



D-7588

T
004.4340285
5232



ESCUELA SUPERIOR POLITECNICA DEL LITORAL
FACULTAD DE INGENIERIA ELECTRICA

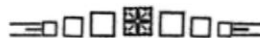
**"Simulación de Sistemas Continuos en
Microcomputadores IBM - PC"**

TESIS DE GRADO
Previa a la Obtención del Título de
INGENIERA en ELECTRICIDAD

Especialización: ELECTRONICA

Presentada por:

RUTH E. SANTANA VILLAMAR



Guayaquil - Ecuador

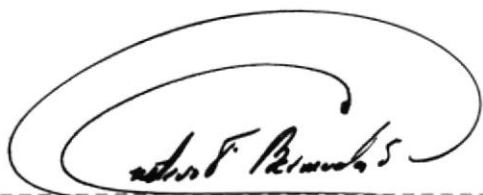
1987

A G R A D E C I M I E N T O

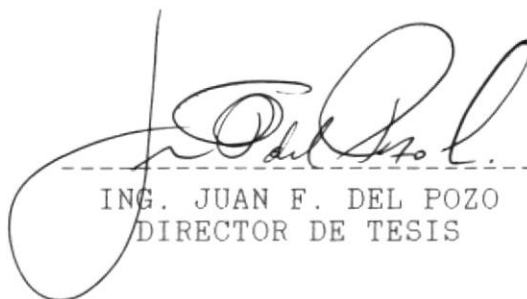
AL ING. JUAN DEL POZO, DIRECTOR DE TESIS, A FERNANDO AVILES, A MIGUEL VACAS, A MARIA PIA GARCIA, A LAS COMPANIAS SINFO-Q Y HARDTEC, Y A TODOS AQUELLOS QUE ME BRINDARON SU AYUDA Y COLABORACION PARA LA REALIZACION DEL PRESENTE TRABAJO.

DEDICATORIA

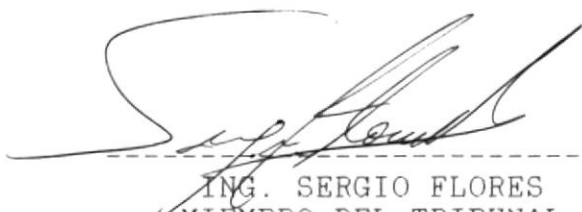
- A MIS PADRES
- A MIS HERMANOS
- A MIS AMIGOS
- A EDGAR .



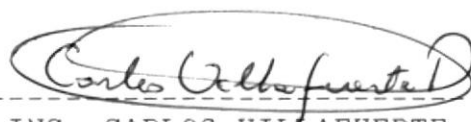
ING. GUSTAVO BERMUDEZ F.
SUB-DECANO DE LA FACULTAD
DE INGENIERIA ELECTRICA



ING. JUAN F. DEL POZO
DIRECTOR DE TESIS



ING. SERGIO FLORES
MIEMBRO DEL TRIBUNAL



ING. CARLOS VILLAFUERTE
MIEMBRO DEL TRIBUNAL

DECLARACION EXPRESA

"LA RESPONSABILIDAD POR LOS HECHOS, IDEAS Y DOCTRINAS EXPUESTOS EN ESTA TESIS, ME CORRESPONDEN EXCLUSIVAMENTE; Y, EL PATRIMONIO INTELECTUAL DE LA MISMA, A LA ESCUELA SUPERIOR POLITECNICA DEL LITORAL".

(REGLAMENTO DE EXAMENES Y TITULOS PROFESIONALES DE LA ESPOL).



RUTH EMPERATRIZ SANTANA VILLAMAR

RESUMEN

Este programa es una herramienta eficaz para la investigación de procesos físicos. Los bloques son los mismos que se utilizan en la simulación analógica y entrega la información buscada en un tiempo bastante corto, permitiendo analizar las salidas, tanto numérica como gráficamente, en todos los bloques del sistema a investigar.

Está diseñado para que sirva en el proceso enseñanza - aprendizaje de Simulación de Sistemas de Control con lo que su uso en las prácticas de Control Automático de los estudiantes de la Facultad de Ingeniería Eléctrica va a ser de mucha utilidad.

El programa está hecho de modo que su ejecución no implique mayor dificultad, de tal suerte que aprender a manejarlo tome muy poco tiempo.

Al utilizar los mismos bloques que se usan en simulación analógica, el paso de utilización de computador analógico a computador digital en simulación de sistemas continuos, no es muy perceptible, en cuanto a preparación de datos se refiere.

INDICE GENERAL

	PAG.
RESUMEN -----	VI
INDICE GENERAL -----	VII
INTRODUCCION -----	9
CAPITULO I	
SISTEMAS DINAMICOS CONTINUOS -----	11
1.1 Sistemas dinámicos continuos. -----	11
1.2 Modelaje de sistemas dinámicos continuos. ----	12
CAPITULO II	
CONSIDERACIONES GENERALES -----	17
2.1 Diagramas de bloques usados en simulación de sistemas dinámicos continuos. -----	17
CAPITULO III	
DISEÑO DEL PROGRAMA -----	22
3.1 Definición del programa. Especificaciones. --	22
3.2 Datos de entrada. -----	24
3.2.1 Preparación de los datos. -----	24
3.2.2 Ingreso de los datos. -----	27
3.3 Obtención de los resultados. -----	28

112	-----	BIBLIOGRAFIA
109	-----	CONCLUSIONES
101	-----	- Obtención de Resultados.
97	-----	- Datos para la Simulación.
89	-----	- Diagramas de Bloques.
82	-----	- Definición de las Ecuaciones.
81	-----	- Planteamiento del Problema.
81	-----	Ejemplo Práctico.
75	-----	- Ejecución del programa.
69	-----	- Entrada de datos.
63	-----	su descripción.
	-----	- Elementos que puede usar el programa con
63	-----	- Introducción.
63	-----	Manual del Usuario.
63	-----	APENDICE
58	-----	4.3 Listado y descripción de las variables usadas.
54	-----	4.2 Descripción de cada una de las subrutinas.
39	-----	4.1 Diagramas de Flujo.
38	-----	ELABORACION DEL PROGRAMA
		CAPITULO IV
36	-----	3.4.2 Presentación en impresora.
35	-----	3.4.1 Presentación en pantalla.
35	-----	3.4 Presentación de los Resultados.
30	-----	3.3.2 Método de Integración usado.
28	-----	3.3.1 Tratamiento de la información ingresada.

INTRODUCCION

El presente trabajo ha sido realizado para permitir la simulación de sistemas dinámicos continuos en microcomputadores IBM - PC.

No pretende ser más que una herramienta para la investigación de estos sistemas, supliendo en algo, la falta de medios a la que se enfrenta el investigador, sea docente o alumno, para la simulación de problemas específicos.

Está basado en un programa para la IBM 1130, el 1130 CSMP (Continuous System Modeling Program) que fue creado con el objeto de simular sistemas continuos complejos. Este programa fue de gran utilidad en las investigaciones y estudios de este tipo de sistemas, pese al inconveniente de trabajar en el IBM 1130, que tenía un tiempo de repuesta muy largo para la ejecución del programa.

Esta versión del CSMP puede correr en cualquier microcomputador compatible con los personales de IBM. Por las características de este tipo de equipo, la velocidad de respuesta se ha incrementado enormemente.

Por otro lado, la difusión de los microcomputadores personales permite que este programa pueda llegar a un mayor número de personas interesadas en investigar problemas de sistemas dinámicos continuos.

La misma Facultad de Ingeniería Eléctrica ha adquirido varios de estos equipos, con lo que este proceso de enseñanza-aprendizaje de simulación de sistemas, se verá ampliamente beneficiado.

Este programa permite el ingreso de datos desde teclado o desde un archivo previamente grabado en diskette. La salida de la información es por pantalla (Gráficos de Media Resolución) o por Impresora.

Ha sido realizado en lenguaje FORTRAN IV con subrutinas en ASSEMBLER que permiten una interacción más eficaz usuario-máquina.

Las pruebas del programa fueron hechas realizando una simulación que necesitaba el Sr. Miguel Vacas Quincha para la terminación de su tesis, parte de esta simulación está presentada en el Apéndice.

CAPITULO I

SISTEMAS DINAMICOS CONTINUOS

1.1 SISTEMAS DINAMICOS CONTINUOS.

Un sistema es una combinación de componentes que actúan conjuntamente y cumplen determinado objetivo. Estos objetivos pueden ser físicos o abstractos, así tenemos sistemas físicos, biológicos, económicos, etc,. El centro de nuestro estudio son los sistemas físicos, es decir, aquellos sistemas que involucran una variedad de cantidades físicas, como: voltaje y corrientes eléctricas, fuerzas y desplazamientos, volúmenes, flujos, temperaturas, etc.

Un sistema está formado por una colección de componentes interconectados, en los cuales están especificados dos juegos de variables dinámicas, las primeras llamadas entradas o excitaciones, y las segundas, salidas o respuestas.

Si las entradas y salidas de un sistema cambian en cualquier instante de tiempo, el sistema es llamado sistema continuo. Para resaltar más este hecho,

variables de entrada y salida son representadas como variables en función de tiempo.

Entonces, las excitaciones y respuestas son funciones que dependen de variables continuas (tiempo).

Un sistema cuyas salidas no dependen únicamente del valor actual de las entradas, sino que las respuestas en un instante dado dependen de valores pasados o futuros de las excitaciones, es un sistema dinámico.

Entonces, un sistema dinámico continuo, es un sistema formado por un conjunto de componentes cuyas entradas y salidas son funciones del tiempo y cuyas salidas dependen de valores presentes, pasados y/o futuros de las entradas.

