

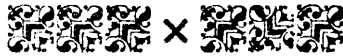
LOAE
C.2



Escuela Superior Politécnica del Litoral

FACULTAD DE INGENIERIA EN MECANICA Y CIENCIAS

DE LA PRODUCCION



“Actualización del Equipo Simulador de Procesos del Laboratorio de Sistemas de Control”

TESIS DE GRADO

Previa a la Obtención del **Título** de:

INGENIERO MECANICO

Presentada por:

FRANCIS RODERICH LOAYZA PAREDES



Guayaquil - Ecuador

1998



AGRADECIMIENTO



D-18646

ING. EDUARDO ORCES

Director de Tesis, por su
ayuda y colaboración para
la realización de este
trabajo.

DEDICATORIA

A mis padres FANNY
PAREDES y FRANCISCO
LOAYZA y a mis
HERMANAS.

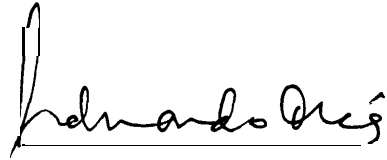
TRIBUNAL DE GRADUACION



ING. EDUARDO IVADENEIRA P.

3

DECANO DE LA FIM



ING. EDUARDO ORCES P.

DIRECTOR DE TESIS



ING. XAVIER HIDALGO

VOCAL



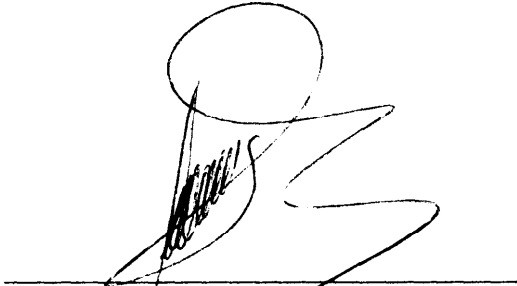
ING. FRANCISCO ANDRADE

VOCAL

DECLARACIÓN EXPRESA

“La responsabilidad del contenido de esta Tesis de Grado, me corresponde exclusivamente, y el patrimonio intelectual de la mismo a la ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL”

(Reglamento de graduación de la ESPOL)



Francis Loayza Paredes

RESUMEN

El presente proyecto consiste en la actualización del equipo simulador de procesos del laboratorio de sistemas de control PCS327 de propiedad de la FIM, el cual tenía un ineficiente sistema de visualización y registro de datos, lo que originaba su poco práctico uso como material didáctico para simulación de procesos de control.

Debido al escaso método didáctico utilizado para realizar la prácticas, generó la idea de la construcción de una tarjeta para adquisición de datos por computadora, de tal manera que la señales eléctricas producidas por el equipo durante el proceso de simulación, puedan ser graficadas en tiempo real en la pantalla de un computador y grabadas para su posterior análisis por cualquier método ya sea matemático o gráfico

La primera parte de esta tesis empieza por una descripción del equipo y sus partes así como su funcionamiento. Se continúa con los principios teóricos utilizados en las prácticas de laboratorio, los cuales consisten en las deducciones de las funciones de transferencia de los diferentes procesos y controladores que se pueden simular, así como las respuestas a las diversas entradas en varios puntos del sistema.

En el tercer capítulo se describe la lógica utilizada par la construcción de la tarjeta de interfase analógico - digital para la adquisición de datos así como sus componentes principales. Se habla también del programa desarrollado en "BASIC" para manejar la tarjeta por medio de un puerto paralelo del computador y también sobre la adquisición y compatibilidad de datos con el programa "MATLAB".

En el capítulo cuarto, se presentan desarrolladas cuatro guías de prácticas basadas en valores típicos de ajuste del sistema con el objetivo de hacer más didáctico el proceso de simulación.

INDICE GENERAL

	Pág.
RESUMEN.....	VI
ÍNDICE GENERAL.....	VIII
ABREVIATURAS.....	XII
SIMBOLOGÍA.....	XIII
ÍNDICE DE FIGURAS.....	x v
ÍNDICE DE TABLAS.....	XVII
INTRODUCCIÓN.....	18
1. PRINCIPIOS DE OPERACION.....	19
1.1 Componentes del equipo.....	19
1.2. Descripción de los componentes del equipo simulador.....	20
1.2.1. Proceso controlado.....	20
1.2.2. El controlador del proceso.....	25
1.2.3. Unidades no lineales.....	29
1.2.4. El indicador de medida.....	31
1.2.5. Interfases.....	32
1.3. Conexiones entre los diferentes elementos que conforman el sistema.....	32

II. PRINCIPIOS TEÓRICOS	35
2.1. Características del proceso.....	36
2.1.1. Sistemas de primer orden.....	37
2.1.2. Sistemas de segundo orden.....	43
2.1.3. Sistemas de orden superior.....	46
2.2. El sistema completo (proceso - controlador).....	47
2.2.1. Sistema Proporcional.....	49
2.2.2. Sistema Proporcional mas Integral.....	51
2.2.3. Sistema Proporcional mas Integral mas Derivativo.....	52
2.2.4. Configuraciones adicionales.....	54
III. ADQUISICIÓN DE DATOS POR COMPUTADORA.....	62
3.1 .El puerto paralelo.....	63
3.2.Tarjeta convertidora analógica digital.....	65
3.2.1.Componentes.....	66
3.2.1.1 .Convertidor analógico digital.....	66
3.2.1.2.El generador de onda cuadrada	70
3.2.1.3.Amplificadores operacionales.....	73
3.2.1.4 Buffers de colector abierto.....	78
3.3.Programas de adquisición e interpretación de datos por computadora.....	79
3.3.1 .Adquisición de datos usando Qbasic.....	82
3.3.2.Uso de Matlab para análisis de datos.....	85

IV. ELABORACIÓN DE GUIAS DE LABORATORIO.....	87
4.1. Práctica # 1 “Simulación de sistemas proporcionales de primero, segundo y tercer orden.	88
Objetivo.	88
Fundamentos teóricos.....	88
Método de Levy.....	90
Experimentación	90
Procedimiento.	91
4.2. Práctica # 2 “Controladores Proporcional, Proporcional + Integral, Proporcional + Integral + Derivativo.....	94
Objetivos.	94
Fundamentos teóricos.....	94
Funciones de transferencia.....	94
Experimentación.	96
Procedimiento.	97
4.3. Práctica # 3 “Respuesta a la frecuencia en lazo abierto”.....	99
Objetivos.	99
Fundamentos teóricos.....	99
Diagramas de Bode.....	100
Diagramas de Nyquist.....	102
Experimentación	103

ABREVIATURAS

μs	Microsegundos
ADC0808	Convertidor analógico a digital
GND	Tierra
m	Control integral derivativo
KHz	Kilo-hertz
LPT	Puerto de impresora
PC	Computador personal
PD	Control proporcional derivativo
PI	Control proporcional integral
PID	Control proporcional integral derivativo
TTL	Lógica de transistor a transistor
V	Voltios
Vcc	Voltaje de polarización
Vpkpk	Voltaje pico a pico

SIMBOLOS

\bar{m}	=	Valor base
τ	=	Constante de tiempo
ϕ	=	Angulo de desfase
Ω	=	Ohmio
A	=	Cable conector que abre el lazo
A_u	=	Ganancia del amplificador operacional
BP	=	Banda proporcional
BP_u	=	Banda proporcional última
C	=	Capacitancia
E	=	Señal de entrada al amplificador operacional
$E(t)$	=	Señal de error
F	=	Faradio
f_c	=	Frecuencia de corte
$G_c(s)$	=	Función de transferencia del controlador
$G_p(s)$	=	Función de transferencia del proceso
K	=	Ganancia del controlador
K'	=	Ganancia del proceso
K_{cu}	=	Ganancia última
M	=	Señal de entrada al sistema
$m(t)$	=	Salida del controlador
R	=	Tubería con resistencia la flujo
R_1	=	Resistencia 1
R_A	=	Resistencia A
RA	=	Razón de amplitud
R_B	=	Resistencia B
R_f	=	Resistencia f
RM	=	Razón de magnitud
R_t	=	Resistencia t
s	=	-Variable de Laplace
S	=	Señal de salida del sistema
t	=	Tiempo
T	=	Período

t_0	=	Tiempo muerto
Td	=	Tiempo derivativo
Ti	=	Tiempo integral
T_u	=	Período de oscilación último
V_0	=	Señal de salida del amplificador operacional
w	=	Frecuencia
w_u	=	Frecuencia última
$X(t)$	=	Señal de entrada al proceso
$Y(t)$	=	Señal de salida del proceso

INDICE DE FIGURAS

	Pág.
Figura 1.1	Panel frontal del simulador PCS327..... 21
Figura 1.2	Proceso de primer orden mas un tiempo muerto..... 20
Figura 1.3	Proceso de segundo orden..... 22
Figura 1.4	Proceso de segundo orden mas un integrador..... 22
Figura 1.5	Estructura básica del simulador..... 23
Figura 1.6	Elementos constituyentes del simulador..... 25
Figura 1.7	Diagrama de amplitud vs tiempo para un controlador de dos posiciones 30
Figura 1.8	Indicador de medida 31
Figura 1.9	Polaridades relativas de los elementos 33
Figura 1.10	Diagrama para conector de lazo abierto y lazo cerrado..... 34
Figura 2.1	Ejemplo de un retardo de tiempo exponencial 36
Figura 2.2	Respuesta de un retardo exponencial a una entrada escalón 37
Figura 2.3	Respuesta de un retardo integral a una entrada escalón..... 38
Figura 2.4	Respuesta de un tiempo muerto a una entrada escalón 39
Figura 2.5	Diagrama de bloques para un proceso de primer orden..... 40
Figura 2.6	Diagrama de bloques para un proceso de primer orden mas un tiempo muerto 42
Figura 2.7	Diagrama de bloques para un proceso de segundo orden..... 44
Figura 2.8	Diagrama de bloques para un proceso de tercer orden..... 46
Figura 2.9	Diagrama de bloques del sistema completo para el simulador PCS327 48
Figura 2.10	Diagrama de bloques del sistema completo para el simulador PCS327 (abreviado). 49
Figura 2.11	Control PI + derivativo del valor medido 56
Figura 2.12	Control PI + retroalimentación derivativa..... 57
Figura 2.13	Control PI + retroalimentación pseudo - derivativa..... 59
Figura 2.14	Control proporcional + retroalimentación pseudo - derivativa..... 61
Figura 3.1	Distribución de los pines del circuito integrado ADC0808..... 67
Figura 3.2	Onda cuadrada. 71
Figura 3.3	Diagrama de conexión del reloj 72
Figura 3.4	Onda cuadrada generada por el oscilador construido con el chip 555 72

	Pág.
Figura 3.5	Función del amplificador operacional 74
Figura 3.6	Conexión básica del amplificador operacional 75
Figura 3.7	Diagrama de conexión del amplificador operacional 76
Figura 3.8	Diagrama completo del circuito op-amp cambiador de escala 77
Figura 3.9	Diagrama esquemático de la tarjeta convertidora analógico a digital. 79
Figura 3.10	Diagrama de flujo para la utilización del convertidor ADC0808... 80
Figura 3.11	Cálculo del ángulo de desfase y relación de magnitudes de dos señales senoidales 83
Figura 4.1.1	Diagrama de bloques del simulador PCS327 88
Figura 4.1.2	Conexiones del proceso para la práctica # 1. 93
Figura 4.1.3	Conexiones del simulador para la práctica #1 93
Figura 4.1.4	Cálculo de la constante de tiempo para un retardo exponencial 91
Figura 4.2.1	Diagrama de bloques del simulador PCS327 94
Figura 4.2.2	Diagrama de conexiones para la práctica #2 98
Figura 4.3.1	Gráfico de la respuesta a una entrada senoidal con retardos exponenciales. 100
Figura 4.3.2	Diagrama de Nyquist 102
Figura 4.3.3	Diagrama de conexiones para la práctica #3 104
Figura 4.4.1	Diagrama de conexiones para la práctica #4 (método de ganancia última). 108
Figura 4.4.2	Diagrama de conexiones para la práctica #4 (método de la curva de reacción). 108

INDICE DE TABLAS

	Pág.
Tabla 3.1 Direcciones del puerto paralelo de un PC	64
Tabla 4.1 Ajustes del controlador para la práctica #2.....	96
Tabla 4.3.1 Frecuencias de corte para diferentes procesos.....	101
Tabla 4.4.1 Método de la ganancia última (Ziegler Nichols).....	106
Tabla 4.4.2 Método de la curva de reacción (Ziegler Nichols).....	107

INTRODUCCIÓN

Este trabajo demuestra que se pueden actualizar y automatizar equipos mecánicos con tecnología antigua como el simulador de procesos del laboratorio de sistemas de control utilizando métodos electrónicos sencillos y de bajo costo que los puede realizar cualquier ingeniero mecánico con ciertos conocimientos de electrónica y computación.

Este proyecto consiste en el diseño y construcción de una tarjeta de adquisición de datos que nos permite acondicionar y digitalizar las señales analógicas de voltaje generadas por el simulador para poder ser ingresadas al computador y por intermedio de un programa se las puede analizar, monitorear, grabar y enviar cualquier señal de control al equipo.

El proyecto está dividido básicamente en cuatro fases que son:

1. Descripción del equipo simulador de procesos de control PCS327.
2. Construcción de la interfase equipo - computadora.
3. Programa para monitoreo y adquisición de datos por computadora.
4. Elaboración de guías de laboratorio.

CAPITULO 1

PRINCIPIOS DE OPERACIÓN

1.1. Componentes del equipo.

En la figura 1.1 se visualiza las partes que básicamente comprende el simulador. Como en todo proceso real, el simulador PCS-327 está provisto de todos sus componentes como son:

Un proceso, que permite manipulaciones y regulaciones para poder simular desde procesos lentos como es la variación de temperatura en un horno, hasta procesos rápidos como son los electrónicos; tanto de primer orden, segundo orden y de orden superior que comprenden procesos mas complejos.

Un **equipo controlador**, en el que se pueden obtener varias configuraciones de controladores proporcionales, de dos y tres posiciones, PI, PD, ID, PID.

Un **sistema de acción** que comprende los elementos actuadores, detectores y transmisores de señales los cuales se encuentran implícitas en el sistema, ya que al simulador se lo ha diseñado de tal forma que estos elementos tienen una función de transferencia unitaria, es decir, que no ocasionan retardos de tiempo ni tiempos muertos, por lo tanto no son necesario manipularlos.

Además el simulador cuenta con **unidades no lineales** que permiten añadir al controlador saturación, banda muerta e histéresis.

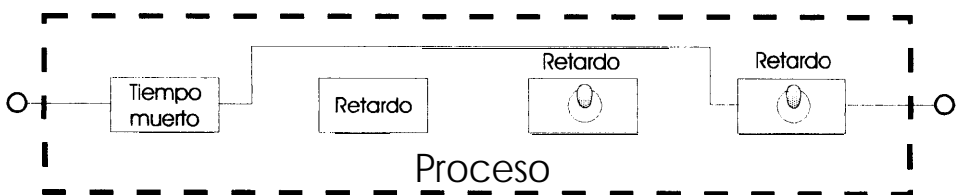
1.2. Descripción de los componentes del equipo simulador.

1.2.1. El proceso controlado

El proceso comprende de elementos como retardos de tiempo, tiempos muertos e integradores, los cuales pueden ser insertados u omitidos del mismo con el uso de cables conectores para formar algunas combinaciones Como lo indican las figuras: 1.2, 1.3 y 1.4.

FIGURA 1.2

PROCESO DE PRIMER ORDEN MAS UN TIEMPO MUERTO.



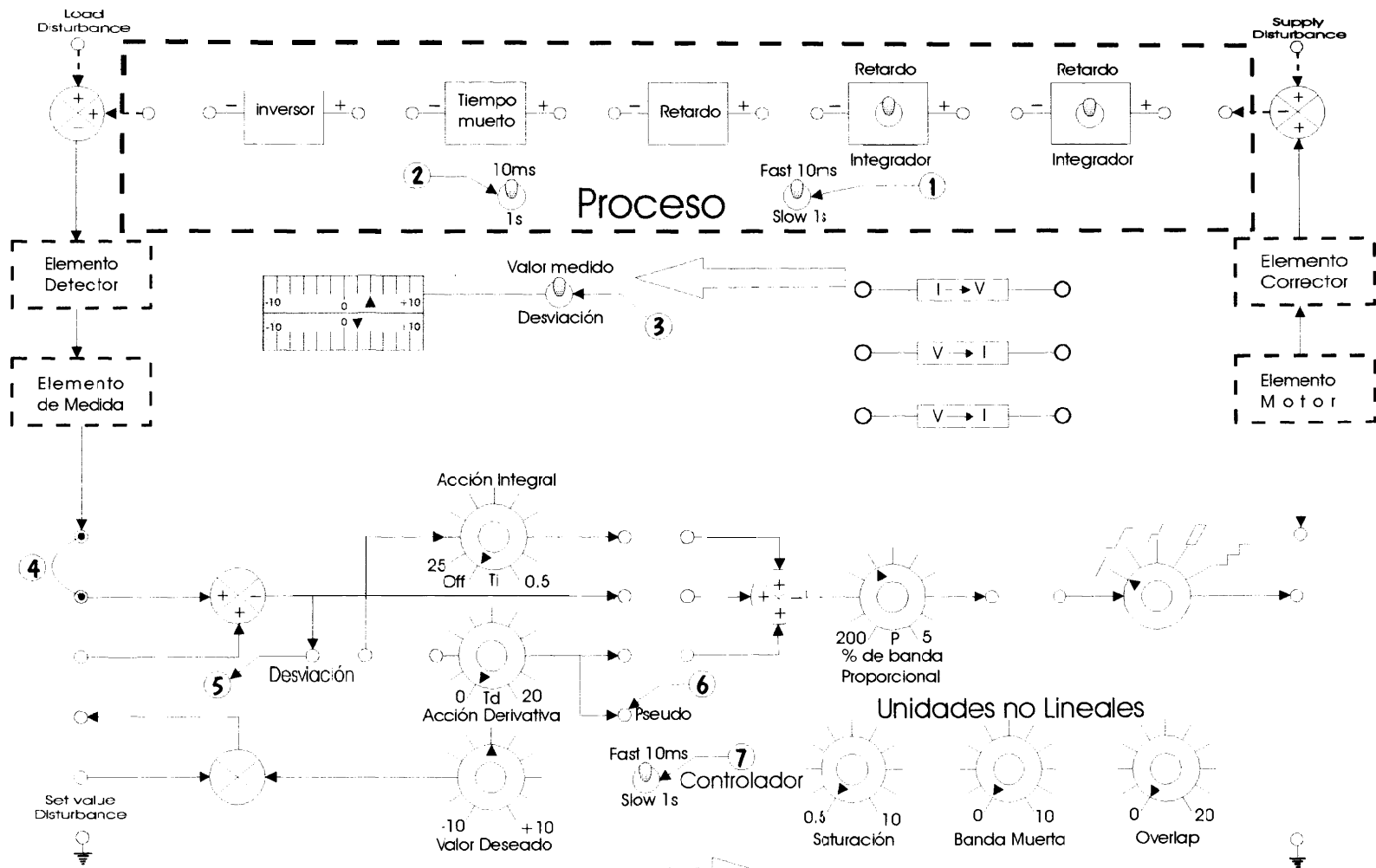


FIGURA: 1.1
 PANEL FRONTAL DEL SIMULADOR PCS327

FIGURA 1.3

PROCESO DE SEGUNDO ORDEN

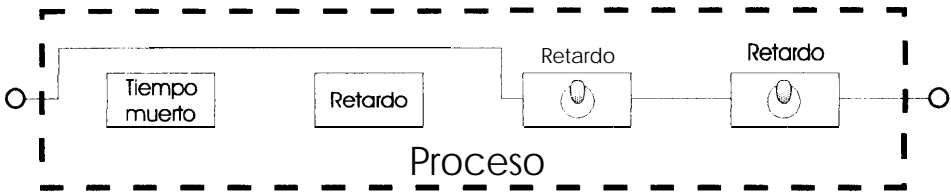
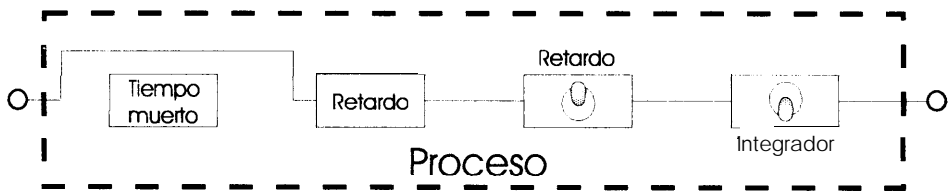


FIGURA 1.4

PROCESO DE SEGUNDO ORDEN MAS UN INTEGRADOR



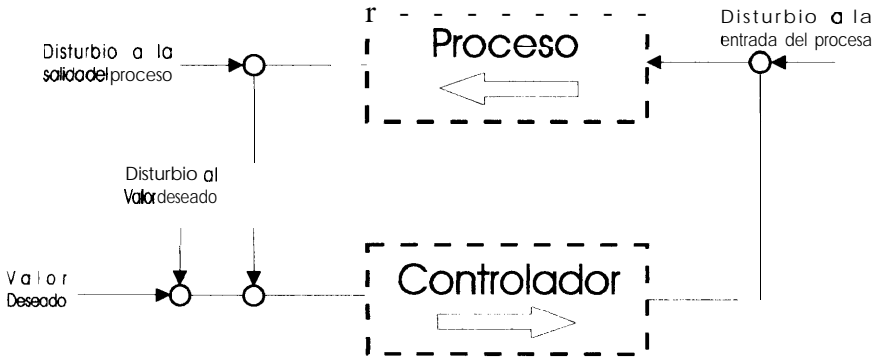
Los elementos que conforman el proceso son los siguientes:

- Un retardo simple con una constante de tiempo de 10 milisegundos o un segundo.
- Dos retardos e integradores (seleccionados por un interruptor) con constantes de tiempo de 10 milisegundos o un segundo.
- Un tiempo muerto de 10 milisegundos o un segundo.
- Un inversor.

En la figura 1.5 se puede apreciar claramente la estructura básica del simulador.

FIGURA 1.5

ESTRUCTURA BÁSICA DEL SIMULADOR.



Con estos elementos se pueden hacer muchas combinaciones diferentes de procesos con características lineales, los cuales nos ayudan a conocer su comportamiento para así encontrar o comprobar métodos que logren estabilizarlo. Así también el proceso dispone de dos conmutadores para la selección de dos velocidades de proceso variando las constantes de tiempo de cada bloque; un conmutador maneja las constantes de tiempo de los tres primeros bloques, y el segundo en cambio maneja el tiempo muerto, pudiendo ser de 0.01 segundos a 1 segundo tanto el tiempo muerto como las constantes de tiempo, por lo que se pueden simular procesos lentos y rápidos.

Cada elemento o bloque del proceso produce un inversión en la polaridad, por tal razón se ha incluido en el mismo un inversor, para el

caso que sea necesario invertir la polaridad de la señal, lo cual se realiza por intermedio de las conexiones. Para más información acerca de las conexiones vea el capítulo 1 sección 1.3.

Adicionalmente se le ha añadido dos elementos receptores, uno a la entrada del proceso “Supply Disturbance” que permite la introducción de una señal que representa un disturbio y la otra a la salida del proceso “Load Disturbance” así mismo para la introducción de un disturbio o carga a la salida del proceso.

Los elementos que vienen implícitos en el proceso son:

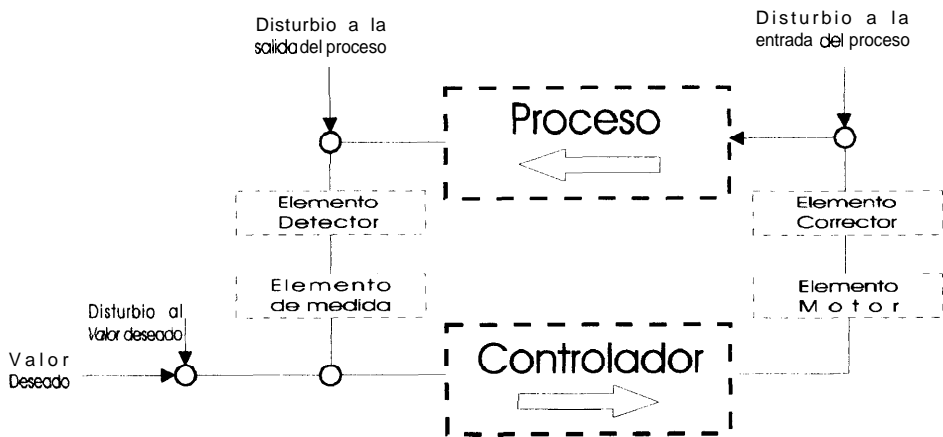
El **elemento corrector**, o elemento final de control que hace las veces de una válvula de control reguladora de caudal; el **elemento motor**, que es quien le da movimiento al elemento corrector, esto puede ser un dispositivo eléctrico o neumático. El **elemento detector**, o sensor que es quien detecta la señal a la salida del proceso y el **elemento de medida**, que mide la señal y la transmite al controlador.

Estos elementos no representan ningún equipo o bloque independiente en el sistema pero están incluidas en el gráfico par indicar la posición en el proceso donde cada elemento podría ser encontrado en la práctica. Los

elementos mencionados anteriormente se encuentran graficados en la figura 1.6.

FIGURA 1.6

ELEMENTOS CONSTITUYENTES DEL SIMULADOR



1.2.2. El controlador del proceso.

En síntesis el controlador es el cerebro del circuito de control por lo tanto es el dispositivo que toma la decisión en el sistema de control, y para hacerlo el controlador:

1. Compara la señal del proceso que llega al transmisor, la variable que se controla contra el punto de control y
2. Envía la señal apropiada a la válvula de control o cualquier otro elemento final de control para mantener la variable que se controla en el punto de control.

proceso, es decir, si el proceso está en la posición “Slow”, el controlador deberá estar en la misma posición.

