo estado es controlable, entonces se dice que el sistema es de controlabilidad del estado completo.

El sistema dado por la ecuación (3.34) es controlable de estado completo si y solo si los vectores  $\mathbf{B}, \mathbf{AB}, \dots, \mathbf{A}^{n-1}\mathbf{B}$  son linealmente independientes, o la matriz de  $n \times n$

$$\left[ \mathbf{B} : \mathbf{AB} : \dots : \mathbf{A}^{n-1}\mathbf{B} \right] \quad (3.35)$$

es de rango  $n$ .

### 3.9 OBSERVABILIDAD [3].

Considere el sistema no forzado descrito por las siguientes ecuaciones:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} \quad (3.36)$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} \quad (3.37)$$

Se dice que el sistema es completamente observable, si cada estado  $\mathbf{X}(t_0)$  se puede determinar a partir de la observación de  $\mathbf{Y}(t)$  en un intervalo finito de tiempo  $t_0 \leq t \leq t_1$ . Por lo tanto el sistema es completamente observable, si cada transición de estado, afecta eventualmente a cada elemento del vector de salida.

El sistema descrito por las ecuaciones (3.36) y (3.37) es completamente observable si y solo si, la matriz de  $n \times m$ :

$$\left[ \mathbf{C}^* : \mathbf{A}^* \mathbf{C}^* : \dots : (\mathbf{A}^*)^{n-1} \mathbf{C}^* \right] \quad (3.38)$$

es de rango  $n$  o tiene  $n$  vectores columna linealmente independientes. Esta matriz se denomina *matriz de observabilidad*.

### 3.10 LA MATRIZ DE HANKEL Y LA DETERMINACION DEL ORDEN DEL MODELO DEL SISTEMA [2].

**Teorema:** Un modelo de  $n$  estados para una función de transferencia SISO de orden  $n$ ,  $H(s)$ , es completamente controlable y completamente observable si y solo si  $H(s)$  es irreducible, o no tiene cancelación de polos y ceros.

Este teorema es aplicable al modelo de estado discreto:

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}u(k) \quad (3.39)$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) \quad (3.40)$$



Este sistema es irreducible si y solo si el sistema es completamente controlable y completamente observable. Entonces :

$$\text{rango } \mathbf{M}_{T_c} = n \quad \text{rango } \mathbf{M}_{T_o} = n$$

donde  $\mathbf{M}_{T_c}$  y  $\mathbf{M}_{T_o}$  son las matrices de controlabilidad y observabilidad discretas. La matriz de Hankel es el producto de las matrices de controlabilidad y observabilidad :

$$\mathbf{H}_T = \mathbf{M}_{T_o} \mathbf{M}_{T_c} \quad (3.41)$$

esto es derivado del hecho que el modelo de la respuesta al pulso discreto  $h_T(k)$  está relacionada al modelo de estado discreto:

$$h_T(k) = \mathbf{C} \mathbf{A}^{k-1} \mathbf{B} \quad k = 1, 2, \dots \quad (3.42)$$

En consideración al orden del modelo del sistema, la matriz de Hankel tiene rango  $n$  si y solo si el modelo de estado es completamente controlable y completamente observable; el modelo de estado es irreducible.

La matriz de Hankel de alto orden  $N \times N$  puede ser formada usando los parámetros de Markov ó  $h_T(k)$ . Entonces cuando el rango de ésta gran matriz de Hankel es determinado, se tendría en teoría un rango  $n$  donde  $n$  es el propio orden, dimensión de estado completamente controlable y completamente observable. La determinación del rango es encontrado por medio de análisis de valores singulares.

**Ejemplo 3.1:** Considérese la siguiente función de transferencia

$$H(s) = \frac{1}{s^3 + 0.4s^2 + 1.14s + 0.22}$$

la cual ha sido discretizada a un  $T= 1s$ . Los parámetros de Markov son arreglados en una matriz de Hankel arbitrariamente seleccionada de  $4 \times 4$ :

$$H_T = \begin{bmatrix} 0.14287 & 0.73603 & 1.1658 & 0.96007 \\ 0.73603 & 1.1658 & 0.96007 & 0.32998 \\ 1.1658 & 0.96007 & 0.32998 & -0.14931 \\ 0.96007 & 0.32998 & -0.14931 & -0.13050 \end{bmatrix}$$

Los 4 valores singulares son:

$$\sigma_1 = 2.6404$$

$$\sigma_3 = 0.34049$$

$$\sigma_2 = 1.4727$$

$$\sigma_4 = 1.24 \times 10^{-8}$$

Ya que  $\sigma_4$  es muy cercana a 0, esto indica que el sistema es de rango 3; la matriz de Hankel de 4x4 tiene 3 valores singulares que no son cero de modo que se concluye que el orden del modelo es 3

**Ejemplo 3.2.** Considere el siguiente sistema:

$$H(s) = \frac{5000(s^2 + 1.11s + 0.101)}{(s^4 + 15s^3 + 83s^2 + 359s + 290)}$$

$$= \frac{5000(s + 1.01)(s + 0.1)}{(s + 1)(s + 10)(s + 2 + 5j)(s + 2 - 5j)}$$

Este sistema se puede reducir a un sistema de tercer orden debido a la proximidad de la cancelación de polos y ceros del polo en -1 y el cero en -1.01

Analizando la matriz de Hankel, esta nos dará la información concerniente al orden del sistema. En particular, si se discretiza el modelo de cuarto orden y formamos una matriz de hankel discreta usando  $T=0.1s$ , tendremos:

$$H_T = \begin{bmatrix} 15.649 & 22.671 & 12.357 & -1.0218 & -11.071 \\ 22.671 & 12.357 & -1.0218 & -11.071 & -15.110 \\ 12.357 & -1.0218 & -11.071 & -15.110 & -15.336 \\ -1.0218 & -11.071 & -15.110 & -15.336 & -11.575 \\ -11.071 & -15.110 & -15.336 & -11.575 & -6.3788 \end{bmatrix}$$

Los valores singulares de  $H_T$  son:

$$\sigma_1 = 47.669 \quad \sigma_4 = 2.25 \times 10^{-4}$$

$$\sigma_2 = 44.929 \quad \sigma_5 = 9.23 \times 10^{-9}$$

$$\sigma_3 = 7.4929$$

Únicamente los primeros tres valores singulares son grandes. El rango de la matriz de Hankel es efectivamente tres. Esto indica que la dimensión de estado importante es tres, no cuatro.

### 3.11 DESCRIPCION DEL ALGORITMO ERA [4].

El algoritmo ERA (Eingensystem Realization Algorithm) usa una matriz de *Hankel*, construida de los parámetros de Markov del sistema. Los parámetros de Markov definen la respuesta al pulso [5]. Por medio de la Descomposición de Valores Singulares (SVD), una realización (un modelo) de orden igual al rango de ésta matriz puede ser encontrado. Para poder obtener una mínima realización el ruido debe de ser eliminado [6].

Las ecuaciones en el espacio de estado en tiempo discreto son escritas usualmente en la forma siguiente:

$$\mathbf{x}(k + 1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}u(k) \quad (3.39)$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) \quad (3.40)$$

En sistemas de tiempo discreto, los parámetros de Markov son justamente los valores de la respuesta al pulso discreto.

Los parámetros de Markov pueden ser obtenidos experimentalmente para formar la matriz de Hankel

$$H(k) = \begin{bmatrix} h(k) & h(k+1) & \cdots & h(k+s-1) \\ h(k+1) & h(k+2) & \cdots & h(k+s) \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ h(k+r-1) & h(k+r) & \cdots & h(k+r+s-2) \end{bmatrix}$$

donde  $r$  y  $s$  deben ser escogidas lo suficientemente grande de modo que el rango de la matriz no sea incrementado por los incrementos de  $r$  y  $s$  [7].

Defínase una matriz de controlabilidad  $\mathbf{P}$ , una matriz de observabilidad  $\mathbf{Q}$  y matrices  $\mathbf{E}_0$  y  $\mathbf{E}_m$  de acuerdo a

$$\mathbf{P} = [\mathbf{B}, \mathbf{A}\mathbf{B}, \mathbf{A}^2\mathbf{B}, \dots, \mathbf{A}^{s-1}\mathbf{B}], \quad (3.43)$$

$$\mathbf{Q} = [\mathbf{C}^T, (\mathbf{C}\mathbf{A})^T, \dots, (\mathbf{C}\mathbf{A}^{r-1})^T], \quad (3.44)$$

$$\mathbf{E}_\beta^T = [\mathbf{I}_\beta, \mathbf{0}_\beta, \dots, \mathbf{0}_\beta]^T, \quad (3.45)$$

$$\mathbf{E}_m^T = [\mathbf{I}_m, \mathbf{0}_m, \dots, \mathbf{0}_m]^T, \quad (3.46)$$

donde  $\beta$  y  $m$  son las dimensiones de el vector de medición  $y(k)$  y el vector de control  $u(k)$ .

$\mathbf{I}_m$ , e  $\mathbf{I}_\beta$  son matrices de identidad de orden  $\beta$  y  $m$ , y  $\mathbf{0}_\beta$  y  $\mathbf{0}_m$  son matrices cero  $\beta \times \beta$  y  $m \times m$ .

Se puede probar que el orden mínimo de cualquier realización de un sistema es igual al rango de  $H(0)$  [2]. Una realización mínima puede ser encontrada usando SVD de  $H(0)$ :

$$H(0) = \mathbf{U}_N \mathbf{S}_N \mathbf{V}_N^T \quad (3.47)$$

donde:

$$\mathbf{S}_N = \text{diag}(d_1, d_2, \dots, d_N) \quad (3.48)$$

y  $d_1$  hasta  $d_N$  son los valores singulares de la matriz de Hankel.

Ruido y no linealidades pueden hacer que el rango de la matriz de Hankel sea más grande que el orden del sistema. En [6] se establece que el orden de la realización puede ser aproximado examinando los valores singulares de la matriz de Hankel  $H(0)$ . Si los valores singulares  $d_{n+1}, d_{n+2}, \dots, d_N$  son muy pequeños comparados con  $d_1, \dots, d_n$ ,  $H(0)$  puede ser asumido de rango  $n$ , y la realización mínima puede ser de rango  $n$ , en otras palabras:

$$H(0) = \mathbf{U}_n \mathbf{S}_n \mathbf{V}_n^T \quad (3.47)$$

donde  $\mathbf{S}_n = \text{diag}(d_1, \dots, d_n)$ , y  $\mathbf{U}_n$  y  $\mathbf{V}_n$  son obtenidas al truncar las últimas  $N-n$  columnas de  $\mathbf{U}_N$  y  $\mathbf{V}_N$  respectivamente. Se puede probar que una realización puede ser obtenida de la manera siguiente:

$$\mathbf{A} = \mathbf{S}_n^{-1/2} \mathbf{U}_n^T H(1) \mathbf{V}_n \mathbf{S}_n^{-1/2}, \quad \mathbf{B} = \mathbf{S}_n^{-1/2} \mathbf{V}_n^T \mathbf{E}_m, \quad \mathbf{C} = \mathbf{E}_p \mathbf{U}_n \mathbf{S}_n^{1/2} \quad (3.49)$$



Las ecuaciones de estado obtenidas, están en forma discreta. Todas las realizaciones derivadas por el ERA son transformadas a realizaciones equivalentes en tiempo continuo usando la función en MATLAB™ “d2c” [8]. Después que la realización en forma de ecuaciones de estado continuas son encontradas, una función de transferencia en términos de numerador y denominador es obtenida usando la función de MATLAB™ “ss2tf” [8]. En el apéndice C se muestran los programas en MATLAB™ que ejecutan el algoritmo ERA. El programa “Hankera.m” forma la matriz de Hankel  $H(k)$  dados los parámetros de Markov  $Y(k)$  del sistema. El programa “era.m” es el que implementa el algoritmo ERA para la identificación de una realización (A,B,C).

### **3.12 EJEMPLO DEL USO DEL ERA EN EL MODELADO DE UN SISTEMA HIDRAULICO.**

En ésta sección se mostrará el uso del ERA para modelar un sistema hidráulico. Un diagrama sencillo de éste sistema se observa en la Figura 3.6, dicha unidad se encuentra en el Laboratorio de Automatización y control de la Universidad Don Bosco.

Un sistema hidráulico es aquel en el cual un liquido fluye a través de bombas, tanques y otros componentes asociados [4]. En la mayoría de sistemas hidráulicos, la fuente de energía es una bomba. La velocidad de la bomba es gobernada por la variación

de la corriente de armadura del motor DC hasta que la razón de flujo de salida deseada es obtenida.

Antes de aplicar una entrada escalón, se deben de realizar mediciones experimentales de la relación de entrada/salida del sistema para determinar la región lineal de las variables de entrada/salida. La relación entrada/salida del sistema hidráulico es mostrada en la Figura 3.7.

Se puede observar que los valores de voltaje de entrada entre 4.5 Voltios (0.7 l/min.) y 12 Voltios (4.5 l/min.) definen la zona lineal aproximada. La amplitud del escalón aplicado al sistema fue seleccionado entre éstos dos límites (desde 5V hasta 8V). Las Figuras 3.8 y 3.9 muestran la respuesta al escalón y la respuesta al pulso respectivamente. Recuérdese que la respuesta al pulso se obtiene de la respuesta escalón por medio de la relación

$$h_T(k) = r(kT) - r(kT - T) \quad (3.19)$$

El archivo "era.m" es usado para calcular las realizaciones de orden 2,4 y 6. Las Figuras 3.11, 3.13 y 3.15 muestran las respuestas escalón de las realizaciones hechas en ERA de orden 2,4 y 6 comparada con los datos experimentales.

Se puede observar que no hay mucha diferencia entre las realizaciones de 2, 4 y 6° orden debido a que a partir de un orden mayor que 4 solo se agregan cancelaciones de polos y ceros, como se observan en las Figuras 3.12, 3.14 y 3.16. La Figura 3.10 plotea la magnitud de los valores singulares de la matriz de Hankel  $H(0)$  obtenido por ERA. Se puede notar que dos valores singulares tienen la más grande magnitud relativa, por lo tanto el mínimo orden fue escogido ser 2.

Las ecuaciones (3.50) y (3.51) presentan la realización discreta de orden 2 y el modelo continuo equivalente:

$$H(z) = \frac{0.039774z + 0.1651}{z^2 - 1.1539z + 0.026646} \quad (3.50)$$

$$H(s) = \frac{-1.5861s + 37.655}{s^2 + 13.225s + 20.627} \quad (3.51)$$

Cabe mencionar que el tiempo de muestreo en éste experimento fue de 0.1ms.

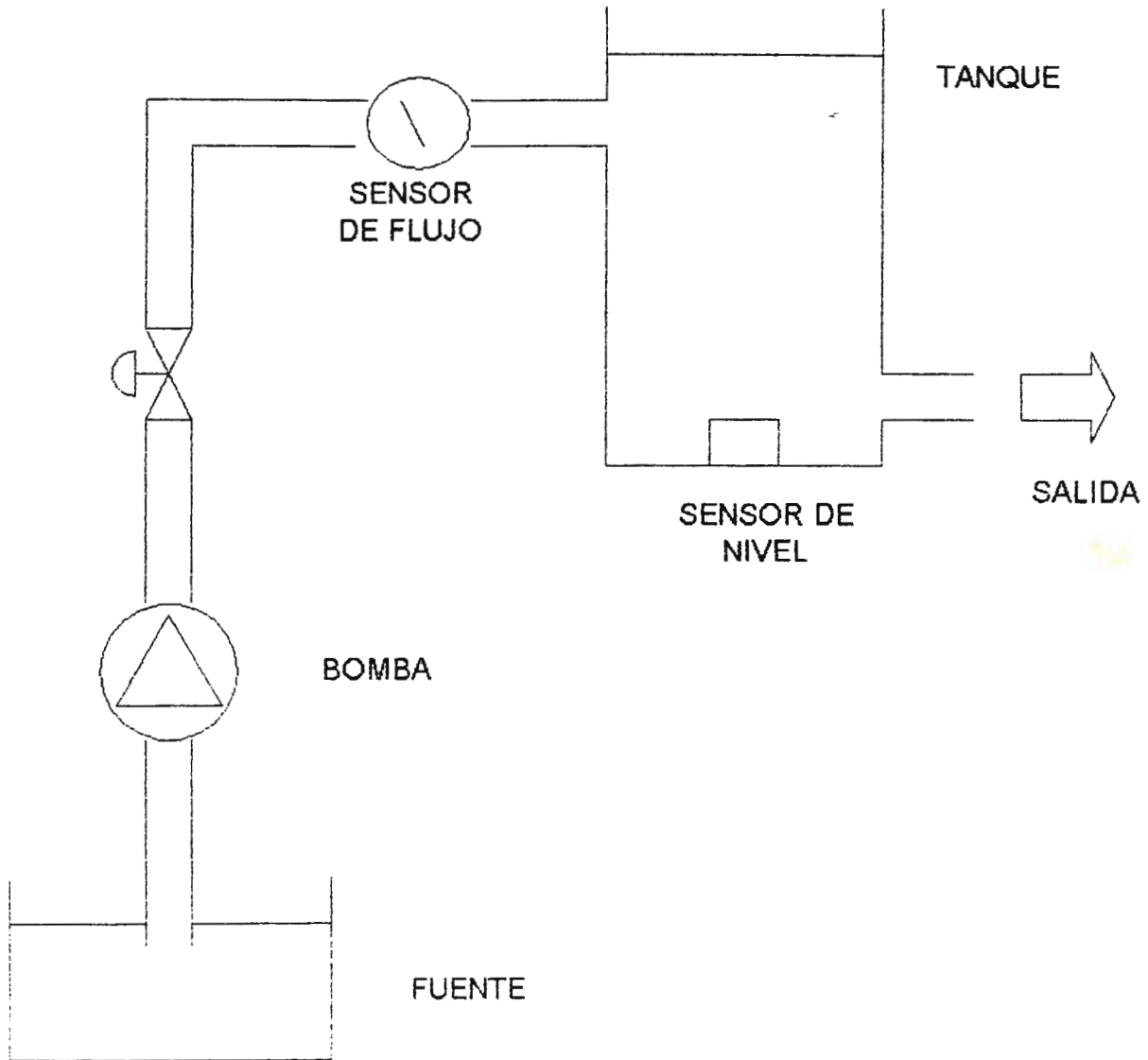


Fig. 3.6 Sistema hidráulico a modelar.

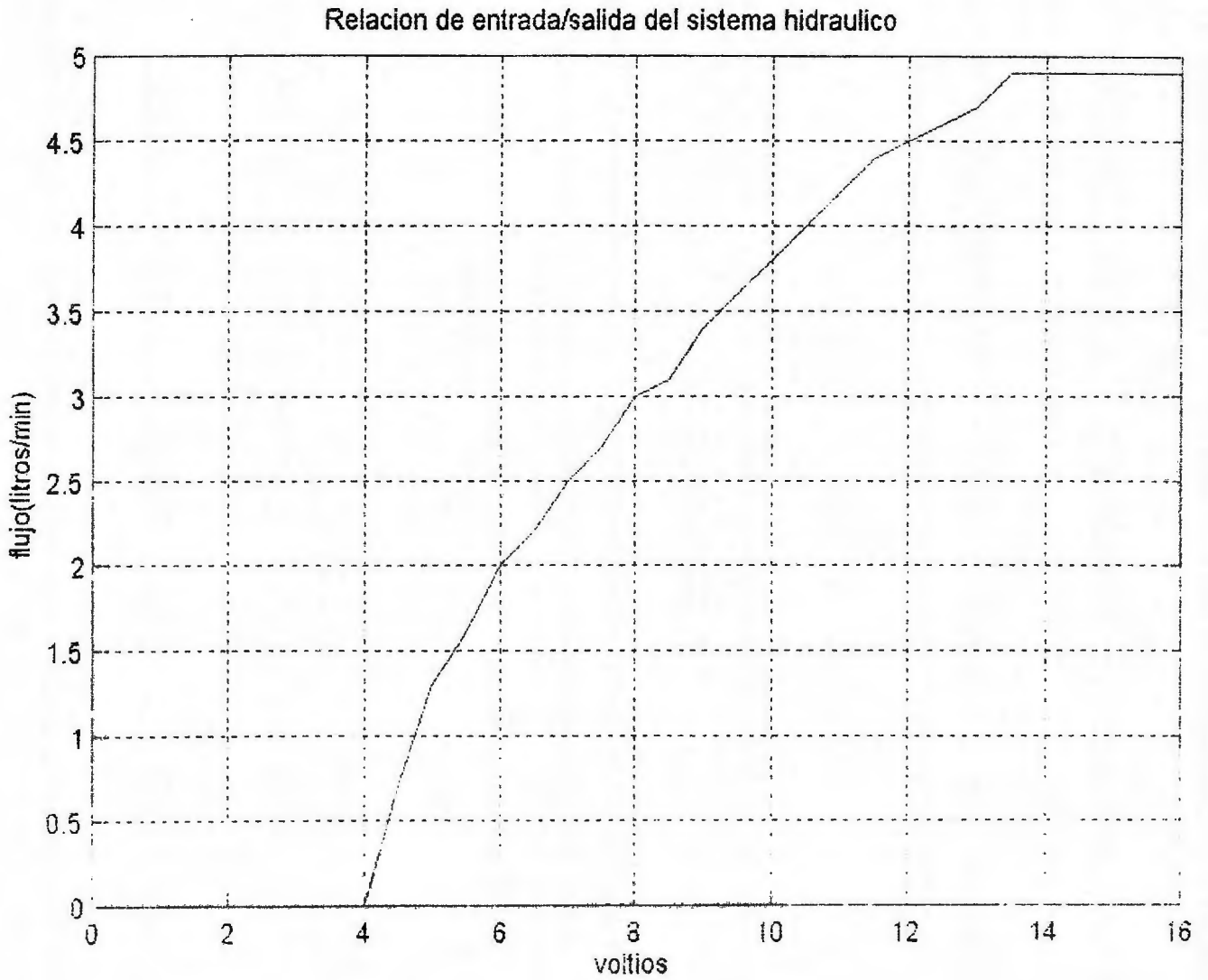


Figura 3.7 Relación entrada/salida del sistema hidráulico.