1.2 MODELAJE DE SISTEMAS DINAMICOS CONTINUOS

La mayoría de los sistemas dinámicos ya sean mecánicos, eléctricos, térmicos, etc., pueden ser caracterizados por ecuaciones diferenciales.

Se puede obtener la respuesta de un sistema a una excitación dada, si se resuelven estas ecuaciones. Para obtener las ecuaciones se utilizan las leyes

físicas que gobiernan el sistema, por ejemplo, las leyes de Kirchoff para sistemas eléctricos o las de Newton para sistemas mecánicos, etc.

Se denomina modelo matemático a la descripción matemática de las características dinámicas del sistema. El primer paso y a la vez, el más importante, que se realiza en el análisis de un sistema es la elaboración de su modelo. Se pueden realizar un sinnúmero de modelos, sin embargo, para cada caso particular, será más ventajoso la utilización de un modelo específico en vez de otro.

Luego de obtener el modelo matemático de un sistema, se podrán usar diversas técnicas analíticas y computacionales para realizar el análisis y síntesis del sistema.

En la elaboración del modelo se debe tener presente que se necesita un compromiso entre la exactitud y la simplicidad, evitando caer en cualquiera de los dos extremos. El modelo a elaborarse debe ajustarse al sistema físico, siendo conveniente, a veces, ignorar ciertas linealidades y parámetros distribuidos que originan derivadas parciales. Si los efectos de estas características en la respuesta son pequeños, se logra un buen resultado en

el análisis del sistema.

Se debe tener presente que los parámetros distribuidos producen poco efecto en operaciones de baja frecuencia, pero estos parámetros pueden influenciar enormemente en la respuesta a alta frecuencia.

Es una buena costumbre, elaborar primero, un modelo simplificado para tener una idea global del mismo y, posteriormente, elaborar un modelo más complicado, que nos de una respuesta más exacta del sistema.

En los sistemas dinámicos continuos podemos establecer una diferencia en base a la linealidad de los mismos.

Un sistema que satisfaga el principio de superposición, es lineal, esto es, si y solo si:

$$H(ax_1 + bx_2) = aHx_1 + bHx_2$$

donde a y b son cualquier constante y x_1 y x_2 son señales de entrada.

Aunque muchas relaciones físicas son representadas por ecuaciones lineales, en la mayoría de los casos,

estas relaciones son no-lineales. En estos sistemas, no es aplicable el principio de superposición y el proceso para hallar la solución de este tipo de problema, se vuelve muy complicado. Es por esto, que frecuentemente se crean sistemas lineales equivalentes en reemplazo de los no-lineales. Estos sistemas son válidos en un rango restringido de operación.

Para análisis de sistemas con una sola entrada y una sola salida, el modelo matemático más conveniente es la utilización de la función de transferencia del mismo.

Se denominan funciones de transferencia a todas aquellas funciones que caracterizan las relaciones de entrada - salida de sistemas lineales invariantes en el tiempo.

La función de transferencia de un sistema está definida como la relación de la transformada de Laplace de la respuesta a la transformada de Laplace de la entrada, bajo la suposición de que todas las condiciones iniciales son cero. Esta función de transferencia es una expresión en términos de los parámetros del sistema y es independiente de la función excitadora o de entrada, pero no provee

ninguna información respecto de la estructura física del sistema.

Por ejemplo, si un sistema invariante en el tiempo, está definido por la ecuación diferencial:

$$a_0 y^{(n)} + a_1 y^{(n-1)} + \dots + a_{n-1} y' + a_n y = b_0 x^{(n)} + b_1 x^{(n-1)} + \dots + b_{n-1} x' + b_n x$$

donde x es la entrada y la salida es y , se tendrá que:

$$\text{Func. Transferencia} = G(s) = \frac{Y(s)}{X(s)} = \frac{b_0 s^n + b_1 s^{n-1} + \dots + b_{n-1} s + b_n}{a_0 s^n + a_1 s^{n-1} + \dots + a_{n-1} s + a_n}$$

CAPITULO II

CONSIDERACIONES GENERALES

2.1 DIAGRAMAS DE BLOQUES USADOS EN SIMULACION DE SISTEMAS DINAMICOS CONTINUOS.

Habíamos analizado anteriormente, que un sistema es un conjunto de componentes y que cada uno de ellos cumple una función. Estas funciones pueden ser mostradas por medio de un diagrama de bloques.

Un diagrama de bloques no es más que una representación gráfica de las funciones de cada componente de un sistema y del flujo de las señales. El diagrama de bloques permite visualizar de una manera real el sistema, a diferencia de una representación matemática, en la cual el sistema es visto en forma abstracta.

Incluso es más fácil comprender el funcionamiento de un sistema por medio del diagrama de bloques que examinando el sistema físico en sí. Todas las variables del sistema se enlazan entre sí por medio

de los bloques funcionales. Cada uno de estos bloques es un símbolo que representa la operación matemática realizada sobre la señal de entrada para obtener la señal de salida, es decir, que en el bloque se coloca la función de transferencia correspondiente. Los bloques se unen unos con otros por medio de flechas unidireccionales, que muestran el flujo de la señales. Por ejemplo:

$$G(s) = \frac{Y(s)}{X(s)}$$

se representará:

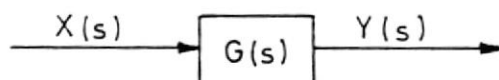


Fig. 2.1: $Y(s) = G(s) X(s)$

La flecha que apunta al bloque es la señal de entrada y la que se aleja es la señal de salida.

Los bloques funcionales pueden estar relacionados por medio de un detector de error, que emite una señal que es la diferencia entre la referencia de entrada y la señal de realimentación del sistema de control. Por ejemplo:

$$\frac{G(s)}{1 + G(s)} = \frac{Y(s)}{X(s)}$$

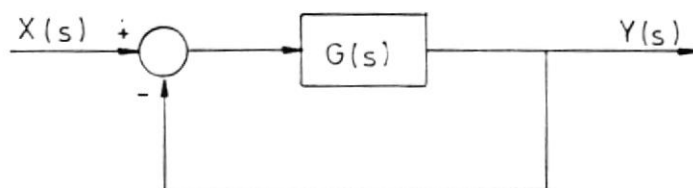


Fig. 2.2

Para elaborar el diagrama de bloques de un sistema, primero se escriben las ecuaciones del sistema, las mismas que deberán describir el comportamiento dinámico de cada componente, luego, se toman las transformadas de Laplace de esas ecuaciones, suponiendo condiciones iniciales iguales a cero, y se representa cada ecuación en forma de Laplace como un bloque. Se juntan los elementos y se obtiene un diagrama de bloques completo.

Para representar un sistema con varias señales de entrada y salida, se usa una interconexión de bloques.

Por ejemplo, las ecuaciones:

$$Y_1(s) = G_{11}(s)X_1(s) + G_{12}(s)X_2(s)$$

$$Y_2(s) = G_{21}(s)X_1(s) + G_{22}(s)X_2(s)$$

representa el sistema:

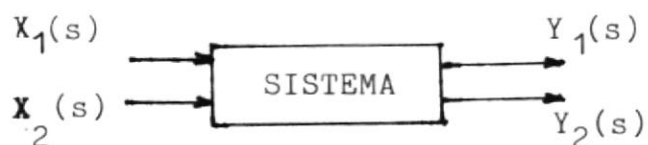


Fig. 2.3

En donde, $G_{ij}(s)$ representa la función de transferencia que relaciona la variable de salida i , con la entrada j .

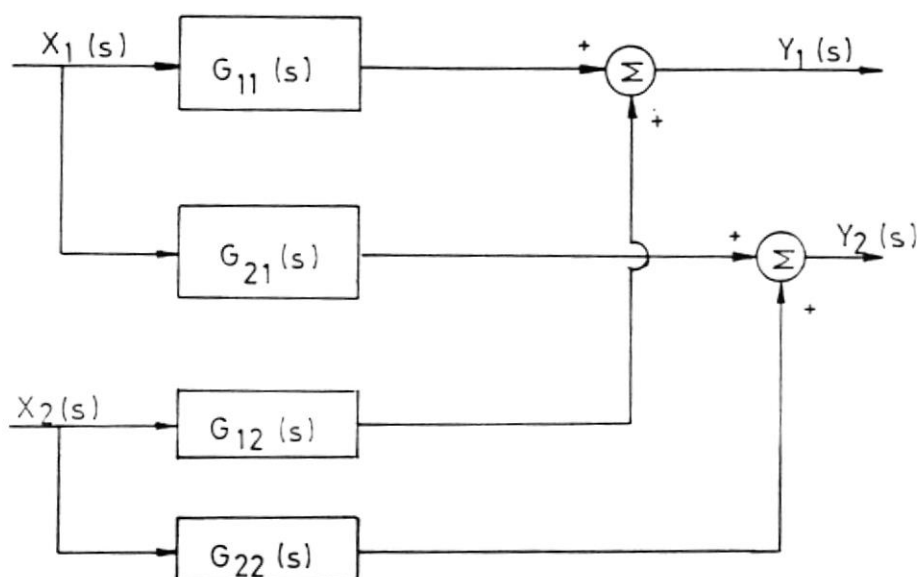


Fig. 2.4

En general, para J entradas e I salidas, escribimos las ecuaciones simultáneas en forma matricial, así:

$$\begin{bmatrix} Y_1(s) \\ Y_2(s) \\ \vdots \\ Y_I(s) \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) & \dots & G_{1J}(s) \\ G_{21}(s) & G_{22}(s) & \dots & G_{2J}(s) \\ \vdots & \vdots & \ddots & \vdots \\ G_{I1}(s) & G_{I2}(s) & \dots & G_{IJ}(s) \end{bmatrix} \begin{bmatrix} X_1(s) \\ X_2(s) \\ \vdots \\ X_J(s) \end{bmatrix}$$

