

**ESCUELA POLITECNICA NACIONAL  
FACULTAD DE INGENIERIA ELECTRICA**

**TESIS DE GRADO**

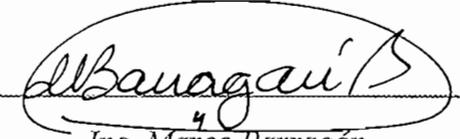
**“ MODELACION, CONTROL Y SIMULACION DINAMICA DEL  
SERVOMECANISMO DE POSICION MOTOMATIC MCL-100 Y  
DEL SISTEMA DE NIVEL DE LIQUIDOS EN  
TANQUES ACOPLADOS ”**

**TESIS PREVIA LA OBTENCION DEL TITULO DE INGENIERO  
EN ELECTRONICA Y CONTROL**

**JUAN E. RAMOS P.**

**Junio, 1996**

*Certifico que el presente trabajo  
ha sido elaborado en su totalidad  
por el señor Juan Ramos P.*

  
Ing. Marco Barragán  
Director de tesis

## Agradecimiento

*Al Ingeniero Marco Barragán  
mi más sincero y profundo  
agradecimiento por la com-  
prensión y apoyo brindado para  
la realización de la presente tesis*

## Dedicatoria

*En esta nueva etapa de mi vida quiero agradecer a todas las personas que en su momento me brindaron toda su ayuda y comprensión. Por ello quiero dedicar esta tesis a mi madre, a mi esposa, y a mis hijos Eduardo y Andrés quienes me han estado acompañando en los mejores pasajes de mi vida.*

*A Dios por haberme cuidado y ayudado en cada instante de mi existencia.*

# I N D I C E

Página

## CAPITULO I

### INTRODUCCION

1.1. Importancia del tema. ....	1
1.2. Justificación sobre la selección de las plantas. ....	2
1.3. Modelos discretos. ....	3
1.3.1. Método de la transformada Z. ....	4
1.3.2. La transformada Z Inversa.....	5
1.4. Contenido. ....	6

## CAPITULO II

### SERVOMECANISMO DE POSICION MOTOMATIC MCL-100.

2.1. Descripción física del MOTOMATIC MCL-100. ....	7
2.2. Modelación del sistema. ....	8
2.3. Funciones de transferencia y diagramas de bloques. ....	14
2.4. Discretización del sistema. ....	15
2.4.1. Discretización de la función de transferencia que representa a la velocidad angular. ....	16
2.4.2. Discretización del control P.I.D. ....	17
2.4.3. Discretización de una red. ....	18
2.5. Análisis del sistema sin compensar. ....	19
2.6. Compensación del sistema. ....	22
2.6.1. Realimentación de velocidad. ....	24
2.6.2. Ubicación arbitraria del cero de una red. ....	26
2.6.3. Compensación por acciones de control. ....	29

## CAPITULO III

### SISTEMA DE NIVEL DE LIQUIDOS EN TANQUES ACOPLADOS.

3.1. Modelación del sistema de Tanques Acoplados.....	32
3.1.1. Modelación de las bombas centrífugas.....	34
3.1.2. Modelación de los transductores capacitivos.....	36
3.2. Descripción física del sistema en variables de estado.....	36
3.3. Controlabilidad del sistema.....	39
3.4. Discretización del sistema.....	42
3.5. Análisis del sistema sin compensar.....	44
3.6. Compensación del sistema mediante la utilización del regulador cuadrático lineal.....	48

## CAPITULO IV

### IV SIMULACION DINAMICA DE LOS SISTEMAS.

4.1. Algoritmo de cálculo matemático y simulación dinámica para el servomecanismo de posición MOTOMATIC MCL-100.....	54
4.1.1. Resultados Obtenidos de la simulación del Motomatic.....	58
a) Realimentación de velocidad.....	61
b) Ubicación arbitraria del cero de una red.....	65
c) Acciones de control.....	68
4.2. Algoritmo de cálculo matemático y simulación dinámica para el sistema de nivel de líquidos en Tanques Acoplados.....	73
4.2.1. Algoritmo de discretización del sistema.....	76
4.2.2. Algoritmo para calcular el Regulador Cuadrático Lineal.....	79
4.2.3. Macros utilizados en el Programa CC.....	82

4.2.4. Resultados Obtenidos de la simulación de los Tanques Acoplados. ....	83
---	----

## **CAPITULO V**

### **CONCLUSIONES. Y RECOMEDACIONES**

5.1. Conclusiones . ....	100
5.2. Recomendaciones . ....	103

### **APENDICES**

A.- Manual de utilización. ....	104
B.- Listado de programas.	
Listado del Motomatic MCL-100      (MOTOR.BAS).....	114
Listado De los Tanques ACOPLADOS   (TANQUES.BAS).....	148

### **BIBLIOGRAFIA.**

# *CAPITULO I*

## *INTRODUCCION*

### *1.1 IMPORTANCIA DEL TEMA.*

El estudio del control de procesos requiere complejos modelos e instrumentos matemáticos para un profundo y eficaz análisis de todas las características de funcionamiento del sistema. Para excluir este problema, a menudo se utilizan unos instrumentos alternativos para sustentar esta teoría, que permiten el estudio del proceso real sobre modelos reducidos o sobre oportunos simuladores

Primeramente debemos reconocer que muchos sistemas físicos son inherentemente muestreados, o su comportamiento puede ser descrito por datos muestreados o modelos digitales. Además, el computador digital es utilizado por la ingeniería de sistemas de control por dos propósitos básicos, el primero es que son usados extensamente para la simulación y computación de sistemas de control dinámicos. Además, la ingeniería de control a menudo cuenta con la simulación en computador digital, para conducir el análisis y el diseño de sistemas complejos los cuales no adoptan alguna de las herramientas analíticas establecidas; por lo tanto, podemos decir que la simulación computacional es a menudo utilizada para chequear los resultados obtenidos por medios analíticos.

Entonces la existencia de un programa que represente y dinamice en la pantalla de un computador el montaje del sistema es muy ventajoso ya que de esta manera el estudiante se centra en el problemas fundamental, dedicando mayor tiempo a la búsqueda alternativas de control, dejando de lado la experiencia adicional en el montaje y diseño del experimento, mejorando de esta manera la calidad del aprendizaje y sin necesidad de utilizar equipos que eventualmente no podrían ser encontrados en el laboratorio, con fines de calibración, etc.

Además, la falta de recursos y la disponibilidad de tiempo para el desarrollo de prototipos de laboratorio son otra causa que han determinado la necesidad de contar con experimentos de fácil adquisición e implementación y que no descuiden a la vez la calidad del aprendizaje que se desprende de ellos, por tales razones, el objetivo fundamental del estudio propuesto es el de emular en un computador el comportamiento de dos sistemas físicos, de tal manera de presentar la dinámica de los mismos como si se hubiera montado el experimento en el laboratorio.

## *1.2 JUSTIFICACIÓN SOBRE LA SELECCIÓN DE LAS PLANTAS.*

Con el desarrollo de la tesis a consideración, lo que se busca es poder aplicar algunos tipos y técnicas de control, con el fin de obtener respuestas que el diseñador pueda considerar aceptables, esto con el objetivo de tratar de cubrir en la medida de lo posible la materia que al respecto se estudia en la especialización y además la posibilidad de poder explicar la teoría del control discreto en las plantas a considerar.

El estudio de sistemas automáticos de control provee una inusual oportunidad para que el estudiante de ingeniería pueda combinar sus conocimientos de principios de dinámica, análisis y síntesis de circuitos, conceptos de control de realimentación, análisis de la función de transferencia, técnicas de compensación, etc; siendo estas las causas para la utilización del servomecanismo de posición Motomatic.

Además la existencia en los laboratorios de control de la facultad del servomecanismo de posición, nos permitirá enfocar la realización de estudios comparativos entre el sistema real y los resultados obtenidos en la simulación.

Un problema que ocurre comúnmente en control de procesos químicos industriales es el control de nivel de fluidos en tanques de almacenamiento, mezclas químicas y tubos de reacción. Una situación típica es aquella en la que se requiere suplir una cantidad de fluido constante a un reactor químico, donde esto puede ser llevado a cabo mediante la implementación de un lazo de realimentación, el cual mantiene el nivel del líquido constante mediante el control del flujo de entrada de líquido.

El control del nivel del fluido y de la altura, es similarmente importante en sistemas de generación de potencia, donde el control de nivel de fluido en calderos es una parte central de reguladores de generación de vapor. En la misma línea, pero en mucha mayor escala, la distribución y la planificación de las fuentes de agua, requerirá las ideas de nivel de fluido, flujo y su respectiva regulación en una medida muy vasta. Por ello, el control de nivel de fluidos, es un problema básico e importante en el control automático y además debido a que las propiedades dinámicas de sistemas de nivel de fluido son naturalmente no lineales, ellas entonces involucran además problemas desafiantes. En particular, la escala

o constante de tiempo en sistemas de control de fluidos es comparativamente grande con relación a otros sistemas, por ejemplo es usual encontrar constantes de tiempo de varias horas y aún de días.

### 1.3 MODELOS DISCRETOS.

En años recientes progresos significativos han sido hechos en cuanto a datos discretos y sistemas de control digital, estos sistemas han ganado popularidad e importancia en todas las industrias debido en parte a los avances hechos en las computadoras digitales, y más recientemente en microcomputadoras.

Datos discretos y sistemas de control digital difieren de los datos continuos o sistemas de control análogos en que las señales en una o más partes de esos sistemas están en forma de un tren de pulsos o un código numérico. Estrictamente el término datos muestreados se refieren a una señal que es modulada por amplitud de pulso, esto es un tren de pulsos cuya información es llevada por amplitudes, mientras que datos digitales usualmente se refieren a señales las cuales son generadas por un computador digital o transductores digitales y además tienen en algunos casos formas codificadas.

Los sistemas en tiempo discreto o sistemas de datos muestreados, son sistemas dinámicos en los cuales una o más variables pueden variar solamente en ciertos instantes. Esos instantes que se han de indicar por  $kT$  ( $k=0,1,2,\dots$ ), pueden especificar en momento en que se realiza alguna medición física o el tiempo en el cual se lee la memoria en un computador digital, etc.; se toma el intervalo de tiempo entre dos instantes de tiempo suficientemente pequeños, de manera que los datos de tiempo entre esos instantes, puedan ser aproximados por interpolación simple.

Los sistemas de tiempo discreto difieren de los de tiempo continuo, en que las señales para un sistema de tiempo discreto, aparecen en forma de datos muestreados; en la práctica se presentan los sistemas de tiempo discreto, cuando se obtienen las mediciones necesarias para el control en una forma intermitente o cuando se comparte un control de gran envergadura a través de una computadora entre diversas plantas, de manera que se envía una señal de control a cada planta periódicamente, siendo estas las encargadas de realizar los cálculos necesarios para el control.

Muchos sistemas de control industrial modernos, son sistemas de tiempo discreto porque invariablemente influyen algunos elementos cuyas entradas y/o salidas son discretas en el tiempo. Sin embargo algunas veces la discretización con operación de muestreo, puede ser enteramente ficticia e introducida únicamente para simplificar el análisis de un sistemas de control que en realidad solo contienen elementos continuos.

La introducción de un computador digital en un sistema, por lo demás analógico produce señales en forma digital en parte del sistema, entonces dicho sistema toma la forma de una combinación mixta análogo digital. La conversión de una señal analógica en la correspondiente señal digital es una aproximación, ya que la señal analógica puede tomar un infinito número de valores, mientras que la cantidad de valores que se puede formar con un determinado número finito de dígitos es limitada. Dicho proceso de aproximación se llama cuantificación.

### 1.3.1. METODO DE LA TRANSFORMADA Z.

Al utilizar una simulación digital, se debe tener un modelo discreto desde el punto de vista de estructura; existiendo modelos aproximados para obtener la transformada  $Z$ , de tal manera que nos permitan obtener modelos aproximados que representen bastante bien al modelo continuo. Y para ello se debe considerar los siguientes parámetros:

- Que tan bien se aproxima la región de estabilidad.
- Que tanto se aproximan las raíces a las exactas.
- Que tanto se aproximan las respuestas del sistema continuo como discreto.

Se puede realizar fácilmente el análisis de sistemas en tiempo discreto mediante el método de la *transformada Z* o en el método del espacio de estado. El método de la transformada  $Z$  tiene la misma relación respecto a los sistemas invariantes en el tiempo y de tiempo discreto, que el método de la transformada de *Laplace* con respecto a los sistemas lineales invariantes en el tiempo y de tiempo continuo.

Ahora se define a la transformada  $Z$  tomando la transformada de Laplace de la señal muestreada  $f^*(t)$ , entonces se tiene:

$$F^*(s) = \mathcal{Z} [f^*(t)] = \sum_{k=0}^{\infty} f(kT) e^{-kTs}$$

Se define a  $e^{Ts} = Z$  y se escribe a  $F^*(S)$  como  $F(z)$ , entonces se tiene:

$$F(z) = F^*(s) = \mathcal{Z} [f^*(t)] = \sum_{k=0}^{\infty} f(kT) Z^{-k}$$

A la función  $F(z)$  se la denomina la transformada  $Z$  de  $f^*(t)$  y la notación para la transformada  $Z$  de  $f^*(t)$  es  $\mathcal{Z} [f^*(t)]$ ; en la transformada  $Z$  sólo se consideran los valores de la señal en los instantes de muestreo, por lo tanto la transformada  $Z$  de  $f(t)$  y la de  $f^*(t)$  dan el mismo resultado, es decir:

$$\mathcal{Z} [F(s)] = \mathcal{Z} [f^*(t)] = \mathcal{Z} [f(t)]$$

### 1.3.2. LA TRANSFORMADA Z-INVERSA.

Como ya es conocido la transformada de Laplace y su inversa son únicas, tanto si  $F(S)$  es la transformada de Laplace de  $f(t)$ , entonces  $f(t)$  es la transformada inversa de Laplace de  $F(S)$ . Sin embargo en la transformada  $Z$ , la transformada inversa no es única, es decir que si la transformada  $Z$  de  $f(t)$  es  $F(Z)$ , la transformada inversa de  $F(Z)$  no es necesariamente  $f(t)$ . Un correcto resultado sería decir que la transformada inversa de  $F(Z)$  es  $f(kT)$  la cual es igual a  $f(t)$  únicamente en los instantes de muestreo.

El problema de la no unicidad de la transformada  $Z$  inversa es una de las limitaciones de dicha transformación, y esta propiedad debería ser recordada cuando se aplique la técnica de la transformada  $Z$ .

$$f(kT) = Z^{-1} [F(Z)] = \text{Transformada } Z \text{ inversa de } F(Z)$$

En general la transformada  $Z$  inversa puede ser calculada mediante la expansión de la función en fracciones parciales, de una manera similar al método utilizado en la Transformada  $S$ . La razón de encontrar dichas fracciones parciales en términos de  $A/z + a$  es tratar de utilizar una tabla de transformaciones, aunque esta represente un tren de pulsos retrasados con una amplitud que decae exponencialmente si  $a$  es un número positivo.

Sin embargo para la simulación en un computador digital se acostumbra a tratar de representar a la función en términos de retrasos unitarios  $Z^{-1}$ , es decir se trata de encontrar la variable requerida, en función de ecuaciones de diferencias.

$$\begin{array}{ccc}
 E(z) & \xrightarrow{\quad} & \boxed{D(z)} & \xrightarrow{\quad} & U(z) & \quad D(z) = \frac{U(z)}{E(z)}
 \end{array}$$

$$u(k) = \text{función}[u(k-1), u(k-2), \dots, e(k), e(k-1), \dots]$$

#### 1.4. CONTENIDO.

El presente literal tiene por objeto realizar una descripción de las organización de la tesis, la misma que está dividida en cinco capítulos y dos apéndices. En el capítulo I se realiza una introducción al tema de tesis, así como una justificación sobre la selección de la plantas y una breve explicación de los sistemas discretos que nos servirán de pauta para el desarrollo del programa implementado.

En el capítulo II se realiza la modelación y todo el desarrollo matemático de el Motomatic, así como su discretización; además se hace un estudio del sistema sin compensación y un análisis de las técnicas de control a implementarse en la simulación.

En el capítulo III se hace la modelación del sistema de Tanques acoplados, así como su necesaria linealización para un punto de operación definido; además se analiza la controlabilidad del sistema, discretización del mismo y el análisis del sistema sin compensar. Finalmente en este capítulo se analiza la compensación de los tanques acoplados mediante la utilización del Regulador Cuadrático Lineal.

En el capítulo IV se presentan los algoritmos utilizados para el funcionamiento del programa de simulación, tanto del Motomatic como de los Tanques Acoplados; para luego incluir los resultados obtenidos con los mismos, y que serán comparados con los obtenidos en los capítulos II y III.

En el capítulo V se exponen las conclusiones y las recomendaciones derivas del trabajo. Finalmente se detalla en los apéndices, el manual de uso del programa y sus respectivos listados.

*CAPITULO II*

*ANALISIS MATEMATICO DEL  
SERVOMECANISMO DE POSICION MOTOMATIC  
MCL-100*

## 2.1 DESCRIPCIÓN FÍSICA DEL MOTOMATIC MCL 100

EL laboratorio de sistema de control MOTOMATIC, consiste del mejor servomotor Dc de precisión disponible, convenientemente empaquetado para la máxima versatilidad, utilizable para experimentación y descubrimiento de principios básicos de la realimentación de control. El equipo ha sido diseñado para ser flexible de tal manera que otros experimentos básicos puedan ser realizados con y en adición de desarrolladores, de tal manera de obtener una instrucción fundamental acerca de los principios de control automático.

El servomecanismo MOTOMATIC consta básicamente de los siguientes componentes:

1. Chasis electrónico, sobre el cual se debe realizar el cableado.
2. Riel: para montar los componentes electromecánicos como el motor - generador, el reductor de velocidad, la carga inercial y el potenciómetro indicador de posición.
3. Sistema motor-generador con un eje común.
4. Unidad reductora de velocidad.
5. Rueda para la introducción de la carga inercial.
6. Potenciómetro indicador de posición.
7. Elementos, como resistencias y cables para conexión.
8. Acopladores mecánicos.
9. Chasis con los medidores (amperímetro, tacómetro)

Existe un amplificador lineal que permite conectar compensaciones tanto en serie como en paralelo; el amplificador como los elementos conectados a éste, pueden constituir una compensación en cascada. Normalmente los elementos conectados al amplificador deberán ser construidos.

Como precaución deberá indicarse que no se deberá conectar ningún generador o fuente de señal a la entrada del amplificador, si no es a través de las resistencias del punto de suma.

El amplificador lineal de potencia está acompañado de tres etapas de amplificación, mediante la utilización de transistores complementarios de silicio con ganancia total de voltaje de aproximadamente 5.

Existe una fuente de +/- 30 VDC de alta corriente para el amplificador de potencia, con un rizado que no excede del 15% bajo condiciones de plena carga.

Una de +/- 21 VDC utilizada como voltaje de referencia para la entrada y el indicador de posición, el rizado máximo es menor al 1%.

## 2.2 MODELACION DEL SISTEMA.

Un diagrama esquemático del circuito se muestra en la figura:

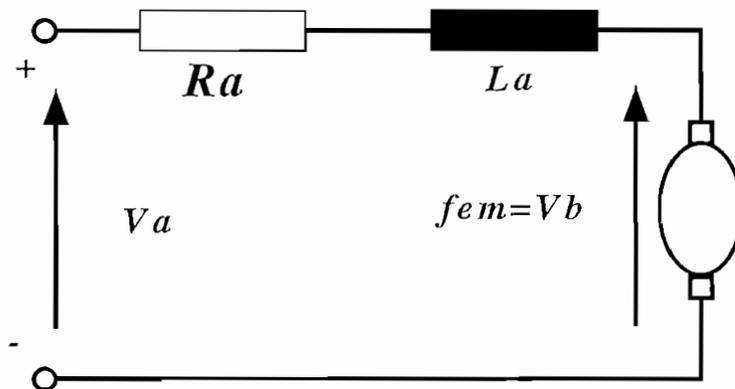


Fig. 2.1. Diagrama eléctrico de un motor DC

$$V_a = I_a * R_a + L_a * \frac{d}{dt} I_a + V_b \quad (2-1)$$

Veamos ahora la ecuación que describe el comportamiento mecánico del circuito donde:

$J_m$  = Momento de inercia de la armadura del motor incluyendo las conexiones mecánicas a ésta .

$f_m$  = Coeficiente de fricción viscosa de la armadura del motor .

$\theta_m$  = Posición angular de la armadura.

La ecuación de torque viene dada por:

$$J_m * \ddot{\theta}_m(t) + f_m * \dot{\theta}_m(t) = T_m(t) \quad (2-2)$$

$$T_m(t) = k_t * I_a(t) \quad (2-3)$$

$$V_b(t) = k_b * \dot{\theta}_m(t) \quad (2-4)$$

tomando la transformada de La Place de las ecuaciones anteriores y combinándolas tenemos:

$$\frac{\theta(s)}{Ia(s)} = \frac{Kt}{s(JmS + fm)} \quad (2-5)$$

$$\frac{W(s)}{Ia(s)} = \frac{Kt}{JmS + fm} \quad (2-6)$$

$$\frac{\theta(s)}{Va(s)} = \frac{Kt}{S [LsJmS^2 + (RaJm + Lafm)S + (Rafm + KbKt)]} \quad (2-7)$$

Del análisis realizado y tomando en cuenta los resultados obtenidos en la ecuación (2-7), tenemos el circuito equivalente de la Figura 2.2., donde se muestra el servomecanismo MOTOMATIC MCL 100 en función de los parámetros físicos propios del mecanismo, y donde no se considera el efecto de las perturbaciones.

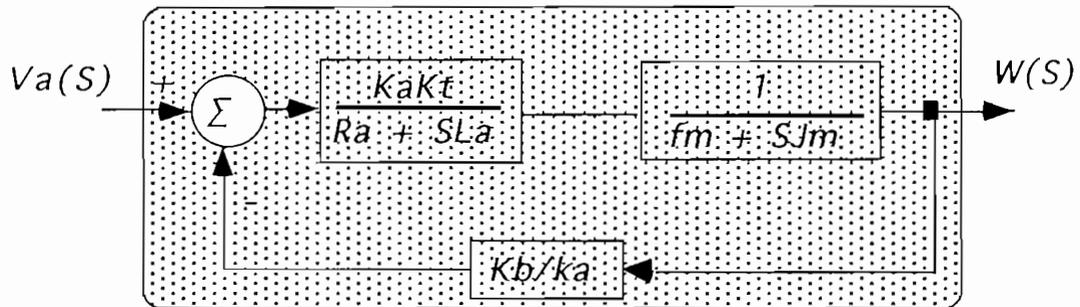


Fig. 2.2. Diagrama de bloques del Motomatic MCL 100

La determinación de los parámetros que definen el comportamiento del MOTOMATIC<sub>1</sub> fueron obtenidos del manual de funcionamiento del mismo (Referencia 6).

### REDUCTOR DE VELOCIDAD

Es un dispositivo mecánico de dos bandas, mediante el cual nosotros podemos obtener algunas relaciones de velocidad y por lo tanto de posición. En la figura (2.3) se ilustra este sistema con las posiciones que puede ocupar la banda y las posibles relaciones de velocidad que se obtiene:

Para el reductor de velocidad tenemos la siguiente función de transferencia:

$$\frac{Wout(s)}{Win(s)} = N \quad \Rightarrow \quad \frac{\theta o(s)}{\theta i(s)} = N \quad (2-8)$$

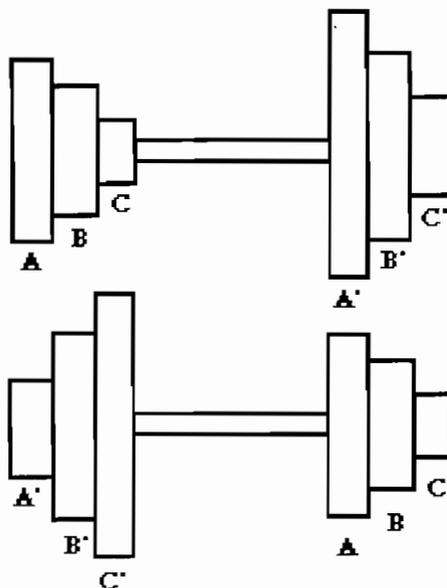


Fig. 2.3. Dispositivo mecánico de Reducción de velocidad

**RELACIONES DE VELOCIDAD**

$$1 : 1$$

$$1 : \sqrt{3}$$

$$1 : 3$$

$$1 : 3\sqrt{3}$$

$$1 : 9$$

**BANDA EN:**

AA'

BA' : AB'

CA' : BB' : AC'

CB' : BC'

CC'

**TACOMETRO.**

Para la realimentación de velocidad utilizamos un tacogenerador de DC acoplado al eje de la máquina, con un campo magnético de imán permanente, por lo tanto podemos asumir que el flujo es constante, y que la salida es proporcional a la velocidad angular de la armadura, así tenemos:

$$Eg(t) = Kg * w(t) \quad (2-9)$$

Y considerando la transformada de Laplace de la ecuación anterior tenemos:

$$\frac{Eg(S)}{W(S)} = Kg \quad (2-10)$$

## SENSOR DE POSICION

Como sensor de posición se utiliza un potenciómetro circular de simple vuelta de 10 K con una parada no mecánica que es utilizada para poder sensar la posición angular y donde existe una zona muerta de  $20^\circ$ .

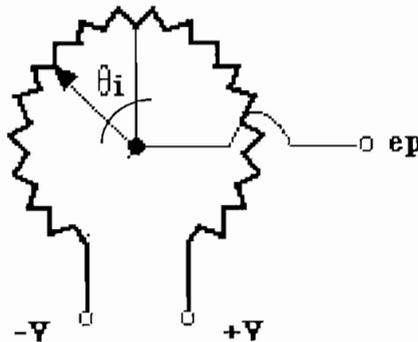


Figura 2.4. Potenciómetro de Posición.

$$ep = kp * \theta - V$$

si se redefine el ángulo de tal manera que:

$$-170^\circ \leq \theta_i \leq 170^\circ$$

tenemos :

$$ep(t) = kp * \theta_i(t) \quad (2-11)$$

de tal manera que:

$$-V_{DC} \leq ep \leq V_{DC}$$

La polaridad de  $E_p$  puede ser cambiada por medio de un switch en al chasis, y este signo puede ser seleccionado de acuerdo a la realimentación requerida.

## SUMADOR.

El punto de suma es un circuito sumador resistivo a cuyos terminales se conectarán ya sea la señal de referencia, como cualquiera de las señales de realimentación, así tenemos el circuito mencionado que se muestra en la figura (2.5):

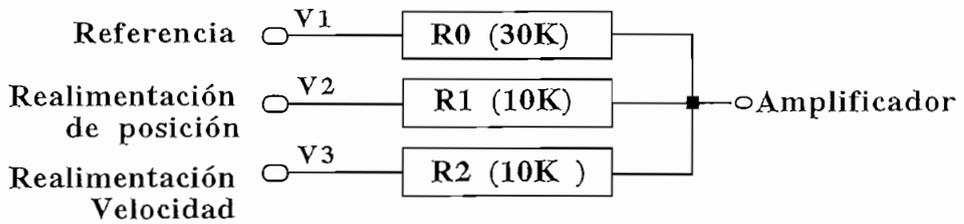


Figura 2.5. Diagrama del punto de suma.

Consideramos que vamos a utilizar un máximo de 3 entradas, es decir: la referencia, la realimentación de velocidad, y la realimentación de posición; obtenemos entonces el circuito de la figura 2.6, donde se muestra conectado el punto de suma a un amplificador operacional.

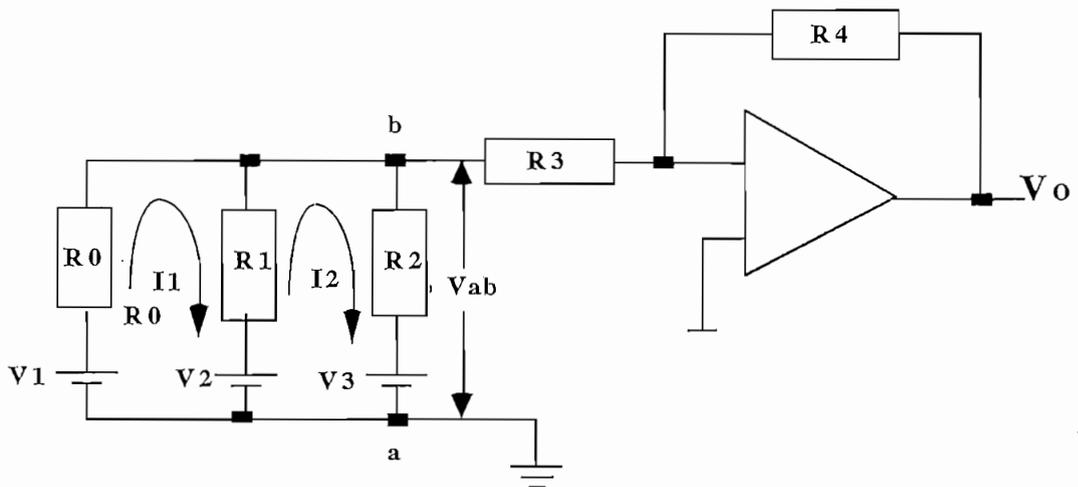


Figura 2.6. Equivalente eléctrico del punto de suma resistivo.

Si definimos las resistencias en función de  $R_0$  tenemos:

$$R_1 = K_1 R_0$$

$$R_2 = K_2 R_0$$

$$R_3 = K_3 R_0$$

$$R_4 = K_4 R_0$$

Procedemos ahora a determinar la expresión que representa la señal de salida  $V_o$ , para lo cual primeramente planteamos la ecuaciones de malla, así tenemos:

$$V_1 - V_2 = (R_0 + K_1 R_0) I_1 - K_1 R_0 I_2 \quad (2-12)$$

$$V_2 - V_3 = K_1 R_0 I_1 - (K_1 R_0 + K_2 R_0) I_2 \quad (2-13)$$

Resolviendo el sistema para encontrar el valor de  $I_1$ , tenemos:

$$I_1 = \frac{K_1 V_1 - K_2 V_2 + K_2 V_1 - K_1 V_3}{(K_1 + K_2 + K_1 K_2) R_0} \quad (2-14)$$

donde el voltaje Thévenin  $V_{ab}$  viene dado por la expresión:

$$V_{ab} = V_1 - I_1 R_0$$

$$V_{ab} = V_1 - \frac{K_1 V_1 - K_2 V_2 + K_2 V_1 - K_1 V_3}{(K_1 + K_2 + K_1 K_2)}$$

$$V_{ab} = \frac{K_1 K_2 V_1 + K_2 V_2 + K_1 V_3}{K_1 + K_2 + K_1 K_2}$$

y la impedancia  $Z_{th}$  de salida es el paralelo de las resistencias, la cual viene dada por:

$$Z_{th} = \frac{1}{\frac{1}{R_0} + \frac{1}{K_1 R_0} + \frac{1}{K_2 R_0}} = \frac{K_1 K_2 R_0}{K_1 + K_2 + K_1 K_2}$$

Obteniendo el circuito equivalente que se muestra a continuación.

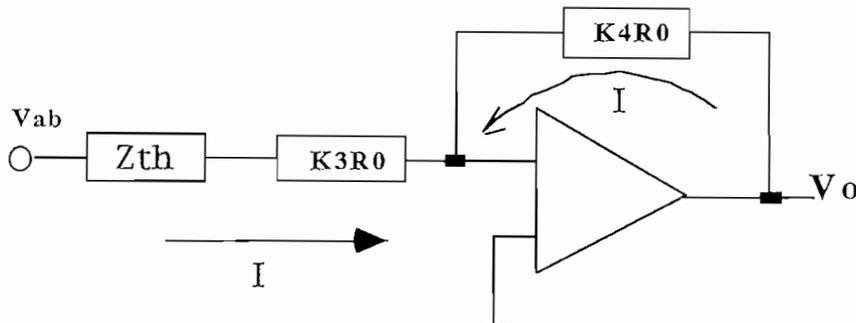


Fig. 2.7. Equivalente Thevenin del punto de suma resistivo

la corriente  $I$  viene definida por la siguiente expresión:

$$I = \frac{V_{ab}}{Z_{th} + K_3 R_0}$$

$$I = \frac{K_1 K_2 V_1 + K_2 V_2 + K_1 V_3}{(K_1 K_2 + K_3(K_1 + K_2 + K_1 K_2)) R_0}$$

Ahora calculamos el voltaje de salida  $V_o$ , el cual viene dado por:

$$V_o = -K_4 R_0 \cdot I = -K_4 \frac{K_1 K_2 V_1 + K_2 V_2 + K_1 V_3}{K_1 K_2 + K_3(K_1 + K_2 + K_1 K_2)}$$

$$V_o = -\frac{K_1 K_2 K_4}{K_1 K_2 + K_3(K_1 + K_2 + K_1 K_2)} \left[ V_1 + \frac{1}{K_1} V_2 + \frac{1}{K_2} V_3 \right] \quad (2-15)$$

Si analizamos la ecuación (2-15), obtenemos el diagrama de bloques equivalente del punto de suma resistivo (Figura 2.8), el cual es función de la ganancia del amplificador, y cuya relación viene dada por la ecuación (2-16), donde *gan* representa dicha ganancia equivalente.

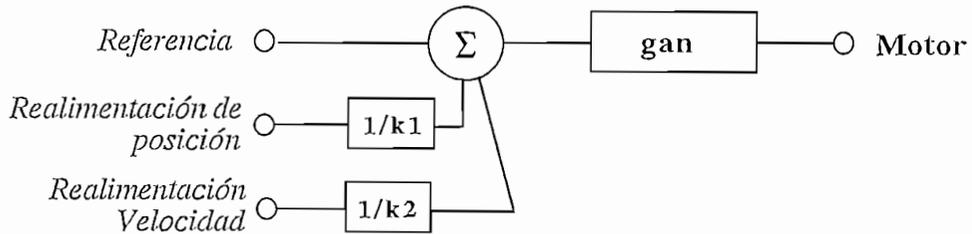


Fig. 2.8. Circuito Equivalente del punto de suma resistivo

$$gan = \frac{K_1 K_2 K_4}{K_1 K_2 + K_3(K_1 + K_2 + K_1 K_2)} \quad (2-16)$$

### 2.3 FUNCIONES DE TRANSFERENCIA Y DIAGRAMAS DE BLOQUES.

Utilizando la ecuación (2-7), y reemplazando por los parámetros físicos del MOTOMATIC, tenemos la siguiente función de transferencia:

$$\frac{\theta(S)}{V_a(S)} = \frac{KA}{S(1/T_1 + S)(1/T_2 + S)} = G(S) \quad (2-17)$$

donde:

$$KA = 1623 * 11.5 * 110 * 180 / \pi$$

$$1/T_1 = 1623 \quad \text{y} \quad 1/T_2 = 11.5$$

y donde las constantes tanto del tacogenerador de DC, como del reductor de velocidad vienen dadas por:

$$Kg = 0.148 * \pi / 180 \quad \text{Volt} * \text{Seg} / \text{Grad}$$

$$Kp = 6.74 * \pi / 180 \quad \text{Volt} / \text{Grad}$$

$$N = \text{Relación de reducción de velocidad}$$

Considerando la Fig. 2.8, tenemos que la entrada 2 corresponde a la realimentación de posición tomada a la salida del reductor de velocidad, mientras que la entrada 3 corresponde a la realimentación de velocidad ; con estas consideraciones obtenemos la función de transferencia del lazo de realimentación  $H(S)$ , de la forma:

$$H(s) = \frac{V_{med}(s)}{\theta_o(s)} = \frac{k_p * N}{k_1} + \frac{k_g}{k_2} S = b + aS \quad (2-18)$$

Al combinar las ecuaciones (2-17) y (2-18), se podemos llegar al diagrama de bloques completo , el cual representa la dinámica del sistema, y lo mostramos en la FIG. 2.9.

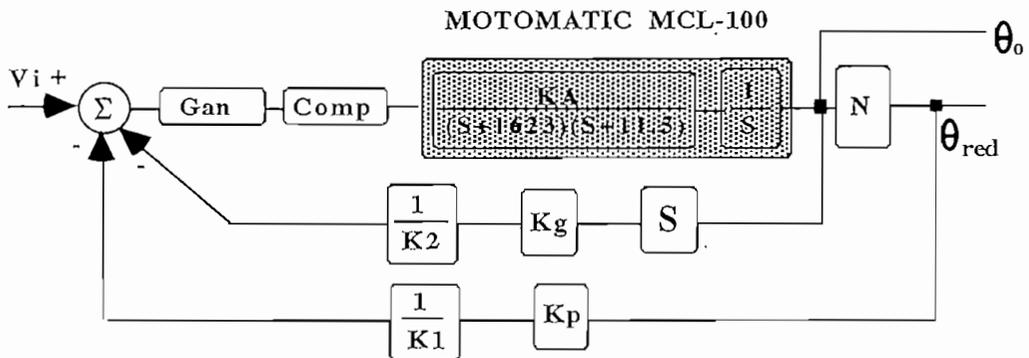


Fig. 2.9 Diagrama de bloque final utilizado para la simulación

#### 2.4. DISCRETIZACION DEL SISTEMA.

Como podemos observar, la función de transferencia del MOTOMATIC  $G(S)$  ecuación (2-17), tiene una transformada Z inmediata de tal manera que tenemos:

$$G(S) = \frac{KA}{S(1+ST1)(1+ST2)} = \frac{\theta_o(S)}{Va(S)} \quad (2-19)$$

$$G(S) = KA * \left[ \frac{1}{S} + \frac{T1}{T2-T1} * \frac{1}{\frac{1}{T1}+S} + \frac{T2}{T1-T2} * \frac{1}{\frac{1}{T2}+S} \right]$$

$$Z(G(S)) = KA * \left[ \frac{Z}{Z-1} + K1 \frac{Z}{Z-e^{-\frac{T}{T1}}} + K2 \frac{Z}{Z-e^{-\frac{T}{T2}}} \right]$$

$$G(Z) = KA * \left[ \frac{Z}{Z-1} + K1 \frac{Z}{Z-a} + K2 \frac{Z}{Z-b} \right]$$

Donde:

$$K1 = \frac{T1}{T2-T1} \quad y \quad K2 = \frac{T2}{T1-T2}$$

$$a = e^{-\frac{T}{T1}} \quad b = e^{-\frac{T}{T2}}$$

Si desarrollamos las expresiones tenemos:

$$G(Z) = \frac{(1+K1+K2)KA Z^3 - [a+b+(1+a)K2+(1+b)K1]KA Z^2 + (ab+K1b+K2a)KA Z}{Z^3 - (1+a+b)Z^2 + (a+b+ab)Z - ab}$$

$$G(Z) = \frac{a_0 Z^3 + a_1 Z^2 + a_2 Z}{b_0 Z^3 + b_1 Z^2 + b_2 Z + b_3} = \frac{\theta(z)}{Va(z)} \quad (2-20)$$

matemáticamente obtenemos que el término  $a_0 = 0$

#### 2.4.1. DISCRETIZACION DE LA FUNCION DE TRANSFERENCIA QUE REPRESENTA A LA VELOCIDAD ANGULAR.

Ahora mediante la utilización de la función de transferencia que representa a la velocidad angular ecuación (2-21), vamos a discretizarla para poder obtener dicha señal en términos de retardos unitarios, esto con el fin de poder realizar la realimentación de velocidad, y poder de esta manera analizar el comportamiento de la posición angular con dicha realimentación, así tenemos:

$$G1(S) = \frac{w(S)}{Va(S)} = \frac{KA}{(1+ST1)(1+ST2)} \quad (2-21)$$

descomponiendo en fracciones parciales la ecuación (2-21) tenemos:

$$G1(S) = \left[ \frac{KA}{T1-T2} * \frac{1}{\frac{1}{T1}+S} + \frac{KA}{T2-T1} * \frac{1}{\frac{1}{T2}+S} \right]$$

si obtenemos la transformada Z de la expresión anterior, tenemos:

$$Z(G1(S)) = \left[ K1 \frac{Z}{Z-e^{-\frac{T}{T1}}} + K2 \frac{Z}{Z-e^{-\frac{T}{T2}}} \right]$$

$$G1(Z) = KA * \left[ K1 \frac{Z}{Z-a} + K2 \frac{Z}{Z-b} \right]$$

Donde las constantes vienen definidas por:

$$K1 = \frac{KA}{T1-T2} \quad \text{y} \quad K2 = \frac{KA}{T2-T1}$$

$$a = e^{-\frac{T}{T1}} \quad \text{y} \quad b = e^{-\frac{T}{T2}}$$

si desarrollamos la expresión anterior, tenemos la siguiente función de transferencia en el plazo Z, así tenemos:

$$G1(Z) = \frac{(K1+K2) Z^2 - [aK2+bK1] Z}{Z^2 - (a+b)Z + ab} = \frac{w(z)}{Va(z)} \quad (2-22)$$

donde también se puede demostrar matemáticamente que  $K1 + K2 = 0$ .

Ahora bien, si consideramos todas las técnicas de compensación que vamos a utilizar en el programa de simulación, como son: ubicación arbitraria del cero de una red, controlador P.I.D., y cualquier tipo de compensador, todas estas funciones también deberán ser discretizadas.

#### 2.4.2. DISCRETIZACION DEL CONTROL P.I.D.

El circuito a implementar y a discretizar es el que se muestra en la figura:

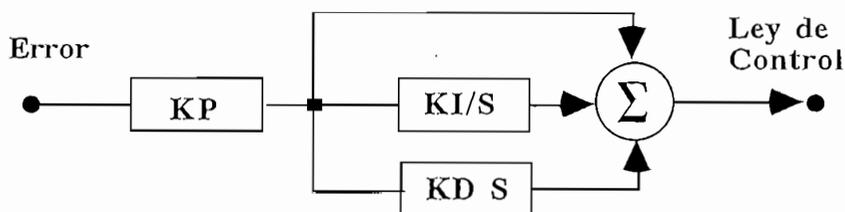


Fig 2.10. Diagrama esquemático de un controlador P.I.D

Escribiendo la relación en términos de Laplace tenemos:

$$U(S) = Kp \left( 1 + \frac{K_I}{S} + K_D S \right) E(S) \quad (2-23)$$

El mismo principio del control P.I.D. puede ser aplicado al control digital; el control proporcional es el mismo que en el caso continuo, mientras existen diferentes caminos de discretización del control integral y del derivativo. Si tomamos el método del trapecio para el control integral y la definición matemática de la derivada para el segundo, obtenemos los siguientes resultados en el plano de la transformada Z:

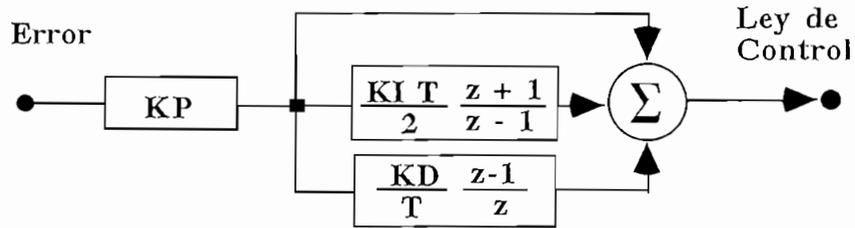


Fig. 2.11. Diagrama discretizado de un controlador P.I.D

Si realizamos las simplificaciones respectivas para representar al controlador en forma de ecuaciones de diferencias, podemos realizar la simulación respectiva del comportamiento del controlador P.I.D en base de los intervalos de muestreo:

$$G_c(s) = \frac{b_0 Z^2 + b_1 Z + b_2}{Z^2 - Z} = \frac{u(s)}{e(s)} \quad (2-24)$$

$$u[kT] = u[(k-1)T] + b_0 e(kT) + b_1 e[(k-1)T] + b_2 e[(k-2)T]$$

donde:

$$b_0 = Kp \left( 1 + \frac{K_I T}{2} + \frac{K_D}{T} \right)$$

$$b_1 = Kp \left( -1 + \frac{K_I T}{2} - \frac{2K_D}{T} \right)$$

$$b_2 = Kp \left( \frac{K_D}{T} \right)$$

#### 2.4.3. DISCRETIZACION DE UNA RED.

Ahora realizamos la discretización de una red por el método de ubicación arbitraria del cero de una red, para lo cual tomamos la función de transferencia de una red de adelanto/retraso de fase, así tenemos:

$$G_c(s) = K_{comp} \frac{S + zero}{S + polo} = \frac{u(s)}{e(s)} \quad (2-25)$$

Para la discretización de esta función utilizamos la transformada bilineal de Tustin, de manera que nosotros tenemos la siguiente expresión:

$$Z(Gc(s)) = Kcomp \frac{\frac{2}{T} \frac{Z-1}{Z+1} + \text{cero}}{\frac{2}{T} \frac{Z-1}{Z+1} + \text{polo}}$$

$$Gc(Z) = \frac{Kcomp (2+T*\text{cero})Z + Kcomp (-2+T*\text{cero})}{(2+T*\text{polo})Z + (-2+T*\text{polo})} \quad (2-26)$$

En cuanto a la utilización de cualquier otro tipo de compensador, el programa diseñado está hecho de tal manera que el interesado deberá ingresar el compensador previamente discretizado.

## 2.5 ANALISIS DEL SISTEMA SIN COMPENSAR

Para poder obtener mejores criterios o alternativas de compensación, vamos a analizar el comportamiento transitorio del sistema sin compensar, y para ello vamos a realizar este estudio mediante la utilización del lugar geométrico de las raíces.

El Lugar Geométrico de las raíces permite encontrar los polos de lazo cerrado partiendo de los polos y ceros de lazo abierto, tomando a la ganancia como parámetro; sin embargo a menos que se especifique lo contrario podría utilizarse cualquier otra variable de la función de transferencia de lazo abierto.

Consideramos primeramente que no existe realimentación de velocidad, ya que esta será tomada en cuenta como una alternativa de control; así nosotros consideramos la ecuación (2-18) con el término en  $S$  igual a cero, es decir:

$$H(s) = \frac{Vmed(s)}{\theta o(s)} = \frac{kp*N}{kI}$$

Y si consideramos la ecuación (2-17), obtenemos la siguiente función de transferencia de lazo abierto  $GH(S)$  de la forma :

$$GH(S) = \frac{gan*KA*b}{S(S+1623)(S+11.5)} = \frac{K}{S(S+1623)(S+11.5)} \quad (2-27)$$

Donde el término  $b = \frac{kp*N}{KI}$ , es función de la relación de reducción de velocidad y del valor de la

constante  $K$ , los mismos que pueden ser cambiados en el programa de simulación. Además  $gan$  viene dado por la ecuación (2-16), y representa la ganancia equivalente del punto de suma resistivo.

Primeramente vamos a analizar el diagrama del Lugar Geométrico de la Raíces mostrado en la Figura 2.13., proveniente de la función de transferencia en lazo abierto ecuación (2-27), para poder de esta manera determinar el rango de estabilidad del sistema, en función de la variación de la ganancia en lazo abierto  $K$ .

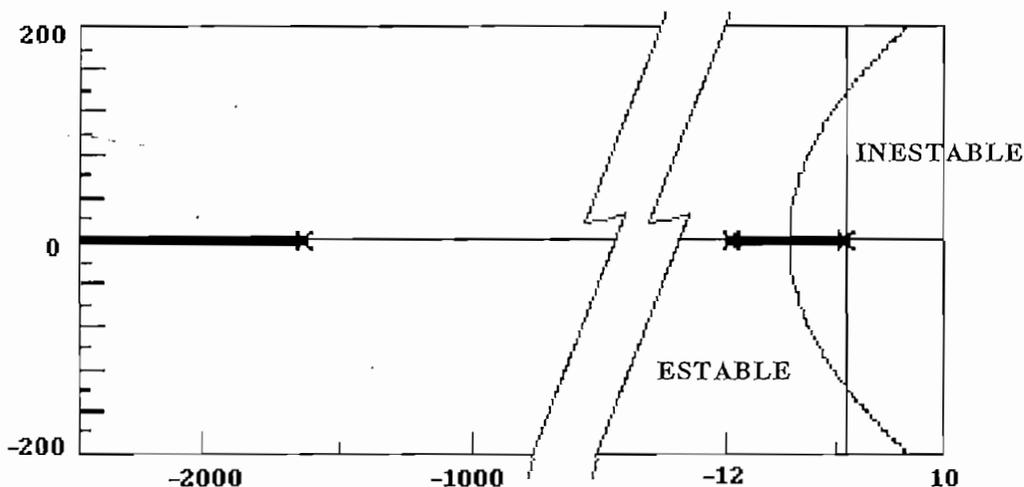


Fig. 2.13. Lugar Geométrico de las raíces del sistema sin compensación

Para determinar este rango de estabilidad vamos a utilizar el criterio de Routh Hurwitz, para el polinomio de lazo cerrado  $P(S)=1+GH(S)$ , el cual tiene la información necesaria acerca de la estabilidad del sistema, así tenemos la ecuación (28).

$$P(s) = S^3 + 1634.5S^2 + 18664.5S + K = 0 \tag{2-28}$$

$S^3$	1	18664.5
$S^2$	1634.5	K
$S^1$	$\frac{30507125.25-K}{K}$	1634.5
$S^0$	K	0

FIG. 2.14. Criterio de estabilidad de ROUTH HURWITZ

Si consideramos como una condición suficiente para afirmar que un sistema es estable, el que no debe existir ningún cambio de signo en la primera columna de la figura 2.14., es decir:

$$\forall K > 0 \quad y \quad \forall \frac{30507125.25-K}{1634.5} > 0$$

entonces tenemos que el sistema es estable para valores de  $K$ , comprendidos entre el siguiente rango de valores:

$$0 < K < (30507125.25 = gan*KA*b*Kp)$$

ahora definiendo a  $Kp$  como una ganancia proporcional, obtenemos el valor máximo que podríamos poner para que el sistema sea estable, es decir:

$$0 < Kp < \left( \frac{30507125.25}{gan*KA*b} \right)$$

$$\text{que para el presente caso : } gan*KA*b = 461262 \Rightarrow 0 < Kp < 66.14 \quad (2-29)$$

Nosotros también podríamos determinar el valor máximo de ganancia para el cual el sistema es sobre amortiguado, es decir que el sistema tiene solamente raíces reales, y esto lo hacemos a partir del polinomio característico  $P(S)$  de lazo cerrado, de tal forma que:

$$P(s) = 1 + GH(S) = 1 + K \frac{N(s)}{D(S)} = 0$$

$$\frac{\partial K}{\partial s} = \frac{\partial \left( -\frac{N(S)}{D(S)} \right)}{\partial s} = 0 \quad (2-30)$$

Y tomando en cuenta la ecuación (2-28), tenemos que:

$$K = -(S^3 + 1634.5S^2 + 18664.5S) \quad (2-31)$$

donde determinamos:

$$\frac{\partial K}{\partial s} = 3S^2 + 3269S + 18664.5 = 0$$

obtenemos las raíces:

$$S1 = -1083.92$$

$$S2 = -5.73977$$

pero como en la raíz  $S1$  no existe L.G.R., la raíz  $S2$  es la que nos van a servir para determinar el

máximo valor de ganancia tal que el sistema sea sobre amortiguado, entonces si reemplazamos S 2 en la ecuación (2-31) tenemos que:

$$K = 53470.4971 = \text{gan} * KA * b * Kp \quad (2-32)$$

$$Kp = \frac{53470.4971}{\text{gan} * KA * b} = \frac{53470.4971}{461262} = 0.1159$$

## 2.6 COMPENSACION DEL SISTEMA.

Sea la función de transferencia en lazo cerrado definida por la siguiente ecuación:

$$\theta_o(S) = \frac{G(S)}{1+GH(S)} * Vi(S)$$

$$\theta_o(S) = \frac{\text{gan} * KA}{S(S+1623)(S+11.5) + \text{gan} * KA * b} * Vi(S) \quad (2-33)$$

Ahora para analizar el comportamiento del MOTOMATIC, nosotros consideramos ciertas condiciones del sistema, como son:

$N = 1/9$	$R0 = 30K\Omega$	$K1 = (R1/R0) = 1/3$
$Kg = 0.148 * \pi / 180 \text{ Volt} * \text{Seg} / \text{Grad}$	$K2 = (R2/R0) = 1/3$	$K3 = (R3/R0) = 1/3$
$Kp = 6.74 * \pi / 180 \text{ Volt} / \text{Grad}$	$K4 = (R4/R0) = 1/3$	

cuyos parámetros K1, K2., K3 y K4 son los que vienen por definición en el equipo original, y sobre los cuales se realizan las prácticas de laboratorio. Además el valor de gan es el valor que introduce como ya hemos mencionamos el punto de suma resistivo, y cuyo valor lo calculamos mediante la utilización de la ecuación (2-16), así tenemos:

$$\text{gan} = 0.1$$

Reemplazando estos valores en la ecuación (2-33) tenemos:

$$\theta_o(S) = \frac{11763367.84}{S^3 + 1634.5S^2 + 18664.5S + 461262} * Vi(S) \quad (2-34)$$

Considerando la función de transferencia representada en la ecuación (2-34), podemos calcular los

polos de lazo cerrado, a partir del polinomio característico  $P(S)$ , como se muestra a continuación:

$$P(S) = S^3 + 1634.5S^2 + 18664.5S + 461262 = 0$$

$$S1 = -5.66 + j 15.88$$

Polos de Lazo Cerrado:  $S2 = -5.66 - j 15.88$

$$S3 = -1623.18$$

Aquí observamos la dominancia de los polos de lazo cerrado  $S1$  y  $S2$ , y por lo tanto se puede aproximar la respuesta del sistema a una de segundo orden, para de esta manera poder predecir las especificaciones de respuesta transitoria del sistema MOTOMATIC ante una entrada paso, las mismas que vienen definidas por:

$$Mp\% = 100 * e^{-\frac{\xi \pi}{\sqrt{1-\xi^2}}}$$

$$ts = \frac{4}{\xi \omega_n} \quad \text{criterio del 2\%}$$

si consideramos el polo  $S1 = -5.66 + j 15.88 = -\xi \omega_n \pm j \omega_n \sqrt{1-\xi^2}$

mediante los cálculos respectivos obtenemos las especificaciones aproximadas del sistema, las cuales son:  $ts \approx .7$  y  $Mp\% \approx 32.6$ , las mismas que se asemejan a los de la respuesta exacta en el tiempo mostrado en la Fig. 2.15, obtenido mediante el programa CC, donde observamos que el sistema se aproxima efectivamente a uno de segundo orden.

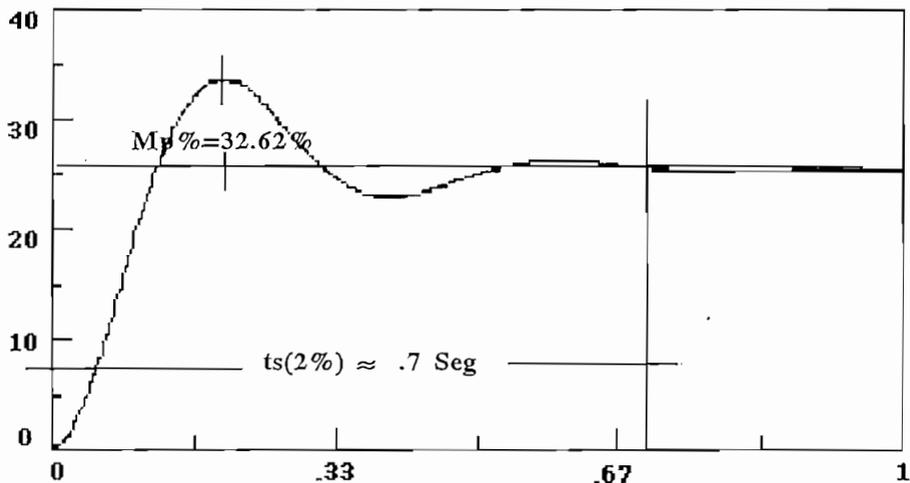


FIG. 2.15. Respuesta del sistema sin compensar.

### 2.6.1. REALIMENTACION DE VELOCIDAD.

Como ya hemos dicho, nosotros vamos a considerar como una de las técnicas a emplearse para la compensación del MOTOMATIC, a la realimentación de velocidad como una manera de mejorar el comportamiento dinámico del sistema.