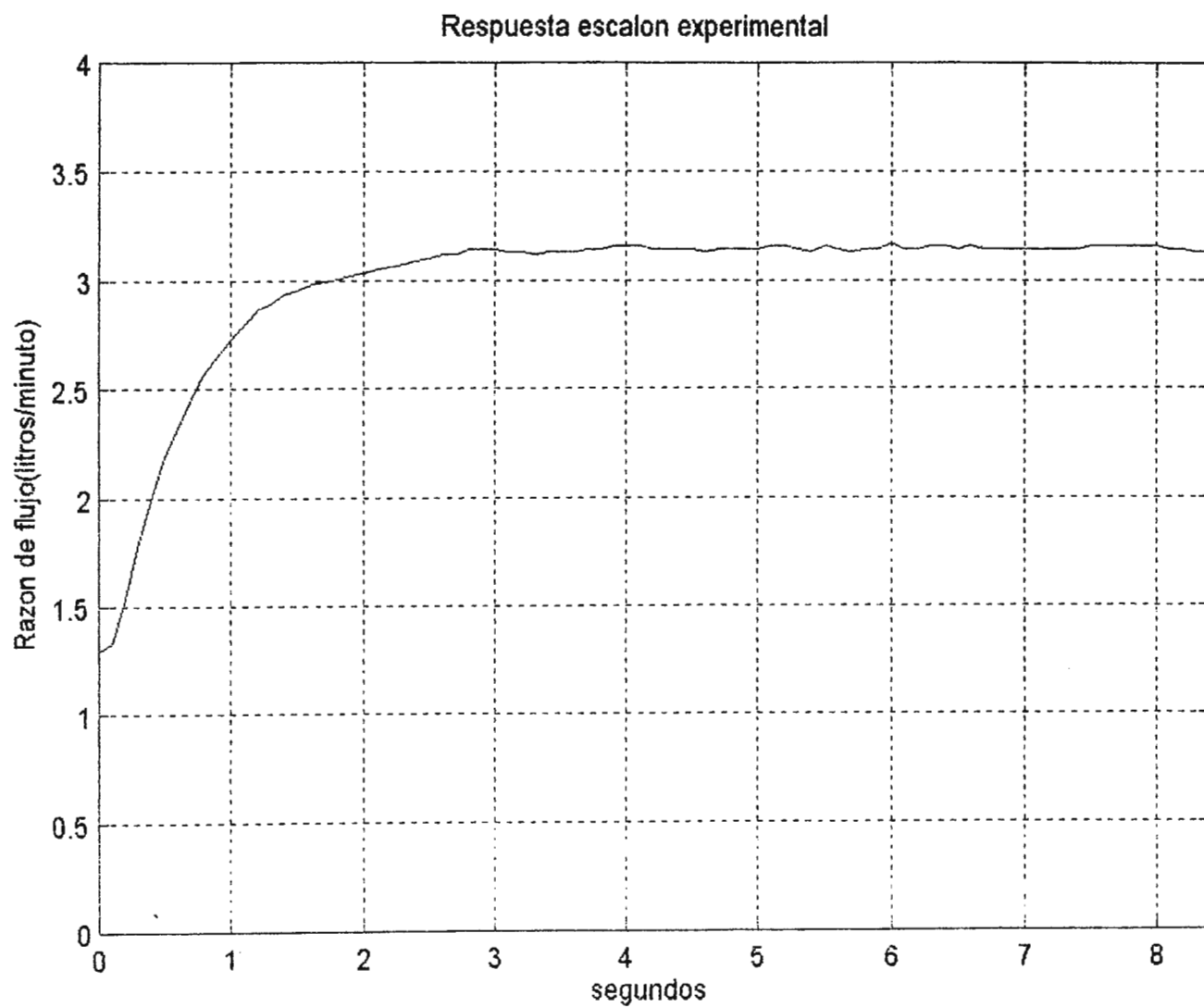


Figura 3.8 Respuesta al escalón experimental.

respuesta pulso del sistema hidraulico

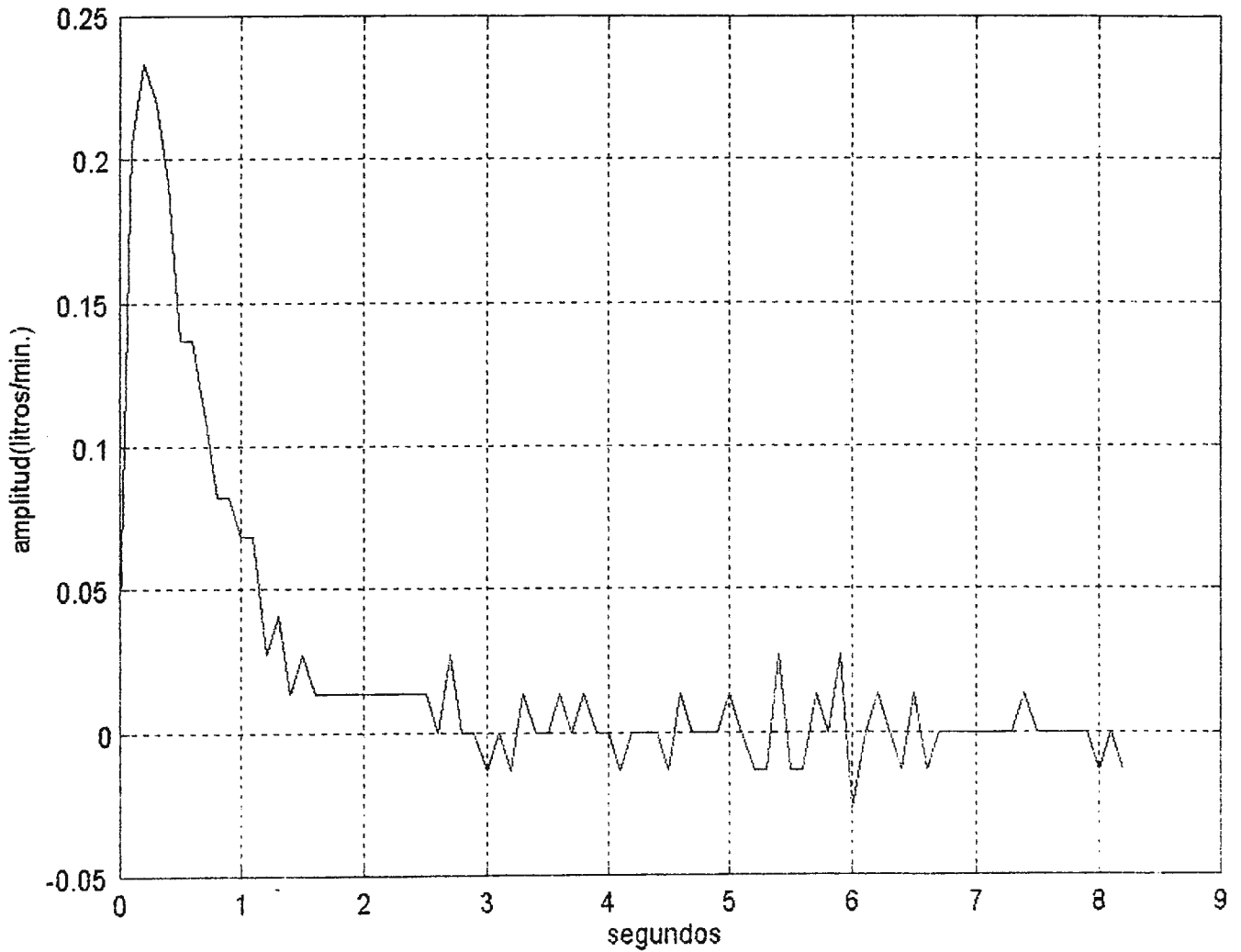


Figura 3.9 Respuesta pulso del sistema hidráulico.

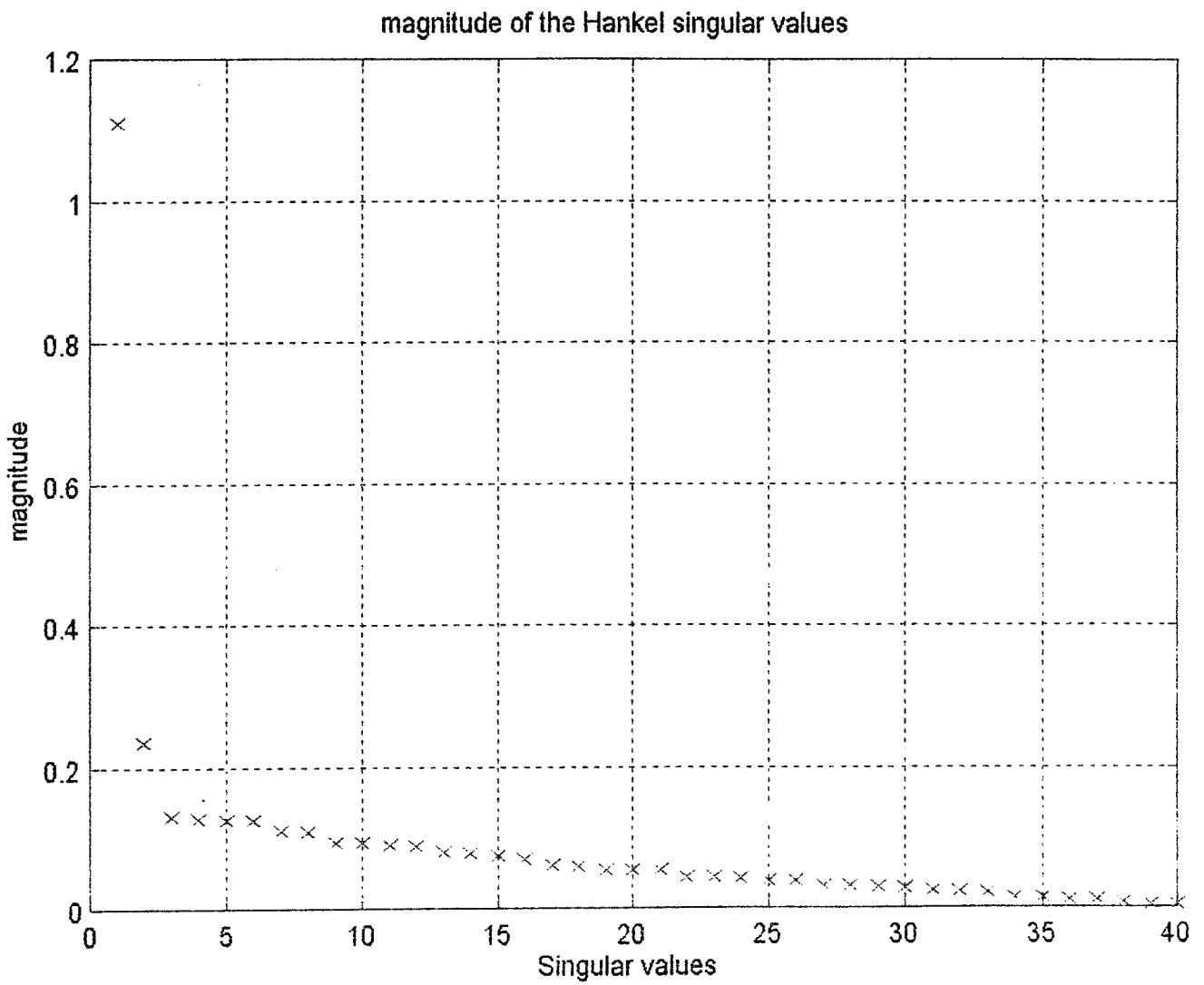


Figura 3.10 Valores singulares de la matriz de Hankel.



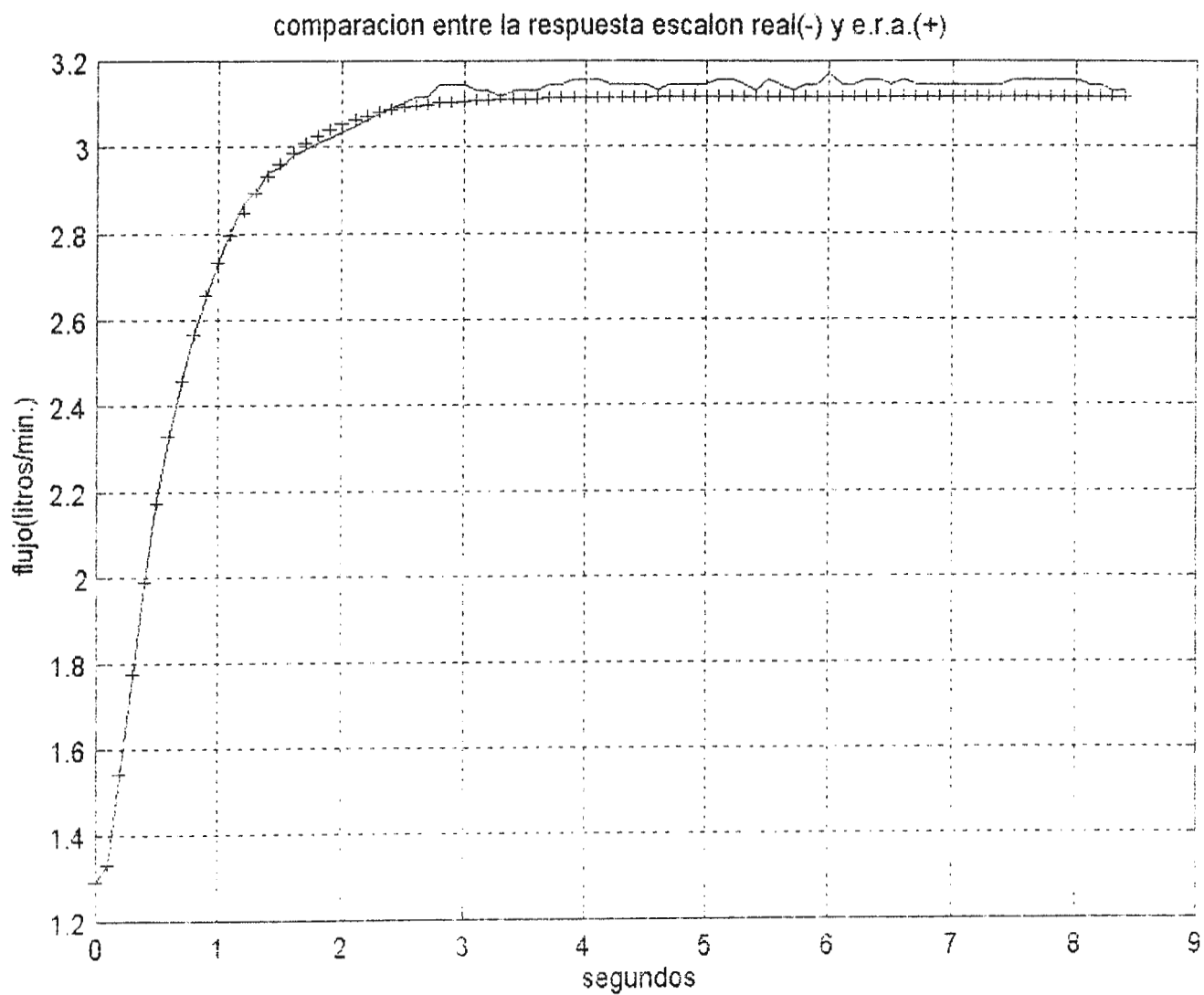


Figura 3.11 Comparación del modelo experimental con el modelo ERA de 2º orden.

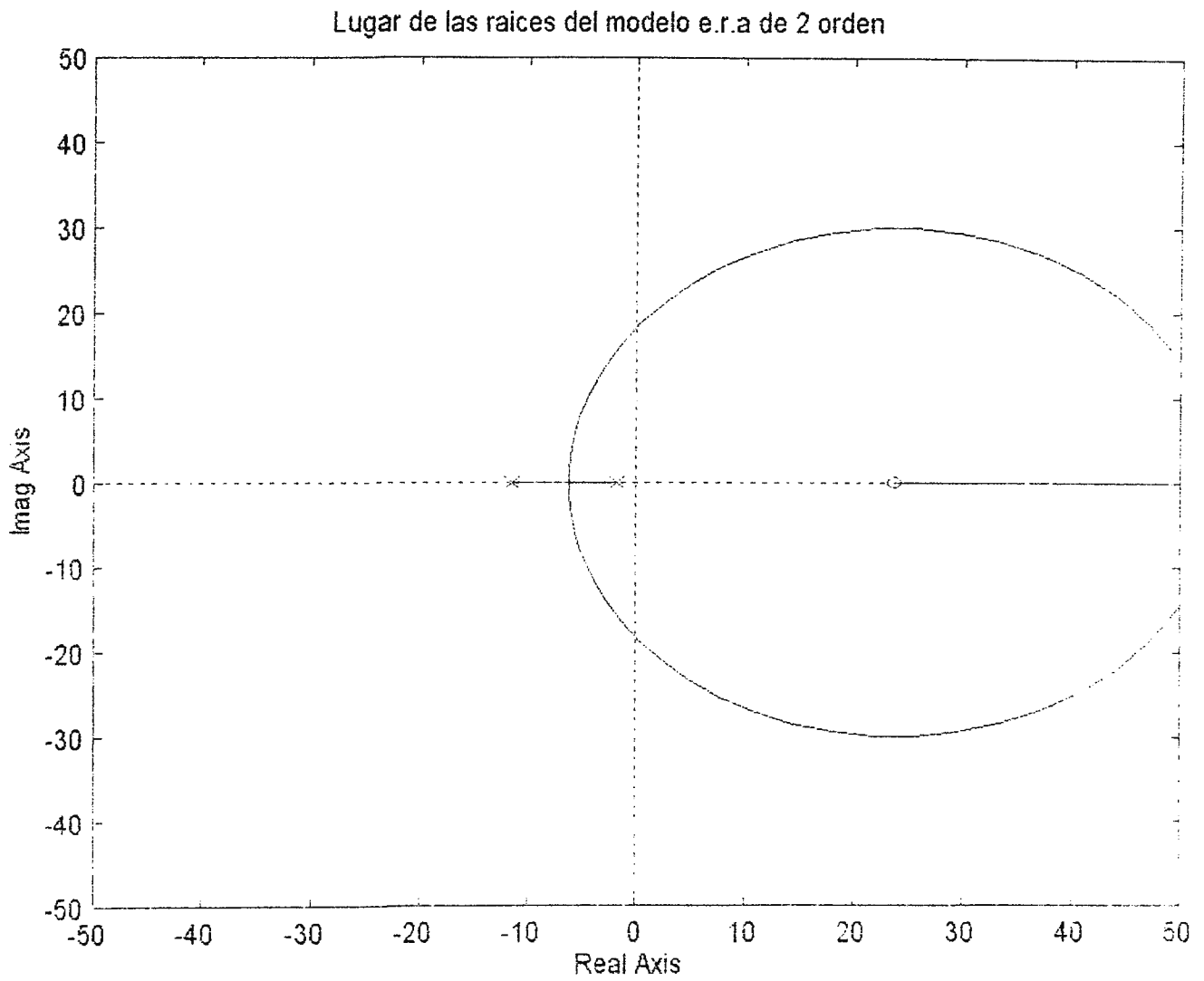


Figura 3.12 Lugar de las raíces del modelo ERA de 2º orden.