Una vez que tenemos un modelo en diagrama de bloques para un sistema determinado, se puede utilizar un computador analógico o digital para simular el sistema. De esta forma, para investigar un diseño no es necesario construirlo, sino que con las condiciones reales del sistema y señales de entrada reales, se puede investigar las señales de salida. Con un computador analógico, se simula el sistema usando la analogía entre el voltaje del amplificador electrónico y la variable del sistema que se modela. Por lo general, un computador analógico tiene las funciones de integración, de multiplicación entre dos variables, sumadores de varias entradas, amplificación y constantes, funciones con las que ya puede realizar una simulación. En un computador digital, se crea un programa en el cual se introducen funciones tales como: integración, sumas de señales, multiplicación entre dos señales, etc., es decir las mismas funciones que se tienen en el computador analógico. La ventaja del uso del computador digital es la de poder utilizar funciones no lineales y generadores de números al azar, con lo que la simulación puede realizarse de un modo más eficaz. Por otro lado, la simulación del sistema se la hace en función del tiempo, con lo cual, la simulación es más exacta, pues se trabaja con las variables reales del sistema.

CAPITULO III

DISEÑO DEL PROGRAMA

3.1 DEFINICION DEL PROGRAMA. ESPECIFICACIONES.

El Programa CSMP es un programa que sirve para la simulación de un sistema. Los datos de entrada de este programa representan los elementos de un computador analógico.

El programa puede simular 30 elementos diferentes, de los cuales, 5 son definibles por el usuario, de modo que pueda contar con funciones especiales como seno, coseno, exponencial, etc..

Los elementos con su representación gráfica y su representación en lenguaje son mostrados en la tabla Tipos de Elementos para Simulación del Apéndice.

Este programa puede manejar hasta 75 bloques funcionales, pero necesita que por lo menos, uno de ellos sea un integrador.

El número máximo de integradores que pueden haber en

un juego de datos es 25.

El programa permite la generación de funciones con un máximo de hasta 3 elementos generadores por juego de datos y un máximo de 25 unidades de retardo.

Cada uno de los elementos define una relación funcional que involucra tres variables de entrada y tres parámetros. La salida de un elemento es una cantidad escalar que viene dada por la relación funcional. Cada elemento es representado por un símbolo de un sólo carácter.

El ingreso de datos se hace directamente desde el teclado o por medio de un archivo previamente grabado en un diskette. Los resultados son mostrados en impresora o en pantalla.

Para ser ejecutado, el CSMP necesita de un equipo con la siguiente configuración mínima:

- Computador IBM-PC o Compatible.
- 256 Kbytes de memoria RAM.
- Un Manejador de Diskette.
- Monitor Monocromático.

Pero sería aconsejable tener:

- Monitor a Colores o Monocromático con capacidad para gráficos.
- Impresora de 132 columnas.

3.2 DATOS DE ENTRADA.

3.2.1 PREPARACION DE LOS DATOS.

Cuando se tenga definido el sistema que se va a simular, se deben establecer claramente sus ecuaciones para poder realizar el diagrama de bloques. Se debe tener presente que no pueden existir más de 75 bloques funcionales. A cada bloque debe asignársele un número único entre 1 y 75 que lo identificará en toda la simulación.

Para cada bloque, se debe identificar el tipo del elemento, sus entradas (de qué bloque provienen) si las tuviere, sus parámetros si los tuviere y si se trata de un generador de función, se deben establecer los valores de intercepción.

La forma más adecuada para preparar los datos.

antes de ingresarlos al computador, es separando la información en tres grupos:

- Datos de Configuración
- Condiciones Iniciales y Parámetros
- Generador de Funciones

DATOS DE CONFIGURACION.- en este grupo de datos se establece, para cada bloque, la siguiente información:

Nombre de la Salida.- que tiene un máximo de 16 caracteres.

Número del Bloque.- debe estar entre 1 y 75 y no puede ser repetido.

Tipo.- el tipo del elemento al que pertenece el bloque.

Entrada 1.- el número del bloque de donde proviene la entrada 1, si es que se utiliza esta entrada.

Entrada 2.- el número del bloque de donde proviene la entrada 2, si es que se utiliza esta entrada.

Entrada 3.- el número del bloque de donde proviene la entrada 3, si es que se utiliza esta entrada.

CONDICIONES INICIALES Y PARAMETROS.- para aquellos bloques que requieran de condiciones iniciales y/o parámetros, se necesita la siguiente información:

Nombre del Parámetro.- que tiene un máximo de 16 caracteres.

Número del Bloque.- debe corresponder al número del bloque cuyos parámetros se van a ingresar.

Condición Inicial o Primer Parámetro.- para aquellos bloques que tienen condición inicial, se ingresa aquí esa información, o de lo contrario, se registra el primer parámetro para aquellos bloques que lo utilicen.

Segundo Parámetro.- se registra el dato del segundo parámetro del bloque si es que lo tiene.

Tercer Parámetro.- se registra el dato del tercer parámetro del bloque si es que lo tiene.

GENERADOR DE FUNCIONES.- este grupo de datos sólo se necesita cuando se ha especificado un Generador de funciones. Se debe registrar:

Número del Bloque.- que corresponde al número del bloque generador de función.

Puntos de Intercepción.- se deben ingresar los 11 puntos de intercepción necesarios para generar la función.

3.2.2 INGRESO DE LOS DATOS

El programa CSMP permite el registro de los datos a través del teclado o por medio de un archivo creado previamente con un editor de texto.

Para poder ingresar la información, ésta debe ser agrupada como se vio en la Sección 3.2.1.

Si el ingreso se lo hace por medio de un editor de texto, primero, se ingresan los datos de configuración, luego, las condiciones iniciales y parámetros y, finalmente, los generadores de funciones. Para cada grupo de datos, se debe ingresar una línea de datos por bloque y entre grupos de datos, se debe dejar una línea en blanco.

Más información sobre el ingreso de los datos para la simulación, la encontrará en el

Apéndice, en la sección de Entrada de Datos.

3.3 OBTENCION DE LOS RESULTADOS

3.3.1 TRATAMIENTO DE LA INFORMACION INGRESADA

Las salidas de los bloques del Sistema son evaluadas cada intervalo de integración. Esta evaluación se la hace siguiendo el orden en que fueron clasificados los elementos. La evaluación de las salidas de los bloques se la hace en la subrutina CSM11.

En esta subrutina merece especial atención el método de evaluación que presentan las relaciones funcionales de algunos de los elementos. El elemento Generador de Números Aleatorios, utiliza un método de congruencias para generarlos.

Estos números son realmente pseudoaleatorios pues son determinísticos, debido a que los procesos aritméticos que se incluyen en el cálculo, determinan cada término de la sucesión. El método aplicado es el multiplicativo de congruencias, en el cual se parte de un valor inicial, el cual es

multiplicado por un factor. Se toman los cuatro últimos dígitos del producto y se lo divide para 10000 y ese es el valor aleatorio. El número aleatorio se lo multiplica por 10000 y este resultado se lo multiplica por el factor y el ciclo se repite. En el programa se asegura que se pueden generar hasta 500 números sin que se repita la serie.

En los Generadores de Funciones, su relación funcional se calcula en base a un proceso de interpolación lineal entre los valores de intercepción especificados.

Las funciones implícitas $Y = f(x, y)$ se las trata a través del elemento WYE. El primer parámetro de este elemento es el criterio de error. Cuando se tiene un valor estimado de Y, se lo compara con el anterior y se comprueba si hay convergencia considerando el criterio de error, evalúa la expresión:

$$1.0 - \frac{f(Y_n, X)}{Y_{n-1}}$$

y la compara con el criterio de error especificado. Si el criterio es excedido, un

nuevo estimado de Y , es calculado de la siguiente manera:

$$Y_{n+1} = (1.0 - P_2) f(Y_n, X) + P_2 Y_n$$

P_2 es el segundo parámetro del elemento WYE y representa un factor que se aplica a la diferencia entre dos estimados sucesivos para obtener una corrección apropiada.

3.3.2 METODO DE INTEGRACION USADO

El cálculo principal dentro del proceso de simulación es la de integración. Como en todo sistema se resuelve por lo menos una ecuación diferencial es de mucha importancia el método de integración usado.

Si la ecuación diferencial es:

$$y' = f(x, y)$$

y se conoce la solución numérica en un punto, (X_n, Y_n) podemos calcular un valor aproximado Y_{n+1} resolviendo para $Y (X_{n+1})$, donde

$$X_{n+1} - X_n = h$$

Podemos resolverlo usando dos técnicas:

- Usando la Expansión de la serie de Taylor.
- Usando la definición de la integral definida

La expansión de la serie de Taylor se puede expresar en la forma:

$$y(x_{n+1}) = y(x_n) + hy'(x_n) + \frac{h^2}{2} y''(x_n) + \dots + \frac{h^p}{p!} y^{(p)}(x_n) + T_{n+1}$$

Donde,

$$T_{n+1} = \frac{h^{p+1} y^{(p+1)}(\bar{x}_{n+1})}{(p+1)!}, \quad x_n < \bar{x}_n < x_{n+1}$$

y denota el error por truncación de la serie de Taylor después del término p.