Si consideramos que en la ecuación (2-18), el término en S representa a la realimentación de velocidad, tenemos:

$$H(s) = \frac{V_{med}(s)}{\theta_o(s)} = \frac{kp*N}{kl} + \frac{kg}{k2}S = b + aS$$

Ahora la función de transferencia en lazo cerrado viene definida como:

$$\frac{\theta_o(s)}{Vi(s)} = \frac{G(S)}{1+GH(S)}$$

resulta que:

$$\theta_o(s) = \frac{KA*gan}{S^3 + 1634.5S^2 + (18664.5 + a*KA*gan)S + (KA*gan*b)} * Vi(s) \quad (2-35)$$

ecuación semejante a la (2-33), en la cual aumenta el factor  $a*KA*gan$ , el mismo que se debe a la realimentación de velocidad. Si reemplazamos valores a utilizarse en la simulación tenemos la siguiente función de transferencia:

$$\theta_o(s) = \frac{11763367.84}{S^3 + 1634.5S^2 + 106772.1S + 461262} * Vi(s) \quad (2-36)$$

Se puede notar claramente la influencia de la realimentación de velocidad, en el lugar geométrico de las raíces mostrado en la FIG. 2.16, donde vemos que dicha realimentación aumenta un cero en la función de transferencia en lazo abierto, el cual no tiene en lazo cerrado, y es precisamente el que determina el cambio del lugar geométrico de las raíces.

Para graficar el Lugar Geométrico de las Raíces, primeramente tomamos en cuenta la función de transferencia  $GH(S)$  en lazo abierto ecuación (2-37), cuya forma es semejante a la ecuación (2-27) más el término debido a la realimentación de velocidad.

$$GH(S) = \frac{gan*KA*(aS+b)}{S(S+1623)(S+11.5)} = \frac{88107.6S+461262}{S(S+1623)(S+11.5)} \quad (2-37)$$

(Nota: A la realimentación de velocidad, también se la conoce como realimentación Taquimétrica, o acción de control Proporcional derivativa en el lazo de realimentación)

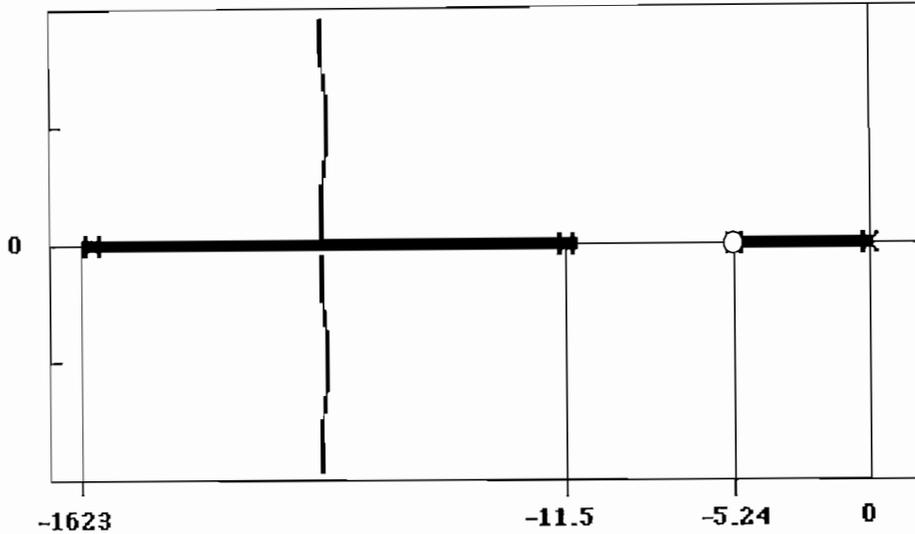


FIG. 2.16 Lugar Geométrico de las Raíces del sistema con realimentación de velocidad

En el gráfico 2.16 podemos notar que la realimentación de velocidad, provoca el desplazamiento del Lugar Geométrico de las Raíces hacia el lado izquierdo, de tal manera que la ubicación de dicho cero hace que el sistema sea absolutamente estable respecto a los valores de ganancia posibles, ya que no contiene ninguna raíz que vaya hacia el semiplano derecho.

Del análisis de la ecuación (2-36), tenemos que el sistema para las condiciones planteadas tiene el siguiente polinomio característico  $P(S)$  que determina los polos de lazo cerrado del sistema con realimentación de velocidad, así tenemos:

$$P(S) = S^3 + 1634.5S^2 + 106772.1S + 461262 = 0 \quad (2-38)$$

	S1=-	4.65
Polos de Lazo Cerrado	S2=-	63.32
	S3=-	-1566.53

Claramente podemos determinar que existe una predominancia del polo S 1, con respecto al resto de polos, por lo tanto podemos aproximar el sistema a uno de primer orden, ya que estos se atenúan muy rápidamente; de esta manera determinamos las especificaciones de respuesta en el tiempo aproximadas:

$$S1 = -4.65 = \frac{1}{T}$$

con el criterio del 2% de error tenemos que  $t_s \approx 4T \approx .861$

Valor que se asemeja al obtenido en la respuesta exacta en el tiempo de la FIG. 2.17, la misma que efectivamente se aproxima mucho a un sistema de primer orden.

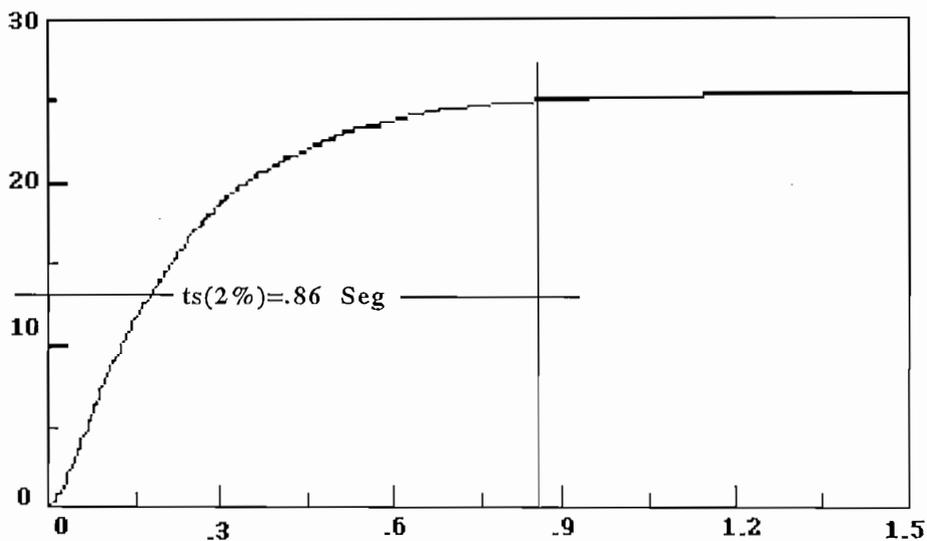


FIG. 2.17. Respuesta en el tiempo del sistema con realimentación de velocidad

### 2.6.2. UBICACION ARBITRARIA DEL CERO DE UNA RED.

Nosotros realizamos el estudio del método de compensación mediante la ubicación arbitraria del cero de una red, de manera que cuando se utilice el programa, simplemente se introduce los valores tanto de sobreimpulso, como de tiempo de establecimiento, con el fin de obtener la respuesta deseada.

Como ejemplo consideremos los siguientes parámetros de diseño:

$$Mp\% = 10$$

$$t_s = 0.5 \text{ seg}$$

de donde el  $M_p\%$  viene dado por la relación:

$$M_p\% = 100 * e^{-\frac{\xi \pi}{\sqrt{1-\xi^2}}} \quad (2-39)$$

despejamos del valor del coeficiente de atenuación  $\xi$ , así tenemos:

$$\xi = \sqrt{\frac{\ln(M_p)^2}{\pi^2 + \ln(M_p)^2}} \Rightarrow \xi = 0.591 \quad (2-40)$$

ahora consideramos el tiempo de establecimiento con un criterio del 2% para el error, así obtenemos:

$$ts = \frac{4}{\xi \omega_n} \Rightarrow \omega_n = \frac{4}{\xi ts} \Rightarrow \omega_n = 13.54 \quad (2-41)$$

Donde finalmente obtenemos el valor del punto de diseño, el cual nos va a permitir determinar el tipo de red de compensación que van a satisfacer nuestros criterio de diseño.

$$P_D = -\xi \omega_n \pm \omega_n \sqrt{1 - \xi^2} \quad (2-42)$$

$$P_D = -8 \pm j10.92$$

Ahora reemplazamos este valor en la función  $GH(S)$  del sistema sin compensar, nosotros determinamos la fase que debe introducir el compensador, tal que se cumpla la condición de fase:

$$\text{FASE}(GH(S))_{PD} + \text{FASE}(G_c(S))_{PD} = \pm 180^\circ \quad (2-43)$$

si obtenemos la fase del sistema en lazo abierto  $GH(S)$  sin compensar, tenemos:

$$\left| \text{GH}(S) \right|_{PD} = \text{FASE}(GH(S)) = 161.16^\circ$$

por lo tanto, vemos que el compensador debe introducir un adelanto de fase de  $18.84^\circ$ . Ahora si consideramos como un criterio de diseño, el que el cero de la red de compensación se coloque bajo el polo deseado, es decir:  $Z_1=8$

$$\left| G_c(S) \right|_{PD} = 18.84 = \left| \frac{Z^{90^\circ}}{P} \right|$$

$$\longrightarrow \phi_P = 71.16$$

donde obtenemos  $P_1=11.73$  , y si considerando la condición de módulo para que el punto de diseño pertenezca al Lugar Geométrico de las raíces, tenemos el siguiente compensador  $G_c(s)$ .

$$G_c(S) = Kc \left( \frac{S + Z_1}{S + P_1} \right) = 0.57 \left( \frac{S + 8}{S + 11.73} \right) \quad (2-44)$$

Los efectos de la red de compensación lo podemos observar en el lugar geométrico de las raíces, el cual lo forzamos para que pase por los polos deseados de diseño, como lo muestra la FIG. 2.18, y cuya función de transferencia de lazo cerrado se muestra a continuación:

$$\theta_o(S) = \frac{G_c G(S)}{1 + G_c G H(S)} * V_i(S)$$

$$\theta_o(S) = \frac{Kc Ka Gan(S + Z_1)}{S(S + 1623)(S + 11.5)(S + P_1) + Kc Ka Gan(S + Z_1)b} * V_i(S)$$

$$\theta_o(S) = \frac{6740409.77(S + 8)}{S^4 + 1655.23S^3 + 37837.2S^2 + 483239.5S + 2114439.6} * V_i(S) \quad (2-45)$$

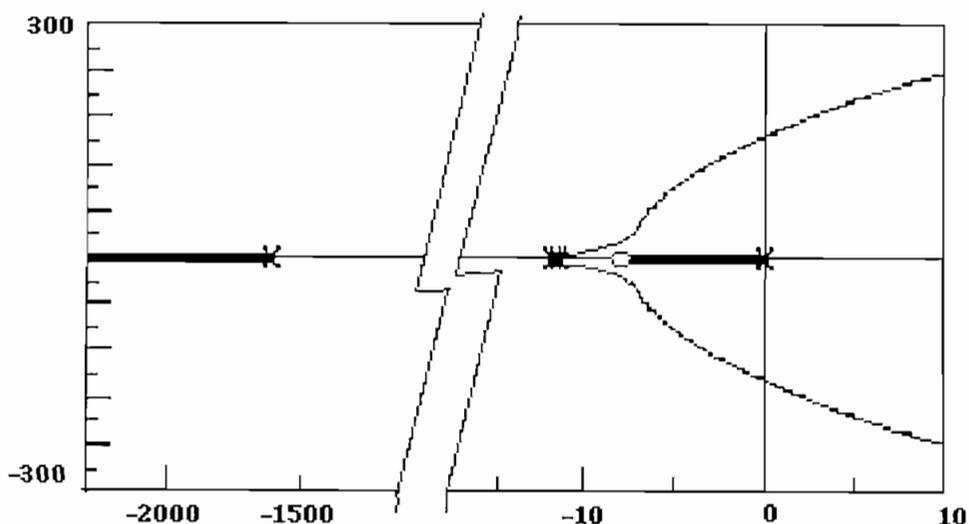


FIG. 2.18 Lugar Geométrico de las Raíces del sistema con Redes de Compensación

Como podemos observar esta red compensadora obliga a que el L.G.R. pase por el punto de diseño, pero esto no determina una dominancia de los mismos, lo cual es observable si determinamos los polos de lazo cerrado:

$$\begin{aligned} S_1 &= -7.10 \\ \text{Polos de Lazo Cerrado } S_{2,3} &= -7.95 \pm j10.91 \\ S_4 &= -1632.23 \end{aligned}$$

aquí podemos observar que el polo  $S_1$  tiende a atenuar más rápidamente la respuesta transitoria del sistema, sin embargo vemos que la respuesta exacta en el tiempo obtenida es mejor que la que nos impusimos como condiciones de diseño, como lo podemos observar en la respuesta exacta en el tiempo de la FIG. 2.19, debiéndose esto precisamente a la presencia del polo  $S_1$ , el cual es comparable con los polos de diseño.

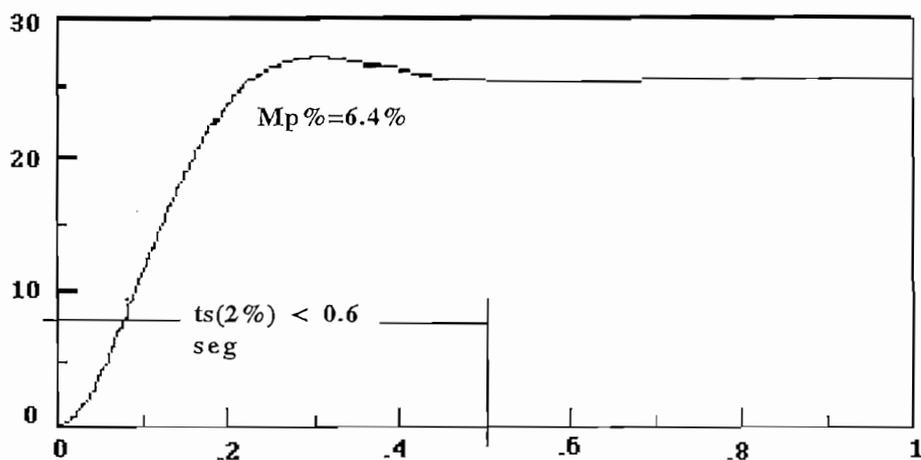


FIG. 2.19 Respuesta en el tiempo del sistema mediante la utilización de Redes de Compensación

(Un método semejante lo empleamos en el programa desarrollado)

### 2.6.3. COMPENSACION POR ACCIONES DE CONTROL.

El problema de diseño de un compensador P.I.D no será abordado debido a la gran variedad de métodos que nosotros podríamos utilizar para obtenerlo, sin embargo este programa está desarrollado para

que el usuario pueda ingresar los valores de  $K_p$ ,  $K_i$ , y  $K_d$ , de tal manera que se podrá determinar la conveniencia o no de determinadas acciones de control, ya sea mediante la visualización de la respuesta dinámica del sistema o también mediante el análisis matemático previo de la función de transferencia equivalente del sistema.

Sea la función de transferencia compensada en lazo cerrado definida por la ecuación (2-46), tenemos:

$$\theta(S) = \frac{G_c G(S)}{1 + G_c G H(S)} * Vi(S)$$

$$\theta(S) = \frac{KA \times gan \times Kp(K_D S^2 + S + K_I)}{S^2(S + 1623)(S + 11.5) + (b \times KA \times gan \times Kp(K_D S^2 + S + K_I))} * Vi(S) \quad (2-46)$$

Como un ejemplo y para fines de prueba del programa, aplicamos la compensación P.I.D. utilizando los siguientes parámetros:

<b><math>K_p=20</math></b>	<b><math>K_i=30</math></b>	<b><math>K_d=0.01</math></b>
----------------------------	----------------------------	------------------------------

Ahora reemplazamos todos los valores en la ecuación (2-46), obteniendo la siguiente matriz de transferencia en lazo cerrado:

$$G_{13}(s) = \frac{2.352674E+08(.01s^2 + s + 30)}{s^4 + 1634.5s^3 + 110912.8s^2 + 9224833s + 2.76745E+08}$$

la cual tiene los siguientes polos de lazo cerrado:

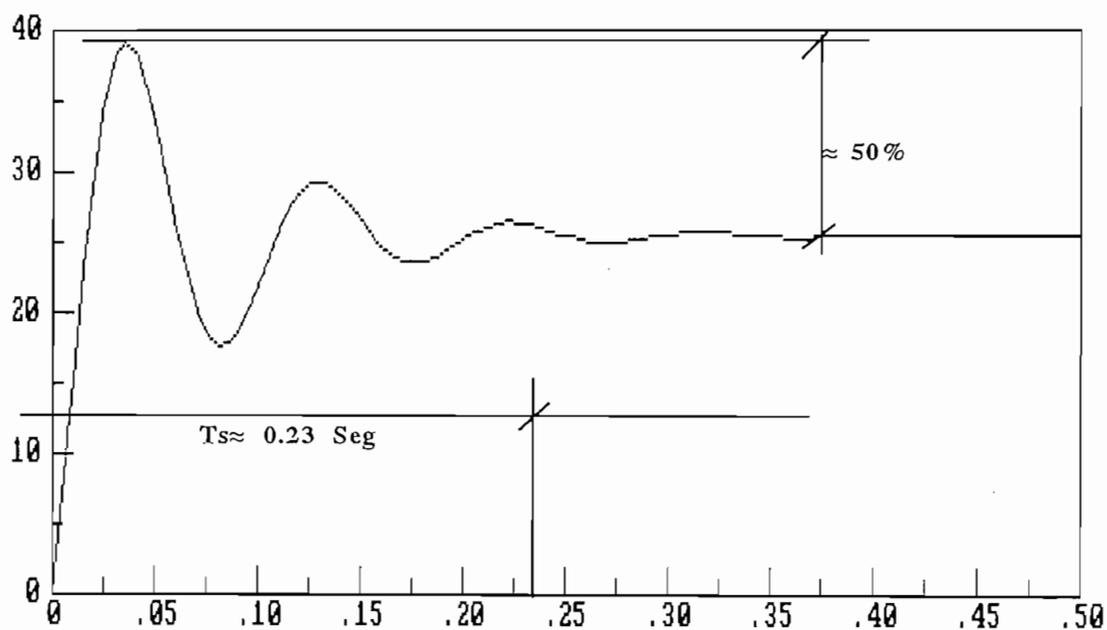
$$\begin{array}{ll} P1 : & -37.83733747536927 \\ P2 : & -14.62052648659633 \\ P3 : & -14.62052648659633 \\ P4 : & -1567.421609551438 \end{array} \quad \begin{array}{l} +j \ 66.72741485152975 \\ -j \ 66.72741485152975 \end{array}$$

aquí podemos observar que el P4 se atenúa más rápidamente, lo que hace que exista una predominancia de

P2,3, y el sistema podemos aproximarlo con uno de segundo orden, pese a que no existe mucha dominancia con respecto a P1.

Entonces si realizamos esta aproximación, el sistema tiene unas especificaciones de respuesta transitoria de máximo sobreimpulso  $\approx 50,24\%$  y un tiempo de establecimiento de  $\approx 0.2735$ , valores que se asemejan a los obtenidos en la figura 2.20.

Esta respuesta será apropiada dependiendo de la utilización del servomecanismo de posición, ya que como vemos el sistema tiene un sobreimpulso demasiado grande.



*Fig.2.20. Respuesta del Motomatic ante un compensador*

La facilidad que presta el programa desarrollado para la simulación del Motomatic está en que nosotros podemos ir variando los parámetros del compensador P.I.D., e ir observando lo que ocurre, de tal manera ir determinando la factibilidad o no de dicha acción de control.

*CAPITULO III*

*SISTEMA DE NIVEL DE LIQUIDOS EN TANQUES  
ACOPLADOS*

### 3.1. MODELACION DEL SISTEMA DE TANQUES ACOPLADOS

El sistema experimental en el que nos basamos para la simulación consiste de dos tanques de almacenamiento acoplados por un grupo de orificios seleccionables, y dependiendo del orificio, existirá mayor o menor influencia del fluido que atraviesa los tanques con relación al flujo de salida de los mismos, la entrada de agua es suministrada por dos bombas cuyo caudal es variable.

El problema de control aquí es básicamente el tratar de mantener estable el nivel de fluido de salida de los tanques, mediante la variación del caudal de entrada a los mismos; y como nosotros demostraremos, este flujo de salida es proporcional al nivel de los líquidos de los tanques, por lo tanto el problema se reduce a tratar de mantener o controlar el nivel de cada uno de los tanques acoplados. Dicho nivel es sentido mediante la utilización de sensores de nivel de tipo capacitivo, y cuya señal nos sirve para alimentar al respectivo controlador, y posteriormente al actuador.

A continuación en la Fig. 3.1., se muestra un diagrama esquemático del Sistema de nivel de Líquidos de Tanques Acoplados.

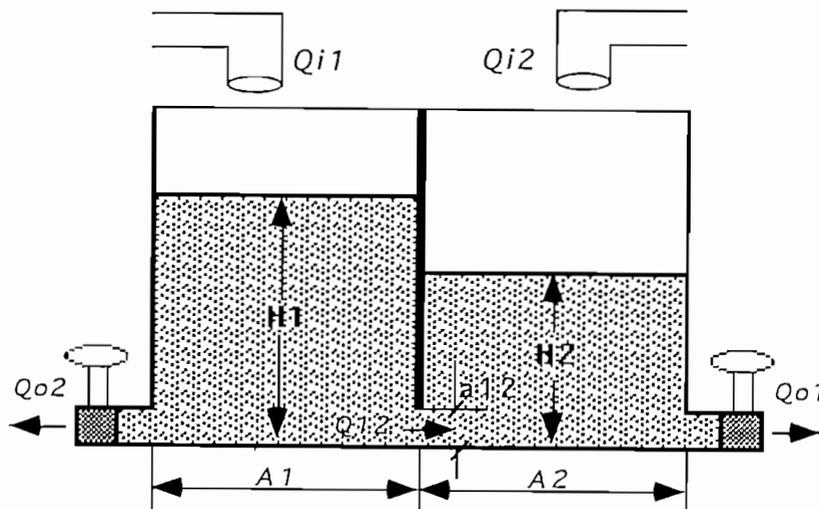


Fig. 3.1. Sistema del nivel de Líquidos en Tanques Acoplados.

donde:

$H_{1,2}$  : Alturas del fluidos de los tanques 1 y 2 respectivamente.

$A_{1,2}$  : Area transversal de los tanques.

$a_{1,2}$  : Area transversal de las válvulas de desagüe.

$Q_{i1,2}$  : Caudales de entrada de los tanques respectivos.

$Q_{o1,2}$  : Caudales de salida de los tanques respectivos.

$Cd_{1,2}$  : Coeficientes de descarga de los orificios respectivos.

$Cd_{12}$  : Coeficiente de descarga inter-tanques.

Y el valor de las constantes del sistema son:

$$A_1 = 37.83 * 10^{-3} \text{ m}^2$$

$$A_2 = 38.42 * 10^{-3} \text{ m}^2$$

$$a_1 = 1.26 * 10^{-3} \text{ m}^2$$

$$a_2 = 1.26 * 10^{-3} \text{ m}^2$$

Como habíamos mencionado, los orificios de interconexión de los tanques tienen diferentes áreas, las cuales se muestran en la Fig. 3.2 .

#TAPON	Diam (mm.)	$a_{12}$ (cm <sup>2</sup> )
0	4.7	0.173
1	9.7	0.738
2	12.2	1.17
3	14.7	1.78
4	19.3	2.93
5	24.3	4.64

Figura 3.2. Areas de los orificios Inter-tanques

Tomando la Fig. 3.1., y realizando un balance de flujos en cada tanque tenemos:

$$Q_{i1} = Q_{c1} + Q_{12} + Q_{o1} \quad (3-1)$$

$$Q_{i2} = Q_{c2} - Q_{12} + Q_{o2} \quad (3-2)$$

Si consideramos que los caudales de almacenamiento en los tanques respectivos, son iguales a:

$$Q_{c1} = \frac{dV1}{dt} = A1 * \frac{dH1}{dt} = A1 * \dot{H}1 \quad (3-3)$$

$$Q_{c2} = \frac{dV2}{dt} = A2 * \frac{dH2}{dt} = A2 * \dot{H}2 \quad (3-4)$$

y donde los caudales de salida vienen dados por:

$$Q_{o1} = Cd1 * a1 \sqrt{2gH1} \quad (3-5)$$

$$Q_{o2} = Cd2 * a2 \sqrt{2gH2} \quad (3-6)$$

$$Q_{12} = Cd12 * a12 \sqrt{2g(H1 - H2)} \quad (3-7)$$

VALVULA 1 $Cd_1$	VALVULA 2 $Cd_2$	Irificios Intertanques $Cd_{12}$
0.395	0.395	TAPON 0 = 0.398
		TAPON 1 = 0.341
		TAPON 2 = 0.333
		TAPON 3 = 0.298
		TAPON 4 = 0.299
		TAPON 5 = 0.317

Fig. 3.3. Coeficientes de descarga para los Tanques Acoplados.

Si reemplazamos las ecuaciones (3-3,4,5,6,7) en (3-1 y 2) tenemos:

$$\dot{H}1 * A1 = \left[ -Cd1 * a1 \sqrt{2gH1} - cd12 * a12 \sqrt{2g(H1 - H2)} + Qi1 \right] \quad (3-8)$$

$$\dot{H}2 * A2 = \left[ Cd12 * a12 \sqrt{2g(H1 - H2)} - cd2 * a2 \sqrt{2gH2} + Qi2 \right] \quad (3-9)$$

### 3.1.1. MODELACION DE LAS BOMBAS CENTRIFUGAS

Para la obtención de las características de las bombas centrífugas, se procedió a medir el caudal de salida que podía suministrar la bomba respectiva al subir el voltaje de alimentación en pasos de 0.5 V ,

de tal manera de tratar de determinar la curva característica de cada una de la bombas , obteniéndose los siguientes resultados.

VOLTAJE DC	CAUDAL Qi1 $10^{-4} \text{ m}^3/\text{s}$	CAUDAL Qi2 $10^{-4} \text{ m}^3/\text{s}$
0.0 - 3.5	0.000	0.000
4.0	0.145	0.189
4.5	0.242	0.300
5.0	0.350	0.472
5.5	0.461	0.541
6.0	0.584	0.651
6.5	0.615	0.725
7.0	0.700	0.769
7.5	0.780	0.894
8.0	0.823	0.985
8.5	0.859	0.985
9.0	0.900	1.220
9.5	0.945	1.320
10.0	0.995	1.390
10.5	1.050	1.450
11.0	1.110	1.470
11.5 - 12.0	1.114	1.470

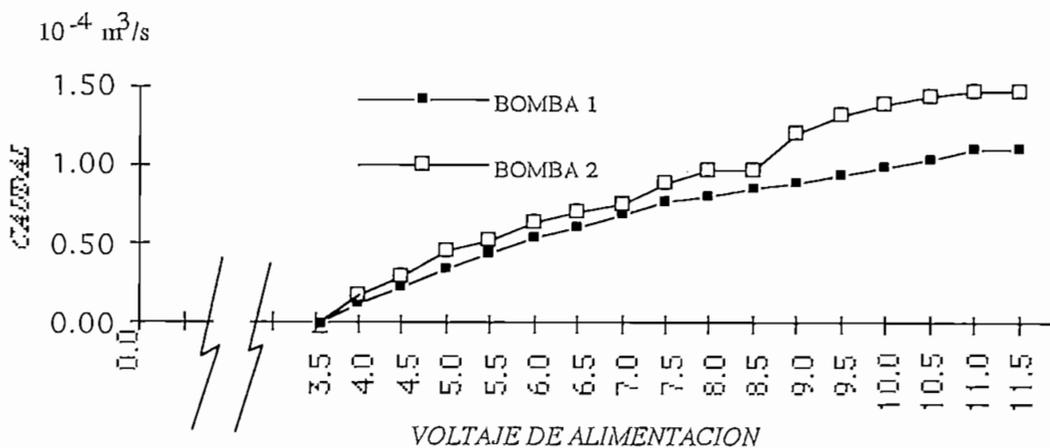


FIG. 3.4. Comportamiento de las bombas.

De el gráfico de la Figura 3.4. podemos darnos cuenta que las bombas tienen una respuesta no lineal , y una zona muerta que afectan aún más a la no linealidad de la planta para el respectivo análisis matemático; por lo tanto como primera aproximación de las curvas realizaremos una regresión lineal para determinar la ecuación de la recta que se aproxime a la curva real.

$Q_i = b_i V_i + a_i$ (cm <sup>3</sup> /Seg)	Bomba1	Bomba2
Pendiente ( $b_i$ )	13.40	18.51
Intersección ( $a_i$ )	-32.10	-51.64
Coeficiente de correlación	0.9796	0.9872

Obteniendo la siguiente representación para las bombas:

$$\begin{bmatrix} Q_{i1} \\ Q_{i2} \end{bmatrix} = \begin{bmatrix} b1 & 0 \\ 0 & b2 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} + \begin{bmatrix} a1 \\ a2 \end{bmatrix} \quad \text{cm}^3 / \text{seg} \quad (3-10)$$

$\forall 3.5 < U_i < 11.5 V$

### 3.1.2. MODELACION DE LOS TRANSDUCTORES CAPACITIVOS.

Los transductores capacitivos son tubos perpendiculares colocados en el interior del fluido, estos tubos mantienen una capacitancia la cual varía con el nivel del líquido en su interior, estos cambios son detectados y proveen una señal eléctrica la cual es proporcional al nivel del líquido.

$$\begin{array}{l} \text{TRANSDUCTOR 1} \\ d_{11} = \frac{y_1}{H_1} = \frac{5.0(V)}{20(\text{cm})} = 0.25 \left( \frac{V}{\text{cm}} \right) \end{array} \quad \begin{array}{l} \text{TRANSDUCTOR 2} \\ d_{22} = \frac{y_2}{H_2} = \frac{3.75(V)}{15(\text{cm})} = 0.25 \left( \frac{V}{\text{cm}} \right) \end{array}$$

De tal manera que obtenemos la siguiente matriz  $\mathbf{d}$

$$\begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = \begin{bmatrix} d_{11} & 0 \\ 0 & d_{22} \end{bmatrix} \begin{bmatrix} H_1 \\ H_2 \end{bmatrix} \quad (3-11)$$

La obtención de los parámetros para el sistema de tanques acoplados que se dispone en la Facultad, fueron realizados y tabulados en la Tesis de Juan Garzón. (REF.1), por lo cual en el presente trabajo solo se mostrarán los resultados.

### 3.2. DESCRIPCION FISICA DEL SISTEMA EN VARIABLES DE ESTADO

Como podemos darnos cuenta las ecuaciones (3-8) y (3-9) corresponden a un sistema no lineal, sin

embargo para poder realizar el estudio del comportamiento del nivel de líquido en los Tanques Acoplados, vamos a linealizarlo considerando pequeñas variaciones respecto a un punto de operación  $\bar{H}_1, \bar{H}_2, \bar{Q}_{i1}, \bar{Q}_{i2}$ , donde se cumple que:

$$H_1 = \bar{H}_1 + h_1 \quad H_2 = \bar{H}_2 + h_2 \quad (3-12)$$

$$Q_{i1} = \bar{Q}_{i1} + q_{i1} \quad Q_{i2} = \bar{Q}_{i2} + q_{i2} \quad (3-13)$$

de manera que obtenemos las siguientes expresiones para el sistema linealizado:

$$\dot{h}_1 = K_{11} * h_1 + K_{12} * h_2 + K_1 * q_{i1} \quad (3-14)$$

$$\dot{h}_2 = K_{21} * h_1 + K_{22} * h_2 + K_2 * q_{i2} \quad (3-15)$$

donde se cumple que;

$$K_{11} = \frac{\partial \dot{H}_1}{\partial H_1} \quad ; \quad K_{12} = \frac{\partial \dot{H}_1}{\partial H_2} \quad ; \quad K_1 = \frac{\partial \dot{H}_1}{\partial Q_{i1}} \quad \Big|_{\bar{H}_1, \bar{H}_2, \bar{Q}_{i1}, \bar{Q}_{i2}}$$

$$K_{21} = \frac{\partial \dot{H}_2}{\partial H_1} \quad ; \quad K_{22} = \frac{\partial \dot{H}_2}{\partial H_2} \quad ; \quad K_2 = \frac{\partial \dot{H}_2}{\partial Q_{i2}} \quad \Big|_{\bar{H}_1, \bar{H}_2, \bar{Q}_{i1}, \bar{Q}_{i2}}$$

y si desarrollamos estas expresiones obtenemos el valor de las constantes, así tenemos:

$$K_{11} = - \left[ \frac{Cd_{12} * a_{12} * \sqrt{2g}}{2A1 * \sqrt{\bar{H}_1 - \bar{H}_2}} + \frac{Cd_1 * a_1 * \sqrt{2g}}{2A1 * \sqrt{\bar{H}_1}} \right] \quad (3-16)$$

$$K_{12} = \frac{Cd_{12} * a_{12} * \sqrt{2g}}{2A1 * \sqrt{\bar{H}_1 - \bar{H}_2}} \quad (3-17)$$

$$K_{21} = \frac{Cd_{12} * a_{12} * \sqrt{2g}}{2A2 * \sqrt{\bar{H}_1 - \bar{H}_2}} \quad (3-18)$$

$$K_{22} = - \left[ \frac{Cd_{12} * a_{12} * \sqrt{2g}}{2A2 * \sqrt{\bar{H}_1 - \bar{H}_1}} + \frac{Cd_2 * a_2 * \sqrt{2g}}{2A2 * \sqrt{\bar{H}_1}} \right] \quad (3-19)$$

$$K1 = \frac{I}{A1} \quad ; \quad K2 = \frac{I}{A2} \quad (3-20)$$

ahora si realizamos el cambio de variables:

$$\begin{aligned} x_1 = h_1 & \rightarrow \dot{x}_1 = \dot{h}_1 \\ x_2 = h_2 & \rightarrow \dot{x}_2 = \dot{h}_2 \end{aligned}$$

obtenemos la representación de los Tanques Acoplados de la forma general de la ecuación de estado, así tenemos:

$$\dot{\mathbf{x}}_{2 \times 1} = \mathbf{A}_{2 \times 2} \mathbf{x}_{2 \times 1} + \mathbf{B}_{2 \times 2} \mathbf{q}^i_{2 \times 1} \quad (3-21)$$

$$\mathbf{h}_{2 \times 1} = \mathbf{C}_{2 \times 2} \mathbf{x}_{2 \times 1} \quad (3-22)$$

donde:

$$\mathbf{A} = \begin{bmatrix} -K11 & K12 \\ K21 & -K22 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} K1 & 0 \\ 0 & K2 \end{bmatrix} \quad (3-23)$$

$$\mathbf{C} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (3-24)$$

Como vimos al modelar las bombas, las cuales están definidas por la ecuación (3-10), también es necesario realizar una linealización de las mismas, considerando pequeñas variaciones respecto del mismo punto de operación  $\bar{Q}i_1, \bar{Q}i_2, \bar{V}_1, \bar{V}_2$ , definido previamente; así tenemos:

$$\begin{bmatrix} q^i_1 \\ q^i_2 \end{bmatrix} = \begin{bmatrix} b_{11} & 0 \\ 0 & b_{22} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \quad (3-25)$$

donde las constantes vienen dadas por:

$$b_{11} = \frac{\partial \dot{Q}i_1}{\partial V_1} = b_1 \quad b_{22} = \frac{\partial \dot{Q}i_2}{\partial V_2} = b_2 \quad \left| \bar{Q}i_1, \bar{Q}i_2, \bar{V}_1, \bar{V}_2 \right. \quad (3-26)$$

y donde se cumple además que:

$$V_1 = \bar{V}_1 + v_1 \quad (3-27)$$

$$V_2 = \bar{V}_2 + v_2 \quad (3-28)$$

llegando finalmente a obtener el siguiente diagrama de bloques equivalente en lazo abierto del sistema de nivel de líquidos en Tanques Acoplados linealizado.

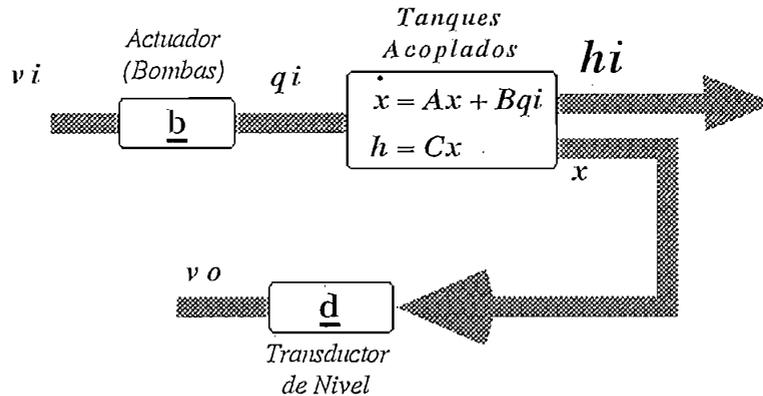


Fig. 3.5 Diagrama equivalente de los Tanques Acoplados

Para determinar el punto de operación, es necesario considerar las características que exigirá más tarde el control, en cuanto a voltaje de control de los manejadores de las bombas, así como también tener un rango considerable alrededor del punto de operación de forma que sea factible de variar y de esta manera poder observar un control más efectivo sobre el nivel del líquido en los Tanques.

### 3.3. CONTROLABILIDAD DEL SISTEMA

Aún cuando la mayor parte de los sistemas físicos son controlables y observables, puede no existir solución a un problema de control óptimo si el sistema considerado no es controlable, y esto se debe a que los modelos matemáticos correspondientes, pueden no poseer las propiedades de controlabilidad y observabilidad. Por lo tanto es necesario conocer bajo que condiciones un sistema es controlable y observable.

Sea el sistema definido por la ecuación de estado:

$$\dot{x} = Ax + Bu \quad (3-29)$$

$$h = Cx \quad (3-30)$$

Donde:

$X$  = Vector de estado (vector  $n$ -dimensional)

$U = \text{Señal de control.}$

$A = \text{Matriz } n \times n$

$B = \text{Matriz } n \times r$

Se dice que el sistema descrito anteriormente es controlable en el tiempo  $t=t_0$ , si es posible construir una señal de control no acotada que transfiera un estado inicial a cualquier estado final en un intervalo de tiempo finito  $t_0 \leq t \leq t_1$ . Si todo estado es controlable, se dice que el sistema es de estado controlable completo. Se puede suponer sin pérdida de generalidad, que el estado final es el origen del espacio de estado y que el tiempo inicial es cero, o sea  $t_0=0$ .

La solución de la ecuación (3-29) es:

$$x(t) = e^{At} x(0) + \int_0^t e^{A(t-\tau)} B u(\tau) d\tau \quad (3-31)$$

Aplicando la definición de controlabilidad completa de estado recién dada, se tiene

$$x(t_1) = 0 = e^{At_1} x(0) + \int_0^{t_1} e^{A(t_1-\tau)} B u(\tau) d\tau \quad (3-32)$$

0

$$x(0) = - \int_0^{t_1} e^{A\tau} B u(\tau) d\tau \quad (3-33)$$

Nótese que puede escribirse

$$e^{A\tau} = \sum_{k=0}^{n-1} \alpha_k(\tau) A^k \quad (3-34)$$

y reemplazando en (3-33) se tiene que

$$x(0) = - \sum_{k=0}^{n-1} A^k B \int_0^{t_1} \alpha_k(\tau) u(\tau) d(\tau) \quad (3-35)$$

si se hace

$$\int_0^{t_1} \alpha_k(\tau) u(\tau) d(\tau) = \beta_k$$

la ecuación (3-35) tiene la forma

$$x(0) = - \sum_{k=0}^{n-1} A^k B \beta_k \quad (3-36)$$

$U = \text{Señal de control.}$

$A = \text{Matriz } n \times n$

$B = \text{Matriz } n \times r$

Se dice que el sistema descrito anteriormente es controlable en el tiempo  $t=t_0$ , si es posible construir una señal de control no acotada que transfiera un estado inicial a cualquier estado final en un intervalo de tiempo finito  $t_0 \leq t \leq t_1$ . Si todo estado es controlable, se dice que el sistema es de estado controlable completo. Se puede suponer sin pérdida de generalidad, que el estado final es el origen del espacio de estado y que el tiempo inicial es cero, o sea  $t_0=0$ .

La solución de la ecuación (3-29) es:

$$x(t) = e^{At} x(0) + \int_0^t e^{A(t-\tau)} B u(\tau) d\tau \quad (3-31)$$

Aplicando la definición de controlabilidad completa de estado recién dada, se tiene

$$x(t_1) = 0 = e^{At_1} x(0) + \int_0^{t_1} e^{A(t_1-\tau)} B u(\tau) d\tau \quad (3-32)$$

0

$$x(0) = - \int_0^{t_1} e^{A\tau} B u(\tau) d\tau \quad (3-33)$$

Nótese que puede escribirse

$$e^{A\tau} = \sum_{k=0}^{n-1} \alpha_k(\tau) A^k \quad (3-34)$$

y reemplazando en (3-33) se tiene que

$$x(0) = - \sum_{k=0}^{n-1} A^k B \int_0^{t_1} \alpha_k(\tau) u(\tau) d(\tau) \quad (3-35)$$

si se hace

$$\int_0^{t_1} \alpha_k(\tau) u(\tau) d(\tau) = \beta_k$$

la ecuación (3-35) tiene la forma

$$x(0) = - \sum_{k=0}^{n-1} A^k B \beta_k \quad (3-36)$$

$$x(0) = \begin{bmatrix} B & AB & \vdots & \vdots & A^{n-1}B \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{n-1} \end{bmatrix} \quad (3-37)$$

Por lo tanto si el sistema es de estado completamente controlable, entonces dado cualquier estado inicial  $x(0)$ , se debe satisfacer la ecuación (3-29); esto requiere que el rango de la matriz de  $\mathbf{n} \times \mathbf{n}$

$$\begin{bmatrix} B & AB & \vdots & \vdots & A^{n-1}B \end{bmatrix} \quad (3-38)$$

, sea igual al orden del sistema.

De este análisis se puede establecer la condición de controlabilidad de estado completo como sigue: el sistema dado por la ecuación (3-29) es de estado completamente controlable si, y solamente si, los vectores  $B, AB, \dots, A^{n-1}B$  son linealmente independientes, o si la matriz de  $\mathbf{n} \times \mathbf{n}$  determinada por la ecuación (3-38) es de rango  $\mathbf{n}$ .

Se puede extender el resultado recién obtenido al caso en el cual el vector de control  $U$  es  $\Gamma$ -dimensional; así se puede probar que la condición de controlabilidad completa de estado es que la matriz siguiente  $\mathbf{n} \times \mathbf{n}\Gamma$

$$\begin{bmatrix} B & AB & \vdots & \vdots & A^{n-1}B \end{bmatrix}$$

sea de rango  $\mathbf{n}$ , o que contenga  $\mathbf{n}$  vectores columna linealmente independientes.

Para el caso de los tanques acoplados tenemos que el sistema es de  $2 \times 2$  definido por las matrices:

$$A = \begin{bmatrix} -K_{11} & K_{12} \\ K_{21} & -K_{22} \end{bmatrix} \quad B = \begin{bmatrix} K_1 & 0 \\ 0 & K_2 \end{bmatrix}$$

Por lo tanto la matriz de controlabilidad está definida por:  $M_c = [B \ ; \ AB]$ , y reemplazando los valores de las matrices A y B tenemos:

$$M_c = \begin{bmatrix} K_1 & 0 & \vdots & -K_1 K_{11} & K_2 K_{12} \\ 0 & K_2 & \vdots & K_1 K_{21} & -K_2 K_{22} \end{bmatrix}$$

De esta matriz de controlabilidad podemos darnos cuenta que el máximo arreglo cuadrado que podemos hacer, cuyo determinante es 2, es decir que el rango de la matriz de controlabilidad es igual al

orden del sistema; por lo tanto el sistema es completamente controlable, para el caso general de que  $K1$  y  $K2$  sean diferentes de cero.

### 3.4 . DISCRETIZACION DEL SISTEMA.

Como se desea calcular el estado  $\mathbf{x}(t)$  utilizando un computador digital, tenemos que convertir la ecuación de estado de tiempo continuo en una ecuación de estado de tiempo discreto. Suponemos que el vector de entrada  $\mathbf{u}(t)$  varía solamente en los instantes de muestreo equiespaciados es decir en  $t=kt$ , para  $k=0,1,2,\dots$ , o lo que es lo mismo  $\mathbf{u}(t)=\mathbf{u}(kT)$  para el  $k$ -ésimo período de muestreo.

Tomando la solución de la ecuación de estado continua del sistema descrito por  $\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}$ , tenemos:

$$\mathbf{x}(t) = e^{At}\mathbf{x}(0) + e^{At} \int_0^t e^{-A\tau} B\mathbf{u}(\tau) d\tau \quad (3-39)$$

para un intervalo de tiempo comprendido entre dos instantes de muestreo consecutivo, es decir  $kT < t \leq (k+1)T$ , donde  $T$  representa el intervalo de muestreo; la ecuación (3-39) puede ser representada como:

$$\mathbf{x}(k+1) = e^{A(k+1)T}\mathbf{x}(0) + e^{A(k+1)T} \int_0^{(k+1)T} e^{-A\tau} B\mathbf{u}(\tau) d\tau \quad (3-40)$$

$$\mathbf{x}(kT) = e^{AkT}\mathbf{x}(0) + e^{AkT} \int_0^{kT} e^{-A\tau} B\mathbf{u}(\tau) d\tau \quad (3-41)$$

multiplicando la ecuación (3-41) por  $e^{AT}$  en y restando de la ecuación (3-40) tenemos:

$$\begin{aligned} \mathbf{x}((k+1)T) &= e^{AT}\mathbf{x}(kT) + e^{A(k+1)T} \int_{kT}^{(k+1)T} e^{-A\tau} B\mathbf{u}(\tau) d\tau \\ &= e^{AT}\mathbf{x}(kT) + e^{AT} \int_0^T e^{-At} B\mathbf{u}(kt) dt \\ &= e^{AT}\mathbf{x}(kT) + e^{AT} \int_0^T e^{-At} B\mathbf{u}(kt) dt \\ &= e^{AT}\mathbf{x}(kT) + \int_0^T e^{-A\lambda} d\lambda * B\mathbf{u}(kT) \text{ donde } \lambda = T - t, \end{aligned} \quad (3-42)$$

Ahora realizamos las siguientes definiciones:

$$A_d = e^{AT} \quad (3-43)$$

$$B_d = \int_0^T e^{At} dt * B \quad (3-44)$$

llegamos finalmente a la ecuación de estado discreta definida por:

$$x(k+1) = A_d x(k) + B_d u(k)$$

Si realizamos la descomponiendo en una serie de Taylor la ecuación (3-43) tenemos:

$$A_d = e^{AT} = I + AT + \frac{A^2 T^2}{2!} + \frac{A^3 T^3}{3!} + \frac{A^4 T^4}{4!} + \dots \quad (3-45)$$

por lo tanto la ecuación (3-45) puede ser escrita de la siguiente manera:

$$B_d = \int_0^T \left( I + At + \frac{A^2 t^2}{2!} + \frac{A^3 t^3}{3!} + \frac{A^4 t^4}{4!} + \dots \right) dt B$$

$$B_d = \left( IT + \frac{A T^2}{2!} + \frac{A^2 T^3}{3!} + \frac{A^3 T^4}{4!} + \dots \right) A^{-1} B \quad (3-46)$$

multiplicando y dividiendo la ecuación (3-46) por la matriz A , tenemos:

$$B_d = \left( AT + \frac{A^2 T^2}{2!} + \frac{A^3 T^3}{3!} + \frac{A^4 T^4}{4!} + \dots \right) A^{-1} B$$

finalmente si sumamos y restamos la ecuación anterior por la matriz Identidad tenemos:

$$B_d = \left( \left( I + AT + \frac{A^2 T^2}{2!} + \frac{A^3 T^3}{3!} + \frac{A^4 T^4}{4!} + \dots \right) - I \right) A^{-1} B \quad (3-47)$$

Ahora reemplazamos la ecuación (3-45) en la ecuación (3-47) obtenemos finalmente otra manera para discretizar la matriz B:

$$B_d = \left( e^{AT} - I \right) A^{-1} B \quad (3-48)$$

### 3.5. ANALISIS DEL SISTEMA SIN COMPENSAR.

Para empezar el estudio del comportamiento del nivel de líquido en los Tanques Acoplados, hagámoslo primeramente con el sistema en lazo abierto definido por la figura 3.5, para lo cual inicialmente realizamos el movimiento del bloque correspondiente al transductor de nivel al camino directo, esto con el fin de tener ahora una señal de referencia en centímetros, de esta manera de obtenemos la figura 3.6.

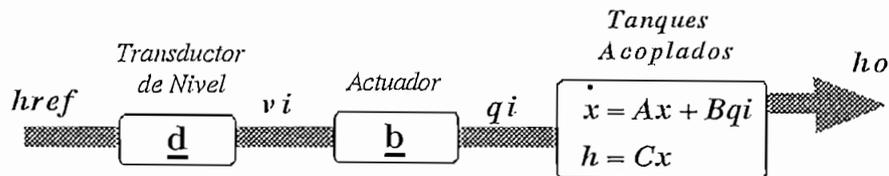


Fig. 3.6. Diagrama equivalente en lazo abierto

En la figura 3.6 se cumple la siguiente ecuación:

$$\begin{aligned}\dot{x} &= Ax + Bdbh_{ref} \\ h &= Cx\end{aligned}\tag{3-49}$$

en estado estable  $\dot{x} = 0$ , por lo que primeramente obtenemos los puntos de equilibrio para el nivel de los tanques, así tenemos:

$$\begin{aligned}x_0 &= -A^{-1}B_{eq}h_{ref} \quad \text{donde} \quad B_{eq} = Bdb \\ h &= Cx\end{aligned}\tag{3-50}$$

de la ecuación (3-50) primeramente podemos observar que la señal de salida no tiende a seguir a la referencia, ya que esta es función de  $-A^{-1}B_{eq}$ .

El análisis del comportamiento del nivel del líquido en los Tanques Acoplados, lo hicimos para el siguiente punto de operación, el mismo que fue obtenido en base de los caudales necesarios para obtener dichos niveles, y de los caudales que podrían ser suministrados por las bombas; así tenemos

$$\bar{H} = \begin{vmatrix} 1 & 3 \\ & 1 & 8 \end{vmatrix} \quad H_{fin} = \begin{vmatrix} 1 & 5 \\ & 2 & 0 \end{vmatrix}\tag{3-51}$$

donde se cumple que la referencia debería ser también la variación alrededor del punto de operación, por lo tanto tenemos:

$$h_{ref} = H_{fin} - \bar{H} = \begin{vmatrix} 2 \\ 2 \end{vmatrix} \quad (3-52)$$

mediante la utilización de las ecuaciones (3-16 a 20), obtenemos las siguientes matrices que representan al sistema lineal en variables de estado.

$$\begin{aligned} A &= \begin{vmatrix} -0.009878 & 0.001801 \\ 0.001801 & -.008666 \end{vmatrix} \\ B &= \begin{vmatrix} 2.643E-03 & 0 \\ 0 & 2.643E-03 \end{vmatrix} \\ C &= \begin{vmatrix} 1 & 0 \\ 0 & 1 \end{vmatrix} \end{aligned} \quad (3-53)$$

la determinación de la matriz **b** que representa a las bombas, lo hicimos en base a la ecuación (3-25), así tenemos:

$$b = \begin{vmatrix} 13.4 & 0 \\ 0 & 18.51 \end{vmatrix} \quad (3-54)$$

mientras que la matriz **d** que representa al transductor de nivel, está determinada por la ecuación (3-11).

Si reemplazamos todas estas matrices en la ecuación (3-50), obtenemos el punto de equilibrio del sistema; así tenemos:

$$x_o = \begin{bmatrix} 2.39 \\ 3.32 \end{bmatrix} \quad (3-55)$$

además mediante el programa CC obtenemos los polos de la matriz función de transferencia y que corresponden a los valores propios de la matriz A, los cuales nos servirán para la elección del período de muestreo, así tenemos:

$$\text{Eigenvalues: } \begin{matrix} -7.37178E-03 \\ -1.117222E-02 \end{matrix} \quad (3-56)$$

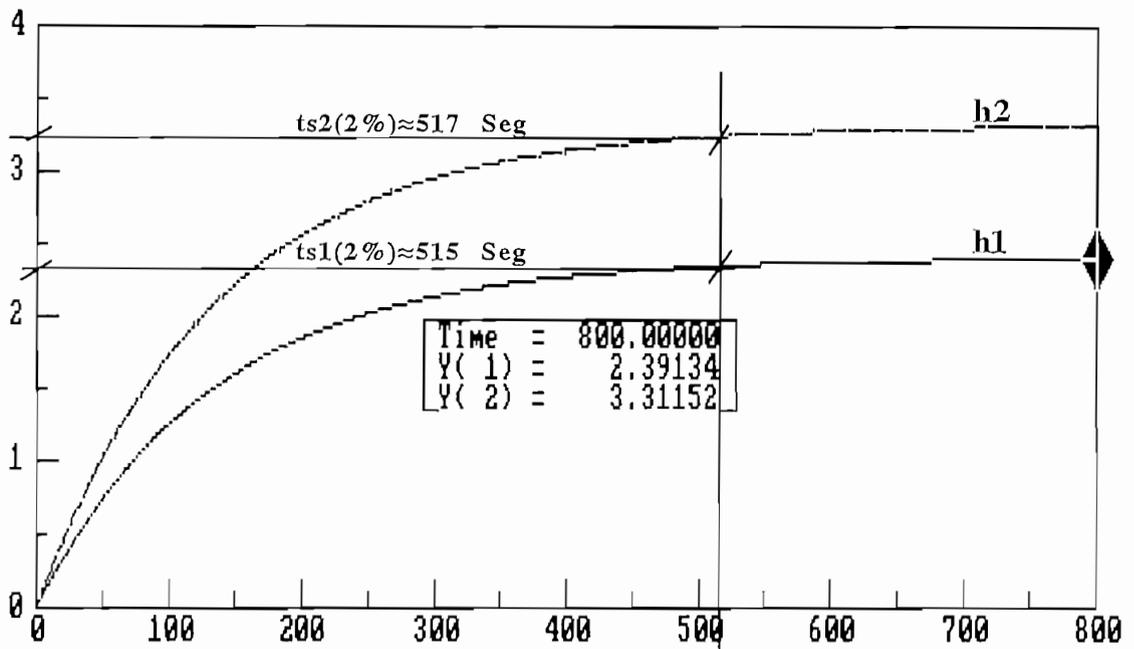


Fig. 3.7. Respuesta de los Tanques Acoplados en lazo abierto.

Como vemos la figura 3.7 corresponde a la respuesta del sistema obtenida mediante el programa CC, donde podemos darnos cuenta primeramente que los valores de estabilidad corresponden a los obtenidos en la ecuación (3-55), y que estos no se estabilizan en la referencia, además podemos determinar que el tiempo de establecimiento de la señal es relativamente grande, es decir estamos tratando con constantes de tiempo en el orden de los segundos (inverso de ecuación 3-56), lo cual debemos considerar en la selección del período de muestreo.

Si bien de este análisis podemos decir que en el sistema en lazo abierto los niveles tienden a seguir a la referencia con un determinado error, no es menos cierto que dicha respuesta es sensible a perturbaciones externas y a variaciones internas de los parámetros del sistema. Por lo tanto es necesario aplicar una acción de control en lazo cerrado, con el fin de lograr un sistema insensible a dichas perturbaciones.

Entonces primeramente vamos a realizar una realimentación unitaria de los estados, para determinar el comportamiento del sistema en lazo cerrado, para lo cual definimos a dicho sistema mediante la figura 3.8.