Para evitar problemas debido a la tendencia del integrador, se ha provisto la posición “Off ” en el control de acción integral, en el cual el término integral es igual a cero. El camino que recorre la señal a través del controlador desde que entra hasta que sale no se invierte al pasar a través de cada término que comprende el controlador, a diferencia como ocurre en el proceso, por tal motivo es de fácil aplicación para procesos externos con retroalimentación.

La salida del controlador pasa a través de un interruptor de varias posiciones que selecciona las unidades no lineales descritas en el párrafo 1.2.3. Así mismo este elemento tiene conectores a la entrada y la salida, de tal manera que mediante cables conectores se lo pueda incluir u omitir del proceso. Similarmente el controlador tiene conectores a la entrada y salida por lo que se lo podría utilizar para controlar procesos externos, o para el mismo proceso cuando se desea simular sistemas en lazo abierto, desconectando simplemente el cable que cierra el lazo. Cable # 4 Fig.: 1.1

Así mismo si se dispone de algún controlador externo que trabaje con los rangos de corriente y voltaje compatibles con los del simulador, puede ser añadido al controlador algún término independiente como los

proporcionales, integrales o derivativos o el controlador completo, en el que se puede excluir por completo al controlador, elemento comparador y las unidades no lineales del simulador.

El valor deseado o punto de control, puede también ser desconectado completamente desde el elemento comparador para poder aplicar un valor deseado externo si es necesario. La salida del valor deseado es la suma del punto de control mas el valor de cualquier disturbio inyectado a la entrada del valor deseado “supply disturbance” Fig. 1.1.

La acción derivativa está diseñada de tal manera que puede ser aislada completamente y usado en diferentes posiciones del sistema como lo indican las figuras 2.11, 2.12, 2.13 y 2.14 del capítulo 2.

Las acciones proporcional e integral pueden añadirse al controlador de manera simultánea, el la que se pueden obtener combinaciones de controladores proporcional, integral, proporcional + integral, proporcional + derivativo, integral + derivativo, proporcional + integral + derivativo, etc..

Además en el controlador tenemos una salida “Deviation”, en la cual podemos medir o visualizar la desviación que es la diferencia entre el valor deseado y el valor medido a la salida del proceso. Figura 1.1, salida # 5.

1.2.3. Unidades no lineales.

Estos elementos proveen de una selección de características no lineales ideales. La posición de los elementos está ubicada entre la salida del controlador y la entrada del proceso de tal manera que permite considerar cualquiera de las dos siguientes características:

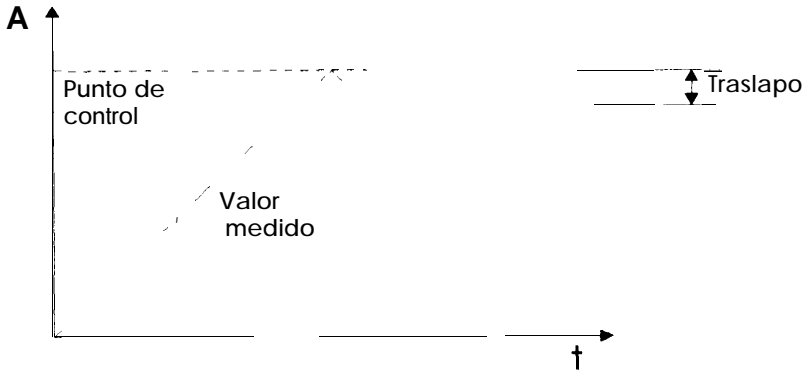
- a) Una representación típica de unidades no lineales mecánicas como son pérdida de carrera, histéresis, banda muerta, etc. que pueden ocurrir en cualquier motor o elementos mecánicos de corrección, medida o detección.

- a) Como representación de varios tipos de controladores con características no lineales como son los de dos y tres posiciones con o sin traslapo.

Un ejemplo de controlador de dos posiciones es un regulador que enciende o apaga la bomba que envía agua a un tanque elevado, el cual es manejada por el sensor (flotador), que se lo ajustaría con el valor deseado cuando el flotador se encuentra en su posición más alta y apague la bomba y cuando el nivel descienda a una cierta posición encienda la bomba, de tal modo que el diagrama de amplitud **versus** tiempo quedaría de la siguiente forma:

FIGURA 1.7

DIAGRAMA DE AMPLITUD vs TIEMPO PARA UN CONTROLADOR DE DOS POSICIONES (REF. 1).



La diferencia entre las posiciones “On” y “Off ” es conocida como traslapo.

Los elementos que comprenden las unidades no lineales son los siguientes:

- Un conmutador
- Un control de límites
- Un control de banda muerta
- Un control de traslapo

Se ha provisto de varias formas de no linealidad con los efectos de los tres controles (Saturación, banda muerta y traslapo “overlap”) en la que las aplicaciones de los párrafos a) y b) están explicados gráficamente en

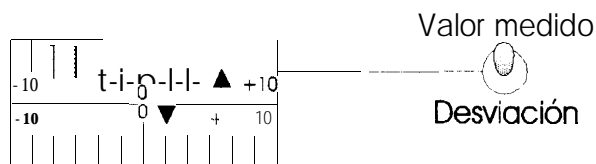
la figura 2 de la referencia 1. Cabe destacar que las características de pérdida de carrera son válidas sólo para un limitado rango de frecuencias de 3 a 60 Hz. En el modo rápido “Fast” (0.01s.) y de 0.03 a 0.6 Hz en el modo lento “Slow” (1s.), modo que está determinado por la velocidad tanto del controlador como del proceso. Las unidades no lineales podrían ser aisladas de la salida del controlador por medio de cables conectores y colocada dentro de cualquier otra posición dentro del sistema para demostrar o visualizar algún efecto especial que esto podría producir.

1.2.4. Indicadores de medida.

El simulador cuenta con dos indicadores de medida analógicos de pluma, ambos con escalas entre - 10 y + 10 voltios. La escala superior indica el valor medido a la salida del proceso o la desviación entre el punto de control y el valor medido por medio de la acción de un conmutador. La escala inferior en cambio indica el valor deseado o punto de control.

FIGURA 1.8

INDICADOR DE MEDIDA



1.2.5. Interfases.

Las unidades de interfase comprenden tres convertidores, dos de voltaje a corriente y uno de corriente a voltaje, los cuales se los puede utilizar para permitir que cualquiera, tanto el proceso como el controlador puedan ser reemplazados por algún equipo externo operando dentro de un rango estándar de 4 a 20 miliamperios. El segundo convertidor voltaje corriente ($V \rightarrow I$) podría ser usado para conexión de una corriente operadora de una tabla registradora. (vea la referencia 1 apéndice 2 para más detalles).

Para más información acerca de las especificaciones técnicas o el diagrama eléctrico del equipo simulador consulte la referencia 1 sección 3 y 6 respectivamente.

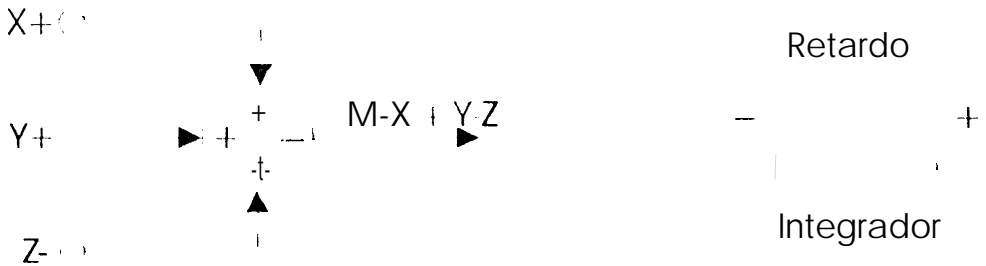
1.3. Conexiones entre los diferentes elementos que conforman el sistema.

Como podemos apreciar en la figura 1.1, algunos elementos que conforman el sistema producen una inversión en la polaridad de la señal, por tal motivo en el proceso se ha incluido un inversor para asegurarse que el número total de inversiones alrededor del lazo cerrado sea siempre positiva. Se puede notar claramente si la señal está o no invertida, visualizando el indicador de medida el cual para un movimiento de la pluma del valor deseado a la derecha, el valor medido se corre hacia la izquierda.

Para verificar el número total de inversiones de polaridad se ha dibujado claramente en la caja del simulador las polaridades relativas de la entrada y la salida de cada uno de los elementos y puntos de suma tal como lo indica la figura 1.9.

FIGURA 1.9

POLARIDADES RELATIVAS DE LOS ELEMENTOS.



Nótese que los puntos de suma se comportan como si se produjera una diferencia tal como lo indica la figura 1.9, por tal razón se debe asegurar que el número de inversiones de polaridad sea positivo (incluyendo los puntos de suma) para evitar retroalimentaciones negativas

En lo que constituye el controlador no existen manipulaciones externas que producirán una inversión en la polaridad, entonces es necesario tomar en cuenta sólo lo que constituye el proceso, por lo tanto se puede estandarizar las conexiones en función del número de bloques conectados en el proceso de la siguiente manera:

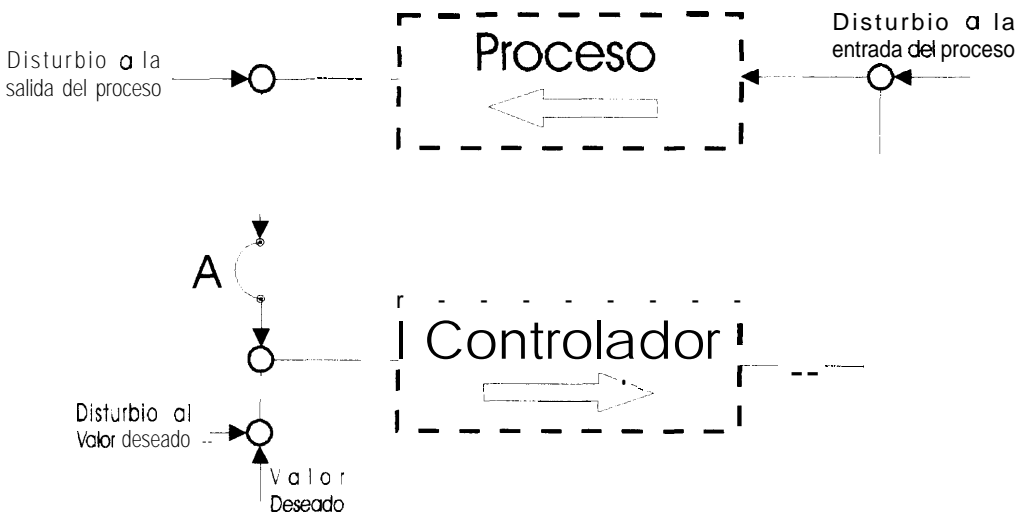
- Número par de bloques conectados → requiere inversor.
- Numero impar de bloques conectados → no requiere inversor.

Estas conexiones se las puede apreciar en la figura 1.2, 1.3 y 1.4. El resto de conexiones que se pueden realizar no influyen en el cambio de polaridad de la señal, por tal motivo no se las ha graficado.

El equipo también presenta gran facilidad para simular sistemas en lazo abierto o en lazo cerrado usando cualquier configuración tanto en el proceso como en el controlador, simplemente usando el cable conector A que abre o cierra el lazo de todo el sistema, tal como lo indica el esquema de la figura 1.10.

FIGURA 1.10

DIAGRAMA PARA CONECTOR DE LAZO ABIERTO Y LAZO CERRADO



CAPITULO 2

PRINCIPIOS TEÓRICOS

En este capítulo se hablará acerca de las funciones de transferencia tanto del proceso como del sistema completo. Estas se pueden calcular gracias a que se conocen todos los términos que intervienen en el proceso, como son las constantes de tiempo y tiempos muertos, y además las variables que dependen de las regulaciones hechas a la banda proporcional, tiempo integral y tiempo derivativo por medio de sus respectivos potenciómetros. Se debe notar también que los elementos implícitos en el proceso como son el elemento detector, corrector, motor y de medida no ocasionan ningún efecto que pueda producir cambios en la señal de medida o de control.

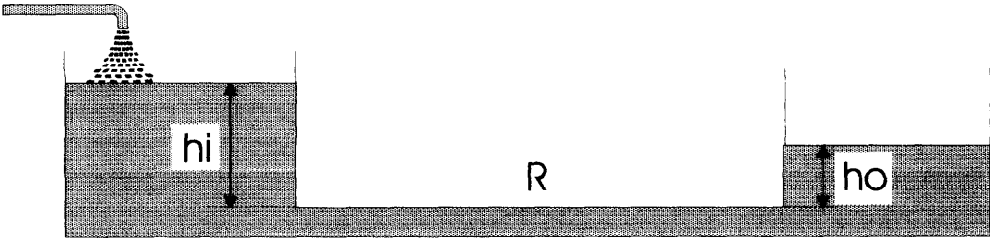
2.1. Características del proceso simulado.

Como se ha visto en el capítulo anterior, el proceso puede estar formado por retardos de tiempo los cuales están divididos en tres tipos principales: el retardo exponencial, el retardo integral y el tiempo muerto que se pueden presentar por una variedad de razones en procesos reales.

El retardo exponencial en un proceso se lo puede explicar por ejemplo en el caso de que se requiere transferir agua desde el tanque izquierdo hacia el derecho por intermedio de la tubería R , como lo indica la figura 2.1.

FIGURA 2.1

EJEMPLO DE UN RETARDO DE TIEMPO EXPONENCIAL

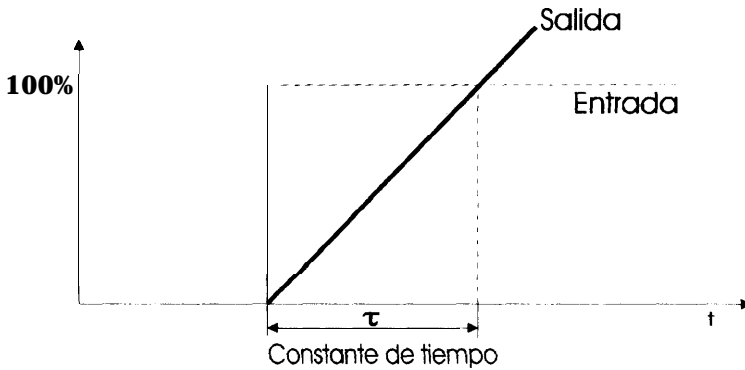


El nivel del líquido del tanque izquierdo se mantiene constante a h_i , mientras el líquido es transferido al tanque derecho a través de la tubería R , por lo tanto, el flujo de agua hacia el tanque derecho es proporcional a la diferencia entre los dos niveles e inversamente proporcional a la resistencia de la tubería R . Esto significa que cuando el tanque derecho está vacío, la cantidad de flujo es alta por

El retardo integral en un proceso está presente cuando un cambio repentino en la entrada, la salida es inicialmente cero pero se incrementa proporcionalmente con el tiempo como lo indica la figura 2.3. Un ejemplo práctico para esto es cuando un motor eléctrico es repentinamente encendido, el cual después de un corto tiempo el eje rota a velocidad constante, entonces el ángulo de rotación del eje se incrementa linealmente con el tiempo. La razón del incremento es gobernada por la constante de tiempo de integración, el cual en este caso es tomada cuando la salida es igual a la entrada.

FIGURA 2.3.

RESPUESTA DE UN INTEGRADOR A UNA ENTRADA ESCALÓN



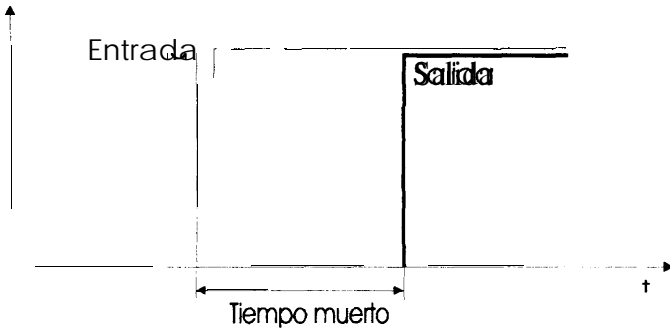
Un ejemplo del **tiempo muerto** en un proceso es el control de temperatura de agua en un tanque de almacenamiento, la cual se la mide a la salida del tanque.

Si cualquier cambio repentino de temperatura ocurre en el interior del tanque, el tiempo en que el sensor detecte este cambio depende entonces de la distancia que existe entre éste y el tanque y además de la velocidad de flujo del líquido a través

de la tubería. Este intervalo de tiempo se lo conoce como tiempo muerto o retardo de distancia - velocidad.

FIGURA 2.4

RESPUESTA DE UN TIEMPO MUERTO A UNA ENTRADA ESCALÓN.



2.1.1. Sistemas de primer orden

Como se ha descrito en el capítulo anterior, para sistemas de primer orden sólo se utiliza un bloque del proceso en la que se puede simular con el conmutador # 1 de la figura 1.1 en la posición hacia arriba un retardo de tiempo exponencial; o a su vez un retardo de tiempo integral con el conmutador en la posición hacia abajo, por lo tanto la función de transferencia del proceso se define en función de la selección del conmutador de la siguiente manera:

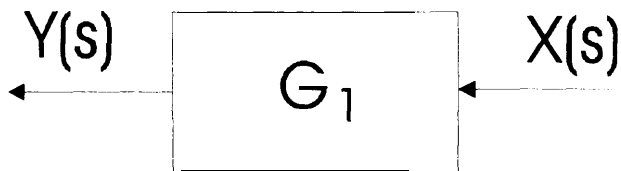
Si $X(t)$ es la señal que entra al proceso, y $Y(t)$ es la señal de salida del mismo, entonces la relación $Y(t) / X(t)$ es igual a función de transferencia del proceso en función del tiempo. Se ha comprobado que las técnicas de

transformada de Laplace y linealización son particularmente útiles para el análisis de la dinámica de los procesos y diseño de sistemas de control, debido a que proporcionan una visión general del comportamiento de gran variedad de procesos e instrumentos, razón por la que se utilizará la transformada de Laplace para describir las ecuaciones que representen tanto el proceso como el sistema completo (proceso - controlador) en lazo abierto o cerrado.

La función de transferencia se define entonces como la relación de la transformada de Laplace de la variable de salida sobre la transformada de Laplace de la variable de entrada, como lo indica la figura 2.5, siendo $Y(t) / X(t) = (k * e^{-at/\tau})$ para procesos de primer orden con retardos de tiempo exponenciales.

FIGURA 2.5

DIAGRAMA DE BLOQUES PARA UN PROCESO DE PRIMER ORDEN



$$G_1 = \text{Laplace} \left\{ G_p(t) = \frac{Y(t)}{X(t)} = k * e^{-at/\tau} \right\}$$

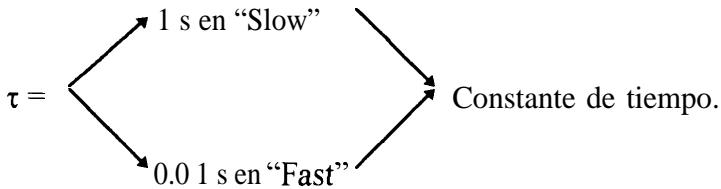
$$G_p(s) = G_1 = \frac{k}{\tau s + a}$$

$$k = k' * \tau$$

Donde:

$G_p(s)$ = Representación general de la función de transferencia.

$k' = 1$: Ganancia del proceso



En general, s representa la variable de Laplace, y la unidad de s es el recíproco de la unidad de la variable independiente tiempo utilizada en la definición de la transformada de Laplace, por lo tanto, la unidad de s es 1/tiempo.

Se puede decir que la ganancia indica cuánto cambia la variable de salida por unidad de cambio en la función de forzamiento o variable de entrada; es decir, la ganancia define la sensibilidad del proceso. τ en cambio proporciona la velocidad de respuesta de la salida $X(t)$ una vez que responde al cambio de la entrada $Y(t)$.

Por lo tanto, la función de transferencia para procesos de primer orden con retardos de tiempo exponenciales se define por la siguiente ecuación:

$$Gp(s) = \frac{1}{\tau s + 1} \quad 2.1$$

Similarmente, si se selecciona por intermedio del conmutador en posición hacia, abajo un retardo de tiempo integral, tenemos entonces que la función de transferencia del proceso aplicando la transformada de Laplace es:

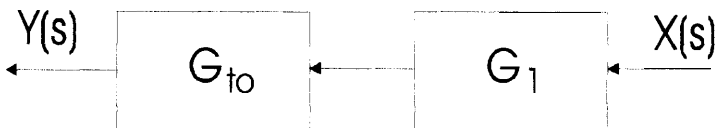
$$Gp(s) = \frac{1}{\tau s} \quad 2.2$$

Así mismo el valor de τ dependerá de la posición del conmutador del proceso en “Fast” o “Slow” como se indicó anteriormente.

Como a todos estos procesos simulados se les puede agregar tiempo muerto, consideramos entonces este caso en el que se describe por el diagrama de bloques de la figura 2.6.

FIGURA 2.6

DIAGRAMA DE BLOQUES PARA UN PROCESO DE PRIMER ORDEN MAS UN TIEMPO MUERTO



Cuando la transformada de Laplace contiene un tiempo muerto (retardo de transporte o de velocidad) en la función de la transformada aparece el término exponencial e^{-t_0*s} donde t_0 es el tiempo muerto; por lo tanto la función de transferencia del proceso de primer orden más un tiempo muerto es la multiplicación de los dos bloques:

$$G_p(s) = G_1 * G_{t_0}$$

donde $G_{t_0} = e^{-t_0*s}$

$$G_p(s) = G_1 * e^{-t_0*s}$$

2.3

2.1.2. Sistemas de segundo orden

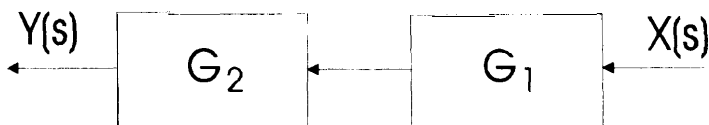
En esta sección se analizarán las funciones de transferencia que se pueden realizar con las combinaciones de dos bloques como son los de retardo de tiempo exponenciales y los retardos de tiempo integrales, por lo que tendremos sistemas de segundo orden con dos retardos exponenciales; un retardo exponencial y uno integral; y dos retardos integrales; adicionalmente a cada una de estas combinaciones se les puede añadir tiempo muerto.

Tomando en cuenta la transformada de Laplace para cada uno de los términos o bloques del proceso conectados en serie, podemos utilizar las reglas del álgebra de diagramas de bloques para encontrar la relación

$Y(s) / X(s)$ que es igual a la multiplicatoria de cada uno de los bloques como se aprecia en la figura 2.7:

FIGURA 2.7

DIAGRAMA DE BLOQUES PARA UN PROCESO DE SEGUNDO ORDEN.



Donde:

$$Gp(s) = \frac{Y(s)}{X(s)} = G_1 * G_2$$

Por lo tanto tenemos que para un proceso con dos retardos de tiempo exponenciales sin tiempo muerto, su función de transferencia está definida por la siguiente ecuación:

$$G_1 = G_2 = \frac{1}{\tau s + 1}$$

$$Gp(s) = \frac{1}{(\tau s + 1)^2}$$

$$Gp(s) = \frac{1}{(\tau s)^2 + 2\tau s + 1}$$

Donde igualmente el valor de τ depende de la selección del conmutador en “Fast” o “Slow”.

Cuando tenemos en cambio un proceso de segundo orden con dos bloques en serie conformado por un retardo exponencial y un retardo integrador, este se comporta de una manera muy diferente al anterior, de tal forma que la función de transferencia para este proceso es:

$$G_1 = \frac{1}{\tau s + 1} \quad \text{Y} \quad G_2 = \frac{1}{s}$$

$$G_p(s) = G_1 * G_2$$

$$G_p(s) = \frac{1}{(\tau s + 1) * s} \quad 2.5$$

Para la combinación de dos retardos de tiempo integrales, la función de transferencia del proceso es entonces el cuadrado de un integrador:

$$G_1 = G_2 = \frac{1}{s}$$

$$G_2 = \frac{1}{(s)^2} \quad 2.6$$

Similarmente como se explicó en la sección 2.1.1, a cada una de las configuraciones anteriores, para procesos de segundo orden se les puede agregar tiempo muerto:

$$G_p(s) = G_1 * G_2 * G_{to}$$

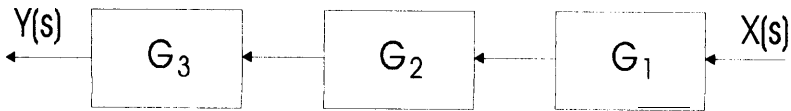
$$G_p(s) = G_1 * G_2 * e^{-t_0 * s} \quad 2.7$$

2.1.3. Sistemas de orden superior

En esta sección se obtendrán los modelos matemáticos que describen el comportamiento de procesos más complejos que los vistos anteriormente como son los de tercer orden. Para estos procesos que igualmente se componen de sistemas de primer orden en serie como lo indica la figura 2.8, se pueden obtener las siguientes configuraciones usando el simulador:

FIGURA 2.8

DIAGRAMA DE BLOQUES PARA UN SISTEMA DE TERCER ORDEN



- Tres retardos exponenciales en serie:

$$G_p(s) = G_1 * G_2 * G_3$$

$$Gp(s) = \frac{1}{(\tau s + 1)^3}$$

$$Gp(s) = \frac{1}{(\tau s)^3 + 3(\tau s)^2 + 3\tau s + 1} \quad 2.8$$

- Dos retardos exponenciales mas un integral:

$$Gp(s) = \frac{1}{(\tau s + 1)^2 * \tau s}$$

$$Gp(s) = \frac{1}{(\tau s)^3 + 2(\tau s)^2 + \tau s} \quad 2.9$$

- Dos retardos integrales mas un exponencial:

$$Gp(s) = \frac{1}{(\tau s)^3 + (\tau s)^2} \quad 2.10$$

Si consideramos el tiempo muerto para estos procesos que es la máxima configuración de bloques que se pueden lograr con el simulador tenemos:

$$G_p(s) = G_1 * G_2 * G_3 * G_{t_0}$$

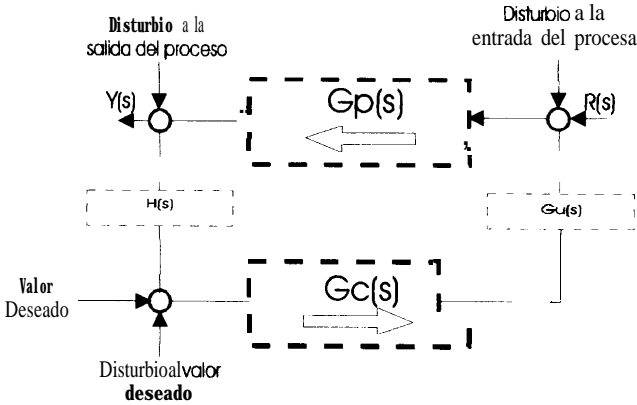
$$G_{,,}(s) = G_1 * G_2 * G_3 e^{-t_0 * s} \quad 2.11$$

2.2. El sistema completo (proceso - controlador)

En esta sección se analizarán las funciones de transferencia de todo el sistema simulado, tomando en cuenta cada una de las configuraciones posibles para el controlador.