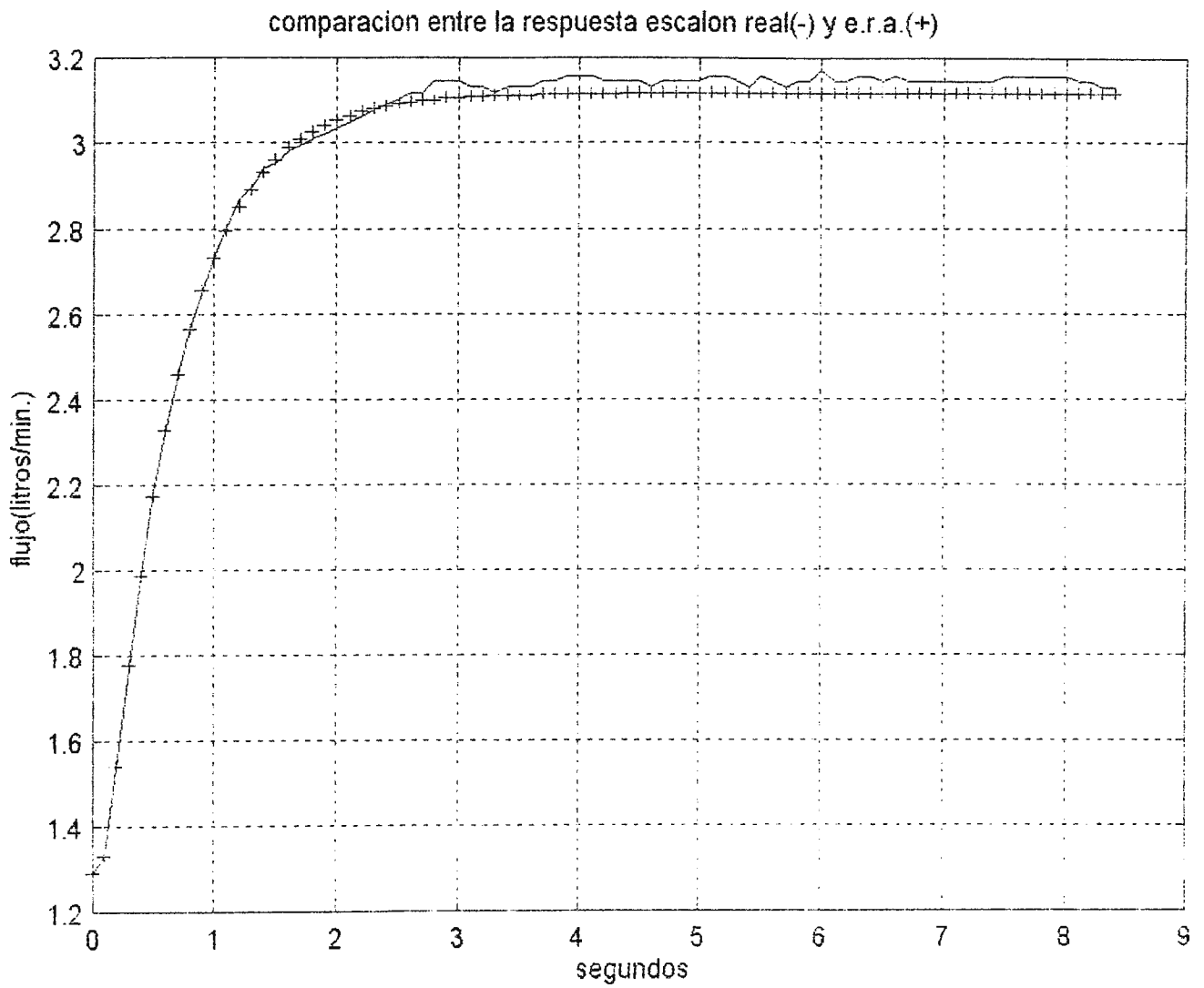


Figura 3.13 Comparación del modelo experimental con el modelo ERA de 4° orden.

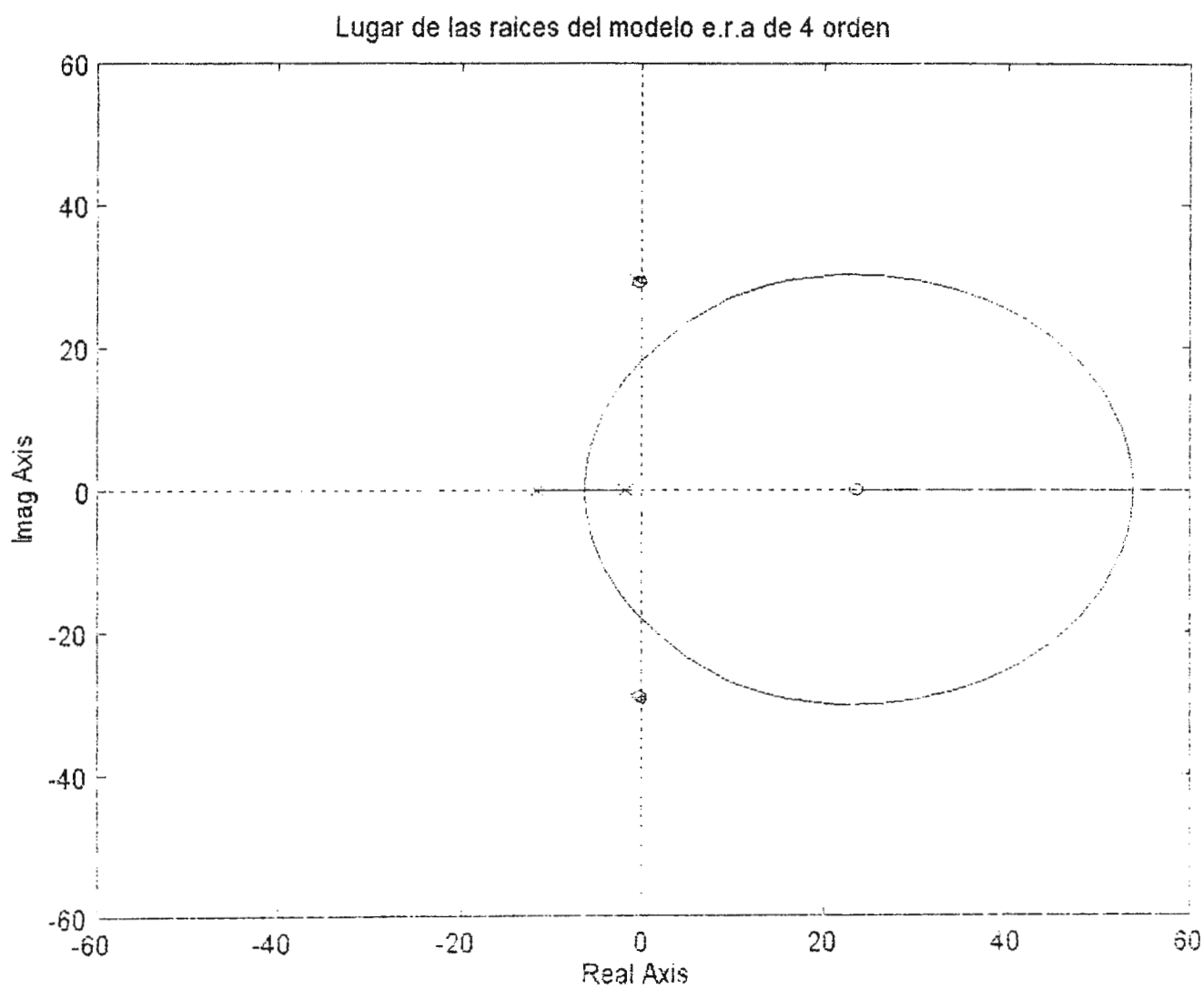


Figura 3.14 Lugar de las raíces del modelo ERA de 4° orden.

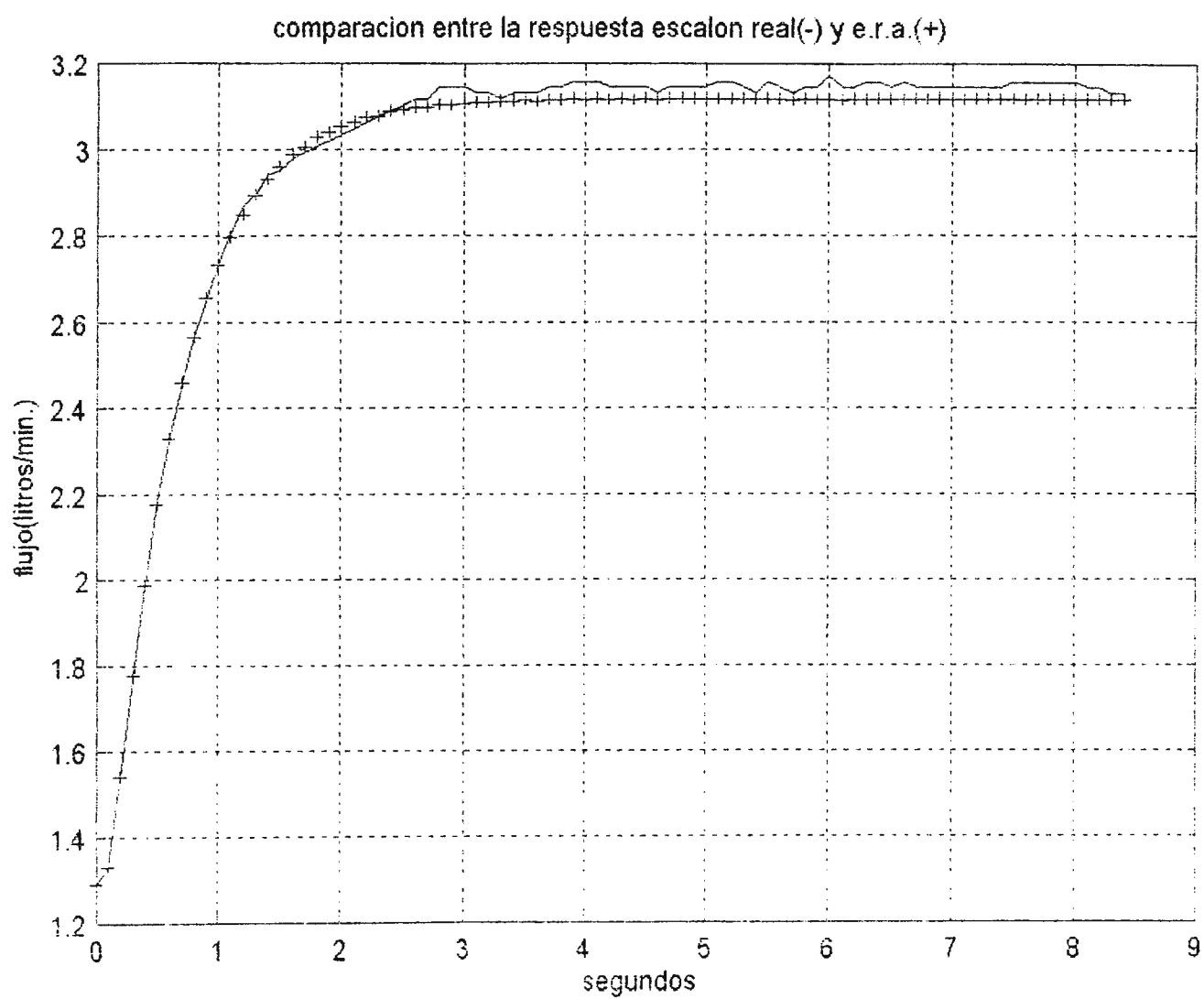


Figura 3.15 Comparación del modelo experimental con el modelo ERA de 6º orden.

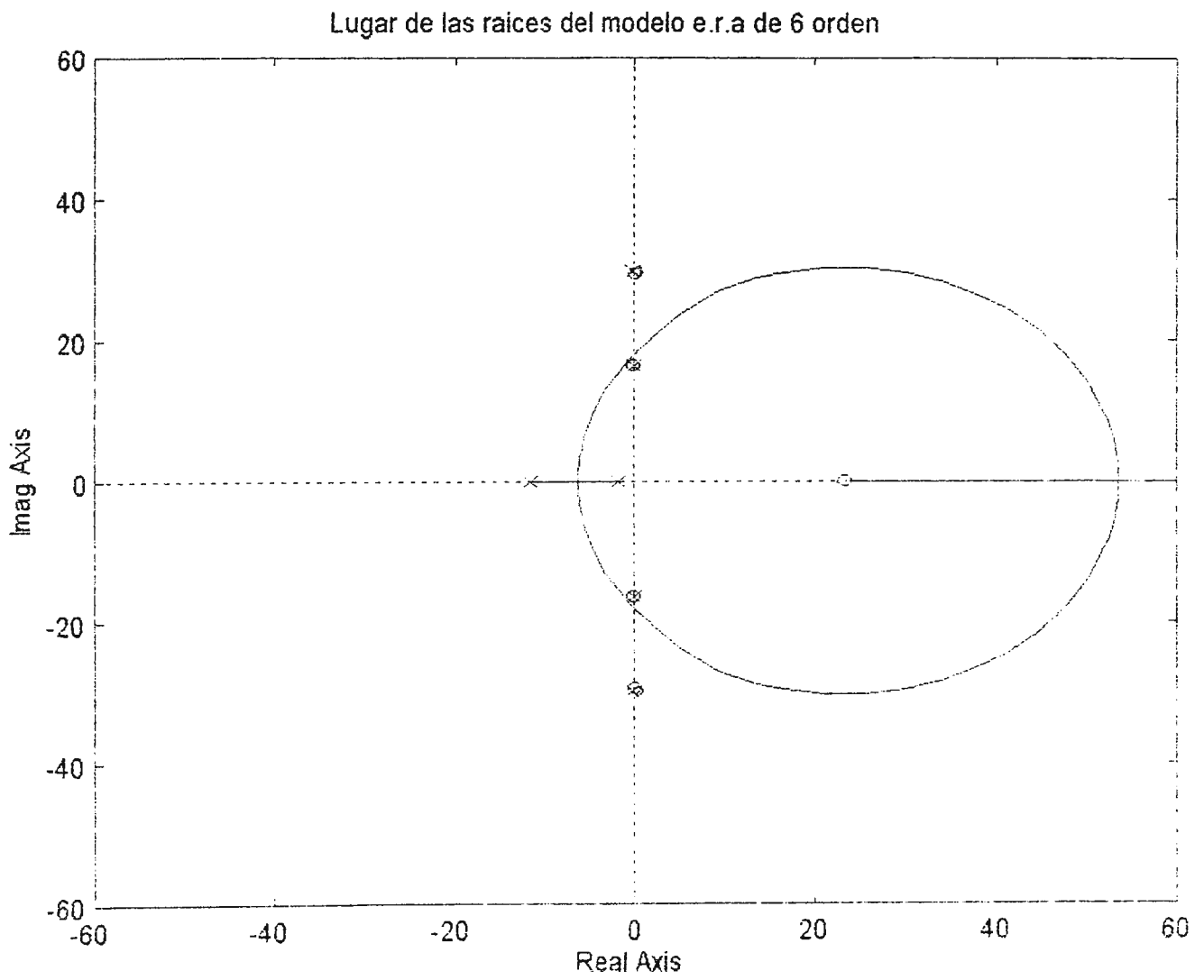


Figura 3.16 Lugar de las raíces del modelo ERA de 6º orden.

# CAPITULO IV

## DISEÑO DE CONTROLADORES DIGITALES Y SIMULACION

### 4.1 INTRODUCCION

En este capitulo presentaremos el uso del equipo adquisidor de datos para modelar sistemas, que luego serán analizados para el diseño de controladores digitales y luego será simulada la respuesta del sistema a una entrada escalón, la cual nos dará una idea de como se comportará el sistema sin necesidad de hacer pruebas con el equipo en si.

### 4.2 USO DEL EQUIPO.

En la Figura 4.1 se observa un diagrama general de la forma en que debe de conectarse el equipo adquisidor de datos a los distintos sistemas a ser modelados.

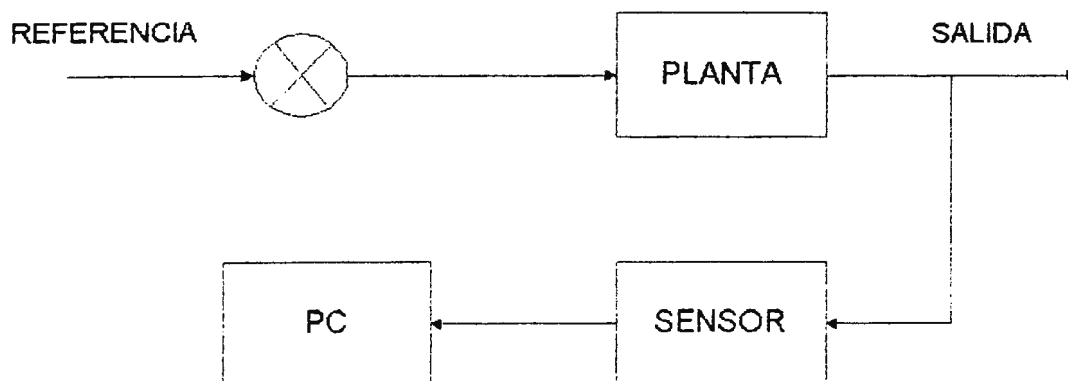


Figura 4.1 Esquema para la adquisición de datos.

La señal de salida del sistema es medida por el sensor y luego esta señal es almacenada en la computadora. Se nota en la Figura 4.1 que la prueba a realizar es en lazo abierto, donde la señal de prueba es aplicada a la entrada llamada referencia.

Más específicamente la Figura 4.2 muestra la conexión de la señal de salida a la computadora.



Figura 4.2 Conexión de la señal de salida de un sistema a la PC.

El sensor proporciona una señal que puede ser de voltaje o corriente. Ya que la entrada del adquisidor de datos comprende de 0-5 Voltios, para el caso de que la señal del sensor sea de corriente, tendrá que utilizarse un acondicionador de señal para traducir la señal de corriente a una señal de voltaje. Para eliminar el ruido se requiere de filtros que estén conectados entre el acondicionador de señal y el adquisidor de datos, estos filtros son del tipo pasa-bajos y su frecuencia de corte dependerá de la constante de tiempo del sistema ( $\tau$ ). En el apéndice [B] se muestran los filtros diseñados para los distintos sistemas a ser modelados. Luego, la salida del adquisidor de datos se conecta al sistema de bus de la computadora para su procesamiento. En el apéndice [D] se encuentran



algunas guías de laboratorio donde se explican con mas detalles el modelado de sistemas dinámicos.

### 4.3 EJEMPLO DE DISEÑO DE CONTROLADORES

En este punto se tratará el diseño de controladores digitales para el caso de un motor D.C., que se encuentra en el Laboratorio de Automatización de la Universidad. Se procederá a hacer el análisis y diseño en el dominio de  $s$  (continuo) y luego se procederá a hacer una transformación en el dominio de  $z$  (discreto). En la Figura 4.3 se muestra la respuesta escalón experimental obtenida por medio del adquisidor de datos. En la Figura 4.4 se observan los valores singulares de la *matriz de Hankel*. El modelo comparado con la respuesta experimental se observa en la Figura 4.5. La función de transferencia del sistema esta dada por la ecuación:

$$G(s) = \frac{6.5865e3S^2 - 1.1537e6S + 8.7889e7}{S^3 + 5.6383e1S^2 + 2.7568e3S + 4.3854e4} \quad (4.1)$$

La función de transferencia del sensor esta dada por :

$$H(s) = 0.002$$

La Figura 4.6 muestra la respuesta en frecuencia del sistema en lazo abierto. Nótese que los márgenes de ganancia y fase para este sistema son negativos, lo que significa que al

tratar de controlar al sistema en lazo cerrado (  $\frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s)H(s)}$  ) el sistema será inestable, tal como se comprueba en la Figura 4.7.

Para poder controlar el sistema comenzaremos por diseñar un compensador *proporcional-integral* (PI), el cual tiene la característica de eliminar el error de estado estable, pero introduce un gran overshoot. El diseño de controladores es mostrado en el apéndice [A]. La Tabla 4.1 muestra la respuesta en frecuencia del sistema, la cual nos servirá para el diseño del controlador.

Magnitud	Fase (grados)	Frecuencia (rad/seg.)
3.3978	-82.983	20.574
3.3030	-96.436	24.124
3.2091	-112.85	28.286
3.0846	-133.68	33.166
2.8234	-160.48	38.888
2.2849	-192.50	45.597
1.5770	-224.42	53.463
0.9907	-251.87	62.687
0.6126	-275.05	73.502
0.3883	-295.63	86.183
0.2568	-314.74	101.05
0.1786	-332.70	118.49
0.1577	-339.21	125.89

Tabla 4.1

Se escoge que el margen de fase requerido es  $55^\circ$  y una ganancia  $k_c=1$ . De la tabla 4.1 se encuentra la frecuencia al cual el ángulo de fase es:

$$-180^\circ + 55^\circ + 5^\circ = -125^\circ$$

Para este ángulo de fase la frecuencia es  $w_1= 33.166$  rad/seg. y la magnitud 3.0846. Por lo tanto la ganancia proporcional es:

$$k_p = \frac{1}{|k_c G_p(jw_1)H(jw_1)|} = \frac{1}{3.0846} = 0.324191143$$

La ganancia integral es:

$$w_0 = 0.1w_1 = 0.1(33.166) = 3.3166$$

$$k_I = k_p w_0 = (0.324191143)(3.3166) = 1.07521$$

La función de transferencia del controlador esta dada por:

$$G_c(s) = 0.3242 + \frac{1.07521}{s}$$

En la Figura 4.8 se muestra la respuesta a un escalón unitario del sistema compensado. Obsérvese la pronunciada inestabilidad del sistema y el alto tiempo de establecimiento. Para minimizar ambas cosas, se le agrega un compensador PID al sistema. La ganancia derivativa se elige arbitrariamente y luego, por prueba y error se llega a la siguiente función de transferencia:

$$G_c(s) = \frac{0.0127s^2 + 0.29s + 18.2}{s}$$

En la Figura 4.9 se muestra la respuesta a un escalón unitario del sistema compensado con PID. El tiempo de establecimiento es aproximadamente 0.75 seg. aunque existe un marcado sobrepaso de aproximadamente 75%. Cuando se ajustan parámetros de controladores, siempre debe haber un compromiso entre la velocidad de respuesta y el overshoot. La Figura 4.10 muestra la respuesta en frecuencia de  $G_c(s)G(s)H(s)$ , donde  $G_c(s)$  es el compensador PI y se observa que efectivamente se ha logrado un margen de fase de  $55^\circ$ , lo mismo se aprecia en la Figura 4.11 para el compensador PID.