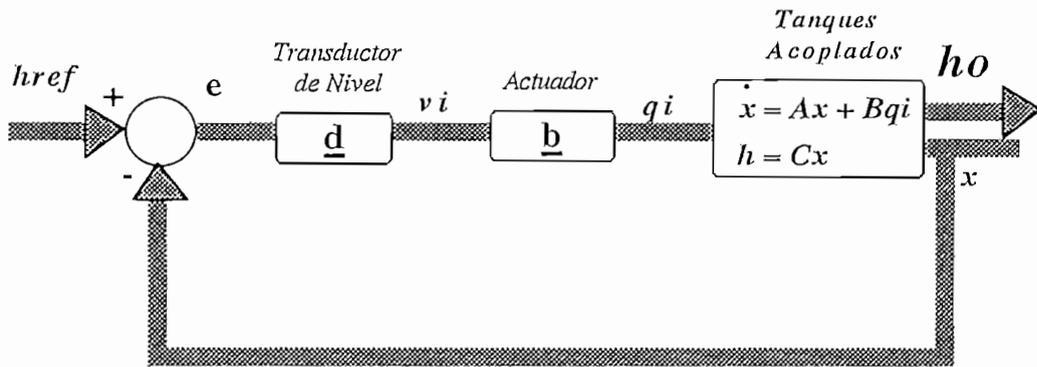


Fig. 3.8. Diagrama equivalente de los Tanques Acoplados con realimentación unitaria

En la figura 3.8 se cumplen las ecuaciones:

$$e = h_{ref} - x$$

$$v_i = dh_{ref} - dx$$

$$q_i = bdh_{ref} - bdx$$

$$\dot{x} = Ax + Bq_i$$

$$\dot{x} = Ax + B(bdh_{ref} - bdx)$$

$$\dot{x} = (A - Bbd)x + Bbdh_{ref}$$

en estado estable  $\dot{x}_i = 0$ , por lo tanto se obtiene los puntos de equilibrio del sistema:

$$0 = (A - Bbd)x_o + Bbdh_{ref}$$

$$x_o = -(A - Bbd)^{-1} Bbdh_{ref} \quad (3-57)$$

ahora utilizamos todas las matrices obtenidas en las ecuaciones (3-53 y 54), para el mismo punto de operación definido en la ecuación (3-51), calculamos en base a la ecuación (3-57) el punto de equilibrio para los niveles de los Tanques, así tenemos:

$$x_o = \begin{bmatrix} 1.07 \\ 1.26 \end{bmatrix} \quad (3-58)$$

el punto de equilibrio que obtuvimos en la ecuación (3-58), corresponde a los valores obtenidos mediante el programa CC, y mostrados en la figura 3.9.

además mediante el programa CC obtenemos los polos de la matriz función de transferencia equivalente y que corresponden a los valores propios de la matriz  $(A - Bbd)$ , así tenemos:

$$\text{Eigenvalues: } \begin{matrix} -1.771197\text{E-}02 \\ -2.191403\text{E-}02 \end{matrix}$$

(3-59)

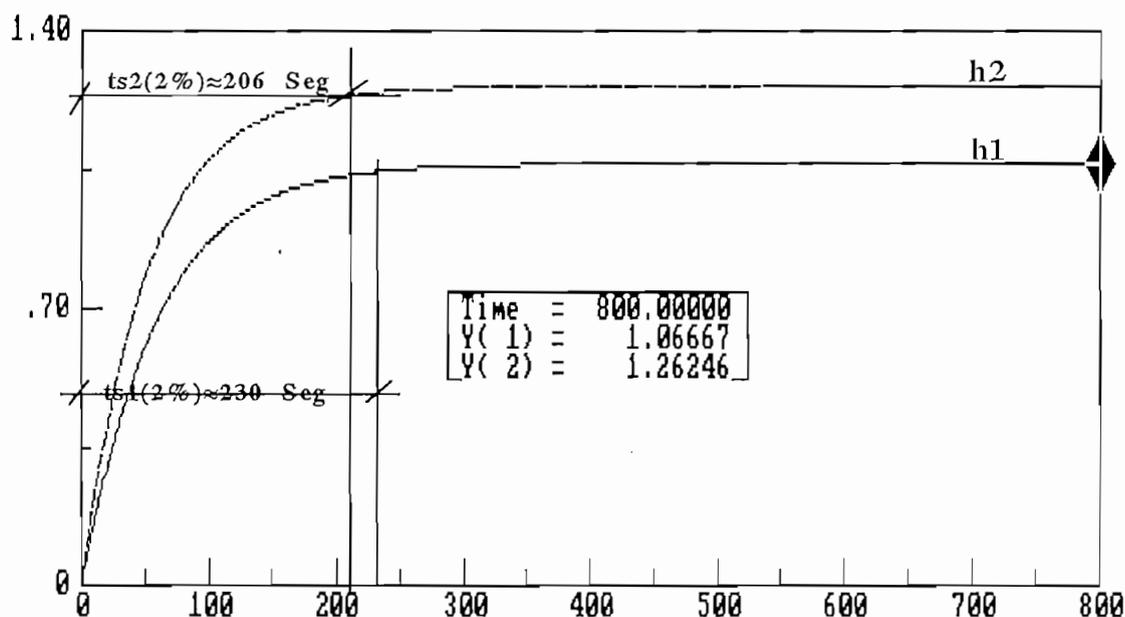


Fig. 3.9. Respuesta de los Tanques con realimentación Unitaria

Si comparamos la figura 3.9., con la respuesta del sistema en lazo abierto (Figura 3.7), vemos que el tiempo de establecimiento disminuye, con el inconveniente de que el error en estado estable aumenta. Pero como lo que nos interesa en el caso de los Tanques Acoplados, es que la desviación de la variable del punto de equilibrio sea mínima, es decir tratar de mantener un nivel estable ante cualquier tipo de perturbación, vamos a aplicar una técnica de regulación como es la aplicación del Regulador Cuadrático Lineal.

### 3.6 COMPENSACION DEL SISTEMA MEDIANTE LA UTILIZACION DEL REGULADOR CUADRATICO LINEAL.

Una de las técnicas de diseño de control óptimo moderno que ha encontrado aplicaciones prácticas generales es el diseño del Regulador Cuadrático Lineal. En un problema de regulación se considera con una referencia del sistema igual a cero, y el objetivo del diseño es llevar los estados o salidas a las

proximidades del punto de equilibrio, mediante la búsqueda de una ley de control que optimice al sistema, mediante la minimización de un cierto criterio o índice de funcionamiento.

Para determinar la ley de control Óptima, es necesario definir un índice de funcionamiento. Este índice es una medida cuantitativa del funcionamiento, midiendo la desviación respecto a un funcionamiento ideal. La especificación de la señal en el intervalo de tiempo de operación es denominada ley de control. Matemáticamente el problema básico de control es determinar la ley de *control óptimo*, sujeta a ciertas restricciones de ingeniería y económicas, que minimice un índice de funcionamiento dado. El índice de funcionamiento debe tener tres propiedades básicas: confiable, de fácil aplicación y selectivo. Debe ser confiable, porque dada una clase de sistema puede ser aplicado con seguridad; también debe ser de fácil aplicación y selectivo ya que el resultado tiene que ser un claro sistema óptimo.

Para sistemas relativamente simples se puede obtener la ley de control en forma analítica mientras que para sistemas más complejos puede ser necesario generar la ley de control óptimo con una computadora digital trabajando en línea.

El Regulador Cuadrático Lineal, tiene por objeto determinar la ley de control óptima  $u^0(x,t)$ , que siendo aplicada como la entrada del sistema, puede transferir el mismo desde un estado inicial a un estado final, mientras se consigue una minimización del índice de funcionamiento. (ver Figura 3.10)

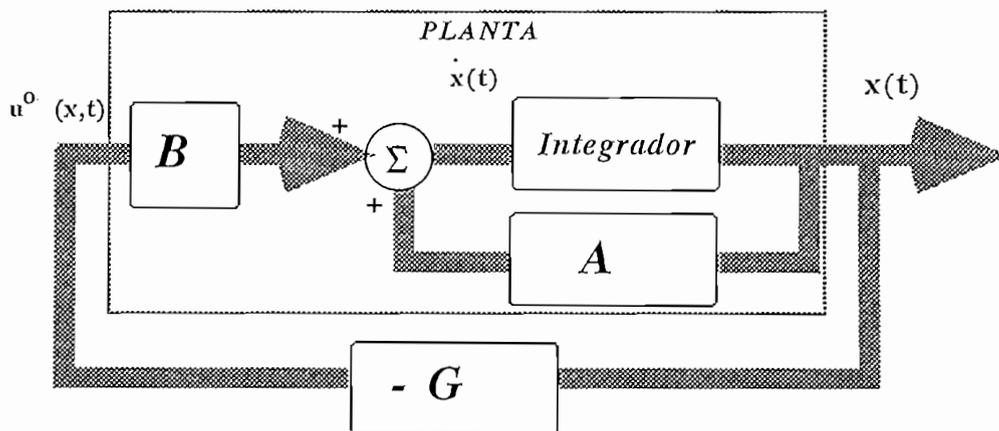


Fig. 3.10. Realimentación Óptima de estado.

Tomando la representación del sistema en variables de estado desde el punto de vista discreto ecuación (3-60) , lo que se desea es desarrollar un algoritmo que permita encontrar la ley de control óptima que será aplicada al sistema.

$$x[(k+1)] = A_d x(k) + B_d u(k) \quad (3-60)$$

Ahora, el problema que nos ocupa constituye en que dado el sistema discreto de la ecuación, se deba encontrar la ley de control óptima que minimice el índice de funcionamiento cuadrático que se presenta en la ecuación (3-61) ; así tenemos:

$$\text{minimizar } J_N = G[x(N)] + \sum_{k=0}^{N-1} F_k[x(k), u(k)] \quad (3-61)$$

donde:

$$G[x(N)] = 1/2 x^T(N) S x(N)$$

$$F_k[x(k), u(k)] = 1/2 x^T(k) Q x(k) + x^T(k) M u(k) + 1/2 u^T(k) R u(k) \quad (3-62)$$

S = Matriz simétrica semidefinida positiva.

Q = Matriz simétrica semidefinida positiva.

R = Matriz simétrica definida positiva.

M = Matriz de ponderación.

Como el objetivo del presente trabajo es aplicar el regulador cuadrático lineal al sistema de tanques acoplados, nosotros no realizaremos el proceso de deducción, sino utilizaremos los resultados que han sido tomada del libro de Benjamin Kuo (REF 3).

Mediante un proceso de inducción se puede llegar a determinar una fórmula general para obtener la ley de control óptima que minimiza el índice de funcionamiento cuadrático, la cual es conocida como la ecuación de diferencias de Riccati , y se muestra a continuación:

$$K(i) = Q + A_d^T K(i+1) A_d - (M^T + B_d^T K(i+1) A_d)^T \quad (3-63) \\ * (R + B_d^T K(i+1) B_d)^{-1} (M^T + B_d^T K(i+1) A_d)$$

donde la ley de control óptima viene definida por:

$$u^0(i) = -(R + B_d^T K(i+1) B_d)^{-1} (M^T + B_d^T K(i+1) A_d) x^0(i) \quad (3-64)$$

Esta ecuación puede ser escrita de la siguiente manera:

$$u^0(i) = -G(i)x^0(i)$$

Donde:

$$G(i) = -(R + B_d^T K(i+1) B_d)^{-1} (M^T + B_d^T K(i+1) A_d) \quad (3-65)$$

este método para obtener la Ecuación de Riccati (3-63) es conocido como programación dinámica, y utiliza el principio de optimalidad. Es interesante notar que este método de programación requiere que  $(R + B_d^T K(i+1) B_d)$  debe tener inversa.

Para el caso de tiempo infinito, o infinito número de intervalos de muestreo,  $N \rightarrow \infty$ , se tiene que  $K(k) \rightarrow K$ . La programación dinámica presentada constituye un método recursivo y se conoce generalmente como resolución recursiva de problemas de control óptimo.

Entonces empezando con la condición inicial  $K(N)=S$ , las ecuaciones (3-63) y (3-64) son resueltas por recursión, para el presente programa se utiliza una condición inicial  $S$  igual a cero, es decir con una ganancia de realimentación con valor inicial nulo, para finalmente adquirir un valor estable luego de algunas iteraciones del proceso recursivo.

Como el objetivo del presente trabajo es aplicar el Regulador Cuadrático Lineal como una alternativa de control, trabajaremos con el modelo completamente lineal, el mismo que esta representado en la fig. 3.11, con una nomenclatura modificada con respecto a la figura 3.10 para de esta manera poder incluir la referencia del sistema.

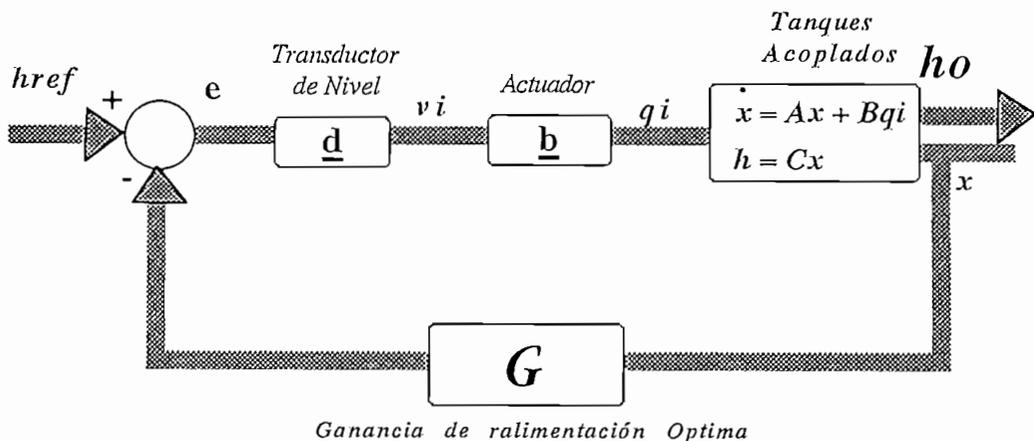


Fig. 3.11. Sistema de Tanques Acoplados compensador mediante un Control Óptimo

En la figura 3.11. se cumplen las siguientes ecuaciones:

$$\begin{aligned}
 e &= h_{ref} - Gx \\
 v_i &= d(h_{ref} - Gx) \\
 q_i &= bd(h_{ref} - Gx) \\
 \dot{x} &= Ax + Bq_i \\
 \dot{x} &= Ax + Bbd(h_{ref} - Gx) \\
 \dot{x} &= (A - BbdG)x + Bbdh_{ref} \quad \text{ó} \quad \dot{x} = A_{eq}x + B_{eq}h_{ref} \quad (3-66) \\
 \text{donde: } A_{eq} &= A - BbdG \quad \text{y} \quad B_{eq} = Bbd
 \end{aligned}$$

en estado estable  $\dot{x}_i = 0$ , por lo tanto el punto de equilibrio viene dado por:

$$\begin{aligned}
 0 &= (A - BGbd)x_0 + BbdR \\
 x_0 &= -(A - BbdG)^{-1} BbdR \quad \text{ó} \quad x_0 = -A_{eq}^{-1} B_{eq} R \quad (3-67)
 \end{aligned}$$

ahora utilizamos todas las matrices obtenidas en las ecuaciones (3-53 y 54), para el mismo punto de operación definido en la ecuación (3-51), y mediante el ingreso las matrices de ponderación Q y R, calculamos el Regulador Cuadrático Lineal mediante la utilización del programa CC, así tenemos:

$$\begin{aligned}
 Q &= \begin{vmatrix} 4 & 0 \\ 0 & 8 \end{vmatrix} \\
 R &= \begin{vmatrix} 30 & 0 \\ 0 & 20 \end{vmatrix} \Rightarrow G = \begin{matrix} \text{Matriz} \\ \text{Optima de estado} \end{matrix} \begin{vmatrix} 0.0604979 & 0.0146040 \\ 0.0302597 & 0.2456985 \end{vmatrix} \quad (3-68)
 \end{aligned}$$

Ahora reemplazamos todas las matrices en la ecuación (3-67), para obtener los puntos de equilibrio de las variables de estado y por consiguiente de las alturas del sistema, así tenemos:

$$x_0 = \begin{bmatrix} 2.07 \\ 2.35 \end{bmatrix} \quad (3-69)$$

el punto de equilibrio obtenido en la ecuación (3-69), corresponde al obtenido mediante la utilización del programa CC, el mismo que se muestra en la figura 3.12.

además mediante el programa CC obtenemos los polos de la matriz función de transferencia equivalente

y que corresponden a los valores propios de la matriz  $(A - BbdG)$ , así tenemos:

$$\text{Eigenvalues: } -9.37263E-03 \\ -1.271161E-02$$

(3-70)

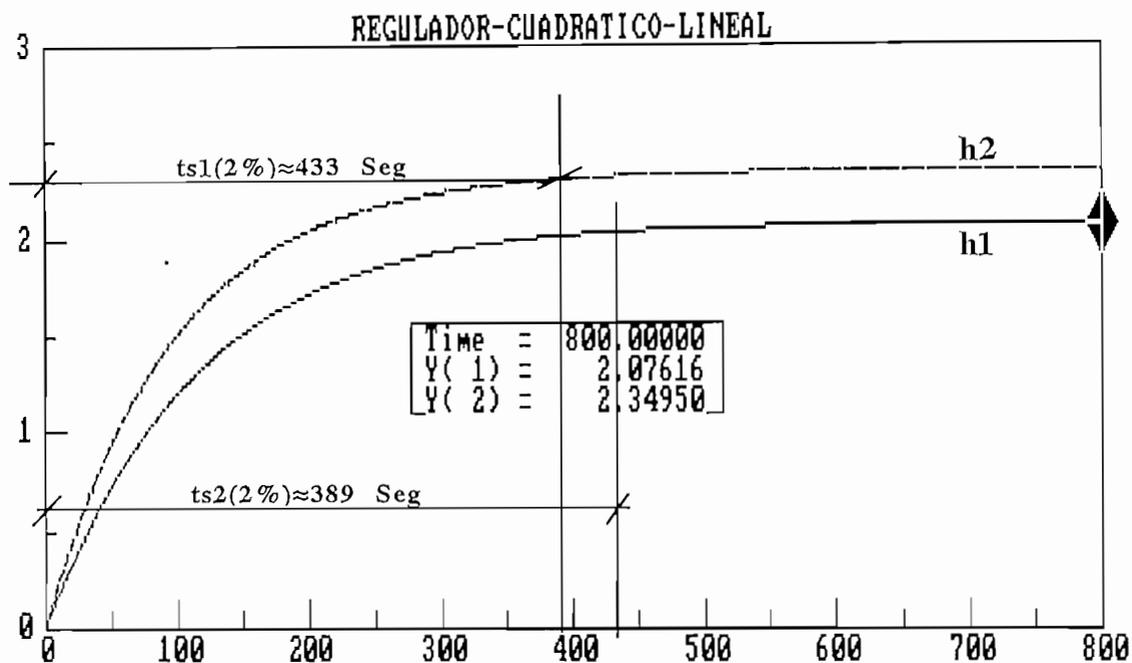


fig. 3.12. Alturas de los tanques con un Regulador Cuadrático Lineal

En la figura 3.12 podemos observar que los niveles de los tanques ya tienden a seguir a la referencia  $h_{ref}$ , pero esta respuesta es más lenta con respecto al sistema con realimentación unitaria mostrado en la figura 3.9, pero sin embargo siendo más rápido que el sistema en lazo abierto mostrado en la figura 3.7.

## ***CAPITULO IV***

# ***SIMULACION DINAMICA DE LOS SISTEMAS***

#### 4.1. ALGORITMO DE CALCULO MATEMATICO Y SIMULACION DINAMICA PARA EL SERVOMECANISMO DE POSICION MOTOMATIC MCL-100.

Antes de empezar a desarrollar los algoritmos para el cálculo matemático de la simulación del Motomatic y de las diferentes técnicas de compensación utilizadas, recordemos el diagrama de bloques que vamos a utilizar para la simulación, el mismo que lo representamos por la figura 4.1.

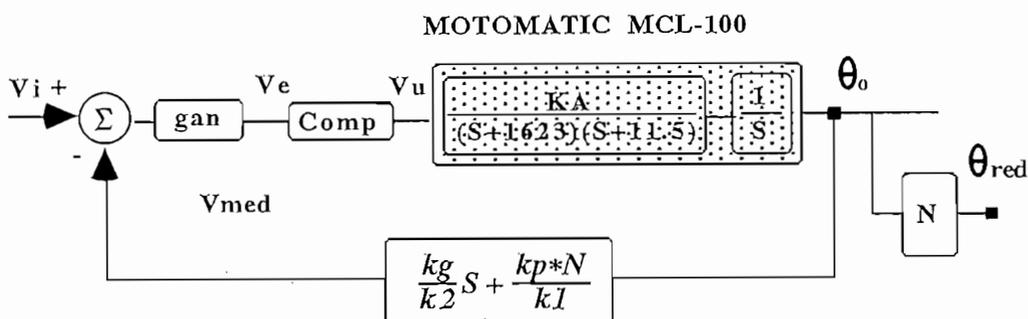


Figura 4.1. Diagrama equivalente utilizado para la simulación.

Este diagrama proviene de una reducción de términos realizada en los lazos de realimentación de la figura 2.9., donde la función de transferencia del MOTOMATIC está dada por:

$$\frac{\theta_o(S)}{V_a(S)} = \frac{KA}{S(1623 + S)(11.5 + S)} = G(S)$$

y donde la variable  $KA$  viene definida por:

$$KA = 1623 * 11.5 * 110 * 180 / \pi$$

El programa de simulación está hecho de tal manera que nosotros podemos ingresar parámetros del Motomatic, como son la relación de reducción de velocidad y los valores de las resistencia del punto de suma de la figura 2.5. Además podemos ingresar condiciones para la simulación como: número de iteraciones, período de muestreo, ángulo inicial y final.

El programa además permite la simulación del sistema compensado, ya sea mediante la utilización redes de compensación o acciones de control, y además el ingreso de cualquier compensador, pero previa la discretización de mismo.

Además es necesario anotar que la realimentación de velocidad nosotros la tomamos como una manera de compensación, ya que como veremos en las pruebas realizadas, lo que hace esta es mejorar el comportamiento dinámico del sistema, actuando como una acción derivativa en el lazo de realimentación.

En la figura 4.2 se muestra el algoritmo general empleado para la simulación del Motomatic, el mismo que permite ejecutar cualquiera de las 6 rutinas implementadas.

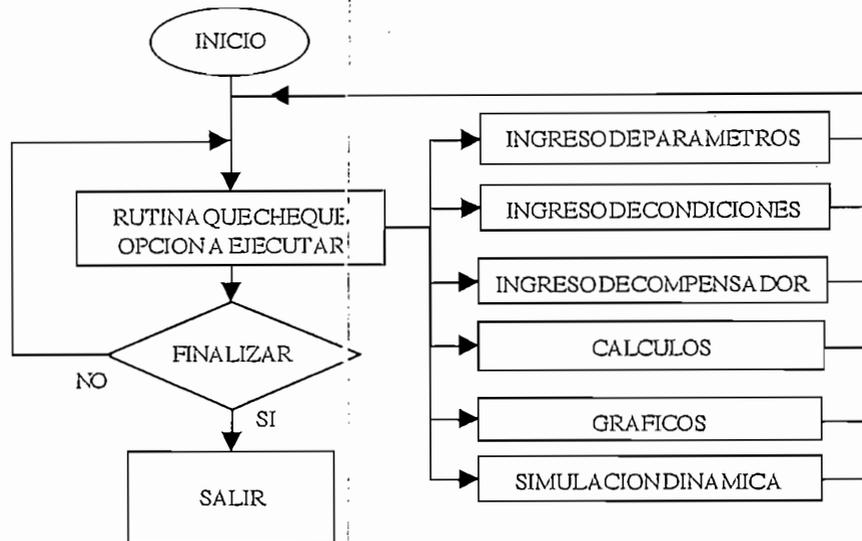


Figura 4.2. Esquema General del Programa de simulación

En la figura 4.3 tenemos el algoritmo detallado para el cálculo de los valores correspondientes a la simulación del Motomatic, y que los vamos a utilizar tanto en la simulación dinámica, como en los gráficos de las variables de interés.

Donde según la nomenclatura utilizada en el algoritmo de la figura. 4.3 tenemos:

$\theta_{red}(t)$  = Angulo medido en el reductor de velocidad.

$R_{ell}$  = Coeficiente de relación de velocidad N

$K_1 = R_1/R_0$

$K_2 = R_1/R_0$

$K_g$  = Es la ganancia del medidor de velocidad.

$K_p$  = Es la ganancia del potenciómetro de posición.

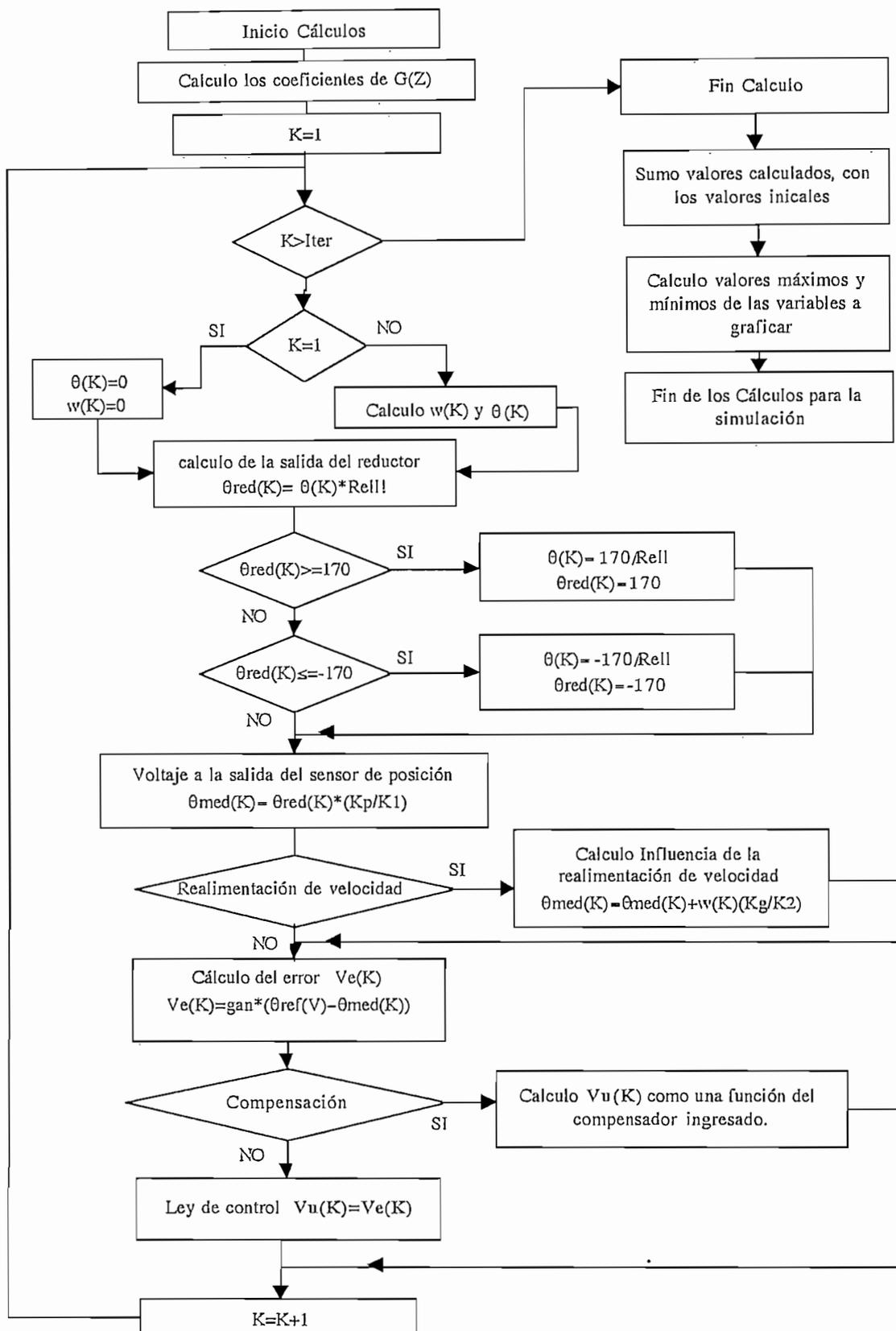


Figura 4.3. Algoritmo del cálculo para la simulación de Motomatic

En la sección 2.6.2 nosotros realizamos el análisis para el cálculo de una red de compensación, de tal manera que la figura 4.4 mostrará el algoritmo utilizado en el programa.

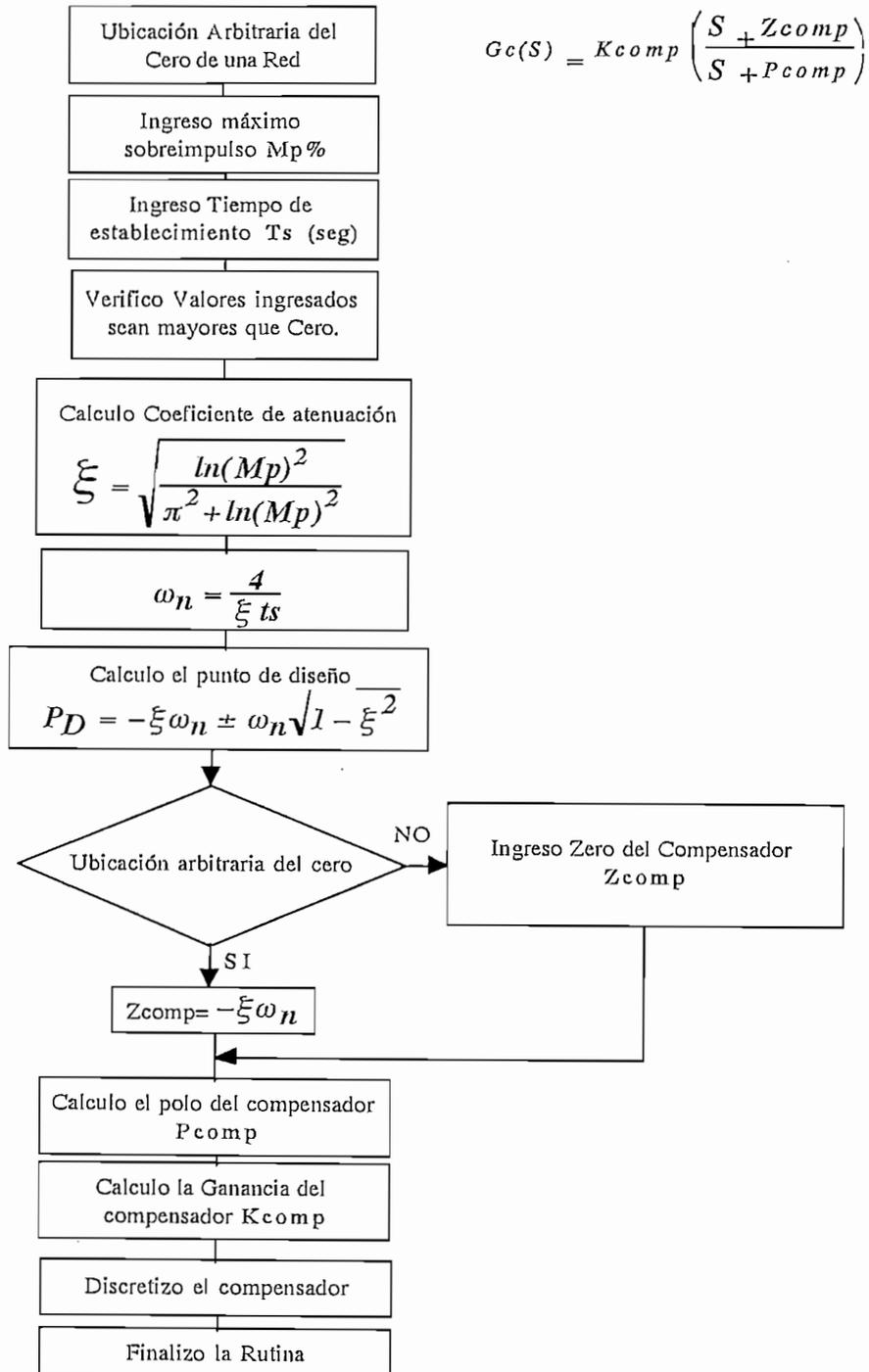


Figura 4.4. Algoritmo para calcular una red de compensación

#### 4.1.1. RESULTADOS OBTENIDOS DE LA SIMULACION DEL MOTOMATIC.

Para probar la validez del software desarrollado a nivel de simulación, presentamos a continuación los resultados obtenidos en base a ciertas condiciones de diseño. Sin embargo por razones de espacio no se presentarán todas las curvas posibles que pueden ser graficadas, las cuales sin embargo podrán ser analizadas para obtener algún comentario adicional, por parte del interesado.

Todos los gráficos mostrados en la presente sección corresponden a una parte de la pantalla del respectivo programa de simulación.

Para este numeral, analizaremos al Motomatic bajo los parámetros y condiciones de la Figura 4.5., de tal manera de encontrar similitudes con los resultados obtenidos analíticamente en el numeral 2.6, y que fueron obtenidos mediante la utilización del programa CC.

Primeramente analizamos el resultado obtenido sin la acción de ningún compensador; así tenemos la figura 4.5, la cual tiene como relación del reductor de velocidad de 1/9; en esta se puede observar la respuesta del ángulo de salida  $\theta(t)$  ante una entrada escalón, la misma que se comporta como un sistema de segundo orden, debido básicamente a la dominancia de los polos analizada en sección 2.6. En la misma figura calculamos el valor del sobreimpulso que es  $\approx 32.68\%$ , y un tiempo de establecimiento (criterio 2%) de  $\approx 0.66$  seg, valores que se asemejan a los obtenidos en la Figura 2.15.

En la Figura 4.6. realizamos el mismo análisis pero ahora tomando en cuenta una relación del reductor de velocidad de 1/1, de donde primeramente podemos determinar que el sobreimpulso es de aproximadamente del  $\approx 74.9\%$ , con tiempo de establecimiento de  $\approx 0.77$  Seg.

Esta variación en la respuesta se debe a que el la señal de salida ya no se ve atenuada por la relación de velocidad N y esto se ve manifiesto como un incremento del término independiente de la ecuación característica (2-29), lo cual obliga al desplazamiento de los polos de lazo cerrado del sistema hacia el lazo derecho del lugar geométrico de las raíces, por lo tanto perdiendo estabilidad.

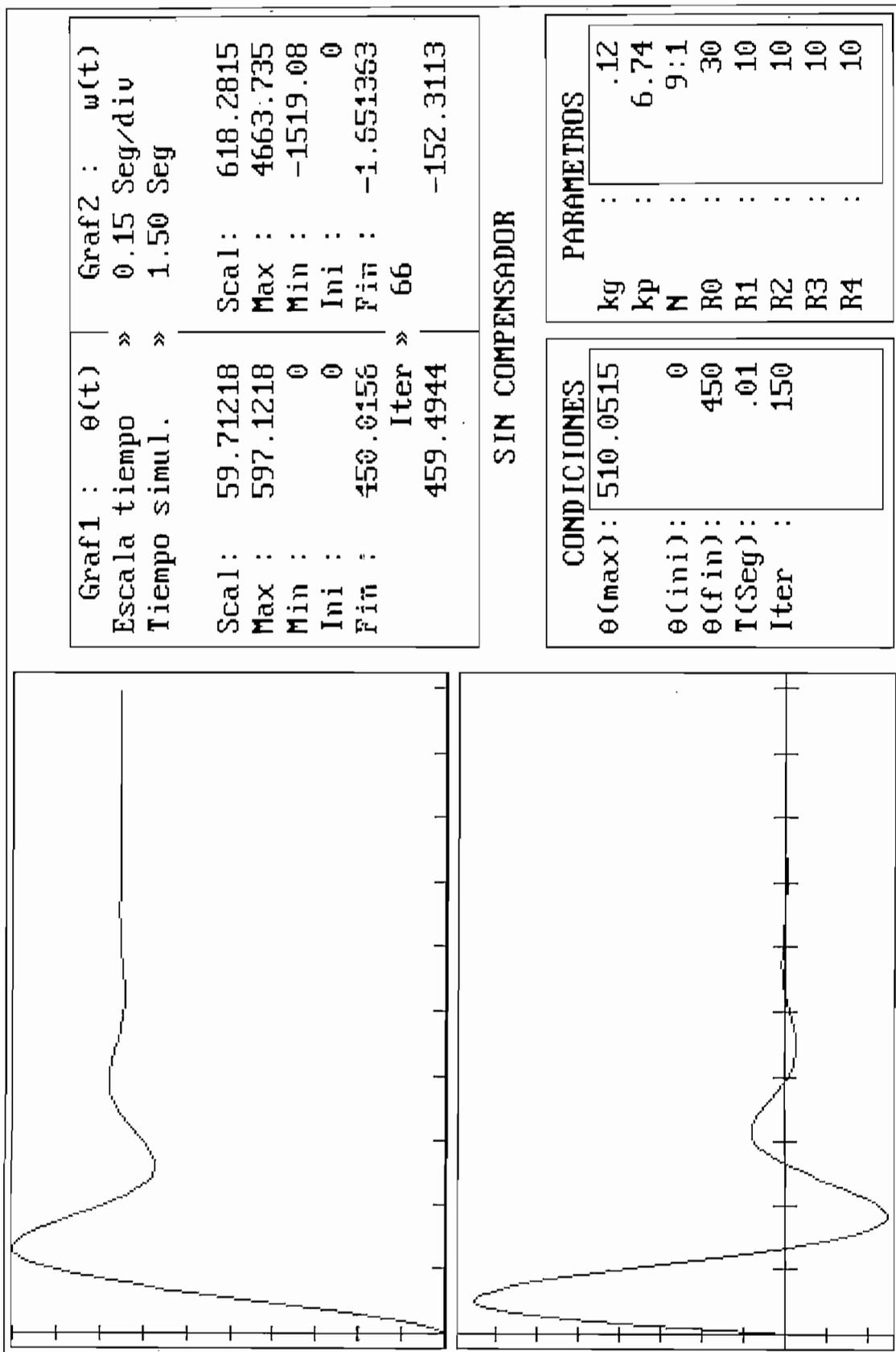


Figura 4.5. Posición y Velocidad angular Vs. Tiempo  
 Motomitic sin compensar N:119

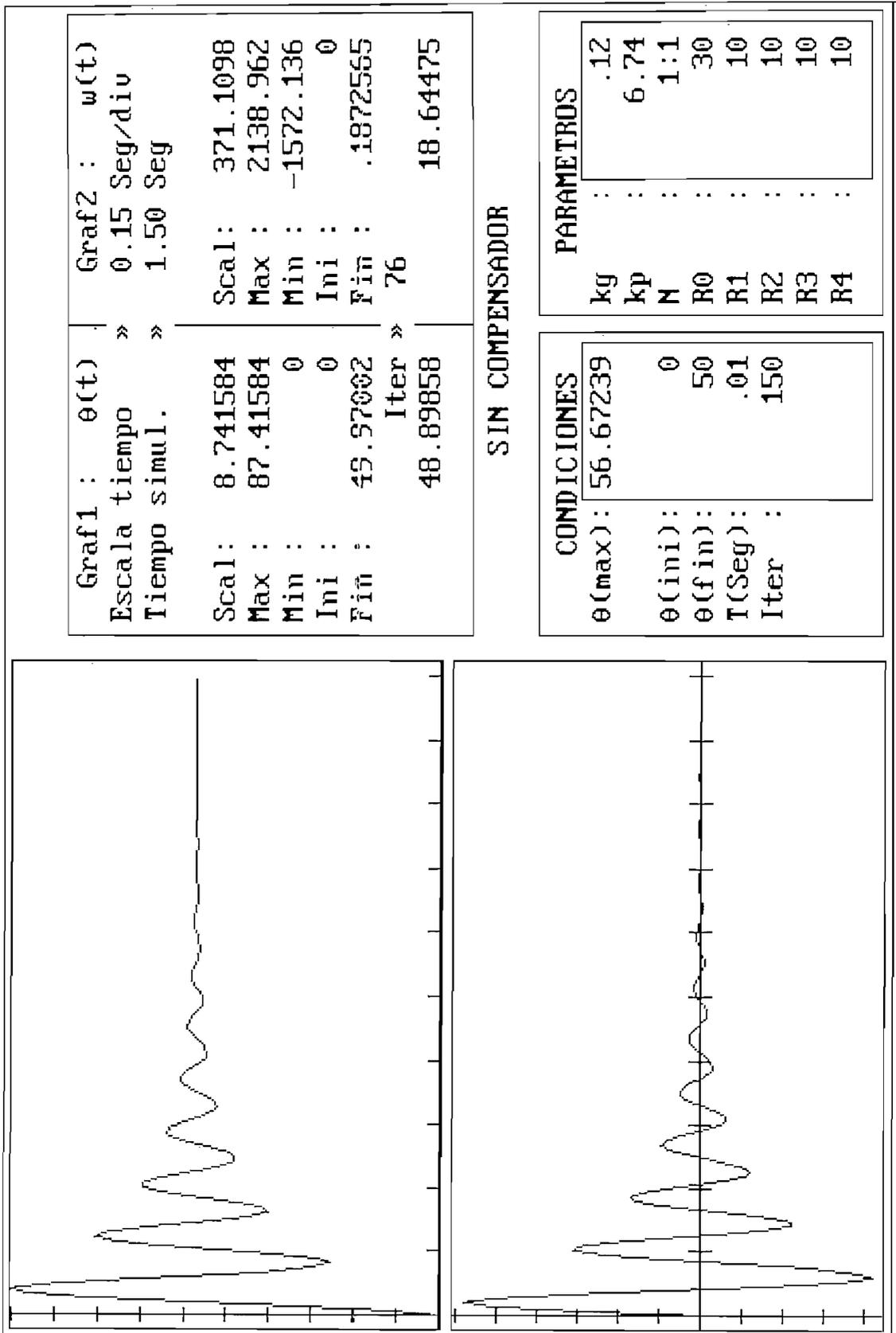


Figura 4.6. Posición y Velocidad Angular Vs. Tiempo

Motomático sin compensar N: 111

a) *REALIMENTACION DE VELOCIDAD.*

Como ya habíamos dicho en la sección 2.6.1, nosotros vamos a considerar a la realimentación de velocidad como una técnica de control, esto con el fin de mejorar su comportamiento dinámico.

En la figura 4.7., podemos observar que la respuesta del sistema ante una entrada paso y con una relación de velocidad de  $1/9$ ; aquí observamos el comportamiento como uno de primer orden ya que como vimos en la ecuación (2-37), los polos del lazo cerrado del sistema son reales, es decir siendo la respuesta del sistema sobreamortiguada. Además podemos determinar aproximadamente el tiempo de establecimiento  $\approx 0.87$  Seg, el mismo que se asemeja al valor obtenido en la Figura 2.17 mediante el programa CC.

En la figura 4.8. se muestra el error de posición, el cual vemos que no tiene la misma forma que la salida del sistema; y esto se debe a que como observamos en la figura 2.9, el error ya no solo es función de la referencia y de la salida, sino también del lazo interno de realimentación .

En la figura 4.9, tenemos la respuesta del sistema ante una entrada paso pero con una relación de velocidad de  $1/1$ ; aquí observamos su comportamiento como uno de segundo orden, con un sobreimpulso de  $\approx 6\%$  y un tiempo de establecimiento de  $\approx 0.118$  Seg, es decir que en este caso aumentamos la rapidez del sistema, pero disminuimos su estabilidad.

La diferencia del comportamiento al variar la relación de velocidad se puede explicar analizando la función de transferencia del lazo de realimentación , el cual se asemeja a una acción de control PD, tal es así que a este tipo de realimentación se la conoce como Taquimétrica o acción de control PD en el lazo de realimentación. Por lo tanto vemos que al reducir la relación de velocidad lo que estamos haciendo es aumentar la ganancia proporcional, es decir haciendo al sistema más inestable.

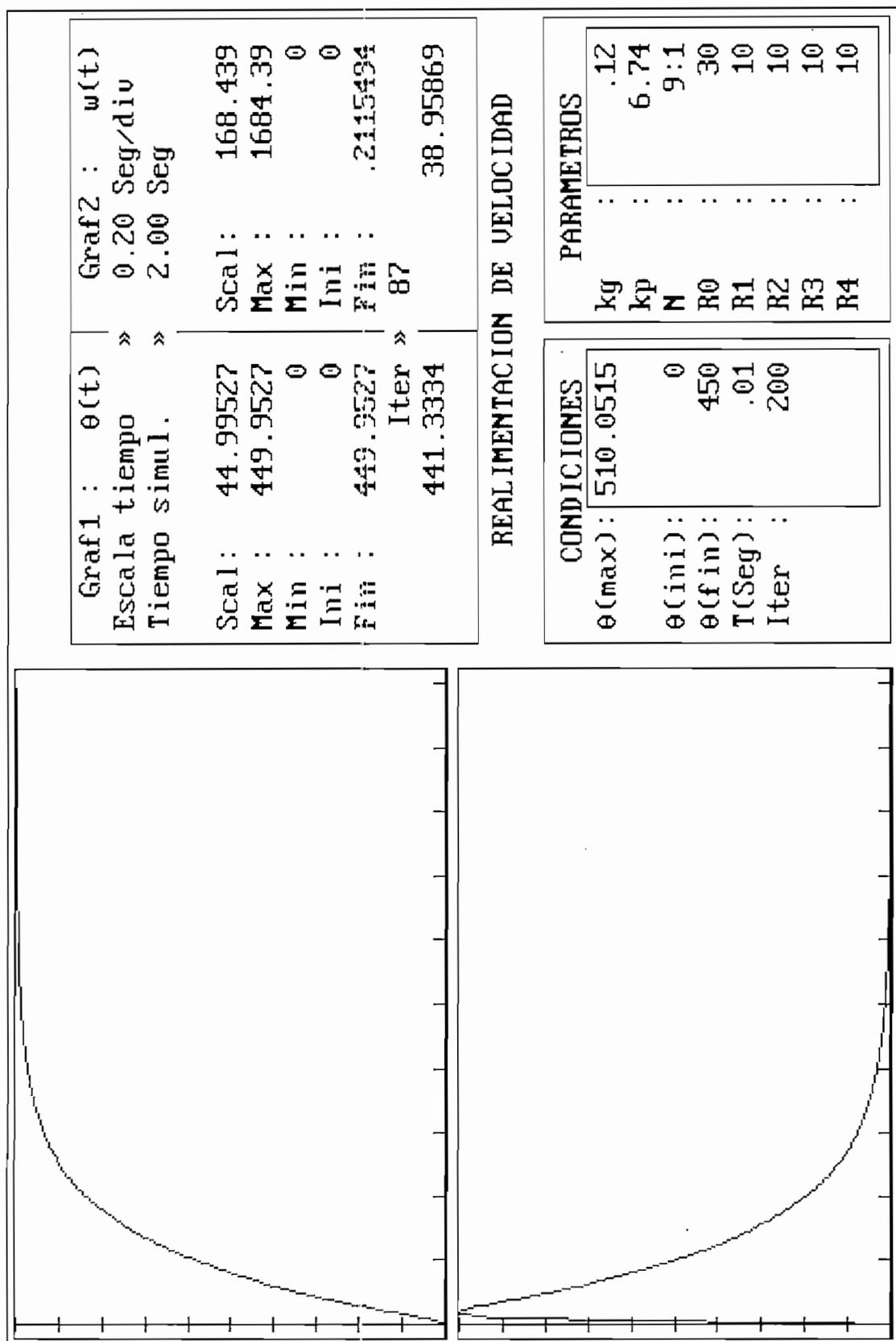
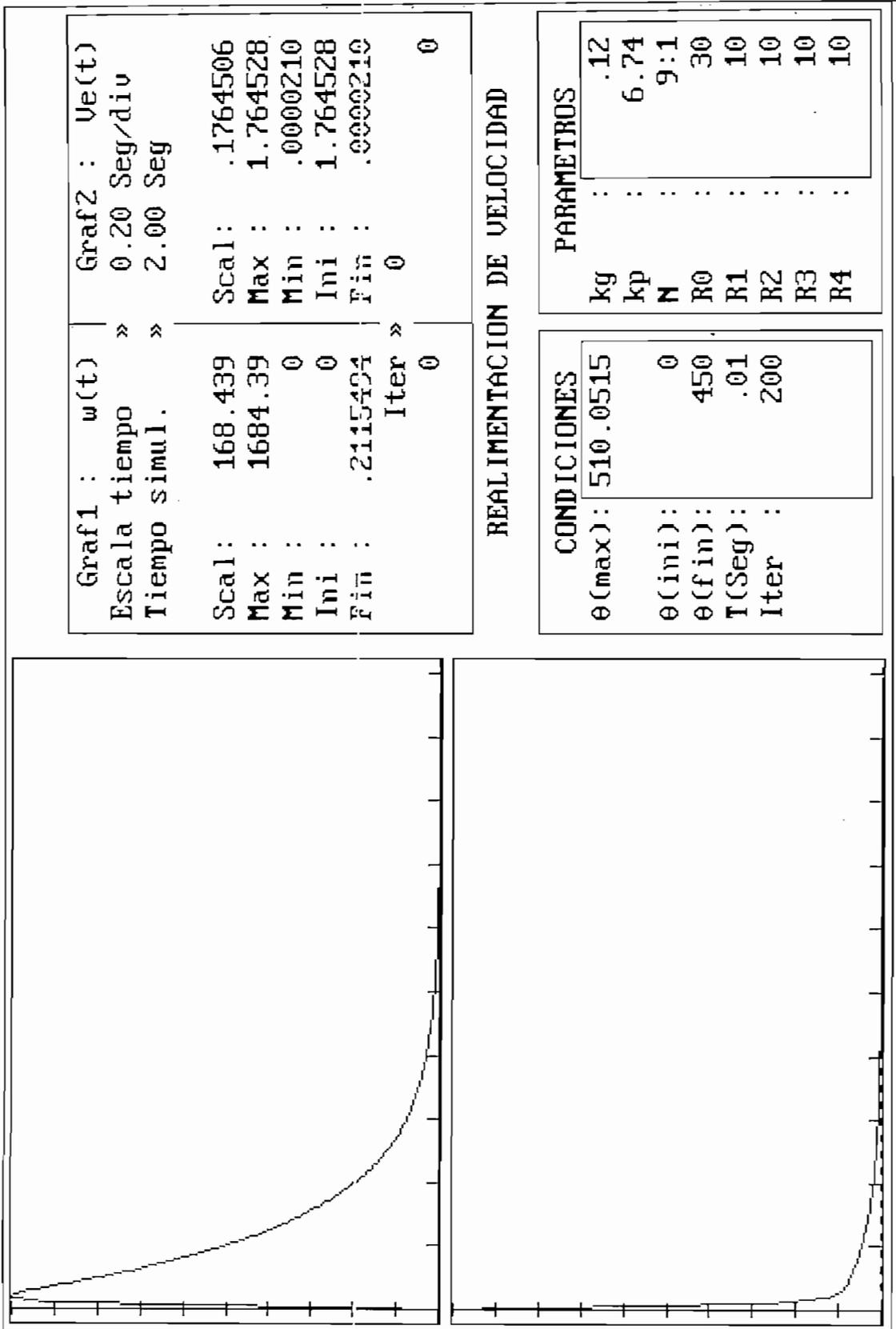


Figura 4.7. Posición y Velocidad Angular Vs. tiempo  
Motomatic con realimentación del Velocidad N: 1/9



Graf1 : $w(t)$	Graf2 : $Ve(t)$
Escala tiempo »	0.20 Seg/div
Tiempo simul. »	2.00 Seg
Scal: 168.439	Scal: .1764506
Max : 1684.39	Max : 1.764528
Min : 0	Min : .0000210
Ini : 0	Ini : 1.764528
Fin : .2115494	Fin : .0000210
Iter »	0
0	0

## REALIMENTACION DE VELOCIDAD

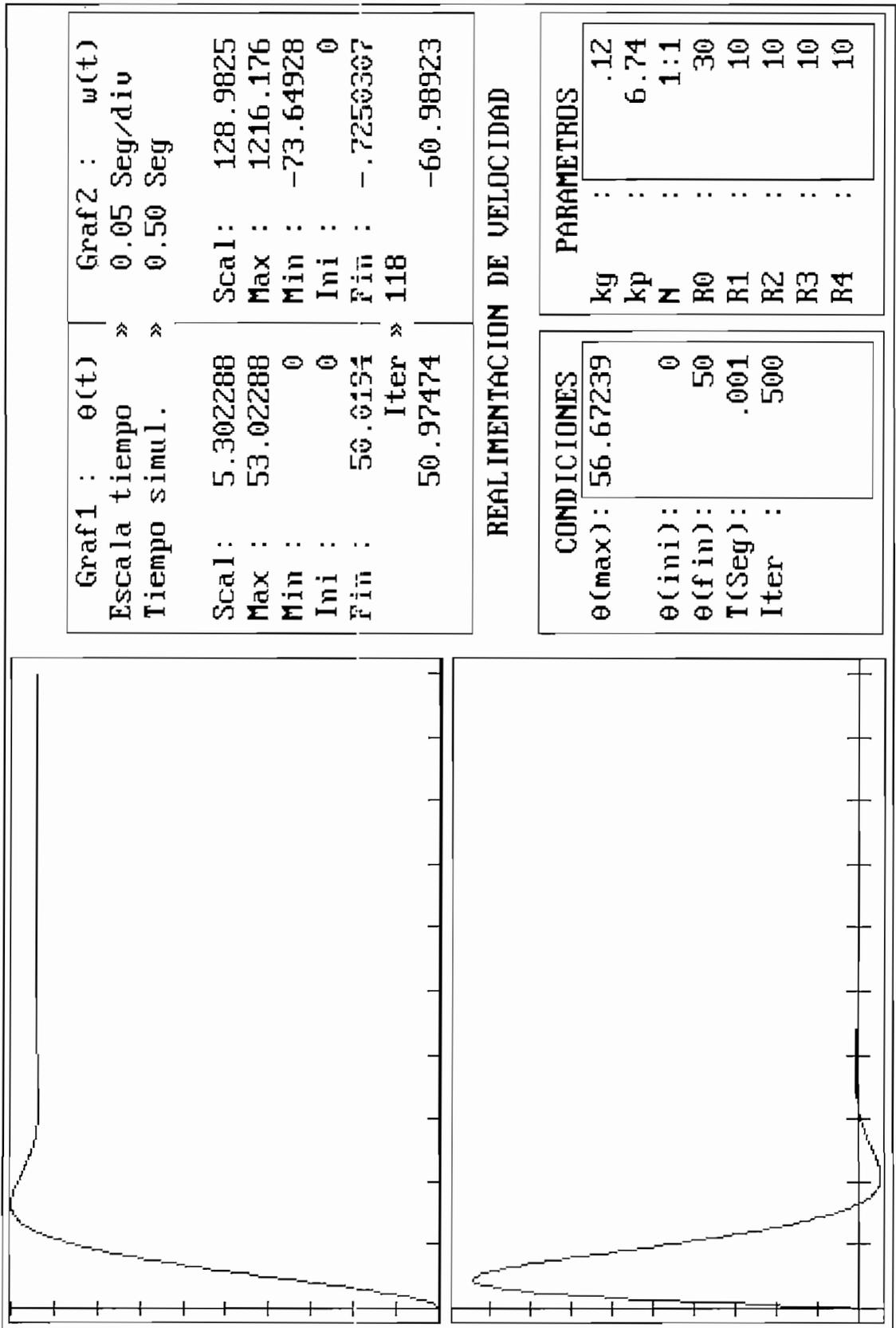
## CONDICIONES

$\theta(\max)$ :	510.0515
$\theta(\text{ini})$ :	0
$\theta(\text{fin})$ :	450
T(Seg):	.01
Iter :	200

## PARAMETROS

kg	: .12
kp	: 6.74
N	: 9:1
R0	: 30
R1	: 10
R2	: 10
R3	: 10
R4	: 10

Figura 4.8. Posición y error de Posición Vs. Tiempo  
Motomatic con realimentación de velocidad N: 119



Graf1 : $\theta(t)$	Graf2 : $w(t)$
Escala tiempo » 0.05 Seg/div	
Tiempo simul. » 0.50 Seg	
Scal: 5.302288	Scal: 128.9825
Max : 53.02288	Max : 1216.176
Min : 0	Min : -73.64928
Ini : 0	Ini : 0
Fin : 50.0194	Fin : -.7250307
	Iter » 118
	50.97474
	-60.98923

REALIMENTACION DE VELOCIDAD

CONDICIONES	
$\theta(\max)$ :	56.67239
$\theta(\text{ini})$ :	0
$\theta(\text{fin})$ :	50
T(Seg):	.001
Iter :	500
PARAMETROS	
kg :	.12
kp :	6.74
N :	1:1
R0 :	30
R1 :	10
R2 :	10
R3 :	10
R4 :	10

Figura 4.9. Posición y Velocidad Angular Vs. Tiempo  
Motomatic con realimentación de Velocidad N: 111

## b) UBICACION ARBITRARIA DEL CERO DE UNA RED

Mp%	:	10
ts (sg)	:	.5
Condiciones de Diseño		

En la figura 4.10. se muestra el compensador obtenido mediante el programa de simulación para  $N=1/9$ , el cual coincide con los obtenidos en la ecuación (2-43), para las mismas condiciones de diseño.

$$\begin{aligned} \text{Punto de diseño} &= -8 \pm j 10.91501 \\ \text{Compensador} &= .5738965 (S + 8) / (S + 11.72583) \end{aligned}$$

*Figura 4.10. Compensador obtenido mediante técnica de ubicación Arbitraria del cero de una red N: 1/9*

En la figura 4.12 se muestra la respuesta del sistema con una red de compensación, para  $N=1/9$ ; en la misma calculamos el valor del sobreimpulso de  $\approx 6.4\%$  y un tiempo de establecimiento de  $\approx 0.43$  Seg, valores que se asemejan a los obtenidos en la figura 2.19. Dicho valores son menores a lo de diseño y esto se debe a que si bien logramos que el lugar geométrico pase por el punto de diseño, esto no garantiza una dominancia del mismo.