La expansión de la serie de Taylor puede ser representada en la forma:

$$y(x_{n+1}) = y(x_n) + h f(x_n) + \frac{h^2}{2} f'(x_n) + \dots + \frac{h^p}{p!} f^{(p)}(x_n)$$

y se desecha el término T_{n+1} .

La evaluación de las derivadas de orden p muy alto,

puede resultar un proceso muy engorroso. Por esta razón, existen métodos numéricos que usan de modo indirecto la expansión de Taylor. Uno de estos métodos es el de Runge - Kutta.

En general los métodos de Runge - Kutta evalúan $f(x,y)$ en más de un punto de las cercanías de (x_n, y_n) , vez de evaluar las derivadas de $f(x,y)$.

El método de Runge-Kutta de orden p es equivalente al desarrollo de la serie de Taylor truncada en el término p .

METODO DE RUNGE - KUTTA DE SEGUNDO ORDEN.- Este método ha sido desarrollado asumiendo que y_{n+1} se puede calcular como:

$$y_{n+1} = y_n + h[K_1 f(x_n, y_n) + K_2 f(x_n + A h, y_n + B h f(x_n, y_n))]$$

que es equivalente a la expansión cuadrática de Taylor:

$$y_{n+1} = y_n + h \left[f(x_n, y_n) + \frac{h}{2} f'(x_n, y_n) \right]$$

Relacionando estas dos expresiones, podemos obtener los valores de las constantes K_1 , K_2 , A y B .

$$K_1 + K_2 = 1$$

$$K_2 A = 1/2$$

$$K_2 B = 1/2$$

De donde:

$$K_1 = 1 - K_2$$

$$A = 1/2K_2$$

$$B = 1/2K_2$$

Si escogemos $K_2 = 0.5$, entonces:

$$K_1 = 0.5$$

$$A = 1$$

$$B = 1$$

Obtenemos:

$$y_{n+1} = y_n + \frac{h}{2} [f(x_n, y_n) + f(x_{n+h}, y_n + hf(x_n, y_n))]$$

Este caso especial se conoce como el método de Euler mejorado.

METODO DE RUNGE-KUTTA DE CUARTO ORDEN.- Este método calcula Y_{n+1} como:

$$y_{n+1} = y_n + \frac{K_1 + 2K_2 + 2K_3 + K_4}{6}$$

Donde,

$$K_1 = h f(x_n, y_n)$$

$$K_2 = h f(x_n + h/2, y_n + K_1/2)$$

$$K_3 = h f(x_n + h/2, y_n + K_2/2)$$

$$K_4 = h f(x_n + h, y_n + K_3)$$

Si $y'(x)$ es independiente de y , entonces:

$$K_1 = h f(x_n)$$

$$K_2 = h f(x_n + h/2)$$

$$K_3 = h f(x_n + h/2)$$

$$K_4 = h f(x_n + h)$$

de donde,

$$y_{n+1} = y_n + h/6 [f(x_n) + 4 f(x_n + h/2) + f(x_n + h)]$$

Luego, de hacer un análisis de los métodos numéricos para el cálculo de la Integral, se optó por utilizar uno de los métodos de Runge - Kutta.

Se hicieron diferentes pruebas, primero, utilizando el método de Runge - Kutta de Segundo Orden, y

finalmente, Runge - Kutta de Cuarto Orden. Entre uno y otro método no hubo una diferencia tan marcada en cuanto a error se refiere, pero en cambio, el número de iteraciones que realiza Runge - Kutta de Cuarto Orden, aumentaron de modo considerable el tiempo de respuesta del programa.

Debido a esto último, se decidió utilizar en el programa, el Método de Runge - Kutta de Segundo Orden.

3.4 PRESENTACION DE LOS RESULTADOS

El CSMP presenta los resultados en impresora y por pantalla. Para una mejor presentación de la información en la impresora y en los gráficos de alta resolución es conveniente primero, seleccionar la opción de datos por pantalla, para saber los máximos y mínimos de las variables en los bloques.

3.4.1 PRESENTACION EN PANTALLA

El CSMP tiene dos tipos de resultados que pueden ser presentados en la pantalla: DATOS POR PANTALLA y GRAFICOS EN PANTALLA.

El programa interroga sobre los numeros de

bloques cuyas salidas se desean conocer (4 salidas), el tiempo total de la corrida y el intervalo de tiempo al que se desea mostrar las salidas de los bloques.

Para cada intervalo de tiempo, se presentan los valores que tienen las salidas de los bloques.

GRAFICOS EN PANTALLA.- Solo se realiza gráficos en pantalla para aquellos microcomputadores que tienen tarjeta para gráficos.

El programa interroga sobre el número del bloque cuya salida se desea graficar, el tiempo total de la corrida, el intervalo de tiempo al que se desea graficar los puntos y los valores máximos y mínimos a graficarse.

Estos gráficos pueden imprimirse si se tiene una impresora gráfica y si antes de ejecutar el CSMP, se ha corrido un programa del sistema operativo llamado GRAPHICS.

3.4.2 PRESENTACION EN IMPRESORA

Los resultados de la ejecución del CSMP pueden ser obtenidos en impresora. Al elegir esta opción, se podrá tener la salida de los bloques

de nuestro interés (4 bloques) y, además, una gráfica de la salida de uno de los bloques anteriores.

Para realizar esto, el CSMP averigua sobre el tiempo total de impresión, el número de bloque cuya salida se desea graficar, el valor máximo y mínimo que presenta la salida de este bloque y los números de tres bloques para imprimir los valores de sus salidas.

CAPITULO IV

ELABORACION DEL PROGRAMA

Para la elaboración de este programa, se tomó como esqueleto principal del mismo, el programa realizado por IBM para el IBM 1130, el 1130 CSMP.

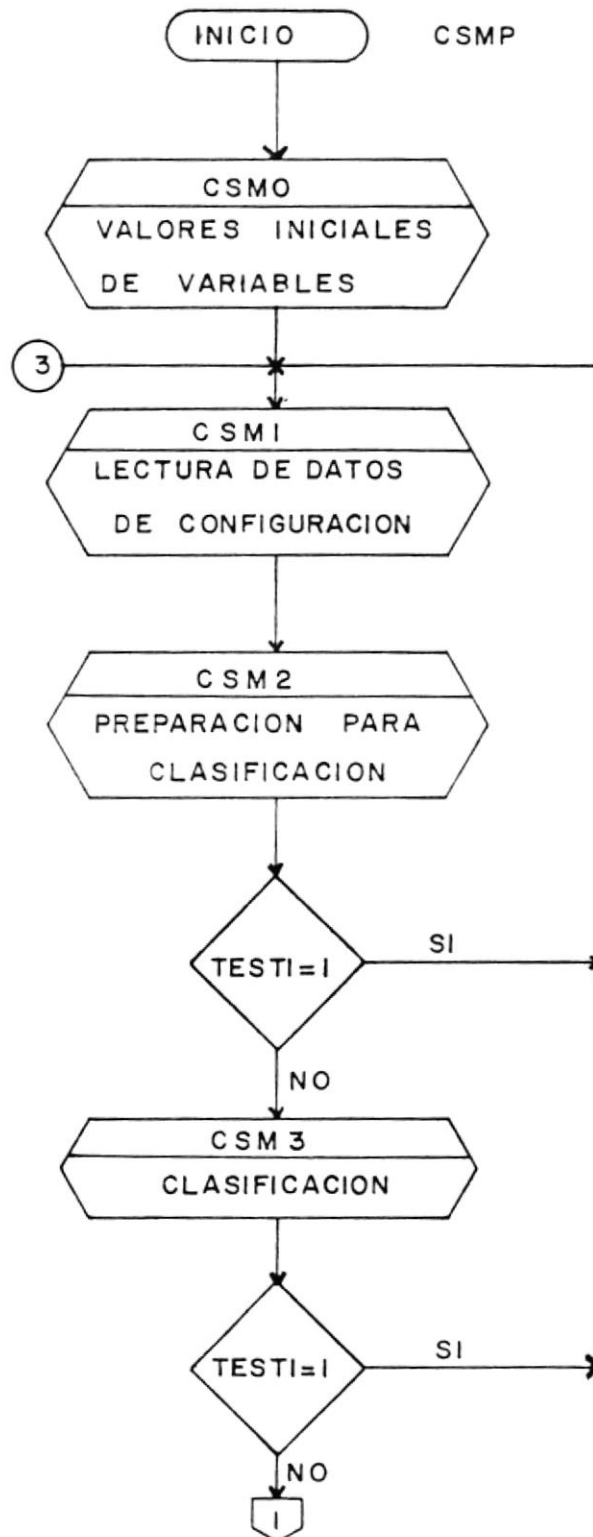
Luego de analizar varias opciones para la realización del programa, entre ellas, desarrollarlo en Lenguaje 'C', PASCAL o BASIC, se decidió finalmente realizarlo en FORTRAN, lenguaje en que IBM desarrolló su 1130 CSMP.