Los filtros diseñados anteriormente son analógicos los cuales no se encuentran en una forma conveniente para ser programados en computadora.

Hay muchas maneras para transformar filtros analógicos a digitales [14], pero existe un método que resulta mas conveniente para esta transformación [12], esta se denomina

*Transformación bilinear.* Esta simplemente establece que la variable  $s$  debe ser sustituida mediante la relación:

$$s = \frac{2}{T} \frac{(z-1)}{(z+1)}$$

Para el caso de los controladores PID cuya función de transferencia análoga es:

$$G_c(s) = K_p + \frac{K_i}{s} + K_d s$$

se usa una sustitución diferente para el componente derivativo del controlador [14]. Esta sustitución es:

$$s = \frac{z-1}{Tz}$$

de este modo la sustitución digital del PID análogo es:

$$G_c(z) = K_p + K_i \frac{2}{T} \left[ \frac{z+1}{z-1} \right] + K_d \left[ \frac{z-1}{Tz} \right]$$

Haciendo las sustituciones anteriores a los controladores análogos PI y PID diseñados anteriormente nos resultan los siguientes compensadores digitales:

$$G_c(z)_{PI} = \frac{0.32972z - 0.3189}{z - 1}$$

$$G_c(z)_{PID} = \frac{1.6510z^2 - 2.739z - 1.27}{z^2 - z}$$

Las ecuaciones anteriores ya están listas para programarse en computadora. Los métodos de programación de filtros digitales por computadora son mostrados en la referencia [14].

Ahora considérese el sistema de la Figura 4.12

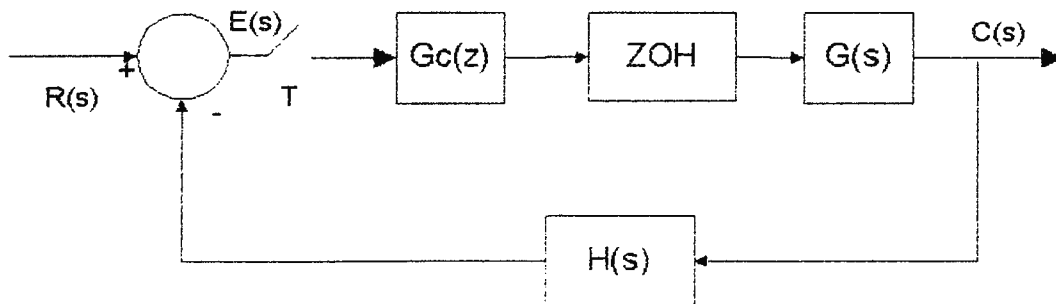


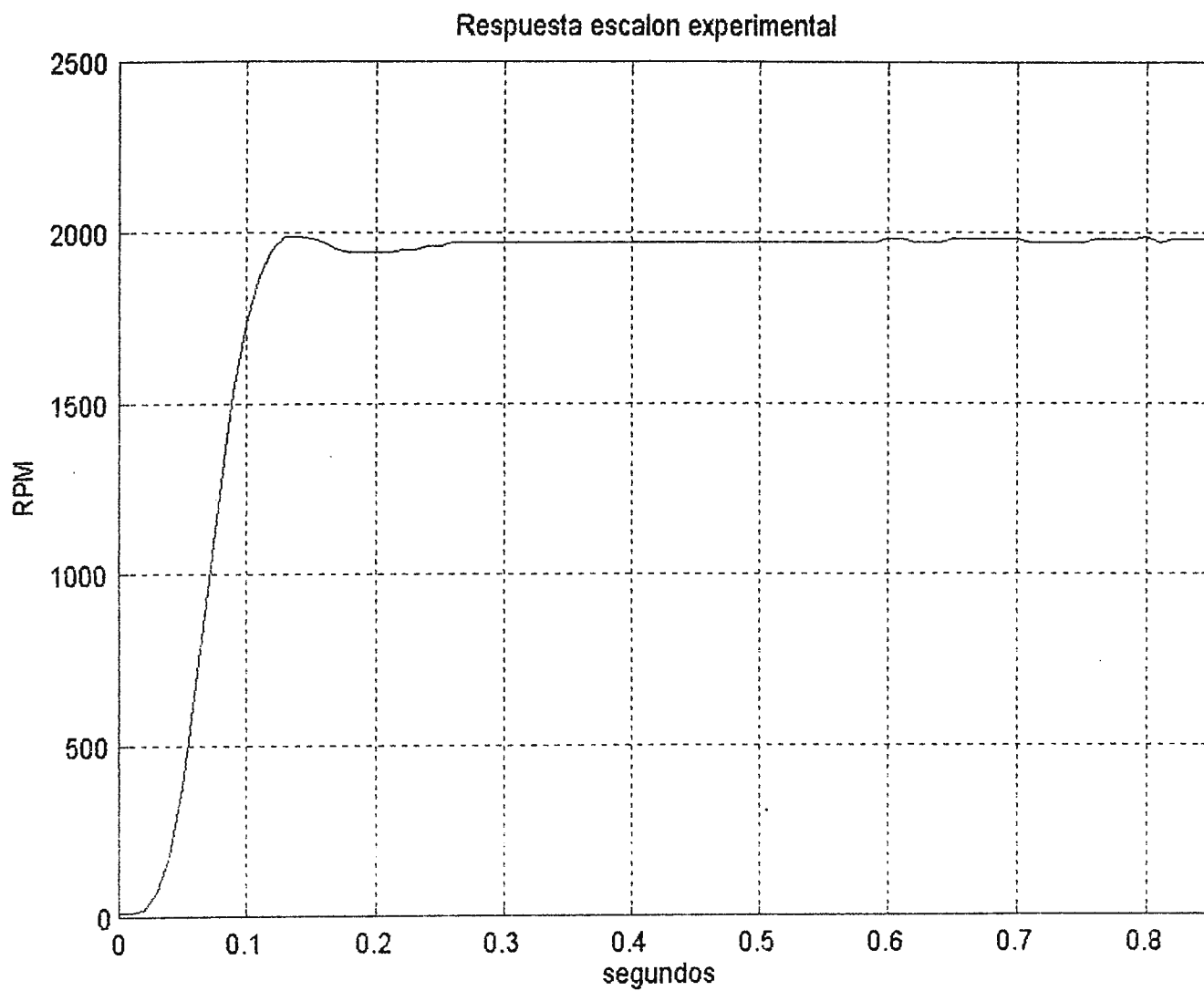
Figura 4-12 Diagrama de bloques de un sistema en lazo cerrado.

donde la función de transferencia de lazo cerrado está definida por

$$\frac{C(z)}{R(z)} = \frac{G_c(z)G(z)}{1 + G_c(z)\overline{GH}(z)}$$

donde  $G_c(z)\overline{GH}(z)$  es la función de transferencia de lazo abierto la cual proporciona información del comportamiento del sistema en lazo cerrado.

Las Figuras 4.13 y 4.14 muestran la respuesta en frecuencia de los sistemas compensados con los controladores digitales donde se puede observar la reducción de los márgenes de estabilidad por lo que se concluye que la digitalización produce inestabilidad. El efecto desestabilizador se reduce mientras mas pequeño sea el tiempo de muestreo  $T$ , el cual es un detalle a tomar en cuenta en el proceso de digitalización.



**Figura 4.3 Respuesta escalón experimental del motor D.C.**



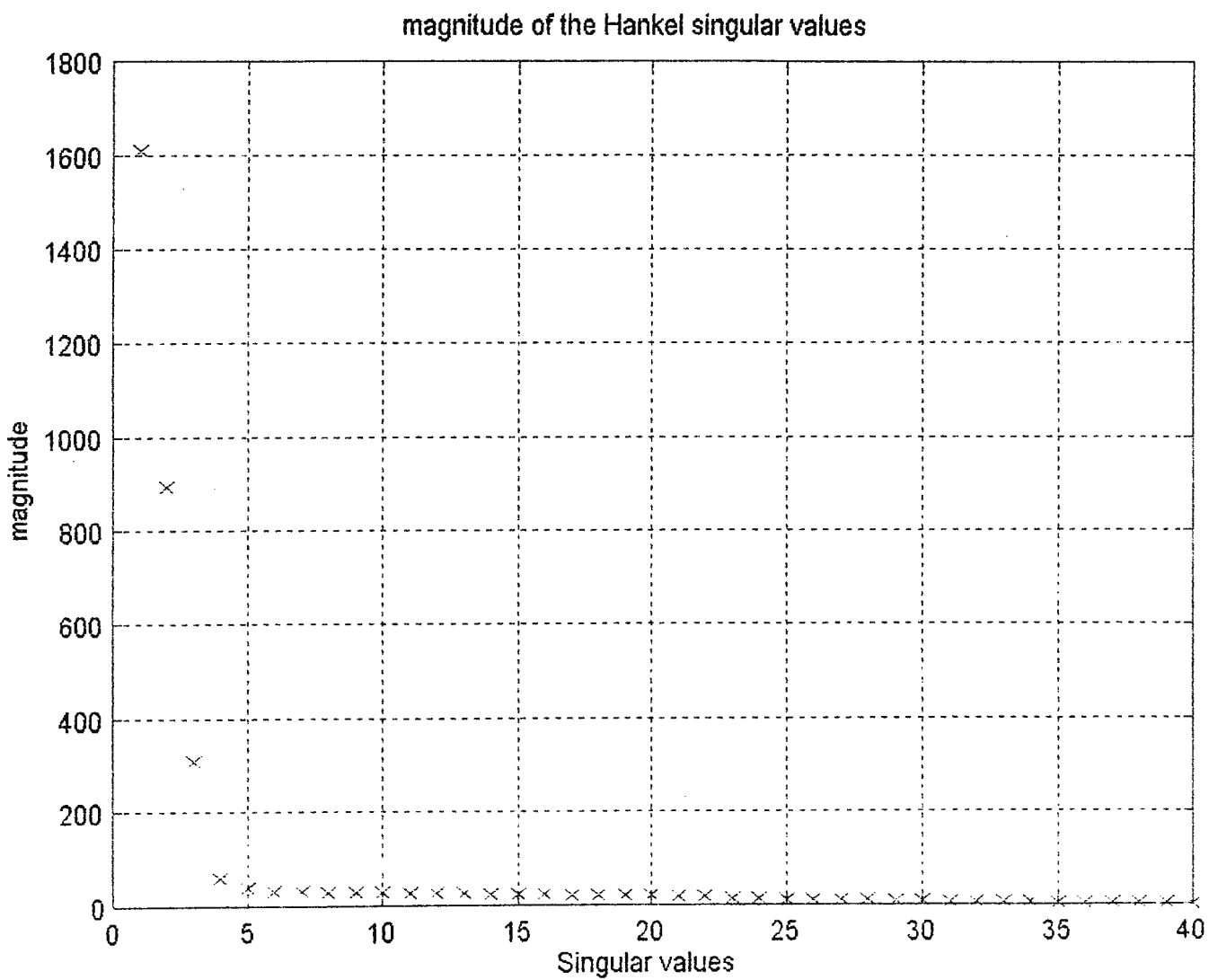


Figura 4.4 Magnitud de los valores singulares de Hankel.

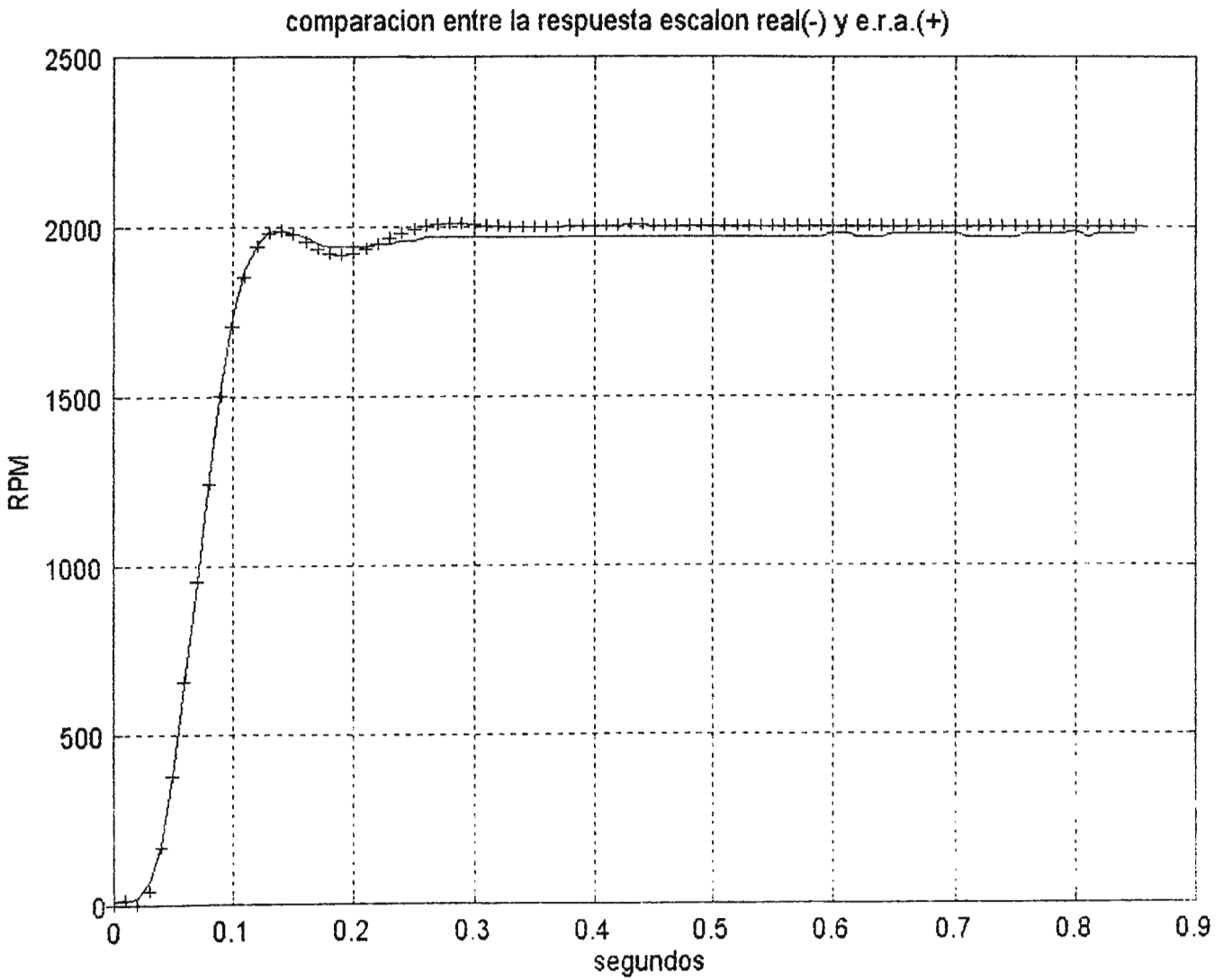


Figura 4.5 Comparación de la respuesta escalón y el modelo E.R.A.

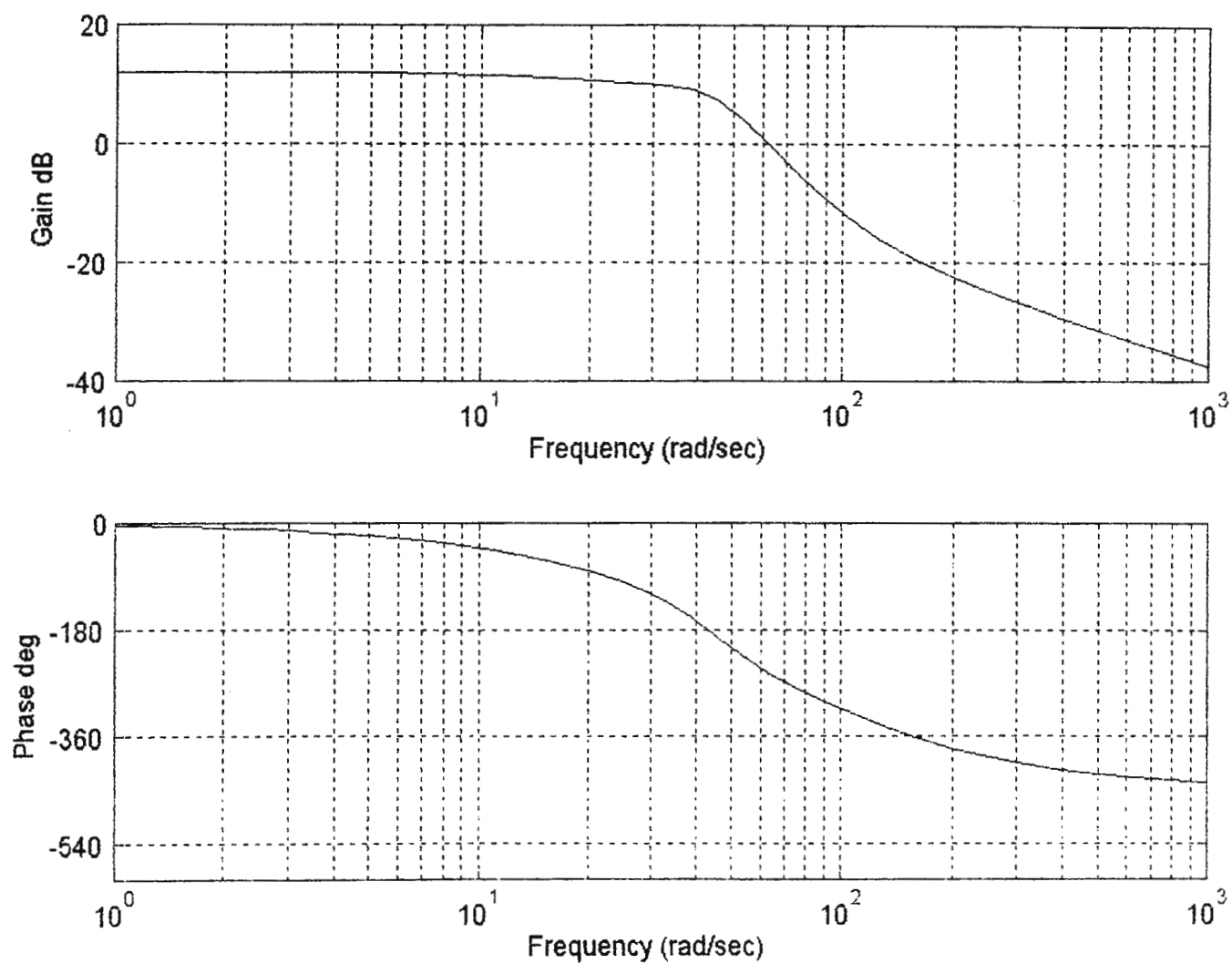


Figura 4.6 Respuesta en frecuencia del sistema de lazo abierto  $G(s)H(s)$ .