En la figura 4.11., se muestra el compensador obtenido mediante el programa de simulación para  $N=1/1$ , el cual al aplicarlo sobre el sistema este tiene la respuesta obtenida en la figura 4.13, donde se puede determinar un sobreimpulso de  $\approx 6.42\%$  y un tiempo de establecimiento de  $\approx 0.43$  Seg.

$$\begin{aligned} \text{Punto de diseño} &= -8 \pm j 10.91501 \\ \text{Compensador} &= 6.376627E-02 (S + 8) / (S + 11.72583) \end{aligned}$$

*Figura 4.11. Compensador obtenido mediante técnica de ubicación Arbitraria del cero de una red N: 1/1*

Al comparar las figuras 4.12 y 4.13. tenemos que las respuestas en los dos casos son similares, y esto se debe a que al utilizar los mismos parámetros de diseño para el compensador, obtenemos un compensador tal que este es función de los nuevos parámetros de la planta, de manera que obliga a los polos de lazo cerrado pasar por el punto de diseño.

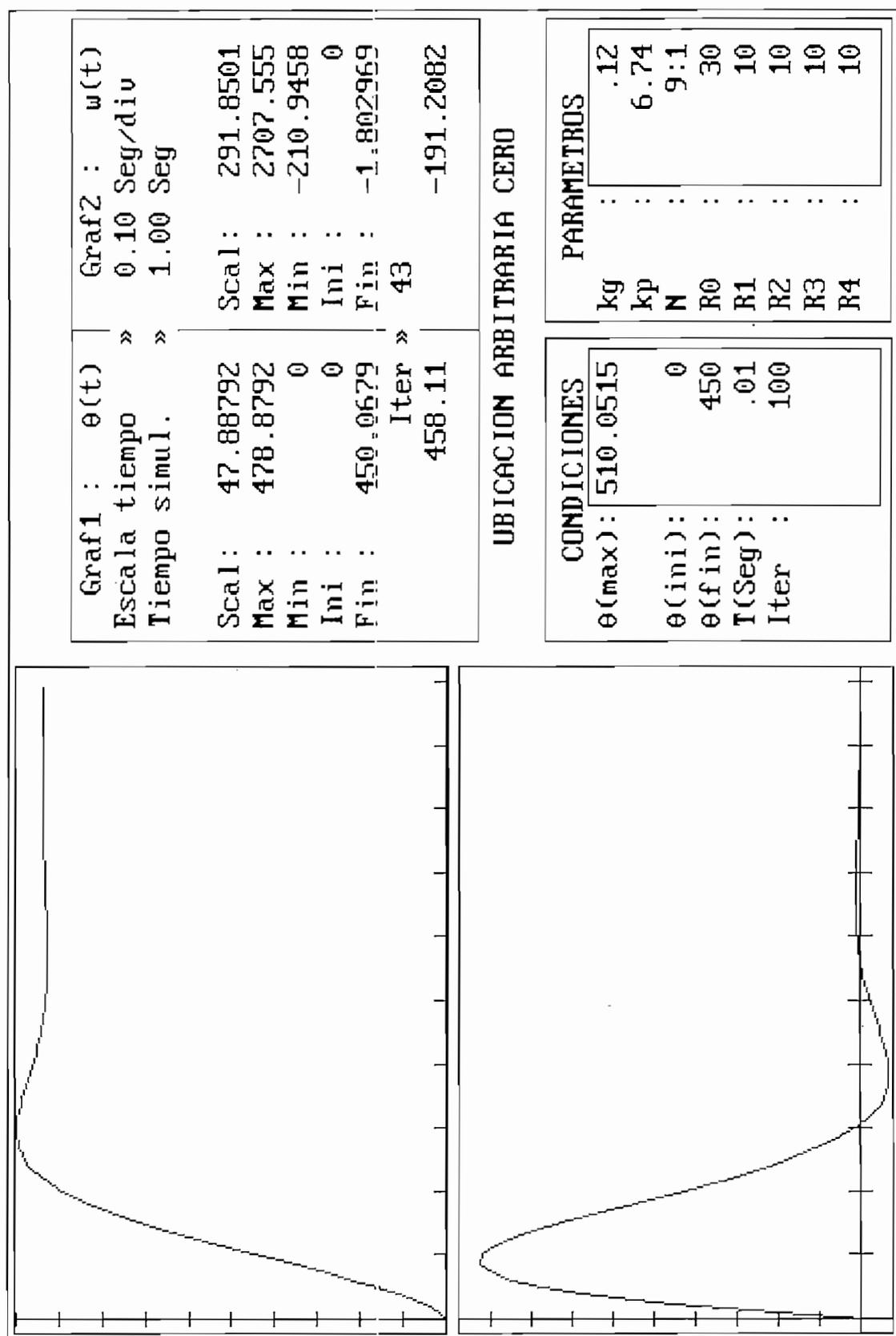


Figura 4.12. Respuesta del Motomatic compensado mediante la utilización de una Red N: 1/9

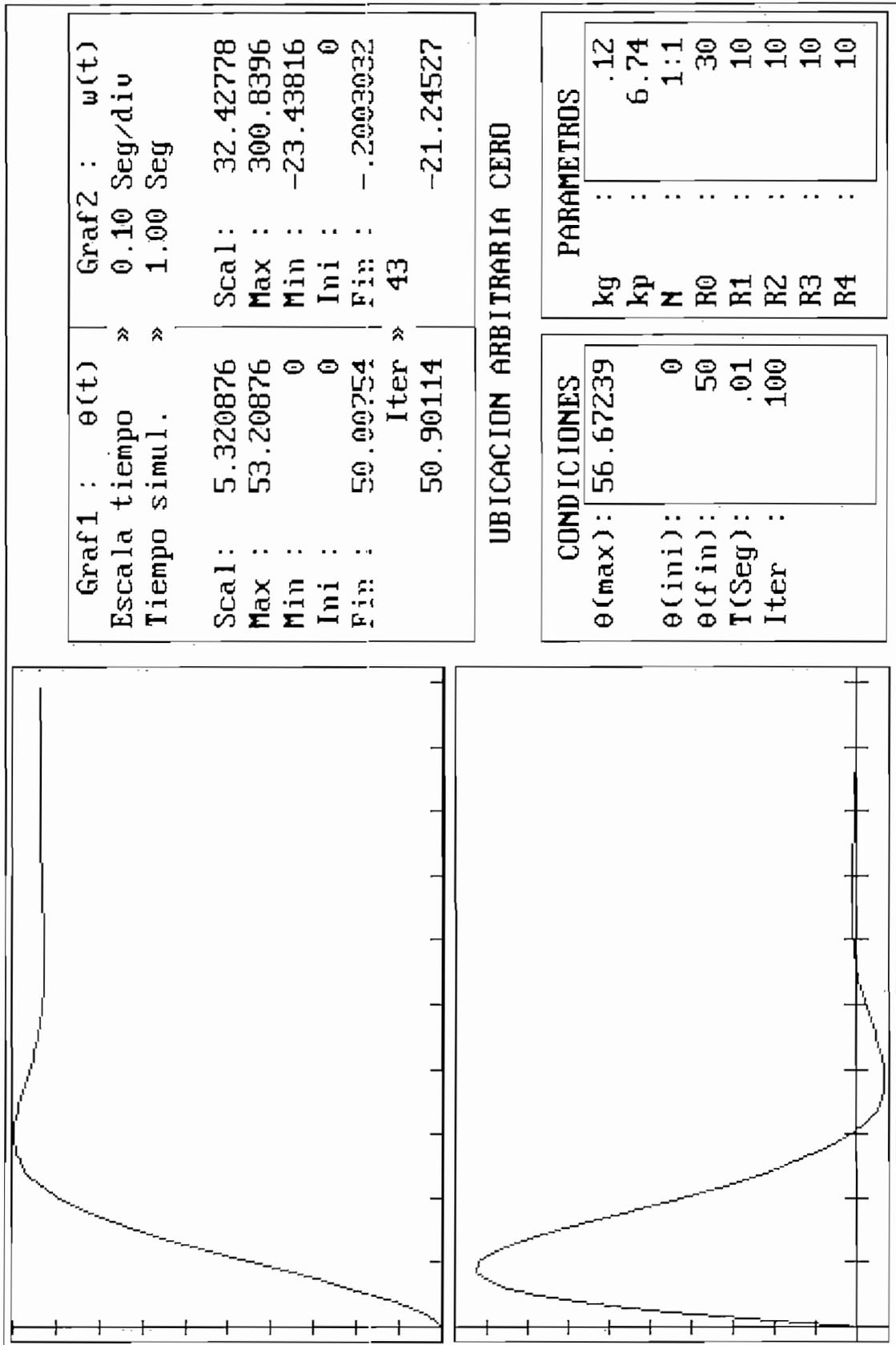


Figura 4.13. Respuesta del Motomatic compensado mediante la utilización de una Red N: 1/9

c) ACCIONES DE CONTROL.

KP : 20  
 KI : 30  
 KD : .01

Parámetros PID

Como podemos observar en la figura 4.15, tenemos el comportamiento del Motomatic con una relación de velocidad  $N=1/9$ , al cual le hemos aplicado una acción de control con los parámetros de control arriba mencionados. En dicha figura determinamos las especificaciones de respuesta transitoria, con un sobreimpulso de  $\approx 54\%$  y un tiempo de establecimiento de  $\approx 0.236$ , valores similares a los obtenidos en la figura 2.20 mediante el programa CC.

Al analizar la respuesta en el tiempo para la relación de reducción de velocidad de  $N=1/1$ , obtenemos la figura 4.16, en la que observamos que es sistema es inestable. Para verificar esto graficamos la respuesta mediante el programa CC, obteniendo la figura 4.14.

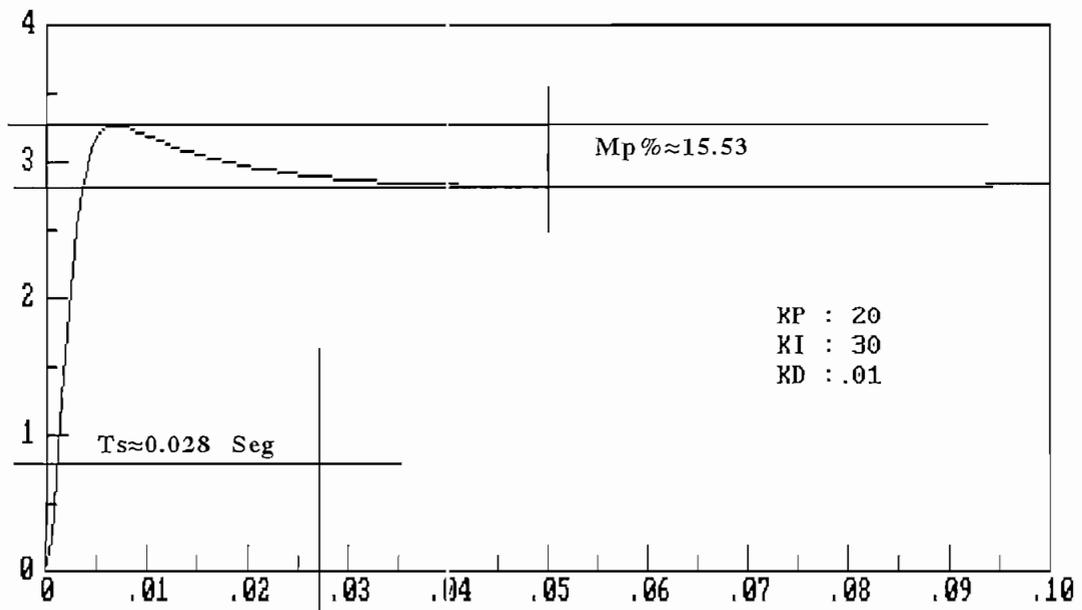


Figura 4.14. Respuesta del Motomatic compensado con una acción PID y  $N: 1/1$

En la figura 4.14 se observa que el sistema es estable, entonces procedemos al realizarnuevamente la simulación pero un período de muestreo menor  $T=0.0005$ , obteniendo el resultado representado en la

figura 4.17, en la que determinamos las especificaciones de respuesta transitoria de máximo sobreimpulso  $\approx 18,96\%$  y un tiempo de establecimiento  $\approx 0.027\text{Seg}$ , respuesta que se asemeja a la obtenida en la figura 4.14.

Si consideramos las raíces del polinomio característico de la ecuación (2-45), evaluada para nuestras condiciones de diseño; obtenemos

$$G20(s) = \frac{2.352674E+08(.01s^2 + s + 30)}{s^4 + 1634.5s^3 + 840936.1s^2 + 8.302716E+07s + 2.490815E+09}$$

cuyos polos de lazo cerrado vienen dados por:

P1 :	-57.76885012361534	+j	19.54357034061871
P2 :	-57.76885012361534	-j	19.54357034061871
P3 :	-759.4811498763847	+j	304.8079410391099
P4 :	-759.4811498763847	-j	304.8079410391099

Para poder muestrear bien la señal de posición, debemos considerar un adecuado período de muestreo, el cual debe ser menor que la menor componente de frecuencia, así tenemos que:

$$T < \frac{2\pi}{\omega_1} (\text{menor componente de frecuencia})$$

$$T < \frac{2\pi}{304.8} = 0.0206$$

si consideramos T unas cinco veces menor, tenemos que el período de muestreo elegido de  $t=0.005$  es el aceptable.

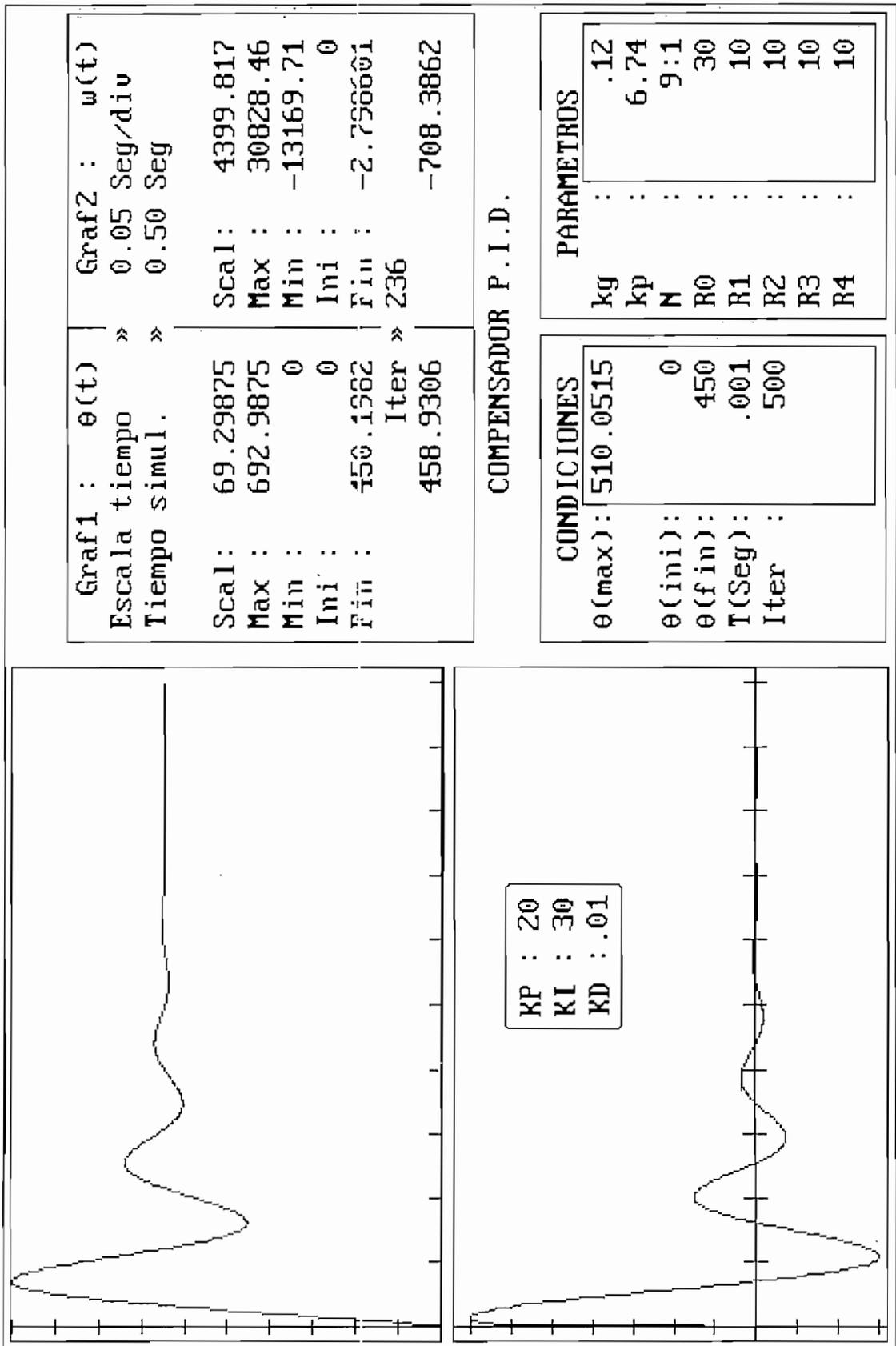


Figura 4.15. Respuesta del Motomatic compensado con una acción de control PID y N:1/9

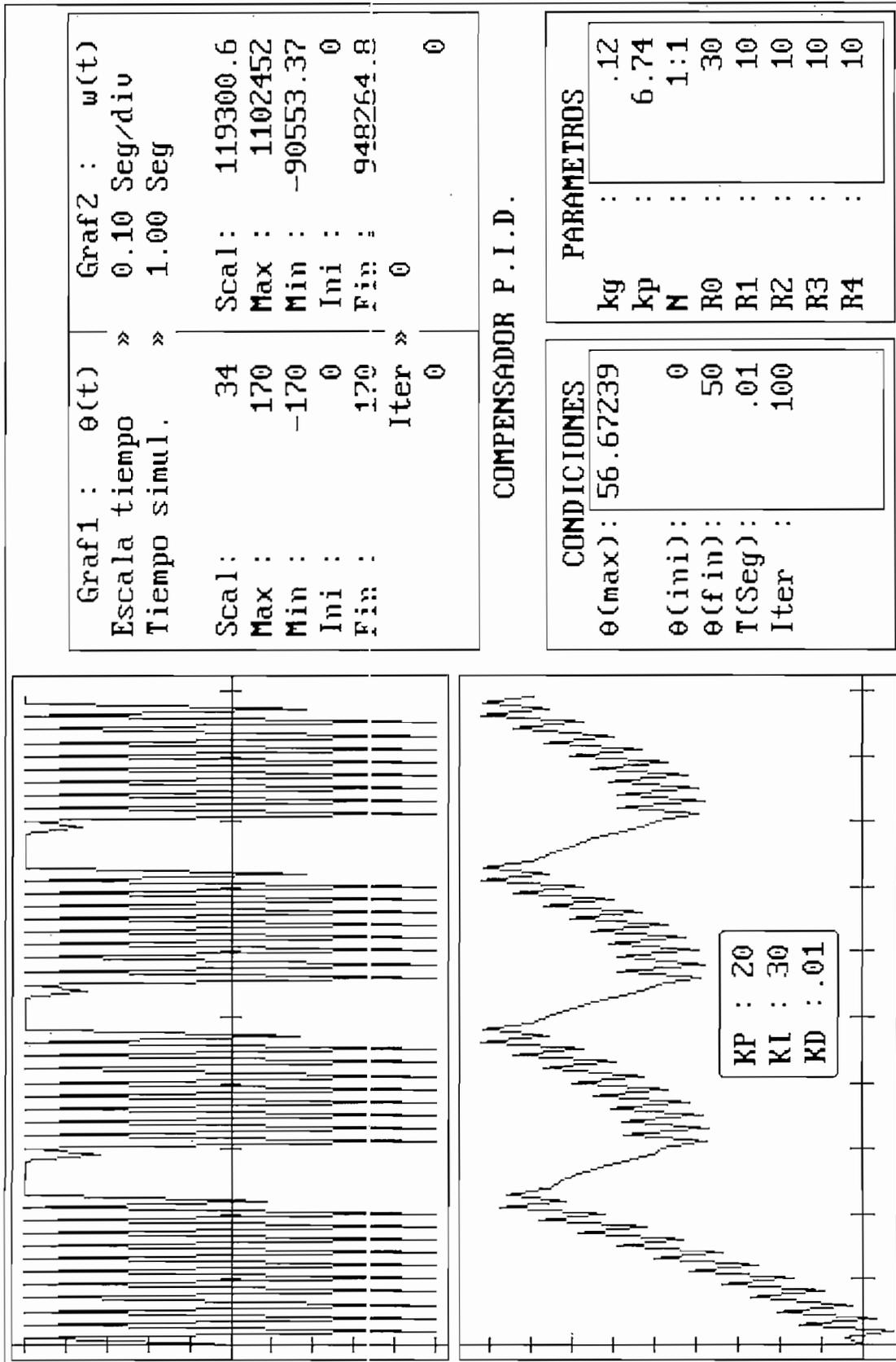


Figura 4.16. Respuesta del Motomatic compensado con una acción de control PID y N:111

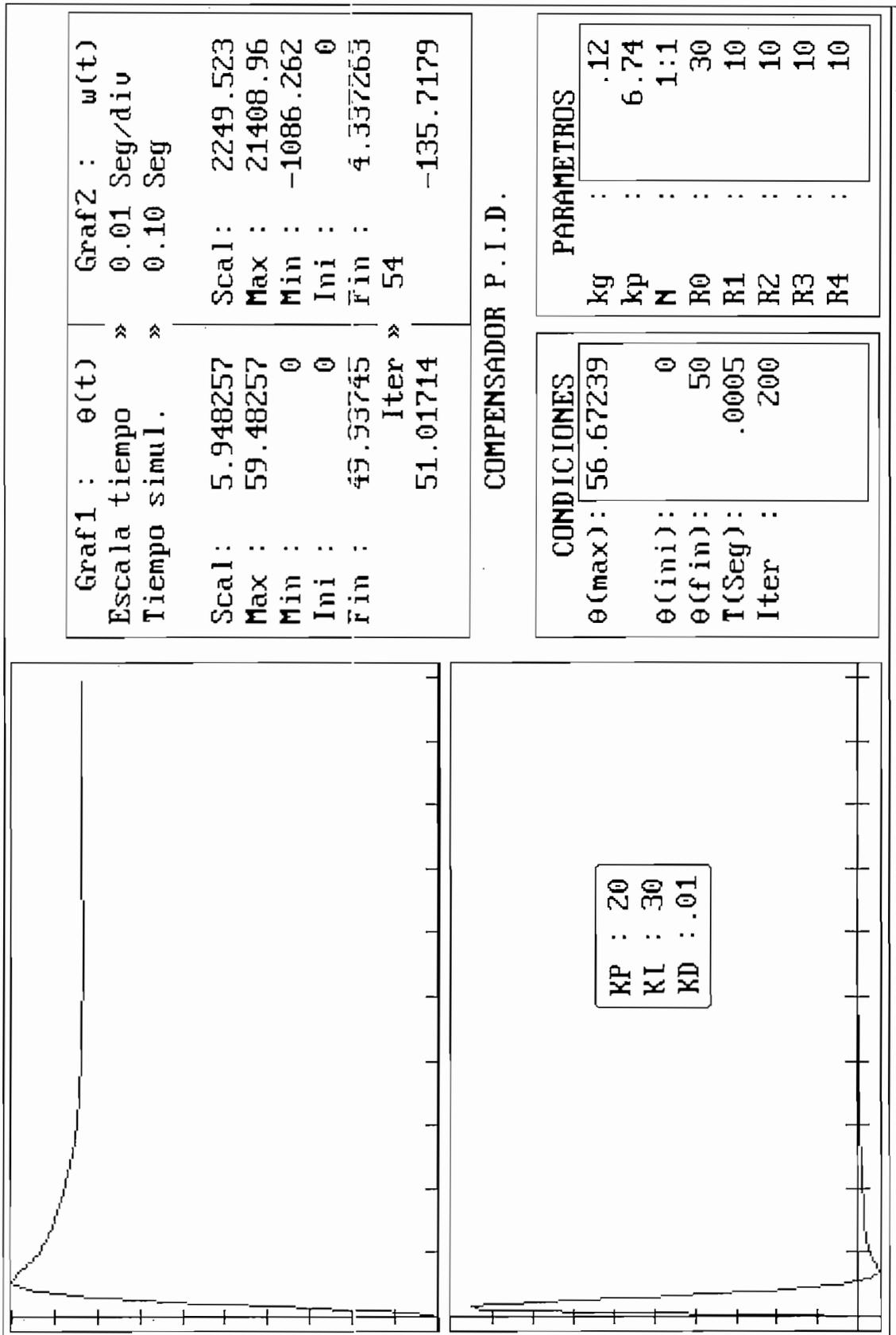


Figura 4.17. Respuesta del Motomatic compensado con una acción de control PID con N:1/1 y un menor período de muestreo

#### 4.2. ALGORITMO DE CALCULO MATEMATICO Y SIMULACION DINAMICA PARA EL SISTEMA DE NIVEL DE LIQUIDOS EN TANQUES ACOPLADOS.

Para empezar a desarrollar los algoritmos de cálculo de las técnica de compensación utilizada, recordemos el diagrama de bloques que vamos a utilizar para la simulación, el mismo que fue obtenido en la sección 3.5.

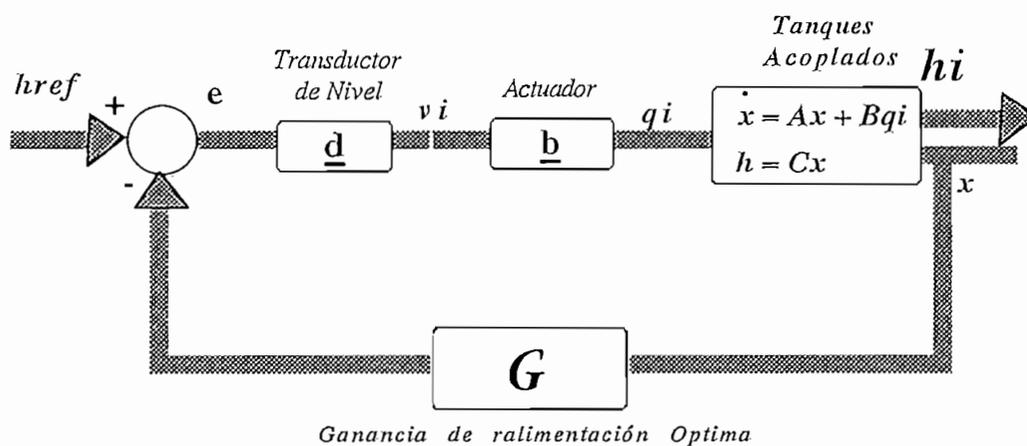


Figura 4.18. Diagrama de bloques lineal utilizado para la simulación.

donde las respectivas matrices fueron obtenidas en las secciones 3.1 y 3.2 ; las mismas que vienen representadas por:

$$\begin{bmatrix} qi_1 \\ qi_2 \end{bmatrix} = \begin{bmatrix} b_{11} & 0 \\ 0 & b_{22} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \quad \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} d_{11} & 0 \\ 0 & d_{22} \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \end{bmatrix}$$

$$A = \begin{bmatrix} -K_{11} & K_{12} \\ K_{21} & -K_{22} \end{bmatrix} \quad B = \begin{bmatrix} K_1 & 0 \\ 0 & K_2 \end{bmatrix} \quad C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

según esto tenemos el algoritmo general de ingreso de información que se muestra en la figura 4.19, que vamos a utilizar para la simulación, en donde nosotros antes de realizar los cálculos para la misma, debemos ingresar previamente las condiciones iniciales, finales, período de muestreo e iteraciones y las matrices para el cálculo del Regulador Cuadrático Lineal.

Es necesario anotar que en cada una de las opciones de la figura 4.19, se realiza un chequeo para ver si es posible ejecutar la misma; así para realizar los gráficos o la simulación de los Tanques, deberemos primeramente realizar los cálculos matemáticos, cuyo algoritmo se encuentra detallado en la figura 4.20.

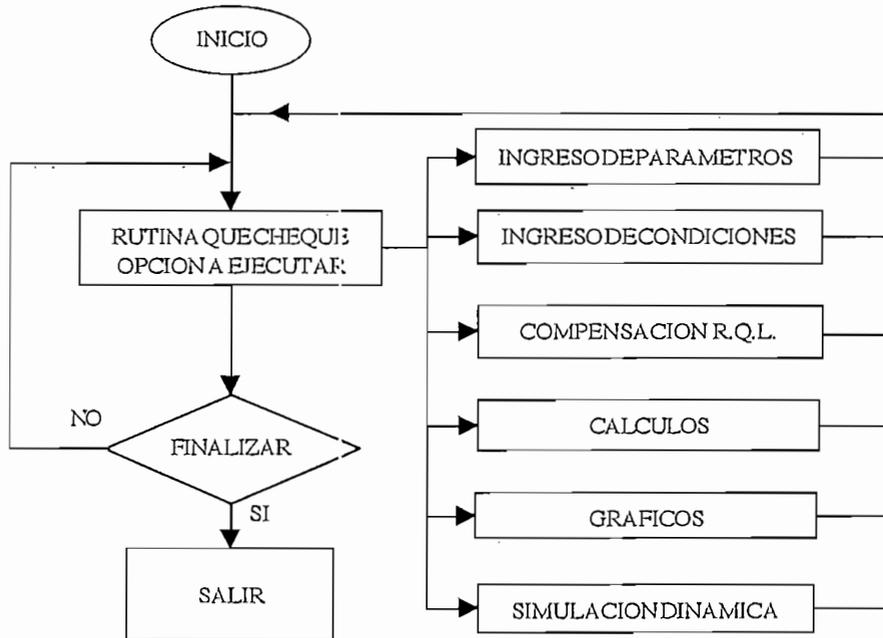


Figura 4.19. Esquema General del programa de simulación.

En la figura 4.20 se muestra el algoritmo detallado para el cálculo de los valores necesarios para la simulación de los Tanques Acoplados; aquí se puede observar que al finalizar los cálculos para el sistema lineal, procedimos a sumar los mismos con sus respectivos valores iniciales; y esto se debe a que esta representación lineal la definimos con respecto de un punto de operación  $\bar{Q}_{i1}, \bar{Q}_{i2}, \bar{V}_{i1}, \bar{V}_{i2}, \bar{H}_1, \bar{H}_2$ .

Además al finalizar todo el proceso de cálculo, existe una subrutina que se encarga de determinar los valores máximos y mínimos de todas las variables que pueden ser graficada.

Para facilitar el análisis del algoritmo empleado definiremos algunas variables según la nomenclatura utilizada, así tenemos:

$H_{ref}$  = Es la diferencia de alturas de los Tanques iniciales y finales

$Q_{i1max}$  = Caudal máximo suministrado por la Bomba 1

$Q_{i2max}$  = Caudal máximo suministrado por la Bomba 2

$Q_{i1}$  = Caudal inicial de la Bomba 1

$Q_{i2}$  = Caudal inicial de la Bomba 2

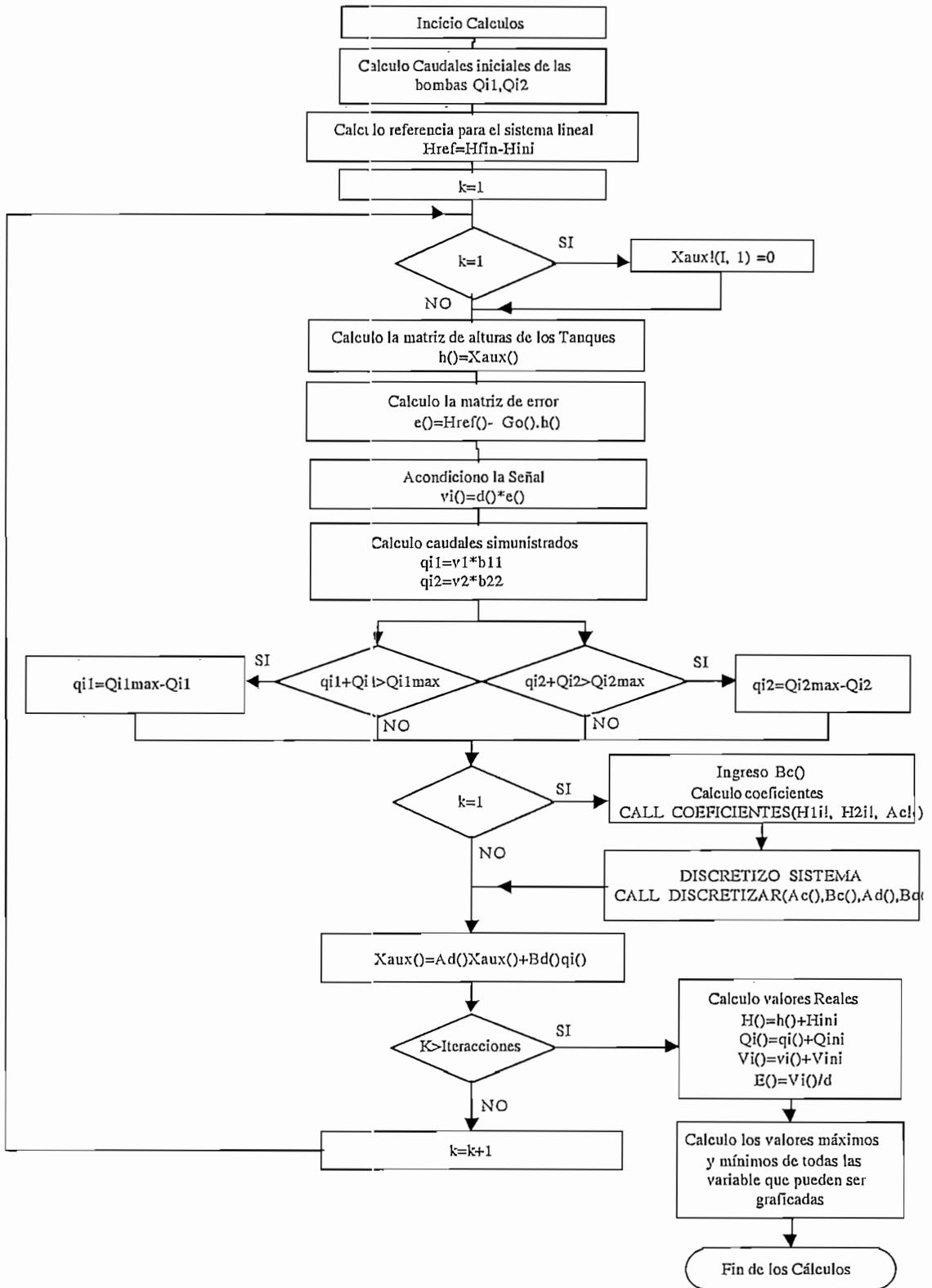


Fig. 4.20. Algoritmo del Cálculo de valores para la simulación de los Tanques Acoplados.

4.2.1. ALGORITMO DE DISCRETIZACION DEL SISTEMA.

Para realizar la discretización del sistema descrito por variables de estado, y que vamos a utilizar en el programa de simulación, utilizaremos las ecuaciones (3-40) y (3-41); así tenemos

$$Ad = e^{AT} = I + AT + \frac{A^2T^2}{2!} + \frac{A^3T^3}{3!} + \frac{A^4T^4}{4!} + \dots \quad (4-1)$$

$$B_d = (e^{AT} - I)A^{-1}B \quad (4-2)$$

si bien  $e^{AT}$  la podríamos evaluar mediante un algoritmo recursivo tomando en consideración  $n$  términos de la serie de la ecuación (4-1), donde la precisión de este método depende de  $n$ ; sin embargo debido a la forma de las matrices que describen a los Tanques Acoplados, vamos a utilizar a considerar en el presente trabajo, el método de Cayley Hamilton, el cual es más exacto y por lo tanto también la precisión de los resultados obtenidos en la simulación..

Para el Método de la transformada de Laplace primeramente consideramos de nuevo la descomposición en series de Taylor de la función  $e^{At}$ .

$$e^{At} = I + At + \frac{A^2t^2}{2!} + \frac{A^3t^3}{3!} + \frac{A^4t^4}{4!} + \dots + \frac{A^nt^n}{n!} \quad n \rightarrow \infty \quad (4-3)$$

el teorema manifiesta que  $e^{At}$  puede ser evaluado de la siguiente manera:

$$e^{At} = c_0I + c_1A + c_2A^2 + \dots + c_{n-1}A^{n-1} \quad (4-4)$$

en donde los coeficientes  $c_i$  no son términos constantes, sino función del tiempo.

Este método se basa en el cálculos de los valores propios obtenidos al resolver la ecuación característica, es decir:

$$\det|\lambda I - A| = 0 \quad (4-5)$$

de tal manera de tener la siguiente función:

$$e^{\lambda t} = c_0 + c_1\lambda + c_2\lambda^2 + \dots + c_{n-1}\lambda^{n-1} \quad (4-6)$$

la misma que deberá ser evaluada en cada uno de los valores propios.

Aplicando la ecuación (4-5) al sistema de Tanques Acoplados tenemos:

$$\det|\lambda I - A| = \det \begin{vmatrix} \lambda + k_{11} & -k_{12} \\ -k_{21} & \lambda + k_{22} \end{vmatrix} = 0 \quad (4-7)$$

$$\lambda^2 + (k_{11} + k_{22})\lambda + (k_{11}k_{22} - k_{12}k_{21}) = 0$$

ahora realizamos un cambio de variables, es decir:

$$b = (k_{11} + k_{22})$$

$$c = (k_{11}k_{22} - k_{12}k_{21})$$

tenemos que:

$$\lambda_{1,2} = \frac{-b \pm \sqrt{b^2 - 4c}}{2} \quad (4-8)$$

Si tomamos en cuenta la ecuación (4-4) y aplicamos para un sistema de segundo orden, tenemos:

$$e^{At} = c_0 I + c_1 A \quad (4-9)$$

$$e^{At} = \begin{vmatrix} c_0 + c_1 k_{11} & c_1 k_{12} \\ c_1 k_{21} & c_0 + c_1 k_{22} \end{vmatrix} \quad (4-10)$$

y evaluando la ecuación (4-9) para cada uno de los valores propios, obtenemos el siguiente sistema de ecuaciones:

$$e^{\lambda_1 t} = c_0 + c_1 \lambda_1 \quad (4-11)$$

$$e^{\lambda_2 t} = c_0 + c_1 \lambda_2 \quad (4-12)$$

resolviendo las ecuaciones (4-11) y (4-12) tenemos que:

$$c_1 = \frac{1}{\lambda_1 - \lambda_2} (e^{\lambda_1 t} - e^{\lambda_2 t}) \quad (4-13)$$

$$c_0 = \frac{1}{\lambda_1 - \lambda_2} (-\lambda_2 e^{\lambda_1 t} + \lambda_1 e^{\lambda_2 t}) \quad (4-14)$$

Finalmente para obtener  $e^{AT}$  tendremos que evaluar la ecuación (4-10) en  $t = T$ .

Este es el proceso utilizado para discretizar el sistema descrito por variables de estado; sin embargo para garantizar este método  $\lambda_{1,2}$  deberán ser reales, diferentes de cero y no deben tener multiplicidad.

Para esto analizaremos el discriminante de la ecuación (4-8), es decir:

$$b^2 - 4c > 0$$

$$\begin{aligned}
 b^2 - 4c &= (k_{11} + k_{22})^2 - 4(k_{11}k_{22} - k_{12}k_{21}) \\
 &= k_{11}^2 + 2k_{11}k_{22} + k_{22}^2 - 4k_{11}k_{22} + k_{12}k_{21} \\
 &= k_{11}^2 - 2k_{11}k_{22} + k_{22}^2 + k_{12}k_{21} \\
 &= (k_{11} - k_{22})^2 + k_{12}k_{21}
 \end{aligned}$$

y considerando que  $k_{12} = k_{21}$  por una condición del diseño según consta en el resultado obtenido en la ecuación (3-15), tenemos que:

$$\begin{aligned}
 b^2 - 4c &= (k_{11} - k_{22})^2 + k_{12}^2 > 0 \\
 \forall k_{11}, k_{22}, k_{12} &\neq 0
 \end{aligned}$$

la cual es una condición necesaria y suficiente tanto para poder decir que el sistema de Tanques Acoplados tiene valores propios tanto diferentes, como reales. De tal manera podemos aplicar el método de Cayley Hamilton sin ningún inconveniente.

Como un caso especial se puede dar que exista un polo  $\lambda$  con multiplicidad  $k$ , y esto se da en el caso de que  $H1 = H2$ . Entonces para este caso deberemos considerar la variante de este método en el caso de existir polos con multiplicidad, el mismo que se expresa:

Si  $\lambda$  tiene multiplicidad  $k$ .

$$1. \quad e^{\lambda t} = c_0 + c_1 \lambda + c_2 \lambda^2 + \dots + c_{n-1} \lambda^{n-1} \quad (4-15)$$

$$2. \quad \frac{de^{\lambda t}}{d\lambda} = t^k e^{\lambda t} = c_1 + c_2 \lambda + \dots + (n-1)c_{n-1} \lambda^{n-2} \quad (4-16)$$

$$k. \quad \frac{d^{k-1} e^{\lambda t}}{d\lambda} = t^{k-1} e^{\lambda t} = (k-1)! c_{k-1} + k! c_k \lambda + \dots + \frac{(n-1)!}{(n-k)!} c_{n-1} \lambda^{n-k} \quad (4-17)$$

Ahora aplicando esta definición a los tanques acoplados para el caso en que  $k=2$ , tenemos:

$$e^{\lambda t} = c_0 + c_1 \lambda_1 \quad (4-18)$$

$$te^{\lambda t} = c_1 \quad (4-19)$$

y resolviendo las ecuaciones (4-18) y (4-19) tenemos que:

$$c_1 = te^{\lambda t} \quad (4-20)$$

$$c_0 = e^{\lambda t} - \lambda te^{\lambda t} \quad (4-21)$$

y nuevamente para obtener  $e^{AT}$  evaluamos la ecuación (4-10) en  $t = T$ .

En la fig. 4.22 se muestra el diagrama de discretización implementado en el programa de simulación de los tanques acoplados.

#### 4.2.2. ALGORITMO PARA CALCULAR EL REGULADOR CUADRÁTICO LINEAL.

Como ya analizamos en la sección 3.6, el proceso para determinar el Regulador Cuadrático Lineal es el conocido como programación lineal, el cual en el cual empezamos con una condición  $K(n)=S$ , las ecuaciones (3-50) y (3-52) son resueltas por recursión. Nosotros consideramos  $S(0)=0$ , es decir que la ganancia de realimentación  $K$  inicialmente tiene un valor nulo, para luego de algunas iteraciones obtener un valor estable y así llegar a obtener la matriz de realimentación óptima de estado  $G$ .

En la figura 4.21 mostramos el algoritmo implementado para el cálculo del regulador cuadrático lineal, aquí consideramos que  $Q()$  y  $R()$  deben ser matrices simétricas y que pese a que el algoritmo así no lo requiere  $R()$  debe tener inversa; sin embargo al realizar el proceso de cálculo, el término de la ecuación de Ricatti  $(R + B_d^T K(i+1) B_d)$  debe tener inversa, es decir el resultado de dicha evaluación debe ser una matriz singular; de no ser así el programa pide que tratemos con otra matriz  $R()$ .

Para todo el proceso de cálculo se utilizan subrutinas de operaciones con matrices, como por ejemplo: Multiplicación, Suma, Inversa, y Traspuesta de matrices.

Es conveniente recordar que las matrices a ingresar deben ser:

$Q$  = Matriz simétrica semidefinida positiva.

$R$  = Matriz simétrica definida positiva.

El programa tiene además la opción de garantizar que las matrices ingresadas  $Q()$  y  $R()$ , sean simétricas, para lo cual solamente el programa nos permite ingresar el término  $Q(1,2)$  y  $R(1,2)$ , para luego obtener  $Q(2,1)=Q(1,2)$  y  $R(2,1)=R(1,2)$ .

Como condición inicial hacemos  $M(0)=0$  para de esta manera realizar las pruebas con las matrices de ponderación de los estados y del control.

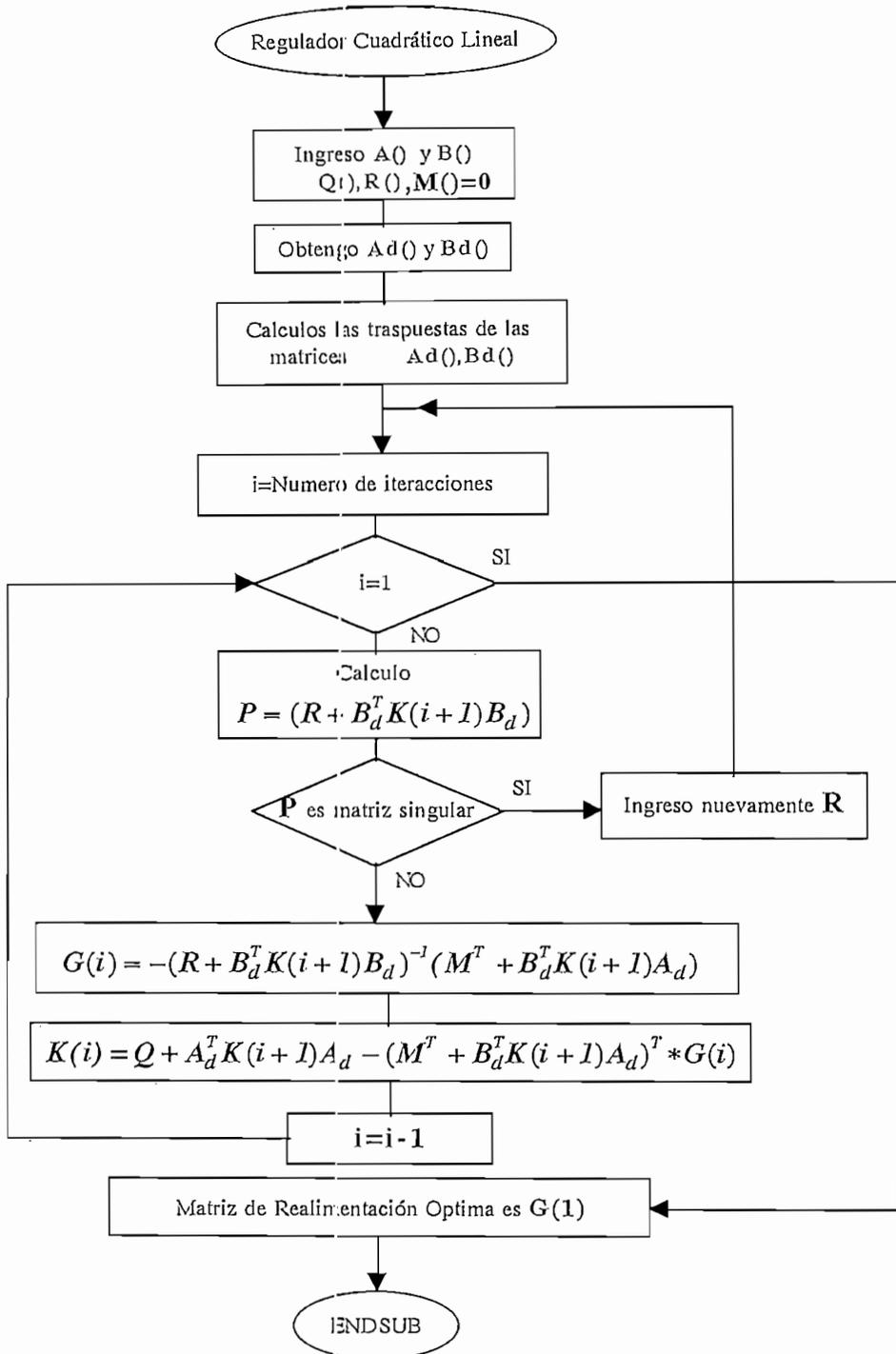


Figura. 4.21. Algoritmo del Cálculo del Regulador Cuadrático Lineal.

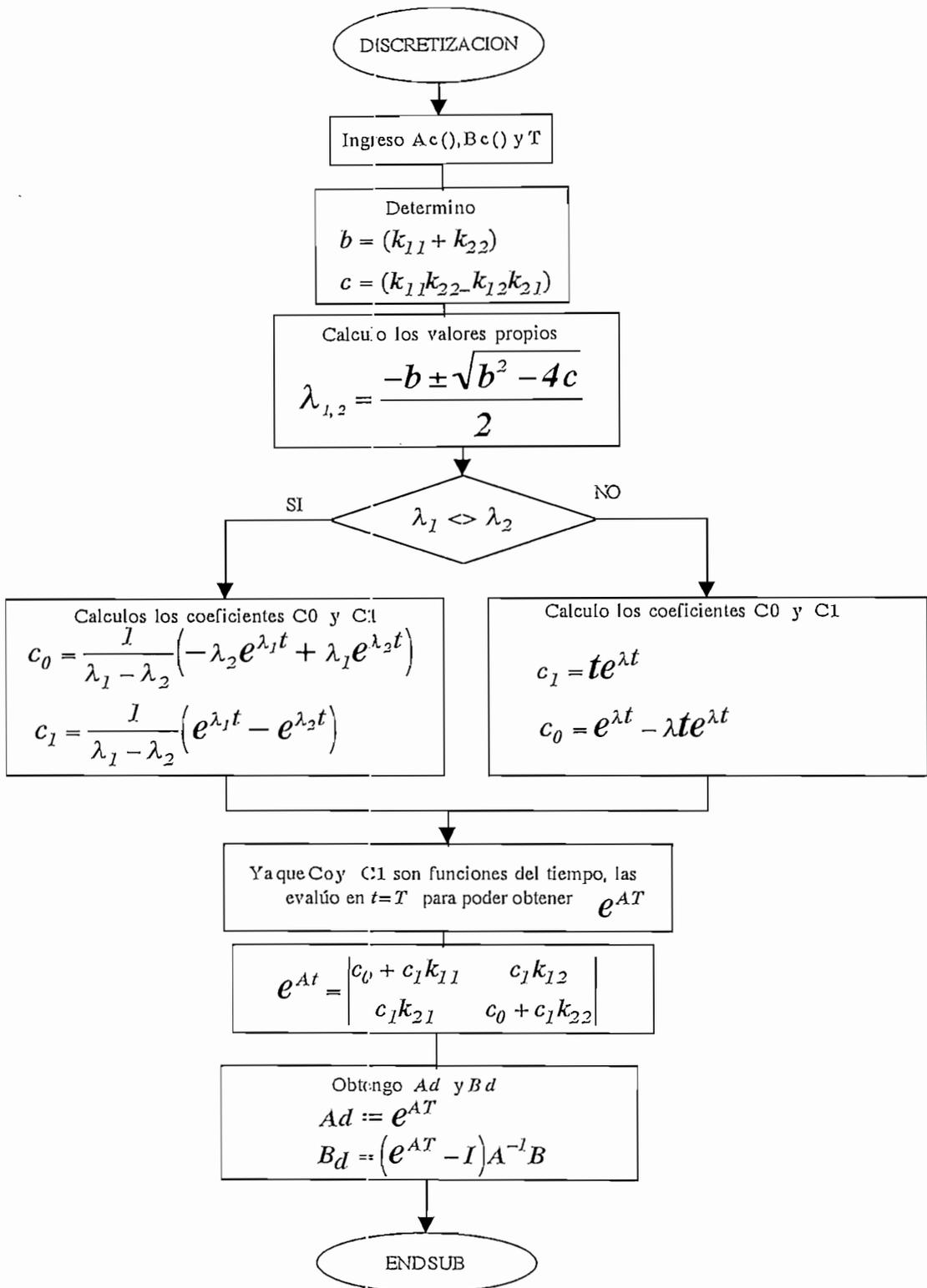


Figura 4.22. Algoritmo de discretización mediante el Teorema de Cayley -Hamilton

#### 4.2.3. MACROS UTILIZADOS EN EL PROGRAMA CC.

El presente literal tiene por objeto mostrar los macros que utilizamos para la comprobación de los resultados obtenidos mediante el programa desarrollado, los mismos que se encargan de obtener el Regulador Cuadrático Lineal y la respuesta en el tiempo de los niveles de tanques.

Los macros **REGULA1.MAC** Y **REGULA2.MAC** se encargan de intercambiar los valores de las matrices de ponderación **Q** y **R** previamente ingresados para el respectivo cálculo de RQL.

@REGULA1

@REGULA2

REGULA1.MAC	REGULA2.MAC
<pre> cls state echo,MATRIZ DE PONDERACION Q pdisplay,P11 echo,MATRIZ DE PONDERACION R pdisplay,P22 LQR,1,P1,P11,P22,P33,1 echo,MATRIZ DE REALIMENTACION DE ESTADO pdisplay,p33 quit </pre>	<pre> cls state echo,MATRIZ DE PONDERACION Q pdisplay,P22 echo,MATRIZ DE PONDERACION R pdisplay,P11 lqr,1,P1,P22,P11,P33,1 echo,MATRIZ DE REALIMENTACION DE ESTADO pdisplay,p33 quit </pre>

El macro **SIMULA.MAC** se encarga de hacer la simulación continua del sistema, para lo cual el momento de la ejecución del mismo hay que introducir el parámetro **&1** que corresponde al tiempo de simulación.

Ej. @Simula,100

**SIMULA.MAC**

CLS

STATE

feedback,4,P1,P33,P2

SIMULATION,P2,5,ENTRADA,2,P4,&1,1,1

PLOT,P2,ALL,AUTO,REGULADOR-CUADRATICO-LINEAL,1,0

CC>@Macro,&1,&2,.....,&n Ejecuta el macro con parámetros externos

#### 4.2.4. RESULTADOS OBTENIDOS DE LA SIMULACIÓN DE LOS TANQUES ACOPLADOS.

Como habíamos mencionado para determinar el punto de operación, es necesario considerar las características que exigirá más tarde el control, en cuanto a voltaje de control de los manejadores de las bombas, así como también tener un rango considerable alrededor del punto de operación de forma que sea factible de variar y de esta manera poder observar un control más efectivo sobre el nivel del líquido en los Tanques.

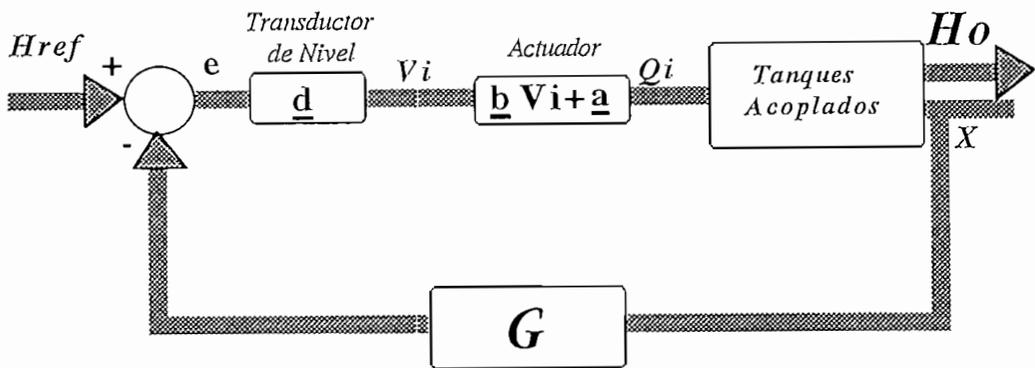


Figura 4.23. Diagrama de bloques no lineal.

Ahora vamos a considerar la figura 4.23, que muestra la representación no lineal del sistema de Tanques Acoplados, aquí vamos a considerar las ecuaciones (3-8,9), las cuales representan el comportamiento no lineal del sistema, y que en equilibrio puede ser representado por las ecuaciones:

$$Q_{i1} = Cd_1 * a_1 \sqrt{2gH_1} + Cd_{12} * a_{12} \sqrt{2g(H_1 - H_2)} \quad (4-22)$$

$$Q_{i2} = Cd_2 * a_2 \sqrt{2gH_2} - Cd_{12} * a_{12} \sqrt{2g(H_1 - H_2)} \quad (4-23)$$

Como podemos darnos cuenta dadas las condiciones iniciales  $H_{o1}$  y  $H_{o2}$ , podemos obtener los caudales necesarios para mantener dichas alturas mediante la utilización de las ecuaciones (4-22,23), con estos caudales podemos calcular el voltaje necesario para que las bombas puedan suministrar dicho caudal, y mediante la utilización de la ecuación (3-10), podemos obtener los voltajes aplicados a las bombas, así tenemos:

$$\begin{aligned} V_{i1} &= (Q_{i1} - a_1) / b_1 \\ V_{i2} &= (Q_{i2} - a_2) / b_2 \end{aligned} \quad (4-24)$$

ahora la señal de error en la figura 4.10, esta representada por:

$$\begin{aligned} e_1 &= (Vi_1 / d_{11}) = Href_1 - (G_{11} \times Ho_1 + G_{12} \times Ho_2) \\ e_2 &= (Vi_2 / d_{22}) = Href_2 - (G_{21} \times Ho_1 + G_{22} \times Ho_2) \end{aligned} \quad (4-24)$$

Finalmente de la ecuación (4-24) podemos obtener las alturas de referencia Href1 y Href2, necesarias para tener en la salida la altura inicial ingresada para la simulación. Este mismo procedimiento lo hacemos para las condiciones finales de simulación, de tal manera de ver si podemos realizar satisfactoriamente la simulación.