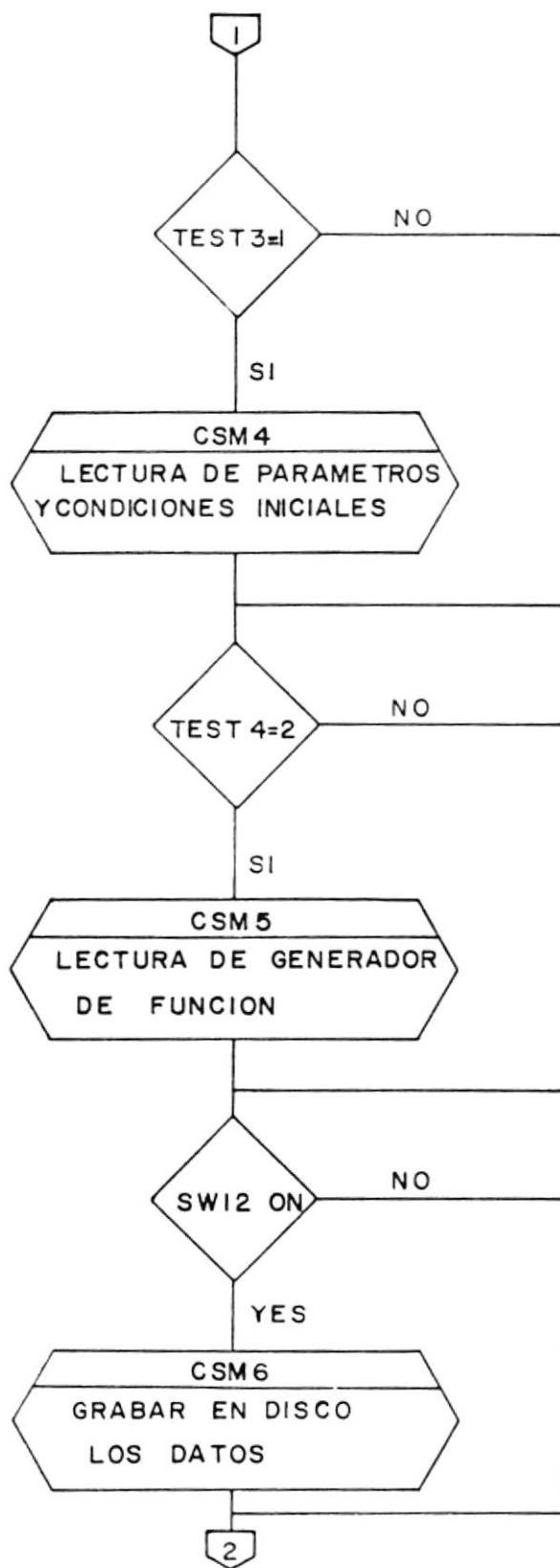
La gran desventaja del FORTRAN frente a otros lenguajes, es la dificultad que presenta en el manejo de pantallas para ingreso y salida de datos. Esta dificultad se la superó por medio del uso de subrutinas en lenguaje ASSEMBLER, que permiten el control sobre el proceso de lectura y escritura de la información.

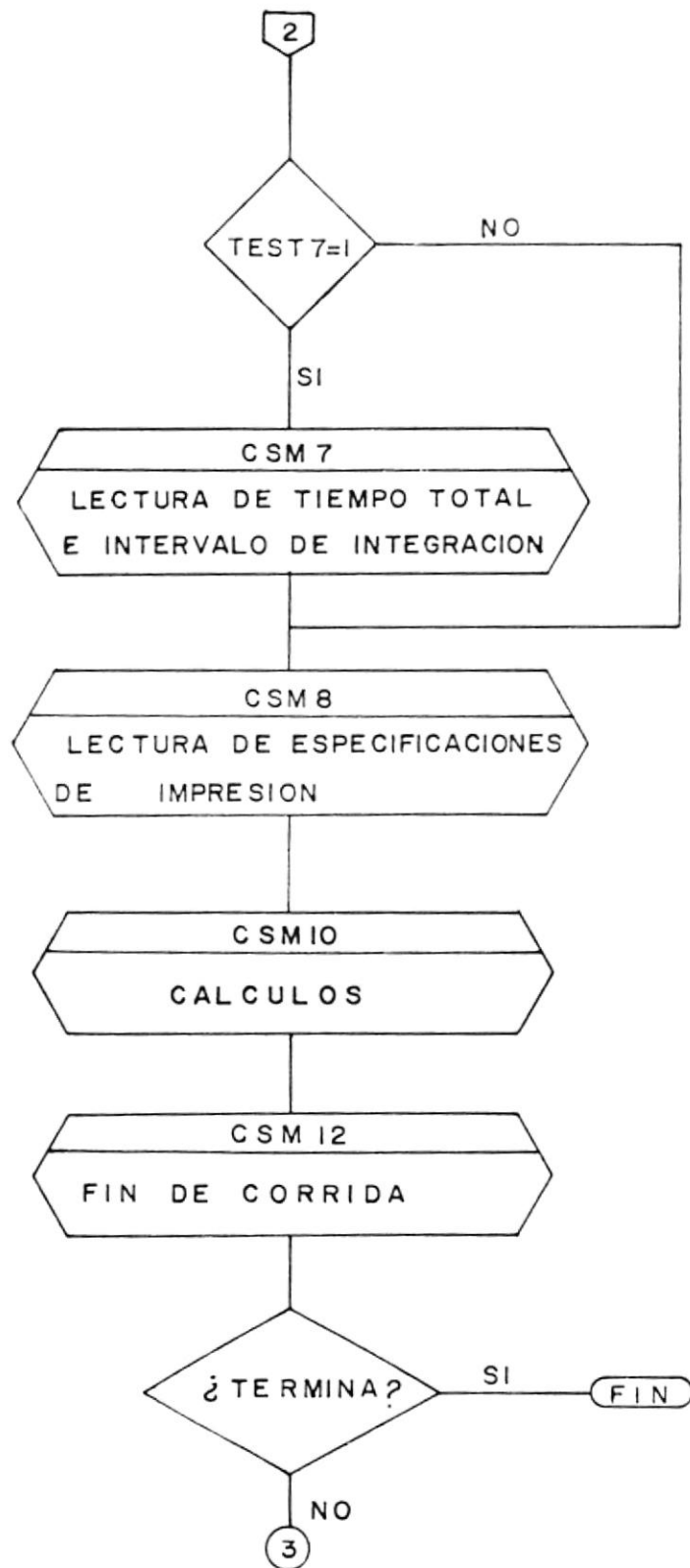
También, se hizo uso del ASSEMBLER para poder realizar los gráficos de alta resolución.

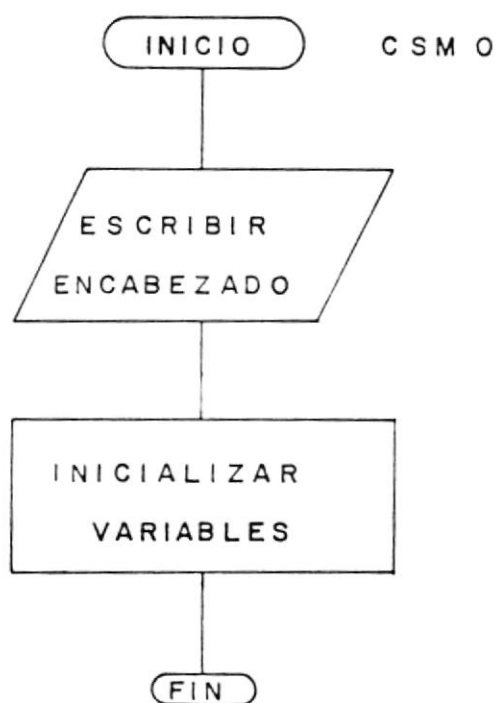
Todas estas subrutinas realizadas en ASSEMBLER, fueron agrupadas en una biblioteca llamada UTIL.LIB.