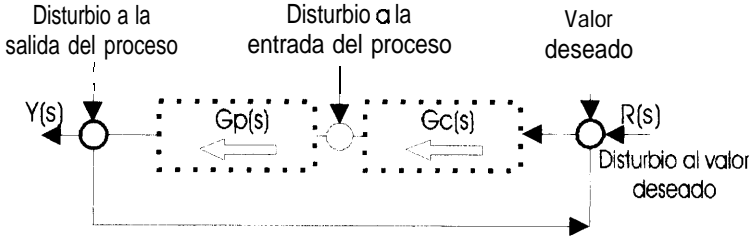
Utilizando un diagrama de bloques aplicado para todo el sistema, como lo indica la figura 2.9, podemos tener una representación gráfica de las funciones de transferencia usando un control por retroalimentación.

FIGURA 2.9
 DIAGRAMA DE BLOQUES DEL SISTEMA COMPLETO PARA EL SIMULADOR PCS327.



donde $G_p(s)$ puede ser cualquiera de las configuraciones del proceso simulado visto en la sección 2.1, y $G_c(s)$ representa al controlador, el cual de igual manera puede tener muy variadas configuraciones como se analizará más adelante. Los elementos $G_p(s)$ y $H(s)$ representan a los elementos motor - corrector y sensor - transmisor respectivamente. A estos dos últimos elementos no se los incorporó en el sistema debido a que sus funciones de transferencia corresponden a un sistema proporcional de ganancia unitaria que no afectan a la ecuación general del sistema, por lo tanto, el diagrama de bloques de la figura 2.9 se lo puede reemplazar por su equivalente de la figura 2.10.

FIGURA 2.10.



2.2.1. Sistema proporcional

Este sistema está manejado por el controlador proporcional que es el tipo más simple de controlador, con excepción del controlador de dos posiciones. La ecuación que describe su funcionamiento es la siguiente:

$$m(t) = \bar{m} + k * E(t) \quad 2.12$$

donde,

$m(t)$ = salida del controlador en mA.

$k = 100 \text{ BP} =$ ganancia del controlador mA/mA

$E(t)$ = señal de error, mA; es la diferencia entre el punto de control y la variable que se controla.

$\bar{m} = 12 \text{ mA} =$ vla or base. El significado de este valor es la salida del controlador cuando el error es cero; $4 < \bar{m} < 20 \text{ mA}$ que corresponden proporcionalmente a $-1 \text{ OV} < \bar{m} < +1 \text{ OV}$.

Para obtener la función de transferencia del controlador proporcional, se obtiene el transformada de Laplace a la ecuación 2.12, y de ahí resulta la siguiente función de transferencia:

$$G_c(s) = k = \frac{100}{BP} \quad 2.13$$

Valiéndose de las reglas de los diagramas de bloques podemos encontrar que la función de transferencia para este sistema en lazo cerrado resulta:

$$\frac{M}{S} = G(s) = \frac{G_p(s) * G_c(s)}{1 + G_p(s) * G_c(s)} \quad 2.14$$

$$G(s) = \frac{k * G_p(s)}{1 + k * G_p(s)} \quad 2.15$$

Para resumir brevemente, los controladores proporcionales son los más simples, con la ventaja de que sólo se tiene un parámetro de ajuste k o BP ; la desventaja de los mismos es que operan con una desviación de la variable que se controla para sistemas en lazo cerrado con retroalimentación. Este error por ejemplo para sistemas de primer orden se lo puede representar por la siguiente ecuación:

$$e(t) = \frac{1}{1 + k} \quad 2.16$$

Entonces, mientras más grande es la ganancia del controlador más pequeña es la desviación, pero con la consecuencia de que el proceso se vuelve más inestable. Para más detalles vea la referencia 2, capítulo 6.

2.2.2. Sistema proporcional mas integral (PI)

Debido a la gran desventaja que presentan los controladores proporcionales al manejar una desviación en estado estacionario, se añade al controlador proporcional la acción integral o de reajuste para eliminar la desviación y en consecuencia el controlador se convierte en un controlador proporcional mas integral **PI** manejado por la siguiente ecuación.

$$m(l) = \bar{m} + kE(t) + \frac{k}{T_i} \int E(t) dt$$

donde T_i es el tiempo de integración o de reajuste en segundos o milisegundos dependiendo de la posición del conmutador “Fast” o “Slow”.

Utilizando las mismas definiciones de las variables de desviación que se dan en las ecuaciones de la sección 2.2.1 y obteniendo la transformada de Laplace se tiene:

$$G_c(s) = k * \left(1 + \frac{1}{T_i s} \right) \quad 2.17$$

Para analizar el comportamiento del sistema completo con controladores PI, usamos la ecuación 2.14 que representa al diagrama de bloques de la figura 2.10:

$$\frac{M}{S} = G(s) = \frac{k * \left(1 + \frac{1}{T_i s} \right) * G_p(s)}{1 + k * \left(1 + \frac{1}{T_i s} \right) * G_p(s)} \quad 2.18$$

por lo tanto esta es la ecuación característica por la que se puede predecir la respuesta de este sistema PI en la que se tienen dos parámetros de ajuste: la banda proporcional **BP** y el tiempo integral **T_i** o rapidez de reajuste. La ventaja de este sistema es que la acción integral o de reajuste elimina la desviación de estado estable.

2.2.3. Sistema proporcional mas integral mas derivativo (PID)

En la **mayoría** de sistemas se añade otro modo de control al controlador PI, este nuevo modo de control es la acción derivativa, que también se conoce como rapidez de derivación o preactuación; tiene como propósito anticipar hacia dónde va el proceso, mediante el cálculo de la derivada del error, su pendiente. La ecuación descriptiva es la siguiente:

$$m(t) = \bar{m} + kE(t) + \left(\frac{k}{T_i} \int E(t)dt\right) + \left(kT_d \frac{dE(t)}{dt}\right)$$

donde T_d es la rapidez de derivación.

Usando las mismas definiciones de variables de desviación que se dan en la sección anterior, y obteniendo la transformada de Laplace se tiene la siguiente función de transferencia ideal para el controlador:

$$G_c(s) = k \left(1 + \frac{1}{T_i s} + T_d s\right) \tag{2.19}$$

Así mismo como se procedió en la sección anterior, partiendo del diagrama de bloques de la figura 2.10 y reemplazando $G_c(s)$ en la ecuación 2.14 se tiene:

$$\frac{M}{S} = G(s) = \frac{k \left(1 + \frac{1}{T_i s} + T_d s\right) G_p(s)}{1 + k \left(1 + \frac{1}{T_i s} + T_d s\right) G_p(s)} \tag{2.20}$$

Siendo esta la ecuación característica del sistema **P+I+D** donde se tienen tres parámetros de ajuste: la banda proporcional, tiempo integral y tiempo derivativo. La rapidez derivativa está en segundos o milisegundos dependiendo de la posición del conmutador “Fast” o “Slow”. La ventaja del modo derivativo es que proporciona la capacidad de ver hacia dónde se dirige el proceso.

2.2.4. Configuraciones adicionales

En esta sección se analizarán ciertas configuraciones que se pueden lograr con la eliminación de la banda proporcional, o a su vez la ubicación del tiempo derivativo en diferentes posiciones del sistema, en la que el controlador PID convencional cuya señal de control es proporcional a la suma de la desviación y de los términos derivativo e integral de la desviación, se pueden hacer algunas variantes simples que se encuentran en varios controladores industriales, o a su vez para ajustar otras configuraciones que pueden ser analizadas como casos de estudio; para esto el simulador PCS327 está provisto de las siguientes facilidades:

- a) La entrada al potenciómetro del término derivativo y la salida de éste se las pueden ubicar en cualquier punto del sistema.

- b) La salida de la acción derivativa, la cual precede del mismo diferenciador, es llevado a un terminal llamado PSEUDO. Correctamente usado da origen a una forma de control llamada PSEUDO - DERIVATIVA, en la cual se evita la principal desventaja de tener un diferenciador físico en el circuito (fig.: 1.1 elemento 6), esto es, que cambios en la entrada debido a la carga, valor deseado o a la entrada del proceso pueden causar sobrecarga temporal en el sistema.

En todas las configuraciones que se verán a continuación se asume que se emplea la acción integral. Sin embargo, si tomamos como necesario que el estado estable de la desviación para cualquier proceso escogido sea cero, es requerido por tanto la acción integral.

Cuando los procesos contienen un término integral, puede parecer que la acción integral en el controlador no sea necesaria para obtener a cero la desviación de estado estable, pero esto es verdadero sólo cuando se consideran cambios en la carga o en el valor deseado. Si ocurre un disturbio a la entrada del proceso, produciría una desviación de estado estable, a menos que se use la acción integral.

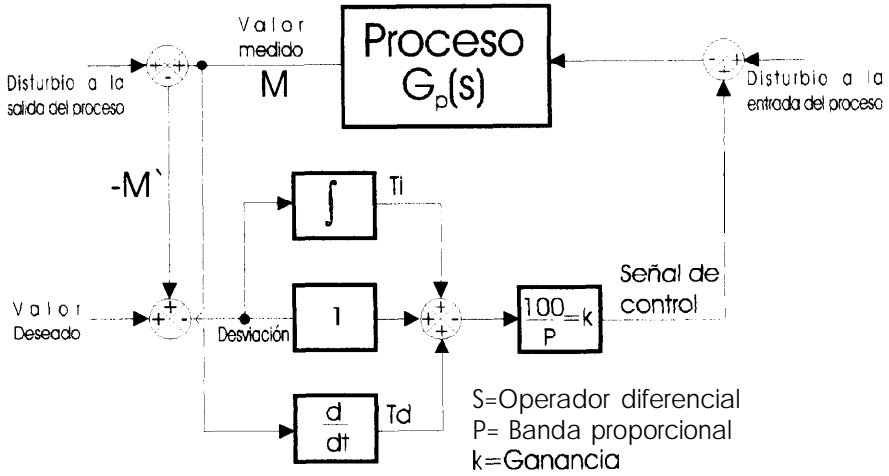
Con lo visto anteriormente se pueden tener las siguientes posibilidades:

1) Control proporcional mas integral mas derivativo del valor medido.

Esta configuración es equivalente a retroalimentación de velocidad o taco generador de un servo sistema.

FIGURA 2. 11

CONTROL **P+I+DERIVATIVO** DEL VALOR MEDIDO



Para asegurar la correcta polaridad del circuito derivativo, este es alimentado desde la salida del proceso o valor medido, como lo indica la figura 2.11. Sin embargo en ausencia de cualquier disturbio a la salida del proceso, las características de control no son afectadas, es decir, se comporta como un controlador **PID** convencional.

Podemos escribir las siguientes ecuaciones para este sistema:

$$M = -k * G_p(s) * [E(1 + \frac{1}{T_i s}) + M(T_d s)] \quad \text{donde} \quad E = M - S$$

Substituyendo **E** y reordenando tenemos:

$$\frac{M}{S} = G(s) = \frac{k * (1 + \frac{1}{T_i s}) * G_p(s)}{1 + k * (1 + \frac{1}{T_i s} + T_d s) * G_p(s)} \tag{2.21}$$

Esta es la ecuación característica por la cual se puede predecir la respuesta del sistema; donde $G_p(s)$ puede ser cualquiera de las combinaciones del proceso.

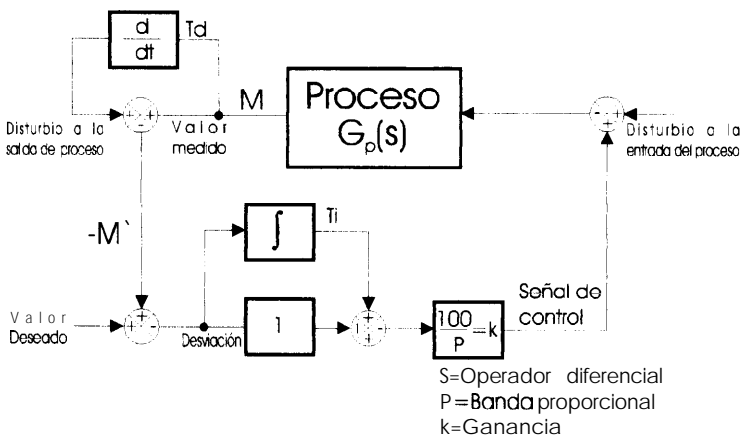
La diferencia con un controlador **PID** convencional es el término $k \cdot T_d \cdot s$ en el numerador de la ecuación 2.20, el cual es el término que da origen a una gran señal de control cuando ocurre un cambio en s .

2) Controlador proporcional + integral + retroalimentación derivativa.

Esta configuración difiere de la anterior sólo en que el término derivativo es sumado al valor medido y por lo tanto, regresa parte de la desviación en lugar de ir al circuito suma, el cual genera la señal de control como se puede apreciar en la figura 2.12.

FIGURA 2.12

CONTROLADOR P + I + RETROALIMENTACIÓN DERIVATIVA



Esta es una diferencia importante que tiene un considerable efecto en el comportamiento del sistema.

La ecuación característica para este sistema se obtiene entonces:

$$M' = -M(1 + T_d * s)$$

$$M = -k * G_p(s) * E * (1 + \frac{1}{T_i s}) \text{ donde } E = M - S$$

$$M = -(M' + S) = M(1 + T_d * s) - S$$

Substituyendo E y reordenando tenemos:

$$\frac{M}{S} = G(s) = \frac{k * (1 + \frac{1}{T_i s}) * G_p(s)}{1 + k * (1 + T_d s) * (1 + \frac{1}{T_i s}) * G_p(s)} \tag{2.22}$$

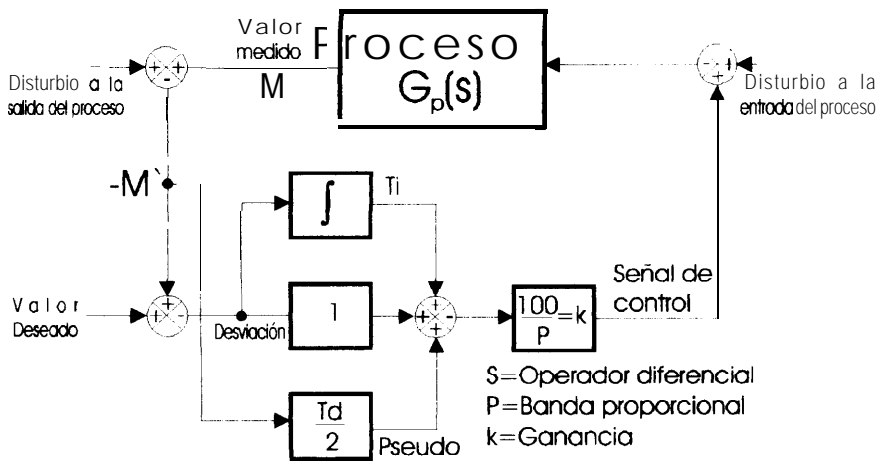
En este caso es interesante notar que la derivada del valor medido vuelve parte de la desviación y pasa subsecuentemente a través del término **integral**. El resultado efectivo es que la señal original es recuperada y esto domina a la idea que el mismo efecto podría ser obtenido sin el uso de un **diferenciador** por la adición en parte del valor medido directamente dentro del circuito suma como se verá en la próxima configuración.

3) Controlador Proporcional mas Integral mas retroalimentación Pseudo - Derivativa.

Para esta configuración, el término derivativo se lo toma del inverso del valor medido, tal como se aprecia en la figura 2.13

FIGURA 2.13

CONTROLADOR P + I + RETROALIMENTACIÓN PSEUDO - DERIVATIVA



En este diagrama, el término $T_d / 2$ representa una simple proporción del valor medido de acuerdo al ajuste del potenciómetro del término derivativo. Por ejemplo, si T_d está ajustado a 1, la proporción del valor medido que alimenta al circuito suma es -0.5 ; procediendo de igual manera, podemos obtener:

$$M = -k * G_p(s) * [E(1 + \frac{1}{T_i s}) - M(\frac{T_d}{2})] \text{ donde } E = M - S$$

Substituyendo y reordenando tenemos:

$$\frac{M}{S} = G(s) = \frac{k * (1 + \frac{1}{T_i s}) * G_p(s)}{1 + k * (1 + \frac{1}{T_i s} + \frac{T_d}{2}) * G_p(s)} \quad 2.23$$

Nótese que hay ahora un circuito no diferenciado para el término derivativo, con la consecuente ventaja de independencia y problemas de ruido.

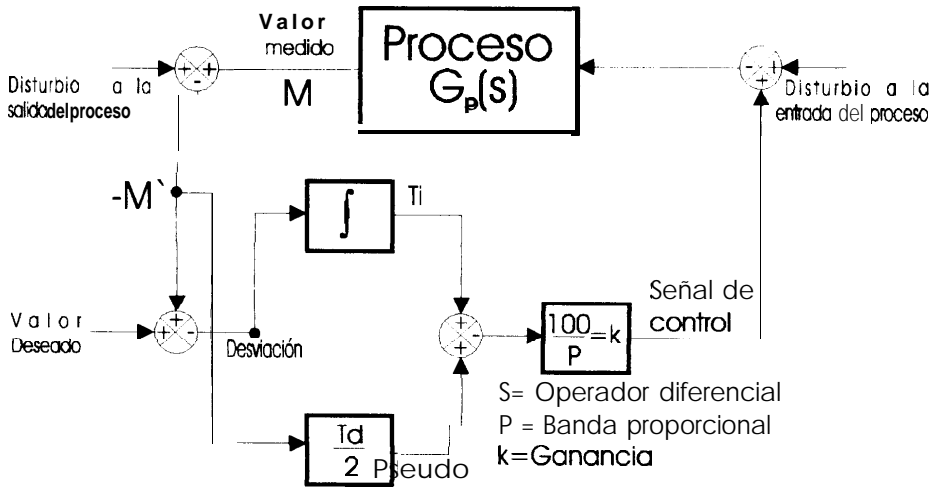
Para algunos procesos esta configuración da **resultado el cual es superior en velocidad de respuesta e independencia de sobrepasos obtenidos** de cualquiera de los arreglos convencionales o no convencionales vistos anteriormente para controladores **PID**.

4) Control Integral mas retroalimentación Pseudo - Derivativa.

Este arreglo es idéntico al anterior con la excepción de que ahora no existe el término proporcional.

FIGURA 2.14

CONTROL INTEGRAL + RETROALIMENTACIÓN PSEUDO - DERIVATIVA



La ecuación característica para este sistema es:

$$\frac{M}{S} = G(s) = \frac{k * (\frac{1}{T_i s}) * G_p(s)}{1 + k * (\frac{1}{T_i s} + \frac{T_d}{2}) * G_p(s)} \quad 2.24$$

Para algunos procesos esto transforma la salida para tener igual función que la configuración 3.

CAPITULO 3

ADQUISICIÓN DE DATOS POR COMPUTADORA

En vista de la necesidad de visualizar de una manera mas detallada los resultados del sistema de control sobre el proceso, se pensó en la **graficación** de las señales del simulador en un monitor de computadora, reemplazando a un registrador en un sistema real, con la ventaja de que en este caso, las señales son grabadas en un archivo numérico para su posterior análisis. Para ello se necesita construir una tarjeta electrónica de interfase entre el simulador y la computadora, utilizando en esta última el puerto paralelo para la comunicación con la tarjeta de interfase.

La comunicación entre la computadora y cualquier periférico puede ser en serie o en paralelo; obviamente la comunicación **entrada/salida** en paralelo es más rápida que en serie, razón por la cual se escogió este puerto.

3.1. El puerto paralelo

El puerto paralelo es un estándar especialmente diseñado para conectar de un ordenador a una impresora, y que dicha interfaz utiliza la transmisión en paralelo de los datos que han de imprimirse por la impresora, disponiendo de varias señales que se utilizan para que el computador controle a la misma y, además, reciba otras señales que envía la impresora al computador para indicarle en que estado se encuentra para un correcto funcionamiento de la comunicación.

El puerto paralelo **LPT(n)** posee un conector de 25 pines en el que se disponen las señales de transmisión, las cuales son compatibles con los niveles **TTL**. En la organización del bus del PC, cada uno de los puertos de la impresora está dividido en tres buses de 8 bits cada uno, en el que cada uno realiza una función bien definida: el bus de datos, estado y control, teniendo cada uno de ellos su propia dirección para su selección.

El sistema MS-DOS puede soportar hasta tres de estos puertos que son: LPT1, LPT2 y LPT3; la correspondencia de estos tres puertos, tomando como referencia el puerto LPT1 se desarrolla en la siguiente tabla, de manera que la dirección de cada parte del mapa **entrada/salida** de LPT1 corresponde con otra dirección LPT2

ó LPT3 con la misma función debido a que se comportan exactamente de la misma forma.

TABLA 3.1

DIRECCIONES DEL PUERTO PARALELO DE UN PC.

PUERTO	DATOS	ESTADO	CONTROL
LPT 1	378	379	37A
LPT2	278	279	27A
LPT3	3BC	3BD	3BE

- **Puerto 378:**

Este es el puerto en el que se escriben los datos para enviar hacia la impresora, estos datos aparecen en el conector de salida. Todos estos datos conforman el byte de información que se envía a la impresora, dichos datos deben formar parte de su código de caracteres de impresión, o de los códigos de escape.

El puerto, obviamente es de salida, aunque se podrá leer solamente para la comprobación de los datos enviados; pero, recordemos que se está hablando de un puerto sólo de salida, no bidireccional.

- **Puerto 379:**

Es un puerto de entrada, en el que se reflejarán las salidas de **estatus** de la impresora. Como se podrá ver en la referencia 4, no se usan todos los bits de

los que dispone este puerto, ya que son cinco las señales previstas en esta **interfaz** para señalar el estado en que se encuentra la impresora.

- **Puerto 37A:**

En este puerto se escriben las señales que van a controlar la impresora; como en el caso anterior, no se van a utilizar todos los bits de dicho puerto, ya que son cuatro las salidas que se disponen en la interfaz **LPT(n)** para que el computador controle la impresora.

3.2. Tarjeta convertidora analógica digital

Esta tarjeta nos permitirá convertir magnitudes análogas de voltaje (como las que se pueden obtener de cualquier punto del simulador) en valores digitales, los cuales pueden ser capturados y procesados por un computador.

Por ser el computador un elemento totalmente digital, este no puede recibir directamente magnitudes análogas como corriente, voltaje, presión, velocidad, fuerza, etc., para realizar **algún** tratamiento de tipo estadístico, gráfico o de almacenamiento de datos, entre otros. Para ello, es necesario disponer de algunos elementos que permitan convertir las magnitudes análogas en valores digitales, con los cuales el computador sí puede realizar las labores propuestas.

Teniendo los valores análogos de voltaje del simulador PCS327 que oscilan en un rango máximo de + 10 Volt. y - 10 Volt., se lo debe convertir en valores

digitales de voltaje que pueden ingresar al computador a través del puerto serial o paralelo o de sus ranuras o “slots” internos que acceden al bus.

En este caso se utilizará el puerto paralelo para ingresar la señal digital al computador, para lo cual se dispondrá entonces de un circuito integrado ADC0808 que nos permite convertir señales analógicas a digitales representadas por un valor binario de 8 bits, que equivalen a valores decimales en un rango de 0 a 256.

3.2.1. Componentes

Como las exigencias de corriente del convertidor, oscilador, “op-amp” y “buffers” son bajas, se acude a la alimentación que proporciona la fuente de un computador, a través del cable “ribbon” que da al tablero de conexiones para PC, en el que se tienen valores de voltajes de 0, 5, -5, 12 y -12 voltios que se utilizan perfectamente para hacer funcionar toda la circuitería de la tarjeta convertidora.