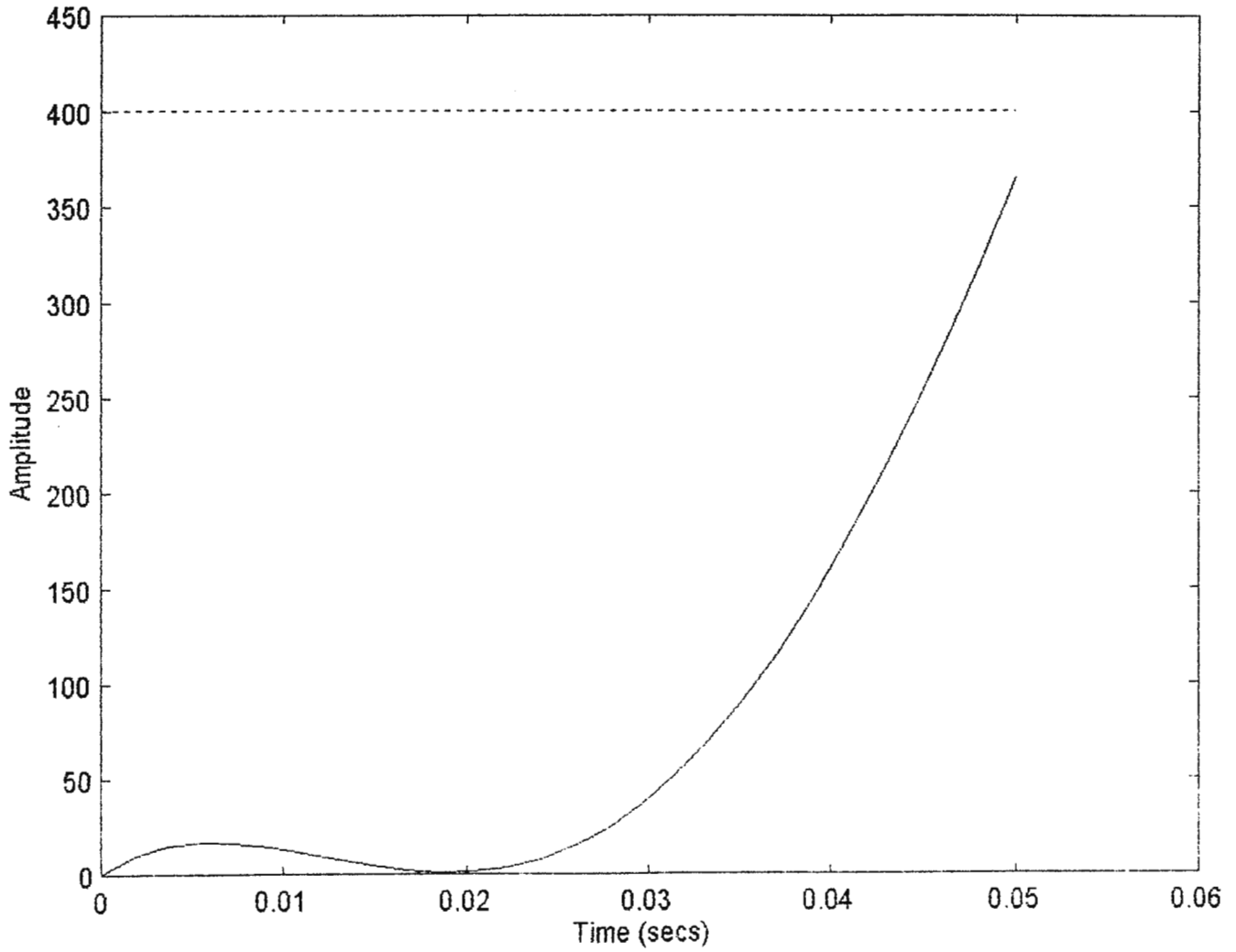
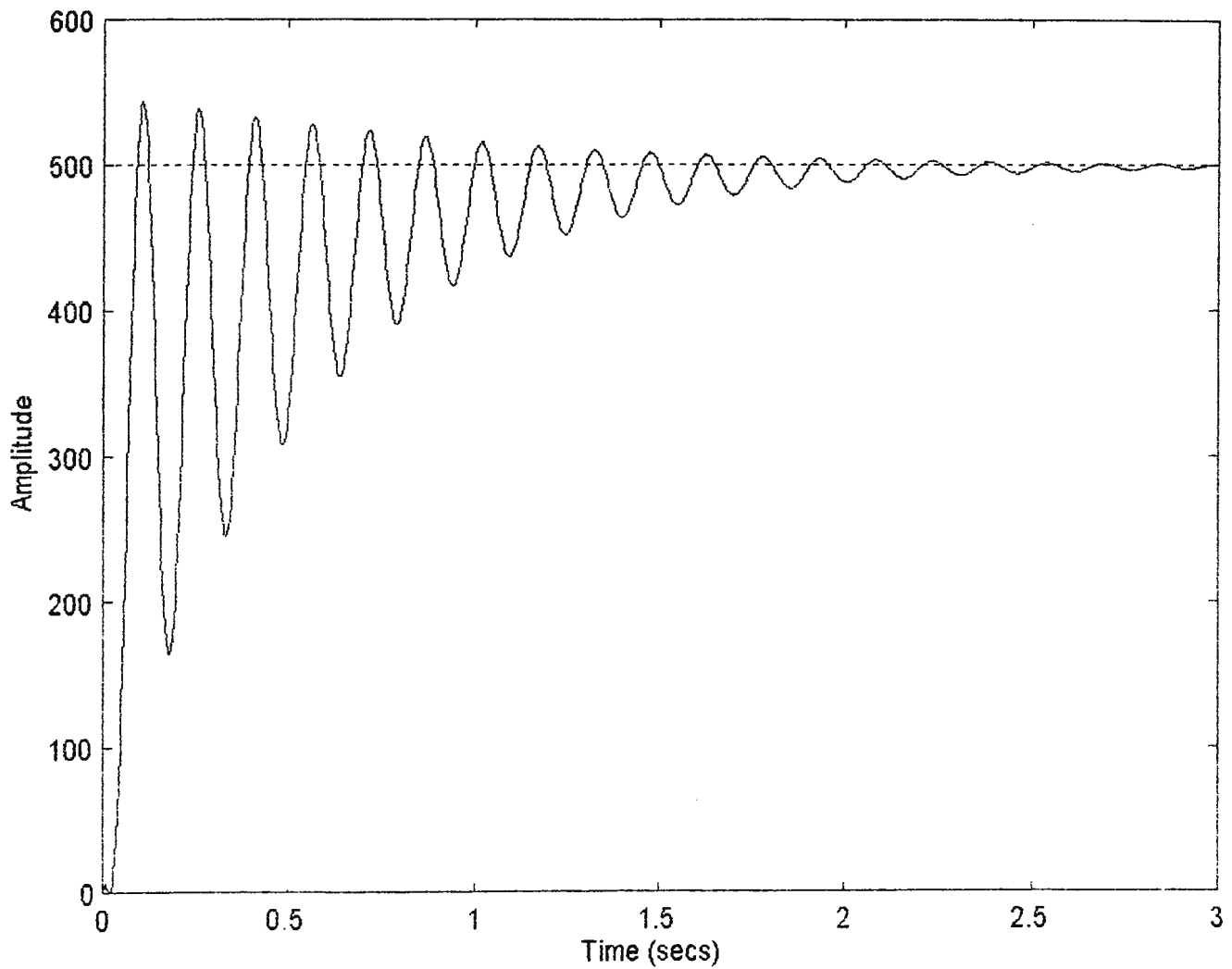


Figura 4.7 Respuesta escalón unitario del sistema de lazo cerrado.



**Figura 4.8 Respuesta al escalón unitario del sistema  
compensado con PI análogo.**

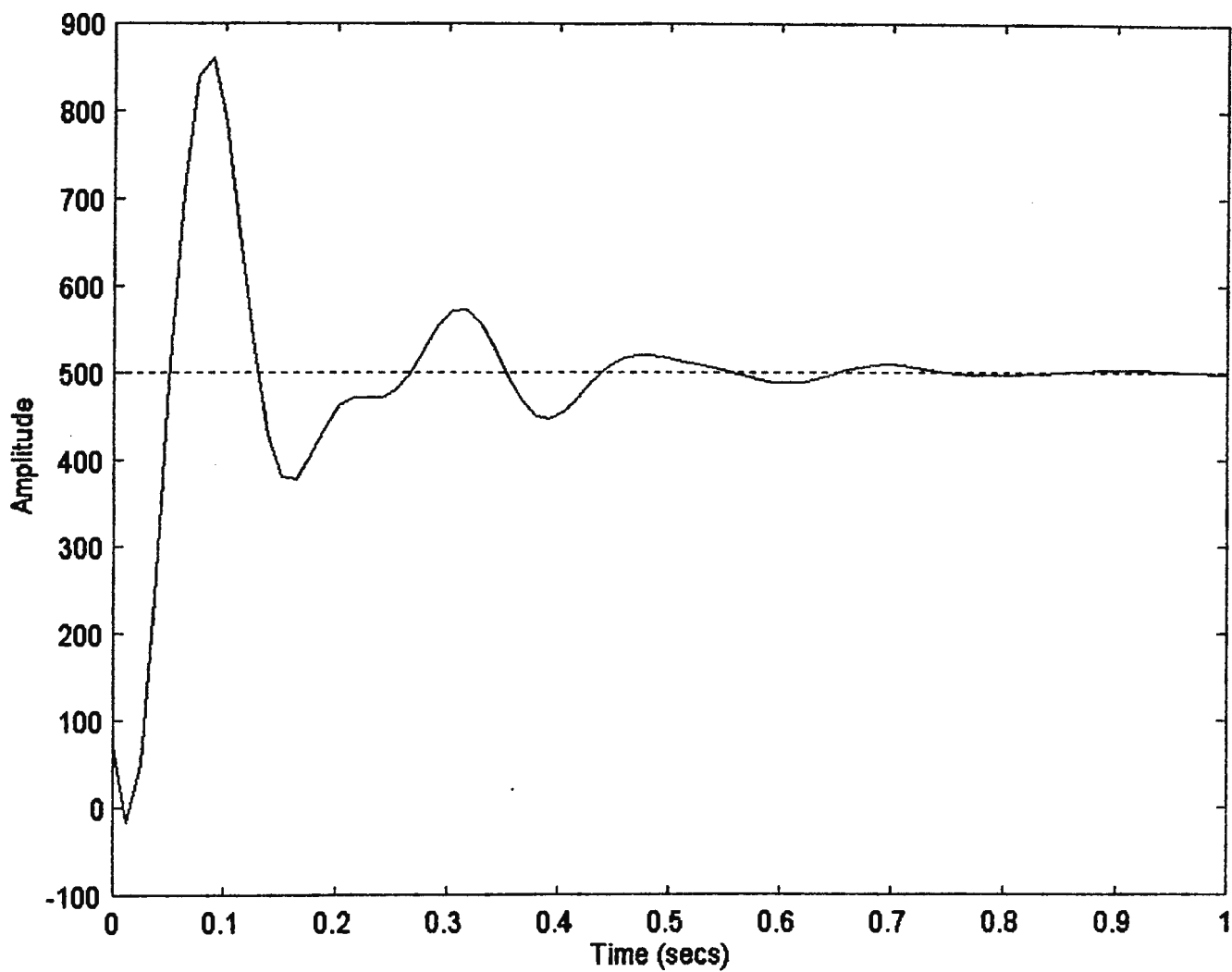


Figura 4.9 Respuesta al escalón unitario del sistema  
compensado con PID análogo.

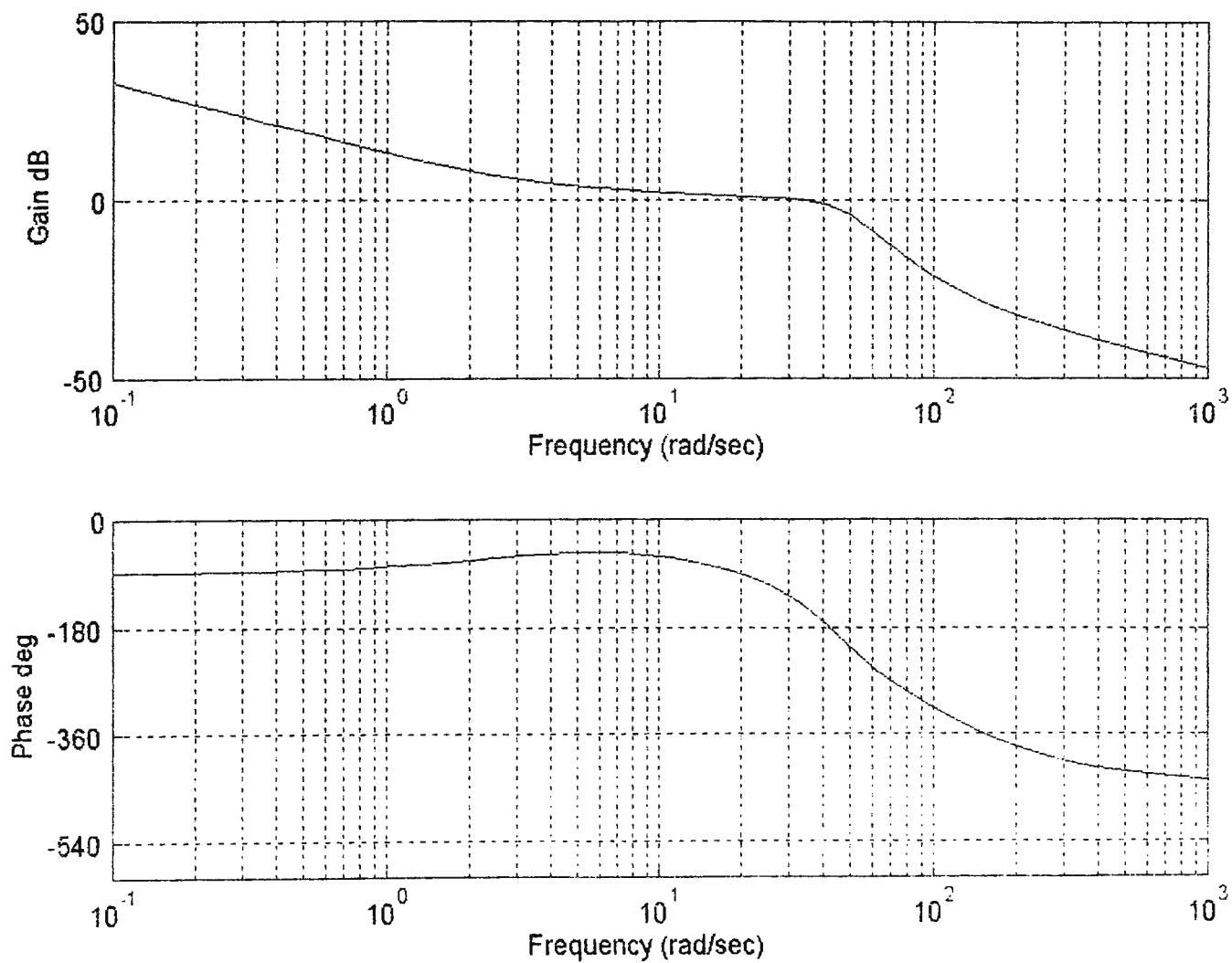


Figura 4.10 Respuesta en frecuencia de  $G_c(s)G(s)H(s)$   
con PI análogo.

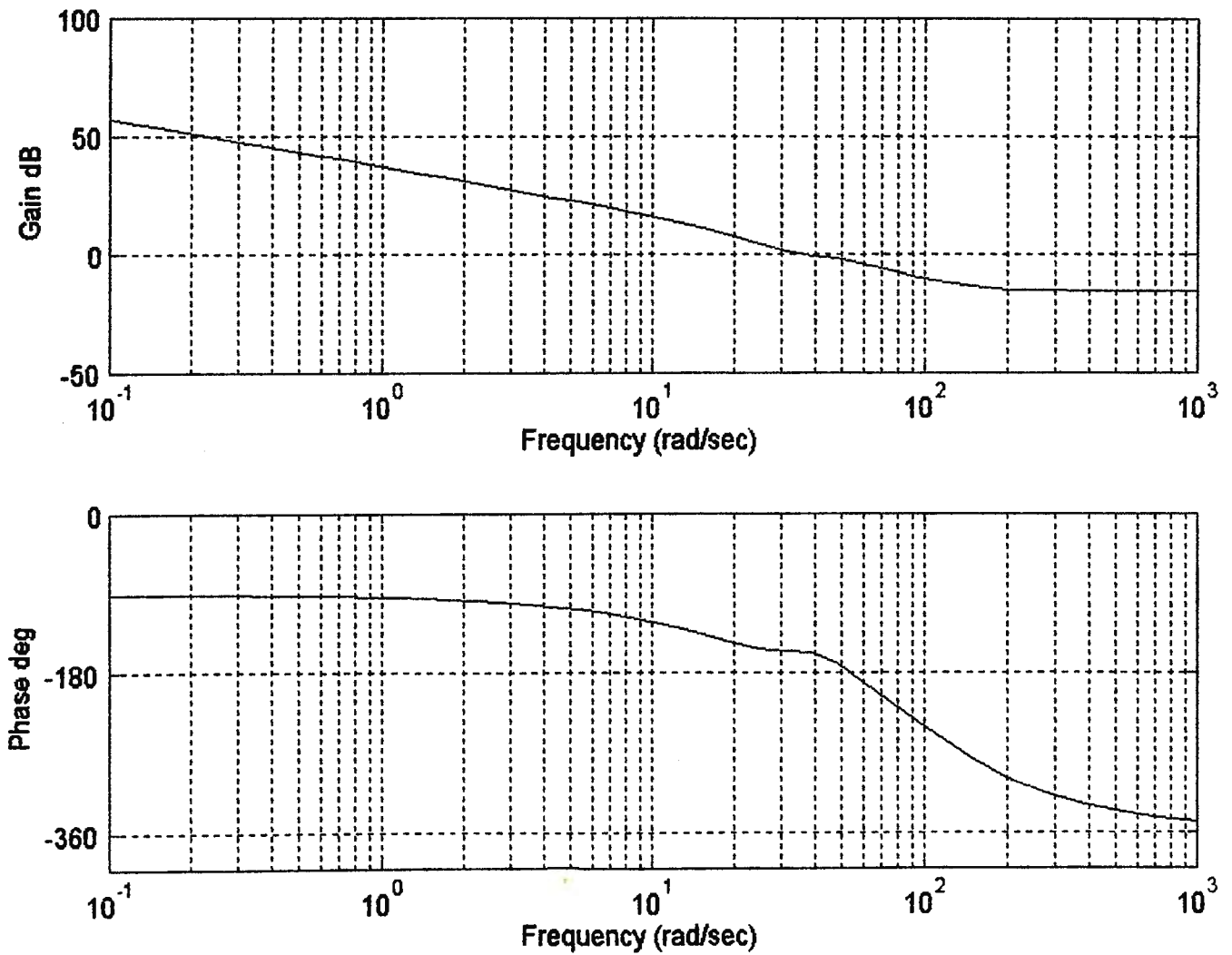


Figura 4.11 Respuesta en frecuencia de  $G_c(s)G(s)H(s)$

con PID analógico.



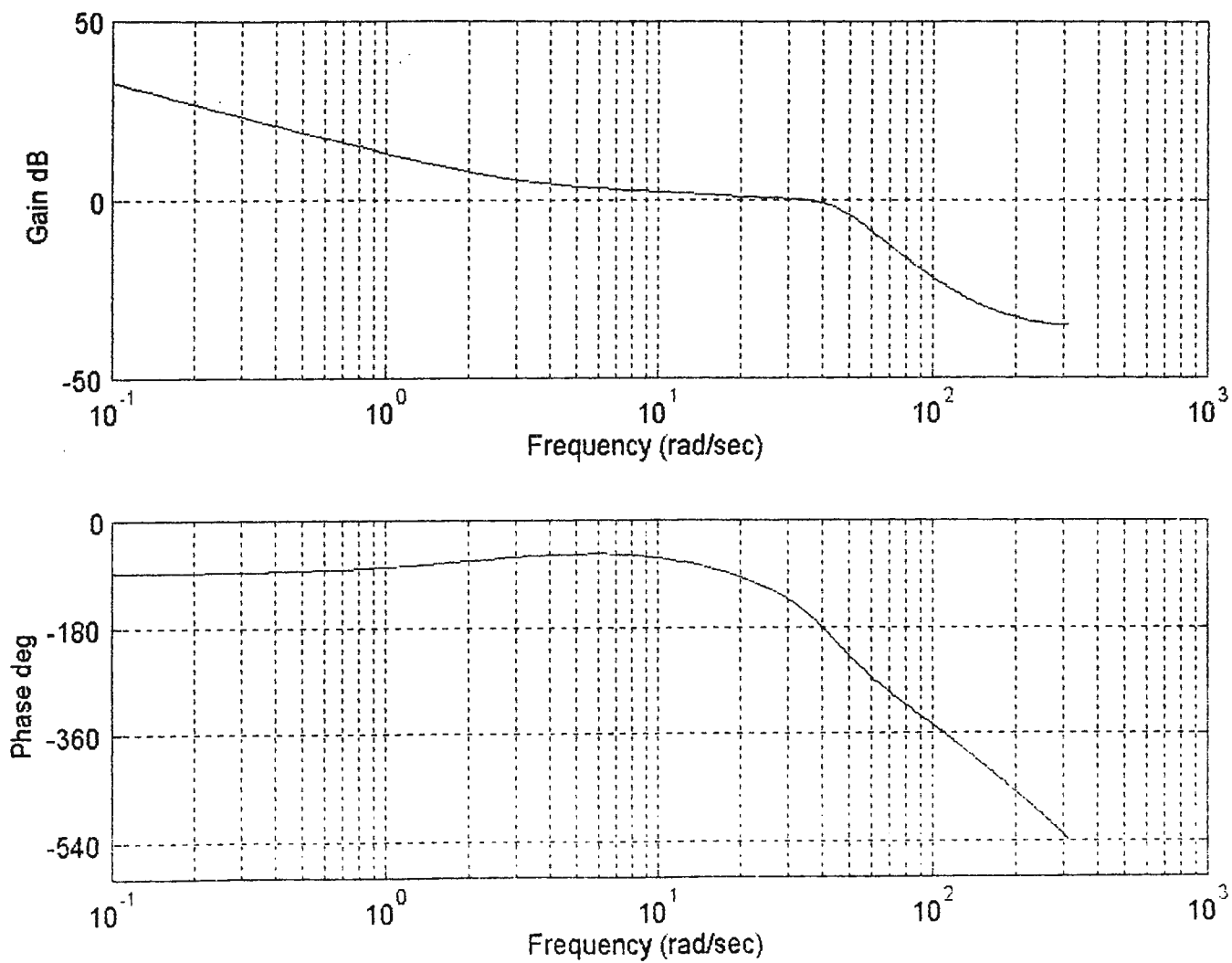


Figura 4.13 Respuesta en frecuencia de  $G_c(z)GH(z)$

con PI digital.