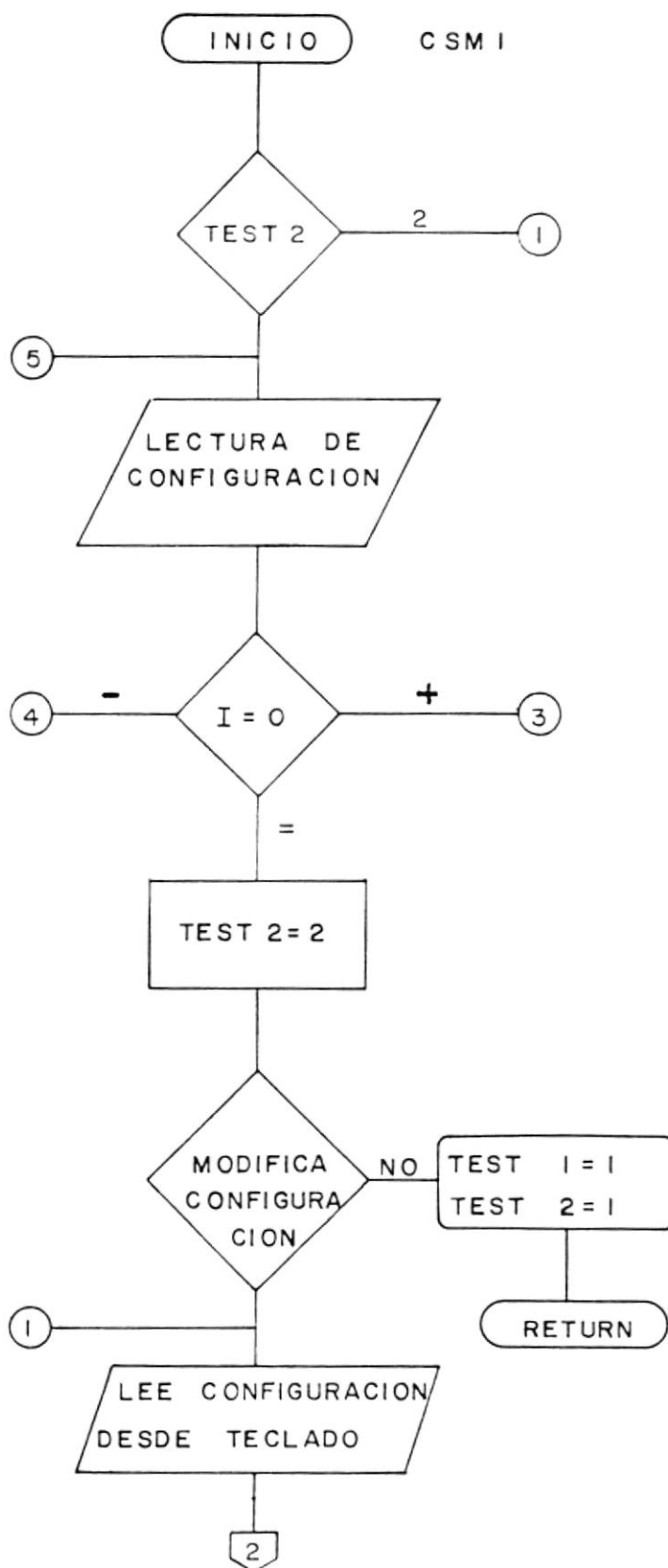
4.1 DIAGRAMAS DE FLUJO

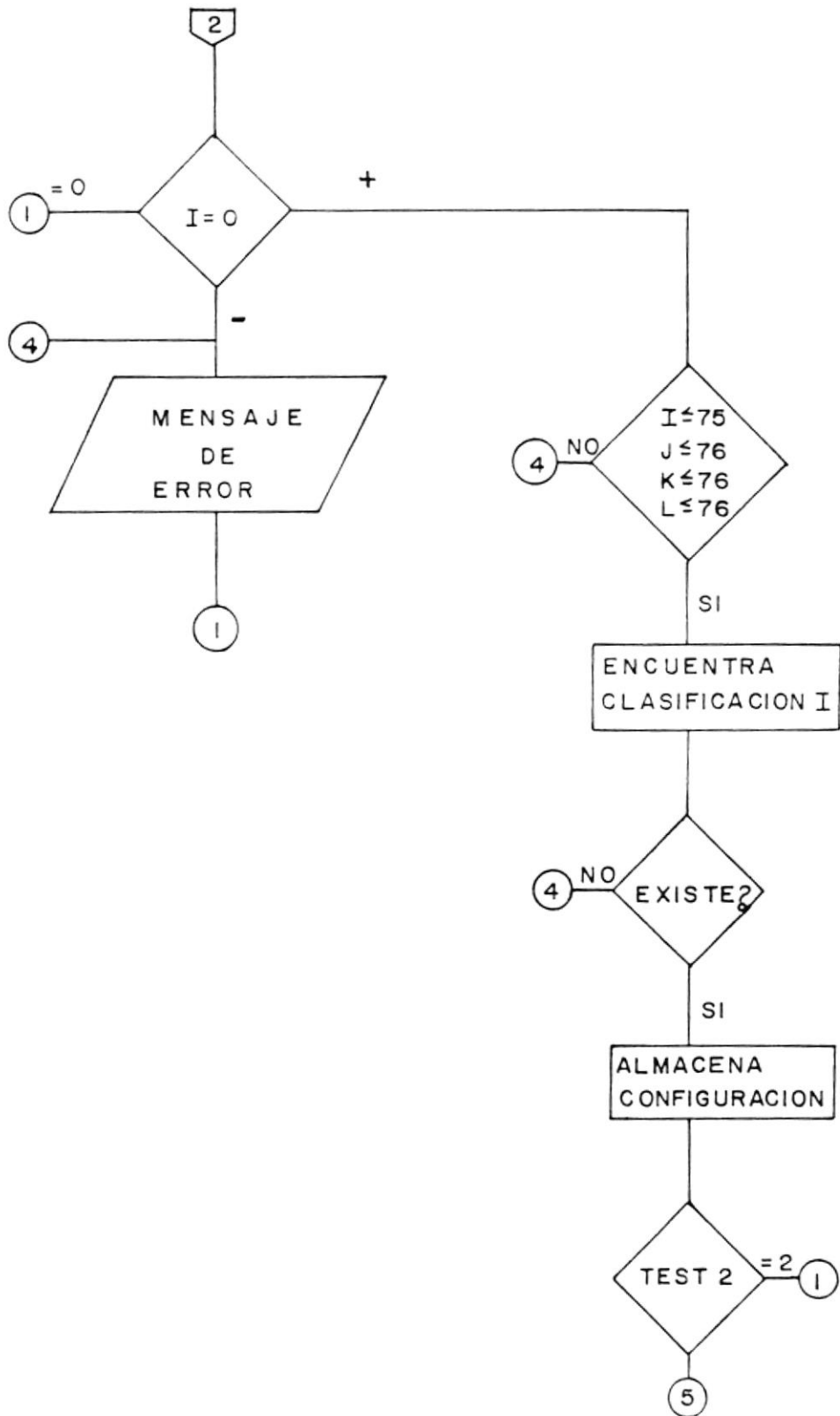


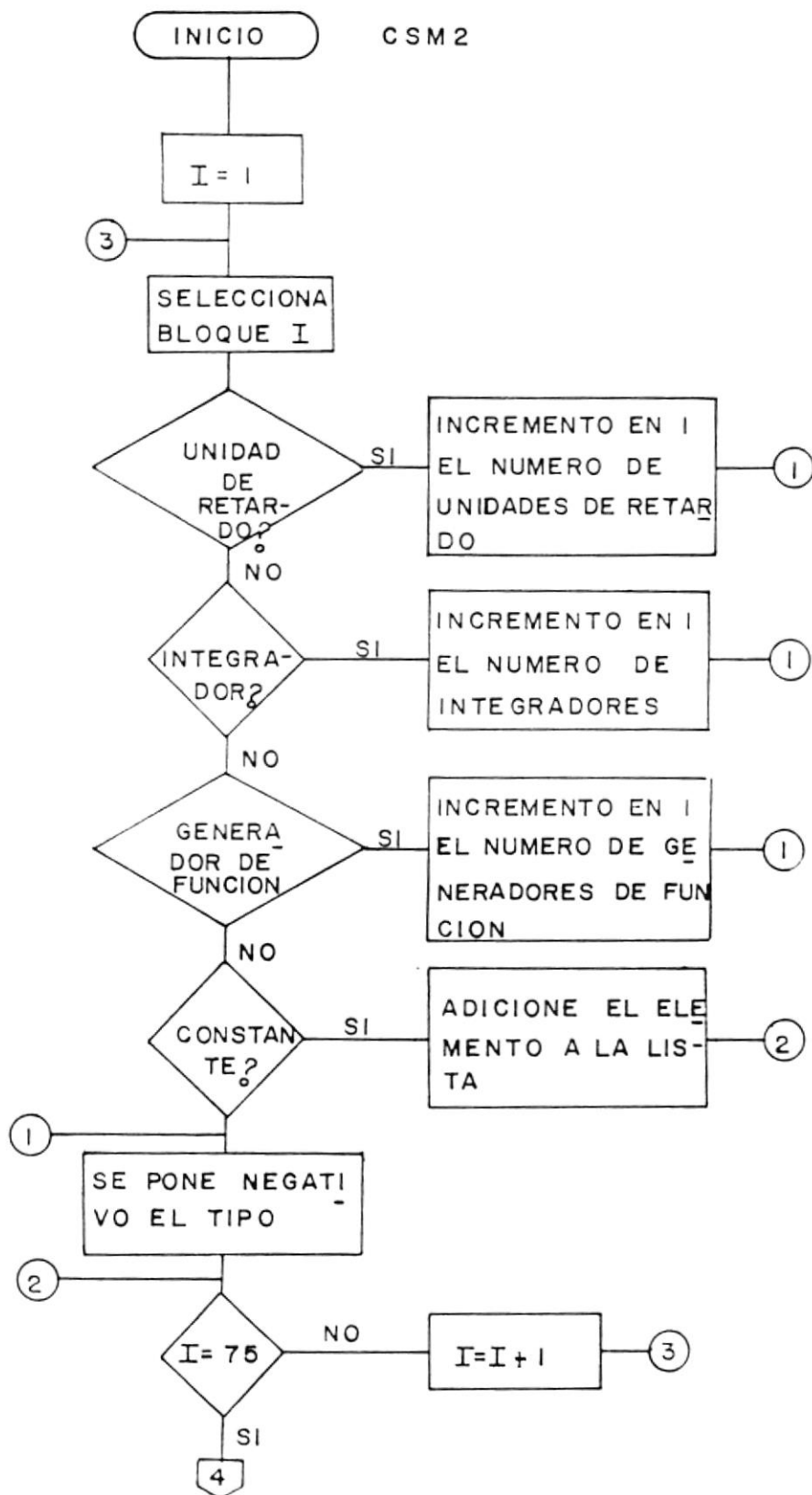