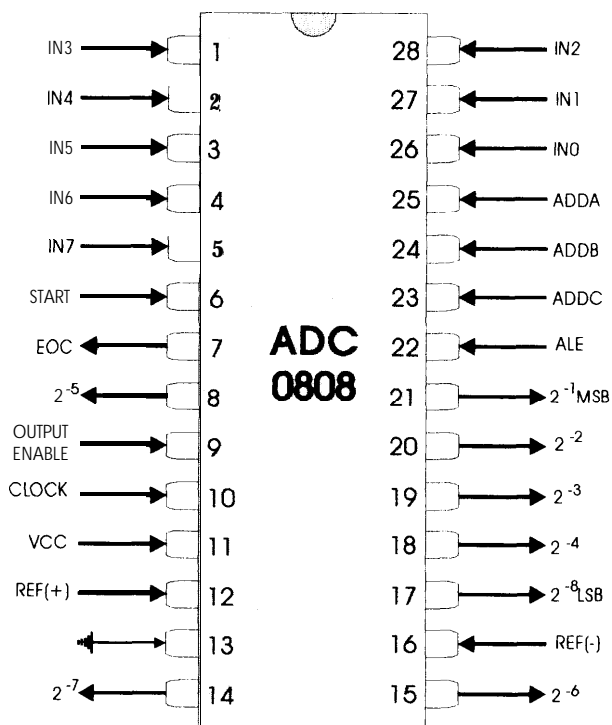
3.2.1.1. El convertidor analógico a digital ADC0808

Uno de los circuitos integrados mas utilizados y conocidos para realizar este proceso es el convertidor análogo a digital ADC0808. Este circuito integrado es un convertidor de 8 bits,

que tiene un multiplexor de 8 canales el cual permite acceder directamente a una de las 8 señales de voltaje.

Entre otras características este circuito tiene un tiempo típico de conversión de $100\ \mu\text{s}$ utilizando un reloj de $500\ \text{kHz}$., utiliza una fuente única de 5 voltios y la circuitería de control y sus salidas se ajustan a los niveles de voltaje de la circuitería TTL. La figura 3.1 muestra la distribución de los pines de este convertidor los cuales tienen las siguientes funciones.

FIGURA 3.1
DISTRIBUCIÓN DE LOS PINES DEL CIRCUITO
INTEGRADO ADC0808



- IN0 a IN7 (entradas) son los 8 canales análogos de entrada, que están multiplexados a un sólo convertidor.
- ADDA a ADDC (entradas) son las tres líneas que permiten seleccionar uno de los 8 canales sobre el cual se va a realizar la conversión. Cuando ellas tienen niveles lógicos bajos (000), el canal seleccionado es el IN0, mientras que cuando tienen niveles lógicos altos (111), el canal seleccionado es el IN7.
- 2^{-1} a 2^{-8} (salidas) son las 8 líneas de salida del convertidor; el primero corresponde al bit más significativo, mientras que el último al menos significativo.
- REF(+) y REF(-) (entradas) son las dos líneas que se utilizarán como referencias para la conversión; ellas conforman una ventana sobre la cual se realizará la conversión: el voltaje al cual se coloque REF(+) estará representado por un valor digital binario de 11111111, mientras que el voltaje sobre el cual se coloque REF(-) lo estará con un valor digital binario de 00000000. En todos casos REF(+) y REF(-) no deben exceder los límites de alimentación del circuito integrado (0 y 5 voltios); usualmente

ellos se colocarán entre estos niveles: REF(-) a OV y REF(+) a 5V.

- ALE “Address **Latch** Enable” o habilitador de fijación de la dirección (entrada) Este fija el dato que se encuentra sobre las líneas ADDA a ADDC, para seleccionar uno de los 8 canales a convertir.
- START (entrada) es la línea con la cual se le ordena al circuito integrado para que empiece una conversión.
- EOC “end off **conversion**” (salida) es la línea sobre la cual el convertidor indica que ha finalizado una conversión y que el dato está listo en las salidas para que sea leído por el sistema. Esta salida puede activar también una interrupción en algún sistema con microcontrolador o microprocesador.
- OUTPUT ENABLE (entrada) es el habilitador para que el convertidor coloque el dato sobre las salidas ya que este dispositivo tiene salidas de alta impedancia.
- CLOCK es la entrada de la señal de reloj para el circuito integrado. Esta señal es indispensable para impulsar la

circuitería interna del convertidor, el cual utiliza el método de aproximaciones sucesivas.

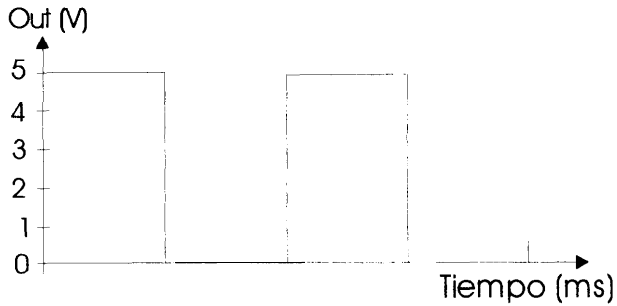
- VCC y GND (entradas) corresponden a la alimentación del circuito, la cual utiliza una fuente de 5 voltios.

3.2.1.2. El generador de onda cuadrada (reloj)

Como hemos visto en el capítulo anterior, el convertidor analógico digital ADC0808 necesita para su funcionamiento la alimentación de una señal pulsante de onda cuadrada como lo indica la figura 3.2 con una amplitud de voltaje de 0 a 5 voltios. De acuerdo con el manual del convertidor, esta señal debe estar dentro de un rango de frecuencias de 10 a 1200 kHz., dependiendo entonces de esta frecuencia la velocidad de conversión o de trabajo del convertidor, debido a que por cada pulso de reloj, realiza la conversión de un bit del byte de salida de este, necesitando entonces 8 pulsos de reloj para la conversión de un nivel de voltaje analógico a un valor digital proporcional entre sus valores de referencia.

FIGURA 3.2

ONDA CUADRADA

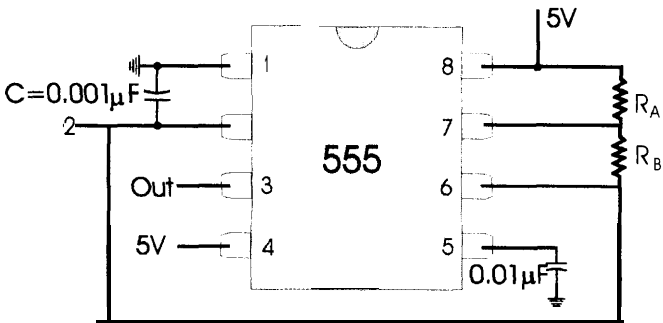


Para obtener esta señal, lo primero que se ocurre pensar es obtenerla del propio computador a través de los “slots” internos que acceden a las tarjetas, pero se realizó una medición con ayuda del osciloscopio y se comprobó que estas señales de reloj tienen frecuencias que andan por el orden de varios MHz.

La otra solución es construir un generador de onda cuadrada con la ayuda del chip 555 que es un circuito integrado temporizador. Las conexiones de las resistencias, capacitancias y voltaje de alimentación o polarización “Vcc” se visualizan en la siguiente figura:

FIGURA 3.3

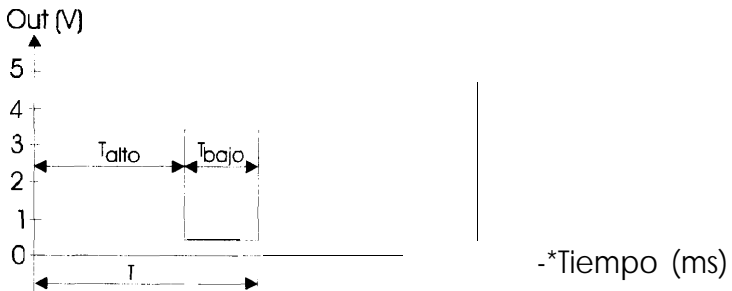
DIAGRAMA DE CONEXIÓN DEL RELOJ



De la cual por el pin # 3 se obtiene la onda cuadrada con la forma que lo indica la figura 3.4

FIGURA 3.4

ONDA CUADRADA GENERADA POR EL OSCILADOR COSTRUIDO CON EL CHIP 555



El cálculo de los intervalos de tiempo durante los cuales la salida es alta y baja puede hacerse empleando las relaciones obtenidas de la referencia 3:

$$T_{alto} \cong 0.7 * (R_A + R_B) * C \quad 3.1$$

$$T_{bajo} \cong 0.7 * R_B * C \quad 3.2$$

El periodo total es:

$$\text{Periodo} = T \cong T_{alto} + T_{bajo} \quad 3.3$$

La frecuencia del circuito estable se calcula empleando la siguiente fórmula:

$$F \cong \frac{1}{T} \cong \frac{1.44}{(R_A + 2R_B) * C} \quad 3.4$$

En este caso para lograr una mayor rapidez de conversión, se utiliza los siguientes valores de resistencia y capacitancia:

$$R_A = 560 \, \Omega$$

$$R_B = 500 \, \Omega$$

$$C = 0.001 \, \mu\text{F}.$$

de donde la frecuencia es:

$$F = 923 \, \text{kHz}.$$

3.2.1.3. Amplificadores operacionales (op-amp)

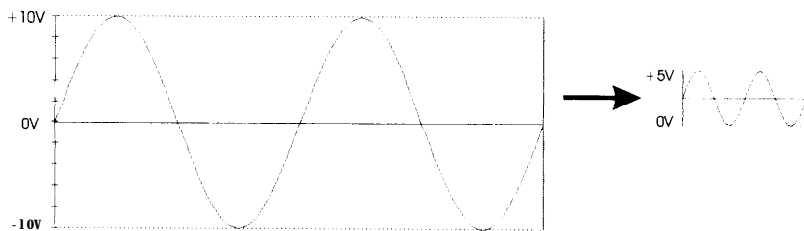
La señal de voltaje variable que se va a convertir usando el ADC0808 necesita estar en un rango máximo de 0 a 5 voltios,

pero estas señales son extraídas del simulador PCS-327 con valores que oscilan entre + 10 y -10 voltios.

El amplificador operacional es un circuito integrado “chip 741” que en una de sus muy variadas aplicaciones, nos permite cambiar la escala de una señal de voltaje variable en un rango de + 10 y - 10 voltios a un rango proporcional entre 0 y 5 voltios tal como lo indica la siguiente figura:

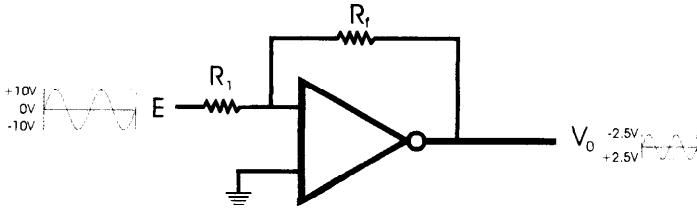
FIGURA 3.5

FUNCIÓN DEL AMPLIFICADOR OPERACIONAL



Las conexiones para el chip 741 “op-amp” se muestran en la figura 3.6. Como se indica, el circuito opera como un cambiador de escala o un multiplicador de ganancia constante y como un inversor, es decir, tiene sus ganancias negativas.

FIGURA 3.6
 CONEXIÓN BÁSICA DE UN AMPLIFICADOR OPERACIONAL



Los resultados muestran que la relación de salida total al voltaje de entrada sólo depende de los valores de las resistencias R_1 y R_f (siempre que la magnitud de la ganancia sea grande).

$$\text{Ganancia} = A_u = \frac{V_0}{E} = \frac{R_f}{R_1} \quad 3.5$$

Para este caso se necesita un circuito que tenga una ganancia de 4, ya que se requiere reducir el intervalo de voltajes a la cuarta parte:

Amplitud del voltaje de entrada = $10 - (-10) = 20$ voltios.

$$\text{Amplitud del voltaje de salida} = \frac{20}{A_u} = -\frac{20}{4} = -5 \text{ voltios}$$

utilizando la relación 3.5 obtenida de la referencia 3 tenemos:

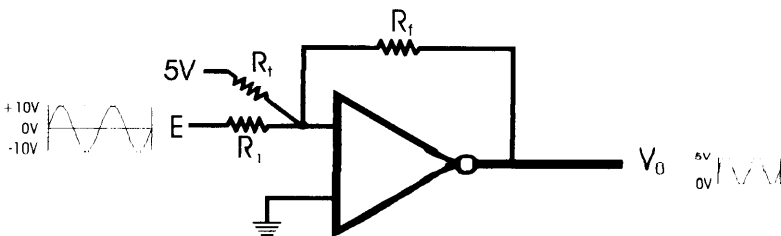
$$\frac{5}{20} = -\frac{R_f}{R_1} = -\frac{1}{4}$$

$$R_1 = -4R_f$$

Pero como se necesita que el voltaje se encuentre entre los rangos de 0 y 5 voltios, se ha sumado al circuito un voltaje de 5 voltios con una resistencia R_t que así mismo obedecen a la misma relación (3.5), de tal forma que la conexión del circuito es como lo indica la siguiente figura 3.7:

FIGURA 3.7

DIAGRAMA DE CONEXIONES DEL “OP-AMP”



$$V_0 = V_{01} + V_{02} \quad 3.6$$

donde:

$$V_0 = -\frac{R_f}{4R_f} E = -\frac{E}{4} \quad 3.7$$

$$\frac{V_{02}}{V_{cc}} = \frac{R_f}{R_t}$$

$$\text{Si } R_t = 2R_f \Rightarrow V_{02} = -\frac{R_f}{2R_f} 5 = -2.5 \text{ voltios}$$

Reemplazando en la ecuación 3.7

$$V_0 = -\frac{E}{4} - 2.5 \text{ (voltios)}$$

En vista de que la ganancia es negativa, a la salida del “op-amp” tenemos una señal invertida (0 a -5 voltios), por tal razón se requiere volver a invertir esta señal para obtener valores entre 0 y 5 voltios, para lo cual utilizamos el mismo circuito básico de conexión del “op-amp” (fig. : 3.6) con ganancia unitaria, requiriéndose para ello únicamente que los valores de las resistencias R_f y R_t sean iguales.

Para esto los valores requeridos de las resistencias deben cumplir los siguientes requisitos:

$$R_t = 2R_f$$

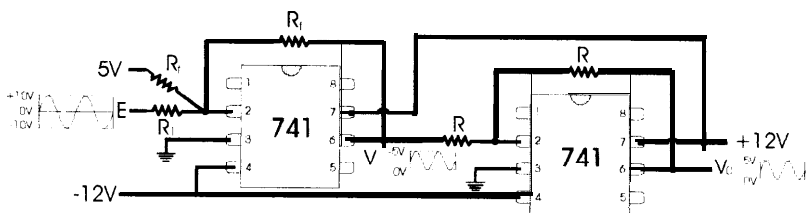
$$R_t = 4R_f$$

$$v_{cc} = 5V \text{ dc.}$$

Por lo tanto el diagrama de la circuitería para un cambiador de escala como el que nosotros utilizaremos es el siguiente:

FIGURA 3.8

DIAGRAMA COMPLETO DEL CIRCUITO “OP-AMP” CAMBIADOR DE ESCALA



3.2.1.4. “Buffers” de colector abierto.

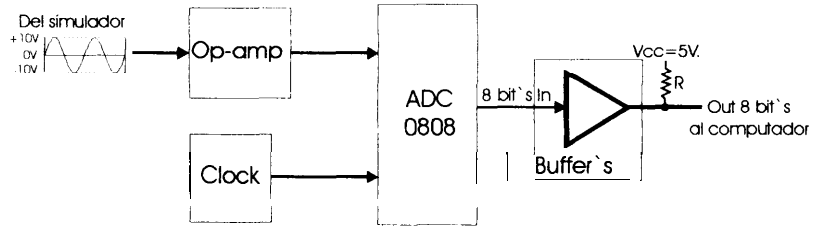
Estos circuitos integrados tienen la función de amplificar la señal de corriente que sale del convertidor ADC0808 para poder ser ingresada al puerto paralelo del computador. Debido a que las **tarjetas IDE** estándar que se está utilizando no disponen de una entrada directa de un “byte” (8 bits) al PC, se está ingresando esta señal por el bus de datos (puerto 378) de tal manera que se está forzando la entrada al computador.

Los “buffers” que se están utilizando son de colector abierto de tal forma que por medio de una resistencia externa se puede regular el nivel de amplificación de corriente, de tal manera que un esquema de lo que constituye la tarjeta de interfase con todos sus elementos, lo indica la figura 3.9.

En el apéndice K se puede apreciar el diagrama completo de conexiones de la tarjeta convertidora analógico - digital con todos sus elementos.

FIGURA 3.9

DIAGRAMA ESQUEMÁTICO DE LA TARJETA



3.3. Programas de adquisición e interpretación de datos por computadora

Con todos los elementos de “hardware” dispuestos, podemos enfocarnos en la parte de “software”. Inicialmente, pensaremos en un proceso simple: dar la orden de conversión al convertidor, esperar un tiempo prudencial hasta que la conversión haya finalizado ($>100\mu\text{s}$), leer el resultado de la conversión y mostrar en la pantalla de manera gráfica este valor; este proceso se realizará continuamente, de tal manera que podamos visualizar la forma de onda de la señal que se está leyendo. Como resultado, este programa convierte la pantalla del computador en un osciloscopio; en ella se pueden visualizar las formas de onda de la señal y su amplitud relativa.

El programa considera que la dirección base de la tarjeta se encuentra a partir de la posición 378 en hexadecimal, el cual corresponde al bus de datos que sirve como entrada de datos; este “byte” de información es el que proviene directamente del convertidor ADC0808. La posición 379 está configurado también como entrada de datos, y de ésta se utiliza sólo el pin # 15 que es por

donde el computador recibe la señal del convertidor cuando este ya ha finalizado el trabajo de conversión (EOC). La posición 37A está configurado en cambio como salidas y es por donde el computador envía la señal al convertidor de habilitación "ouput enable", inicio de conversión "start " y selección de canal analógico "ADDA".

La figura 3.10 muestra el diagrama de flujo sólo para el proceso de intercambio de información entre el computador y la tarjeta convertidora analógico - digital.

FIGURA 3.10
DIAGRAMA DE FLUJO PARA LA UTILIZACIÓN DEL CONVERTIDOR
ADC0808



Es obvio que el lenguaje BASIC no es el más adecuado para realizar este proceso, y peor aún si se tiene una versión interpretada. El lenguaje C tampoco es el más adecuado, ya que la compilación agrega una cantidad de códigos que toma lenta la ejecución del programa y limita la frecuencia máxima con la cual se puede visualizar la lectura. Lo ideal es utilizar el lenguaje ensamblador, con lo cual se consigue un código más eficiente, realizando una lectura más directa y por lo tanto más rápida.

Sin embargo, el realizar el programa en lenguaje BASIC es efectivo en algunas situaciones en donde un osciloscopio resultaría inadecuado; por ejemplo, la forma de las ondas con frecuencias ultrabajas (como las que se producen en el simulador PCS327) no pueden ser visualizadas adecuadamente en un osciloscopio, debido a la poca persistencia de su imagen (sólo se ve que la traza sube y baja, pero no se puede identificar a primera vista se trata de una onda senoidal, triangular, diente de sierra, etc.). Pero con el computador la duración de las imágenes es controlable, aún las formas de ondas más lentas se visualizan perfectamente. Además se tiene la ventaja de que en un computador estas señales se las pueden archivar para análisis posteriores o para mantenerlas como documento.

3.3.1. Adquisición de datos usando QBASIC

El programa completo consta de 3 opciones principales que son:

Graficar en pantalla AMPLITUD vs TIEMPO	1
Graficar en pantalla y GRABAR datos de AMPLITUD vs TIEMPO	2
Graficar DIAGRAMA DE FASE Y RELACIÓN DE AMPLITUD	3
SALIR	4

INGRESE OPCIÓN (1 a 4):

La opción “1” permite **graficar** en el monitor y en tiempo real los datos de “amplitud vs tiempo” de dos señales diferentes a la vez, las cuales se las diferencia por el color de línea en el monitor. También nos permite ajustar la velocidad de barrido de la pantalla escalada en valores del 1 al 10; o sea, para procesos con cambios rápidos de amplitud se ajusta la velocidad de barrido en valores cercanos a 10, y para procesos lentos, a la inversa. Cuando se desee parar el proceso, se oprime la tecla F4, y la imagen que se tiene en pantalla se mantiene hasta oprimir cualquier otra tecla.

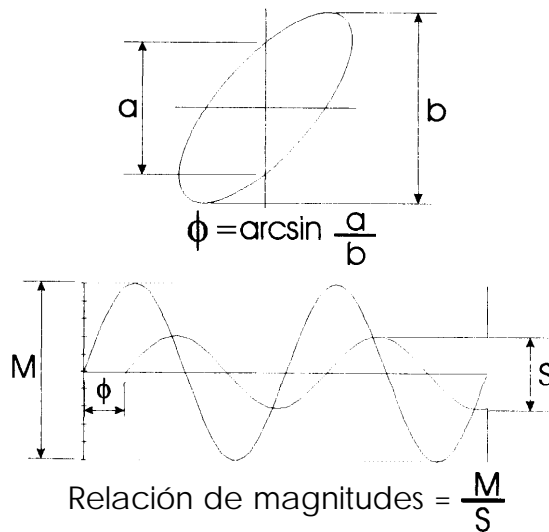
La opción “2” en cambio, a más de realizar el mismo trabajo que la opción “1”, nos permite además guardar en archivos independientes los valores que se están **graficando** tanto de amplitud como de tiempo, tomando como punto de partida para este último ($t = 0$) el primer dato

que se lee hasta que se detiene el proceso oprimiendo la tecla F4. Estos archivos contienen los datos guardados en una matriz de una sola columna por n filas, donde $0 < n < 600$, que son datos suficientes como para poder hacer cualquier análisis de estos valores.

La opción “3” nos permite graficar en pantalla la relación de amplitudes de las dos señales y a su vez el ángulo de desfase entre ellas, que se lo puede medir usando la relación descrita en la figura 3.10 cuando los dos grupos de datos **graficados** corresponden a señales senoidales. Esta opción es muy útil para realizar los diagramas de Nyquist y de Bode, que son en los cuales se necesita además saber el ángulo de desfase entre estas dos señales.

FIGURA 3.11

CÁLCULO DEL ÁNGULO DE DESFASE Y RELACIÓN DE MAGNITUDES DE DOS SEÑALES SENOIDALES



Para realizar las mediciones correspondientes, es necesario imprimir el gráfico de la pantalla, para lo cual en el sistema MS-DOS se carga “graphics” antes de ejecutar el programa en BASIC, y luego después de corrido el programa, con un “print screen” se envía la imagen de la pantalla a la impresora conectada al mismo puerto por intermedio de un “data-switch”, el cual se lo coloca en la posición de la dirección de envío de datos, es decir, del computador a la impresora o del computador a la tarjeta de control. En el sistema WINDOWS es posible capturar esta imagen oprimiendo las teclas “Alt + print screen”, imagen se queda guardada en el portapapeles y se la puede pegar usando cualquier programa gráfico como el “Paint” que viene incluido en el mismo Windows.

En el apéndice A podemos apreciar el programa PCS327.BAS con todas sus cuatro subrutinas que corresponden a las cuatro opciones del menú principal explicadas anteriormente.

En los apéndices B y C se encuentran impresos los programas FOURIER y LEVY que son una reproducción del original descritas en la referencia 4 con algunas modificaciones, especialmente en la introducción y lectura de datos.

El programa FOURIER.BAS nos permite cambiar los datos de amplitud de la respuesta a una entrada escalón en función del tiempo $A=f(t)$ a función de frecuencia dominante $A=f(i\omega)$ usando la transformada de Fourier.

El programa LEVY.BAS usa un método modificado de mínimos cuadrados para aproximarse a la curva de respuesta del sistema en términos de frecuencia dominante y así estimar la función de transferencia.

3.3.2. Uso de MATLAB para análisis de datos.

El programa “MATLAB” es una herramienta matemática muy poderosa que nos permite trabajar fácilmente con matrices, números complejos, etc.. Para que los datos tomados utilizando la tarjeta convertidora analógico a digital y empleando como lenguaje de programación el QBASIC sean leídos o interpretados como un arreglo por el programa “MATLAB”, es necesario que el nombre del archivo a grabarse tenga extensión **m** (**nombre.m**); con un archivo de estas características, se lo puede abrir directamente desde “MATLAB” con el siguiente comando:

load dirección nombre.m (ENTER ↵)



en donde el nombre del archivo define a la matriz de datos, por ejemplo: si el nombre que se le dio al archivo del tiempo se lo grabó con el nombre de *T.m*, al abrirlo este archivo en “MATLAB” con la instrucción *load dirección T.m*, la matriz *T* define todos los datos guardados en ese archivo. Para visualizar estos datos en pantalla basta sólo con teclear *T(ENTER ↵)*.

Si se desea **graficar** los datos de amplitud vs tiempo siendo amplitud (*A.m*) una matriz con el mismo número de filas y columnas que (*T.m*), se lo hace con el siguiente comando:

plot (A, T) (ENTER ↵).

CAPITULO 4

ELABORACIÓN DE GUIAS DE LABORATORIO

En este capítulo se detallan cuatro guías de prácticas del laboratorio de sistemas de control, en la que se las ha elaborado en base a valores típicos de ajuste del controlador y del proceso para hacer más comprensible para el estudiante el desarrollo de la práctica y más preciso en la obtención de los datos. Cabe destacar que estos valores típicos se los ha encontrado en base a la experimentación y análisis de errores para cada una de las prácticas; pudiéndose sin embargo, realizar experimentos con otros valores de ajuste para el controlador y el proceso con errores menores al 8%.

Los datos y resultados de cada una de las prácticas descritas en los subcapítulos 4.1, 4.2, 4.3 y 4.4 se exponen en los apéndices referidos al final de cada subcapítulo.

Práctica # 1

SIMULACIÓN DE SISTEMAS PROPORCIONALES DE PRIMERO, SEGUNDO Y TERCER ORDEN.