Sin embargo nosotros podríamos trabajar con una escala de referencia mucho mayor a 40cm (10Voltios), simplemente considerando un voltaje mayor suministrado en la referencia, pero siempre debemos considerar que puede existir una saturación de las bombas.

Como ejemplo utilizaremos las condiciones utilizada en la sección 3.6., además mediante los macros desarrollados para el programa CC, vamos a realizar la comparación con los resultados obtenidos por el programa de simulación.

CONDICIONES		PARAMETROS	
H1ini :	13	A1 :	378.3
H2ini :	18	A2 :	378.3
H1fin :	15	a1 :	ON
H2fin :	20	a2 :	ON
T(seg):	2	a12 :	0
iter :	400		

*Figura 4.24. Condiciones y Parámetros utilizados para la simulación*

Para las condiciones definidas en la figura 4.24, el programa obtuvo las matrices de la figura 4.25 que nosotros utilizaremos en el programa CC, para obtener la respuesta en el tiempo y compararlas con los gráficos obtenidas en la simulación

**TANQUES ACOPLADOS**  
**REG. CUADRAT. LINEAL**

Matrices para		la Simulación
Ac(1,1):	-.0098780	Qi1(max): 111.4
Ac(1,2):	.0018010	Qi2(max): 147
Ac(2,1):	.0018010	
Ac(2,2):	-.0086660	Qi1(ini): 72.62894
Be(1,1):	.0088530	Qi2(ini): 100.299
Be(1,2):	0	
Be(2,1):	0	Qi1(fin): 78.52158
Be(2,2):	.0122290	Qi2(fin): 105.3558

*Figura 4.25. Matrices utilizadas para la simulación.*

En la figura 4.25. además podemos ver que el programa nos muestra dependiendo de las alturas iniciales y finales ingresadas, los caudales necesarios para mantener dichas alturas, así como también los valores máximos de caudales que pueden ser suministrados por las bombas según lo obtenido en la sección 3.1; estos valores son los que nos permiten elegir adecuadamente el punto de diseño para evitar la saturación de las bombas.

**TANQUES ACOPLADOS**  
**REG. CUADRAT. LINEAL**

$(Q+) \wedge (R+)$		
Q(1,1):	4	g(1,1): .0598850
Q(1,2):	0	g(1,2): .0145410
Q(2,1):	0	g(2,1): .0303070
Q(2,2):	8	g(2,2): .2428906
R(1,1):	30	H1r(0): 32.33248
R(1,2):	0	H2r(0): 37.59996
R(2,1):	0	H1r(T): 34.24032
R(2,2):	20	H2r(T): 39.23911

*Figura 4.26. Matrices para el cálculo del Regulador Cuadrático Lineal  
Mayor peso a R()*

En la figura 4.26 nosotros tenemos los valores ingresados de las matrices  $Q()$ ,  $R()$ , así como también la matriz de realimentación óptima obtenida por el programa, la misma que se asemeja al valor obtenido por el programa CC, y que se muestra en la figura 4.27. Además nosotros obtenemos los valores de referencia necesarios para poder obtener a la salida los valores requeridos por la figura 4.24.

```

CC>state
STATE>MATRIZ DE PONDERACION Q
STATE>pdisplay, P11
  1:      4          0
  2:      0          8

STATE>MATRIZ DE PONDERACION R
STATE>pdisplay, P22
P22:
  1:      30          0
  2:      0          20

STATE>LQR, 1, P1, P11, P22, P33, 1
LQR poles:   -9.37263E-03
             -1.271161E-02

STATE>MATRIZ DE REALIMENTACION DE ESTADO
STATE>pdisplay, p33
P33:
  1: 6.049795D-02  1.460404D-02
  2: 3.025971D-02  2.456985D-01

```

Figura 4.27. Matriz de realimentación óptima obtenida mediante el programa CC

En la figura 4.28 se muestran los resultados obtenidos por el programa, al aplicar el Regulador Cuadrático Lineal pero dándole mayor peso a la matriz que pondera al control ( $\mathbf{R}$ ), la misma que se asemeja a los valores obtenidos en la figura 3.12, pero sumados a los valores del punto operación definido; aquí podemos observar además que la salida tiende a seguir a la referencia pero con menor rapidez, podemos por lo tanto concluir que este caso se asemeja a un problema de seguimiento.

En la figura 4.29 vemos el comportamiento de las bombas, las mismas que suministran el caudal necesario para mantenernos alrededor del punto de operación definido.

En la figura 4.30 nosotros tenemos el cálculo del regulador cuadrático lineal, pero dándole mayor peso a la matriz que pondera al los estados ( $\mathbf{Q}$ ), dicho valor se asemeja al obtenido por el programa CC y que se muestra en la figura 4.31; en la figura 4.30 podemos observar además que los valores de referencia necesarios para mantener estos niveles a la salida, son mayores que el rango de medida de los tanques, sin embargo como lo habíamos mencionado anteriormente, la señal de referencia que en realidad es una señal de voltaje de 0-10V (0-40cm), podría ser incrementada para obtener los niveles deseados.

Si consideramos lo antes mencionado obtenemos la figura 4.33, la cual se asemeja a la obtenida en la figura 4.32 mediante el programa CC, aquí podemos observar que el sistema tiende a estabilizarse más rápidamente, pero difiriendo en gran porcentaje de la referencia, es decir manteniendo un mayor error de estado estable, por lo tanto obtenemos en este caso resultados aceptables de regulación.

De lo expuesto anteriormente podemos concluir que dependiendo de los requerimientos que se desee en el sistema, debemos dar mayor peso a la matriz ya sea la que pondera a los estados o al control, teniéndose un compromiso entre ellos.

En las figuras 4.35 a 4.38, tenemos algunas pruebas realizadas para altos valores de la matriz  $\mathbf{R}()$ , en las cuales se puede observar que los niveles de los tanques se tienden a comportar como si el sistema fuera en lazo abierto, y esto se debe al bajo valor de tiene la matriz de realimentación Óptima para altos valores de la matriz que pondera al control. Además se incluye una simulación figuras 4.39 y 4.40, en la que se tiene mayor peso a la matriz  $\mathbf{Q}()$ , pero los niveles iniciales son tales que la referencia del sistema se mantiene dentro del rango de medida.

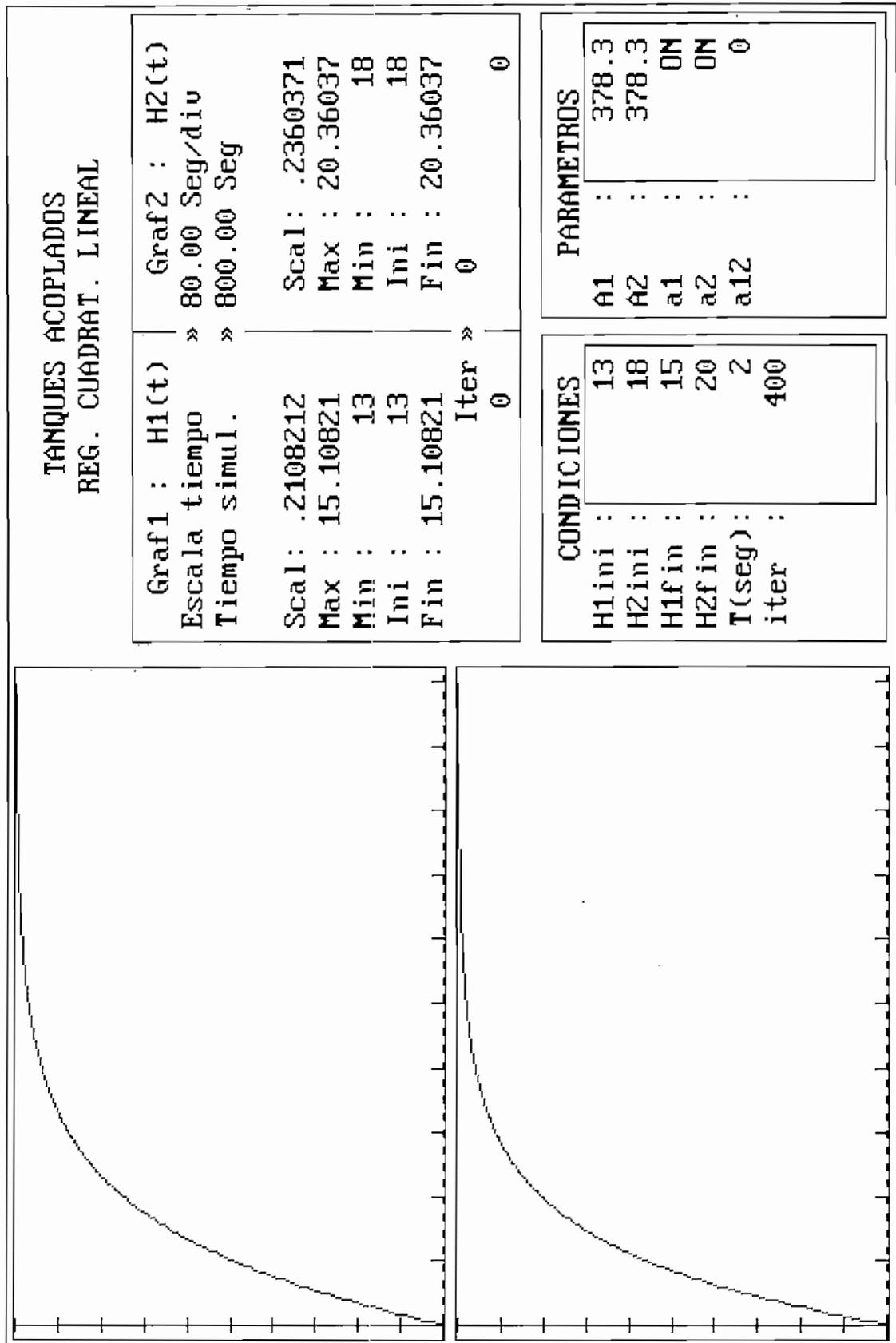


Figura 4.28. Niveles de los Tanques obtenidos en la simulación.

Mayor peso a R()

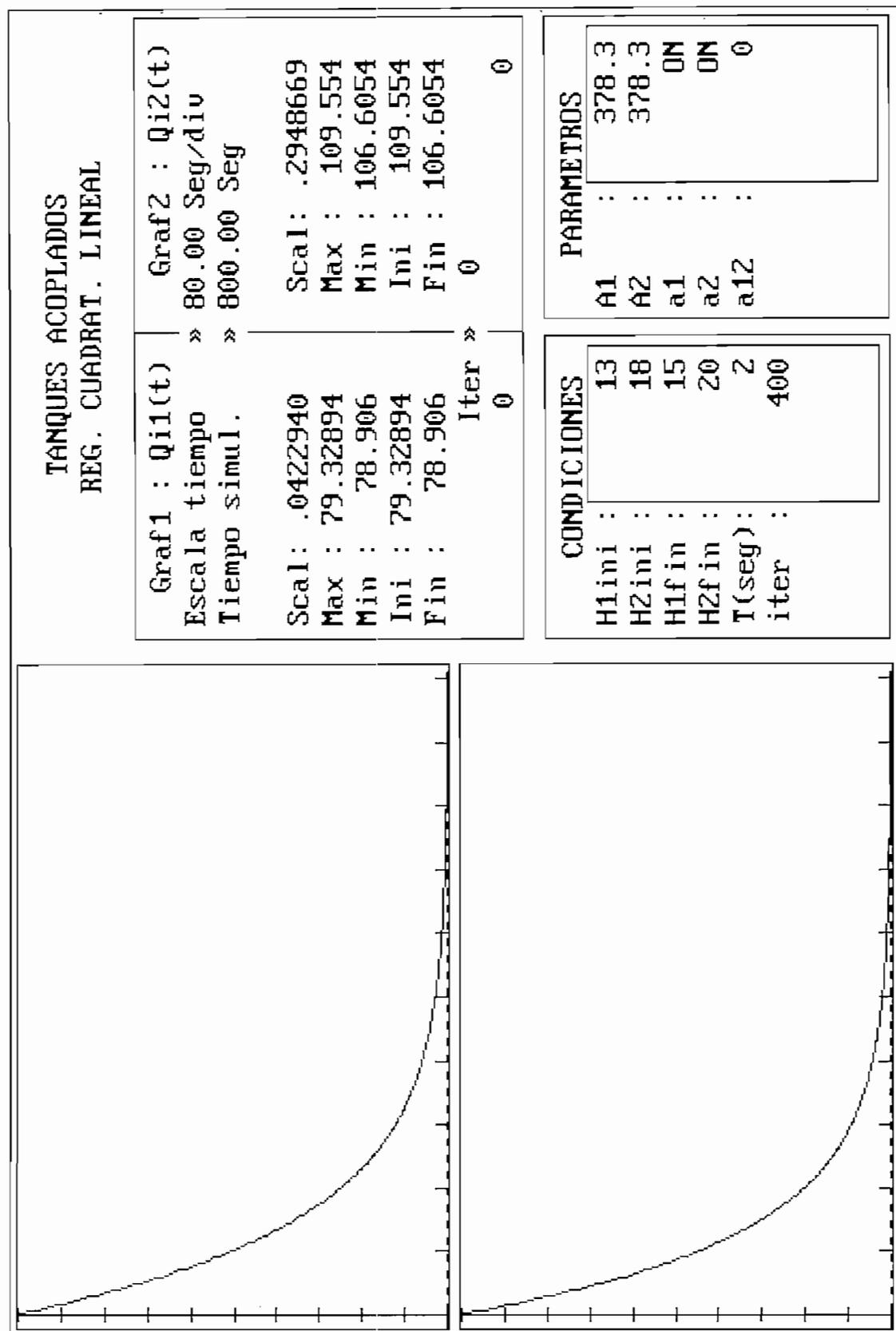


Figura 4.29. Caudales suministrados por las bombas.

Mayor peso a R()

TANQUES ACOPLADOS  
REG. CUADRAT. LINEAL

$(Q+)$	$\wedge$	$(R+)$	
Q(1,1):»	15		g(1,1): 1.101589
Q(1,2):	0		g(1,2): .0822686
Q(2,1):	0		g(2,1): .1128196
Q(2,2):	12		g(2,2): 1.143377
R(1,1):	4		H1r(0): 47.09372
R(1,2):	0		H2r(0): 54.88138
R(2,1):	0		H1r(T): 51.22042
R(2,2):	4		H2r(T): 58.48652

*Figura 4.30. Matrices para el Cálculo del Regulador Cuadrático Lineal  
Mayor peso a Q().*

```

CC>state
STATE>MATRIZ DE PONDERACION Q
STATE>pdisplay,P11
  1:   15           0
  2:    0          12

STATE>MATRIZ DE PONDERACION R
STATE>pdisplay,P22

P22:
  1:    4           0
  2:    0           4

STATE>LQR,1,P1,P11,P22,P33,1

LQR poles:   -1.969645E-02
             -2.310337E-02

STATE>MATRIZ DE REALIMENTACION DE ESTADO
STATE>pdisplay,p33

P33:
  1: 1.123770D+00  8.292729D-02
  2: 1.145508D-01  1.169930D+00

```

Figura 4.31. Regulador Cuadrático Lineal obtenido mediante el CC

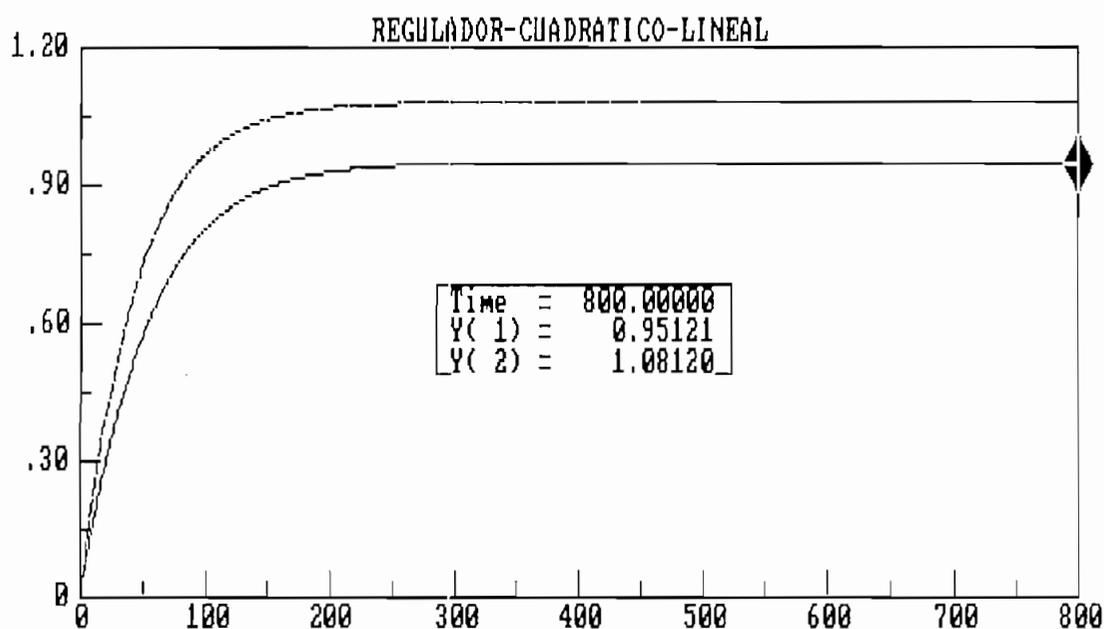
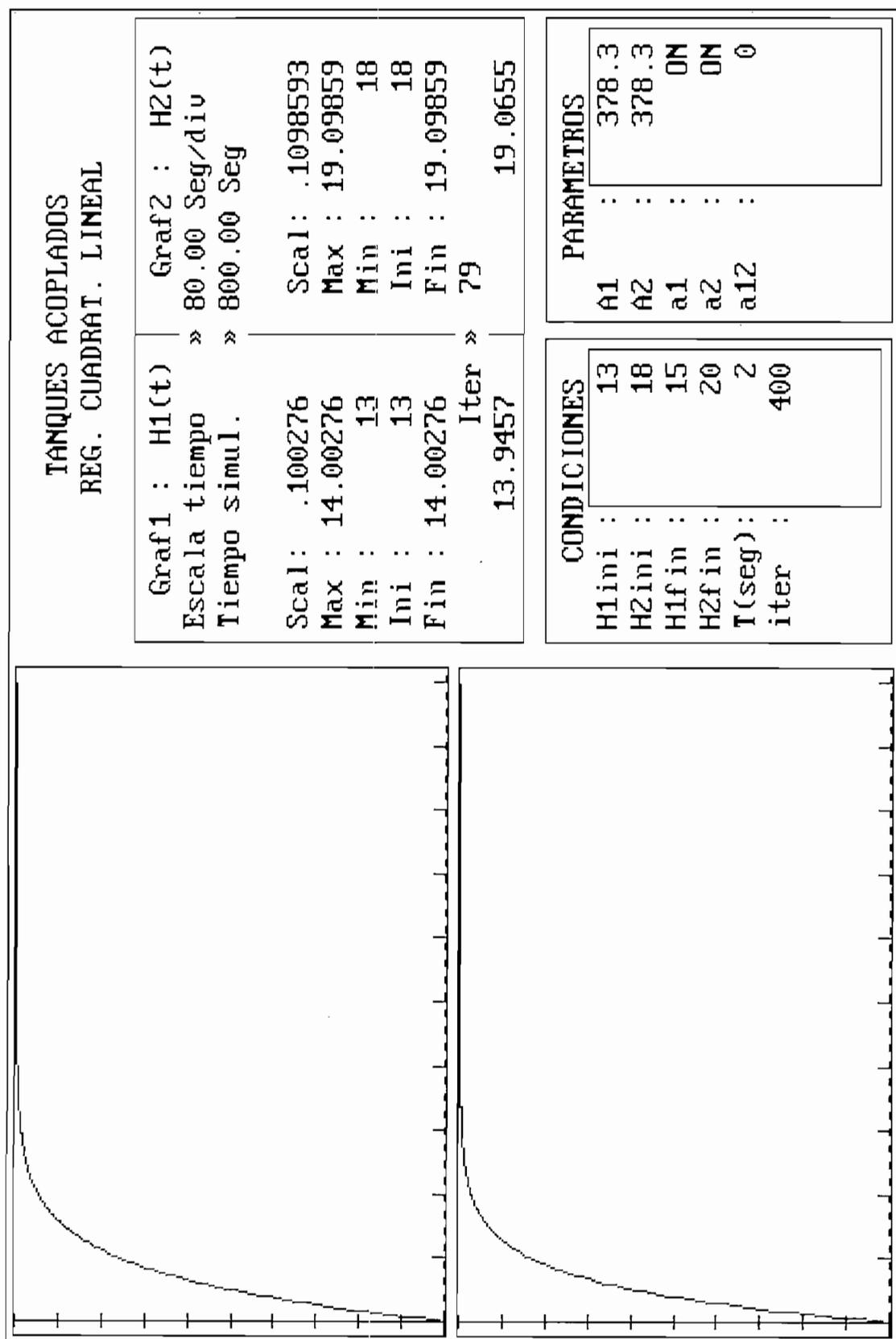


Figura 4.32. Respuesta obtenida mediante el programa CC  
Mayor peso a  $Q()$



*Figura 4.33. Niveles de los Tanques obtenidos en la simulación.*  
*Mayor peso a Q0*

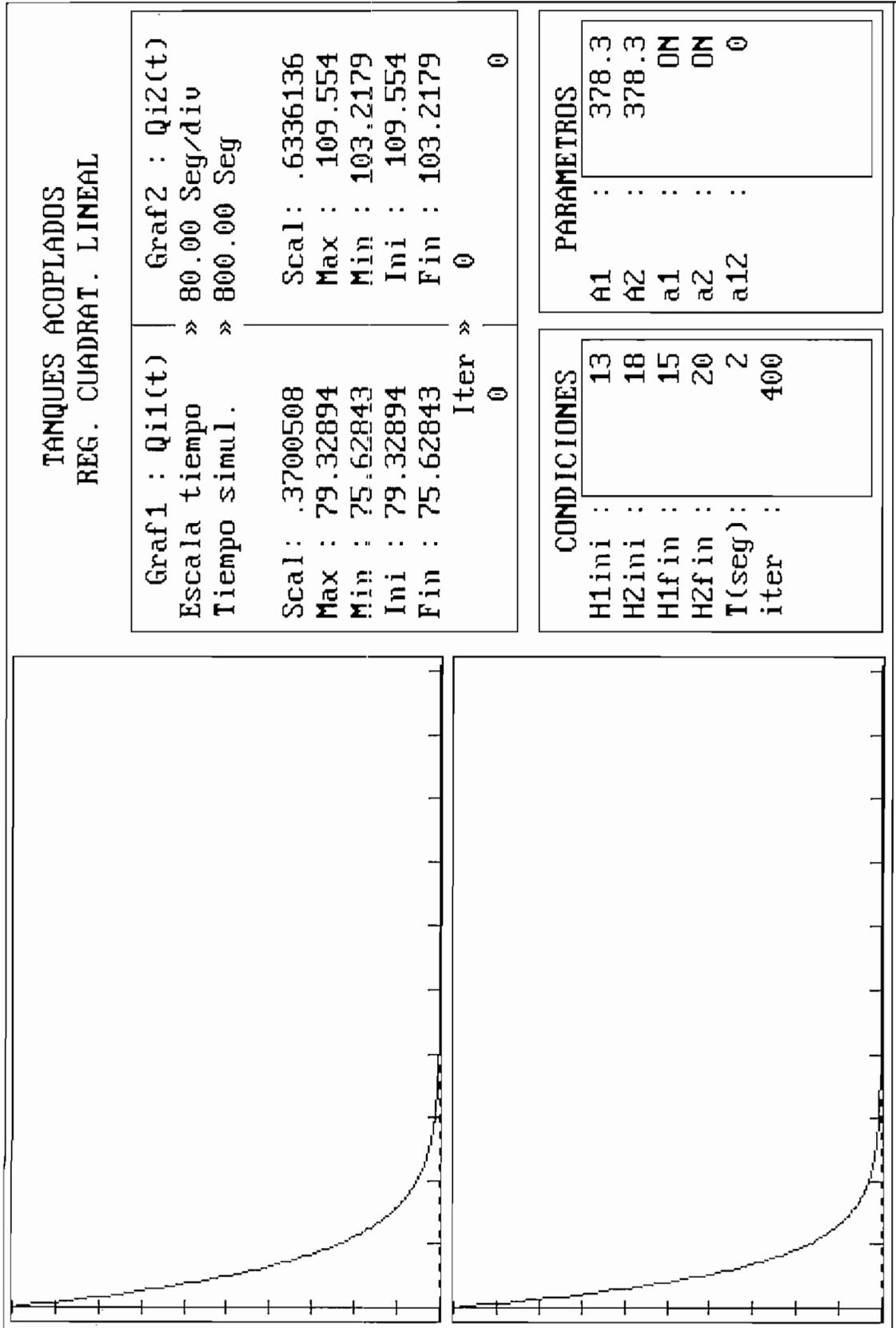
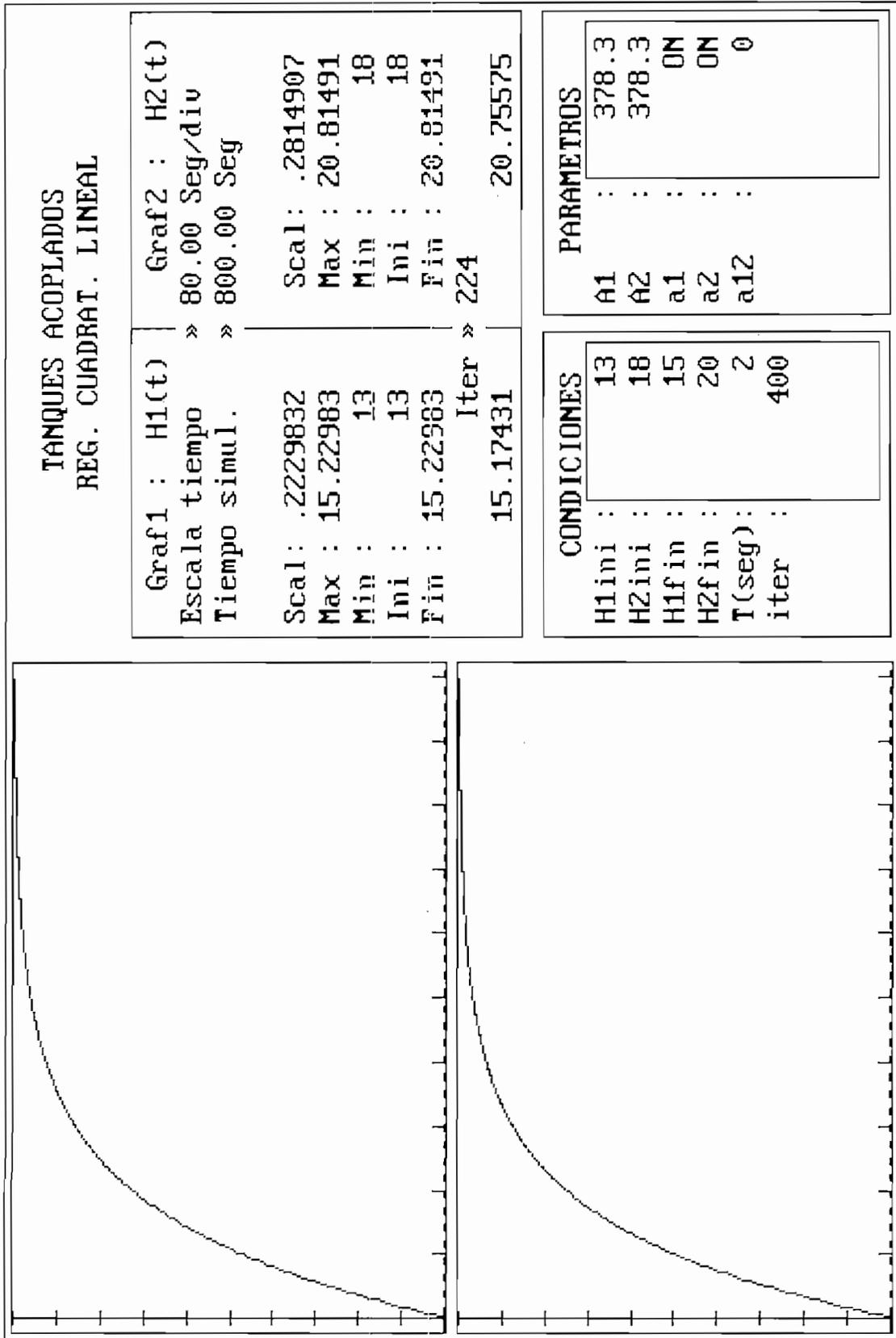


Fig. 4.34. Caudales suministrados por las bombas de agua.  
Mayor peso a Q()

TANQUES ACOPLADOS  
REG. CUADRAT. LINEAL

$(Q+)$	$\wedge$	$(R+)$		
Q(1,1):		4	g(1,1):	.0366510
Q(1,2):		0	g(1,2):	.0102163
Q(2,1):		0	g(2,1):	.0141960
Q(2,2):		8	g(2,2):	.1065461
R(1,1):		50	H1r(0):	31.95259
R(1,2):		0	H2r(0):	34.93632
R(2,1):		0	H1r(T):	33.80532
R(2,2):		50	H2r(T):	36.27056

*Figura 4.35. Regulador Cuadrático Lineal  
Mayor Peso a R()*



*Fig. 4.36. Niveles de los Tanques obtenidos en la simulación.  
Mayor peso a R()*

TANQUES ACOPLADOS  
REG. CUADRAT. LINEAL

$(Q+) \wedge (R+)$		
Q(1,1):»	4	g(1,1): .0012540
Q(1,2):	0	g(1,2): .0003930
Q(2,1):	0	g(2,1): .0003270
Q(2,2):	8	g(2,2): .0023020
R(1,1):	1500	H1r(0): 31.31561
R(1,2):	0	H2r(0): 32.87964
R(2,1):	0	H1r(T): 33.07789
R(2,2):	2500	H2r(T): 33.97766

*Figura. 4.37 Regulador Cuadrático*

*Mayor peso a R()*

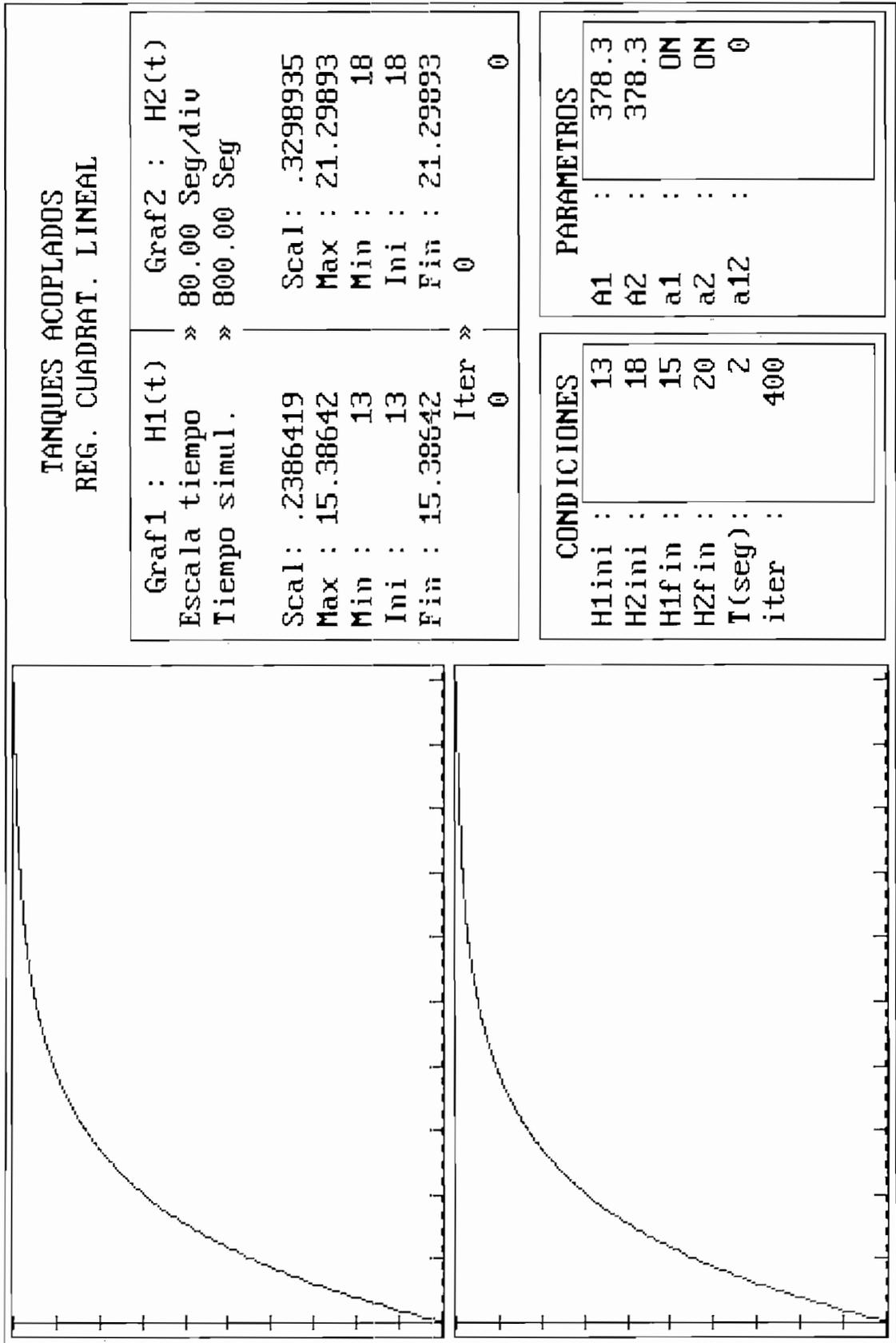


Fig. 4.38. Niveles de los Tanques obtenidos en la simulación.

Mayor peso a R()

TANQUES ACOPLADOS  
REG. CUADRAT. LINEAL

CONDICIONES		PARAMETROS	
H1ini :	7	A1 :	378.3
H2ini :	6	A2 :	378.3
H1fin :	9	a1 :	ON
H2fin :	8	a2 :	ON
T(seg):	2	a12 :	0
iter :	400		

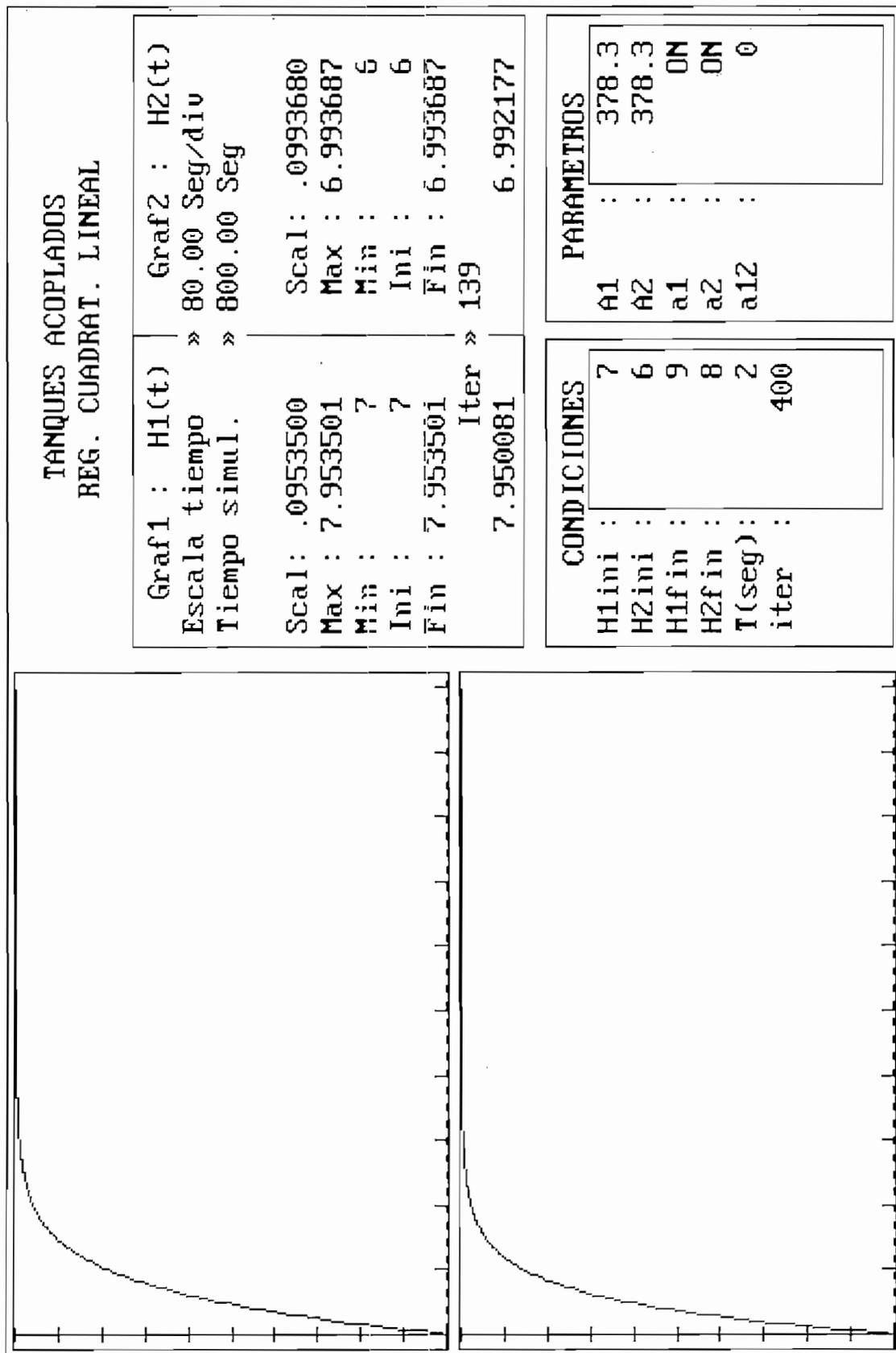
  

Matrices para	la Simulación
Ac(1,1): -.0150360	Qi1(max): 111.4
Ac(1,2): .0040280	Qi2(max): 147
Ac(2,1): .0040280	
Ac(2,2): -.0159180	Qi1(ini): 61.3451
Be(1,1): .0088530	Qi2(ini): 50.92405
Be(1,2): 0	
Be(2,1): 0	Qi1(fin): 69.15065
Be(2,2): .0122290	Qi2(fin): 59.2736

(Q+) ^ (R+)		
Q(1,1):	15	g(1,1): .8733791
Q(1,2):	0	g(1,2): .1237702
Q(2,1):	0	g(2,1): .1697344
Q(2,2):	12	g(2,2): .8643058
R(1,1):	4	H1r(0): 34.78019
R(1,2):	0	H2r(0): 28.53801
R(2,1):	0	H1r(T): 39.1045
R(2,2):	4	H2r(T): 32.41042

*Fig. 4.39. Condiciones utilizadas para la simulación  
Mayor peso a Q()*



*Fig. 4.40. Niveles de los Tanques obtenidos en la simulación.*  
*Mayor peso a Q()*

*CAPITULO V*

*CONCLUSIONES Y RECOMENDACIONES*

## 5.1. CONCLUSIONES.

Se cumplió con el objetivo de aplicar algunos conceptos y técnicas de control univariable y multivariable, logrando además que la persona interesada en utilizar los programas desarrollados se dedique a buscar alternativas de control, sin que tenga que preocuparse por el montaje y diseño de la planta.

Si bien los programas fueron desarrollados en Quick Basic, los mismos fueron hechos tomando en cuenta todas las facilidades para que el usuario pueda desplazarse entre las diferentes alternativas y no se encuentre con una presentación simple.

Los programas desarrollados consta de dos partes perfectamente definidas: simulación dinámica y gráficos; en cuanto a la simulación, el software desarrollado nos permite mostrar en la pantalla del computador, una simulación del comportamiento de las variables de interés, pero estas no son en tiempo real, y esto se debe a que dependiendo del computador donde se ejecute, se debería crear una rutina para los retardos introducidos para simular el movimiento. Además como vimos en el caso de los tanques acoplados, el tiempo real del proceso está en el orden de los 300 a 800 Seg, lo que implicaría estar observando la pantalla durante ese tiempo, razón por la cual también introducimos retardos pequeños para poder visualizar mejor el experimento. En cuanto a la visualización de los gráficos, los programas están hechos de tal manera que nosotros podamos visualizar en la pantalla dos de las variables de interés del proceso, de tal manera de poder realizar análisis y comparaciones entre ellas.

El realizar una simulación en computador, nos permite analizar experimentos que tienen limitaciones tanto de diseño como de construcción; ya que analizar una función de transferencia y diseñar el controlador, desde el punto de vista analítico no presenta mayores inconvenientes; pero como el proceso puede tener limitaciones propias en cuanto a valores que pueden tomar ciertas variables o parámetros del proceso, estas limitarán el diseño de dicho controlador. Así vemos en el caso del Motomatic, que existe una zona muerta en el sensor de posición, lo cual limitará el rango de las variables del compensador.

De igual manera al considerar los Tanques Acoplados tuvimos que tomar en cuenta limitaciones que tenían las bombas y los voltajes de control que podían ser aplicados, así como también determinar un punto de operación alrededor del cual podamos tener un amplio rango de control.

Nosotros hemos realizado el análisis del comportamiento de dos sistemas, desde el punto de vista discreto, ante lo cual pudimos concluir al analizar las pruebas realizadas con el programa, que lo que en realidad obtenemos no es la señal  $f(t)$ , sino la señal  $f(KT)$  la misma que es igual a  $f(t)$  pero únicamente en los instantes de muestreo, razón por la cual es necesario realizar una acertada elección del período de muestreo.

Al obtener la transformada Z de una función  $G(s)$  por métodos aproximativos, y no por una transformación directa, puede darse el caso que por discretización un sistema estable pueda transformarse en inestable, dependiendo del período de muestreo utilizado.

El programa CC, que utilizamos para realizar la comprobación de los resultados par cada uno de los programas de simulación desarrollados, fue de gran ayuda, ya que así pudimos determinar que el período de muestreo utilizado en un determinado momento en el Motomatic, fue tal que siendo estable, el mismo se comporte como inestable. Además gracias a todas las opciones que tiene para sistemas multivariables, pudimos obtener el Regulador Cuadrático Lineal y comprobar con el obtenido por el programa desarrollado.

Mediante el programa podemos determinar el valor de la matriz A, para ciertos niveles de los Tanques Acoplados, y por lo tanto podemos determinar que tanto podemos alejarnos de dicho punto de operación específico y seguir considerando al sistema lineal.

Se determinó la utilización solamente del Regulador Cuadrático Lineal como la técnica de compensación a emplearse en el caso de los Tanques Acoplados, por la no linealidad existente en el modelo matemático que representa a los mismos. Pero no debemos perder de vista que esta técnica solamente es aplicable para sistemas lineales, o en este caso alrededor de un punto de operación determinado. Sin embargo el resultado obtenido podemos aplicarlo para un rango mayor, ya que el

Regulador Cuadrático Lineal minimiza un índice de comportamiento cuadrático, y como al realizar la linealización aparecen términos de orden cuadrático y superiores, y los primeros deben verse minimizados.

En el caso del Motomatic, nosotros compensamos al sistema definido en el plano S con técnicas conocidas, y luego encontramos la transformada Z, lo cual es mucho más fácil que hacer todo el análisis con el modelo discreto, es decir en el plano Z.

Desde el punto de vista de la implementación en el computador es más fácil, la implementación de un sistema mediante retardos unitarios, que determinar la respuesta en el tiempo mediante la transformada inversa de  $F(S)$ .

La técnica de obtención del Regulador Cuadrático Lineal, se basa en la asignación de tres matrices de ponderación: Q, R, y M. La matriz M pondera la relación entre los estados y los controles por lo que se utiliza como matriz igual a cero a fin de disminuir al iteración entrada - salida.

Cuando la ponderación especificada por Q() es alta, se tiene un excelente regulador óptimo, lo cual permite que los estados adquieran su estado estable rápidamente y se disponga de un alto rechazo a las perturbaciones; sin embargo se tiene la desventaja que el valor estable difiere en gran porcentaje del valor de referencia especificado. También pudimos observar que si el valor de Q() es demasiado grande el sistema tiende a comportarse como si estuviera en lazo abierto.

Por el contrario, si la ponderación asignada al control R() es alta, se puede obtener el valor estable de los estados igual al valor de referencia; sin embargo el estado estable se lo adquiere en más tiempo que para el caso anterior.

Como el cálculo del Regulador Cuadrático Lineal es un proceso recursivo, es importante tener en cuenta que mientras mayor número de iteraciones, el valor de la matriz de regulación óptima cumplirá la función de minimizar el índice de funcionamiento cuadrático.

## 5.1. RECOMENDACIONES.

Si embargo que el programa tiene todas las facilidades de trabajo de un ambiente Windows, sería conveniente realizar la traducción del programa a una versión más avanzada, como podría ser VisualBasic, Visual C, u otras. Sin embargo esto podría quedar como un trabajo posterior para la persona interesada en realizarlo.

Es necesario aclarar que la persona interesada en utilizar el programa deberá tener una idea clara de los rangos de las variables a ingresar para tratar de evitar conflictos en la ejecución del mismo, ya que si bien el programa tiene rutina de chequeo de valores, podría darse de que no se haya considerado algún caso específico.

Al analizar el Motomatic, si bien consideramos la mayor cantidad de variables posibles, para que la simulación sea más exacta, debería agregarse el parámetro de la corriente en el motor, ya que si bien la respuesta del sistema está dentro de los parámetros de diseño, a lo mejor debido a la velocidad de la respuesta, el sistema está requiriendo más corriente que la que podría suministrar la fuente de poder, y por lo tanto la respuesta podría ser diferente a la esperada.

Se podría además plantear otras técnicas de compensación para el sistema de Tanques Acoplados como el desacoplamiento por realimentación de salida, técnicas en el dominio de la frecuencia, etc.

Los programas desarrollados no permiten realizar una impresión de los resultados obtenidos, sin embargo mediante la utilización de utilitarios de captura de la pantalla, como los que dispone el Word Perfect o el Word, se podría proceder a la captura de dichos gráficos y posteriormente imprimirlos, lo cual debe ser realizado por el diseñador.

## *APENDICES*

## APENDICE A.

### MANUAL DE USO.

La presente sección tiene por objeto dar los instructivos necesarios para poder utilizar el programa desarrollado, de manera que el diseñador aproveche al máximo de los beneficios que presta el mismo.

Como primera parte deberemos tener el claro que la persona que va a trabajar con los sistemas deberán tener en claro las magnitudes de los parámetros a ingresar, de manera de evitar posibles problemas al realizar la simulación. Sin embargo como ya mencionamos en el capítulo IV, el programa se hizo con la mayor cantidad de rutinas para el chequeo de los datos que podían ser ingresados; pese a lo cual podría eventualmente haber alguna opción que no haya sido considerada.

El programa está desarrollado en dos módulos en forma de archivos ejecutables: **MOTOR.EXE**, **TANQUES.EXE**, esto se hizo de esta manera debido a la cantidad grande de memoria que requería tener en un solo módulo todo. Además los programas fueron compilados en Quick Basic 4.5, para ser ejecutados en cualquier computador con **DOS** y monitor **VGA**.

### MANUAL PARA LA SIMULACION DEL MOTOMATIC MCL-100

El programa de simulación del Motomatic está hecho de tal manera de que el usuario del mismo tenga en la misma ventana la mayor cantidad de información posible, en cuanto a los parámetros y condicione de la simulación, así como las gráficas obtenidas y la simulación propiamente dicha.

En la Fig. A1 se muestra la ventana principal para la simulación del Motomatic, en la cual debemos considerar el ingreso inicial tanto de parámetros, como condiciones para la simulación, sin los cuales no es posible ejecutar ninguno de las otras opciones. Aquí es necesario anotar que como el máximo valor del ángulo  $\theta$  (*máx*) que podemos alcanzar depende del coeficiente de reducción **N** y **R1**, es conveniente ingresar primeramente este valor es decir ejecutar la opción **F4**, ya que al realizar esto automáticamente se actualiza en valor de dicho ángulo para de esta manera poder tener un mayor rango de valores en las condiciones iniciales y finales de los ángulos.

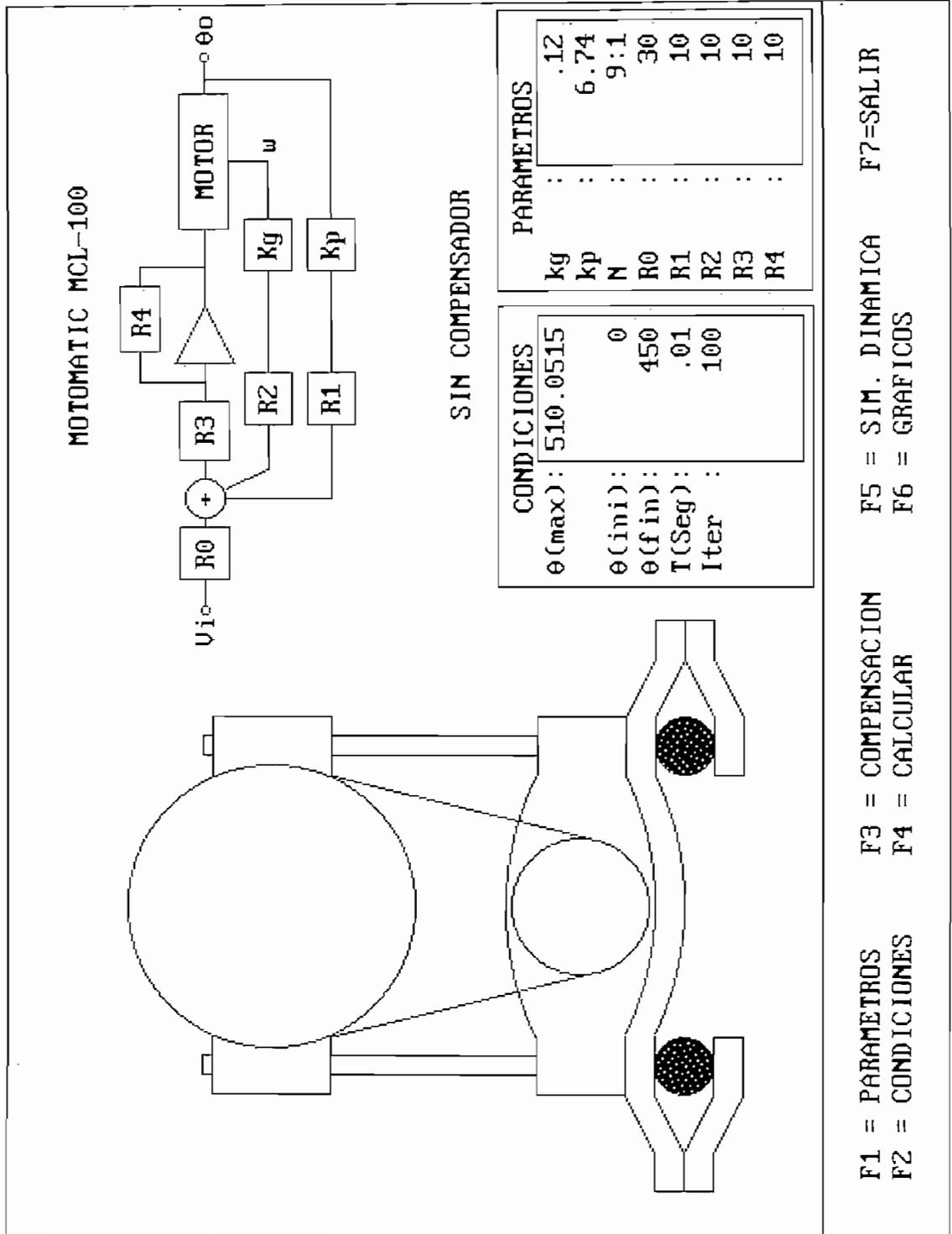


Fig. A1 VENTANA PRINCIPAL DEL PROGRAMA DE SIMULACION DEL MOTOMATIC MCL-100

En la Fig. A2 se muestra la ventana en el cual nosotros podemos visualizar dos gráficos simultáneamente, con todos los valores críticos de los mismos. Además es necesario anotar que nosotros podemos realizar el cambio de la variable a graficar simplemente con el uso de las Flechas  $\left[ \uparrow \right]$   $\left[ \downarrow \right]$ .

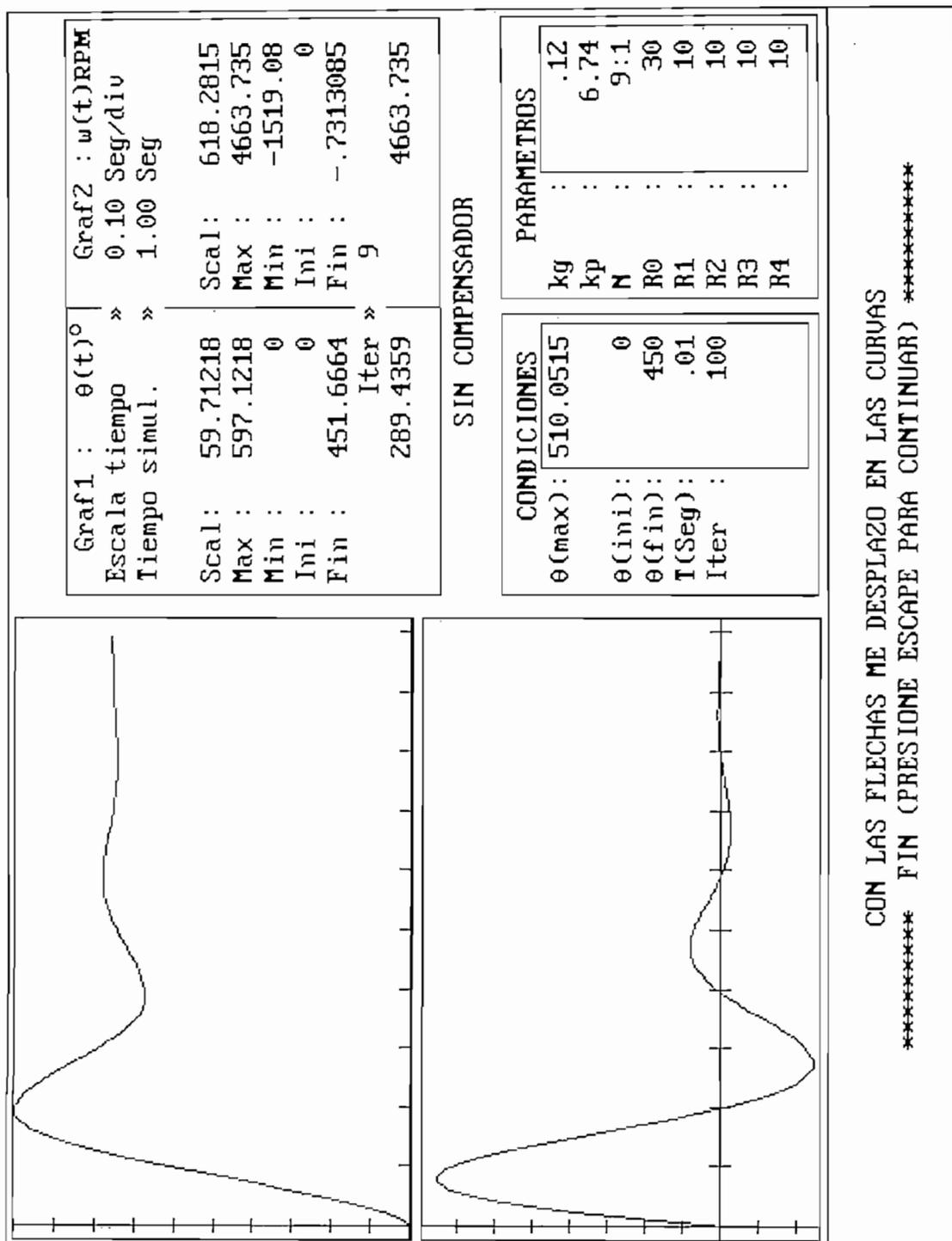


Fig. A2 VENTANA DE PRESENTACION DEL GRAFICOS DE LA SIMULACION DEL  
MOTOMATIC MCL-100

En la Fig. A3 se muestra la ventana donde se realiza la simulación dinámica en la cual se observa como varían tanto el ángulo del primario, como en el del secundario del reductor de velocidad. Para poder detener la ejecución de la simulación si fuera el caso, hay que presionar la tecla ESC.