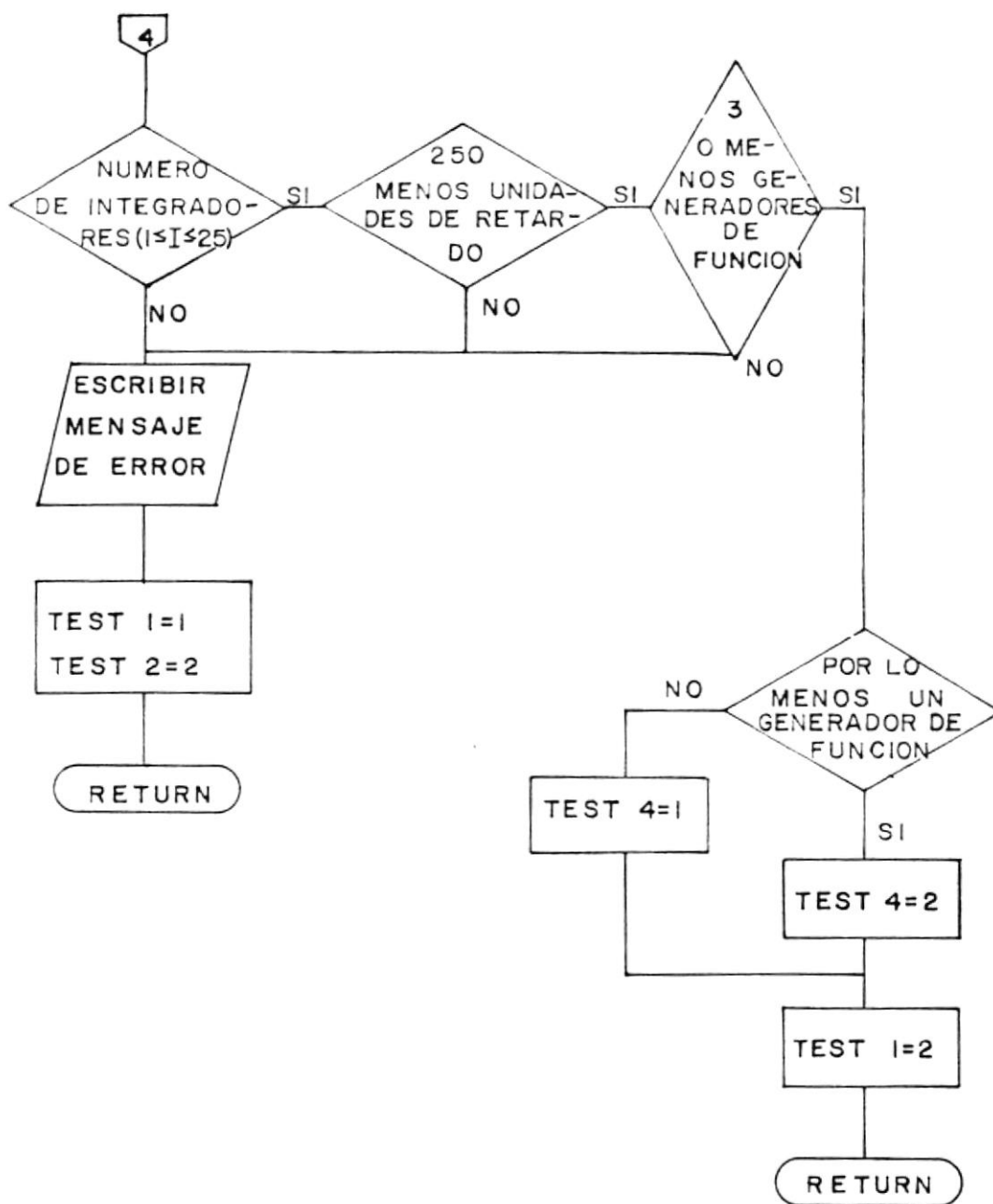


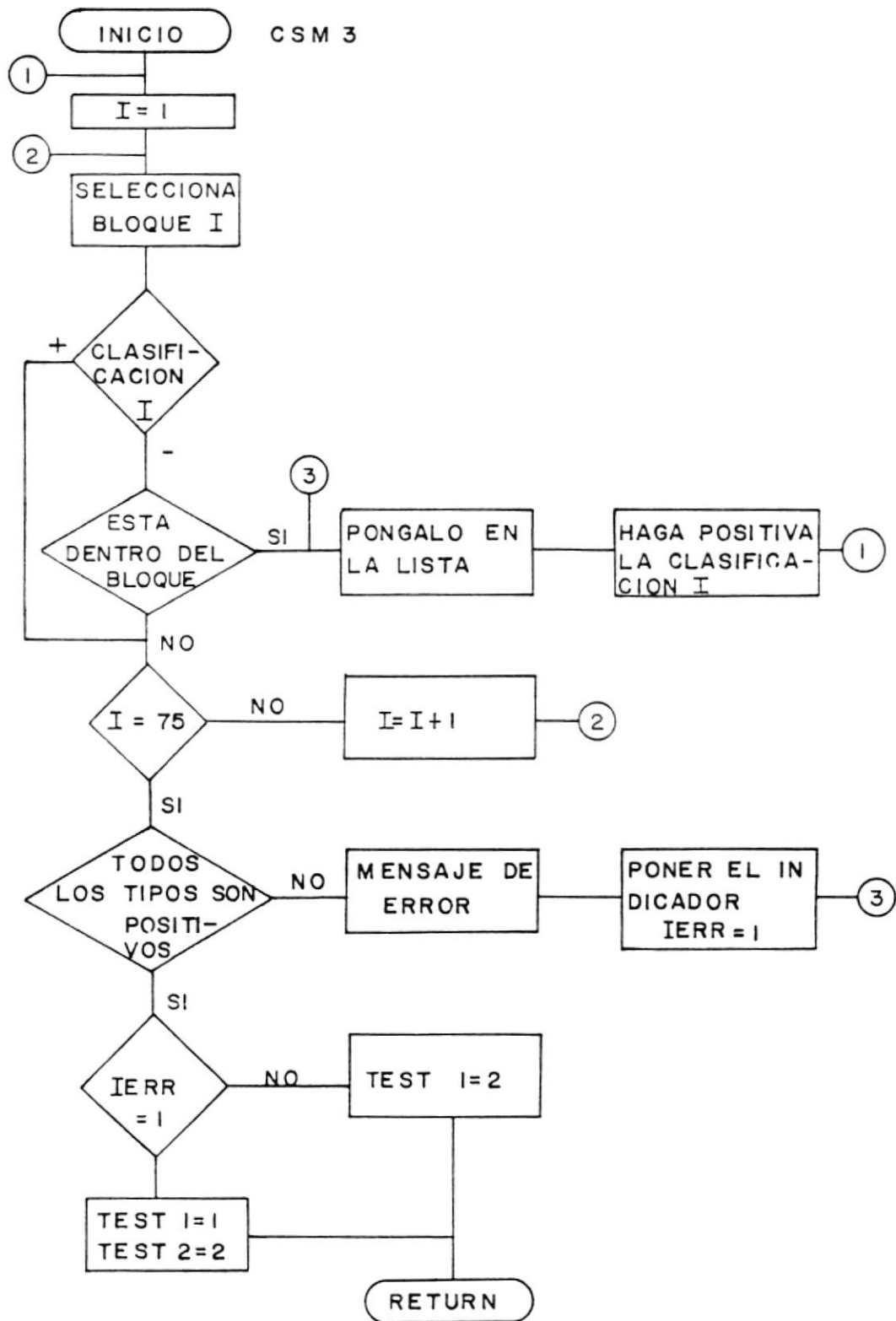


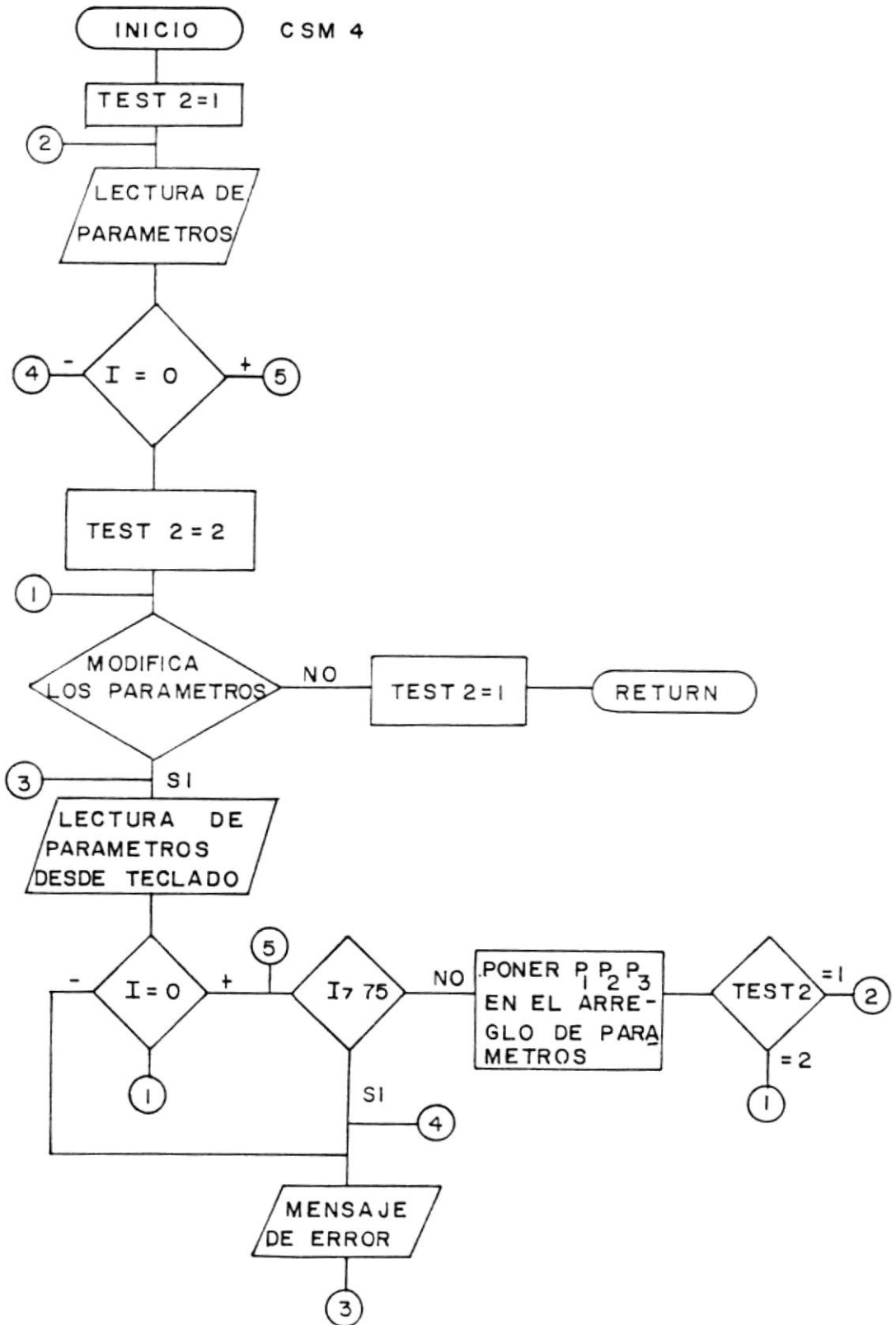