Objetivo:

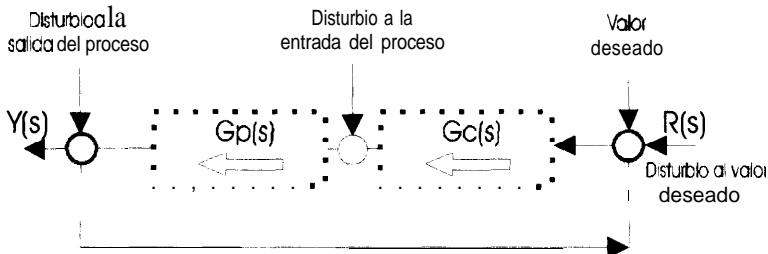
Dar a conocer al estudiante las respuestas a una entrada escalón de sistemas proporcionales de primero, segundo y tercer orden utilizando el simulador de procesos de control PCS327. También se propone encontrar la función de transferencia de un proceso desconocido haciendo uso del método de Levy.

Fundamentos teóricos:

Considerando el diagrama de bloques que representa al simulador de PCS327 que se encuentra en la figura 4.1.1 tenemos que la función de transferencia del sistema es:

FIGURA 4.1.1

DIAGRAMA DE BLOQUES DEL SIMULADOR PCS327



$$\mathbf{G(s)} = \frac{\mathbf{k * G_p(s)}}{1 + \mathbf{k * G_p(s)}} \quad 4.1.1$$

donde:

$$G_p(s) = \frac{1}{\tau s + 1} \quad \text{para procesos de primer orden.} \quad 4.1.2$$

$$G_p(s) = \frac{1}{(\tau s + 1)^2} \quad \text{para procesos de segundo orden.} \quad 4.1.3$$

$$G_p(s) = \frac{1}{(\tau s + 1)^3} \quad \text{para procesos de tercer orden.} \quad 4.1.4$$

reemplazando en la ecuación 4.1.1 tenemos la siguientes funciones de transferencia que representan a los sistemas:

$$G(s) = \frac{k}{\tau s + k + 1} \quad \text{para sistemas de primer orden.} \quad 4.1.5$$

$$G(s) = \frac{k}{(\tau s)^2 + 2\tau s + k + 1} \quad \text{para sistemas de segundo orden} \quad 4.1.6$$

$$G(s) = \frac{k}{(\tau s)^3 + 3(\tau s)^2 + 3\tau s + k + 1} \quad \text{para sistemas de tercer orden} \quad 4.1.7$$

donde $k = \frac{100}{BP}$ y $\tau = \begin{cases} 1 \text{ en la posición "slow"} \\ 0.01 \text{ en la posición "fast"} \end{cases}$

Método de Levy:

Normalmente los procesos industriales son de alto orden y posiblemente no lineales, lo que dificulta la aplicación de técnicas estándares de diseño. El método de Levy es una técnica que nos permite la identificación de un proceso desconocido, es decir, podemos encontrar un modelo matemático de bajo orden que represente a ese proceso.

Este método se basa en la respuesta a una entrada escalón, en donde usa un método modificado de mínimos cuadrados para aproximarse a la curva de respuesta del sistema en términos de frecuencia dominante, por lo tanto, para utilizar este método se debe primero pasar los datos de la respuesta en función de tiempo dominante a función de frecuencia dominante.

Experimentación:

- Ajuste todos los switches y controles como lo indica en los gráficos 4.1.2 y 4.1.3. y el potenciómetro del valor deseado indicando (-5) voltios y los switches de control de velocidad en la posición “slow” y la banda proporcional $BP = 100$.
- Con la ayuda de un generador de funciones inserte una onda cuadrada de 10 voltios pkpk a baja frecuencia de tal forma que parezca una entrada escalón a “set value disturbance” y grabe los datos de amplitud y tiempo en archivos con extensión **.m** (nombre.m).

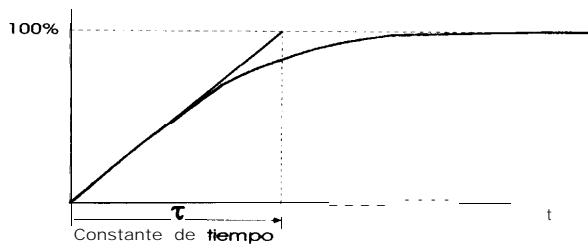
- Repita el procedimiento descrito anteriormente para cada una de las configuraciones del proceso dadas en el gráfico 4.1.2.

Procedimiento:

- Utilizando el programa “Matlab” plotee e imprima el gráfico para encontrar la constante de tiempo (τ) del proceso, tal como lo indica la figura 4.1.4.

FIGURA 4.1.4

CÁLCULO DE LA CONSTANTE DE TIEMPO PARA UN RETARDO EXPONENCIAL



- Edite los archivos obtenidos del experimento anterior y elimine los transientes desde $t=0$ hasta el momento en que se dio la función escalón, y verifique que los 3 archivos de cada proceso tengan el mismo número de datos.
- Utilice el programa “Fourier.bas” para pasar los datos de respuesta en tiempo dominante a frecuencia dominante, donde el rango de frecuencias se lo podría estimar basándose en la frecuencia de corte $fc=1/\tau$, donde: $0.1*fc < \omega_n < 2*fc$.

- Con los datos obtenidos del programa “Fourier.bas”, utilice el programa “Levy.bas” para encontrar la función de transferencia.
- Use el comando “tftplot” del programa “MATLAB” para graficar la respuesta a una entrada escalón de una función de transferencia dada, tanto para la teórica deducidas en las ecuaciones 4.1.5, 4.1.6 y 4.1.7, como para las encontradas con el método de “Levy”.
- Compare y analice los atributos y las raíces de los 3 gráficos obtenidos para cada experimento realizado.

FIGURA 4.1.2

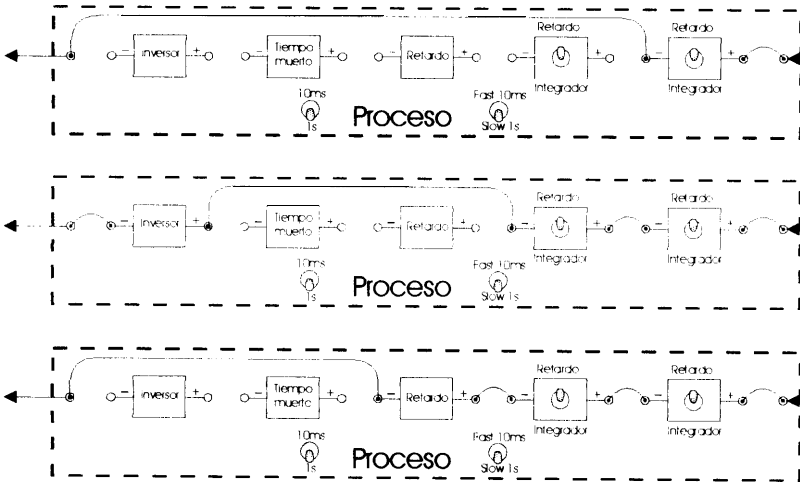
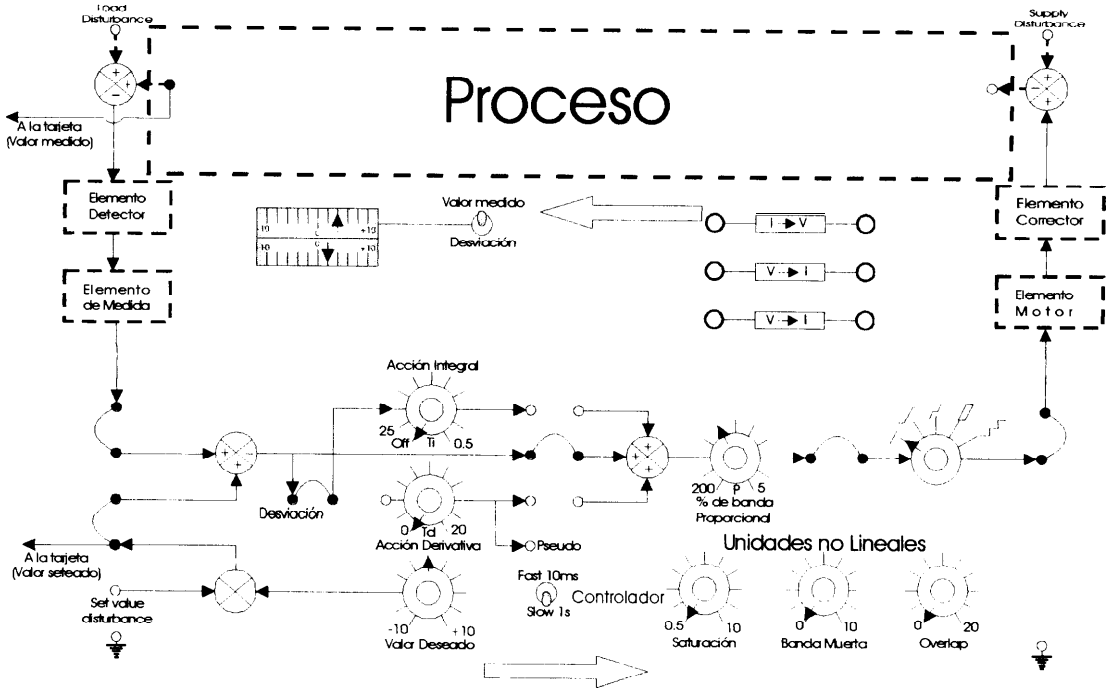


FIGURA 4.1.3



Práctica # 2

CONTROLADORES PROPORCIONAL, P+I Y P+I+D.

Objetivo:

Determinar las diferencias del comportamiento de controladores P, PI y PID ante una entrada escalón al valor deseado y a la vez comparar estos gráficos con los obtenidos a partir de la función de transferencia del sistema.

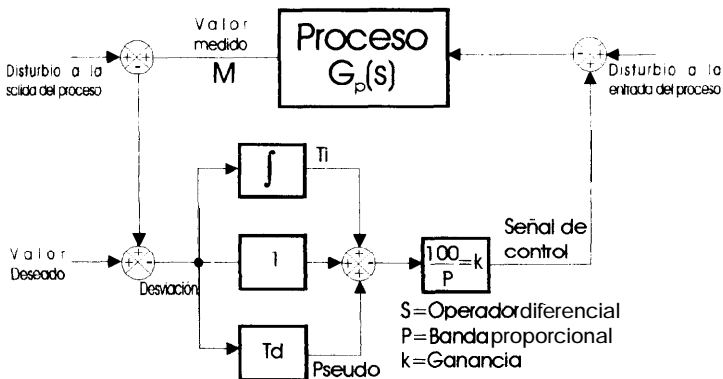
Fundamentos teóricos

Funciones de transferencia.

- Sistema proporcional:

FIGURA 4.2.1

DIAGRAMA DE BLOQUES DEL SIMULADOR PCS327



- **Sistema PID**

Si se conecta adicionalmente el modo integral y el modo derivativo del diagrama de la figura 4.2.1 tenemos que la función de transferencia del sistema es:

$$G(s) = \frac{k * (1 + \frac{1}{T_i s} + T_d s) * G_p(s)}{1 + k * (1 + \frac{1}{T_i s} + T_d s) * G_p(s)} \quad \text{para sistemas de tercer orden} \quad 4.2.5$$

de igual forma reemplazando $G_p(s)$ en la ecuación anterior tenemos:

$$G(s) = \frac{k T_i T_d s^2 + k T_i s + k}{T_i s^4 + 3 T_i s^3 + T_i s^2 (3 + k T_d) + T_i s (1 + k) + k} \quad 4.2.6$$

Experimentación:

- Ajuste todos los switches y controles como lo indica el gráfico 4.2.2 con los potenciómetros de banda proporcional, tiempo integral y derivativo ajustados con los valores que muestra la siguiente tabla:

TABLA 4.1

AJUSTES DEL CONTROLADOR PARA LA PRACTICA # 2.

Controlador P	Controlador PI	Controlador PID
BP = 100 % / %	BP = 60 % / %	BP = 60 % / %
T _i = Off	T _i = 2.5 s.	T _i = 1.5 s.
T _d = 0 s.	T _d = 0 s.	T _d = 0.2 s.

- Con la ayuda de un generador de funciones inserte 10 voltios aproximadamente en el punto “set value disturbance” y grabe los datos de amplitud y tiempo en archivos con extensión **.m** (**nombre.m**).
- Repita este procedimiento para cada una de las configuraciones dadas en la tabla 4.1.

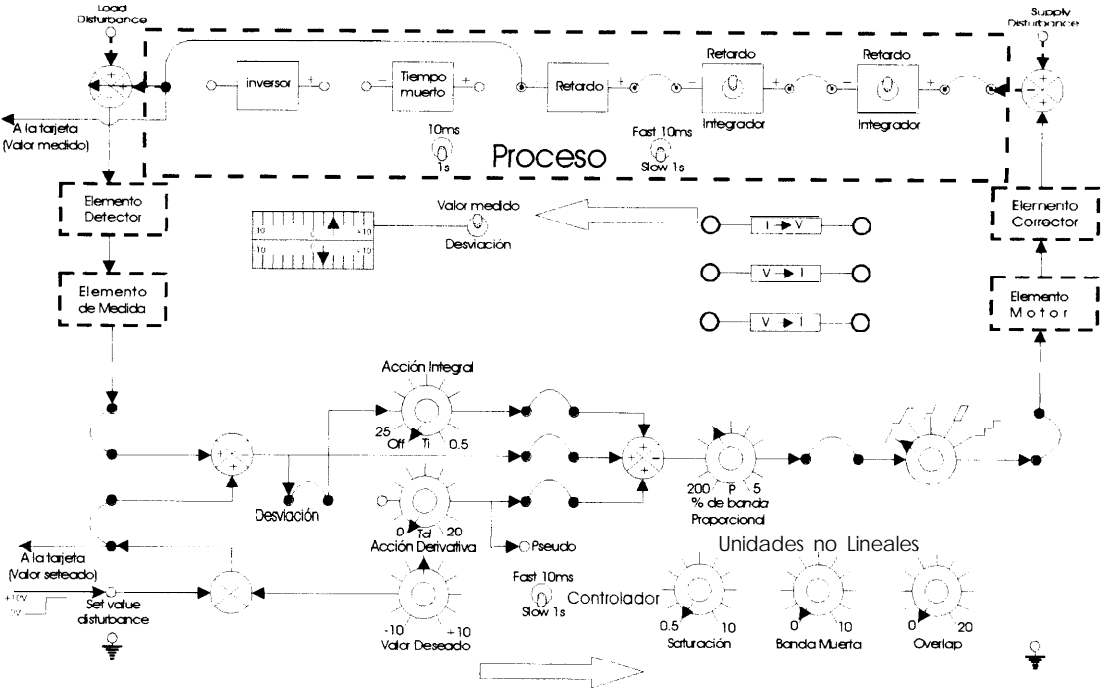
Procedimiento:

- Edite los archivos obtenidos del experimento anterior y elimine los transientes desde $t=0$ hasta el momento en que se dio la función escalón, y verifique que los 3 archivos de cada proceso tengan el mismo número de datos.
- Use el comando “**tftplot**” del programa “Matlab” para **graficar** la respuesta a una entrada escalón de las funciones de transferencia deducidas en las ecuaciones 4.2.2, 4.2.4 y 4.2.6, y plotee los datos obtenidos experimentalmente usando los comandos “load **dirección\nombre.m**” para abrir los archivos y el comando “plot (**nombre 1 ,nombre2**)” para graficarlos.
- Compare y analice los atributos de los 2 gráficos obtenidos para cada experimento realizado como son:
 - % de sobrepaso (overshot)

- Tiempo muerto (delay time)
- Tiempo de ascenso (Rise time)
- Tiempo de asentamiento (settling time)
- Error de estado estable.

FIGURA 4.2.2

DIAGRAMA DE CONEXIONES PARA LA PRACTICA #2



Práctica # 3

RESPUESTA A LA FRECUENCIA EN LAZO ABIERTO.

Objetivo:

Encontrar la función de transferencia de un proceso desconocido, a partir de los diagramas de Bode, y además comparar estos diagramas (Bode y Nyquist) y sus propiedades con los obtenidos a partir de la función de transferencia teórica en lazo abierto.

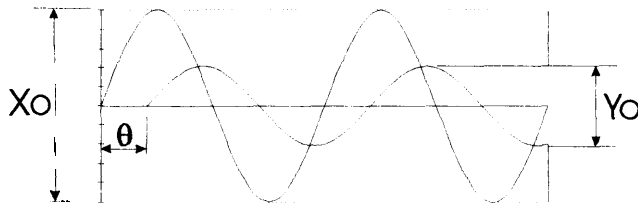
Fundamentos teóricos:

Para realizar estos diagramas es necesario aplicar la prueba senoidal que consisten en mover el valor deseado “set value disturbance” en forma senoidal ($x(t) = X_o \text{sen}(wt)$), lo cual se consigue con un generador de ondas de frecuencia variable; de la misma forma, la respuesta de salida del proceso en lazo abierto se hace senoidal ($y(t) = Y_o \text{sen}(wt + \theta)$), obviamente desfasado un ángulo θ . Para cada valor de la frecuencia (w) de la señal de entrada se obtienen los datos de relación de amplitud (RA) y ángulo de desfase (θ) de la señal de salida que nos indican un punto en estos diagramas.

A continuación se describen algunos términos utilizados en el estudio de la respuesta a la frecuencia indicados en la figura 4.3.1:

FIGURA 4.3.1

GRAFICA DE LA RESPUESTA A UNA ENTRADA SENOIDAL DE RETARDOS EXPONENCIALES



- Razón de amplitud $RA = \frac{Y_o}{X_o}$
- Razón de magnitud $RM = \frac{RA}{k}$ donde $k = \frac{100}{BP}$ es la ganancia de estado estacionario; representado en decibeles: $RM = 20 \text{Log} \left(\frac{RA}{k} \right)$.
- θ es el ángulo de desfase; cuando es positivo, se trata de un ángulo de adelanto, y negativo cuando es un retardo.

Diagramas de Bode:

El diagrama de Bode es la representación gráfica más común de las relaciones RM, θ y w . Este diagrama consta de dos gráficas que son: 1). $\text{Log } RM$ vs $\text{Log } w$ y 2). θ VS $\text{Log } w$.

En estos dos diagramas se puede apreciar claramente la frecuencia de corte que es donde las asíntotas cambian de pendiente en el gráfico: Log RM vs Log ω . Se puede generalizar que la frecuencia última para los siguientes procesos está en función de θ como lo indica la siguiente tabla:

TABLA 4.3.1

Procesos de primer orden	ω_c se da a los 45°
Procesos de segundo orden	ω_c se da a los 90°

En esta práctica se utilizará un proceso de segundo orden con la siguiente función de transferencia en lazo abierto: $G(s) = G_p(s)G_c(s)$

$$G(s) = \frac{k}{(\tau s)^2 + 2\tau s + 1} \quad 4.3.1$$

Sustituyendo $i\omega$ por s :

$$G(i\omega) = \frac{k}{1 - \omega^2 \tau^2 - i2\tau\omega}$$

La razón de amplitud es:

$$RA = |G(i\omega)| = \frac{|G_1|}{|G_2|} = \frac{k}{\sqrt{(1 - \omega^2 \tau^2)^2 + (2\tau\omega)^2}} \quad 4.3.2$$

y el ángulo de fase es:

$$\theta = -\tan^{-1}\left(\frac{2\tau w}{1-w^2\tau^2}\right)$$

4.3.3

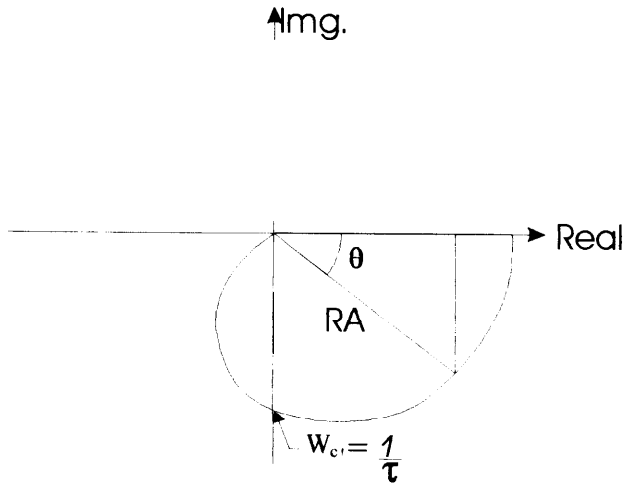
Diagrama de Nyquist:

El diagrama de Nyquist consiste en **graficar** la función compleja $G(iw)$ en un plano real y complejo a la vez. Con éste método se tiene la ventaja de que sólo se hace una sola gráfica que va desde $w = 0$ hasta $w = \infty$.

Para un proceso de segundo orden en que las ecuaciones para RA y θ se representan en las relaciones 4.3.2 y 4.3.3 respectivamente están **graficadas** en el diagrama polar como lo indica la figura 4.3.2.

FIGURA 4.3.2

DIAGRAMA DE NYQUIST

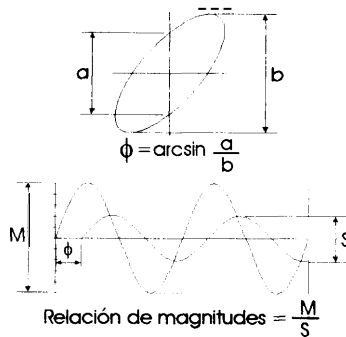


Experimentación:

- Ajuste los swichts y perillas como lo indica el gráfico 4.3.3 de tal manera que se tenga un proceso de segundo orden en lazo abierto con ganancia unitaria (BP -100).
- Utilizando el sistema MS-DOS cargar “graphics” antes de ejecutar el programa en “basic”.
- Insertar una onda senoidal a “set value disturbance” con frecuencias entre 0.01 a 2 Hz. y amplitud 8 voltios pkpk.
- Incrementar progresivamente la frecuencia de la señal de entrada e imprimir el gráfico en pantalla para cada valor de frecuencia presionando “print screen”.

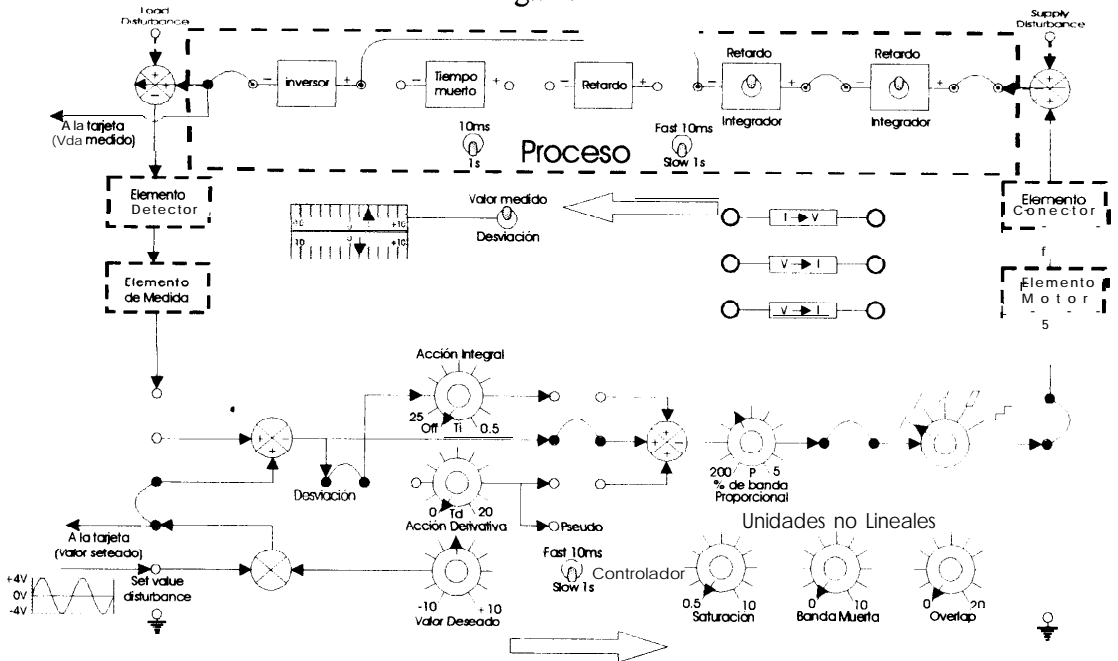
Procedimiento:

- Para cada gráfico calcule el ángulo de desfase (θ), relación de amplitudes (RA) y la frecuencia ω tomando como referencia el tiempo impreso en la hoja.



- Con los datos obtenidos anteriormente grafique **20** Log RA4 vs ω y θ vs ω en gráficos semilogarítmicos en ω y estime la función de transferencia $G(s)=G_p(s)G_c(s)$ a partir del criterio de la pendiente del diagrama de Bode.
- Realice el diagrama polar de Nyquist a partir de los valores de RA y θ tal como lo indica la figura 4.3.2.
- Haga un análisis comparativo entre los diagramas obtenidos experimentalmente y los teóricos basados en la función de transferencia en lazo abierto del proceso (ec. 4.3.1) usando los comandos “bplot y plrplot” del programa MATLAB.

Fig.: 4.3 3



Práctica # 4

MÉTODOS EMPÍRICOS PARA AJUSTE DE CONTROLADORES POR RETROALIMENTACIÓN

Objetivos:

Dar a conocer al estudiante algunos métodos empíricos por el cual pueden encontrar los valores tanto de banda proporcional, tiempo integral y derivativo de controladores para obtener un mejor rendimiento y precisión en el proceso de control.