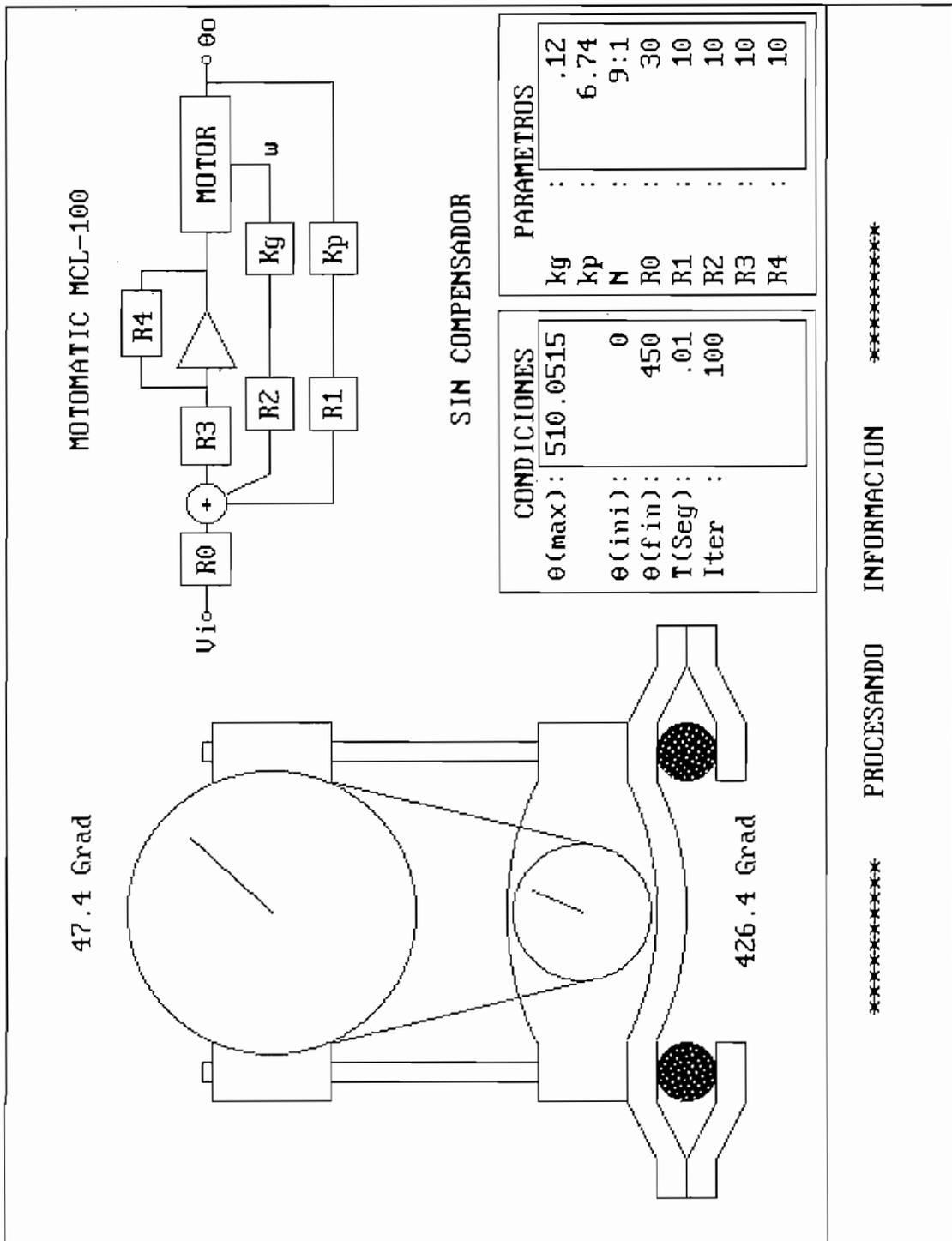


Fig. A3 VENTANA DONDE SE MUESTRA LA SIMULACION DINAMICA DEL

MOTOMATIC MCL-100

En la figura A4 se puede observar la ventana de dialogo para la eleccion del compensador a utilizar, y esta además se muestra en la parte superior de las ventanas de ingreso de los parámetros y condiciones iniciales.

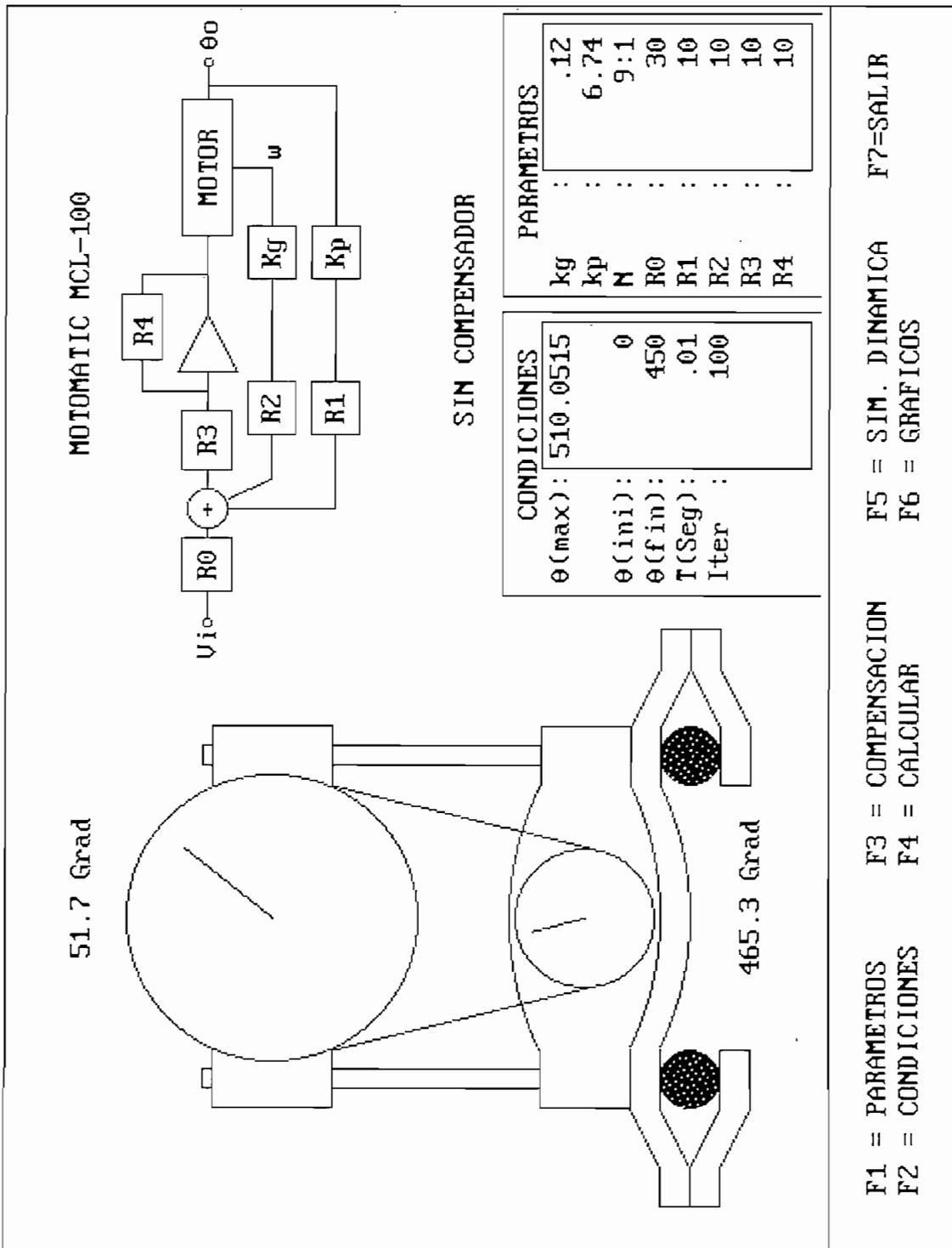


Fig. A4 VENTANA DONDE SE REALIZA LA ELECCION DEL COMPENSADOR A UTILIZAR EN EL MOTOMATIC MCL-100

## MANUAL PARA LA SIMULACION DE LOS TANQUES ACOPLADOS

El programa de simulación de los Tanques Acoplados está hecho de igual manera que el del Motomatic, es decir el usuario puede tener en la misma ventana la mayor cantidad de información posible, en cuanto a los parámetros y condiciones de la simulación, así como las gráficas obtenidas .

El programa está hecho de tal manera que no se pueda aplicar ningún comando equivocadamente, ya que siempre se realiza el chequeo mediante la utilización de parámetros internos del programas.

En la Fig. A5 se muestra la ventana principal para la simulación de los Tanques Acoplados, en la cual hay que considerar el ingreso inicial tanto de parámetros, como el orificio intertanques a utilizarse, o si están abiertas o no las válvulas de desfogue. Además aquí se ingresa las condiciones iniciales para la simulación, pero cabe recordar que estas condiciones no son de la señal de referencia, sino del nivel real de los Tanques. También podemos observar en esta figura se muestran tanto las condiciones iniciales, como finales de los caudales que deberían ser suministrados por la bombas para mantener dichos niveles del líquido; para de esta manera determinar un punto de operación adecuado y ver si no saturamos las bombas.

En la Figura A6 se muestra el ingreso de las matrices que van a permitirnos el cálculo del Regulador Cuadrático Lineal; además en esta figura se muestra los valores que necesitaríamos dar a la señal de referencia para poder obtener los niveles deseados. Como ya habíamos mencionado en la sección 4.4, cuando nosotros damos mayor peso a la matriz de ponderación del control, se da el caso de que la señal de referencia requerida sea superior al rango del nivel de referencia.

En la figura A7 se muestra la ventana de visualización de los gráficos provenientes de la simulación, aquí nos podemos ir moviendo sobre las curvas de tal manera de observar el valor que toman las misma; para salir de esta opción es necesario presionar la tecla de ESC.

Finalmente en la Figura A8 se muestra la simulación dinámica, la cual se la hace sobre un rango del nivel de los Tanques acoplados, y esto se debe básicamente a que las variaciones del nivel de salida no deben ser muy grandes para de esta manera seguir considerando al sistema lineal.

TANQUES ACOPLADOS REG. CUADRAT. LINEAL	
Matrices para	la Simulación
Ac(1,1): -.0098780	Qi1(max): 111.4
Ac(1,2): .0018010	Qi2(max): 147
Ac(2,1): .0018010	
Ac(2,2): -.0086660	Qi1(ini): 72.62894
Be(1,1): .0088530	Qi2(ini): 100.299
Be(1,2): 0	
Be(2,1): 0	Qi1(fin): 78.52158
Be(2,2): .0122290	Qi2(fin): 105.3558
CONDICIONES	
H1ini : 13	
H2ini : 18	
H1fin : 15	
H2fin : 20	
T(seg) : 2	
iter : 400	
PARAMETROS	
A1 : 378.3	
A2 : 378.3	
a1 : ON	
a2 : ON	
a12 : 0	
F1 = PARAMETROS	F3 = R.Q.L
F2 = CONDICIONES	F4 = CALCULAR
	F5 = SIM. DINAMICA
	F6 = GRAFICOS
	F7=SALIR

Fig. A5 VENTANA PRINCIPAL DEL PROGRAMA DE SIMULACION DE LOS  
TANQUES ACOPLADOS

TANQUES ACOPLADOS REG. CUADRAT. LINEAL	
(Q+) ^ (R+)	
Q(1,1):	4
Q(1,2):	0
Q(2,1):	0
Q(2,2):	8
R(1,1):	30
R(1,2):	0
R(2,1):	0
R(2,2):	20
g(1,1):	.0598850
g(1,2):	.0145410
g(2,1):	.0303070
g(2,2):	.2428906
H1r(0):	32.33248
H2r(0):	37.59996
H1r(T):	34.24032
H2r(T):	39.23911
CONDICIONES	
H1ini :	13
H2ini :	18
H1fin :	15
H2fin :	20
T(seg) :	2
iter :	400
PARAMETROS	
A1 :	378.3
A2 :	378.3
a1 :	ON
a2 :	ON
a12 :	0

INGRESANDO COMPENSACION

Fig. A6 VENTANA DE INGRESO DE LOS PARAMETROS DE CALCULO DEL  
REGULADOR CUADRATICO LINEAL

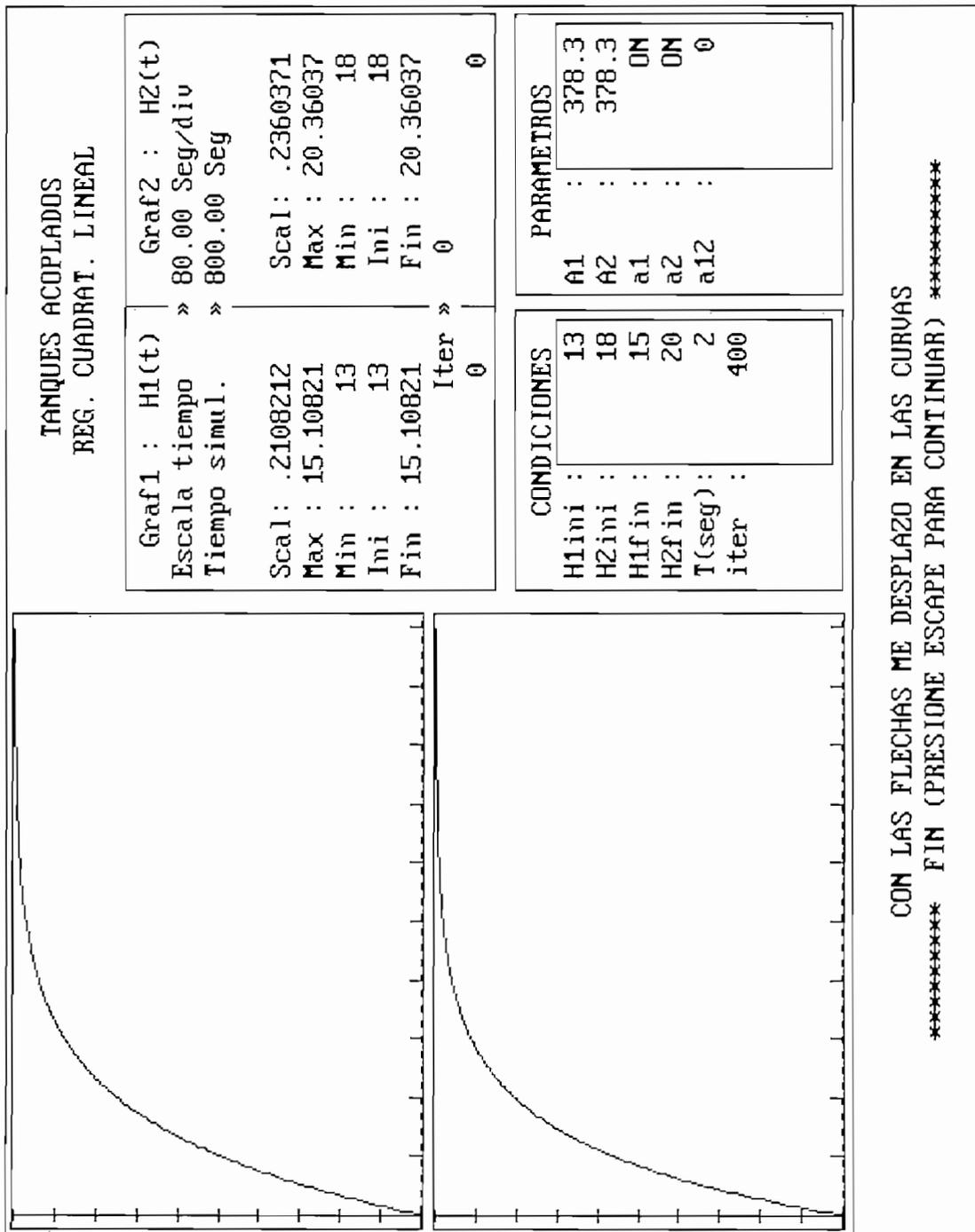


Fig. A7 VENTANA DE VISUALIZACION DE LOS GRAFICOS PROVENIENTES DE LA SIMULACION DE LOS TANQUES ACOPLADOS

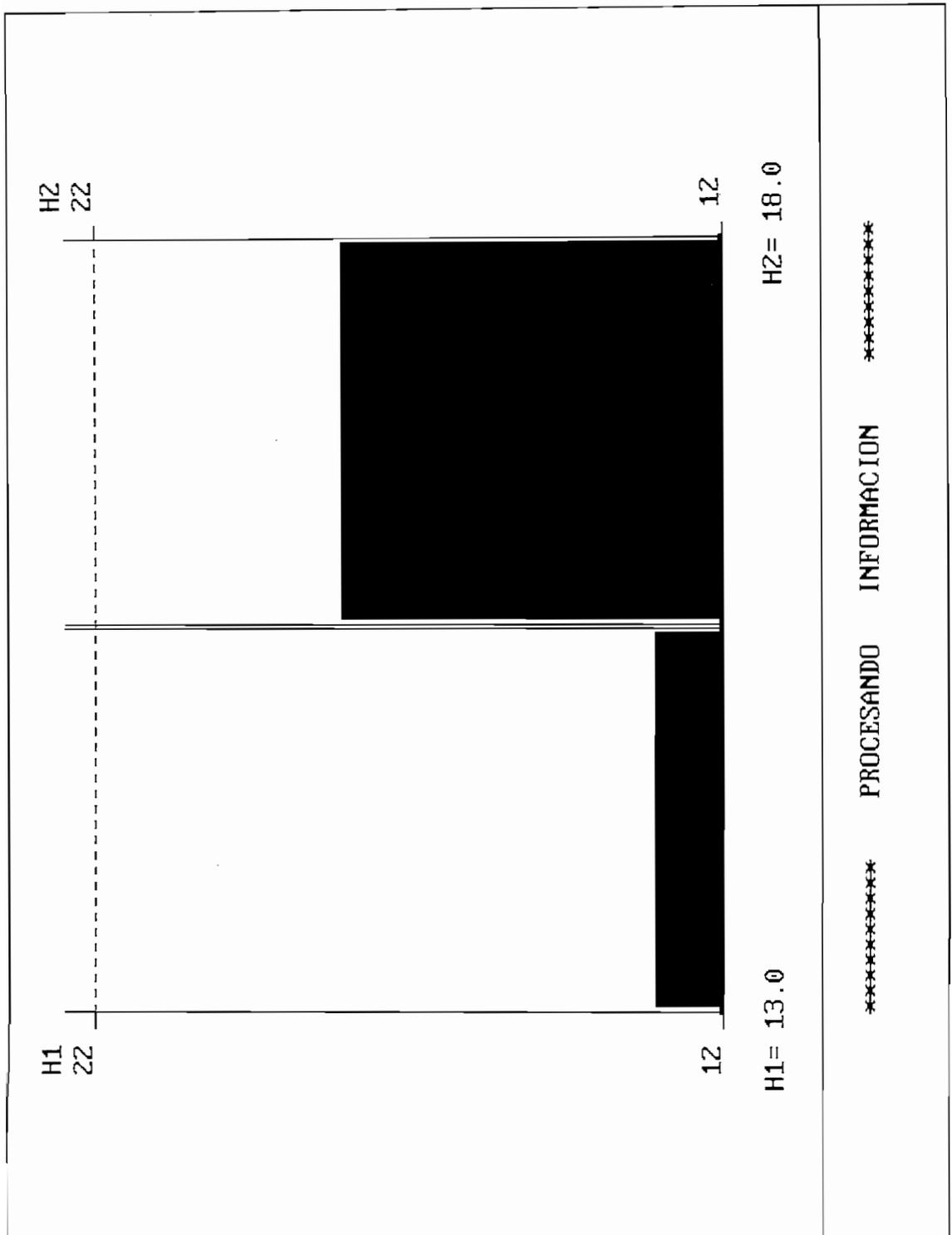


Fig. A8 VENTANA DONDE SE MUESTRA LA SIMULACION DINAMICA  
DEL NIVEL EN LOS TANQUES ACOPLADOS

***MOTOR.BAS***

## APENDICE B.

### LISTADO DE PROGRAMAS.

### MOTOR.BAS

---



---

#### SIMULACION DEL MOTOMATIC MCL-100

---



---

```

DECLARE SUB BORRAR ()
DECLARE SUB RESULTADOS ()
DECLARE SUB ESCALA (LINEA%, TSIM!, GRAFICO%, DATOS!, MAXMIN!())
DECLARE SUB FASEMODULO (n!, M!, XZ!(), XP!(), REAL!, IMAG!, FASE!, MODULO!)
DECLARE SUB GRAFICOS (LINEA%, TSIM!, GRAFICO%, DATOS!(), MAXMIN!())
DECLARE SUB MAX (n%, l!, DATOS!(), MAXMIN!())
DECLARE SUB GRAFMOTOR (ROW%, COLUM%, SALIDA%)
DECLARE SUB ERROR ()
DECLARE SUB PROCESANDO ()
DECLARE SUB MENUS ()
DECLARE SUB BORRARCONDI ()
DECLARE SUB ERRORM1 ()
DECLARE SUB MENUMOTOR1 ()
DECLARE SUB MENUMOTOR2 ()
DECLARE SUB RELACION (ROW%, COLUM%, SALIDA!)
DECLARE SUB VALIDAR (ROW%, COLUM%, NUM!)
DECLARE SUB MODELO ()
DECLARE SUB MOTORR ()
DECLARE SUB VENTANAS1 ()
DECLARE SUB PARAINIM ()
DECLARE SUB CONDINIM ()
DECLARE SUB CONDICIONESM ()
DECLARE SUB PARAMETROSM ()
DECLARE SUB VENTCONDI ()
DECLARE SUB VENTPARA ()
DECLARE SUB VENTANAS ()
DECLARE SUB MOSTRAR (NUM!)
DECLARE SUB AUTOR ()

DECLARE FUNCTION MENU (SUPX, SUPY, n%, MAT) AS STRING
DECLARE FUNCTION NUEVOVALORS ()

```

---



---

#### Definición de Variables Globales

---



---

```

COMMON SHARED TETA!, TETA!
COMMON SHARED TETA VI!, TETA VF!, TETA MAX!
COMMON SHARED TM!, ITERM!
COMMON SHARED KG!, KP!, KP1!, K0!, K1!, K2!, K3!, K4!, PI
COMMON SHARED R0!, R1!, R2!, R3!, R4!, Rell!
COMMON SHARED BLANCO, AZUL, BLANCOI, MAGENTA, NEGRO, GRISI
COMMON SHARED CERO, NUEVE, ENTER, PUNTO, MENOS, DELETE
COMMON SHARED ARRIBA, ABAJO, DERECHA, IZQUIERDA, FONDO, ROJO
COMMON SHARED F1, F2, F3, F4, F5, F6, F7, F8, F9, F10

```

```

KG! = .148 * 3.141592654# / 180: KP! = 6.74 * 3.141592654# / 180
R0! = 30: R1! = 10: R2! = 10: R3! = 10: R4! = 10: Rell! = 1
KA! = 110 * 180 / 3.141592654#
T1! = 1 / (1623): T2! = 1 / (11.5): T3! = 1 / (1635)

```

ARRIBA = 72: ABAJO = 80: DERECHA = 77: IZQUIERDA = 75  
 BLANCO = 7: AZUL = 1: BLANCOI = 11: MAGENTA = 5: NEGRO = 0: GRISI = 15  
 FONDO = 14: ROJO = 4: CERO = 48: NUEVE = 57: ENTER = 13: PUNTO = 46: MENOS = 45  
 DELETE = 8: ESCAPE = 27: F1 = 59: F2 = 60: F3 = 61: F4 = 62  
 F5 = 63: F6 = 64: F7 = 65: F8 = 66: F9 = 67: F10 = 68  
 PI! = 3.141592654#  
 '\$DYNAMIC

---



---

## MODULO PRINCIPAL

---



---

```

INICIOM:
FLAG1 = 0: FLAG2 = 0
FLAG3 = 0: FLAG4 = 0
FLAG5 = 0: FLAG6 = 0
GRAFM = 0: GRAFM1=0
GRAFM1% = 1
GRAFM2% = 2
T! = TM!
N! = Rell!
CAMBIO = 0
CAMBIO1 = 0

CLS
CALL VENTANAS
CALL MOTORR
CALL VENTPARA
CALL VENTCONDI
CALL MODELO
VIEW PRINT 15 TO 16
LOCATE , 50: COLOR FONDO: PRINT " SIN COMPENSADOR "
CALL PARAINIM
CALL CONDINIM

INICIOM1:
CALL MENUMOTOR1
DO
  DO
    CHECK$ = INKEY$
    LOOP UNTIL CHECK$ <> ""
    OPCION$ = CHECK$

  SELECT CASE OPCION$

    CASE CHR$(0) + CHR$(F1)      "Ingreso de Parámetros"
      FLAG2 = 1
      FLAG5 = 0
      CALL PARAMETROSM
      IF N! <> Rell! AND UBICACION = 1 THEN
        CAMBIO1 = 1
        N! = Rell!
        GOTO UBICACION
      END IF
      CALL CONDINIM
      CALL VENTCONDI

    CASE CHR$(0) + CHR$(F2)      "Ingreso de Condiciones"
      FLAG1 = 1
      FLAG5 = 0
  
```

```

CALL CONDICIONESM
IF T! <> TM! THEN
  T! = TM!
  IF PID = 1 THEN
    CAMBIO = 1
    GOTO PID
  ELSEIF UBICACION = 1 THEN
    CAMBIO = 1
    GOTO UBICACION2
  END IF
END IF

CASE CHR$(0) + CHR$(F3)      "Ingreso de la Acción de Control"
  IF FLAG1 = 1 THEN
    FLAG3 = 1
    GOTO COMPMOTOR
  ELSE
    CALL ERRORM1
    CALL MENUMOTOR1
  END IF

CASE CHR$(0) + CHR$(F4)      "Inicio Cálculos para la simulación"
  IF FLAG1 = 1 THEN
    N! = Re!!
    T! = TM!
    FLAG5 = 1
    IF TETA! = TETA! THEN
      FLAG1 = 0
    ELSE
      GOTO MOTOMATIC
    END IF
  ELSE
    CALL ERRORM1
    CALL MENUMOTOR1
  END IF

CASE CHR$(0) + CHR$(F5)      "Presento simulación dinámica"
  IF FLAG5 = 1 THEN
    GOTO SIMDINMOT
  ELSE
    CALL ERROR0
    CALL MENUMOTOR1
  END IF

CASE CHR$(0) + CHR$(F6)      "Presento Gráficos"
  IF FLAG5 = 1 THEN
    FLAG6 = 1
    CALL BORRAR
    CALL RESULTADOS
    GOTO GRAFICOSM
  ELSE
    CALL ERROR0
    CALL MENUMOTOR1
  END IF

CASE CHR$(0) + CHR$(F7)      "Salida del programa"
  GOTO SALIDAM

CASE ELSE
  BEEP
END SELECT
LOOP

```

CHAIN "A:SIMULAMENU" ' "MENU PRINCIPAL"

---



---

**Ingreso de la Acción de Control**

---



---

COMPMOTOR:

Ganancia! = K1! \* K2! \* K4!

Ganancia! = Ganancia! / (K1! \* K2! + K3! \* (K1! + K2! + K1! \* K2!))

CALL MENUMOTOR2

DO

  DO

    CHECK\$ = INKEY\$

  LOOP UNTIL CHECK\$ <> ""

  OPCION\$ = CHECK\$

  SELECT CASE OPCION\$

    CASE CHR\$(0) + CHR\$(F1)      "Salgo sin Cambios"

      EXIT DO

    CASE CHR\$(0) + CHR\$(F2)      "Compensador P.I.D."

      VIEW PRINT 15 TO 16

      LOCATE , 48: PRINT "  COMPENSADOR P.I.D.  "

      CALL MENUS

      FLAG4 = 1

      PID = 1

      COMPENSADORM = 1

      UBICACION = 0

      REALIMENTACION = 0

      VIEW PRINT 23 TO 26

      LOCATE , 44: PRINT "KP  :"; SPACES\$(9 - LEN(STR\$(KPP!))); KPP!

      LOCATE , 44: PRINT "KI  :"; SPACES\$(9 - LEN(STR\$(KI!))); KI!

      LOCATE , 44: PRINT "KD  :"; SPACES\$(9 - LEN(STR\$(KD!))); KD!

      CALL VENTCONDI

COMPMOTOR1:

  VIEW PRINT 23 TO 26

  LOCATE , 44: PRINT "KP  :"; : CALL VALIDAR(23, 43, KPP!)

PROPORCIONAL1:

  IF KPP! = 0 THEN

    SOUND 2000, 5

    VIEW PRINT 23 TO 26

    LOCATE , 44: PRINT "KP  :"; : CALL VALIDAR(23, 43, KPP!)

    GOTO PROPORCIONAL1

  END IF

  CALL VENTCONDI

  LOCATE , 44: PRINT "KI  :"; : CALL VALIDAR(24, 43, KI!)

  CALL VENTCONDI

  LOCATE , 44: PRINT "KD  :"; : CALL VALIDAR(25, 43, KD!)

  CALL VENTCONDI

PID:

  Gcn!(0) = KPP! \* (1 + (KI! \* TM!) / 2 + KD! / TM!)

  Gcn!(1) = KPP! \* (-1 + (KI! \* TM!) / 2 - (2 \* KD!) / TM!)

  Gcn!(2) = KPP! \* (KD! / TM!)

  Gcn!(3) = 0

  Gcd!(0) = 1

  Gcd!(1) = -1

  Gcd!(2) = 0

  Gcd!(3) = 0

  IF CAMBIO = 0 THEN

    OTROS\$ = NUEVOVALOR\$

    IF OTROS\$ = "S" THEN

      CALL MENUS

      GOTO COMPMOTOR1

```

ELSE
    EXIT DO
END IF
END IF
CAMBIO = 0
EXIT DO

CASE CHR$(0) + CHR$(F3)    "Ubicación Arbitraria del cero"
VIEW PRINT 15 TO 16
LOCATE , 48: PRINT " UBICACION ARBITRARIA CERO "
CALL MENUS
FLAG4 = 1
PID = 0
UBICACION = 1
COMPENSADORM = 1
REALIMENTACION = 0

UBICACION:
REDIM Gsnum!(1), Gsden!(3)
Gsnum!(1) = 0
Gsnum!(0) = KA! * 1623 * 11.5 * Ganancia! * KP1! / K1!
Gsden!(3) = 1: Gsden!(2) = 1634.5
Gsden!(1) = 18664.5: Gsden!(0) = 0

IF CAMBIO1 = 1 THEN
    GOTO UBICACION1
END IF
VIEW PRINT 24 TO 26
LOCATE , 44: PRINT "Mp% "; SPACES$(9 - LEN(STR$(MP!))); MP!
LOCATE , 44: PRINT "ts(sg):"; SPACES$(9 - LEN(STR$(TS!))); TS!
CALL VENTCONDI

COMP MOTOR2:
VIEW PRINT 24 TO 26
LOCATE , 44: PRINT "Mp% ";: CALL VALIDAR(24, 43, MP!)

COMP1:
IF MP! <= 0 THEN
    SOUND 2000, 5
    VIEW PRINT 24 TO 26
    LOCATE , 44: PRINT "Mp% ";:
    CALL VALIDAR(24, 43, MP!)
    GOTO COMP1
END IF
CALL VENTCONDI
LOCATE , 44: PRINT "ts(sg):";: CALL VALIDAR(25, 43, TS!)

COMP2:
IF TS! <= 0 THEN
    SOUND 2000, 5
    VIEW PRINT 25 TO 26
    LOCATE , 44: PRINT "ts(sg):";:
    CALL VALIDAR(25, 43, TS!)
    GOTO COMP2
END IF
CALL VENTCONDI

MP1! = MP! / 100
COEF! = SQR(LOG(MP1!) ^ 2 / (PI! ^ 2 + LOG(MP1!) ^ 2))
WN = 4 / (COEF! * TS!)
PDREAL! = -COEF! * WN
PDIMAG! = WN * SQR(1 - COEF! ^ 2)

VIEW PRINT 28 TO 30
LOCATE , 5
PRINT "
";:

```

```

PRINT "
LOCATE , 5
PRINT " "; ;
PRINT "
VIEW PRINT 28 TO 30
LOCATE , 18
PRINT " DESEA UBICAR PERSONALMENTE EL CERO (S/N)"
DO
    CHECK$ = INKEY$
    BICAR$ = CHECK$
    LOOP UNTIL CHECK$ = "S" OR CHECK$ = "s" OR CHECK$ = "N" OR CHECK$ = "n"
    IF UBICAR$ = "S" THEN
        LOCATE , 28
        PRINT "Zero comp = "; : CALL VALIDAR(29, 35, ZCOMP!)
        IF TS! <= 0 THEN
            SOUND 2000, 5
            VIEW PRINT 29 TO 30
            LOCATE , 28
            PRINT "Zero comp = "; : CALL VALIDAR(29, 35, ZCOMP!)
            CALL VALIDAR(25, 40, ZCOMP!)
            GOTO COMP2
        END IF
    ELSE
        ZCOMP! = -PDREAL!
    END IF
END IF

```

UBICACION1:

```

FASEMODULO 1, 3, Gsnum!(), Gsden!(), PDREAL!, PDIMAG!, FASE!, MODULOP!
PCOMP! = ZCOMP! + ABS(PDIMAG!) / (TAN((90 - FASE!) * PI / 180))
REDIM GCSnum!(1), GCSden!(1)
GCSnum!(1) = 1
GCSnum!(0) = ZCOMP!
GCSden!(1) = 1
GCSden!(0) = PCOMP!
FASEMODULO 1, 1, GCSnum!(), GCSden!(), PDREAL!, PDIMAG!, FASE!, MODULOC!
KCOMP! = 1 / (MODULOP! * MODULOC!)
VIEW PRINT 28 TO 30
LOCATE , 5
PRINT " "; ;
PRINT "
LOCATE , 5
PRINT " "; ;
PRINT "
VIEW PRINT 28 TO 30
LOCATE , 5
PRINT " Polo deseado = "; PDREAL!; " +/- j "; PDIMAG!
LOCATE , 5
IF ZCOMP! > 0 THEN
    PRINT " Compensador = "; KCOMP!; " ( S + "; ZCOMP!; ;
ELSE
    PRINT " Compensador = "; KCOMP!; " ( S - "; ABS(ZCOMP!); ;
END IF

IF PCOMP! > 0 THEN
    PRINT ") / ( S + "; PCOMP!; ;)"
ELSE
    PRINT ") / ( S - "; ABS(PCOMP!); ;)"
END IF
DO
    CHECK$ = INKEY$
    LOOP UNTIL CHECK$ <> ""

```

```

IF CAMBIO1 = 0 THEN
  OTRO$ = NUEVOVALOR$
  IF OTRO$ = "S" THEN
    CALL MENUS
    GOTO COMPMOTOR2
  END IF
END IF
CAMBIO1 = 0

```

UBICACION2:

```

Gcn!(0) = KCOMP! * (2 + TM! * ZCOMP!)
Gcn!(1) = -KCOMP! * (2 - TM! * ZCOMP!)
Gcn!(2) = 0
Gcn!(3) = 0
Gcd!(0) = 2 + TM! * PCOMP!
Gcd!(1) = -2 + TM! * PCOMP!
Gcd!(2) = 0
Gcd!(3) = 0
EXIT DO

```

```

CASE CHR$(0) + CHR$(F4)      "Ingreso Cualquier otro compensador"
FOR I = 0 TO 2
  Gcn(I) = 0: Gcd(I) = 0
NEXT I
VIEW PRINT 15 TO 16
LOCATE , 48: PRINT "  OTRO COMPENSADOR  "
CALL MENUS
FLAG4 = 1
PID = 0
COMPENSADORM = 1
UBICACION = 0
REALIMENTACION = 0
VIEW PRINT 17 TO 26
J% = 18
LOCATE , 48: PRINT " GC(Z)  "
CALL VENTCONDI
CALL VENTPARA
FOR I = 0 TO 3
  LOCATE , 44
  PRINT "a("; STR$(I); ") :"; SPACE$(9 - LEN(STR$(Gcn!(I)))) : Gcn!(I)
  J% = J% + 1
NEXT I
FOR I = 0 TO 3
  LOCATE , 44
  PRINT "b("; STR$(I); ") :"; SPACE$(9 - LEN(STR$(Gcd!(I)))) : Gcd!(I)
  J% = J% + 1
NEXT I
CALL VENTCONDI

```

COMPMOTOR3:

```

VIEW PRINT 18 TO 26
J% = 18
FOR I = 0 TO 3
  LOCATE , 44
  PRINT "a("; STR$(I); ") :"; : CALL VALIDAR(J%, 43, Gcn!(I))
  CALL VENTCONDI
  J% = J% + 1
NEXT I
FOR I = 0 TO 3
  LOCATE , 44
  PRINT "b("; STR$(I); ") :"; : CALL VALIDAR(J%, 43, Gcd!(I))
  J% = J% + 1
  CALL VENTCONDI

```

```

NEXT I

OTRO$ = NUEVOVALOR$
IF OTRO$ = "S" THEN
    CALL MENUS
    GOTO COMPMOTOR3
ELSE
    EXIT DO
END IF

CASE CHR$(0) + CHR$(F5)    "Realimentaciòn de Velocidad"
    VIEW PRINT 15 TO 16
    LOCATE , 48: PRINT " REALIMENTACION DE VELOCIDAD "
    FOR I = 0 TO 2
        Gcn(I) = 0: Gcd(I) = 0
    NEXT I
    REALIMENTACION = 1
    COMPENSADORM = 0
    UBICACION = 0
    PID = 0
    EXIT DO

CASE CHR$(0) + CHR$(F6)    "Sin compensador"
    VIEW PRINT 15 TO 16
    LOCATE , 48: PRINT " SIN COMPENSADOR "
    FOR I = 0 TO 2
        Gcn(I) = 0: Gcd(I) = 0
    NEXT I
    FLAG4 = 0
    PID = 0
    COMPENSADORM = 0
    UBICACION = 0
    REALIMENTACION = 0
    EXIT DO

CASE ELSE
    BEEP

END SELECT
LOOP
IF FLAG4 = 1 THEN
    CALL BORRARCONDI
    CALL CONDINIM
END IF
FLAG4 = 0
FLAG5 = 0
GOTO INICIOM1

```

---



---

### CALCULO DE LOS VALORES PARA LA SIMULACION

---



---

MOTOMATIC:

Ganancia! = K1! \* K2! \* K4!

Ganancia! = Ganancia! / (K1! \* K2! + K3! \* (K1! + K2! + K1! \* K2!))

AP! = EXP(-TM! / T1!)

BP! = EXP(-TM! / T2!)

K1P! = T1! / (T2! - T1!)

K2P! = T2! / (T1! - T2!)

Gnum!(0) = 0

Gnum!(1) = -(AP! + BP! + (1 + AP!) \* K2P! + (1 + BP!) \* K1P!) \* KA! \* TM!

```
Gnum!(2) = (AP! * BP! + K1P! * BP! + K2P! * AP!) * KA! * TM!
Gnum!(3) = 0
Gden!(0) = 1
Gden!(1) = -(1 + AP! + BP!)
Gden!(2) = AP! + AP! * BP! + BP!
Gden!(3) = -AP! * BP!
```

```
K1P! = KA! / (T1! - T2!)
K2P! = KA! / (T2! - T1!)
```

```
Wnum!(0) = 0
Wnum!(1) = -(BP! * K1P! + AP! * K2P!) * TM!
Wden!(0) = 1
Wden!(1) = -(AP! + BP!)
Wden!(2) = AP! * BP!
```

```
REDIM MOTOR!(6, ITERM!), MAXIMOS!(6, 2)
CALL PROCESANDO
```

```
FOR K = 1 TO ITERM! STEP 1
  VIEW PRINT 29 TO 30
  LOCATE , 39: PRINT K
  IF K = 1 THEN
    MOTOR!(1, K) = 0
    MOTOR!(2, K) = 0
  ELSE
    MOTOR!(2, K) = 0
    FOR I = 1 TO 2
      MOTOR!(2, K) = MOTOR!(2, K) - Wden!(I) * MOTOR!(2, K - I)
    NEXT
    MOTOR!(2, K) = MOTOR!(2, K) + Wnum!(1) * MOTOR!(4, K - 1)

    MOTOR!(1, K) = 0
    FOR I = 1 TO 3
      IF K - I > 0 THEN
        MOTOR!(1, K) = MOTOR!(1, K) + Gnum!(I) * MOTOR!(4, K - I)
        MOTOR!(1, K) = MOTOR!(1, K) - Gden!(I) * MOTOR!(1, K - I)
      END IF
    NEXT
  END IF
END IF
```

```
*****
'CALCULO DE LA VARIABLE MEDIDA
*****
```

```
MOTOR!(6, K) = MOTOR!(1, K) * RELL!
IF MOTOR!(6, K) >= 170 THEN
  MOTOR!(1, K) = 170 / RELL!
  MOTOR!(6, K) = 170
ELSEIF MOTOR!(6, K) <= -170 THEN
  MOTOR!(1, K) = -170 / RELL!
  MOTOR!(6, K) = -170
END IF
MOTOR!(5, K) = KP! * MOTOR!(6, K) / K1!
IF REALMENTACION = 1 THEN
  MOTOR!(5, K) = MOTOR!(5, K) + KG! * MOTOR!(2, K) / K2!
END IF
```

```
*****
'CALCULO DE LA VARIABLE ERROR
*****
MOTOR!(3, K) = (TETA V F! - TETA V I!) - MOTOR!(5, K)
MOTOR!(3, K) = MOTOR!(3, K) * Ganancia!
```

```

*****
'CALCULO DE LA LEY DE CONTROL
*****

```

```

IF COMPENSADORM = 1 THEN
  MOTOR!(4, K) = 0
  FOR I = 0 TO 3
    IF K - I > 0 THEN
      MOTOR!(4, K) = MOTOR!(4, K) + Gcn!(I) * MOTOR!(3, K - I)
    END IF
  NEXT I
  FOR I = 1 TO 3
    IF K - I > 0 THEN
      MOTOR!(4, K) = MOTOR!(4, K) - Gcd!(I) * MOTOR!(4, K - I)
    END IF
  NEXT I
  MOTOR!(4, K) = MOTOR!(4, K) / Gcd!(0)
ELSE
  MOTOR!(4, K) = MOTOR!(3, K)
END IF

```

```

NEXT K

```

```

FOR I = 1 TO ITERM! STEP 1

```

```

  MOTOR!(1, I) = MOTOR!(1, I) + TETA!
  MOTOR!(2, I) = MOTOR!(2, I) / 6
  MOTOR!(5, I) = MOTOR!(5, I) + TETA!
  MOTOR!(6, I) = MOTOR!(1, I) * REL!

```

Angulo real

Velocidad en RPM

Angulo medido (V)

Angulo despues reductor

```

NEXT I

```

```

CALL MAX(6, ITERM!, MOTOR!(), MAXIMOS!())

```

```

CALL MNUMOTOR1

```

```

GOTO INICIOM1

```

```

*****
***** SIMULACION DINAMICA DEL MOTOMATIC *****
*****

```

```

SIMDINMOT:

```

```

GRAFM = 1

```

```

VIEW (5, 4)-(325, 411)

```

```

CLS

```

```

VIEW

```

```

WINDOW

```

```

IF FLAG6 = 1 THEN

```

```

  VIEW (323, 4)-(635, 219)

```

```

  CLS

```

```

  VIEW

```

```

  WINDOW

```

```

END IF

```

```

CALL MOTORR

```

```

CALL MODELO

```

```

VIEW (99, 62)-(245, 208)

```

```

WINDOW (-1.3, -1.3)-(1.3, 1.3)

```

```

CIRCLE (X, Y), .1, 0, , , .5

```

```

FOR K = 1 TO ITERM!

```

```

  CHECK$ = INKEY$

```

```

  IF CHECK$ = CHR$(ESCAPE) THEN

```

```

    GOTO FINAL1

```

```

  END IF

```

```
TETA = MOTOR!(6, K) * 3.1415927# / 180
TETA1 = MOTOR!(6, K - 1) * 3.1415927# / 180
```

```
X = COS(TETA)
Y = SIN(TETA)
X1 = COS(TETA1)
Y1 = SIN(TETA1)
    VIEW (99, 62)-(245, 208)
    WINDOW (-1.3, -1.3)-(1.3, 1.3)
LINE (0, 0)-(X1, Y1), 0
LINE (0, 0)-(X, Y)
```

```
VIEW PRINT 3 TO 4
LOCATE , 18
PRINT USING "####.# Grad"; MOTOR!(6, K)
```

```
X = COS(TETA / RELL!)
Y = SIN(TETA / RELL!)
X1 = COS(TETA1 / RELL!)
Y1 = SIN(TETA1 / RELL!)
```

```
VIEW (137, 257)-(207, 327)
WINDOW (-1.3, -1.3)-(1.3, 1.3)
LINE (0, 0)-(X1, Y1), 0
LINE (0, 0)-(X, Y)
```

```
VIEW PRINT 24 TO 25
LOCATE , 18
PRINT USING "####.# Grad"; MOTOR!(1, K)
IF K = 1 THEN
    CALL PROCESANDO
    DO
        COLOR FONDO
        VIEW PRINT 28 TO 30
        LOCATE , 13
        PRINT "***** (PRESIONE ENTER PARA CONTINUAR) *****"
        CHECK$ = INKEY$
        LOOP UNTIL CHECK$ = CHR$(ENTER)
        CALL PROCESANDO
    END IF
    FOR J = 1 TO 10000
        NEXT J
```

```
NEXT K
FINAL1:
VIEW
WINDOW
GOTO INCIOM1
```

```
*****
***** RUTINA GRAFICA PARA EL MOTOMATIC *****
*****
```

```
GRAFICOSM:
REDIM REDUC$(6)
REDUC$(1) = "  $\theta(t)^\circ$ "
REDUC$(2) = "  $\omega(t)$ RPM"
REDUC$(3) = "  $V_e(t)$ "
REDUC$(4) = "  $V_u(t)$ "
REDUC$(5) = "  $V_{med}(t)$ "
REDUC$(6) = "  $\theta_{red}(t)$ "
```

```
SOUND 4200, 8
```

```
VIEW PRINT 3 TO 10
LOCATE , 46: PRINT "Graf1 :", "", REDUC$(GRAFM1%)
VIEW PRINT 3 TO 10
LOCATE , 65: PRINT "Graf2 :", "", REDUC$(GRAFM2%)
```

```
VIEW PRINT 4 TO 15
FOR I = 1 TO 10
LOCATE , 44
PRINT "          "
NEXT I
CALL RESULTADOS
```

```
VIEW PRINT 4 TO 15
LOCATE , 44
PRINT "Escala tiempo  Ø"; :
PRINT USING "####.## Seg/div"; ITERM! * TM! / 10
LOCATE , 44
PRINT "Tiempo simul.  Ø"; :
PRINT USING "####.## Seg"; ITERM! * TM!
```

```
VIEW PRINT 3 TO 10
LOCATE , 46: PRINT "Graf1 :"; : CALL GRAFMOTOR(3, 40, GRAFM1%)
VIEW PRINT 3 TO 10
LOCATE , 65: PRINT "Graf2 :"; : CALL GRAFMOTOR(3, 59, GRAFM2%)
```

```
VIEW (6, 5)-(324, 204)
CLS
IF MAXIMOS!(GRAFM1%, 1) <> MAXIMOS!(GRAFM1%, 2) THEN
CALL GRAFICOS(1, ITERM!, GRAFM1%, MOTOR!(), MAXIMOS!())
END IF
```

```
VIEW (6, 211)-(324, 410)
CLS
IF MAXIMOS!(GRAFM2%, 1) <> MAXIMOS!(GRAFM2%, 2) THEN
CALL GRAFICOS(2, ITERM!, GRAFM2%, MOTOR!(), MAXIMOS!())
END IF
```

```
CALL ESCALA(1, ITERM!, GRAFM1%, MOTOR!(), MAXIMOS!())
CALL ESCALA(2, ITERM!, GRAFM2%, MOTOR!(), MAXIMOS!())
```

```
‘ *****
‘ Desplazamiento sobre las curvas graficadas
‘ *****
```

```
VIEW PRINT 28 TO 30
LOCATE , 5
PRINT "          CON LAS FLECHAS ME DESPLAZO EN LAS CURVAS          "
LOCATE , 5
PRINT "          ***** FIN (PRESIONE ESCAPE PARA CONTINUAR) *****          "
```

```
K = 0
DO
SS = INKEY$
SELECT CASE SS
CASE CHR$(0) + CHR$(DERECHA)
IF K = ITERM! THEN
K = 0
ELSE
K = K + 1
END IF

CASE CHR$(0) + CHR$(IZQUIERDA)
```

```

        IF K = 0 THEN
            K = ITERM!
        ELSE
            K = K - 1
        END IF

        CASE CHR$(ESCAPE)
        EXIT DO

        CASE ELSE
        END SELECT

        VIEW PRINT 12 TO 15
        LOCATE , 56
        PRINT "Iter 0";
        PRINT USING "####", K
        LOCATE , 70
        MOSTRAR (MOTOR!(GRAFM2%, K))
        VIEW PRINT 13 TO 15
        LOCATE , 51
        MOSTRAR (MOTOR!(GRAFM1%, K))

    LOOP UNTIL S$ = CHR$(ESCAPE)

    VIEW
    WINDOW
    GOTO INICIOM1

    !*****
    !*  SALIDA DE LA EJECUCION DEL MOTOMATIC  *
    !*****

    SALIDAM:
    VIEW PRINT 28 TO 30
    LOCATE , 5
    PRINT "           ";
    PRINT "           "
    LOCATE , 5
    PRINT "           ";
    PRINT "           "

    VIEW PRINT 28 TO 30
    LOCATE , 20
    PRINT "ESTA SEGURO QUE DESEA SALIR DEL PROGRAMA (S/N)"
    SOUND 5000, 8
    DO
        CHECK$ = INKEY$
    LOOP UNTIL CHECK$ = "S" OR CHECK$ = "s" OR CHECK$ = "N" OR CHECK$ = "n"

    IF CHECK$ = "S" OR CHECK$ = "s" THEN
        CLS
        END
    ELSE
        GOTO INICIOM1
    END IF
    FIN:
    END

```

---



---

FIN DEL PROGRAMA PRINCIPAL

---



---

SUBROUTINAS MOTOMATIC

```
DEFDBL O-Z
SUB AUTOR
```

```
CLS
REDIM MAT$(10)
REDIM LONGI(10)
SUPX = 6: SUPY = 12
MAT$(1) = " ESCUELA POLITECNICA NACIONAL "
MAT$(2) = " FACULTAD DE INGENIERIA ELECTRICA "
MAT$(3) = " DEPARTAMENTO DE ELECTRONICA Y CONTROL "
MAT$(4) = " "
MAT$(5) = " Realizado por: JUAN E. RAMOS P. "
MAT$(6) = " "
MAT$(7) = " Dirigido por: ING. MARCO BARRAGAN "
MAT$(8) = " "
MAT$(9) = " "
MAT$(10) = " E.P.N. 1996 "
```

```
LOCATE 1, 1, 0
LON% = LEN(MAT$(1))
COLOR FONDO
LOCATE SUPX, SUPY
PRINT "┌ "; STRING$(LON% + 10, 205); "┐ "
LOCATE SUPX + 1, SUPY
PRINT "│ "; SPACES$(5); MAT$(1); SPACES$(5); "│ "
LOCATE SUPX + 2, SUPY
PRINT "└ "; STRING$(LON% + 10, 205); "┘ "
CURSOR% = 2
```

```
n% = 10
J = 3
FOR K = 2 TO n%
    LOCATE SUPX + J, SUPY
    PRINT "│ ";
    PIV% = 5 + LON% - LEN(MAT$(K))
    PRINT SPACES$(5); MAT$(K); SPACES$(PIV%);
    J = J + 1
    PRINT "│ "
```

```
NEXT K
LOCATE SUPX + 2 + n%, SUPY
PRINT "└ "; STRING$(LON% + 10, 205); "┘ "
COLOR BLANCO, NEGRO
FOR I = 1 TO 5
NEXT
END SUB
```

```
REM $STATIC
DEFDBL O-Z
```

```
SUB BORRAR
```

```
VIEW (5, 4)-(323, 409)
CLS
VIEW
```

WINDOW

VIEW (323, 4)-(635, 219)  
CLS  
VIEW  
WINDOW

CALL VENTANAS1  
CALL RESULTADOS

END SUB

#### SUB BORRARCONDI

VIEW PRINT 18 TO 26  
FOR I = 1 TO 8  
    LOCATE , 44  
    PRINT "           "  
NEXT I  
CALL VENTCONDI

END SUB

#### SUB BORRARPARA

VIEW PRINT 18 TO 26  
FOR I = 1 TO 8  
    LOCATE , 63  
    PRINT "           "  
NEXT I  
CALL VENTPARA  
END SUB

#### SUB CONDICIONESM

KP1! = KP! \* RELL!  
K1! = R1! / R0!  
K2! = R2! / R0!  
K3! = R3! / R0!  
K4! = R4! / R0!

VIEW PRINT 20 TO 26  
LOCATE , 44: PRINT "θ(ini):"; : CALL VALIDAR(20, 43, TETA1!)  
REINGRESOM1:  
IF ABS(TETA1!) > TETAMAX! THEN  
    SOUND 2000, 5  
    VIEW PRINT 20 TO 26  
    LOCATE , 44: PRINT "θ(ini):"; :  
    CALL VALIDAR(20, 43, TETA1!)  
    GOTO REINGRESOM1  
END IF  
CALL VENTCONDI  
TETA VI! = TETA1! \* KP1! / (K1!)  
LOCATE , 44

LOCATE , 44: PRINT "θ(fin):"; : CALL VALIDAR(21, 43, TETA F!)  
REINGRESOM2:  
IF ABS(TETA F!) > TETAMAX! THEN

```

    SOUND 2000, 5
    VIEW PRINT 21 TO 26
    LOCATE , 44: PRINT "θ(fin):":
    CALL VALIDAR(21, 43, TETAf!)
    GOTO REINGRESOM2
END IF
CALL VENTCONDI
TETAfV! = TETAf! * KP1! / (K1!)

LOCATE , 44: PRINT "T(Seg):"; : CALL VALIDAR(22, 43, TM!)
REINGRESOM3:
IF TM! <= 0 THEN
    SOUND 2000, 5
    VIEW PRINT 22 TO 25
    LOCATE , 44: PRINT "T(Seg):"; :
    CALL VALIDAR(22, 43, TM!)
    GOTO REINGRESOM3
END IF
CALL VENTCONDI

LOCATE , 44: PRINT "Iter :"; : CALL VALIDAR(23, 43, ITERM!)
REINGRESOM4:
IF ITERM! <= 0 OR (ITERM! - INT(ITERM!)) <> 0 THEN
    SOUND 2000, 5
    VIEW PRINT 23 TO 26
    LOCATE , 44: PRINT "Iter :"; :
    CALL VALIDAR(23, 43, ITERM!)
    GOTO REINGRESOM4
END IF
CALL VENTCONDI

END SUB

SUB CONDIM

KP1! = KP! * RELL!
K1! = R1! / R0!
TETAMAX! = K1! * 20 / KP1!

VIEW PRINT 17 TO 26
LOCATE , 48: PRINT "CONDICIONES   PARAMETROS"
CALL VENTCONDI: CALL VENTPARA
LOCATE , 44
PRINT "θ(max):"; SPACE$(9 - LEN(STR$(TETAMAX!))); TETAMAX!

VIEW PRINT 20 TO 26
LOCATE , 44
IF ABS(TETAi!) > ABS(TETAMAX!) THEN
    TETAi! = TETAMAX!
END IF
TETAiV! = TETAi! * KP1! / (K1!)
PRINT "θ(ini):"; SPACE$(9 - LEN(STR$(TETAi!))); TETAi!

LOCATE , 44
IF ABS(TETAf!) > ABS(TETAMAX!) THEN
    TETAf! = TETAMAX!
END IF
TETAfV! = TETAf! * KP1! / (K1!)
PRINT "θ(fin):"; SPACE$(9 - LEN(STR$(TETAf!))); TETAf!