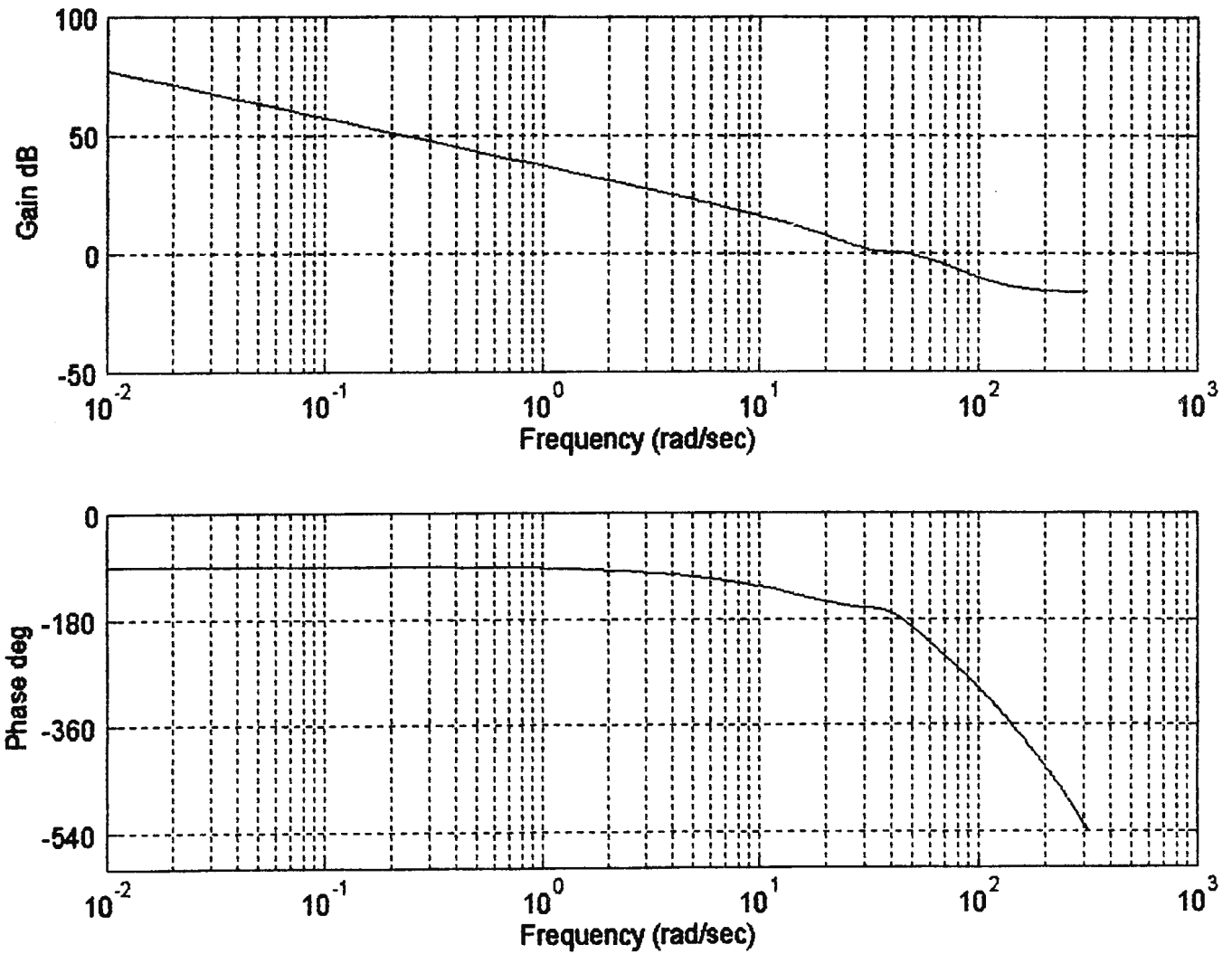


Figura 4.14 Respuesta en frecuencia de  $G_c(z)GH(z)$   
con PID digital.

## CONCLUSIONES Y RECOMENDACIONES

Con modelado con el algoritmo E.R.A. , se obtuvieron resultados satisfactorios, ya que los modelos obtenidos representan con una gran exactitud las dinámicas de los sistemas con los que se experimentó. El algoritmo E.R.A. resulta ser una poderosa herramienta para las realizaciones de sistemas industriales, y así poder describir el comportamiento de este sistema en simulaciones por computadora. Teniendo ya la descripción matemática del sistema, se procedió al análisis del sistema y al diseño de controladores digitales por medio de procedimientos analíticos que son imposibles de realizar sin un modelo matemático. Ahora la industria y las universidades cuentan con un equipo que es capaz de modelar cualquier sistema dinámico, siempre y cuando los sensores sean de señales normalizadas.

Se recomienda continuar con la investigación de nuevos algoritmos para el modelado matemático, o en todo caso, diseñar programas para computadora que ejecuten algoritmos ya conocidos además del E.R.A. con el fin de comparar los distintos métodos y así, seleccionar el que mejor se adapte a un sistema dado. Sería interesante conocer algoritmos usados en áreas que son muy delicadas tales como la industria militar y la espacial.

Por otro lado, es necesario saber como implementar controladores digitales en computadora, para luego colocarlo en línea con la planta a ser controlada y corroborar que

tan fiel es el modelo obtenido con el algoritmo de este proyecto, lo cual sería un paso más de la investigación científica en este país que tanto la necesita.

## APENDICE A

### DISEÑO DE CONTROLADORES ANALOGOS POR EL METODO DE RESPUESTA EN FRECUENCIA.

#### A.1 INTRODUCCION

El procedimiento considerado para el diseño de los controladores es el método de respuesta en frecuencia. Una ventaja que se tiene es que se puede determinar la estabilidad de un sistema en lazo cerrado desde un mapeo de la función en lazo abierto. Asumimos que el sistema ha sido convertido a un modelo con realimentación unitaria, teniendo como ecuación característica

$$1 + G_c(s)G_p(s) = 0 \quad (\text{A-1})$$

Teniendo la función de transferencia de la planta  $G_p(s)$ , se desea determinar la función de transferencia del compensador  $G_c(s)$  tal que la función de transferencia de lazo cerrado cumpla con las características que se desean.

Trabajando con sistemas de control, se debe de ser cuidadoso en no confundir una función de transferencia en lazo abierto (A-1) con la de lazo cerrado

$$T(s) = \frac{G_c(s) G_p(s)}{1 + G_c(s) G_p(s)} \quad (\text{A-2})$$

## A.2 COMPENSACION

En esta sección, básicamente se considerará la compensación para un sistema SISO (single-input, single-output), tal como se muestra en la Figura A.1. Este sistema tiene como ecuación característica de lazo abierto

$$1 + G_c(s)G_p(s)H(s) = 0 \quad (\text{A-3})$$

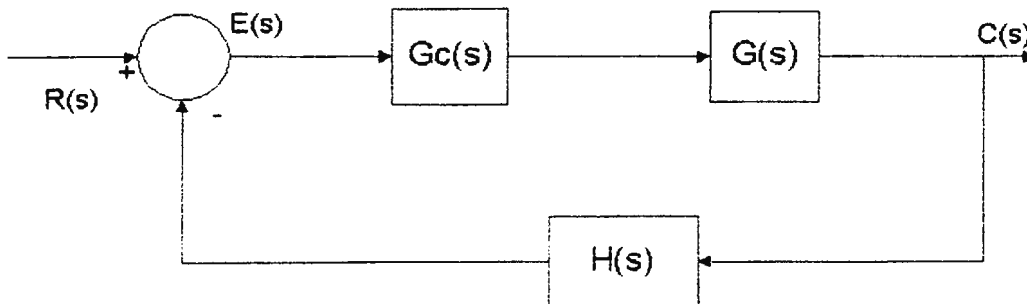


Figura A-1

Se desea diseñar el compensador  $G_c(s)$  tal que el sistema tenga ciertas características específicas. Los efectos de la compensación en las características del sistema son indicadas por la localización de las raíces en la ecuación característica (A-1).

### A.3 CONTROLADORES DE ATRASO DE FASE

En esta sección asumimos que el compensador es de primer orden y tiene como función de transferencia

$$G_c(s) = \frac{1 + s / \omega_0}{1 + s / \omega_p} \quad (\text{A-4})$$

Este compensador tiene ganancia de dc unitaria. El problema de diseño es por lo tanto determinar el cero del compensador,  $-\omega_0$ , y el polo del compensador,  $-\omega_p$ , de modo que el sistema de control de lazo cerrado tenga ciertas características especificadas.

Si  $\omega_0 > \omega_p$ , el compensador es llamado *compensador en adelanto de fase*; si  $\omega_0 < \omega_p$ , es llamado *compensador en atraso de fase*.

El diagrama de Bode del compensador en atraso de fase es mostrado en la Figura A-2.

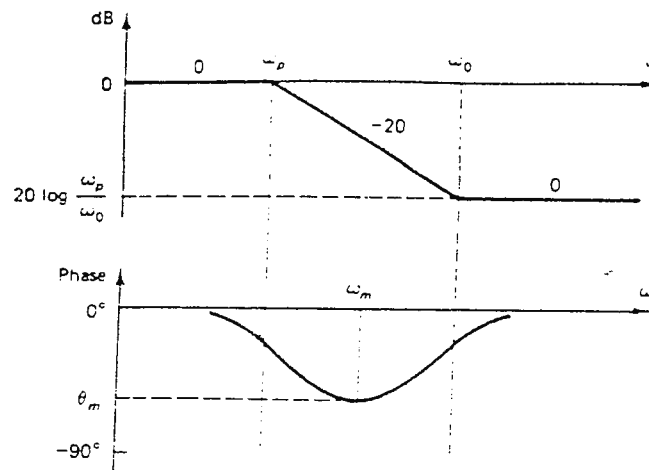


Figura A-2

Nótese que el efecto del compensador en atraso de fase es el de reducir la ganancia en alta frecuencia y de introducir un atraso de fase. Obsérvese también que de hecho este es una forma de filtro pasa-bajos; esto es, las altas frecuencias son atenuadas en relación con las bajas frecuencias. Este tipo de controladores generalmente reduce la ganancia lo cual tiene un efecto estabilizador en el sistema, y produce un atraso de fase lo cual genera un efecto desestabilizador en el sistema. Por lo tanto hay que ser cuidadoso en colocar el atraso de fase en un rango de frecuencia que no afecte la estabilidad del sistema.

El efecto de la compensación en atraso de fase es mostrado en el diagrama de Bode de la Figura A-3. Las magnitudes del cero y el polo del compensador son escogidos muy pequeños comparados con la frecuencia a la cual la magnitud de  $G_p(j\omega)H(j\omega)$  es la unidad.



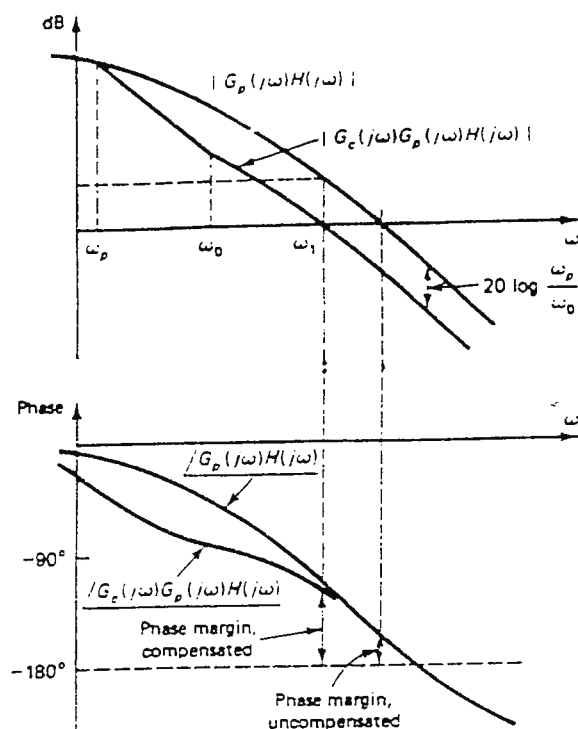


Figura A.3

El procedimiento para el diseño del controlador en atraso de fase es el siguiente:

- 1- Ajuste la ganancia dc de  $G_p(s)H(s)$  por el factor  $K$ , para satisfacer las especificaciones de baja frecuencia.
- 2- Determine la frecuencia  $\omega_1$ , en la cual el ángulo de fase de  $G_p(j\omega)H(j\omega)$  sea igual a  $(-180^\circ + \phi_m + 5^\circ)$ , donde  $\phi_m$  es margen de fase especificado.
- 3- La magnitud del cero es dada por

$$\omega_0 = 0.1\omega_1 \quad (\text{A-5})$$

4- La relación del polo y el cero del compensador esta dada por

$$\frac{\omega_p}{\omega_0} = \frac{1}{|K_c G_p(j\omega_1)H(j\omega_1)|} \quad (\text{A-6})$$

y

$$\omega_p = \frac{0.1\omega_1}{|K_c G_p(j\omega_1)H(j\omega_1)|} \quad (\text{A-7})$$

5- La función de transferencia del compensador es entonces

$$G_c(s) = \frac{K_c(1 + s/\omega_0)}{1 + s/\omega_p}$$

#### A.4 COMPENSACION EN ADELANTO DE FASE

Para el diseño del compensador en adelanto de fase se asumirá un compensador de ganancia dc unitaria de la forma

$$G_c(s) = \frac{1 + s/\omega_0}{1 + s/\omega_p} \quad (\text{A-8})$$

Para el compensador en adelanto de fase  $\omega_0 < \omega_p$ . Se desea determinar los valores de  $\omega_0$  y  $\omega_p$ , de modo que ciertos criterios de diseño sean satisfechos para el sistema de lazo cerrado.

El diagrama de Bode del controlador en adelanto de fase es de la forma de la Figura A-4.

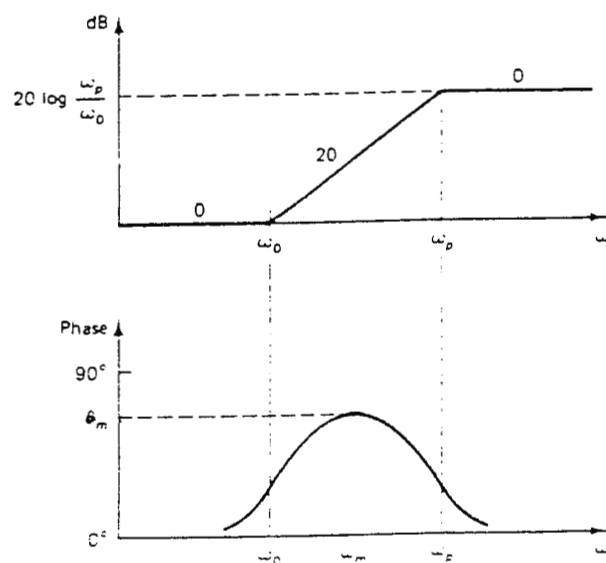


Figura A-4

Se puede observar que el compensador en adelanto de fase es una forma de filtro pasa alto, en el que las altas frecuencias son amplificadas relativas a las bajas frecuencias. El controlador introduce ganancia en altas frecuencias, lo cual en general es desestabilizador. Sin embargo, el ángulo de fase positivo del controlador tiene un efecto

estabilizador. De aquí que se ha de tener cuidado en escoger la localización del polo y el cero de modo que el efecto estabilizador domine.

## A.5 PROCESO DE DISEÑO DE CONTROLADORES EN ADELANTO DE FASE

Se comenzara parte del desarrollo con la ecuación característica del sistema

$$1 + G_c(s)G_p(s)H(s) = 0 \quad (\text{A-3})$$

y la función de transferencia del compensador.

$$G_c(s) = \frac{a_1s + a_0}{b_1s + 1} \quad (\text{A-9})$$

Nótese que la forma asumida del compensador es diferente al de la sección anterior, ya que no se esta limitando al compensador a tener ganancia de DC unitaria. El compensador se representa de la forma (A-9) para que la ganancia dc sea representada únicamente por el coeficiente  $a_0$ .

Las especificaciones del margen de fase  $\phi_m$  debe ser realizada a la frecuencia  $\omega_1$

$$G_c(j\omega_1)G_p(j\omega_1)H(j\omega_1) = 1\angle(-180^\circ + \phi_m) \quad (\text{A-10})$$

Luego se desea determinar los coeficientes  $a_1$  y  $b_1$  del compensador tal que (A-10) sea satisfecha para una ganancia específica dc del compensador.

De (A-9) y (A-10) tenemos

$$\begin{aligned} G_c(j\omega_1)G_p(j\omega_1)H(j\omega_1) &= \left[ \frac{a_0 + j\omega_1 a_1}{1 + j\omega_1 b_1} \right] G_p(j\omega_1)H(j\omega_1) \\ &= 1 \angle (-180^\circ + \phi_m) \end{aligned} \quad (\text{A-11})$$

Desarrollando (A-11) para  $a_1$  y  $b_1$  en términos de  $a_0$  y  $\omega_1$ , esto es conveniente para definir  $\theta$  que es el ángulo de  $G_c(j\omega_1)$ , tal que, para (A-11)

$$\theta = \arg G_c(j\omega_1) = -180^\circ + \phi_m - \arg G_p(j\omega_1)H(j\omega_1) \quad (\text{A-12})$$

despejando  $a_1$  y  $b_1$  de (A-11) tenemos

$$\begin{aligned} a_1 &= \frac{1 - a_0 |G_p(j\omega_1)H(j\omega_1)| \cos \theta}{\omega_1 |G_p(j\omega_1)H(j\omega_1)| \sin \theta} \\ b_1 &= \frac{\cos \theta - a_0 |G_p(j\omega_1)H(j\omega_1)|}{\omega_1 \sin \theta} \end{aligned} \quad (\text{A-13})$$

En un diseño dado, las especificaciones del error de estado estable pueden ser usadas para determinar la ganancia DC del compensador. Si se especifica un *settling time*  $T_s$ , puede ser expresado como

$$\omega_1 = \frac{8}{T_s \tan \phi_m} \quad (\text{A-14})$$

Hay que hacer notar que ésta ecuación es exacta solamente para sistemas standares de segundo orden, para sistemas de mayor orden es solamente una aproximación que podría crear inexactitud

En resumen, los pasos de diseño son los siguientes:

1. De las especificaciones de diseño, determinar el margen de fase  $\phi_m$ , la ganancia ( $a_0$ ) dc del compensador y el *settling time*  $T_s$ .
2. Con  $T_s$  y  $\phi_m$  usar (A-14) para calcular  $\omega_1$ , la frecuencia a la cual el margen de fase ocurre.
3. De (A-12), calcular  $\theta$ , el ángulo del compensador a  $\omega_1$
4. Desarrollar (A-13) para  $a_1$  y  $b_1$

Hay que hacer notar que el ángulo de fase del compensador debe ser positivo. Por lo tanto de (A-12),

$$\theta = -180^\circ + \phi_m - \arg G_p(j\omega_1)H(j\omega_1) > 0^\circ$$

o

$$\arg G_p(j\omega_1)H(j\omega_1) < -180^\circ + \phi_m \quad (\text{A-15})$$

También, la ganancia del compensador en adelanto de fase para  $\omega > 0$  es siempre mas grande que la ganancia dc, entonces  $G_c(j\omega_1) > a_0$ , y de (A-11)

$$\left| G_p(j\omega_1)H(j\omega_1) \right| < \frac{1}{a_0} \quad (\text{A-16})$$

En resumen, algunas posibles ventajas para el compensador en atraso de fase son:

1. Las características en baja frecuencia son mejoradas, en comparación con la ganancia de compensación.
2. Los márgenes de estabilidad son mejorados o mantenidos.
3. El ancho de banda es reducido, lo cual es una ventaja si el ruido de alta frecuencia es un problema.

Algunas posibles desventajas son:

1. La reducción del ancho de banda puede ser un problema en algunos sistemas.
2. La respuesta temporal del sistema podría ser lenta.

Para los compensadores de adelanto de fase, algunas posibles ventajas son:

1. Los márgenes de estabilidad son mejorados.
2. La velocidad de la respuesta temporal es mejorada.
3. Estos compensadores se requieren para estabilizar ciertos tipos de sistemas.

Algunas posibles desventajas son:

1. Cualquier problema de ruido de alta frecuencia será acentuado.
2. Grandes señales podrán ser generadas, los cuales podrían dañar el sistema o resultaría en una operación no-lineal del sistema.

## **A.6 CONTROLADORES PID**

Cuando se suma un controlador en atraso de fase con uno en adelanto de fase el resultado es un controlador de atraso-adelanto. El controlador en atraso-adelanto ofrece una mayor flexibilidad que si se usara uno en atraso y uno en adelanto separadamente. En esta sección se considerará una forma diferente de controlador atraso-adelanto. Ésta forma es probablemente la mas comúnmente empleada en sistemas de control de lazo cerrado y es usada mas exclusivamente en sistemas de control industrial.

El tipo de controlador presentado es un controlador PID. El diagrama de bloques dicho controlador es mostrado en la Figura A-5.



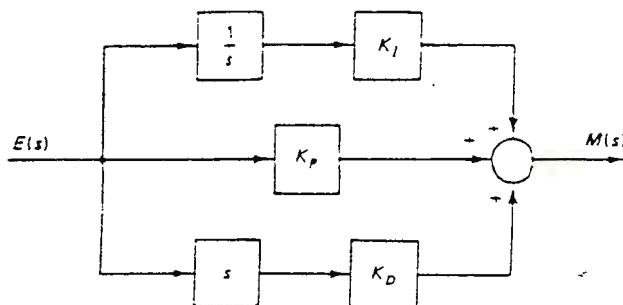


Figura A-5

La ecuación para el controlador es dado por

$$G_c(s) = K_p + \frac{K_i}{s} + K_d s \quad (\text{A-17})$$

Tres parámetros del controlador son determinados por el proceso de diseño: la ganancia proporcional  $K_p$ , la ganancia integral  $K_i$  y la ganancia derivativa  $K_d$ .