Parte teórica:

En esta práctica se ven dos métodos de ajuste del controlador, haciendo hincapié de que cada método puede ser válido para cada caso particular en la industria, debiendo el ingeniero de control buscar el que mejor se adapte a su sistema.

Método de la ganancia última:

Este método consiste en producir una oscilación de amplitud constante del valor medido, estando el sistema en automático o lazo cerrado.

Para realizar la práctica por este método se seguirán los siguientes pasos:

1. Ajuste todos los controles e interruptores como se indica en el gráfico 4.4.1 (acción integral y derivativa desconectadas)
2. Con el sistema en lazo cerrado reduzca la banda proporcional (BP) hasta que con un pequeño movimiento del valor deseado, el sistema entre en oscilación continua de amplitud constante.
3. Obtenga la ganancia última (K_{cu}) o banda proporcional última (BP_u) y el periodo de oscilación último (T_u) a partir del gráfico. Calcule y compare K_{cu} y T_u teóricos partiendo de la ecuación característica $(1+k*G_p(s))$ siendo $G_p(s) = \frac{1}{(s+1)^3}$ usando el método de sustitución directa.
4. Con la ganancia y el periodo último, utilice las fórmulas de la tabla 4.4.1 para calcular los parámetros de ajuste del controlador con las cuales se producen respuestas de la razón de asentamiento de un cuarto.
5. Ajuste el controlador y compruebe la respuesta del sistema.

TABLA 4.4.1

MÉTODO DE LA GANANCIA ÚLTIMA (ZIEGLER Y NICHOLS)

Tipo de controlador	Ganancia y banda proporcional $K_c = 100/BP_u$	Tiempo de integración T_i	Tiempo de derivación T_d
P	$2*BP_u$		
P+I	$2.2*BP_u$	$T_u / 1.2$	
P+I+D	$1.7*BP_u$	$T_u / 2$	$T_u / 8$

Método de la curva de reacción:

Este método consiste en cambio en aplicar una entrada escalón al sistema en lazo abierto, y graficar la curva de reacción. El procedimiento a seguir es el siguiente:

1. Ajuste todos los controles e interruptores como se indica en el gráfico 4.4.2 (acción integral y derivativa desconectadas y el sistema en lazo abierto).
2. Aplicar un pequeño cambio escalón al sistema, insertando una señal de +5 voltios en el punto “set value disturbance”, y obtener la curva de reacción del proceso hasta que el valor deseado se estabilice en otro punto.
3. Con el gráfico obtenido anteriormente, medir la constante de tiempo, tiempo muerto y ganancia del proceso.
4. Con los valores obtenidos en el párrafo anterior, utilice las fórmulas de la tabla 4.4.2 para calcular los parámetros de ajuste del controlador con las cuales se producen respuestas de la razón de asentamiento de un cuarto.
5. Ajuste el controlador y compruebe la respuesta del sistema.

TABLA 4.4.2

MÉTODO DE LA CURVA DE REACCIÓN (ZIEGLER Y NICHOLS)

Tipo de controlador	Ganancia y banda proporcional $K_c = 100/BP$	Tiempo de integración T_i	Tiempo de derivación T_d
P	$\frac{1}{K} \left(\frac{t_o}{\tau}\right)^{-1}$		
P + I	$\frac{0.9}{K} \left(\frac{t_o}{\tau}\right)^{-1}$	$3.33 * t_o$	
P + I + D	$\frac{1.2}{K} \left(\frac{t_o}{\tau}\right)^{-1}$	$2.0 * t_o$	$0.5 * t_o$

FIGURA 4.4.1

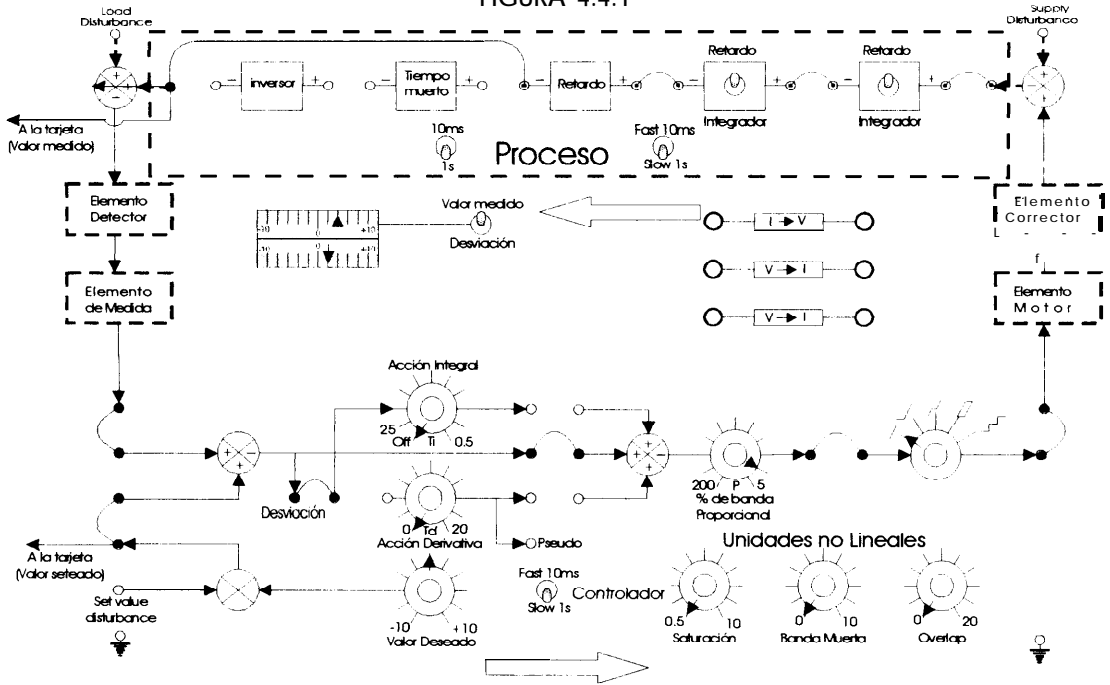
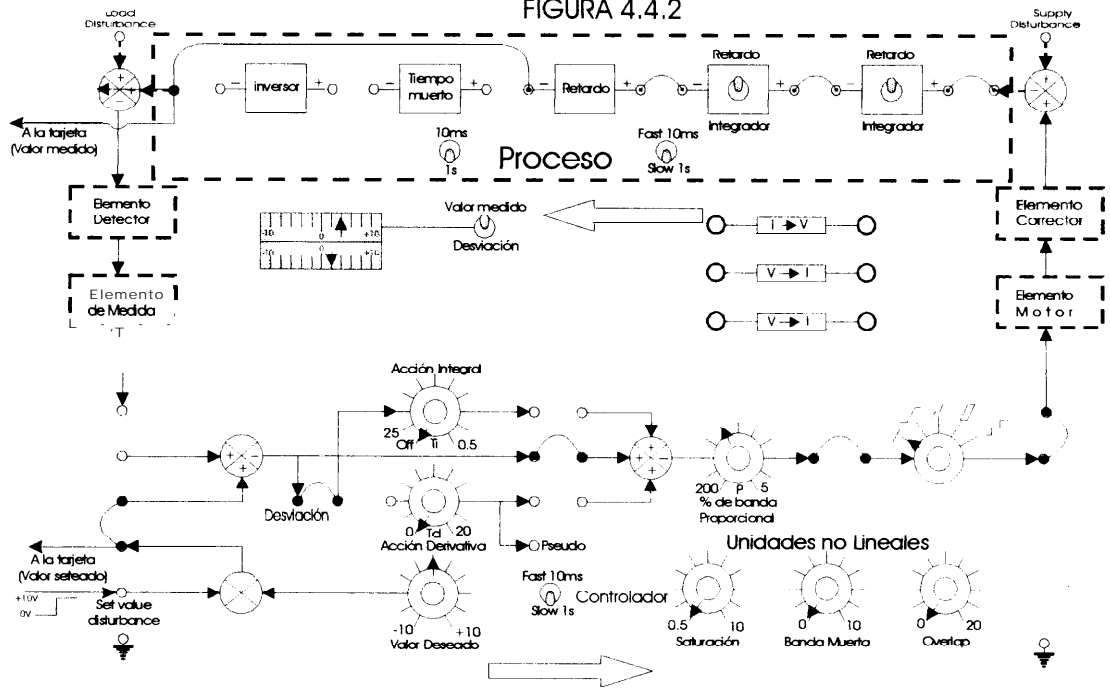


FIGURA 4.4.2



CONCLUSIONES

1. Realizando un análisis de resultados y experiencias del presente proyecto, podemos asegurar que se ha mejorado substancialmente el método de registro del estado del proceso (monitoreo en tiempo real) del simulador PCS327, y también se ha logrado la implementación en forma paralela de un sistema de adquisición de datos que nos permite grabar numéricamente la variación del estado del proceso tomando datos a intervalos de tiempo de varios milisegundos a minutos.
2. De igual forma es posible utilizar las interfases $I \rightarrow V$ del simulador PCS327 para monitorear y controlar procesos externos reales con la finalidad de aplicar los métodos vistos en las prácticas del laboratorio para encontrar su función de transferencia; relación que es muy útil para realizar simulaciones matemáticas y gráficas para determinar los parámetros óptimos de control, con la ventaja de que se puede aplicar a procesos rápidos con constantes de tiempo de varios milisegundos hasta procesos muy lentos con constantes de tiempo de varias horas, sin disminuir su grado de precisión.
3. Cabe destacar también que es de gran utilidad la implementación de sistemas similares a cualquier proceso industrial, ya que no sólo se puede utilizar para monitoreo y adquisición de datos del estado del proceso, si no también para realizar un control automático y preciso del mismo con la facilidad de una rápida modificación y los beneficios que presta un computador personal.

RECOMENDACIONES

En vista de que este proyecto es sólo un medio demostrativo para material didáctico, es preciso indicar que la interfase y el programa realizado en “Qbasic” para manejar la interfase, se la ha diseñado exclusivamente para el equipo simulador de procesos de control PCS327 en la que las señales de entrada están dentro de un rango de ± 10 voltios y 4 a 20 mA., no pudiéndose ingresar señales que estén fuera de esos rangos ya que se corre el peligro de quemar el equipo, o en su defecto utilizar el mismo programa para manejar interfases diferentes.

Si es necesario aplicar sistemas similares para la automatización de procesos mas complejos, es recomendable utilizar PLC`s como interfase entre el computador y el proceso, los cuales permiten hacer uso de software mucho más elaborados y amigables como por ejemplo “labview”, “concept”, “modsoft”, etc., que permiten la visualización de las variables del proceso y el proceso físico de una manera gráfica en tiempo real, aparte de los beneficios del control y adquisición de datos.

APÉNDICES

APÉNDICE A

```
REM***** PROGRAMA PCS327*****
CLS
LOCATE 8, 8
PRINT "Grafkar en pantalla AMPLITUD vs TIEMPO" 1"
LOCATE 10, 8
PRINT "Graficar en pantalla y GRABAR datos de AMPLITUD vs TIEMPO" 2"
LOCATE 12, 8
PRINT "Grafkar DIAGRAMA DE FASE Y RELACIÓN DE AMPLITUD" 3"
LOCATE 14, 8
PRINT "SALIR" 4"
LOCATE 16, 8
INPUT "INGRESE OPCIÓN (1 a 4): ", opc
SELECT CASE opc
CASE 1
GOSUB grafico
CASE 2
GOSUB Grabar
CASE 3
GOSUB fase
CASE 4
GOSUB sale
END SELECT
END
```

grafko:

```
10 REM declare function d AS INTEGER
20 ' $DYNAMIC
30 REDIM ts(640)
40 REDIM tm(640)
50 REDIM m(640) AS INTEGER
60 REDIM n(640) AS INTEGER
70 CLS
80 INPUT "Ingrese el factor de velocidad de barrido (1-10): "; vb
90 vb = CINT((-1.555 * vb) + 16.555)
100 CLS :
110 SCREEN 12
120 WINDOW (630, 460)-(1, 1)
130 VIEW (1, 1)-(630, 460), , 7
140 REM encerrar bus de datos y control de puertos
150 OUT &H378, 0
160 OUT &H37A, 235
170 salir = 0
180 ON KEY(4) GOSUB PH
190 KEY(4) ON
200 WHILE salir = 0
205 ts1# = TIMER
210 FOR o = 1 TO 30
220 CLS
230 f = 12
240 FOR i = 1 TO 11
250 LOCATE (2.5 * i), 2
260 PRINT (f - 2 * i)
270 NEXT i
280 FOR p = vb TO (640 * vb)
290 d = FIX(p \ vb)
300 REM habilitar el ADC0808 a output enable (pin 1 - enable)
310 OUT &H37A, 10
320 REM Seleccionar canal de entrada analógica IN0 (PIN 14 -ADDA)
330 OUT &H37A, 10
340 REM envio la señal de start al convertidor (pin 16 - star)
350 OUT &H37A, 230
360 FOR i = 1 TO 5: NEXT
370 OUT &H37A, 10
380 REM Verificar si ya terminó la conversión EOC al PIN 15
```

```

390 a = INP(&H379)
400 IF (a AND 8) = 8 THEN
410 ELSE GOTO 390
420 END IF
430 y1 = INP(&H378)
440 m(d) = CINT((520 / 256) * y1) - 38
442 ts(d) = TIMER - ts1#
444 LOCATE 27, 67
446 PRINT ts(d)
450 m(0) = m(1)
460 LINE ((d - 1), m(d - 1))-(d, m(d)), 1
470 REM conversión finalizada
480 REM Seleccionar canal de entrada analógica IN1 (PIN 14 -ADA)
490 OUT &H37A, 8
500 REM envío la señal de start al convertidor (pin 16 - star)
510 OUT &H37A, 236
520 FOR i = 1 TO 5: NEXT
530 OUT &H37A, 8
540 REM Verificar si ya terminó la conversión EOC al PIN 15
550 a = INP(&H379)
560 IF (a AND 8) = 8 THEN
570 y2 = INP(&H378)
580 ELSE GOTO 550
590 END IF
600 n(d) = CINT((520 / 256) * y2) - 38
610 n(0) = n(1)
620 LINE ((d - 1), n(d - 1))-(d, n(d)), 4
630 REM conversión finalizada
640 NEXT p
650 NEXT o
660 WEND
670 END
Grabar:
1010 REM declare function d AS INTEGER
1020 ' $DYNAMIC
1030 REDIM ts(640)
1040 REDIM tm(640)
1050 REDIM m(640) AS INTEGER
1060 REDIM n(640) AS INTEGER
1065 CLS
1070 INPUT "Escriba el nombre de la variable tiempo del valor seteado: "; TSS$
1075 INPUT "Escriba el nombre de la variable amplitud del valor seteado: "; ass$
1080 INPUT "Escriba el nombre de la variable amplitud del valor medido: "; amm$
1090 OPEN TSS$ FOR OUTPUT AS #1
1100 OPEN ass$ FOR OUTPUT AS #2
1105 OPEN amm$ FOR OUTPUT AS #3
11 10 INPUT "Ingrese el factor de velocidad de barrido (I-10): "; vb
11 15 vb = CINT((-1.555 * vb) + 16.555)
1120 CLS :
1130 SCREEN 12
1140 WINDOW (630, 460)-(1, 1)
1150 VIEW (1, 1)-(630, 460), , 7
1170 REM encerar bus de datos y control de puertos
11 80 OUT &H378, 0
1190 OUT &H37A, 235
1200 salir = 0
1210 ON KEY(4) GOSUB PH
1220 KEY(4) ON
1230 WHILE salir = 0
1240 ts1# = TIMER
1280 FOR o = 1 TO 30
1290 CLS
1292 f = 12
1294 FOR i = 1 TO 11
1296 LOCATE (2.5 * i), 2
1297 PRINT (f * 2 * i)
1298 NEXT i
1300 FOR p = vb TO (640 * vb)

```

```

1310 d = FIX(p / vb)
1340 REM habilitar el ADC0808 a output enable (pin 1 - enable)
1360 OUT &H37A, 10
1400 REM Seleccionar canal de entrada analógica IN0 (PIN 14 -ADDA)
1500 OUT &H37A, 10
1550 REM envio la señal de start al convertidor (pin 16 - star)
1560 OUT &H37A, 230
1580 FOR i = 1 TO 20: NEXT
1590 OUT &H37A, 10
1600 REM Verificar si ya terminó la conversión EOC al PIN 15
1700 a = INP(&H379)
1800 IF (a AND 8) = 8 THEN
1810 ELSE GOTO 1700
1820 END IF
1830 y1 = INP(&H378)
1840 m(d) = CINT((520 / 256) * y1) - 38
1842 ts(d) = TIMER - tsl#
1844 LOCATE 27, 67
1846 PRINT ts(d)
1850 m(0) = m(1)
1860 ts(d) = TIMER - tsl#
1870 IF ((p / vb) = d) AND (ts(d) <> ts(d - 1)) THEN
1890 PRINT #1, ts(d)
1892 r = (-25.061 + (.1020408 * m(d)))
1895 PRINT #2, r
1897 ELSE GOTO 1900
1898 END IF
1900 LINE ((d - 1), m(d - 1))-(d, m(d)), 1
1910 REM conversión finalizada
1920 REM Seleccionar canal de entrada analógica IN1 (PIN 14 -ADA)
1930 OUT &H37A, 8
1940 REM envio la señal de start al convertidor (pin 16 - star)
1950 OUT &H37A, 236
1960 FOR i = 1 TO 20: NEXT
1970 OUT &H37A, 8
1980 REM Verificar si ya terminó la conversión EOC al PIN 15
1990 a = INP(&H379)
2000 IF (a AND 8) = 8 THEN
2010 y2 = INP(&H378)
2020 ELSE GOTO 1990
2030 END IF
2040 n(d) = CINT((520 / 256) * y2) - 38
2050 n(0) = n(1)
2060 IF (p / vb) = d AND (ts(d) <> ts(d - 1)) THEN
2065 s = (-25.061 + (.1020408 * n(d)))
2070 PRINT #3, s
2075 ELSE GOTO 2080
2077 END IF
2080 LINE ((d - 1), n(d - 1))-(d, n(d)), 4
2090 REM conversión finalizada
2100 NEXT p
2110 NEXT o
2120 CLOSE #1
2130 CLOSE #2
2140 CLOSE #3
2150 WEND

```

```
OPEN TSS$ FOR INPUT AS #4
```

```
k=0
```

```
DO
```

```
k = k + 1
```

```
INPUT #4, t(k)
```

```
LOOP UNTIL (EOF(l))
```

```
nnn = k
```

```
CLOSE #4
```

```
OPEN TSS$ FOR OUTPUT AS #5
```

```
FOR i=0 TO nnn-1
```

```
frac = (t(nnn) - t(1)) / nnn
```

fase:

3010 REM declare function d AS INTEGER

3020 ' \$DYNAMIC

3030 REDIM ts(640)

3040 REDIM tm(640)

3050 REDIM m(640) AS INTEGER

3060 REDIM n(640) AS INTEGER

3062 REDIM mm(640) AS INTEGER

3065 REDIM nn(640) AS INTEGER

3070 CLS

3080 INPUT "Ingrese el factor de velocidad de barrido (l-10): "; vb

3084 escala = 1

3086 PRINT "Ingrese un factor de escala para mejorar la visualización"

3088 INPUT "del ángulo de desfase a altas frecuencias (default=l): "; escala

3090 vb = CINT((-1.555 * vb) + 16.555)

3100 CLS :

3110 SCREEN 12

3120 WINDOW (630, 460)-(1, 1)

3130 VIEW (1, l)-(630, 460), , 7

3140 REM encerrar bus de datos y control de puertos

3150 OUT &H378,0

3160 OUT &H37A, 235

3170 salir = 0

3180 ON KEY(4) GOSUB PH

3190 KEY(4) ON

3200 WHILE salir = 0

3205 ts[#] = TIMER

3210 FOR o = 1 TO 30

3220 CLS

3230 FOR i = 1 TO 4

3240 LINE (1, 57.5 * i)-(640, 57.5 * i)

3250 NEXT i

FOR i = 4 TO 7

LINE (160, 57.5 * i)-(480, 57.5 * i)

NEXT i

3260 FOR i = 1 TO 5

3270 LINE ((80 + 80 * i), 230)-((80 + 80 * i), 480)

3280 NEXT i

3290 f = 15: b = 0

3300 FOR i = 1 TO 5

3310 LOCATE (b + 14 + i), 2

3320 PRINT (f - 5 * i)

3330 b = b + 2.5

3340 NEXT i

3350 FOR p = vb TO (640 * vb)

3360 d = FIX(p / vb)

3370 REM habilitar el ADC0808 a output enable (pin 1 - enable)

3380 OUT &H37A, 10

3390 REM Seleccionar canal de entrada analógica IN0 (PIN 14 -ADDA)

3400 OUT &H37A, 10

3410 REM envío la señal de start al convertidor (pin 16 - star)

3420 OUT &H37A, 230

3430 FOR i = 1 TO 20: NEXT

3440 OUT &H37A, 10

3450 REM Verificar si ya terminó la conversión EOC al PIN 15

3460 a = INP(&H379)

3470 IF (a AND 8) = 8 THEN

3480 ELSE GOTO 3460

3490 END IF

3500 yl = INP(&H378)

```

3510 m(d) = yl: mm(d) = escala * (1.8 * yl + 80)
3515 m(d) = CINT((320 / 256) * yl) - 30
3517 ts(d) = TIMER - tsI#
3518 LOCATE 27, 70
3519 PRINT ts(d)
3520 m(0) = m(1): mm(0) = mm(1)
3525 LINE ((d - 1), m(d - 1))-(d, m(d)), 1
3530 REM conversión finalizada
3540 REM Seleccionar canal de entrada analógica IN1 (PIN 14 -ADA)
3560 OUT &H37A, 8
3570 REM envío la señal de start al convertidor (pin 16 - star)
3580 OUT &H37A, 236
3590 FOR i = 1 TO 20: NEXT
3600 OUT &H37A, 8
3610 REM Verificar si ya terminó la conversión EOC al PIN 15
3620 a = INP(&H379)
3630 IF (a AND 8) = 8 THEN
3640 y2 = INP(&H378)
3650 ELSE GOTO 3620
3660 END IF
3670 n(d) = y2: nn(d) = 1.8 * y2 + 120
3675 n(d) = CINT((320 / 256) * y2) - 30
3680 n(0) = n(1): nn(0) = nn(1)
3685 LINE ((d - 1), n(d - 1))-(d, n(d)), 4
3690 LINE (mm(d - 1), nn(d - 1))-(mm(d), nn(d)), 4
3700 REM conversión finalizada
3710 NEXT p
3720 NEXT o
3730 WEND
3740 END

```

sale:

```

5000 PH:
5002 OUT &H37A, 232
5004 OUT &H379, 135
5006 OUT &H378, 0
5010 salir = 1

```


APÉNDICE B

```
10 CLS : PRINT
20 PRINT "***** Programa FOURIER *****": PRINT
30 PRINT "Este programa convierte una respuesta escalon a una respuesta a la"
40 PRINT "frecuencia equivalente."
50 PRINT "Para escala logaritmica debe cambiar las lineas 800 y 870."
60 PRINT : PRINT
90 DIM T(700), AM(700), DA(700)
100 PRINT "Los datos de la respuesta escalon deben ser introducidos como"
110 PRINT "coordenadas (tiempo,amplitud)."grafico visualmente identico a la respuesta original."
160 PRINT : PRINT
190 REM
200 REM INTRODUCIR Y CHEQUEAR DATOS DE RESPUESTA ESCALON
210 REM
220 INPUT "INGRESE EL NOMBRE DEL ARCHIVO DEL TIEMPO: "; T$
225 INPUT "INGRESE EL NOMBRE DEL ARCHIVO AMPLITUD DEL VALOR MEDIDO: "; AM$
230 PRINT
232 OPEN T$ FOR INPUT AS #1
234 OPEN AM$ FOR INPUT AS #2
236 k = 0
238 DO
240 k = k + 1
242 INPUT #1, T(k)
255 LOOP UNTIL (EOF(1))
260 CLOSE #1
265 k = 0
270 DO
275 k = k + 1
280 INPUT #2, AM(k)
285 NN = k
290 LOOP UNTIL (EOF(2))
295 CLOSE #2
300 REM CLS : PRINT : PRINT
320 REM
330 PRINT "Chequear los datos de entrada": PRINT
335 REM FOR i = 1 TO 5000: NEXT i
340 PRINT "PUNTO TIEMPO AMPLITUD"
350 FOR i = 1 TO NN
360 PRINT i, T(i), AM(i)
370 NEXT i
380 PRINT : INPUT "Desea cambiar algun valor (S/N)"; Q$
390 IF Q$ = "N" OR Q$ = "n" THEN 510
400 IF Q$ <> "S" AND Q$ <> "s" THEN 380
410 PRINT
420 INPUT "NUMERO DEL PUNTO DE DATOS A CAMBIAR"; i
430 PRINT "INTRODUCIR NUEVO TIEMPO Y AMPLITUD"
440 INPUT T(i), AM(i)
450 PRINT : INPUT "Desea cambiar mas valores (S/N)"; Q$
460 IF Q$ = "S" OR Q$ = "s" THEN 420
470 GOTO 320
480 REM
490 REM NORMALIZAR DATOS
500 REM
510 N1 = NN - 1: N2 = NN - 2
520 T1 = T(1): BB = AM(1): AA = AM(NN) - BB
530 FOR i = 1 TO NN
540 T(i) = T(i) - T1: AM(i) = (AM(i) - BB) / AA
550 NEXT i
560 PRINT : PRINT
570 PRINT "EL ESCALON EMPIEZA EN EL TIEMPO"; T1
580 PRINT "Y TIENE UNA AMPLITUD DE"; AA
590 PRINT : PRINT "LOS DATOS HAN SIDO NORMALIZADOS PARA QUE LA GANANCIA SEA 1"
```