```

```

LOCATE , 44
PRINT "T(Seg):"; SPACE$(9 - LEN(STR$(TM!))); TM!
LOCATE , 44
PRINT "Iter :"; SPACE$(9 - LEN(STR$(ITERM!))); ITERM!
CALL VENTCONDI

END SUB

```

**SUB ERROR0**

```
VIEW PRINT 28 TO 30
```

```

LOCATE , 5
PRINT "           ";
PRINT "           "
LOCATE , 5
PRINT "           ";
PRINT "           "

```

```
VIEW PRINT 28 TO 30
```

```

LOCATE , 18
PRINT "HA Y QUE PRIMERO UTILIZAR LA OPCION *CALCULAR*"
SOUND 2000, 5
DO
LOOP UNTIL INKEY$ <> ""

```

```
END SUB
```

**SUB ERRORM1**

```
VIEW PRINT 28 TO 30
```

```

LOCATE , 5
PRINT "           ";
PRINT "           "
LOCATE , 5
PRINT "           ";
PRINT "           "

```

```
VIEW PRINT 28 TO 30
```

```

LOCATE , 20
PRINT " INGRESE CONDICIONES PARA LA SIMULACIOM"
SOUND 2000, 5
CALL VENTANAS
DO
CHECK$ = INKEY$
LOOP UNTIL CHECK$ <> ""

```

```
END SUB
```

**SUB ERRORM2**

```
VIEW PRINT 28 TO 30
```

```

LOCATE , 5
PRINT "           ";
PRINT "           "
LOCATE , 5

```

```

PRINT "                ";:
PRINT "                "

VIEW PRINT 28 TO 30
LOCATE , 20
PRINT "PERIODO DE MUESTREO DEBE SER DIFERENTE DE CERO"
SOUND 2000, 5
DO
LOOP UNTIL INKEY$ <> ""

END SUB

SUB ERRORM3

VIEW PRINT 28 TO 30
LOCATE , 5
PRINT "                ";:
PRINT "                "
LOCATE , 5
PRINT "                ";:
PRINT "                "

VIEW PRINT 28 TO 30
LOCATE , 20
PRINT "ITERACCIONES DEBEN SER DIFERENETES DE CERO"
SOUND 2000, 5
DO
LOOP UNTIL INKEY$ <> ""

END SUB

SUB ESCALA (LINEA%, TSIM!, GRAFICO%, DATOS!, MAXMIN!())

YMAX! = MAXMIN!(GRAFICO%, 1)
YMIN! = MAXMIN!(GRAFICO%, 2)

IF YMAX! > 0 AND YMIN! >= 0 THEN
  DELTAY1 = YMIN!
  DELTAY2 = YMAX!
  INCY = (YMAX! - YMIN!) / 10

  ELSEIF YMAX! <= 0 AND YMIN! < 0 THEN
    DELTAY1 = YMAX!
    DELTAY2 = YMIN!
    INCY = (YMIN! - YMAX!) / 10
ELSE
  DELTAY1 = 1.05 * YMIN!
  DELTAY2 = 1.05 * YMAX!
  EJEY1 = 1.05 * YMIN!
  EJEY2 = 1.05 * YMAX!
  INCY = (YMAX! - YMIN!) / 10
END IF

*****
*IMPRIME LOS VALORES CARACTERISTICOS DE LA CURVA DESEADA
*****

IF LINEA% = 1 THEN

```

```

VIEW PRINT 7 TO 15
LOCATE ,44
PRINT "Scal: ";:
MOSTRAR (INCY)
LOCATE ,44
PRINT "Max: ";:
MOSTRAR (YMAX!)
LOCATE ,44
PRINT "Min: ";:
MOSTRAR (YMIN!)
LOCATE ,44
PRINT "Ini: ";:
MOSTRAR (DATOS!(GRAFICO%, 1))
LOCATE ,44
PRINT "Fin: ";:
MOSTRAR (DATOS!(GRAFICO%, TSIM!))
ELSE
VIEW PRINT 7 TO 15
LOCATE ,63
PRINT "Scal: ";:
MOSTRAR (INCY)
LOCATE ,63
PRINT "Max: ";:
MOSTRAR (YMAX!)
LOCATE ,63
PRINT "Min: ";:
MOSTRAR (YMIN!)
LOCATE ,63
PRINT "Ini: ";:
MOSTRAR (DATOS!(GRAFICO%, 1))
LOCATE ,63
PRINT "Fin: ";:
MOSTRAR (DATOS!(GRAFICO%, TSIM!))
END IF
END SUB

```

SUB FASEMODULO (n, M, XZ!(), XP!(), REAL!, IMAG!, FASE!, MODULO!)

REDIM XZ1(n), XP1(M)

\*\*\*\*\*  
'CALCULO DE LA FASE Y MODULO DEL NUMERADOR  
\*\*\*\*\*

```

u = 2 * REAL!
V = -REAL! ^ 2 - IMAG! ^ 2
XZ1(0) = XZ!(n)
XZ1(1) = XZ!(n - 1) + u * XZ1(0)
FOR K = 2 TO n
    XZ1(K) = XZ!(n - K) + u * XZ1(K - 1) + V * XZ1(K - 2)
NEXT K
REALZ = XZ1(n) - (XZ1(n - 1) * REAL!)
IMAGZ = XZ1(n - 1) * IMAG!

```

\*\*\*\*\*  
'CALCULO DE LA FASE Y MODULO DEL DENOMINADOR  
\*\*\*\*\*

```

XP1(0) = XP!(M)
XP1(1) = XP!(M - 1) + u * XP1(0)

```

```

FOR K = 2 TO M
  XP1(K) = XP1(M - K) + u * XP1(K - 1) + V * XP1(K - 2)
NEXT K
REALP = XP1(M) - XP1(M - 1) * REAL!
IMAGP = XP1(M - 1) * IMAG!

*****
'DIVIDO NUMERADOR / DENOMINADOR
*****

DENOMINADOR = REALP ^ 2 + IMAGP ^ 2
REALL! = (REALZ * REALP + IMAGZ * IMAGP) / DENOMINADOR
IMAGG! = (IMAGZ * REALP - IMAGP * REALZ) / DENOMINADOR
MODULO = SQR(REALL! ^ 2 + IMAGG! ^ 2)
IF REALL! = 0 THEN
  ANGULO! = 90
ELSE
  ANGULO! = 180 * ATN(IMAGG! / REALL!) / 3.141592654#
END IF
*****
'REPRESENTO EN COORDENADAS POLARES
*****

IF IMAGG! >= 0 AND REALL! > 0 THEN
  ANGULO! = ANGULO!
ELSEIF IMAGG! < 0 AND REALL! < 0 THEN
  ANGULO! = ANGULO! + 180
ELSEIF IMAGG! > 0 AND REALL! < 0 THEN
  ANGULO! = ANGULO! + 180
ELSE
  ANGULO! = ANGULO! + 360
END IF

IF ANGULO >= 0 THEN
  ANGULO! = 180 - ANGULO!
ELSE
  ANGULO! = -180 - ANGULO!
END IF
FASE! = ANGULO
END SUB

SUB GRAFICOS (LINEA %, TSIM!, GRAFICO %, DATOS!0, MAXMIN!0)

INCX = TSIM! / 10
DELTAX1 = -.02 * TSIM!
DELTAX2 = 1.02 * TSIM!
EJEX1 = -.05 * TSIM!
EJEX2 = 1.05 * TSIM!

YMAX! = MAXMIN!(GRAFICO%, 1)
YMIN! = MAXMIN!(GRAFICO%, 2)

IF YMAX! > 0 AND YMIN! >= 0 THEN
  DELTAY1 = YMIN!
  DELTAY2 = YMAX!
  INCY = (YMAX! - YMIN!) / 10

  ELSEIF YMAX! <= 0 AND YMIN! < 0 THEN
    DELTAY1 = YMAX!
    DELTAY2 = YMIN!

```

```

        INCY = (YMIN! - YMAX!) / 10
ELSE
    DELTAY1 = 1.05 * YMIN!
    DELTAY2 = 1.05 * YMAX!
    EJEY1 = 1.05 * YMIN!
    EJEY2 = 1.05 * YMAX!
    INCY = (YMAX! - YMIN!) / 10
END IF

BARX = .05 * (DELTAY2 - DELTAY1)
BARY = .02 * (EJEX2 - EJEX1)
WINDOW (DELTAX1, DELTAY1)-(DELTAX2, DELTAY2)

*****
'GRAFICO LOS EJES PARA LOS GRAFICOS
*****
IF YMAX! < 0 AND YMIN! >= 0 OR YMAX! > 0 AND YMIN! <= 0 THEN
    LINE (EJEX1, 0)-(EJEX2, 0)
ELSE
    EJE = DELTAY1
    LINE (EJEX1, EJE)-(EJEX2, EJE), , &HF0F0
END IF
LINE (0, DELTAY1)-(0, DELTAY2)

*****
'GRAFICO MARCAS PARA EL EJE X
*****
IF YMAX! < 0 AND YMIN! >= 0 OR YMAX! > 0 AND YMIN! <= 0 THEN
    FOR I = 0 TO 10
        LINE (INCX * I, -BARX / 2)-(INCX * I, BARX / 2)
    NEXT
ELSE
    EJEY = DELTAY1
    FOR I = 0 TO 10
        LINE (INCX * I, EJEY + BARX / 2)-(INCX * I, EJEY - BARX / 2)
    NEXT
END IF

*****
'GRAFICO MARCAS PARA EL EJE Y
*****
IF YMIN! >= 0 AND YMAX! > 0 THEN
    FOR I = 0 TO 10
        LINE (-BARY / 2, YMIN! + INCY * I)-(-BARY / 2, YMIN! + INCY * I)
    NEXT I
ELSEIF YMAX! <= 0 AND YMIN! < 0 THEN
    FOR I = 0 TO 10
        LINE (-BARY / 2, YMAX! + INCY * I)-(-BARY / 2, YMAX! + INCY * I)
    NEXT I
ELSE
    CONT = 0
    FOR I = 0 TO EJEY2 STEP INCY
        LINE (-BARY / 2, CONT * INCY)-(-BARY / 2, CONT * INCY)
        CONT = CONT + 1
    NEXT I

    CONT = 0
    FOR I = 0 TO EJEY1 STEP -INCY
        LINE (-BARY / 2, -CONT * INCY)-(-BARY / 2, -CONT * INCY)
        CONT = CONT + 1
    NEXT I
END IF

```

```

*****
***** GRAFICO LA CURVA DESEADA *****
*****
PSET (0, DATOS!(GRAFICO%, 1))
FOR I = 1 TO TSIM!
    LINE -(I - 1, DATOS!(GRAFICO%, I))
NEXT
END SUB

SUB GRAFMOTOR (ROW%, COLUM%, SALIDA%)

DIM REDUC$(6)
REDUC$(1) = "  $\theta(t)$ "
REDUC$(2) = "  $\omega(t)$ RPM"
REDUC$(3) = "  $V_c(t)$ "
REDUC$(4) = "  $V_u(t)$ "
REDUC$(5) = "  $V_{med}(t)$ "
REDUC$(6) = "  $\theta_{red}(t)$ "

CONTADOR = 1
MARCA% = COLUM% + 11
LOCATE ROW%, MARCA%
PRINT "0"
PASO = COLUM% + 13
LOCATE ROW%, PASO

DO
    LOCATE ROW%, PASO
    $$ = INKEY$
    SELECT CASE $$
        CASE CHR$(0) + CHR$(ARRIBA)
            FLAG1 = 1
            IF CONTADOR = 6 THEN
                CONTADOR = 1
            ELSE
                CONTADOR = CONTADOR + 1
            END IF
            PRINT REDUC$(CONTADOR)

        CASE CHR$(0) + CHR$(ABAJO)
            FLAG1 = 1
            IF CONTADOR = 1 THEN
                CONTADOR = 6
            ELSE
                CONTADOR = CONTADOR - 1
            END IF
            PRINT REDUC$(CONTADOR)

        CASE CHR$(ENTER)
            IF FLAG1 = 1 THEN
                SALIDA% = CONTADOR
            END IF
            LOCATE ROW%, MARCA%
            PRINT " "
            EXIT DO

        CASE ELSE
            END SELECT
    LOOP UNTIL $$ = CHR$(13)

```

END SUB

SUB MAX (n%, I!, DATOS!(), MAXMIN!())

```

FOR I = 1 TO n%
  YMAX! = DATOS!(I, 1)
  YMIN! = DATOS!(I, 2)

  FOR J = 1 TO I!
    IF YMAX! < DATOS!(I, J) THEN
      YMAX! = DATOS!(I, J)
    END IF
    IF YMIN! > DATOS!(I, J) THEN
      YMIN! = DATOS!(I, J)
    END IF
  NEXT J
  MAXMIN!(I, 1) = YMAX!
  MAXMIN!(I, 2) = YMIN!

```

NEXT I

END SUB

DEFSNG O-Z

DEFDBL O-Z

SUB MENUMOTOR1

VIEW PRINT 28 TO 30

LOCATE , 5

PRINT "F1 = PARAMETROS    F3 = COMPENSACION    F5 = SIM. DINAMICA    F7=SALIR "

LOCATE , 5

PRINT "F2 = CONDICIONES    F4 = CALCULAR            F6 = GRAFICOS            "

END SUB

SUB MENUMOTOR2

VIEW PRINT 28 TO 30

LOCATE , 5

PRINT "                            " ; :

PRINT "                            " ; :

LOCATE , 5

PRINT "                            " ; :

PRINT "                            " ; :

VIEW PRINT 28 TO 30

LOCATE , 5

PRINT "F1 = CONTINUAR    F2 = COMPENSADOR P.I.D.    F3 = UBICACION POLO/CERO"; ""

LOCATE , 5

PRINT "F4 = OTRO COMPENSADOR    F5 = REALIMENTACION W    F6 = SIN COMPENSACION

"

CALL VENTANAS

END SUB

## SUB MENUS

VIEW PRINT 28 TO 30

LOCATE , 5

PRINT " " ; :

PRINT " " ; :

LOCATE , 5

PRINT " " ; :

PRINT " " ; :

VIEW PRINT 28 TO 30

LOCATE , 28

PRINT " INGRESANDO COMPENSACION"

SOUND 2000, 5

END SUB

DEFSNG O-Z

## SUB MODELO

VIEW PRINT 3 TO 4

LOCATE , 48: PRINT " MOTOMATIC MCL-100 " "

CIRCLE (322, 103), 2, , , 1

LINE (324, 103)-(338, 103) 'OK

LINE (338, 90)-(365, 115), , B 'OK

LINE (365, 103)-(370, 103) 'OK

CIRCLE (380, 103), 10, , , 1 'OK

LINE (390, 103)-(400, 103) 'OK

LINE (400, 90)-(430, 115), , B 'OK

LINE (430, 103)-(450, 103) 'OK

LINE (450, 88)-(450, 118) 'OK operacional

LINE (450, 88)-(480, 103) 'OK operacional

LINE (450, 118)-(480, 103) 'OK operacional

LINE (480, 103)-(520, 103) 'OK

LINE (440, 70)-(440, 103) 'OK

LINE (440, 70)-(456, 70) 'OK

LINE (456, 60)-(490, 80), , B 'OK

LINE (490, 70)-(500, 70) 'OK

LINE (500, 70)-(500, 103) 'OK

LINE (520, 90)-(590, 115), , B 'OK

LINE (590, 103)-(607, 103) 'OK

CIRCLE (609, 103), 2, , , 1 'OK

LINE (418, 123)-(445, 147), , B 'OK

LINE (498, 123)-(525, 147), , B 'OK

LINE (445, 135)-(498, 135) 'OK

LINE (525, 135)-(597, 135) 'OK

LINE (597, 167)-(597, 103) 'OK

LINE (395, 135)-(385, 114) 'OK

LINE (418, 135)-(395, 135) 'OK

LINE (418, 155)-(445, 179), , B 'OK

LINE (498, 155)-(525, 179), , B 'OK

LINE (445, 167)-(498, 167) 'OK

LINE (525, 167)-(597, 167) 'OK

LINE (418, 167)-(380, 167) 'OK

LINE (380, 167)-(380, 113) 'OK

VIEW PRINT 5 TO 17

LOCATE , 59: PRINT "R4"

LOCATE , 50: PRINT ""

```

LOCATE , 39: PRINT "Xi"; : LOCATE , 44: PRINT "R0"; :
LOCATE , 48: PRINT "+"; : LOCATE , 52: PRINT "R3"; :
LOCATE , 68: PRINT "MOTOR"; : LOCATE , 78: PRINT "Xo"
LOCATE , 50: PRINT ""
LOCATE , 54: PRINT "R2"; : LOCATE , 64: PRINT "Kg"
LOCATE , 60: PRINT ""
LOCATE , 54: PRINT "R1"; : LOCATE , 64: PRINT "Kp"
LOCATE , 60: PRINT ""

```

```
END SUB
```

### SUB MOSTRAR (NUM!)

```

DATO$ = STR$(NUM!)
IF ABS(NUM!) < 1 THEN
  IF LEN(DATO$) <= 9 THEN
    PRINT SPACES(9 - LEN(STR$(NUM!))); NUM!
  ELSE
    IF ABS(NUM!) < 1E-09 THEN
      PRINT USING "#.###^"; NUM!
    ELSE
      EXPONENTE = VAL(RIGHT$(DATO$, 2))
      SIGNO$ = LEFT$(DATO$, 1)
      CEROS$ = STRING$(EXPONENTE - 1, "0")
      VALOR = VAL(LEFT$(DATO$, (9 - EXPONENTE)))
      LARGO = 8 - (EXPONENTE)
      NUMERO$ = ""
      FOR I = 1 TO 9
        IF I < 3 THEN
          NUMERO1$ = MID$(STR$(ABS(VALOR)), I, 1)
          IF NUMERO1$ = "" THEN
            NUMERO1$ = "0"
          END IF
          NUMERO$ = NUMERO$ + NUMERO1$
        END IF
      NEXT I
      PRINT SIGNO$; "."; CEROS$; MID$(NUMERO$, 2, LARGO)
    END IF
  END IF
ELSE
  PRINT SPACES(9 - LEN(STR$(NUM!))); NUM!
END IF
END SUB

```

### SUB MOTORR

```

COLOR FONDO
INICIAL = 64 * PI / 180: FINAL = 116 * PI / 180
INICIAL1 = 245 * PI / 180: FINAL1 = 295 * PI / 180
CIRCLE (172, 135), 73, , , , 1' CIRCULO MAYOR
CIRCLE (172, 292), 35, , , , 1' CIRCULO MENOR
CIRCLE (172, 410), 156, , INICIAL, FINAL, 1'SEMICIRCULO SUPERIOR
CIRCLE (173, 174), 156, , INICIAL1, FINAL1, 1'SEMICIRCULO INFERIO A
CIRCLE (173, 189), 156, , INICIAL1, FINAL1, 1'SEMICIRCULO INFERIO A

```

```
'LADO DERECHO
```

```
LINE (300, 360)-(320, 360)*1
```

```
LINE (320, 360)-(320, 330)*2
```

LINE (320, 330)-(300, 330)'3  
 LINE (320, 345)-(300, 345)'4  
 LINE (300, 345)-(270, 330)'5  
 LINE (300, 330)-(270, 315)'6  
 LINE (300, 345)-(270, 360)'7  
 LINE (300, 360)-(270, 375)'8  
 LINE (270, 360)-(240, 360)'9  
 LINE (270, 375)-(240, 375)'10  
 LINE (240, 375)-(240, 360)'11  
 LINE (270, 330)-(240, 330)'12  
 LINE (270, 315)-(240, 315)'13  
 LINE (270, 315)-(270, 270)'14  
 LINE (270, 270)-(240, 270)'15  
 LINE (250, 270)-(260, 165), , B'16y17  
 LINE (270, 165)-(240, 165)'18  
 LINE (270, 165)-(270, 105)'19  
 LINE (270, 105)-(240, 105)'20  
 LINE (250, 105)-(260, 100), , B'21  
 LINE (104, 165)-(137, 296)'22  
 CIRCLE (90, 345), 15, , , , 1  
 PAINT (90, 345)

#### 'LADO IZQUIERDO

LINE (25, 360)-(45, 360)'1  
 LINE (25, 360)-(25, 330)'2  
 LINE (25, 330)-(45, 330)'3  
 LINE (25, 345)-(45, 345)'4  
 LINE (45, 345)-(75, 330)'5  
 LINE (45, 330)-(75, 315)'6  
 LINE (45, 345)-(75, 360)'7  
 LINE (45, 360)-(75, 375)'8  
 LINE (75, 360)-(105, 360)'9  
 LINE (75, 375)-(105, 375)'10  
 LINE (105, 375)-(105, 360)'11  
 LINE (75, 330)-(105, 330)'12  
 LINE (75, 315)-(105, 315)'13  
 LINE (75, 315)-(75, 270)'14  
 LINE (75, 270)-(105, 270)'15  
 LINE (85, 270)-(95, 165), , B'16y17  
 LINE (75, 165)-(105, 165)'18  
 LINE (75, 165)-(75, 105)'19  
 LINE (75, 105)-(105, 105)'20  
 LINE (85, 105)-(95, 100), , B'21  
 LINE (207, 296)-(239, 165)'22  
 CIRCLE (255, 345), 15, , , , 1  
 PAINT (255, 345)

END SUB

DEFDBL O-Z

#### FUNCTION NUEVO VALOR\$

VIEW PRINT 28 TO 30

LOCATE , 5

PRINT " ; :

PRINT " "

LOCATE , 5

PRINT " ; :

PRINT " "

```

VIEW PRINT 28 TO 30
LOCATE , 20
PRINT "  DESEA REINGRESAR LOS VALORES (S/N)"
SOUND 2000, 5
DO
  CHECK$ = INKEY$
  NUEVOVALOR$ = CHECK$
  LOOP UNTIL CHECK$ = "S" OR CHECK$ = "s" OR CHECK$ = "N" OR CHECK$ = "n"

END FUNCTION

```

### SUB PARAINIM

```

ESP$ = "  "
VIEW PRINT 17 TO 26
LOCATE , 48: PRINT "CONDICIONES  PARAMETROS"
CALL VENTPARA: CALL VENTCONDI
LOCATE , 63: PRINT "kg  : "; " "; SPACES$(7 - LEN(STR$(.12))); .12
LOCATE , 63: PRINT "kp  : "; " "; SPACES$(7 - LEN(STR$(6.74))); 6.74
LOCATE , 63: PRINT "N   : "; " "; "  1:1"
LOCATE , 63: PRINT "R0  : "; " "; SPACES$(7 - LEN(STR$(R0!))); R0!
LOCATE , 63: PRINT "R1  : "; " "; SPACES$(7 - LEN(STR$(R1!))); R1!
LOCATE , 63: PRINT "R2  : "; " "; SPACES$(7 - LEN(STR$(R2!))); R2!
LOCATE , 63: PRINT "R3  : "; " "; SPACES$(7 - LEN(STR$(R3!))); R3!
LOCATE , 63: PRINT "R4  : "; " "; SPACES$(7 - LEN(STR$(R4!))); R4!
CALL VENTPARA

END SUB

```

### SUB PARAMETROSM

```

VIEW PRINT 20 TO 21
LOCATE , 63: PRINT "N   : ";: CALL RELACION(20, 62, REL!): CALL VENTPARA
MAXR0! = R0!
KP1! = KP! * REL!

VIEW PRINT 22 TO 26
LOCATE , 63: PRINT "R1  : ";: CALL VALIDAR(22, 62, R1!)
PARA1:
IF R1! > MAXR0! OR R1! < 10 THEN
  SOUND 2000, 5
  VIEW PRINT 22 TO 26
  LOCATE , 63
  PRINT "R1(max)"; SPACES$(9 - LEN(STR$(MAXR0!))); MAXR0!
  CALL VENTPARA
  DO
  LOOP UNTIL INKEY$ <> ""
  VIEW PRINT 22 TO 26
  LOCATE , 63: PRINT "R1  : ";:
  CALL VALIDAR(22, 62, R1!)
  GOTO PARA1
END IF
CALL VENTPARA

LOCATE , 63: PRINT "R2  : ";: CALL VALIDAR(23, 62, R2!)
PARA2:
IF R2! < 10 THEN
  SOUND 2000, 5

```

```

VIEW PRINT 23 TO 26
LOCATE , 63: PRINT "R2  : " ; :
CALL VALIDAR(23, 62, R2!)
GOTO PARA2
ENDIF
CALL VENTPARA

LOCATE , 63: PRINT "R3  : " ; : CALL VALIDAR(24, 62, R3!)
PARA3:
IF R3! < 10 THEN
  SOUND 2000, 5
  VIEW PRINT 24 TO 26
  LOCATE , 63: PRINT "R3  : " ; :
  CALL VALIDAR(24, 62, R3!)
  GOTO PARA3
ENDIF
CALL VENTPARA

LOCATE , 63: PRINT "R4  : " ; : CALL VALIDAR(25, 62, R4!)
PARA4:
IF R4! < 10 THEN
  SOUND 2000, 5
  VIEW PRINT 25 TO 26
  LOCATE , 63: PRINT "R4  : " ; :
  CALL VALIDAR(25, 62, R4!)
  GOTO PARA4
ENDIF
CALL VENTPARA

K1! = R1! / R0!
K2! = R2! / R0!
K3! = R3! / R0!
K4! = R4! / R0!
END SUB

SUB PROCESANDO

VIEW PRINT 28 TO 30
LOCATE , 5
PRINT "                " ; :
PRINT "                "
LOCATE , 5
PRINT "                " ; :
PRINT "                "

VIEW PRINT 28 TO 30
LOCATE , 16
PRINT "***** PROCESANDO INFORMACION *****"
SOUND 2000, 5

END SUB

SUB RELACION (ROW%, COLUM%, SALIDA!)

DIM REDUC$(5), REDUCC!(5)
REDUC$(1) = " 1:1"
REDUC$(2) = " 3:1"
REDUC$(3) = " 3:1"

```

```
REDUC$(4) = "3*3:1"
REDUC$(5) = " 9:1"
```

```
REDUCC!(1) = 1 / 1
REDUCC!(2) = 1 / SQR(3)
REDUCC!(3) = 1 / 3
REDUCC!(4) = 1 / (3 * SQR(3))
REDUCC!(5) = 1 / 9
```

```
CONTADOR = 1
MARCA% = COLUM% + 8
LOCATE ROW%, MARCA%
PRINT "Ø"
PASO = COLUM% + 12
LOCATE ROW%, PASO
```

```
DO
```

```
  LOCATE ROW%, PASO
  S$ = INKEY$
  SELECT CASE S$
```

```
    CASE CHR$(0) + CHR$(ARRIBA)
      FLAG1 = 1
      IF CONTADOR = 5 THEN
        CONTADOR = 1
      ELSE
        CONTADOR = CONTADOR + 1
      END IF
      PRINT REDUC$(CONTADOR)
```

```
    CASE CHR$(0) + CHR$(ABAJO)
      FLAG1 = 1
      IF CONTADOR = 1 THEN
        CONTADOR = 5
      ELSE
        CONTADOR = CONTADOR - 1
      END IF
      PRINT REDUC$(CONTADOR)
```

```
    CASE CHR$(ENTER)
      IF FLAG1 = 1 THEN
        SALIDA! = REDUCC!(CONTADOR)
      END IF
      LOCATE ROW%, MARCA%
      PRINT " "
      EXIT DO
```

```
  CASE ELSE
    END SELECT
  LOOP UNTIL S$ = CHR$(13)
END SUB
```

### SUB RESULTADOS

```
LINE (337, 30)-(635, 219), , B
LINE (486, 30)-(486, 219)
```

```
END SUB
```

```
DEFSNG O-Z
```

## SUB SALIDAS (ROW%, COLUM%, SALIDA)

```

MARCA% = COLUM% + 8
LOCATE ROW%, MARCA%
PRINT "*"
PASO = COLUM% + 14
LOCATE ROW%, PASO

```

```
DO
```

```

LOCATE ROW%, PASO
S$ = INKEY$
SELECT CASE S$
  CASE CHR$(0) + CHR$(ARRIBA)
    FLAG1 = 1
    CONTADOR = 1
    PRINT "OFF"

  CASE CHR$(0) + CHR$(ABAJO)
    FLAG1 = 1
    CONTADOR = 2
    PRINT "ON"

  CASE CHR$(ENTER)
    IF FLAG1 = 1 THEN
      IF CONTADOR = 1 THEN
        SALIDA = 0
      ELSE
        SALIDA = 1.26
      END IF
    END IF
    LOCATE ROW%, MARCA%
    PRINT " "
    EXIT DO

```

```

  CASE ELSE
  END SELECT
LOOP UNTIL S$ = CHR$(13)
END SUB

```

```
DEFINT C
```

```
'
```

## SUB SALIDAS1 (ROW%, COLUM%, SALIDA)

```

DIM CONT$(5), CONTT!(5), Cd!(5)
CONT$(0) = "0": CONTT!(0) = .173: Cd!(0) = .398
CONT$(1) = "1": CONTT!(1) = .738: Cd!(1) = .341
CONT$(2) = "2": CONTT!(2) = 1.17: Cd!(2) = .333
CONT$(3) = "3": CONTT!(3) = 1.78: Cd!(3) = .298
CONT$(4) = "4": CONTT!(4) = 2.93: Cd!(4) = .299
CONT$(5) = "5": CONTT!(5) = 4.64: Cd!(5) = .317

```

```

MARCA% = COLUM% + 9
LOCATE ROW%, MARCA%
PRINT "*"
PASO = COLUM% + 17
LOCATE ROW%, PASO
CONTADOR = 0

```

DO

```
LOCATE ROW%, PASO
S$ = INKEY$
```

```
SELECT CASE S$
  CASE CHR$(0) + CHR$(ARRIBA)
    FLAG1 = 1
    IF CONTADOR = 5 THEN
      CONTADOR = 0
    ELSE
      CONTADOR = CONTADOR + 1
    END IF
    PRINT CONT$(CONTADOR)

  CASE CHR$(0) + CHR$(ABAJ0)
    FLAG1 = 1
    IF CONTADOR = 0 THEN
      CONTADOR = 5
    ELSE
      CONTADOR = CONTADOR - 1
    END IF
    PRINT CONT$(CONTADOR)

  CASE CHR$(ENTER)
    IF FLAG1 = 1 THEN
      SALIDA = CONTADOR
      Area12! = CONTT!(CONTADOR)
      Cd12! = Cd!(CONTADOR)
    END IF
    LOCATE ROW%, MARCA %
    PRINT " "
    EXIT DO
  CASE ELSE
    END SELECT
LOOP UNTIL S$ = CHR$(13)
END SUB
```

DEFSNG C

SUB VALIDAR (ROW%, COLUM%, NUM!)

```
AUXILIAR! = NUM!
MARCA % = COLUM% + 8
LOCATE ROW%, MARCA %
PRINT "Ø"
PASO1 = COLUM% + 9
PASO = COLUM% + 17
DO
  LOCATE ROW%, PASO
  DO
    CHECK$ = INKEY$
    LOOP UNTIL CHECK$ <> ""
    NUMERO$ = CHECK$

  SELECT CASE NUMERO$

    CASE CHR$(CERO) TO CHR$(NUEVE)
    IF FLAG4 = 0 THEN
      IF FLAG3 = 0 THEN
        LOCATE ROW%, PASO1
```

```

        PRINT "      "
        LOCATE ROW%, PASO
        FLAG3 = 1
    END IF
    IF CONTADOR = 0 AND NUMERO$ = "0" THEN
        FLAG4 = 1
    END IF
    IF CONTADOR < 8 THEN
        FLAG2 = 1
        CIFRA = CIFRA + 1
        CONTADOR = CONTADOR + 1
        DATA$ = DATA$ + CHECK$
        LOCATE ROW%, PASO - CONTADOR
        PRINT DATA$
    ELSE
        BEEP
    END IF
ELSE
    BEEP
END IF

    CASE CHR$(MENOS)
    IF FLAG3 = 0 THEN
        LOCATE ROW%, PASO1
        PRINT "      "
        LOCATE ROW%, PASO
        FLAG3 = 1
    END IF
    IF FLAG2 = 0 AND FLAG1 = 0 THEN
        CONTADOR = CONTADOR + 1
        FLAG2 = 1
        CIFRA = CIFRA + 1
        DATA$ = DATA$ + CHECK$
        LOCATE ROW%, PASO - CONTADOR
        PRINT DATA$
    ELSE
        BEEP
    END IF

    CASE CHR$(PUNTO)
    FLAG4 = 0
    IF FLAG3 = 0 THEN
        LOCATE ROW%, PASO1
        PRINT "      "
        LOCATE ROW%, PASO
        FLAG3 = 1
    ELSE
    END IF
    IF FLAG1 = 0 THEN
        CONTADOR = CONTADOR + 1
        FLAG1 = 1
        CIFRA = CIFRA + 1
        SEGURO = CIFRA
        DATA$ = DATA$ + CHECK$
        LOCATE ROW%, PASO - CONTADOR
        PRINT DATA$
    ELSE
        BEEP
    END IF

    CASE CHR$(DELETE)
    IF CIFRA > 0 THEN

```

```

IF CIFRA = SEGURO THEN
    FLAG1 = 0
END IF
CIFRA = CIFRA - 1
CONTADOR = CONTADOR - 1
DATA$ = LEFT$(DATA$, CIFRA)
IF CIFRA = 0 THEN
    DATA1$ = "0"
    LOCATE ROW%, PASO1
    PRINT "    "
    LOCATE ROW%, PASO - CONTADOR - 1
    PRINT DATA1$
ELSE
    LOCATE ROW%, PASO1
    PRINT "    "
    LOCATE ROW%, PASO - CONTADOR
    PRINT DATA$
END IF
IF CONTADOR = 0 THEN
    FLA1 = 0: FLAG2 = 0: FLAG3 = 0: FLAG4 = 0
END IF
ELSE
    BEEP
END IF

    CASE CHR$(ENTER)
IF CONTADOR = 0 THEN
    LOCATE ROW%, MARCA %
    PRINT " "
    NUM! = AUXILIAR!
    EXIT DO
ELSE
    NUM! = VAL(DATA$)
    LOCATE ROW%, MARCA %
    PRINT " "
    EXIT DO
END IF

    CASE CHR$(0) + CHR$(ABAJOS)
IF CONTADOR = 0 THEN
    LOCATE ROW%, MARCA %
    PRINT " "
    NUM! = AUXILIAR!
    EXIT DO
ELSE
    NUM! = VAL(DATA$)
    LOCATE ROW%, MARCA %
    PRINT " "
    EXIT DO
END IF

    CASE ELSE
    BEEP
END SELECT
LOOP
END SUB

```

**SUB VENTANAS**

SCREEN 12  
COLOR FONDO

LINE (1, 1)-(639, 479), , B 'BORDE EXTERIOR  
LINE (1, 415)-(639, 415)  
END SUB

#### SUB VENTANAS1

SCREEN 12

\*\*\*\*\*

\* RESTITUYE LAS VENTANAS \*

\*\*\*\*\*

LINE (5, 4)-(325, 205), , B  
LINE (5, 210)-(325, 411), , B  
END SUB

#### SUB VENTCONDI

LINE (337, 250)-(482, 410), , B  
LINE (402, 270)-(477, 405), , B  
END SUB

#### SUB VENTPARA

LINE (490, 250)-(635, 410), , B 'VENTANA DE PARAMETROS  
LINE (555, 270)-(630, 405), , B  
END SUB

**FIN SUBROUTINAS MOTOMATIC**

*TANQUES.BAS*

## TANQUE.BAS

---



---

### SIMULACION DEL LOS TANQUES ACOPLADOS

---



---

```

DECLARE SUB CAUDALES (H1!, H2!)
DECLARE SUB COEF1 ()
DECLARE FUNCTION SIMULACION! ()
DECLARE SUB DISCRETIZAR (Ac!(), Bc!(), Ad!(), Bd!())
DECLARE SUB MOSTRAR (NUM!)
DECLARE SUB COEF (Ac!(), Bc!())
DECLARE SUB TANQUES1 ()
DECLARE SUB ESCALA (LINEA%, TSIM!, GRAFICO%, DATOS!(), MAXMIN!())
DECLARE SUB GRAFICOS (LINEA%, TSIM!, GRAFICO%, DATOS!(), MAXMIN!())
DECLARE SUB COEFICIENTES (H1!, H2!, Ac!())
DECLARE SUB RESTA (n%, M%, A!(), B!(), C!())
DECLARE SUB ERROR1 ()
DECLARE SUB CUADRATICO ()
DECLARE SUB INVERSA (Ma!(), Inv!(), DET!)
DECLARE SUB TRANS (n!, M!, A!(), AT!())
DECLARE SUB MAX (n%, l!, DATOS!(), MAXMIN!())
DECLARE SUB GRAFTANQUES (ROW%, COLUM%, SALIDA%)
DECLARE SUB ERROR0 ()
DECLARE SUB PROCESANDO ()
DECLARE SUB MENUS ()
DECLARE SUB ERRORM1 ()
DECLARE SUB MENUTANQUES ()
DECLARE SUB SALIDAS (ROW%, COLUM%, SALIDA!)
DECLARE SUB SALIDAS1 (ROW%, COLUM%, SALIDA!)
DECLARE SUB VALIDAR (ROW%, COLUM%, NUM!)
DECLARE SUB MULT (n%, M%, l%, A!(), B!(), C!())
DECLARE SUB SUMA (n%, M%, A!(), B!(), C!())
DECLARE SUB VENTANAS1 ()
DECLARE SUB PARAINIT ()
DECLARE SUB CONDINIT ()
DECLARE SUB CONDICIONEST ()
DECLARE SUB PARAMETROST ()
DECLARE SUB VENTCONDI ()
DECLARE SUB VENTPARA ()
DECLARE SUB VENTANAS ()
DECLARE SUB RESULTADOS ()
DECLARE FUNCTION FACTORIAL! (M%)
DECLARE FUNCTION NUEVOVALOR$ ()

```

---



---

#### Definición de Variables Globales

---



---

```

COMMON SHARED A1!, A2!, Area1!, Area2!, Area12!
COMMON SHARED Qi1!, Qi2!, Vi1!, Vi2!
COMMON SHARED Cd1!, Cd2!, Cd12!, d1!, b11!, b22!
COMMON SHARED TM!, H1i!, H2i!, H1f!, H2f!, ITERT!, PI
COMMON SHARED BLANCO, AZUL, BLANCOI, MAGENTA, NEGRO, GRISI
COMMON SHARED CERO, NUEVE, ENTER, PUNTO, MENOS, DELETE
COMMON SHARED ARRIBA, ABAJO, DERECHA, IZQUIERDA, FONDO, ROJO
COMMON SHARED F1, F2, F3, F4, F5, F6, F7, F8, F9, F10
COMMON SHARED MAXIM, MINIM

```

```

A1! = 378.3: A2! = 378.3: Area1! = 1.26: Area2! = 1.26: Area12! = .173
Cd1! = .395: Cd2! = .395: Cd12! = .398: b11! = 13.4: b22! = 18.51

```

ARRIBA = 72: ABAJO = 80  
 BLANCO = 7: AZUL = 1: BLANCOI = 11: MAGENTA = 5: NEGRO = 0: GRISI = 15  
 FONDO = 14: ROJO = 4: CERO = 48: NUEVE = 57: ENTER = 13: PUNTO = 46: MENOS = 45  
 DELETE = 8: ESCAPE = 27: DERECHA = 77: IZQUIERDA = 75  
 F1 = 59: F2 = 60: F3 = 61: F4 = 62  
 F5 = 63: F6 = 64: F7 = 65: F8 = 66  
 d1! = .25

\$\$\$DYNAMIC

---



---

**MODULO PRINCIPAL**

---



---

INICIOT:

REDIM Ac!(2, 2), Bc!(2, 2), Cl!(2, 2), g!(2, 2)  
 REDIM AUX!(3), Gccn!(4), Gccd!(4), Q!(2, 2), R!(2, 2)  
 REDIM COMP\$(4)

COMP\$(1) = " TANQUES ACOPLADOS "  
 COMP\$(2) = " REG. CUADRAT. LINEAL "

Qi1max! = 111.4  
 Qi2max! = 147  
 GRAFT1% = 1: GRAFT2% = 2  
 FLAG1 = 0: FLAG2 = 0  
 FLAG3 = 0: FLAG4 = 0  
 FLAG5 = 0: AUTO = 0  
 COMPEN = 1  
 SCREEN 12

INICIOT1:

CLS  
 CALL MENUTANQUES  
 CALL VENTANAS  
 CALL VENTANAS1  
 CALL RESULTADOS  
 CALL PARAINIT  
 CALL CONDINIT  
 VIEW PRINT 2 TO 3  
 LOCATE , 51: PRINT COMP\$(1)  
 VIEW PRINT 3 TO 4  
 LOCATE , 51: PRINT COMP\$(2)

IF FLAG1 = 1 THEN

VIEW PRINT 5 TO 15  
 FOR I = 1 TO 9  
 style="padding-left: 4em;">LOCATE , 44  
 style="padding-left: 4em;">PRINT " " "  
 NEXT I  
 CALL RESULTADOS  
 Bc!(1, 1) = d1! \* b11! / 378.4: Bc!(1, 2) = 0  
 Bc!(2, 2) = d1! \* b22! / 378.4: Bc!(2, 1) = 0  
 CALL COEF1  
 CALL COEFICIENTES(H1i!, H2i!, Ac!())  
 CALL COEF(Ac!(), Bc!())

END IF

INICIOT2:

CALL MENUTANQUES

DO

```
DO
CHECK$ = INKEY$
LOOP UNTIL CHECK$ <> ""
OPCION$ = CHECK$
```

```
SELECT CASE OPCION$
```

```
    CASE CHR$(0) + CHR$(F1)      "Ingreso de Condiciones"
```

```
        FLAG1 = 1
        FLAG4 = 1
        FLAG5 = 0
        CALL CONDICIONEST
        VIEW PRINT 5 TO 15
        FOR I = 1 TO 9
            LOCATE , 44
            PRINT "                "
        NEXT I
        Bc!(1, 1) = d1! * b11! / 378.4: Bc!(1, 2) = 0
        Bc!(2, 2) = d1! * b22! / 378.4: Bc!(2, 1) = 0

        CALL RESULTADOS
        CALL CAUDALES(H1f!, H2f!)
        CALL COEFICIENTES(H1i!, H2i!, Ac!())
        CALL COEF1
        CALL COEF(Ac!(), Bc!())

        REDIM Hini!(2), Hfin!(2), Ref!(2)
        Hini!(1) = H1i!: Hini!(2) = H2i!
        Hfin!(1) = H1f!: Hfin!(2) = H2f!
        Ref!(1) = Hfin!(1) - Hini!(1)
        Ref!(2) = Hfin!(2) - Hini!(2)
```

```
    CASE CHR$(0) + CHR$(F2)      "Ingreso de Parámetros"
```

```
        FLAG2 = 1
        FLAG4 = 1
        FLAG5 = 0
        CALL PARAMETROST
```

```
        VIEW PRINT 5 TO 15
```

```
        FOR I = 1 TO 9
            LOCATE , 44
            PRINT "                "
        NEXT I
```

```
        CALL RESULTADOS
```

```
        IF FLAG1 = 1 THEN
```

```
            CALL CAUDALES(H1f!, H2f!)
            Bc!(1, 1) = d1! * b11! / 378.4: Bc!(1, 2) = 0
            Bc!(2, 2) = d1! * b22! / 378.4: Bc!(2, 1) = 0
            CALL COEF1
            CALL COEFICIENTES(H1i!, H2i!, Ac!())
            CALL COEF(Ac!(), Bc!())
```

```
        END IF
```

```
    CASE CHR$(0) + CHR$(F3)      "Ingreso del Regulador Cuadrático Lineal"
```

```
        IF FLAG1 = 1 THEN
            GOTO COMPTANQUES
        ELSE
            CALL ERRORM1
            CALL MENUTANQUES
        END IF
```

```

CASE CHR$(0) + CHR$(F4)          "Inicio Cálculos para la simulación"

    IF FLAG4 = 1 AND FLAG3 = 1 THEN
        FLAG6 = 1
        GOTO REGULADORLINEAL
    END IF
SIMULA:
    IF FLAG1 = 1 AND FLAG3 = 1 THEN
        FLAG5 = 1
        GOTO TANQUES
    ELSE
        CALL ERRORM1
        CALL MENUTANQUES
    END IF

CASE CHR$(0) + CHR$(F5)          "Presento simulación dinámica"
    IF FLAG5 = 1 THEN
        GOTO SIMDINTAN
    ELSE
        CALL ERROR0
        CALL MENUTANQUES
    END IF

CASE CHR$(0) + CHR$(F6)          "Presento Gráficos"
    IF FLAG5 = 1 THEN
        GOTO GRAFICOST
    ELSE
        CALL ERROR0
        CALL MENUTANQUES
    END IF

CASE CHR$(0) + CHR$(F7) "Salida del programa"
    GOTO SALIDAT

CASE CHR$(ESCAPE)
    IF FLAG1 = 1 THEN
        VIEW PRINT 5 TO 16
        FOR I = 1 TO 11
            LOCATE , 44
            PRINT " "
        NEXT I
        CALL RESULTADOS
        Bcl(1, 1) = d1! * b11! / 378.4: Bcl(1, 2) = 0
        Bcl(2, 2) = d1! * b22! / 378.4: Bcl(2, 1) = 0
        CALL COEF1
        CALL COEFICIENTES(H1i!, H2i!, Acl())
        CALL COEF(Acl(), Bcl())
    END IF

CASE ELSE
    BEEP
END SELECT
LOOP
CHAIN "A:\SIMULA\MENU"

```

---



---

**Ingreso Regulator Cuadrático Lineal.**

---



---

```

COMPTANQUES:
VIEW PRINT 5 TO 16
FOR I = 1 TO 11
    LOCATE , 44
    PRINT " "

```

```

NEXT I
CALL RESULTADOS
FLAG3 = 1
FLAG4 = 0
FLAG5 = 0
CALL MENUS
VIEW PRINT 5 TO 14
LOCATE , 45: PRINT " (Q+) ^ (R+)      "
RESULTADOS
FOR I = 1 TO 2
  FOR J = 1 TO 2
    LOCATE , 44
    PRINT "Q("; RIGHT$(STR$(I), 1); ", "; RIGHT$(STR$(J), 1); ")": "; :
    MOSTRAR (Q!(I, J))
  NEXT J
NEXT I
VIEW PRINT 11 TO 15
FOR I = 1 TO 2
  FOR J = 1 TO 2
    LOCATE , 44
    PRINT "R("; RIGHT$(STR$(I), 1); ", "; RIGHT$(STR$(J), 1); ")": "; :
    MOSTRAR (R!(I, J))
  NEXT J
NEXT I
CALL RESULTADOS

VIEW PRINT 6 TO 10
FOR I = 1 TO 2
  FOR J = 1 TO 2
    LOCATE , 63
    PRINT "g("; RIGHT$(STR$(I), 1); ", "; RIGHT$(STR$(J), 1); ")": "; :
    MOSTRAR (g!(I, J))
  NEXT J
NEXT I
VIEW PRINT 11 TO 15
LOCATE , 63: PRINT "H1r(0):"; : MOSTRAR (H1refi)
LOCATE , 63: PRINT "H2r(0):"; : MOSTRAR (H2refi)
LOCATE , 63: PRINT "H1r(T):"; : MOSTRAR (H1reff)
LOCATE , 63: PRINT "H2r(T):"; : MOSTRAR (H2refF)
CALL RESULTADOS

COMPTANQUES1:
K% = 6
VIEW PRINT 6 TO 10
FOR I = 1 TO 2
  FOR J = 1 TO 2
    LOCATE , 44
    PRINT "Q("; RIGHT$(STR$(I), 1); ", "; RIGHT$(STR$(J), 1); ")": "; :
Q1:
    IF I = 2 AND J = 1 THEN
      Q!(I, J) = Q!(J, I)
      MOSTRAR (Q!(I, J))
    ELSE
      CALL VALIDAR(K%, 43, Q!(I, J)): CALL RESULTADOS
    END IF
    IF Q!(I, J) < 0 THEN
      SOUND 4200, 8
      GOTO Q1
    END IF
    K% = K% + 1
  NEXT J
NEXT I

```

```

K% = 11
VIEW PRINT 11 TO 15
FOR I = 1 TO 2
  FOR J = 1 TO 2
    LOCATE , 44
    PRINT "R("; RIGHT$(STR$(I), 1); ", "; RIGHT$(STR$(J), 1); "):"; :
R1:
    IF I = 2 AND J = 1 THEN
      R!(I, J) = R!(J, I)
      MOSTRAR (R!(I, J))
    ELSE
      CALL VALIDAR(K%, 43, R!(I, J)): CALL RESULTADOS
    END IF
    IF R!(I, J) < 0 THEN
      SOUND 4200, 8
      GOTO R1
    END IF
    K% = K% + 1
  NEXT J
NEXT I

DO
  OTRO$ = NUEVOVALOR$
LOOP UNTIL OTRO$ = "S" OR OTRO$ = "s" OR OTRO$ = "N" OR OTRO$ = "n"

IF OTRO$ = "S" OR OTRO$ = "s" THEN
  CALL MENUS
  GOTO COMPTANQUES1
ELSE
  GOTO REGULADORLINEAL
END IF
REGULADORLINEAL1:
GOTO INICIOT2

```

---



---

**CALCULO DE LOS VALORES PARA LA SIMULACION**

---



---

TANQUES:

```

REDIM TANQUE!(8, ITERT!), MAXIMOS!(8, 2)
REDIM Xaux!(2, 1), Xaux1!(2, 1), bu!(2, 1)
REDIM NIVELHIGH(2), Uaux!(2, 1), Ad!(2, 2), Bd!(2, 2)

```

```

'AUTO = SIMULACION
AUTO = 0
CALL CAUDALES(H1i!, H2i!)
CALL PROCESANDO

```

```

FOR K = 1 TO ITERT!
  VIEW PRINT 29 TO 30
  LOCATE , 39: PRINT K
  IF K = 1 THEN
    FOR I = 1 TO 2
      Xaux!(I, 1) = 0
    NEXT
  END IF

```

```

FOR J = 1 TO 2

```

```

*****
' CALCULO DE LA SALIDA
*****
TANQUE!(J, K) = Xaux!(J, 1)

*****
' CALCULO DEL ERROR
*****
IF FLAG3 = 1 THEN
  REGUL! = 0
  FOR I = 1 TO 2
    REGUL! = REGUL! + g!(J, I) * TANQUE!(I, K)
  NEXT I
  TANQUE!(J + 2, K) = Ref!(J) - REGUL!
ELSE
  TANQUE!(J + 2, K) = Ref!(J) - TANQUE!(J, K)
END IF

*****
' ACONDICIONO LA SENAL
*****
TANQUE!(J + 6, K) = d1! * TANQUE!(J + 2, K)

*****
' CALCULO CAUDALES DE LAS BOMBAS
*****

IF J = 1 THEN
  IF caudal! + Qi1! > Qi1max! THEN
    caudal! = Qi1max! - Qi1!
  ELSE
    caudal! = TANQUE!(J + 6, K) * b11!
  END IF
ELSE
  IF caudal! + Qi2! > Qi2max! THEN
    caudal! = Qi2max! - Qi2!
  ELSE
    caudal! = TANQUE!(J + 6, K) * b22!
  END IF
END IF

TANQUE!(J + 4, K) = caudal!
Uaux!(J, 1) = TANQUE!(J + 4, K)
NEXT J

IF AUTO = 0 THEN
  IF K = 1 THEN
    Bc!(1, 1) = 1 / 378.4: Bc!(1, 2) = 0
    Bc!(2, 2) = 1 / 378.4: Bc!(2, 1) = 0
    CALL COEFICIENTES(H1i!, H2i!, Ac!())
    CALL DISCRETIZAR(Ac!(), Bc!(), Ad!(), Bd!())
  END IF
ELSE
  Bc!(1, 1) = 1 / 378.4: Bc!(1, 2) = 0
  Bc!(2, 2) = 1 / 378.4: Bc!(2, 1) = 0
  CALL COEFICIENTES(TANQUE!(1, K), TANQUE!(2, K), Ac!())
  CALL DISCRETIZAR(Ac!(), Bc!(), Ad!(), Bd!())
END IF

CALL MULT(2, 2, 1, Ad!(), Xaux!(), Xaux1!())
CALL MULT(2, 2, 1, Bd!(), Uaux!(), bu!())
CALL SUMA(2, 1, Xaux1!(), bu!(), Xaux!())

```

NEXT K

\*\*\*\*\*

'Obtendo los valores reales de la simulación

\*\*\*\*\*

FOR K = 1 TO ITERT!

TANQUE!(1, K) = Hini!(1) + TANQUE!(1, K)

TANQUE!(2, K) = Hini!(2) + TANQUE!(2, K)

TANQUE!(5, K) = Qi1! + TANQUE!(5, K)

TANQUE!(6, K) = Qi2! + TANQUE!(6, K)

TANQUE!(7, K) = Vi1! + TANQUE!(7, K)

TANQUE!(8, K) = Vi2! + TANQUE!(8, K)

TANQUE!(3, K) = TANQUE!(7, K) / d1!

TANQUE!(4, K) = TANQUE!(8, K) / d1!