Si en la ecuación (A-17)  $K_d$  es igual a cero, tenemos un controlador PI. El procedimiento de diseño de la sección A.3 puede ser aplicado para el controlador PI con algunas pequeñas modificaciones. El procedimiento de diseño del controlador es el siguiente:

1. Ajuste la ganancia DC de  $G_p(s)H(s)$  por el factor  $K_c$  para satisfacer las especificaciones de baja frecuencia.
2. Encuentre la frecuencia  $\omega_1$  a la cual el ángulo de  $G_p(j\omega_1)H(j\omega_1)$  es igual a  $(-180^\circ + \phi_m + 5^\circ)$ .
3. La ganancia  $K_p$ , es entonces dada por

$$K_p = \frac{1}{|K_c G_p(j\omega_1)H(j\omega_1)|} \quad (\text{A-18})$$

- 4- La magnitud del cero es dada por

$$\omega_0 = \frac{K_1}{K_p} = 0.1\omega_1 \quad (\text{A-19})$$

$$K_1 = 0.1\omega_1 K_p = \omega_0 K_p$$

- 5- La función de transferencia del controlador es entonces

$$G_c(s) = K_c (K_p + K_i / s) \quad (\text{A-20})$$

Si en la ecuación (A-17)  $K_i$  es igual a cero el controlador es un PD. Ya que el compensador PD es en adelanto de fase, el procedimiento de diseño dado en la sección

A-5 puede ser empleado. Sin embargo, un procedimiento analítico es desarrollado en la siguiente sección.

## A.7 DISEÑO ANALÍTICO DEL CONTROLADOR PID

Primero se asume que la función de transferencia del controlador esta dada por:

$$G_c(s) = K_p + \frac{K_i}{s} + K_d s \quad (\text{A-21})$$

A la frecuencia  $\omega_1$  ocurre el margen de fase:

$$G_c(j\omega_1)G_p(j\omega_1)H(j\omega_1) = 1 \angle (-180^\circ + \phi_m)$$

Si el ángulo de  $G_c(j\omega_1)$  es denotado por  $\theta$ , entonces:

$$\theta = \arg G_c(j\omega_1) = -180^\circ + \phi_m - \arg G_p(j\omega_1)H(j\omega_1) \quad (\text{A-22})$$

De (A-21) y (A-22):

$$k_p + j(k_d \omega_1 - \frac{k_i}{\omega_1}) = |G_c(j\omega_1)|(\cos \theta + j \sen \theta) \quad (\text{A-23})$$

Donde:

$$|G_c(j\omega_1)| = \frac{1}{|G_p(j\omega_1)H(j\omega_1)|} \quad (\text{A-24})$$

Asumiendo que se conoce  $\omega_1$  entonces se conoce  $\theta$  y  $|G_c(j\omega_1)|$ . La frecuencia del margen de fase  $\omega_1$  podría calcularse especificando el tiempo de establecimiento  $T_s$ . En adición, la ganancia  $K_i$  debe escogerse para satisfacer las especificaciones de baja frecuencia, ya que a baja frecuencia, el controlador PID es dominado por el termino integral.

Primero, de (A-23) igualando las partes reales nos da

$$K_p = \frac{\cos\theta}{|G_p(j\omega_1)H(j\omega_1)|} \quad (\text{A-25})$$

e igualando las partes imaginarias nos da

$$K_d\omega_1 - \frac{K_i}{\omega_1} = \frac{\sin\theta}{|G_p(j\omega_1)H(j\omega_1)|} \quad (\text{A-26})$$

Las ecuaciones de diseño (A-25) y (A-26) son generales para aplicar a los controladores PID o cualquier variación de estos. Por ejemplo, las ecuaciones para el

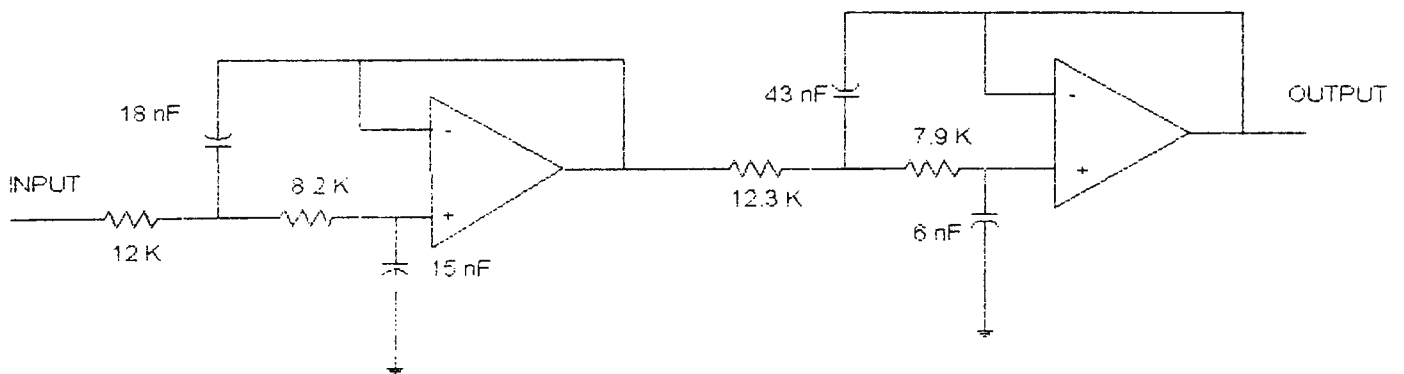
diseño de un controlador PD son obtenidas haciendo  $K_i = 0$  en (A-26). Entonces la ecuación para  $K_d$  es :

$$K_d = \frac{\sin\theta}{\omega_1 |G_p(j\omega_1) H(j\omega_1)|} \quad (A-27)$$

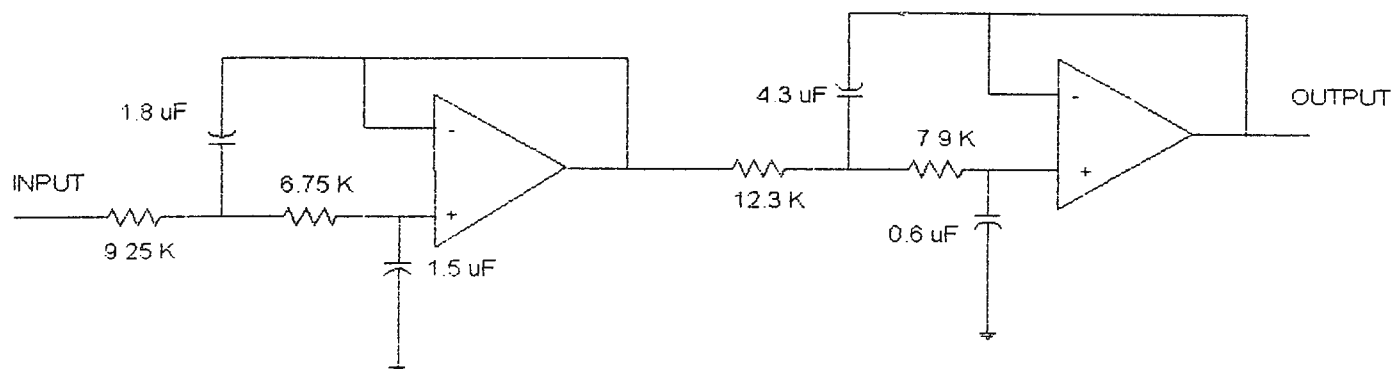
## APENDICE B

### FILTROS DISEÑADOS PARA LAS DISTINTAS CONSTANTES DE TIEMPO DE LOS SISTEMAS A MODELAR

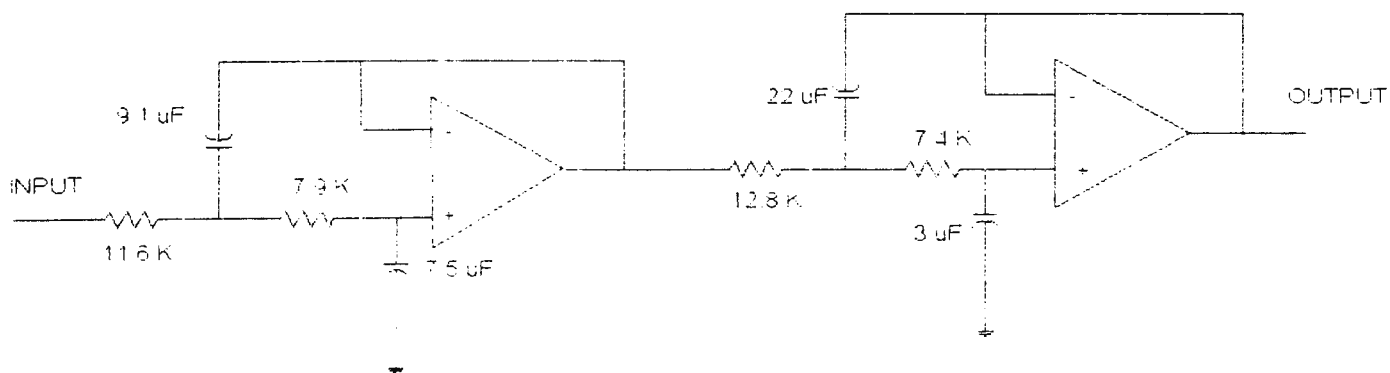
- PARA SISTEMAS DE 1mS A 100mS ( Frecuencia de corte de 1 KHz )



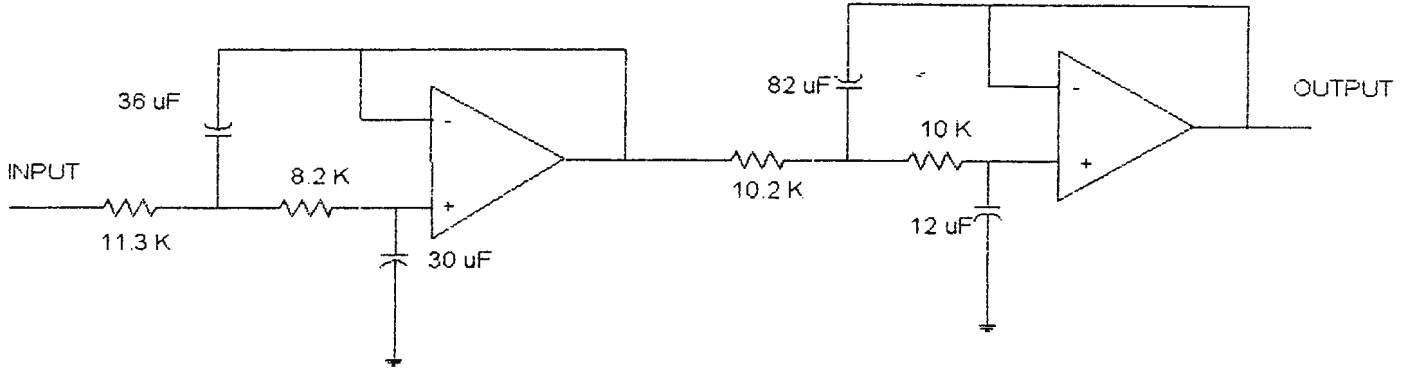
- PARA SISTEMAS DE 100 ms a 500 ms (Frecuencia de corte de 10Hz)



- PARA SISTEMAS DE 500 ms a 2s (Frecuencia de corte de 2 Hz)



- PARA SISTEMAS DE 2s A 10s (Frecuencia de corte 0.5 Hz)





## APENDICE C

### Programas en MATLAB que desarrollan el algoritmo ERA

#### “ERA.M”

```

% An m-file that implements the Eigensystem Realization Algorithm for the
% Identification of a Realization ( a,b,c) of a System from its impulse response.
% The Markov parameters of the system are entered as matrix Y, which has p rows
% (p=number of outputs) and nt*m columns ( m=number of inputs, nt= samples taken)
% For the r*p x s*m Hankel matrix H(k): r: row repetition ( number of blocks y()
% in a column). s: number of column blocks in a row ( number of samples per row).
% If norder is not given, it is asked for after the SVD of the Hankel matrix.
% SEE ALSO: HANKERA.M.
% Use as [ a,b,c]=era( y,m,r,s,norder);
% Enrique A. Medina. ECE Department. Ohio University. June, 1991.

```

```
function [ a,b,c]=era( y,m,r,s,norder)
```

```
[p,dummy]= size(y); % p=number of outputs of the system
```

```
% Form Hankel Matrices H(0) and H(1):
```

```
hank0=hankera(y,1,m,r,s);
```

```
hank1=hankera(y,2,m,r,s);
```

```
[mh,nh]= size(hank0);
```

```
disp([' Using ',int2str( mh),' by ',int2str( nh),' Hankel Matrix'])
```

```
% Singular Value Decomposition of H(0):
```

```
[pp,dd,qq]= svd(hank0);
```

```
hsv=diag( dd);
```

```
plot( hsv,'x'); title(' magnitude of the Hankel singular values'); grid
```

```

xlabel('Singular values'); ylabel(' magnitude'); meta thes217
pause
format short e

if nargin == 4

kk=input(' Please assign the desired order of the realization >');

else
kk=norder;
end

% Construct Realization of the Specified Order:

dd=diag( hsv(1:kk)); pp=pp(:,1:kk); qq=qq(:,1:kk); % Truncate SVD of H(0).
dtohalf=dd^(.5);
dtomhalf=dd^(-.5);
ept=[ eye(p) zeros(p,(r-1)*p)]; em=[ eye(m); zeros((s-1)* m,m)];

a=dtomhalf*pp'*hank1*qq*dtomhalf;
b=dtohalf*qq'*em;
c=ept*pp*dtohalf;

```

### "HANKERA.M"

% An m-file that forms the Hankel matrix  $H(k)$  given the Markov parameters  $y(k)$  of the system in matrix  $Y$  [ $p \times (nt \times m)$ ].

%  $p$ : number of outputs.  $m$ : number of inputs.  $nt$ : number of samples.

% The ( $r \times p \times s \times m$ ) Hankel matrix is formed as:

```
%
%           | y(k)   y(k+1) y(k+2) ... y(k+s-1) |           This Hankel
%           |                                     |           matrix is used
%           | y(k+1) y(k+2) y(k+3) ... y(k+s)   |           in the ERA
%   H(k) = | .       .       .                   .       |           system
%           | .       .       .                   .       |           identification
%           | .       .       .                   .       |           algorithm.
%           | y(k+r-1) y(k+r)   ... y(k+r+s-2) |
```

%

%  $s$ : number of column blocks  $y()$  in a row (number of samples in a row).

%  $r$ : number of blocks  $y()$  in a column (row repetition number).

% Use as  $h=hankera(y,k,m,r,s)$ ;

% Enrique A. Medina. ECE Department. Ohio University. June, 1991.

```
function h=hankera( y,k,m,r,s);
```

```
[p,ly]= size(y); % p: number of outputs of the system.
```

```
if ly < (s+r-1)*m;
```

```
    error(' Not enough samples in Y')
```

```
end
```

```
h=zeros( r*p,s*m);
```

```
for ii=1:r
```

```
h((ii-1)*p+1:ii*p,:)=y(:,(k+ii-2)*m+1:(k+ii-2+s)*m);
```

```
end
```

# APENDICE D

## MANUAL DE USO

### GUIA # 1

## ENTENDIENDO EL USO DEL ADQUISIDOR DE DATOS USANDO UN SISTEMA HIDRAULICO

### REQUERIMIENTOS DE HARDWARE Y SOFTWARE PARA LA ADQUISICION DE DATOS

- Una computadora IBM PC en la cual la señal de CLOCK en el bus del sistema sea de 8 MHz .
- Lenguaje de programación TURBO C.
- Programa MATLAB.

#### I) OBJETIVOS

- A) Entender el hardware de adquisición de datos.
- B) Entender el software de adquisición de datos.

## II) INTRODUCCION TEORICA

En la Figura 1 se observa el diagrama de bloques del circuito adquisidor de datos, el cual nos muestra una operación general del circuito.

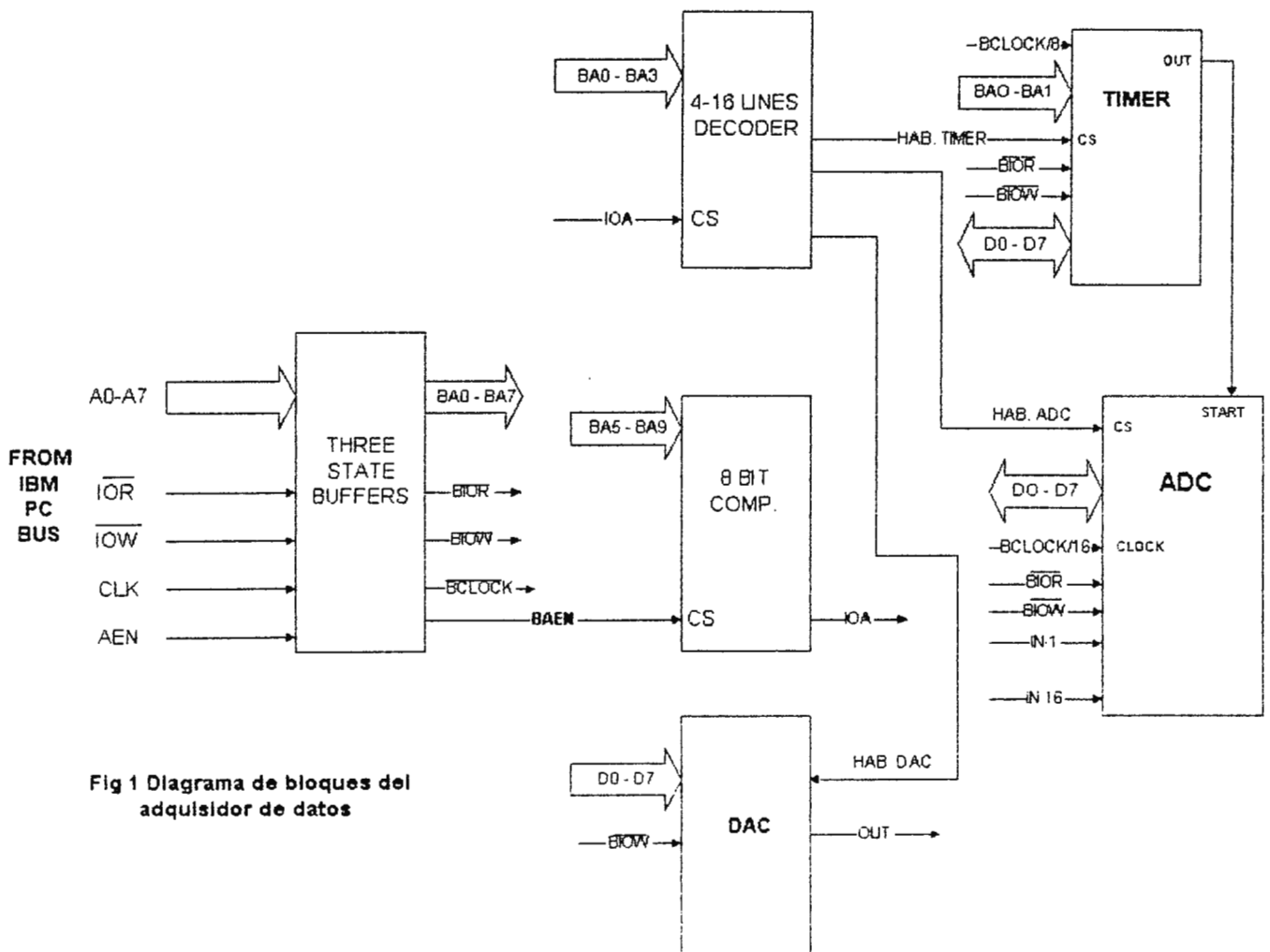


Fig 1 Diagrama de bloques del adquisidor de datos

- Las líneas de direcciones, datos y control se toman del bus IBM PC.
- Las líneas de direcciones y control son amplificadas por medio de buffers de tres estados para que puedan ser manejadas por los distintos dispositivos.
- El comparador de 8 bits utiliza el bus de direcciones y una palabra de control cargada para habilitar al decodificador de 4-16 líneas el cual a la vez, se encarga de habilitar a los restantes dispositivos, dependiendo de la dirección de puerto de salida seleccionada.
- El circuito timer, genera los pulsos necesarios para la entrada START del convertidor análogo-digital, por lo que, este circuito determina la frecuencia de muestreo a utilizar.
- El convertidor análogo-digital consta de 16 entradas analógicas lo cual lo hace muy versátil.
- El sistema también consta de un convertidor digital-análogo para poder generar una señal de salida en un momento dado.
- Nótese la división del reloj del sistema en los circuitos timer y convertidor análogo-digital, lo cual es ejecutado por un circuito divisor de frecuencia, con el fin de proporcionar la frecuencia específica de operación a cada circuito integrado.

Un sistema hidráulico es aquel en el cual un líquido fluye a través de bombas, tanques y otros componentes asociados. En la mayoría de sistemas hidráulicos, la fuente de energía es una bomba. La velocidad de la bomba es gobernada por la variación de corriente de armadura del motor DC, hasta que la razón de flujo de salida deseada es obtenida.

El sistema a usar para este experimento es mostrado en la Figura 2.