APÉNDICE B

```
10 CLS : PRINT
20 PRINT "***** Programa FOURIER *****": PRINT
30 PRINT "Este programa convierte una respuesta escalon a una respuesta a la"
40 PRINT "frecuencia equivalente."
50 PRINT "Para escala logaritmica debe cambiar las lineas 800 y 870."
60 PRINT : PRINT
90 DIM T(700), AM(700), DA(700)
100 PRINT "Los datos de la respuesta escalon deben ser introducidos como"
110 PRINT "coordenadas (tiempo,amplitud)."grafico visualmente identico a la respuesta original."
160 PRINT : PRINT
190 REM
200 REM INTRODUCIR Y CHEQUEAR DATOS DE RESPUESTA ESCALON
210 REM
220 INPUT "INGRESE EL NOMBRE DEL ARCHIVO DEL TIEMPO: "; T$
225 INPUT "INGRESE EL NOMBRE DEL ARCHIVO AMPLITUD DEL VALOR MEDIDO: "; AM$
230 PRINT
232 OPEN T$ FOR INPUT AS #1
234 OPEN AM$ FOR INPUT AS #2
236 k = 0
238 DO
240 k = k + 1
242 INPUT #1, T(k)
255 LOOP UNTIL (EOF(1))
260 CLOSE #1
265 k = 0
270 DO
275 k = k + 1
280 INPUT #2, AM(k)
285 NN = k
290 LOOP UNTIL (EOF(2))
295 CLOSE #2
300 REM CLS : PRINT : PRINT
320 REM
330 PRINT "Chequear los datos de entrada": PRINT
335 REM FOR i = 1 TO 5000: NEXT i
340 PRINT "PUNTO TIEMPO AMPLITUD"
350 FOR i = 1 TO NN
360 PRINT i, T(i), AM(i)
370 NEXT i
380 PRINT : INPUT "Desea cambiar algun valor (S/N)"; Q$
390 IF Q$ = "N" OR Q$ = "n" THEN 510
400 IF Q$ <> "S" AND Q$ <> "s" THEN 380
410 PRINT
420 INPUT "NUMERO DEL PUNTO DE DATOS A CAMBIAR"; i
430 PRINT "INTRODUCIR NUEVO TIEMPO Y AMPLITUD"
440 INPUT T(i), AM(i)
450 PRINT : INPUT "Desea cambiar mas valores (S/N)"; Q$
460 IF Q$ = "S" OR Q$ = "s" THEN 420
470 GOTO 320
480 REM
490 REM NORMALIZAR DATOS
500 REM
510 N1 = NN - 1: N2 = NN - 2
520 TI = T(1): BB = AM(1): AA = AM(NN) - BB
530 FOR i = 1 TO NN
540 T(i) = T(i) - TI: AM(i) = (AM(i) - BB) / AA
550 NEXT i
560 PRINT : PRINT
570 PRINT "EL ESCALON EMPIEZA EN EL TIEMPO"; TI
580 PRINT "Y TIENE UNA AMPLITUD DE"; AA
590 PRINT : PRINT "LOS DATOS HAN SIDO NORMALIZADOS PARA QUE LA GANANCIA SEA 1"
```

```

600 PRINT "A CERO FRECUENCIA."
610 FOR i = 1 TO NI
620 DA(i) = (AM(i + 1) - AM(i)) / (T(i + 1) - T(i))
630 NEXT i
640 PRINT : PRINT
650 REM
660 REM ESTABLECER RANGO DE FRECUENCIAS Y ESCALA LINEAL
670 REM
680 INPUT "NUMERO DE PUNTOS DE FRECUENCIA (<41)"; NW
690 IF NW > 1 THEN 720
700 PRINT "El numero de puntos de frecuencia debe ser mayor a 1."
710 GOTO 680
720 PRINT : PRINT
730 PRINT "INTRODUCIR LOS VALORES MAXIMO Y MINIMO DE FRECUENCIA"
740 INPUT "FRECUENCIA MINIMA ="; WM
750 INPUT "FRECUENCIA MAXIMA ="; WX
760 IF WM > 0 AND WX > 0 THEN 790
770 PRINT "Todas las frecuencias deben ser positivas"
780 GOTO 720
790 IF WM = WX THEN WX = 10 * WM
800 PRINT : PRINT
805 REM 800 PRINT:PRINT:DW=(WX/WM)^(1/(NW-1)): Escala Logaritmica
810 PRINT "INGRESE EL NOMBRE DEL ARCHIVO DE SALIDA CON DATOS DE"
815 INPUT "RESPUESTA CON FRECUENCIA DOMINANTE: "; FD$
820 REM RUTINA PARA CALCULAR LA RESPUESTA A LA FRECUENCIA
830 OPEN FD$ FOR OUTPUT AS #3
840 PRINT "LA RESPUESTA A LA FRECUENCIA ES:": PRINT
850 PRINT "FRECUENCIA REAL IMAGINARIA"
860 FOR i = 0 TO NW - 1
870 W = WM + i * (WX - WM) / (NW - 1)
875 REM 870 W=WM*DW^i: Escala logaritmica
880 RE = (DA(N1) * SIN(W * T(NN)) - DA(1) * SIN(W * T(1))) / W
890 IM = (DA(N1) * COS(W * T(NN)) - DA(1) * COS(W * T(1))) / W
900 FOR J = 1 TO N2
910 J1 = J + 1
920 WH = W * T(J1)
930 DF = (DA(J) - DA(J1)) / W
940 DR = DF * SIN(WH); DI = DF * COS(WH)
950 RE = RE + DR; IM = IM + DI
960 NEXT J
980 PRINT W, RE, IM
985 PRINT #3, W, RE, IM
990 NEXT i
1000 REM
1010 REM CAMBIAR EL RANGO DE FRECUENCIAS, SI SE DESEA
1020 REM
1030 PRINT : PRINT "Desea cambiar el rango de frecuencias (S/N)"; : INPUT Q$
1040 IF Q$ = "S" OR Q$ = "s" THEN 640
1060 IF Q$ <> "N" AND Q$ <> "n" THEN 1030
1062 CLOSE #1
1064 CLOSE #2
1066 CLOSE #3
1070 END

```

APÉNDICE C

```
10 CLS : PRINT
20 REM PROGRAMA 'LEVY'.
30 PRINT "ESTE PROGRAMA ENCUENTRA UNA FUNCION DE TRANSFERENCIA A PARTIR DE"
40 PRINT "LOS DATOS DE RESPUESTA A LA FRECUENCIA, USANDO EL METODO DE AJUSTE"
50 PRINT "COMPLEJO DE CURVAS DE LEVY.; "; ""
60 PRINT "EL METODO NO INCLUYE POLOS EN EL ORIGEN."
70 REM
110 DIM P(15, 15), AB(15), D2(40)
120 DIM FW(40), FR(40), FI(40), CP(15)
130 DIM IP(15), IQ(15), B(15), C(15)
140 PRINT : PRINT
150 REM
160 PRINT "ESPECIFICAR ORDEN DEL NUMERADOR Y DENOMINADOR."
170 REM
180 PRINT "ORDEN DEL DENOMINADOR (< 6)";
190 INPUT M: IF M > 5 THEN 180
200 PRINT "ORDEN DEL NUMERADOR (< 6)";
210 INPUT NN: N = NN + 1: IF M >= NN THEN 240
220 PRINT "EL ORDEN DEL NUMERADOR NO DEBE SER MAYOR QUE "
230 PRINT "EL ORDEN DEL DENOMINADOR": PRINT : GOTO 180
240 MN = M + N: PRINT : PRINT
250 FOR I = 1 TO MN
260 FOR J = 1 TO MN
270 P(I, J) = 0
280 NEXT J
290 CP(I) = 0
300 NEXT I
310 REM
320 REM INTRODUCIR DATOS DE FRECUENCIA.
330 REM
340 PRINT "LOS DATOS DE RESPUESTA A LA FRECUENCIA DEBEN SER DADOS EN TERMINO?"
350 PRINT "DE FRECUENCIA, PARTE REAL E IMAGINARIA ."
360 PRINT : PRINT
390 PRINT "NUMERO DE PUNTOS DE FRECUENCIA (< 41)"
395 PRINT : PRINT
400 PRINT "INGRESE EL NOMBRE DEL ARCHIVO CON DATOS"
410 INPUT "DE FRECUENCIA DOMINANTE: "; FD$
420 OPEN FD$ FOR INPUT AS #1
430 K = 0
440 WHILE NOT EOF(I)
450 K = K + 1
460 INPUT #1, FW(K), FR(K), FI(K)
470 NW = K
480 WEND
490 REM
500 PRINT : PRINT
510 PRINT "Chequear los datos de entrada"
515 FOR I = 1 TO 5000: NEXT I
520 PRINT "PUNTO FRECUENCIA REAL IMAGINARIA"
530 FOR I = 1 TO NW
540 PRINT I, FW(I), FR(I), FI(I)
550 NEXT I
560 REM
570 REM CHEQUEAR LOS DATOS DE FRECUENCIA
580 REM
590 PRINT : INPUT "Desea cambiar algun valor (S/N)"; Q$
600 IF Q$ = "N" OR Q$ = "n" THEN 710
610 IF Q$ <> "S" AND Q$ <> "s" THEN 590
620 INPUT "NUMERO DEL PUNTO DE DATOS A SER CAMBIADO"; I
630 PRINT "INTRODUCIR NUEVA FRECUENCIA,REAL,E IMAGINARIA"
640 INPUT FW(I), FR(I), FI(I)
650 PRINT : INPUT "Desea cambiar mas valores (S/N)"; Q$
660 IF Q$ = "S" OR Q$ = "s" THEN 620
670 GOTO 500
680 REM
```

```

690 REM DETERMINAR TAMAÑOS DE LOS ARREGLOS Y CONSTANTES
700 REM 710 NP=2*NN+1:N1=N+1:N2=N+2:N3=N+3
710 NP = 2 * NN + 1: N1 = N + 1: N2 = N + 2: N3 = N + 3
720 MP = MN + NN: MQ = MN + M - 1: ZM = MN - 1
730 REM
740 REM RUTINA PRINCIPAL PARA ESTABLECER LA MATRIZ ESQUELETO P,
750 REM CP ES EL PRODUCTO VECTORIAL y ASOCIADO CON Px, Y AB ES
760 REM EL VECTOR x DE COEFICIENTES DE LA FUNCION DE TRANSFERENCIA"
770 REM
780 FOR KI = 1 TO NW
790 FH = FW(KI) * FW(KI)
800 P(1, 1) = P(1, 1) + D2(KI)
810 CP(I) = CP(1) + FR(KI) * D2(KI)
820 IF N = 1 THEN 920
830 SH = D2(KI)
840 FOR I = 3 TO NP STEP 2
850 IF I > N THEN 890
860 SH = -SH * FH
870 P(1, I) = P(1, I) + SH
880 GOTO 910
890 II = I - N + 1: SH = SH * FH
900 P(II, N) = P(II, N) + SH
910 NEXT I
920 TI = D2(KI) * FW(KI) * FI(KI)
930 P(I, NI) = P(I, NI) + TI
940 IF M = 1 THEN 1130
950 S2 = -D2(KI) * FR(KI)
960 FOR J = N2 TO MP STEP 2
970 IF J > MN THEN 1050
980 S2 = -S2 * FH
990 P(1, J) = P(1, J) + S2
1000 JJ = J + 1
1010 IF JJ > MN THEN 1080
1020 TI = -TI * FH
1030 P(I, JJ) = P(1, JJ) + TI
1040 GOTO 1120
1050 I = J - MN + 1: S2 = S2 * FH: JJ = J + 1
1060 IF I > N THEN 1120
1070 P(I, MN) = P(I, MN) + S2
1080 I = JJ + 1 - MN
1090 IF I > N THEN 1120
1100 T1 = T1 * FH
1110 P(I, MN) = P(I, MN) + T1
1120 NEXT J
1130 FX = (FI(KI) * FI(KI) + FR(KI) * FR(KI)) * D2(KI) * FH
1140 P(N1, NI) = P(N1, NI) + FX
1150 IF M = 1 THEN 1240
1160 FOR I = N3 TO MQ STEP 2
1170 IF I > MN THEN 1210
1180 FX = -FX * FH
1190 P(N1, I) = P(N1, I) + FX
1200 GOTO 1230
1210 FX = FX * FH: J = I - MN + 1 + N
1220 P(J, MN) = P(J, MN) + FX
1230 NEXT I
1240 NEXT KI
1250 REM
1260 REM EN ESTA SECCION SE LLENA LA MATRIZ P
1270 REM
1280 IF N < 3 THEN 1370
1290 MH = -1: H = 1
1300 FOR J = 2 TO N
1310 H = H * MH: JI = J - 1
1320 FOR L = 1 TO NN
1330 K = L + 1
1340 P(J, L) = P(JI, K) * H
1350 NEXT L
1360 NEXT J

```

```

1370 IF N = 1 THEN 1460
1380 MH = -1: H = 1
1390 FOR J = 2 TO N
1400 H = H * MH: J1 = J - 1
1410 FOR L = N1 TO ZM
1420 K = L + 1
1430 P(J, L) = P(J1, K) * H
1440 NEXT L
1450 NEXT J
1460 IF M < 3 THEN 1550
1470 MH = -1: H = 1
1480 FOR J = N2 TO MN
1490 H = H * MH: J1 = J - 1
1500 FOR L = N1 TO ZM
1510 K = L + 1
1520 P(J, L) = P(J1, K) * H
1530 NEXT L
1540 NEXT J
1550 X1 = 1: X2 = -1: I1 = 0
1560 FOR I = 1 TO N
1570 I1 = I1 + 1
1580 IF I1 >= 3 THEN I1 = 1: X1 = X1 * X2
1590 JJ = 0: Y1 = 1
1600 FOR J = NI TO MN
1610 JJ = JJ + 1
1620 IF JJ >= 3 THEN JJ = 1: Y1 = Y1 * X2
1630 P(J, I) = P(I, J) * X1 * Y1
1640 NEXT J
1650 NEXT I
1660 REM
1670 REM EL VECTOR CP SE DETERMINA A PARTIR DE ELEMENTOS APROPIADOS DENTRO
DE
1680 REM LA MATRIZ P.
1690 REM
1700 IF N = 1 THEN 1770
1710 X1 = 1
1720 FOR I = 1 TO NN
1730 I1 = I + 1
1740 CP(I1) = P(I, N1) * X1
1750 X1 = -X1
1760 NEXT I
1770 IF M = 1 THEN 1830
1780 FOR I = N2 TO MN STEP 2
1790 J = I - 1
1800 CP(I) = P(J, NI)
1810 NEXT I
1820 REM
1830 REM INVERTIR LA MATRIZ P
1840 REM
1850 GOSUB 2410
1860 REM
1870 REM HALLAR AB=INVERSA(P)*CP
1880 REM
1890 FOR I = 1 TO MN
1900 AB(I) = 0
1910 FOR J = 1 TO MN
1920 AB(I) = AB(I) + P(I, J) * CP(J)
1930 NEXT J
1940 NEXT I
1950 REM
1960 REM IMPRIMIR LOS COEFICIENTES DE LA FUNCION DE TRANSFERENCIA
1970 REM
1980 PRINT : PRINT
1990 PRINT "LA FUNCION DE TRANSFERENCIA ES DE LA FORMA:"
2000 PRINT : PRINT "A(0)+A(1)*S+A(2)*S^2+..."
2010 PRINT " "
2020 PRINT "B(0)+B(1)*S+B(2)*S^2+..."
2030 PRINT : PRINT "DONDE:"

```

```

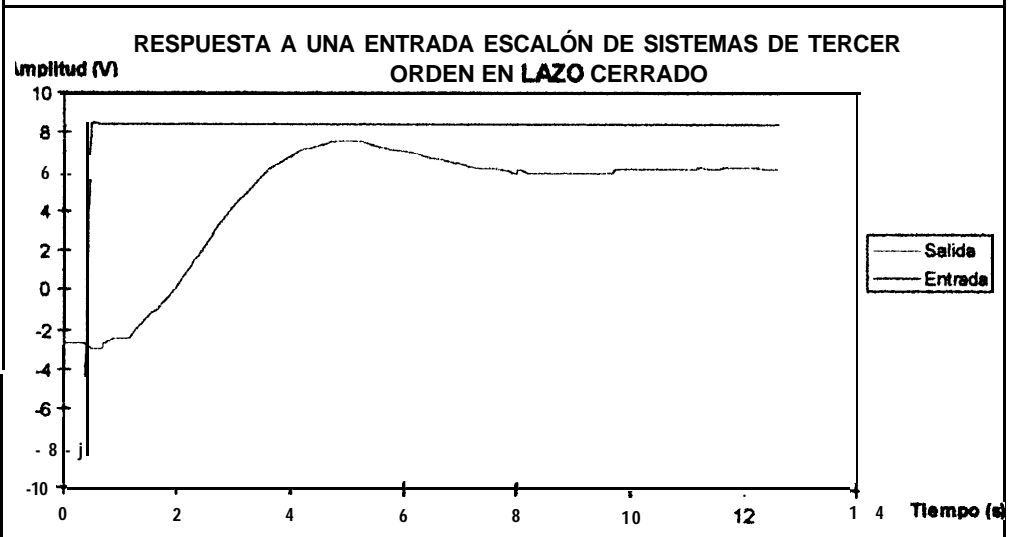
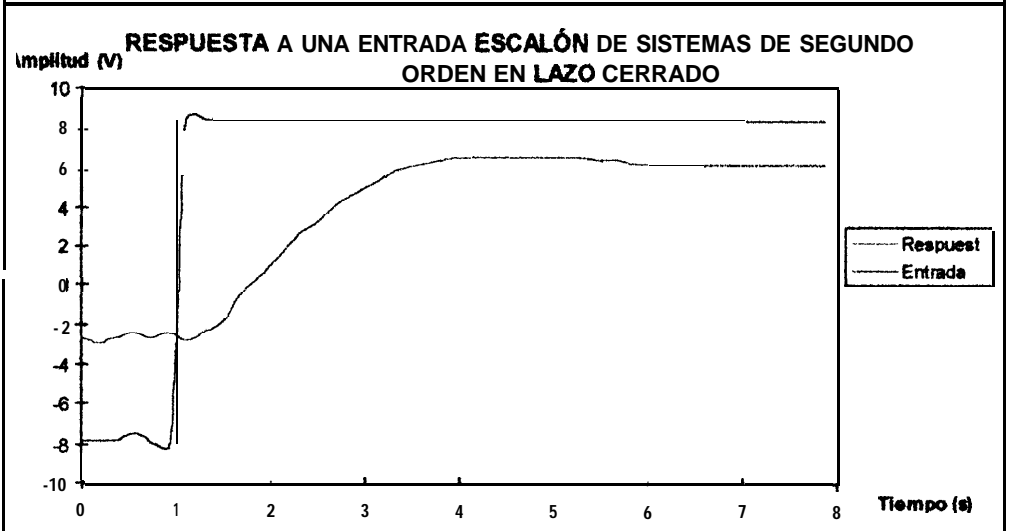
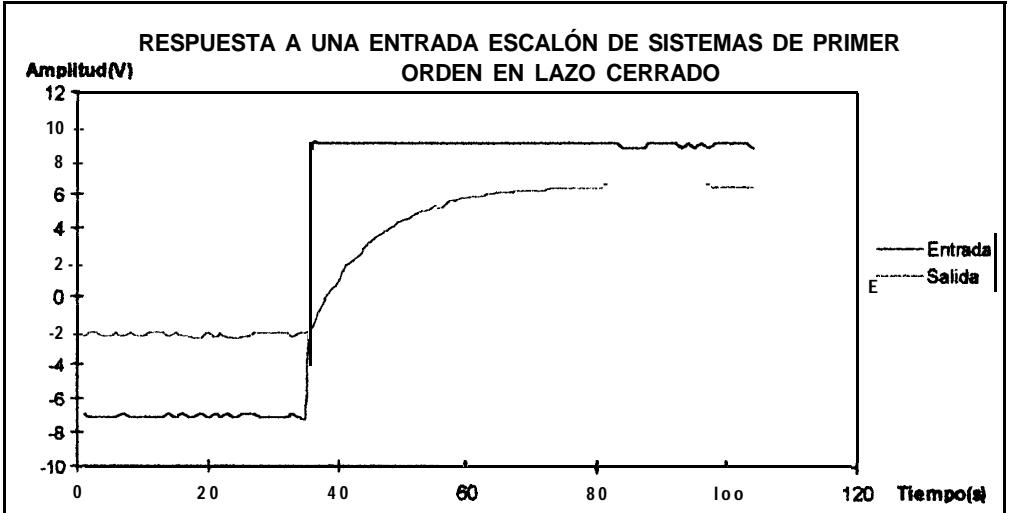
2040 PRINT
2050 FOR I = 1 TO N
2060 PRINT "A("; I - 1; ")= "; AB(I)
2070 NEXT I
2080 PRINT : PRINT "B( 0 )= 1"
2090 FOR I = N + 1 TO MN
2100 PRINT "B("; I - N; ")= "; AB(I)
2110 NEXT I
2120 REM
2130 REM REMOVER EL SESGO, SI SE REQUIERE
2140 REM
2150 PRINT : PRINT
2160 INPUT "Refinar la estimacion (S/N)"; Q$
2170 IF Q$ = "N" OR Q$ = "n" THEN 2370
2180 IF Q$ <> "S" AND Q$ <> "s" THEN 2160
2190 REM
2200 REM FIJAR D2 Y REINTRODUCIR DATOS
2210 REM
2220 FOR J = 1 TO NW
2230 W = FW(J); RE = FR(J); IM = FI(J)
2240 SR = 0; SI = W; DR = 1; DI = 0
2250 FOR I = N + 1 TO MN
2260 DR = DR + AB(I) * SR
2270 DI = DI + AB(I) * SI
2280 HS = SR; SR = -SI * W; SI = HS * W
2290 NEXT I
2300 D2(J) = 1 / (DR * DR + DI * DI)
2310 NEXT J
2320 FOR I = 1 TO MN
2330 FOR J = 1 TO MN
2340 P(I, J) = 0: NEXT J
2350 CP(I) = 0: NEXT I
2360 GOTO 780
2370 END
2380 REM
2390 REM SUBROUTINA PARA HALLAR LA INVERSA DE LA MATRIZ P
2400 REM
2410 FOR K = 1 TO MN
2420 PVOT = 0
2430 FOR I = K TO MN
2440 FOR J = K TO MN
2450 IF (ABS(P(I, J) - ABS(PVOT) <= 0)) THEN 2470
2460 PVOT = P(I, J): IP(K) = I: IQ(K) = J
2470 NEXT J
2480 NEXT I
2490 IF (ABS(PVOT) = 0) THEN 2870
2500 II = IP(K): PP = IQ(K)
2510 IF (II - K = 0) THEN 2550
2520 FOR J = 1 TO MN
2530 Z = P(II, J): P(II, J) = P(K, J): P(K, J) = Z
2540 NEXT J
2550 IF (PP - K = 0) THEN 2590
2560 FOR I = 1 TO MN
2570 Z = P(I, PP): P(I, PP) = P(I, K): P(I, K) = Z
2580 NEXT I
2590 FOR J = 1 TO MN
2600 IF (J - K = 0) THEN 2620
2610 GOTO 2640
2620 B(J) = 1 / PVOT: C(J) = 1
2630 GOTO 2660
2640 B(J) = -1 * P(K, J) / PVOT
2650 C(J) = P(J, K)
2660 P(K, J) = 0: P(J, K) = 0
2670 NEXT J
2680 FOR I = 1 TO MN
2690 FOR J = 1 TO MN
2700 P(I, J) = P(I, J) + C(I) * B(J)
2710 NEXT J

```

```
2720 NEXT I
2730 NEXT K
2740 MX = MN + 1
2750 FOR L = 1 TO MN
2760 K = MX - L: II = IP(K): PP = IQ(K)
2770 IF (II - K = 0) THEN 2810
2780 FOR I = 1 TO MN
2790 Z = P(I, II): P(I, II) = P(I, K): P(I, K) = Z
2800 NEXT I
2810 IF (PP - K = 0) THEN 2850
2820 FOR J = 1 TO MN
2830 Z = P(PP, J): P(PP, J) = P(K, J): P(K, J) = Z
2840 NEXT J
2850 NEXT L
2860 RETURN
2870 PRINT "ERROR EN LA INVERSION DE LA MATRIZ P"
2880 RETURN
```


APÉNDICE D

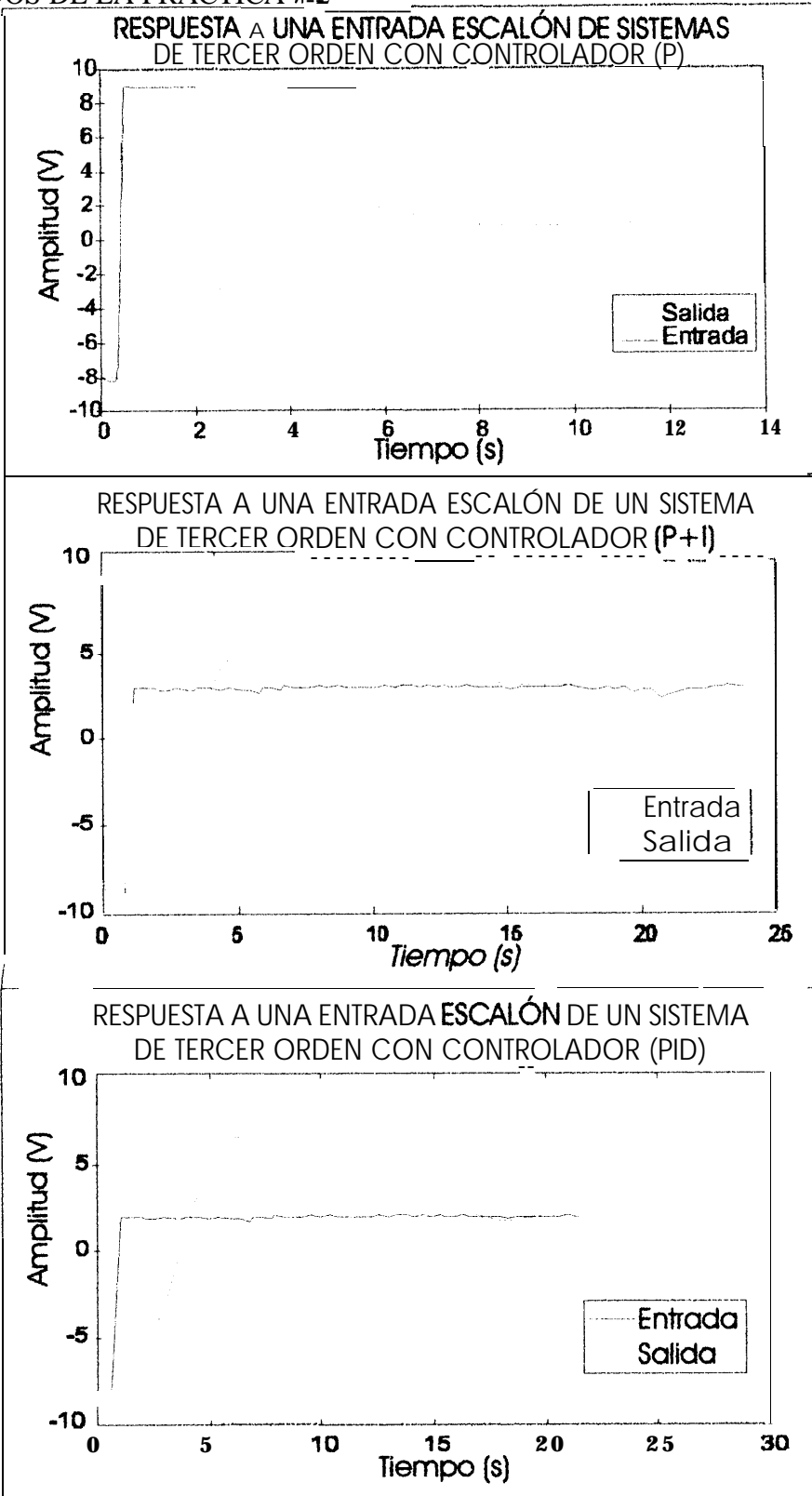
RESULTADOS DE LA PRÁCTICA # 1



Nótese en estas respuestas a una entrada escalón el error de estado estable que presentan los controladores proporcionales.