NEXT K

CALL MAX(8, ITERT!, TANQUE!(), MAXIMOS!())

VIEW PRINT 28 TO 30

LOCATE , 6

PRINT " \*\*\*\*\* CALCULOS PARA LA SIMULACION LISTA \*\*\*\*\*"

LOCATE , 6

PRINT " "

SOUND 4200, 8

DO

CHECK\$ = INKEY\$

LOOP UNTIL CHECK\$ <> ""

CALL MENUTANQUES

GOTO INICIOT2

\*\*\*\*\*

\*\*\*\*\* SIMULACION DINAMICA DE LOS TANQUES\*\*\*\*\*

\*\*\*\*\*

SIMDINTAN:

CLS

IF MAXIMOS!(1, 1) > MAXIMOS!(2, 1) THEN

MAXIM = MAXIMOS!(1, 1)

ELSE

MAXIM = MAXIMOS!(2, 1)

END IF

IF MAXIMOS!(1, 2) < MAXIMOS!(2, 2) THEN

MINIM = MAXIMOS!(1, 2)

ELSE

MINIM = MAXIMOS!(2, 2)

END IF

MAXIM = INT(MAXIM) + 1

MINIM = INT(MINIM) - 1

CALL PROCESANDO

CALL VENTANAS

CALL TANQUES1

H11 = -.02

H12 = -1.085

H21 = .025

H22 = 1.09

COLOR AZUL

LINE (120, 364)-(520, 364)

VIEW (121, 45)-(519, 364)

WINDOW (-1.1, MINIM)-(1.1, MAXIM)

NIVEL1! = MINIM

```

NIVEL2! = MINIM
DO
  COLOR AZUL
  IF NIVEL1! < H1i! THEN
    LINE (H11, NIVEL1!)-(H12, NIVEL1!)
    NIVEL1! = NIVEL1! + .01
  END IF
  IF NIVEL2! < H2i! THEN
    LINE (H21, NIVEL2!)-(H22, NIVEL2!)
    NIVEL2! = NIVEL2! + .01
  END IF
  COLOR FONDO
  VIEW PRINT 25 TO 26
  LOCATE , 11: PRINT USING "H1= ##.#"; NIVEL1!; :
  LOCATE , 63: PRINT USING "H2= ##.#"; NIVEL2!
LOOP WHILE NIVEL1! < H1i! OR NIVEL2! < H2i!
FOR I = 1 TO 1000: NEXT I

COLOR AZUL
FOR K = 2 TO ITERT!

  CHECK$ = INKEY$
  IF CHECK$ = CHR$(ESCAPE) THEN
    GOTO FINAL2
  END IF

  NIVEL1! = TANQUE!(1, K)
  NIVEL11! = TANQUE!(1, K - 1)
  NIVEL2! = TANQUE!(2, K)
  NIVEL22! = TANQUE!(2, K - 1)

  COLOR AZUL
  IF NIVEL1! < NIVEL11! THEN
    LINE (H11, NIVEL1!)-(H12, NIVEL11!), 0, BF
  ELSE
    LINE (H11, NIVEL1!)-(H12, NIVEL11!), 1, BF
  END IF

  IF NIVEL2! < NIVEL22! THEN
    LINE (H21, NIVEL2!)-(H22, NIVEL22!), 0, BF
  ELSE
    LINE (H21, NIVEL2!)-(H22, NIVEL22!), 1, BF
  END IF
  FOR I = 1 TO 30000: NEXT I
  COLOR FONDO
  VIEW PRINT 25 TO 26
  LOCATE , 11: PRINT USING "H1= ##.#"; NIVEL1!; :
  LOCATE , 63: PRINT USING "H2= ##.#"; NIVEL2!
NEXT K

FINAL2:
DO
COLOR FONDO
VIEW PRINT 28 TO 30
LOCATE , 13
PRINT "***** FIN (PRESIONE ENTER PARA CONTINUAR) *****"
CHECK$ = INKEY$
LOOP UNTIL CHECK$ = CHR$(ENTER)
SOUND 4200, 8
VIEW
WINDOW
GOTO INCIOT1

```

```

*****
***** RUTINA GRAFICA PARA DE LOS TANQUES*****
*****
GRAFICOST:
REDIM REDUC$(8)
REDUC$(1) = " H1(t)"
REDUC$(2) = " H2(t)"
REDUC$(3) = " e1(t)"
REDUC$(4) = " e2(t)"
REDUC$(5) = "Qi1(t)"
REDUC$(6) = "Qi2(t)"
REDUC$(7) = " V1(t)"
REDUC$(8) = " V2(t)"

SOUND 4200, 8
VIEW PRINT 5 TO 17
FOR I = 1 TO 11
LOCATE , 44
PRINT " "
NEXT I

VIEW PRINT 5 TO 10:LOCATE , 46: PRINT "Graf1 : "; REDUC$(GRAFT1%)
VIEW PRINT 5 TO 10:LOCATE , 65: PRINT "Graf2 : "; REDUC$(GRAFT2%)
CALL RESULTADOS

VIEW PRINT 6 TO 15
LOCATE , 44
PRINT "Escala tiempo Ø"; :
PRINT USING "###.## Seg/div"; ITERT! * TM! / 10
LOCATE , 44
PRINT "Tiempo simul. Ø"; :
PRINT USING "####.## Seg"; ITERT! * TM!

VIEW PRINT 5 TO 10
LOCATE , 46: PRINT "Graf1 :"; : CALL GRAFTANQUES(5, 42, GRAFT1%)
VIEW PRINT 5 TO 10
LOCATE , 65: PRINT "Graf2 :"; : CALL GRAFTANQUES(5, 61, GRAFT2%)

CALL ESCALA(1, ITERT!, GRAFT1%, TANQUE!(), MAXIMOS!())
CALL ESCALA(2, ITERT!, GRAFT2%, TANQUE!(), MAXIMOS!())

VIEW (6, 5)-(324, 204)
CLS
IF MAXIMOS!(GRAFT1%, 1) <> MAXIMOS!(GRAFT1%, 2) THEN
CALL GRAFICOS(1, ITERT!, GRAFT1%, TANQUE!(), MAXIMOS!())
END IF
VIEW (6, 211)-(324, 410)
CLS
IF MAXIMOS!(GRAFT2%, 1) <> MAXIMOS!(GRAFT2%, 2) THEN
CALL GRAFICOS(2, ITERT!, GRAFT2%, TANQUE!(), MAXIMOS!())
END IF

' *****
' Desplazamiento sobre las curvas graficadas
' *****
VIEW PRINT 28 TO 30
LOCATE , 5
PRINT " CON LAS FLECHAS ME DESPLAZO EN LAS CURVAS "
LOCATE , 5
PRINT " ***** FIN (PRESIONE ESCAPE PARA CONTINUAR) ***** "
K = 0

```

```

DO
  S$ = INKEY$
  SELECT CASE S$
    CASE CHR$(0) + CHR$(DERECHA)
      IF K = ITERT! THEN
        K = 0
      ELSE
        K = K + 1
      END IF

    CASE CHR$(0) + CHR$(IZQUIERDA)
      IF K = 0 THEN
        K = ITERT!
      ELSE
        K = K - 1
      END IF

    CASE CHR$(ESCAPE)
      EXIT DO

    CASE ELSE
      END SELECT

  VIEW PRINT 14 TO 17
  LOCATE , 56
  PRINT "Iter 0"; :
  PRINT USING "####"; K
  LOCATE , 69
  MOSTRAR (TANQUE!(GRAFT2%, K))
  VIEW PRINT 15 TO 17
  LOCATE , 49
  MOSTRAR (TANQUE!(GRAFT1%, K))
  LOOP UNTIL S$ = CHR$(ESCAPE)

VIEW
WINDOW
GOTO INCIOT2

*****
!*  SALIDA DE LA EJECUCION DE LOS TANQUES *
*****
SALIDAT:
VIEW PRINT 28 TO 30
LOCATE , 5
PRINT "           "; :
PRINT "           "
LOCATE , 5
PRINT "           "; :
PRINT "           "

VIEW PRINT 28 TO 30
LOCATE , 20
PRINT "ESTA SEGURO QUE DESEA SALIR DEL PROGRAMA (S/N)"
SOUND 4200, 8
DO
  CHECK$ = INKEY$
  LOOP UNTIL CHECK$ = "S" OR CHECK$ = "s" OR CHECK$ = "N" OR CHECK$ = "n"

IF CHECK$ = "S" OR CHECK$ = "s" THEN
  VIEW PRINT
  CLS
  END

```

```
ELSE
  GOTO INICIOT1
END IF
```

```
*****
' CALCULO INTERACTIVO DEL REGULADOR CUADRATICO LINEAL
*****
```

```
REGULADORLINEAL:
```

```
REDIM MT!(2, 2), BT!(2, 2), AT!(2, 2), K1!(2, 2)
REDIM BTK!(2, 2), KA1!(2, 2), BTK!(2, 2), BTKB!(2, 2), C1T!(2, 2)
REDIM B1I!(2, 2), B1II!(2, 2), BTKA1!(2, 2), C1!(2, 2), G1!(2, 2)
REDIM C1TG!(2, 2), ATK!(2, 2), ATKA!(2, 2), D!(2, 2), K0!(2, 2)
REDIM KA!(2, 2, ITERT! + 1), G0!(2, 2, ITERT!)
```

```
Bc!(1, 1) = d1! * b11! / 378.4: Bc!(1, 2) = 0
```

```
Bc!(2, 2) = d1! * b22! / 378.4: Bc!(2, 1) = 0
```

```
M!(1, 1) = 0: M!(1, 2) = 0
```

```
M!(2, 2) = 0: M!(2, 1) = 0
```

```
CALL COEFICIENTES(H1i!, H2i!, Ac!())
```

```
CALL DISCRETIZAR(Ac!(), Bc!(), Ad!(), Bd!())
```

```
CALL CUADRATICO
```

```
FOR I = 1 TO 2
```

```
  FOR J = 1 TO 2
```

```
    KA!(I, J, ITERT! + 1) = 0
```

```
  NEXT J
```

```
NEXT I
```

```
CALL TRANS(2, 2, M!(), MT!())
```

```
CALL TRANS(2, 2, Bd!(), BT!())
```

```
CALL TRANS(2, 2, Ad!(), AT!())
```

```
FOR K = ITERT! TO 1 STEP -1
```

```
  VIEW PRINT 29 TO 30
```

```
  LOCATE , 39: PRINT K
```

```
  FOR I = 1 TO 2
```

```
    FOR J = 1 TO 2
```

```
      KA1!(I, J) = KA!(I, J, K + 1)
```

```
    NEXT J
```

```
  NEXT I
```

```
CALL MULT(2, 2, 2, BT!(), KA1!(), BTK!())
```

```
CALL MULT(2, 2, 2, BTK!(), Bd!(), BTKB!())
```

```
CALL SUMA(2, 2, BTKB!(), R!(), B1!())
```

```
CALL INVERSA(B1!(), B1I!(), DET!)
```

```
'CHEQUEO SI ES MATRIZ SINGULAR
```

```
IF DET! = 0 THEN
```

```
  K% = 10
```

```
  CALL ERRORT1
```

```
  VIEW PRINT 10 TO 14
```

```
  FOR I = 1 TO 2
```

```
FOR J = 1 TO 2
```

```
  LOCATE , 44
```

```
  PRINT "R("; RIGHT$(STR$(I), 1); ", "; RIGHT$(STR$(J), 1); ")": ;
```

```
  CALL VALIDAR(K%, 43, R!(I, J)): CALL RESULTADOS
```

```
  K% = K% + 1
```

```
NEXT J
```

```
NEXT I
```

```
  GOTO REGULADORLINEAL
```

```
END IF
```

```
CALL MULT(2, 2, 2, BTK!(), Ad!(), BTKA!())
```

```

CALL SUMA(2, 2, MT!(), BTKA!(), C1!())
CALL MULT(2, 2, 2, B1!(), C1!(), G1!())
FOR I = 1 TO 2
  FOR J = 1 TO 2
    GO!(I, J, K) = G1!(I, J)
  NEXT J
NEXT I
CALL TRANS(2, 2, C1!(), C1T!())
CALL MULT(2, 2, 2, C1T!(), G1!(), C1TG!())
CALL MULT(2, 2, 2, AT!(), KA1!(), ATK!())
CALL MULT(2, 2, 2, ATK!(), Ad!(), ATKA!())
CALL SUMA(2, 2, Q!(), ATKA!(), D!())
CALL RESTA(2, 2, D!(), C1TG!(), K0!())
FOR I = 1 TO 2
  FOR J = 1 TO 2
    KA!(I, J, K) = K0!(I, J)
  NEXT J
NEXT I
NEXT K

VIEW PRINT 28 TO 30
LOCATE , 5
PRINT "                " ; :
PRINT "                "
LOCATE , 5
PRINT "                " ; :
PRINT "                "
VIEW PRINT 28 TO 30
LOCATE , 9
PRINT " ***** Regulator Cuadratico Listo *****"
SOUND 4200, 8

!*****
' IMPRIMO LA GANANCIA DE REALIMENTACION OPTIMA
!*****
IF FLAG6 = 1 THEN
VIEW PRINT 5 TO 16
FOR I = 1 TO 11
  LOCATE , 44
  PRINT "                "
NEXT I
CALL RESULTADOS
FLAG3 = 1
FLAG4 = 0
FLAG5 = 0
CALL MENUS
VIEW PRINT 5 TO 14
LOCATE , 45: PRINT " (Q+) ^ (R+) "
RESULTADOS
FOR I = 1 TO 2
  FOR J = 1 TO 2
    LOCATE , 44
    PRINT "Q("; RIGHT$(STR$(I), 1); ", "; RIGHT$(STR$(J), 1); ");": :
    MOSTRAR (Q!(I, J))
  NEXT J
NEXT I
VIEW PRINT 11 TO 15
FOR I = 1 TO 2
  FOR J = 1 TO 2
    LOCATE , 44
    PRINT "R("; RIGHT$(STR$(I), 1); ", "; RIGHT$(STR$(J), 1); ");": :
    MOSTRAR (R!(I, J))
  NEXT J
NEXT I

```

```

NEXT J
NEXT I
CALL RESULTADOS
END IF

VIEW PRINT 6 TO 10
FOR I = 1 TO 2
  FOR J = 1 TO 2
    LOCATE , 63
    g!(I, J) = G0!(I, J, 1)
    PRINT "g("; RIGHT$(STR$(I), 1); ", "; RIGHT$(STR$(J), 1); ")"; :
    MOSTRAR (g(I, J))
    CALL RESULTADOS
  NEXT J
NEXT I

VIEW PRINT 11 TO 15
CALL CAUDALES(H1i!, H2i!)
H1refi = (Vi1! + (g!(1, 1) * H1i! * d1!) + (g!(1, 2) * H2i! * d1!)) / d1!
H2refi = (Vi2! + (g!(2, 1) * H1i! * d1!) + (g!(2, 2) * H2i! * d1!)) / d1!
LOCATE , 63: PRINT "H1r(0)"; : MOSTRAR (H1refi)
LOCATE , 63: PRINT "H2r(0)"; : MOSTRAR (H2refi)
CALL CAUDALES(H1f!, H2f!)
H1reff = (Vi1! + (g!(1, 1) * H1f! * d1!) + (g!(1, 2) * H2f! * d1!)) / d1!
H2reff = (Vi2! + (g!(2, 1) * H1f! * d1!) + (g!(2, 2) * H2f! * d1!)) / d1!
LOCATE , 63: PRINT "H1r(T)"; : MOSTRAR (H1reff)
LOCATE , 63: PRINT "H2r(T)"; : MOSTRAR (H2reff)

IF H1refi > 40 OR H1refi > 40 OR H1reff > 40 OR H2reff > 40 THEN
  VIEW PRINT 28 TO 30
  LOCATE , 5
  PRINT "          "; :
  PRINT "          "
  LOCATE , 5
  PRINT "          "; :
  PRINT "          "

  VIEW PRINT 28 TO 30
  LOCATE , 9
  PRINT " *** REFERENCIAS INICIALES O FINALES ESTAN FUERA DE RANGO ***"
  SOUND 4200, 8
END IF
CALL RESULTADOS

DO
  CHECK$ = INKEY$
  LOOP UNTIL CHECK$ <> ""
  IF FLAG6 = 1 THEN
    FLAG6 = 0
    GOTO SIMULA
  ELSE
    GOTO REGULADORLINEA1
  END IF
FIN:

```

---



---

**FIN DEL PROGRAMA PRINCIPAL**

---



---

SUBROUTINAS TANQUES ACOPLADOS

DEFDBL O-Z  
SUB AUTOR

```
CLS
REDIM MAT$(10)
REDIM LONGI(10)
SUPX = 6: SUPY = 12
MAT$(1) = " ESCUELA POLITECNICA NACIONAL "
MAT$(2) = " FACULTAD DE INGENIERIA ELECTRICA "
MAT$(3) = " DEPARTAMENTO DE ELECTRONICA Y CONTROL "
MAT$(4) = " "
MAT$(5) = " Realizado por: JUAN E. RAMOS P. "
MAT$(6) = " "
MAT$(7) = " Dirigido por: ING. MARCO BARRAGAN "
MAT$(8) = " "
MAT$(9) = " "
MAT$(10) = " E.P.N. 1996 "
```

```
LOCATE 1, 1, 0
LON% = LEN(MAT$(1))
COLOR MAGENTA,AZUL
LOCATE SUPX, SUPY
PRINT "┌ "; STRING$(LON% + 10, 205); "┐ "
LOCATE SUPX + 1, SUPY
PRINT "│ "; SPACES$(5); MAT$(1); SPACES$(5); "│ "
LOCATE SUPX + 2, SUPY
PRINT "└ "; STRING$(LON% + 10, 205); "┘ "
CURSOR% = 2
```

```
n% = 10
J = 3
FOR K = 2 TO n%
    LOCATE SUPX + J, SUPY
    PRINT "│ ";
    PIV% = 5 + LON% - LEN(MAT$(K))
    PRINT SPACES$(5); MAT$(K); SPACES$(PIV%);
    J = J + 1
    PRINT "│ "
```

```
NEXT K
LOCATE SUPX + 2 + n%, SUPY
PRINT "└ "; STRING$(LON% + 10, 205); "┘ "
COLOR BLANCO, NEGRO
FOR I = 1 TO 5
NEXT
END SUB
```

REM \$STATIC  
DEFDBL O-Z

SUB BORRAR

```
CLS
CALL MENUTANQUES
CALL VENTANAS
CALL VENTANAS1
```

```
CALL PARAINIT
CALL CONDINIT
CALL RESULTADOS
```

```
END SUB
```

#### SUB BORRARCONDI

```
VIEW PRINT 18 TO 26
FOR I = 1 TO 8
    LOCATE , 44
    PRINT "          "
NEXT I
CALL VENTCONDI
```

```
END SUB
```

#### SUB BORRARPARA

```
VIEW PRINT 18 TO 26
FOR I = 1 TO 8
    LOCATE , 63
    PRINT "          "
NEXT I
CALL VENTPARA
END SUB
```

```
DEFSNG O-Z
```

#### SUB CAUDALES (H1!, H2!)

```
Qi1! = Cd1! * Area1! * SQR(2 * 980 * H1!)
Qi2! = Cd2! * Area2! * SQR(2 * 980 * H2!)
Qi12! = Cd12! * Area12! * SQR(2 * 980 * ABS(H1! - H2!))
IF H1! > H2! THEN
    Qi1! = Qi1! + Qi12!
    Qi2! = Qi2! - Qi12!
ELSE
    Qi1! = Qi1! - Qi12!
    Qi2! = Qi2! + Qi12!
END IF
```

```
Vi1! = (Qi1! + 32.2) / b11!
Vi2! = (Qi2! + 51.64) / b22!
```

```
END SUB
```

#### SUB COEF (Ac!0, Bc!0)

```
VIEW PRINT 14 TO 16
FOR I = 1 TO 2
    LOCATE , 44
    PRINT "          "
NEXT I
```

```

VIEW PRINT 5 TO 14
LOCATE , 48: PRINT "Matrices para la Simulaci#n"
FOR I = 1 TO 2
  FOR J = 1 TO 2
    LOCATE , 44
    PRINT "Ac("; RIGHT$(STR$(I), 1); ", "; RIGHT$(STR$(J), 1); "):"; :
    MOSTRAR (Ac!(I, J))
  NEXT J
NEXT I
NEXT I
FOR I = 1 TO 2
  FOR J = 1 TO 2
    LOCATE , 44
    PRINT "Be("; RIGHT$(STR$(I), 1); ", "; RIGHT$(STR$(J), 1); "):"; :
    MOSTRAR (Be!(I, J))
  NEXT J
NEXT I
CALL RESULTADOS

END SUB

```

### SUB COEF1

```

VIEW PRINT 5 TO 8
LOCATE , 68: PRINT "      "
LOCATE , 62: PRINT "Qi1(max):"; : MOSTRAR (111.4)
LOCATE , 62: PRINT "Qi2(max):"; : MOSTRAR (147)

```

```

VIEW PRINT 9 TO 14
CALL CAUDALES(H1i!, H2i!)
LOCATE , 62: PRINT "Qi1(ini):"; : MOSTRAR (Qi1!)
LOCATE , 62: PRINT "Qi2(ini):"; : MOSTRAR (Qi2!)

```

```

VIEW PRINT 12 TO 14
CALL CAUDALES(H1f!, H2f!)
LOCATE , 62: PRINT "Qi1(fin):"; : MOSTRAR (Qi1!)
LOCATE , 62: PRINT "Qi2(fin):"; : MOSTRAR (Qi2!)

```

```

CALL RESULTADOS
CALL VENTANAS

```

```

END SUB

```

```

DEFDBL O-Z

```

### SUB COEFICIENTES (H1!, H2!, Ac!())

```

REDIM Ac!(2, 2)
IF H1! = 0 AND H2! = 0 THEN
  FOR I = 1 TO 2
    FOR J = 1 TO 2
      Ac!(I, J) = 0
    NEXT J
  NEXT I
  GOTO COEF2
END IF

```

```

Ac!(1, 1) = -Cd1! * Area1! * SQR(2 * 980) / (2 * A1! * SQR(H1!))
Ac!(1, 2) = 0: Ac!(2, 1) = 0
Ac!(2, 2) = -Cd2! * Area2! * SQR(2 * 980) / (2 * A2! * SQR(H2!))

```

```

IF H1! <> H2! THEN
  SEGUNDO! = Cd12! * Area12! * SQR(2 * 980) / (2 * A1! * SQR(ABS(H1! - H2!)))
  Ac!(1, 1) = Ac!(1, 1) - SEGUNDO!
  Ac!(2, 2) = Ac!(2, 2) - SEGUNDO!
  Ac!(1, 2) = SEGUNDO!
  Ac!(2, 1) = SEGUNDO!
END IF

```

```

COEF2:
END SUB

```

```

DEFSNG O-Z

```

### SUB CONDICIONEST

```

VIEW PRINT 18 TO 25
LOCATE , 44: PRINT "H1ini :"; : CALL VALIDAR(18, 43, H1i!)
REINGRESOT1:
IF H1i! < 1 OR H1i! > 40 THEN
  SOUND 4200, 8
  VIEW PRINT 18 TO 25
  LOCATE , 44: PRINT "H1ini :"; :
  CALL VALIDAR(18, 43, H1i!)
  GOTO REINGRESOT1
END IF
CALL VENTCONDI

```

```

LOCATE , 44: PRINT "H2ini :"; : CALL VALIDAR(19, 43, H2i!)
REINGRESOT2:
IF H2i! < 1 OR H2i! > 40 THEN
  SOUND 4200, 8
  VIEW PRINT 19 TO 25
  LOCATE , 44: PRINT "H2ini :"; :
  CALL VALIDAR(19, 43, H2i!)
  GOTO REINGRESOT2
END IF
CALL VENTCONDI

```

```

LOCATE , 44: PRINT "H1fin :"; : CALL VALIDAR(20, 43, H1f!)
REINGRESOT3:
IF H1f! < 1 OR H1f! > 40 THEN
  SOUND 4200, 8
  VIEW PRINT 20 TO 25
  LOCATE , 44: PRINT "H1fin :"; :
  CALL VALIDAR(20, 43, H1f!)
  GOTO REINGRESOT3
END IF
CALL VENTCONDI

```

```

LOCATE , 44: PRINT "H2fin :"; : CALL VALIDAR(21, 43, H2f!)
REINGRESOT4:
IF H2f! < 1 OR H2f! > 40 THEN
  SOUND 4200, 8
  VIEW PRINT 21 TO 25
  LOCATE , 44: PRINT "H2fin :"; :
  CALL VALIDAR(21, 43, H2f!)
  GOTO REINGRESOT4
END IF
CALL VENTCONDI

```

```

LOCATE , 44: PRINT "T(seg):"; : CALL VALIDAR(22, 43, TM!)
REINGRESOT5:
IF TM! <= 0 THEN
    SOUND 4200, 8
    VIEW PRINT 22 TO 25
    LOCATE , 44: PRINT "T(seg):"; :
    CALL VALIDAR(22, 43, TM!)
    GOTO REINGRESOT5
END IF
CALL VENTCONDI

```

```

LOCATE , 44: PRINT "iter :"; : CALL VALIDAR(23, 43, ITERT!)
REINGRESOT6:
IF ITERT! <= 0 OR (ITERT! - INT(ITERT!)) <> 0 THEN
    SOUND 4200, 8
    VIEW PRINT 23 TO 25
    LOCATE , 44: PRINT "iter :"; :
    CALL VALIDAR(23, 43, ITERT!)
    GOTO REINGRESOT6
END IF
CALL VENTCONDI

```

```

END SUB

```

#### SUB CONDINIT

```

ESP$ = "    "
VIEW PRINT 17 TO 27
LOCATE , 48: PRINT "CONDICIONES  PARAMETROS"
CALL VENTCONDI: CALL VENTPARA
LOCATE , 44: PRINT "H1ini "; SPACES$(9 - LEN(STR$(H1i!))); H1i!
LOCATE , 44: PRINT "H2ini "; SPACES$(9 - LEN(STR$(H2i!))); H2i!
LOCATE , 44: PRINT "H1fin "; SPACES$(9 - LEN(STR$(H1f!))); H1f!
LOCATE , 44: PRINT "H2fin "; SPACES$(9 - LEN(STR$(H2f!))); H2f!
LOCATE , 44: PRINT "T(seg):"; SPACES$(9 - LEN(STR$(TM!))); TM!
LOCATE , 44: PRINT "iter :"; SPACES$(9 - LEN(STR$(ITERT!))); ITERT!
CALL VENTCONDI
END SUB

```

```

DEFDBL O-Z

```

#### SUB CUADRATICO

```

VIEW PRINT 28 TO 30
LOCATE , 5
PRINT "          ";
PRINT "          "
LOCATE , 5
PRINT "          ";
PRINT "          "

```

```

VIEW PRINT 28 TO 30
LOCATE , 12
PRINT "***** CALCULANDO REALIMENTACION OPTIMA DE ESTADO *****"
SOUND 4200, 8

```

```

END SUB

```

```

DEFINT I-L, N
DEFDBL A-H, M

```

'DISCRETIZACION DE MATRICES

**SUB DISCRETIZAR (Ac!0, Bc!0, Ad!0, Bd!0)**

REDIM IDENT!(2, 2), Adi!(2, 2), Ac2!(2, 2)

REDIM Ad!(2, 2), Bd!(2, 2), AcI!(2, 2)

B! = -(Ac!(1, 1) + Ac!(2, 2))

C! = (Ac!(1, 1) \* Ac!(2, 2) - Ac!(2, 1) \* Ac!(1, 2))

IF Ac!(1, 2) = 0 AND Ac!(2, 1) = 0 THEN

AL1! = -B!

AL2! = -B!

C1! = TM! \* EXP(AL1! \* TM!)

C0! = EXP(AL1! \* TM!) - TM! \* AL1! \* EXP(AL1! \* TM!)

ELSE

AL1! = (-B! + SQR(B! ^ 2 - 4 \* C!)) / 2

AL2! = (-B! - SQR(B! ^ 2 - 4 \* C!)) / 2

C1! = (EXP(AL1! \* TM!) - EXP(AL2! \* TM!)) / (AL1! - AL2!)

C0! = (-AL2! \* EXP(AL1! \* TM!) + AL1! \* EXP(AL2! \* TM!)) / (AL1! - AL2!)

END IF

Ad!(1, 1) = C0! + C1! \* Ac!(1, 1)

Ad!(1, 2) = C1! \* Ac!(1, 2)

Ad!(2, 1) = C1! \* Ac!(2, 1)

Ad!(2, 2) = C0! + C1! \* Ac!(2, 2)

FOR I = 1 TO 2

FOR J = 1 TO 2

IF I = J THEN

IDENT!(I, J) = -1

ELSE

IDENT!(I, J) = 0

ENDIF

NEXT J

NEXT I

CALL SUMA(2, 2, IDENT!(), Ad!(), Adi!())

CALL INVERSA(Ac!(), AcI!(), DET!)

CALL MULT(2, 2, 2, Adi!(), AcI!(), Ac2!())

CALL MULT(2, 2, 2, Ac2!(), Bc!(), Bd!())

END SUB

DEFSNG A-N

**SUB ERROR0**

VIEW PRINT 28 TO 30

LOCATE , 5

PRINT " " " ; :

PRINT " " "

LOCATE , 5

PRINT " " " ; :

PRINT " " "

VIEW PRINT 28 TO 30

LOCATE , 18

PRINT "HAY QUE PRIMERO UTILIZAR LA OPCION \*CALCULAR\*"

SOUND 4200, 8

```
DO
LOOP UNTIL INKEY$ <> ""
```

```
END SUB
```

### SUB ERRORM1

```
VIEW PRINT 28 TO 30
LOCATE , 5
PRINT "                " ; :
PRINT "                "
LOCATE , 5
PRINT "                " ; :
PRINT "                "
```

```
VIEW PRINT 28 TO 30
LOCATE , 15
PRINT " INGRESE CONDICIONES Y EL R.Q.L. PARA LA SIMULACIOM"
SOUND 4200, 8
CALL VENTANAS
DO
    CHECK$ = INKEY$
LOOP UNTIL CHECK$ <> ""
```

```
END SUB
```

### SUB ERRORM2

```
VIEW PRINT 28 TO 30
LOCATE , 5
PRINT "                " ; :
PRINT "                "
LOCATE , 5
PRINT "                " ; :
PRINT "                "
```

```
VIEW PRINT 28 TO 30
LOCATE , 20
PRINT "PERIODO DE MUESTREO DEBE SER DIFERENTE DE CERO"
SOUND 4200, 8
DO
LOOP UNTIL INKEY$ <> ""
```

```
END SUB
```

### SUB ERRORM3

```
VIEW PRINT 28 TO 30
LOCATE , 5
PRINT "                " ; :
PRINT "                "
LOCATE , 5
PRINT "                " ; :
PRINT "                "
```

```
VIEW PRINT 28 TO 30
LOCATE , 20
PRINT "ITERACCIONES DEBEN SER DIFERENETES DE CERO"
```

```
SOUND 4200, 8
DO
LOOP UNTIL INKEY$ <> ""

END SUB
```

### SUB ERROR1

```
VIEW PRINT 28 TO 30
LOCATE , 5
PRINT "           ";
PRINT "           "
LOCATE , 5
PRINT "           ";
PRINT "           "
```

```
VIEW PRINT 28 TO 30
LOCATE , 20
PRINT "MATRIZ NO SINGULAR HAY QUE REINGRESAR R+0"
SOUND 4200, 8
CALL VENTANAS

END SUB
```

### SUB ESCALA (LINEA%, TSIM!, GRAFICO%, DATOS!(), MAXMIN!())

```
YMAX! = MAXMIN!(GRAFICO%, 1)
YMIN! = MAXMIN!(GRAFICO%, 2)
```

```
IF YMAX! > 0 AND YMIN! >= 0 THEN
    DELTAY1 = YMIN!
    DELTAY2 = YMAX!
    INCY = (YMAX! - YMIN!) / 10
```

```
ELSEIF YMAX! <= 0 AND YMIN! < 0 THEN
    DELTAY1 = YMAX!
    DELTAY2 = YMIN!
    INCY = (YMIN! - YMAX!) / 10
```

```
ELSE
    DELTAY1 = 1.05 * YMIN!
    DELTAY2 = 1.05 * YMAX!
    EJEY1 = 1.05 * YMIN!
    EJEY2 = 1.05 * YMAX!
    INCY = (YMAX! - YMIN!) / 10
```

```
END IF
```

```
REM *****
REM IMPRIME LOS VALORES CARACTERISTICOS DE LA CURVA DESEADA
REM *****
```

```
IF LINEA% = 1 THEN
    VIEW PRINT 9 TO 15
    LOCATE , 44
    PRINT "Scal:";
    MOSTRAR (INCY)
    LOCATE , 44
    PRINT "Max:";
    MOSTRAR (YMAX!)
    LOCATE , 44
```

```

PRINT "Min :";:
MOSTRAR (YMIN!)
LOCATE , 44
PRINT "Ini :";:
MOSTRAR (DATOS!(GRAFICO%, 1))
LOCATE , 44
PRINT "Fin :";:
MOSTRAR (DATOS!(GRAFICO%, TSIM!))
ELSE
VIEW PRINT 9 TO 15
LOCATE , 64
PRINT "Scal:";:
MOSTRAR (INCY)
LOCATE , 64
PRINT "Max :";:
MOSTRAR (YMAX!)
LOCATE , 64
PRINT "Min :";:
MOSTRAR (YMIN!)
LOCATE , 64
PRINT "Ini :";:
MOSTRAR (DATOS!(GRAFICO%, 1))
LOCATE , 64
PRINT "Fin :";:
MOSTRAR (DATOS!(GRAFICO%, TSIM!))
END IF
END SUB

DEFINT I-N
DEFSNG O-Z
DEFDBL A, H

SUB ESCALAR (n, M, ESCAL!, A!(), B!())
'
'MULTIPLICACION DE UN ESCALAR POR UNA MATRIZ
'

FOR I = 1 TO n
FOR J = 1 TO M
B!(I, J) = ESCAL! * A!(I, J)
NEXT J
NEXT I

END SUB

DEFDBL B-G, O-T

FUNCTION FACTORIAL! (M%) STATIC

IF M% > 0 THEN
FACTORIAL! = FACTORIAL!(M% - 1) * M%
ELSE
FACTORIAL! = 1!
END IF
END FUNCTION

DEFSNG A-N
DEFDBL U-Z

```

SUB FASEMODULO (n, M, XZ!, XP!, REAL!, IMAG!, FASE!, MODULO!)

REDIM XZ1(n), XP1(M)

\*\*\*\*\*  
'CALCULO DE LA FASE Y MODULO DEL NUMERADOR  
\*\*\*\*\*

u = 2 \* REAL!  
v = -REAL! ^ 2 - IMAG! ^ 2  
XZ1(0) = XZ!(n)  
XZ1(1) = XZ!(n - 1) + u \* XZ1(0)  
FOR K = 2 TO n  
    XZ1(K) = XZ!(n - K) + u \* XZ1(K - 1) + v \* XZ1(K - 2)  
NEXT K  
REALZ = XZ1(n) - (XZ1(n - 1) \* REAL!)  
IMAGZ = XZ1(n - 1) \* IMAG!

\*\*\*\*\*  
'CALCULO DE LA FASE Y MODULO DEL DENOMINADOR  
\*\*\*\*\*

XP1(0) = XP!(M)  
XP1(1) = XP!(M - 1) + u \* XP1(0)  
FOR K = 2 TO M  
    XP1(K) = XP!(M - K) + u \* XP1(K - 1) + v \* XP1(K - 2)  
NEXT K  
REALP = XP1(M) - XP1(M - 1) \* REAL!  
IMAGP = XP1(M - 1) \* IMAG!

\*\*\*\*\*  
'DIVIDO NUMERADOR / DENOMINADOR  
\*\*\*\*\*

DENOMINADOR = REALP ^ 2 + IMAGP ^ 2  
REALL! = (REALZ \* REALP + IMAGZ \* IMAGP) / DENOMINADOR  
IMAGG! = (IMAGZ \* REALP - IMAGP \* REALZ) / DENOMINADOR  
MODULO = SQR(REALL! ^ 2 + IMAGG! ^ 2)  
IF REALL! = 0 THEN  
    ANGULO! = 90  
ELSE  
    ANGULO! = 180 \* ATN(IMAGG! / REALL!) / 3.141592654#  
END IF

\*\*\*\*\*  
'REPRESENTO EN COORDENADAS POLARES  
\*\*\*\*\*

IF IMAGG! >= 0 AND REALL! > 0 THEN  
    ANGULO! = ANGULO!  
ELSEIF IMAGG! < 0 AND REALL! < 0 THEN  
    ANGULO! = ANGULO! + 180  
ELSEIF IMAGG! > 0 AND REALL! < 0 THEN  
    ANGULO! = ANGULO! + 180  
ELSE  
    ANGULO! = ANGULO! + 360  
END IF

IF ANGULO >= 0 THEN  
    ANGULO! = 180 - ANGULO!  
ELSE  
    ANGULO! = -180 - ANGULO!  
END IF

```
FASE! = ANGULO
END SUB
```

```
SUB GRAFICOS (LINEA%, TSIM!, GRAFICO%, DATOS!(), MAXMIN!())
```

```
INCX = TSIM! / 10
DELTAX1 = -.02 * TSIM!
DELTAX2 = 1.02 * TSIM!
EJEX1 = -.05 * TSIM!
EJEX2 = 1.05 * TSIM!
```

```
YMAX! = MAXMIN!(GRAFICO%, 1)
YMIN! = MAXMIN!(GRAFICO%, 2)
```

```
IF YMAX! > 0 AND YMIN! >= 0 THEN
    DELTAY1 = YMIN!
    DELTAY2 = YMAX!
    INCY = (YMAX! - YMIN!) / 10
```

```
    ELSEIF YMAX! <= 0 AND YMIN! < 0 THEN
        DELTAY1 = YMAX!
        DELTAY2 = YMIN!
        INCY = (YMIN! - YMAX!) / 10
```

```
ELSE
    DELTAY1 = 1.05 * YMIN!
    DELTAY2 = 1.05 * YMAX!
    EJEY1 = 1.05 * YMIN!
    EJEY2 = 1.05 * YMAX!
    INCY = (YMAX! - YMIN!) / 10
```

```
END IF
```

```
BARX = .05 * (DELTAY2 - DELTAY1)
BARY = .02 * (EJEX2 - EJEX1)
WINDOW (DELTAX1, DELTAY1)-(DELTAX2, DELTAY2)
```

```
*****
```

```
'GRAFICO LOS EJES PARA LOS GRAFICOS
```

```
*****
```

```
IF YMAX! < 0 AND YMIN! >= 0 OR YMAX! > 0 AND YMIN! <= 0 THEN
    LINE (EJEX1, 0)-(EJEX2, 0)
```

```
ELSE
```

```
    EJE = DELTAY1
    LINE (EJEX1, EJE)-(EJEX2, EJE), , &HF0F0
```

```
END IF
```

```
LINE (0, DELTAY1)-(0, DELTAY2)
```

```
*****
```

```
'GRAFICO MARCAS PARA EL EJE X
```

```
*****
```

```
IF YMAX! < 0 AND YMIN! >= 0 OR YMAX! > 0 AND YMIN! <= 0 THEN
```

```
    FOR I = 0 TO 10
```

```
        LINE (INCX * I, -BARX / 2)-(INCX * I, BARX / 2)
```

```
    NEXT
```

```
ELSE
```

```
    EJEY = DELTAY1
```

```
    FOR I = 0 TO 10
```

```
        LINE (INCX * I, EJEY + BARX / 2)-(INCX * I, EJEY - BARX / 2)
```

```
    NEXT
```

```
END IF
```

```

*****
'GRAFICO MARCAS PARA EL EJE Y
*****
IF YMIN! >= 0 AND YMAX! > 0 THEN
  FOR I = 0 TO 10
    LINE (-BARY / 2, YMIN! + INCY * I)-(BARY / 2, YMIN! + INCY * I)
  NEXT I
ELSEIF YMAX! <= 0 AND YMIN! < 0 THEN
  FOR I = 0 TO 10
    LINE (-BARY / 2, YMAX! + INCY * I)-(BARY / 2, YMAX! + INCY * I)
  NEXT I
ELSE
  CONT = 0
  FOR I = 0 TO EJEY2 STEP INCY
    LINE (-BARY / 2, CONT * INCY)-(BARY / 2, CONT * INCY)
    CONT = CONT + 1
  NEXT I

  CONT = 0
  FOR I = 0 TO EJEY1 STEP -INCY
    LINE (-BARY / 2, -CONT * INCY)-(BARY / 2, -CONT * INCY)
    CONT = CONT + 1
  NEXT I
END IF

```

```

*****
***** GRAFICO LA CURVA DESEADA *****
*****
PSET (0, DATOS!(GRAFICO%, 1))
FOR I = 1 TO TSIM!
  LINE -(I - 1, DATOS!(GRAFICO%, I))
NEXT
END SUB

```

```

SUB GRAFTANQUES (ROW%, COLUMN%, SALIDA%)

```

```

DIM REDUC$(8)
REDUC$(1) = " H1(t)"
REDUC$(2) = " H2(t)"
REDUC$(3) = " e1(t)"
REDUC$(4) = " e2(t)"
REDUC$(5) = " Qi1(t)"
REDUC$(6) = " Qi2(t)"
REDUC$(7) = " V1(t)"
REDUC$(8) = " V2(t)"

```

```

CONTADOR = 1
MARCA% = COLUMN% + 9
LOCATE ROW%, MARCA%
PRINT "Ø"
PASO = COLUMN% + 12
LOCATE ROW%, PASO

```

```

DO
  LOCATE ROW%, PASO
  $$ = INKEY$
  SELECT CASE $$
    CASE CHR$(0) + CHR$(ARRIBA)
      FLAG1 = 1
      IF CONTADOR = 8 THEN

```

```

        CONTADOR = 1
    ELSE
        CONTADOR = CONTADOR + 1
    END IF
    PRINT REDUC$(CONTADOR)

    CASE CHR$(0) + CHR$(ABAJ0)
    FLAG1 = 1
    IF CONTADOR = 1 THEN
        CONTADOR = 8
    ELSE
        CONTADOR = CONTADOR - 1
    END IF
    PRINT REDUC$(CONTADOR)

    CASE CHR$(ENTER)
    IF FLAG1 = 1 THEN
        SALIDA% = CONTADOR
    END IF
    LOCATE ROW%, MARCA%
    PRINT " "
    EXIT DO

    CASE ELSE
    END SELECT
LOOP UNTIL S$ = CHR$(13)

END SUB

SUB INVERSA (Ma!0, Inv!0, DET!)

DET! = Ma!(1, 1) * Ma!(2, 2) - Ma!(1, 2) * Ma!(2, 1)
IF DET! < 0 THEN
    Inv!(1, 1) = Ma!(2, 2) / DET!
    Inv!(2, 2) = Ma!(1, 1) / DET!
    Inv!(1, 2) = -Ma!(1, 2) / DET!
    Inv!(2, 1) = -Ma!(2, 1) / DET!
END IF
END SUB

SUB MAX (n%, !!, DATOS!0, MAXMIN!0)

FOR I = 1 TO n%
    YMAX! = DATOS!(I, 1)
    YMIN! = DATOS!(I, 2)

    FOR J = 1 TO !!
        IF YMAX! < DATOS!(I, J) THEN
            YMAX! = DATOS!(I, J)
        END IF
        IF YMIN! > DATOS!(I, J) THEN
            YMIN! = DATOS!(I, J)
        END IF
    NEXT J
    MAXMIN!(I, 1) = YMAX!
    MAXMIN!(I, 2) = YMIN!
NEXT I

```

END SUB

### SUB MENUS

VIEW PRINT 28 TO 30

LOCATE , 5

PRINT "                                   ";:

PRINT "                                   "

LOCATE , 5

PRINT "                                   ";:

PRINT "                                   "

VIEW PRINT 28 TO 30

LOCATE , 28

PRINT " INGRESANDO COMPENSACION"

SOUND 4200, 8

END SUB

### SUB MENUTANQUES

VIEW PRINT 28 TO 30

LOCATE , 5

PRINT "F1 = PARAMETROS      F3 = R.Q.L              F5 = SIM. DINAMICA    F7=SALIR "

LOCATE , 5

PRINT "F2 = CONDICIONES    F4 = CALCULAR      F6 = GRAFICOS                                   "

END SUB

DEFSNG O-Z

### SUB MOSTRAR (NUM!)

DATOS\$ = STR\$(NUM!)

IF ABS(NUM!) < 1 THEN

    IF LEN(DATOS\$) <= 9 THEN

        PRINT SPACE\$(9 - LEN(STR\$(NUM!))); NUM!

    ELSE

        IF ABS(NUM!) < 1E-09 THEN

            PRINT USING "#.###^####"; NUM!

        ELSE

            EXPONENTE = VAL(RIGHT\$(DATOS\$, 2))

            SIGNOS\$ = LEFT\$(DATOS\$, 1)

            CEROS\$ = STRING\$(EXPONENTE - 1, "0")

            VALOR = VAL(LEFT\$(DATOS\$, (9 - EXPONENTE)))

            LARGO = 8 - (EXPONENTE)

            NUMEROS\$ = ""

            FOR I = 1 TO 9

                IF I < 3 THEN

                    NUMERO1\$ = MID\$(STR\$(ABS(VALOR)), I, 1)

                    IF NUMERO1\$ = "" THEN

                        NUMERO1\$ = "0"

                    END IF

                    NUMEROS\$ = NUMEROS\$ + NUMERO1\$

                END IF

            NEXT I

            PRINT SIGNOS\$; ". "; CEROS\$; MID\$(NUMEROS\$, 2, LARGO)

        END IF

```

    END IF
ELSE
    PRINT SPACES$(9 - LEN(STR$(NUM!))); NUM!
END IF
END SUB

```

```

DEFINT I-N
DEFDBL A-H, O-Z
'
'MULTIPLICACION DE MATRICES
'

```

```

SUB MULT (n, M, I, A!(), B!(), C!())

```

```

FOR I = 1 TO n
    FOR J = 1 TO I
        R! = 0
        FOR K = 1 TO M
            R! = R! + A!(I, K) * B!(K, J)
        NEXT
        C!(I, J) = R!
    NEXT J
NEXT I
END SUB

```

```

DEFSNG A-N

```

```

FUNCTION NUEVOVALOR$

```

```

VIEW PRINT 28 TO 30
LOCATE , 5
PRINT "                "; :
PRINT "                "
LOCATE , 5
PRINT "                "; :
PRINT "                "

```

```

VIEW PRINT 28 TO 30
LOCATE , 20
PRINT "  DESEA REINGRESAR LOS VALORES (S/N)"
SOUND 4200, 8
DO
    CHECK$ = INKEY$
    NUEVOVALOR$ = CHECK$
LOOP UNTIL CHECK$ = "S" OR CHECK$ = "s" OR CHECK$ = "N" OR CHECK$ = "n"

```

```

END FUNCTION

```

```

DEFSNG O-Z

```

```

SUB PARAINIT

```

```

DIM CONT$(2)
CONT$(1) = "OFF"
CONT$(2) = " ON"

```

```

IF Area1! = 0 THEN
    AArea1! = 1
ELSE

```

```

    AArea1! = 2
END IF
IF Area2! = 0 THEN
    AArea2! = 1
ELSE
    AArea2! = 2
END IF

ESP$ = "    "
VIEW PRINT 17 TO 25
LOCATE , 48: PRINT "CONDICIONES    PARAMETROS "
CALL VENTPARA: CALL VENTCONDI
LOCATE , 63: PRINT "A1  :"; SPACES$(9 - LEN(STR$(A1!))); A1!
LOCATE , 63: PRINT "A2  :"; SPACES$(9 - LEN(STR$(A2!))); A2!
LOCATE , 63: PRINT "a1  :"; LEFT$(ESP$, 6); CONT$(AArea1!)
LOCATE , 63: PRINT "a2  :"; LEFT$(ESP$, 6); CONT$(AArea2!)
LOCATE , 63: PRINT "a12 :"; SPACES$(9 - LEN(STR$(AArea12!))); AArea12!
CALL VENTPARA

END SUB

```

### SUB PARAMETROST

```

VIEW PRINT 20 TO 25

LOCATE , 63: PRINT "a1  :"; CALL SALIDAS(20, 62, Area1!)
CALL VENTPARA
LOCATE , 63: PRINT "a2  :"; CALL SALIDAS(21, 62, Area2!)
CALL VENTPARA
LOCATE , 63: PRINT "a12 :"; CALL SALIDAS1(22, 61, Area12!)
CALL VENTPARA
END SUB

```

### DEFDBL O-Z

#### SUB PROCESANDO

```

VIEW PRINT 28 TO 30

```

```

LOCATE , 5
PRINT "                "; :
PRINT "                "
LOCATE , 5
PRINT "                "; :
PRINT "                "

```

```

VIEW PRINT 28 TO 30

```

```

LOCATE , 16
PRINT "***** PROCESANDO INFORMACION *****"
SOUND 4200, 8

```

```

END SUB

```

### SUB RELACION (ROW%, COLUM%, SALIDA!)

```

DIM REDUC$(5), REDUCC!(5)
REDUC$(1) = " 1:1"
REDUC$(2) = " ^3:1"
REDUC$(3) = " 3:1"
REDUC$(4) = "3^3:1"
REDUC$(5) = " 9:1"

```

```

REDUCC!(1) = 1 / 1
REDUCC!(2) = 1 / SQR(3)
REDUCC!(3) = 1 / 3
REDUCC!(4) = 1 / (3 * SQR(3))
REDUCC!(5) = 1 / 9

CONTADOR = 1
MARCA% = COLUM% + 8
LOCATE ROW%, MARCA%
PRINT "*"
PASO = COLUM% + 12
LOCATE ROW%, PASO

DO
  LOCATE ROW%, PASO
  S$ = INKEY$
  SELECT CASE S$

    CASE CHR$(0) + CHR$(ARRIBA)
      FLAG1 = 1
      IF CONTADOR = 5 THEN
        CONTADOR = 1
      ELSE
        CONTADOR = CONTADOR + 1
      END IF
      PRINT REDUC$(CONTADOR)

    CASE CHR$(0) + CHR$(ABAJ0)
      FLAG1 = 1
      IF CONTADOR = 1 THEN
        CONTADOR = 5
      ELSE
        CONTADOR = CONTADOR - 1
      END IF
      PRINT REDUC$(CONTADOR)

    CASE CHR$(ENTER)
      IF FLAG1 = 1 THEN
        SALIDA! = REDUCC!(CONTADOR)
      END IF
      LOCATE ROW%, MARCA%
      PRINT " "
      EXIT DO

    CASE ELSE
      END SELECT
  LOOP UNTIL S$ = CHR$(13)
END SUB

DEFINT I-N
DEFDBL A-H

```

```

SUB RESTA (n, M, A!(), B!(), C!())

```

```

FOR I = 1 TO n
  FOR J = 1 TO M
    C!(I, J) = A!(I, J) - B!(I, J)
  NEXT J
NEXT I
END SUB

```

DEFSNG A-N

SUB RESULTADOS

LINE (337, 59)-(635, 240), , B

LINE (484, 59)-(484, 240)

END SUB

DEFSNG O-Z

SUB SALIDAS (ROW%, COLUM%, SALIDA)

MARCA% = COLUM% + 8  
 LOCATE ROW%, MARCA%  
 PRINT "Ø"  
 PASO = COLUM% + 14  
 LOCATE ROW%, PASO

DO

LOCATE ROW%, PASO  
 S\$ = INKEY\$  
 SELECT CASE S\$  
   CASE CHR\$(0) + CHR\$(ARRIBA)  
     FLAG1 = 1  
     CONTADOR = 1  
     PRINT "OFF"

  CASE CHR\$(0) + CHR\$(ABAJO)  
     FLAG1 = 1  
     CONTADOR = 2  
     PRINT "ON"

  CASE CHR\$(ENTER)  
 IF FLAG1 = 1 THEN  
   IF CONTADOR = 1 THEN  
     SALIDA = 0  
   ELSE  
     SALIDA = 1.26  
   END IF  
 END IF

LOCATE ROW%, MARCA%  
 PRINT " "  
 EXIT DO

  CASE ELSE  
 END SELECT  
 LOOP UNTIL S\$ = CHR\$(13)

END SUB

DEFINT C

SUB SALIDAS1 (ROW%, COLUM%, SALIDA)

DIM CONT\$(5), CONTT!(5), Cd!(5)  
 CONT\$(0) = "0": CONTT!(0) = .173: Cd!(0) = .398

```

CONT$(1) = "1": CONTT!(1) = .738: Cd!(1) = .341
CONT$(2) = "2": CONTT!(2) = 1.17: Cd!(2) = .333
CONT$(3) = "3": CONTT!(3) = 1.78: Cd!(3) = .298
CONT$(4) = "4": CONTT!(4) = 2.93: Cd!(4) = .299
CONT$(5) = "5": CONTT!(5) = 4.64: Cd!(5) = .317

```

```

MARCA% = COLUM% + 9
LOCATE ROW%, MARCA%
PRINT "Ø"
PASO = COLUM% + 17
LOCATE ROW%, PASO
CONTADOR = 0

```

```
DO
```

```

LOCATE ROW%, PASO
S$ = INKEY$

SELECT CASE S$
CASE CHR$(0) + CHR$(ARRIBA)
FLAG1 = 1
IF CONTADOR = 5 THEN
CONTADOR = 0
ELSE
CONTADOR = CONTADOR + 1
END IF
PRINT CONT$(CONTADOR)

CASE CHR$(0) + CHR$(ABAJA)
FLAG1 = 1
IF CONTADOR = 0 THEN
CONTADOR = 5
ELSE
CONTADOR = CONTADOR - 1
END IF
PRINT CONT$(CONTADOR)

CASE CHR$(ENTER)
IF FLAG1 = 1 THEN
SALIDA = CONTADOR
Area12! = CONTT!(CONTADOR)
Cd12! = Cd!(CONTADOR)
END IF
LOCATE ROW%, MARCA%
PRINT " "
EXIT DO
CASE ELSE
END SELECT
LOOP UNTIL S$ = CHR$(13)

```

```
END SUB
```

```
DEFSNG C
```

```
FUNCTION SIMULACION
```

```

VIEW PRINT 28 TO 30
LOCATE , 5
PRINT " " ;
PRINT " " ;
LOCATE , 5

```

```

PRINT "                " ; :
PRINT "                "

VIEW PRINT 28 TO 30
LOCATE , 20
PRINT " DESEA REALIZAR SIMULACION REAL (S/N)"
SOUND 4200, 8
DO
    CHECK$ = INKEY$
LOOP UNTIL CHECK$ = "S" OR CHECK$ = "s" OR CHECK$ = "N" OR CHECK$ = "n"

IF CHECK$ = "S" OR CHECK$ = "s" THEN
    SIMULACION = 1
ELSE
    SIMULACION = 0
END IF

END FUNCTION

DEFINT I-N
DEFDBL A-H, O-Z
'

SUB SUMA (n, M, A!(), B!(), C!())

FOR I = 1 TO n
    FOR J = 1 TO M
        C!(I, J) = A!(I, J) + B!(I, J)
    NEXT J
NEXT I
END SUB

DEFSNG A-N
SUB TANQUES1
COLOR FONDO
VIEW PRINT 2 TO 4
LOCATE , 12: PRINT "H1          " ; :
PRINT "                H2"
LOCATE , 11: PRINT STR$(MAXIM); " _          " ; :
PRINT "                _"; STR$(MAXIM)
VIEW PRINT 23 TO 24
LOCATE , 11: PRINT STR$(MINIM); " _          " ; :
PRINT "                _"; STR$(MINIM)

LINE (120, 30)-(120, 365) '1
LINE (120, 45)-(520, 45), , &HF0F0
LINE (120, 365)-(520, 365), , &HF0F0
LINE (520, 30)-(520, 365) '7
LINE (319, 30)-(319, 365) '8
LINE (321, 365)-(321, 30) '10

END SUB

SUB TRANS (n, M, A!(), AT!())

FOR I = 1 TO M
    FOR J = 1 TO n
        AT!(I, J) = A!(J, I)
    NEXT J

```

NEXT I

END SUB

DEFSNG O-Z

SUB VALIDAR (ROW%, COLUM%, NUM!)

'\*\*\*\*\*  
' \* SUBROUTINA QUE CHEQUEA LOS DATOS INGRESADOS \*  
'\*\*\*\*\*

AUXILIAR! = NUM!

MARCA% = COLUM% + 8

LOCATE ROW%, MARCA%

PRINT "Ø"

PASO1 = COLUM% + 9

PASO = COLUM% + 17

DO

LOCATE ROW%, PASO

DO

CHECK\$ = INKEY\$

LOOP UNTIL CHECK\$ <> ""

NUMERO\$ = CHECK\$

SELECT CASE NUMERO\$

CASE CHR\$(CERO) TO CHR\$(NUEVE)

IF FLAG4 = 0 THEN

IF FLAG3 = 0 THEN

LOCATE ROW%, PASO1

PRINT " "

LOCATE ROW%, PASO

FLAG3 = 1

END IF

IF CONTADOR = 0 AND NUMERO\$ = "0" THEN

FLAG4 = 1

END IF

IF CONTADOR < 8 THEN

FLAG2 = 1

CIFRA = CIFRA + 1

CONTADOR = CONTADOR + 1

DATA\$ = DATA\$ + CHECK\$

LOCATE ROW%, PASO - CONTADOR

PRINT DATA\$

ELSE

BEEP

END IF

ELSE

BEEP

END IF

CASE CHR\$(MENOS)

IF FLAG3 = 0 THEN

LOCATE ROW%, PASO1

PRINT " "

LOCATE ROW%, PASO

FLAG3 = 1

END IF

IF FLAG2 = 0 AND FLAG1 = 0 THEN

CONTADOR = CONTADOR + 1

FLAG2 = 1

CIFRA = CIFRA + 1

```

    DATA$ = DATA$ + CHECKS$
    LOCATE ROW%, PASO - CONTADOR
    PRINT DATA$
ELSE
    BEEP
END IF

    CASE CHR$(PUNTO)
    FLAG4 = 0
IF FLAG3 = 0 THEN
    LOCATE ROW%, PASO1
    PRINT "   "
    LOCATE ROW%, PASO
    FLAG3 = 1
ELSE
END IF
IF FLAG1 = 0 THEN
    CONTADOR = CONTADOR + 1
    FLAG1 = 1
    CIFRA = CIFRA + 1
    SEGURO = CIFRA
    DATA$ = DATA$ + CHECKS$
    LOCATE ROW%, PASO - CONTADOR
    PRINT DATA$
ELSE
    BEEP
END IF

    CASE CHR$(DELETE)
IF CIFRA > 0 THEN
    IF CIFRA = SEGURO THEN
        FLAG1 = 0
    END IF
    CIFRA = CIFRA - 1
    CONTADOR = CONTADOR - 1
    DATA$ = LEFT$(DATA$, CIFRA)
    IF CIFRA = 0 THEN
        DATA1$ = "0"
        LOCATE ROW%, PASO1
        PRINT "   "
        LOCATE ROW%, PASO - CONTADOR - 1
        PRINT DATA1$
    ELSE
        LOCATE ROW%, PASO1
        PRINT "   "
        LOCATE ROW%, PASO - CONTADOR
        PRINT DATA$
    END IF
    IF CONTADOR = 0 THEN
        FLA1 = 0: FLAG2 = 0: FLAG3 = 0: FLAG4 = 0
    END IF
ELSE
    BEEP
END IF

    CASE CHR$(ENTER)
IF CONTADOR = 0 THEN
    LOCATE ROW%, MARCA%
    PRINT " "
    NUM! = AUXILIAR!
    EXIT DO
ELSE

```

```

        NUM! = VAL(DATAS$)
        LOCATE ROW%, MARCA%
        PRINT " "
        EXIT DO
    END IF

    CASE CHR$(0) + CHR$(ABAJ0)
    IF CONTADOR = 0 THEN
        LOCATE ROW%, MARCA%
        PRINT " "
        NUM! = AUXILIAR!
        EXIT DO
    ELSE
        NUM! = VAL(DATAS$)
        LOCATE ROW%, MARCA%
        PRINT " "
        EXIT DO
    END IF

    CASE ELSE
    BEEP
    END SELECT
LOOP
END SUB

```

#### SUB VENTANAS

```

SCREEN 12
COLOR FONDO
LINE (1, 1)-(639, 479), , B 'BORDE EXTERIOR
LINE (1, 415)-(639, 415)
END SUB

```

#### SUB VENTANAS1

```

SCREEN 12
*****
* RESTITUYE LAS VENTANAS *
*****
LINE (5, 4)-(325, 205), , B
LINE (5, 210)-(325, 411), , B
END SUB

```

#### SUB VENTCONDI

```

LINE (337, 250)-(482, 410), , B
LINE (402, 270)-(477, 405), , B
END SUB

```

#### SUB VENTPARA

```

LINE (490, 250)-(635, 410), , B 'VENTANA DE PARAMETROS
LINE (555, 270)-(630, 405), , B
END SUB

```

**FIN SUBROUTINAS TANQUES ACOPLADOS**

**B I B L I O G R A F I A .**

1.- Garzón García, Juan Fernando

**Diseño y Construcción de un control Analógico y Digital de un sistema de Tanques Acoplados.**

TESIS DE GRADO. EPN 1990.

2.- Wellstead, P.E.

**CE5 COUPLED TANK APPARATUS.**

CONTROL SYSTEMS CENTRE, UMIST, 1982.

3.- KUO BENJAMÍN.

**Digital Control Systems.**

INTERNATIONAL EDITIONS, 1981

4.- KATSUHICO OGATA

**Ingeniería de Control Moderno.**

PRENTICE/HALL INTERNATCIONAL, 1978

5.- YASUNDO TAKAHASHI

**Control and Dinamic Systems.**

Copyright by Addison-Wesley Publishing Company Inc.

6.- Electro-Craft Corporation 1968.

**MOTOMATIC Control System Laboratory**

7.- Microsoft QuickBASIC 4.5.

**Programing in BASIC**

Copyright 1988,1990 Microsoft Corporation