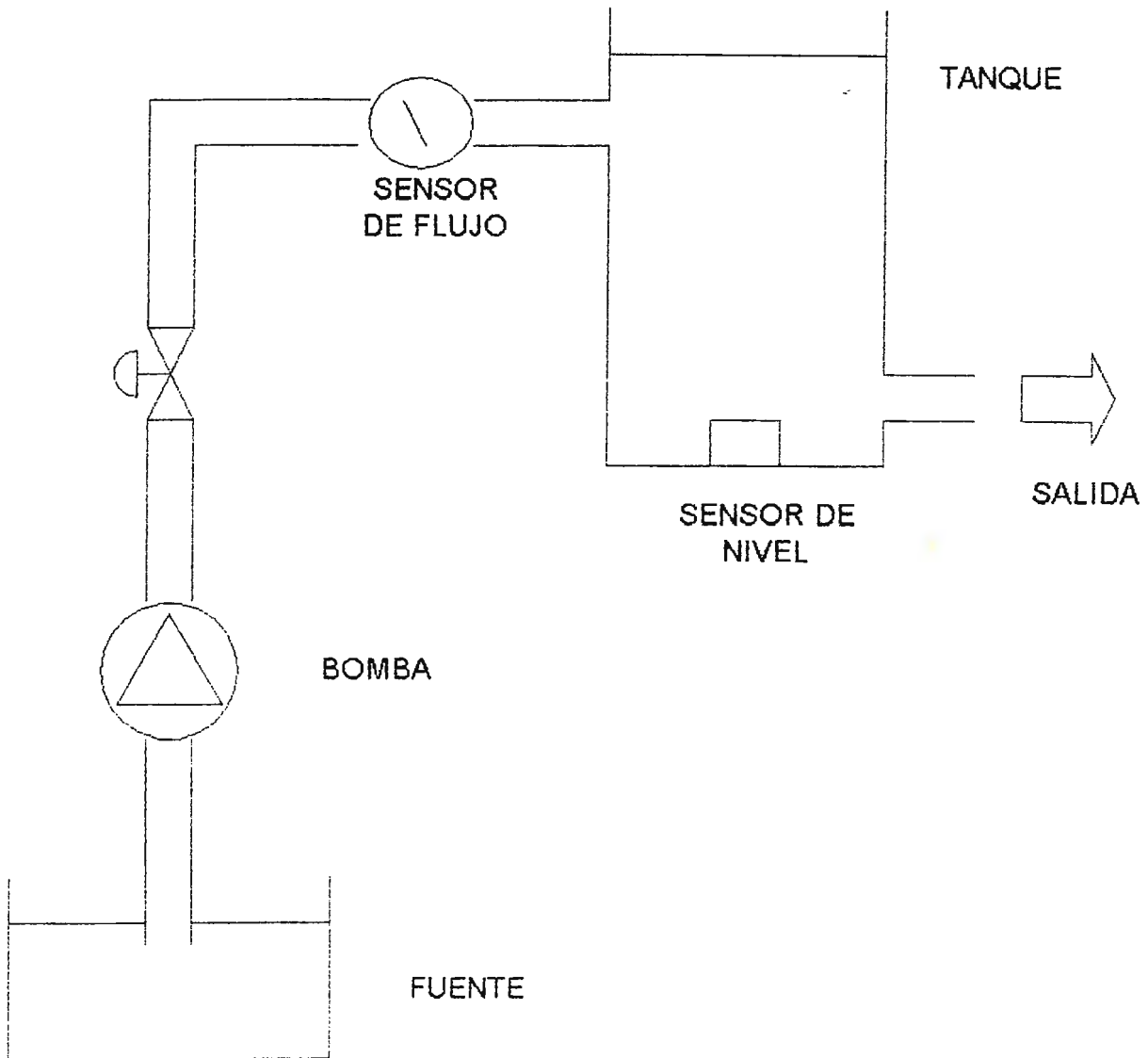


Figura 2 Sistema hidráulico



Para mostrar el funcionamiento del adquisidor de datos en este experimento al sistema se le aplicará una señal escalón. Antes de aplicar una señal escalón, se deben de realizar mediciones experimentales de la relación entrada-salida del sistema para determinar la región lineal de las variables de entrada-salida. En este experimento la variable de entrada será el voltaje aplicado a la bomba y la señal de salida es flujo.

La prueba será en lazo abierto y la Figura 3 muestra la conexión que deberá ser realizada.

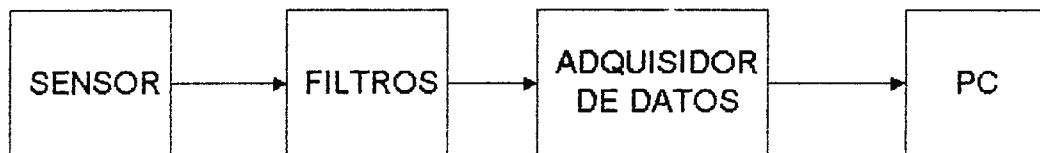


Figura 3

### III) PROCEDIMIENTO

INSTRUCCIONES	COMENTARIOS
<p><b>PARTE I: CONEXIÓN Y ELECCION DE LOS COMPONENTES DEL SISTEMA.</b></p>	
<p>1) Refiérase al manual del modulo G30B/EV (*1) para conectar el sistema hidráulico en lazo abierto.</p>	<p>(*1) Modulo hidráulico ubicado en el Laboratorio de Automatización y Control de la Universidad Don Bosco.</p>
<p>2) Abra la válvula de escape del liquido totalmente, para vaciar el contenido del tanque.</p>	
<p>3) Ya que la entrada es el voltaje aplicado a la bomba, aplique un escalón al sistema simplemente encendiendo el equipo y espere a que el sistema empiece a estabilizarse, esto es para determinar la constante de tiempo (<math>\tau</math>) del sistema.</p>	
<p>4) Con la constante de tiempo encontrada, elija el tiempo de muestreo (<math>T_s</math>) y el filtro a utilizar (*2).</p>	
<p>5) Conecte el adquisidor a la computadora y al sistema hidráulico según lo muestra la Figura 3 (*3).</p>	<p>(*2) Utilice el criterio de <math>T_s = 10 \tau</math></p>
<p><b>PARTE II. DETERMINACION DE LA ZONA LINEAL</b></p>	
<p>1) Aplique voltaje a la bomba gradualmente y llene la Tabla 1.</p>	<p>(*3) Seleccione el sensor de flujo.</p>

<b>VOLTIOS</b>	<b>FLUJO (Lt/min.)</b>
0	
0.5	
1	
1.5	
2	
2.5	
3	
3.5	
4	
4.5	
5	
5.5	
6	
6.5	
7	
7.5	
8	
8.5	
9	
9.5	
10	
10.5	
11	
11.5	
12	
12.5	
13	
13.5	
14	
14.5	
15	
15.5	
16	

Tabla 1

## INSTRUCCIONES

## COMENTARIOS

2) Grafique los datos obtenidos y seleccione la zona lineal aproximada para la aplicación del escalón.

3) De la zona lineal seleccione los dos voltajes que se aplicaran a la bomba para la simulación del escalón (\*1).

- estos dos voltajes causarán un determinado flujo, con un tester mida en el sensor que voltaje de salida corresponde a estos dos flujos para asegurarse que el voltaje de salida del sensor no sobrepase el voltaje máximo de entrada del adquisidor que es de 5V.

4) Con las dos fuentes y el interruptor conecte el circuito que se muestra en la Figura 4.

(\*1) Para simular este escalón se necesitaran dos fuentes de voltaje y un interruptor

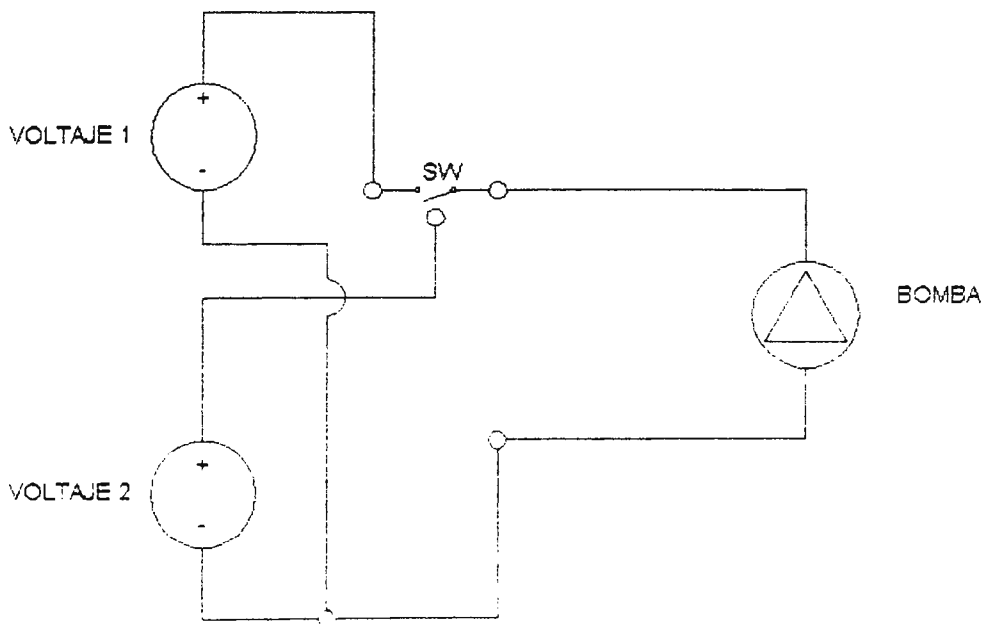


Figura 4

INSTRUCCIONES	COMENTARIOS
<b>PARTE III: USO DEL SOFTWARE.</b>	
1) Encienda la computadora.	
2) Alimente el sistema hidráulico con el voltaje 1 obtenido de la linealización.	
3) Ingrese a MS-DOS e ingrese al directorio TURBO C <ul style="list-style-type: none"> <li>• corra el programa "Adqui.c".</li> <li>• el programa preguntará "período de muestreo?", en este momento ingrese el valor (*1) y luego presione la tecla 'enter' (*2).</li> </ul>	(*1) Deberá de ingresarse números enteros, seguido de la unidad respectiva ejemplo: si el tiempo es de 1 milisegundo se deberá escribir "1m".
4) Cuando el display numérico en el sistema hidráulico muestre un valor estable, detenga el programa (*3).	(*2) El programa debe de correrse al mismo tiempo en el que se acciona el interruptor para causar el escalón.
5) Salga del directorio TURBO C e ingrese a Windows, luego corra el programa MATLAB.	
6) Corra el programa "muestreo.m" digitando "muestreo".	
7) El programa pedirá el período de muestreo, y los límites superior e inferior de la escala 'Y', en este caso el flujo resultante, introduzca los valores y presione la tecla 'enter'.	(*3) Los datos quedarán almacenados en el archivo "matriz.dat".
8) El programa presentará el gráfico resultante de la prueba.	
9) Salga del programa, apague la computadora y el equipo hidráulico.	

## GUIA #2

### MODELADO DE SISTEMA HIDRAULICO UTILIZANDO EL ALGORITMO ERA

#### I) OBJETIVOS

- A) Utilizar los resultados del experimento #1 para modelar el sistema.
- B) Simular respuestas del sistema con controladores diseñados.

#### II) INTRODUCCION TEORICA

El algoritmo ERA (Eingensystem Realization Algorithm) usa una matriz de *Hankel*, construida de los parámetros de Markov del sistema. Los parámetros de Markov definen la respuesta al pulso. Por medio de la Descomposición de Valores Singulares (SVD), una realización (un modelo) de orden igual al rango de ésta matriz puede ser encontrado. Para poder obtener una mínima realización el ruido debe de ser eliminado.

Las ecuaciones en el espacio de estado en tiempo discreto son escritas usualmente en la forma siguiente:

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}u(k)$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k)$$

En sistemas de tiempo discreto, los parámetros de Markov son justamente los valores de la respuesta al pulso discreto.

Los parámetros de Markov pueden ser obtenidos experimentalmente para formar la matriz de Hankel

$$H(k) = \begin{bmatrix} h(k) & h(k+1) & \dots & h(k+s-1) \\ h(k+1) & h(k+2) & \dots & h(k+s) \\ \cdot & \cdot & \dots & \cdot \\ \cdot & \cdot & \dots & \cdot \\ \cdot & \cdot & \dots & \cdot \\ h(k+r-1) & h(k+r) & \dots & h(k+r+s-2) \end{bmatrix}$$

donde  $r$  y  $s$  deben ser escogidas lo suficientemente grande de modo que el rango de la matriz no sea incrementado por los incrementos de  $r$  y  $s$ .

Defínase una matriz de controlabilidad  $\mathbf{P}$ , una matriz de observabilidad  $\mathbf{Q}$  y matrices  $\mathbf{E}_b$  y  $\mathbf{E}_m$  de acuerdo a

$$\mathbf{P} = [\mathbf{B}, \mathbf{AB}, \mathbf{A}^2\mathbf{B}, \dots, \mathbf{A}^{s-1}\mathbf{B}],$$

$$\mathbf{Q} = [\mathbf{C}^T, (\mathbf{CA})^T, \dots, (\mathbf{CA}^{r-1})^T],$$

$$\mathbf{E}_\beta^T = [\mathbf{I}_\beta, \mathbf{0}_\beta, \dots, \mathbf{0}_\beta]^T,$$

$$\mathbf{E}_m^T = [\mathbf{I}_m, \mathbf{0}_m, \dots, \mathbf{0}_m]^T,$$

donde  $\beta$  y  $m$  son las dimensiones de el vector de medición  $y(k)$  y el vector de control  $u(k)$ .

$\mathbf{I}_m$ , e  $\mathbf{I}_\beta$  son matrices de identidad de orden  $\beta$  y  $m$ , y  $\mathbf{0}_\beta$  y  $\mathbf{0}_m$  son matrices cero  $\beta \times \beta$  y  $m \times m$ .

Se puede probar que el orden mínimo de cualquier realización de un sistema es igual al rango de  $H(0)$  [2]. Una realización mínima puede ser encontrada usando SVD de  $H(0)$ :

$$H(0) = \mathbf{U}_N \mathbf{S}_N \mathbf{V}_N^T$$

donde:

$$\mathbf{S}_N = \text{diag}(d_1, d_2, \dots, d_N)$$



y  $d_1$  hasta  $d_N$  son los valores singulares de la matriz de Hankel.

Ruido y no linealidades pueden hacer que el rango de la matriz de Hankel sea más grande que el orden del sistema. En [6] se establece que el orden de la realización puede ser aproximado examinando los valores singulares de la matriz de Hankel  $H(0)$ . Si los valores singulares  $d_{n+1}, d_{n+2}, \dots, d_N$  son muy pequeños comparados con  $d_1, \dots, d_n$ ,  $H(0)$  puede ser asumido de rango  $n$ , y la realización mínima puede ser de rango  $n$ , en otras palabras:

$$H(0) = \mathbf{U}_n \mathbf{S}_n \mathbf{V}_n^T$$

donde  $\mathbf{S}_n = \text{diag}(d_1, \dots, d_n)$ , y  $\mathbf{U}_n$  y  $\mathbf{V}_n$  son obtenidas al truncar las últimas  $N-n$  columnas de  $\mathbf{U}_N$  y  $\mathbf{V}_N$  respectivamente. Se puede probar que una realización puede ser obtenida de la manera siguiente:

$$\mathbf{A} = \mathbf{S}_n^{-1/2} \mathbf{U}_n^T \mathbf{H}(1) \mathbf{V}_n \mathbf{S}_n^{-1/2}, \quad \mathbf{B} = \mathbf{S}_n^{-1/2} \mathbf{V}_n^T \mathbf{E}_m, \quad \mathbf{C} = \mathbf{E}_p \mathbf{U}_n \mathbf{S}_n^{1/2}$$

Las ecuaciones de estado obtenidas, están en forma discreta. Todas las realizaciones derivadas por el ERA son transformadas a realizaciones equivalentes en tiempo continuo usando la función en MATLAB™ "d2c" [8]. Después que la realización en forma de ecuaciones de estado continuas son encontradas, una función de transferencia en términos de numerador y denominador es obtenida usando la función de MATLAB™ "ss2tf" [8]. En el apéndice C se muestran los programas en MATLAB™ que ejecutan el algoritmo ERA. El programa "Hankera.m" forma la matriz de Hankel  $H(k)$  dados los parámetros de Markov  $Y(k)$  del sistema. El programa "era.m" es el que implementa el algoritmo ERA para la identificación de una realización (A,B,C).

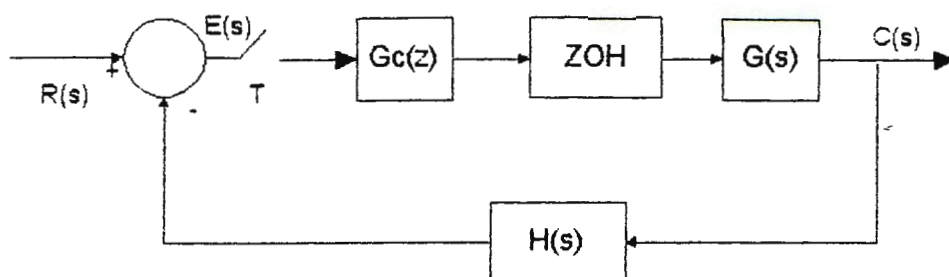
## II) PROCEDIMIENTO

INSTRUCCIONES	COMENTARIOS
<b>PARTE I: MODELADO CON ERA</b>	
1) Encienda la computadora y bajo ambiente Windows entre al programa MATLAB.	
2) Corra el programa "modelo.m" digitando 'modelo'.	
3) El programa le preguntara "período de muestreo utilizado?", en este momento introdúzcalo (*1) y presione la tecla 'enter'.	(*1) Este período de muestreo debe ser el mismo utilizado para la adquisición de datos.
4) El programa mostrará la respuesta impulso del sistema, presione nuevamente la tecla 'enter'.	
5) El programa mostrará ahora los valores singulares de la matriz de Hankel, luego tendrá que introducir el orden del modelo requerido (*2).	*2) El orden será igual a la cantidad de valores singulares de mayor peso
6) La siguiente gráfica mostrara la respuesta al escalón unitario del modelo obtenido con el algoritmo ERA, presione nuevamente la tecla 'enter'.	
7) En la siguiente gráfica se observara la comparación del modelo experimental con el modelo ERA.	

INSTRUCCIONES	COMENTARIOS
<p>8) La siguiente gráfica mostrara el diagrama del lugar de las raíces, donde se muestran la distribución de los polos y ceros del sistema.</p> <p>9) Repita los pasos del 2 al 9 hasta que se obtenga el modelo satisfactorio.</p> <p>10) Digite "numc", presione la tecla 'enter' y el programa mostrará el numerador de la función de transferencia obtenida (*3).</p> <p>11) Digite "denc", presione la tecla 'enter' y el programa mostrará el denominador de la función de transferencia obtenida.</p>	<p>(*3) La función de transferencia obtenida (numc,denc) es analógica.</p>
<p><b>PARTE II. SIMULACION DEL SISTEMA EN LAZO CERRADO CON UN CONTROLADOR DIGITAL DISEÑADO.</b></p>	
<p>1) Con el modelo obtenido y los métodos de diseño mostrados en [1], diseñe un controlador PID digital de modo que el sistema presente un margen de fase (<math>\phi_m</math>) mayor de <math>55^\circ</math> y un tiempo de establecimiento menor a 1 segundo.</p> <p>2) Corra el programa "control.m" digitando 'control'.</p> <p>3) El programa pedirá el numerador y denominador del controlador digital diseñado, luego presione 'enter'.</p> <p>4) El programa mostrara una gráfica de la respuesta escalón unitario del sistema de lazo cerrado representado en la siguiente gráfica</p>	

## INSTRUCCIONES

## COMENTARIOS



- 5) La siguiente gráfica presentara el diagrama de Bode del sistema de lazo abierto  $G_c(z)\overline{GH}(z)$  y corrobore el margen de fase especificado en el paso 1.
- 6) Salga del programa y apague la computadora.

## **Bibliografía.**

- [1] Savant, Rodden, Carpenter, Electronic Design. Circuits and systems , The Benjamin/ Cummings Publishing, Inc., 1991.
- [2] J.G. Reid, Linear System Fundamentals , Mc-Graw Hill. 1983.
- [3] Katsuhiko Ogata", Modern Control Engineering . Prentice-Hall, 1993.
- [4] R. Chinchilla, " Design and Evaluation of Undergraduate Experiments Using the Bytronic Laboratory Set-Up", Master of Science Thesis, Ohio University, 1991.
- [5] T. Kailath, Linear Systems , Prentice-Hall, 1980.
- [6] E. Medina, " Multi Input, Multi-output System Identification from Frequency Responses Samples with Applications to the Modeling of Large Space Structures", Master of Science Thesis, 1991.
- [7] R. W. Longman and J.N. Juang, " Recursive Form of the Eigensystem Realization Algorithm for System Identification", AIAA Journal of Guidance and Control , 1989.
- [8] The Math Works Inc. 386-Matlab™ User's Guide , 1990.

- [9] M. Waite, S. Prata, C:Step-by-Step, Howard W. Sams & Company, 1989.
- [10] W. Tompkins, J. Webster, Interfacing Sensors to the IBM PC, Prentice Hall, 1988.
- [11] B. Kuo, Digital Control Systems, Saunders College Publishing, 1992.
- [12] P. Katz, Digital Control Using Microprocessors, Prentice Hall, 1981.
- [13] C. Phillips, R. Harbor, Feedback Control Systems, Prentice Hall, 1993.
- [14] C. Phillips, T. Nagle, Digital Control Systems, Prentice Hall, 1993.