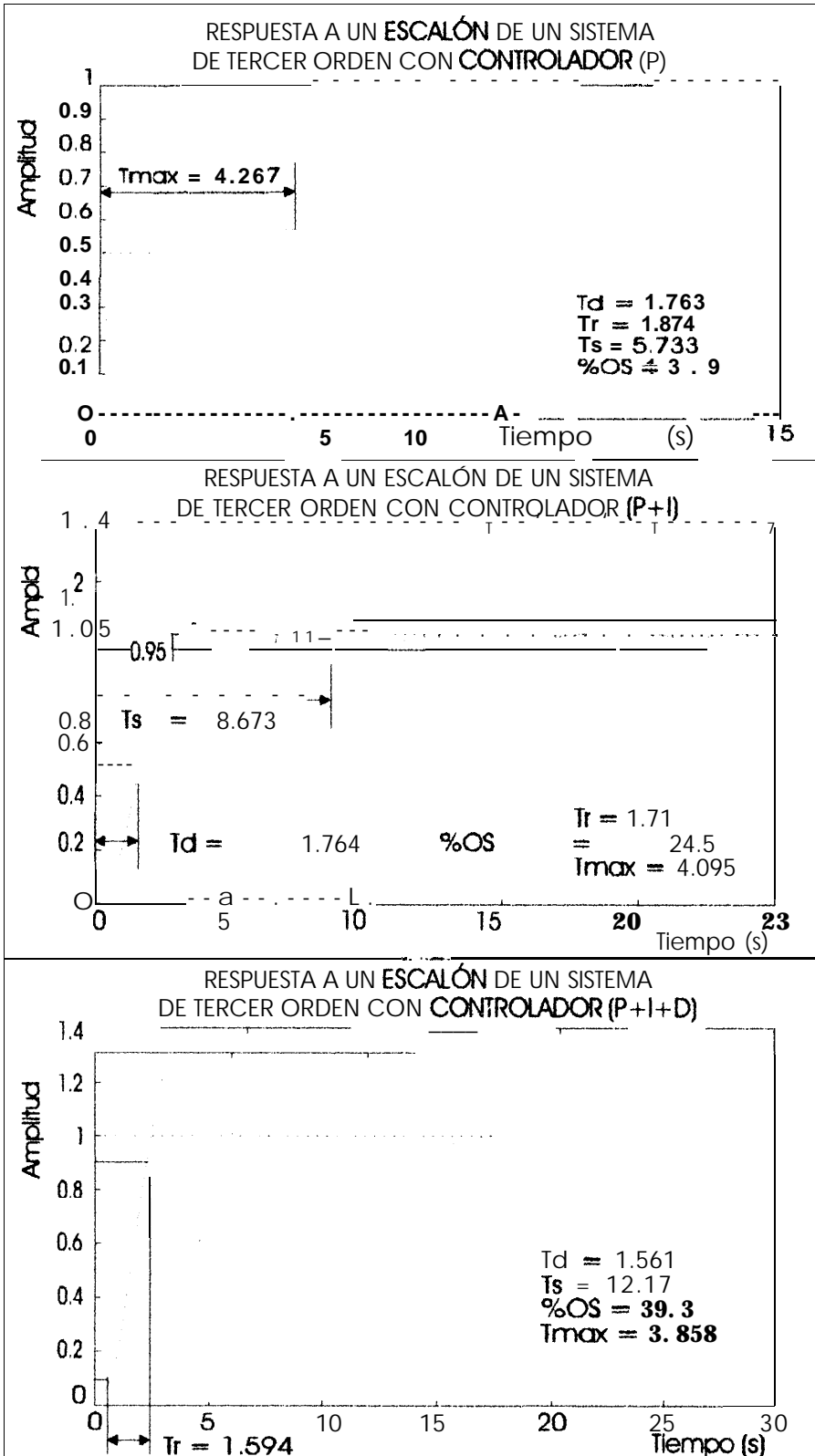
APÉNDICE E

RESULTADOS DE LA PRÁCTICA #2



En estos gráficos tomados de datos reales pueden diferenciarse las respuestas de los diferentes tipos de controladores, tanto el error de estado estable como el tiempo de estabilización.

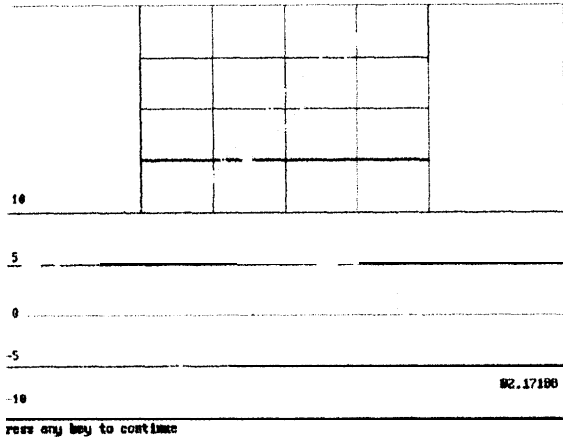
APÉNDICE F



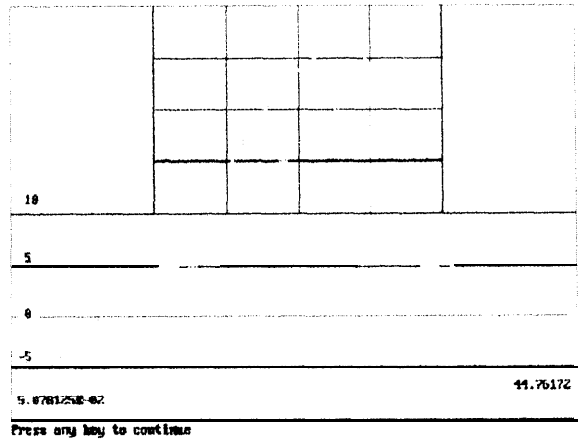
Gráficos comparativos con los del APÉNDICE E basados en las funciones de transferencia teóricas descritas por las ecuaciones 4.2.2 - 4 y 6 del capítulo 4, analizadas y graficadas por el programa MATLAB.

APÉNDICE G

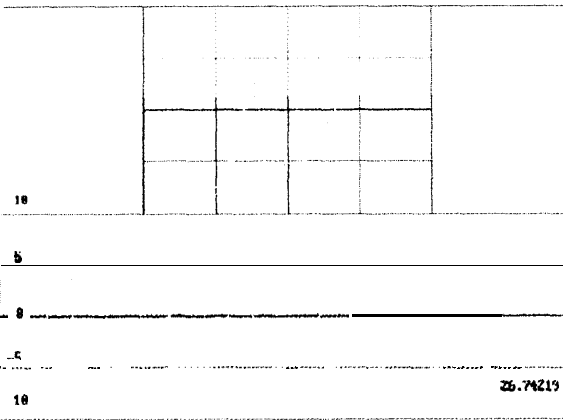
1



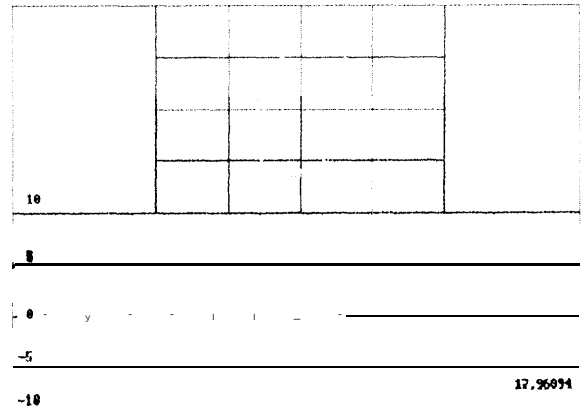
2



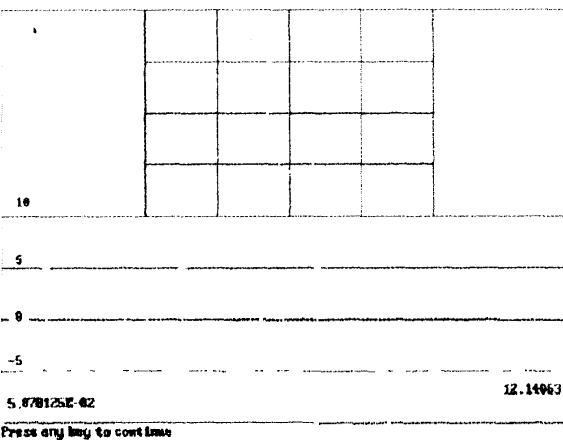
3



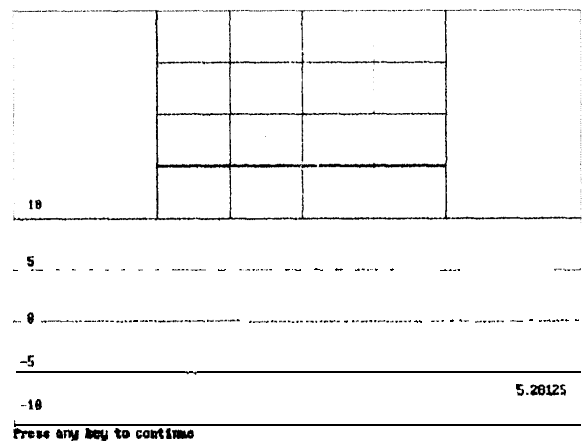
4



5



6



APÉNDICE H

DIAGRAMA DE BODE (M vs w) EN BASE A LOS DATOS TOMADOS A PARTIR DEL APÉNDICE (G)

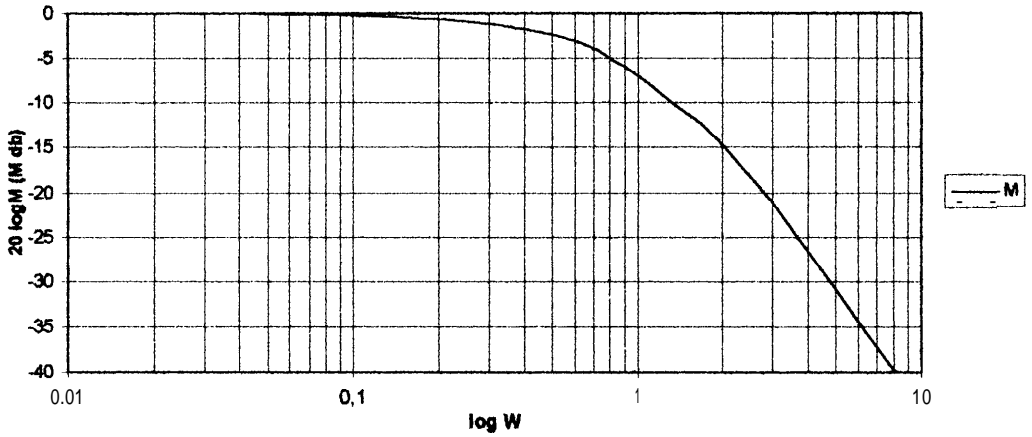


DIAGRAMA DE BODE (Teta vs w) BASADO EL LOS DATOS OBTENIDOS A PARTIR DEL APÉNDICE (G)

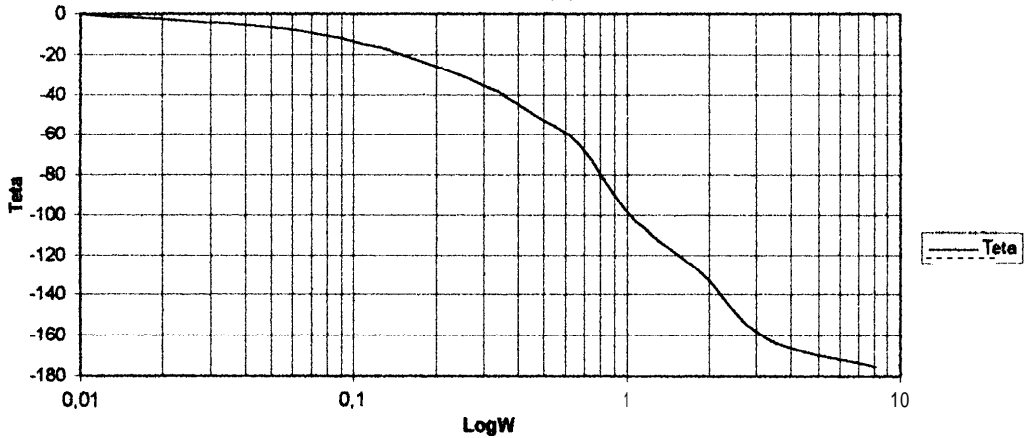
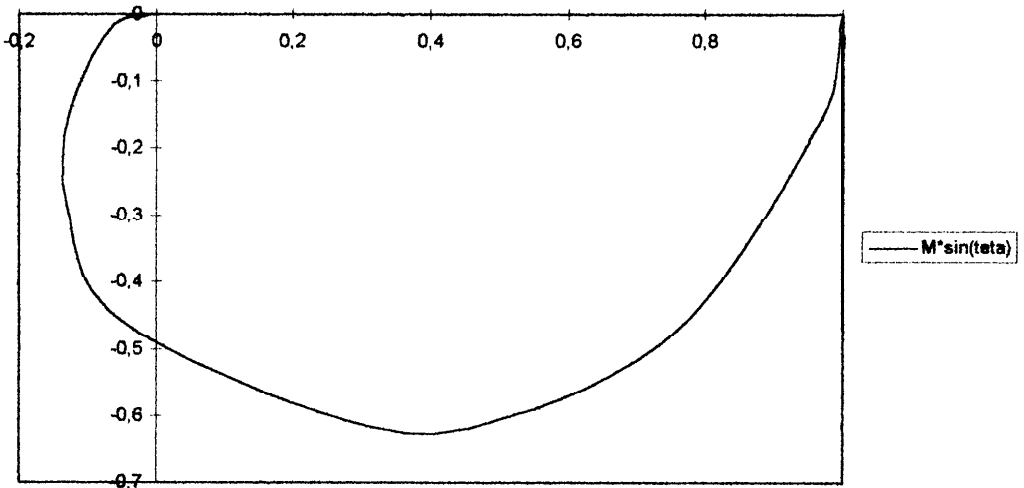
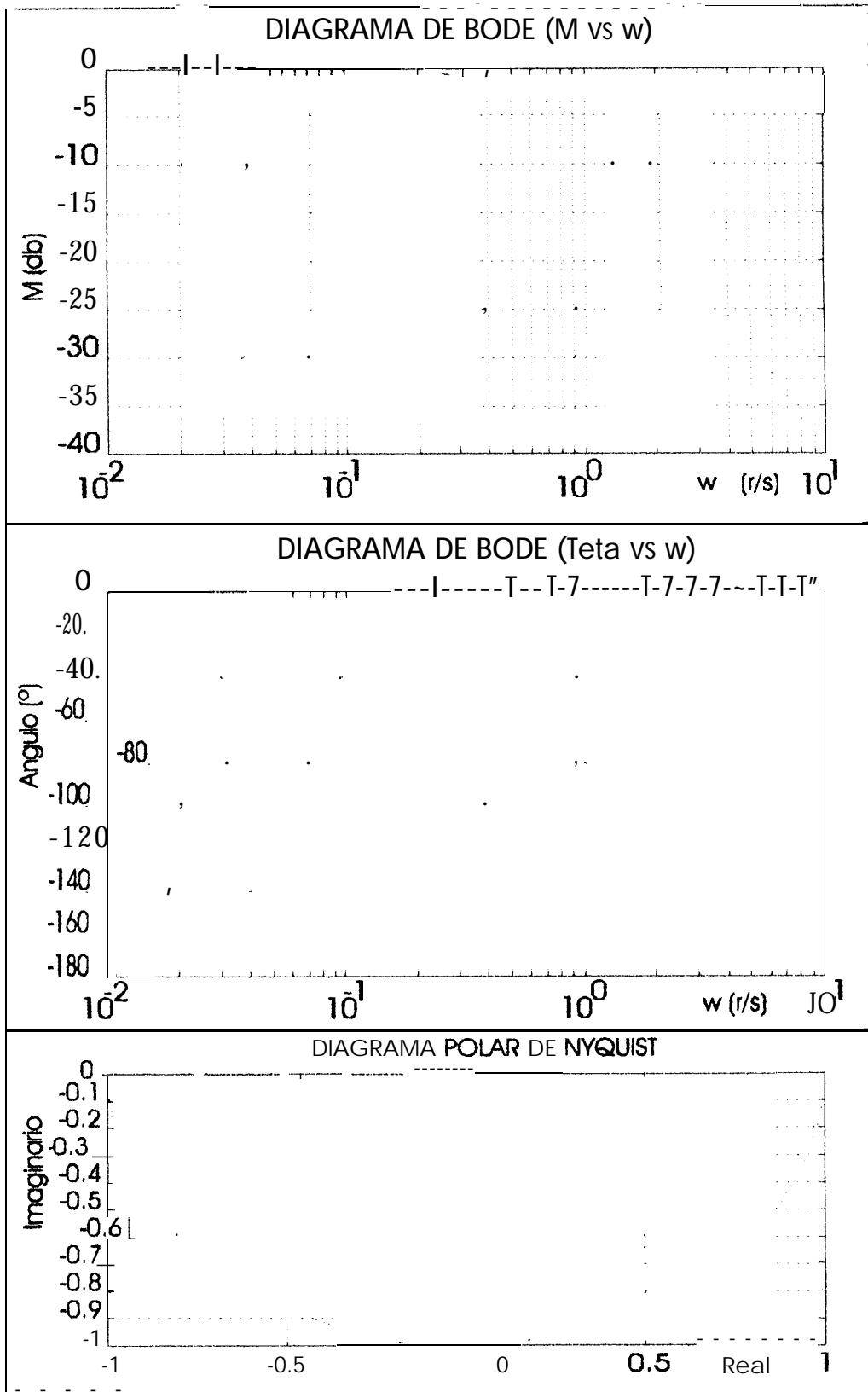


DIAGRAMA POLAR DE NYQUIST BASADO EN LOS DATOS OBTENIDOS DEL APÉNDICE (G)



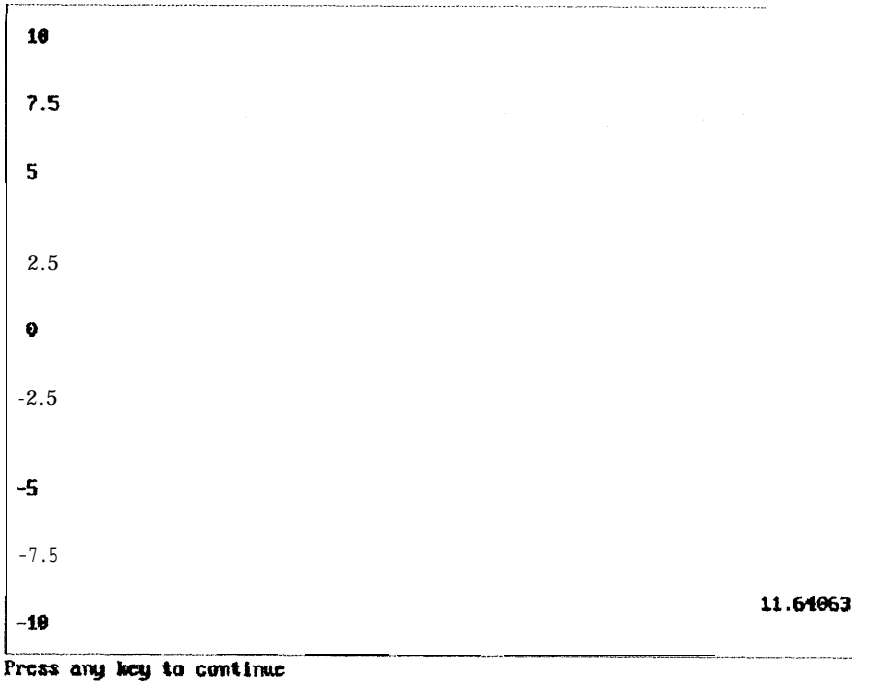
A partir del diagrama de Bode (M vs w) se puede encontrar la función de transferencia del sistema, notándose claramente la pendiente de -40 db. por década que indicaría un sistema de segundo orden con ganancia unitaria.

APÉNDICE I

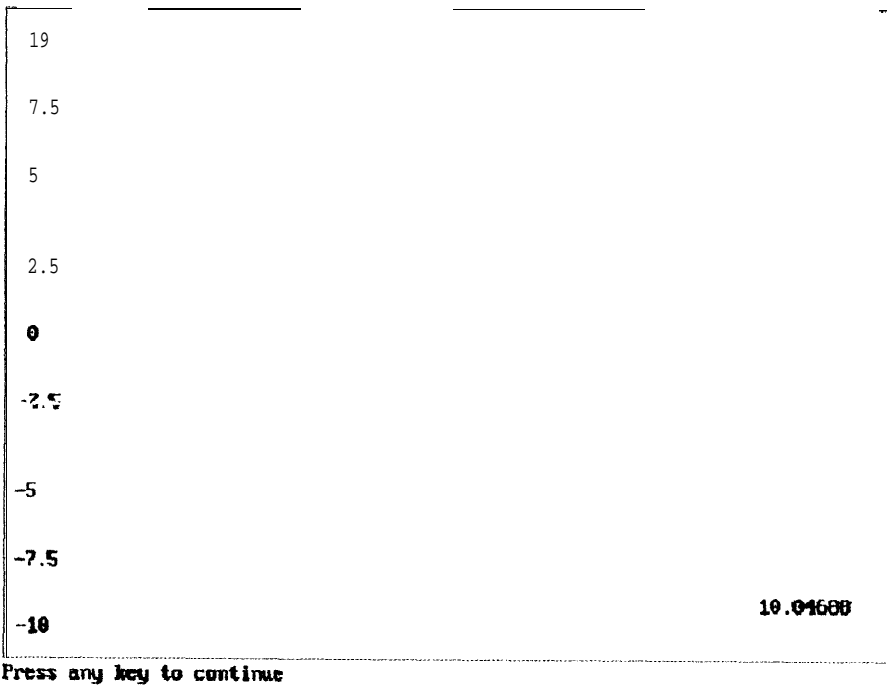


Gráficos comparativos con los del APÉNDICE H basados en la función de transferencia teórica descrita por la ecuación 4.3.1 del capítulo 4, analizadas y grafkadas por el programa **MATLAB**.

APÉNDICE J



Método de la curva de reacción



Método de ganancia última

1. FEEDBACK INSTRUMENTS LIMITED, Process Control Simulator PCS 327, Editorial FI Ltd, Crowborough, England.
2. SMITH * CORRIPIO, Control Automático de Procesos, Editorial Limusa Noriega Editores, México 1995.
3. EDISON MARIN, Implementación de un sistema didáctico para adquisición de datos y control de procesos por computadora, Facultad de Ingeniería Mecánica, 1997
4. BENJAMIN C. KUO, Matlab Tools for Control System Analisis And Design, Editorial Prentice Hall International, New Jersey, 1994
5. S. THOMPSON, Control System Engineering And Design, Longman Scientific & Technical, Hong Kong, 1989