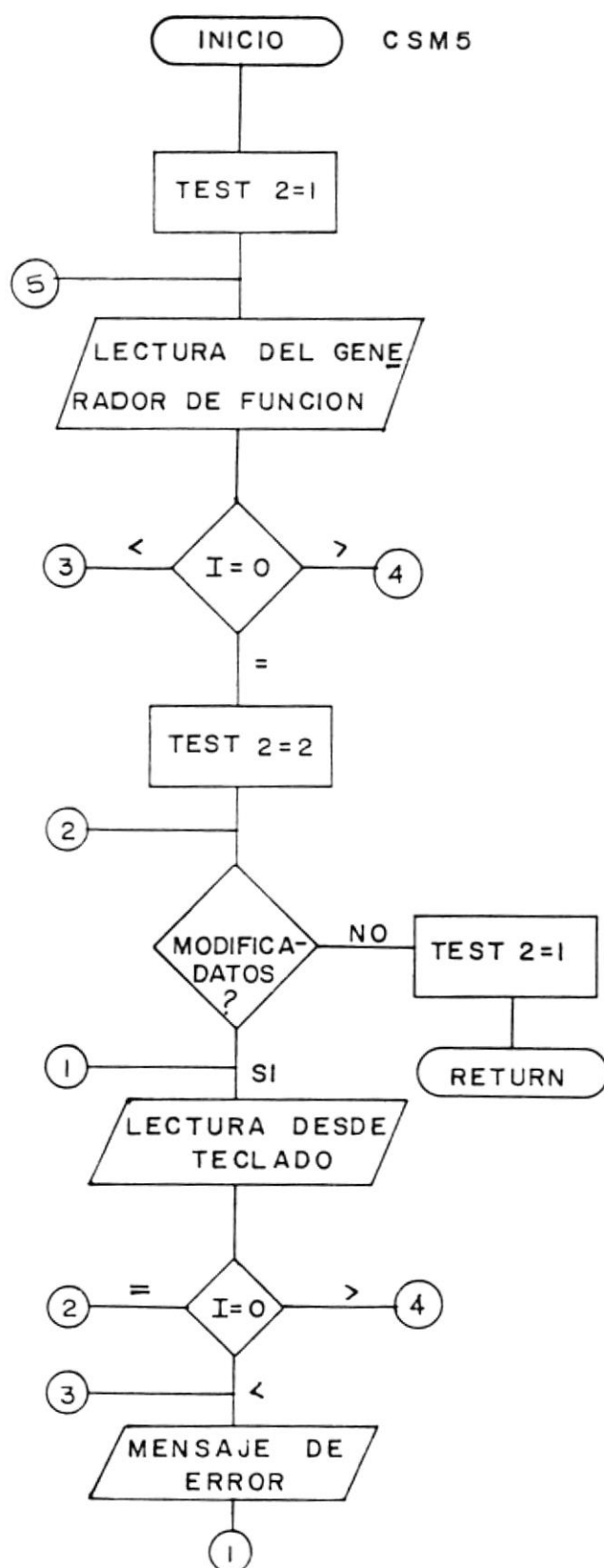


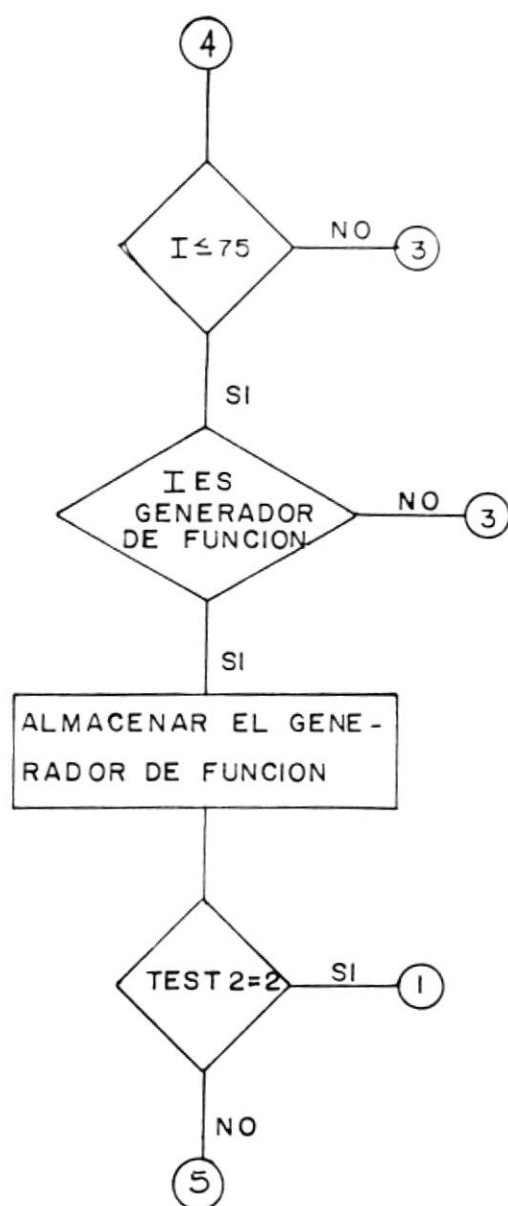


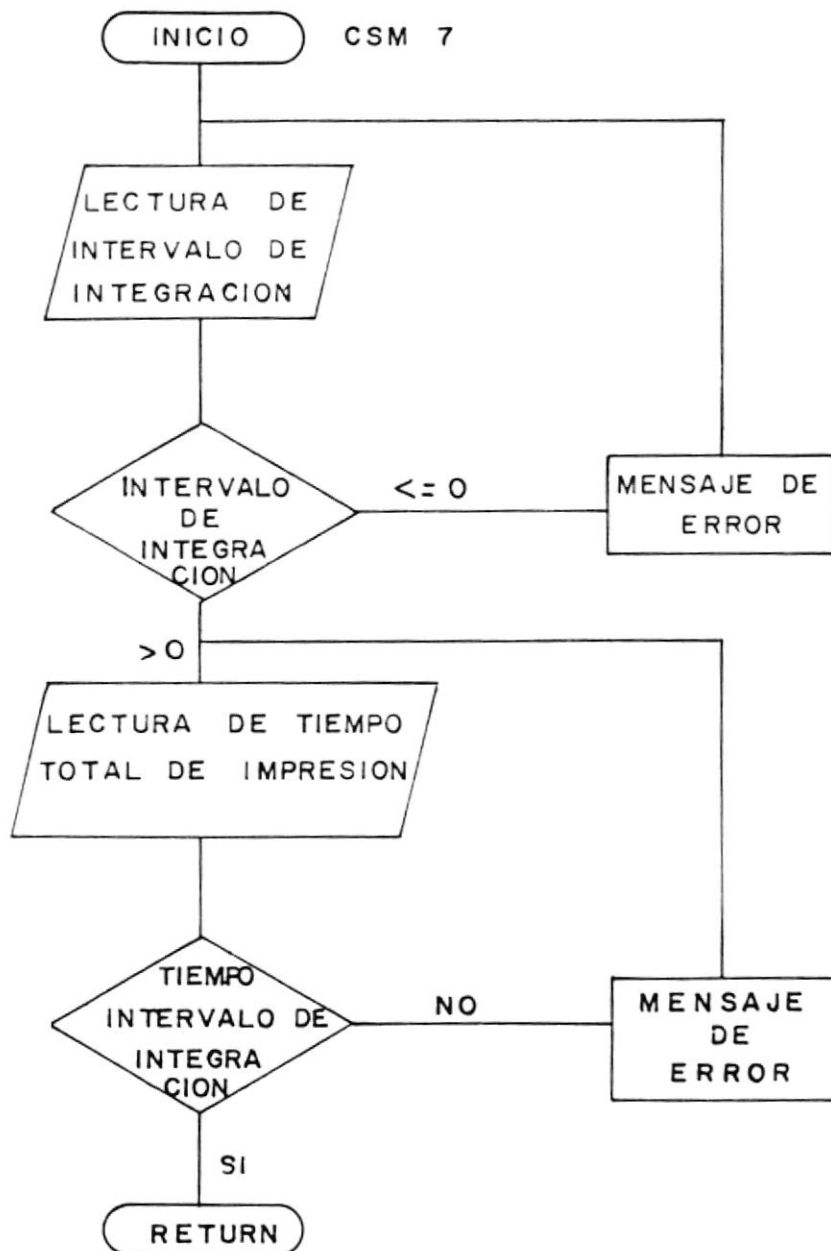


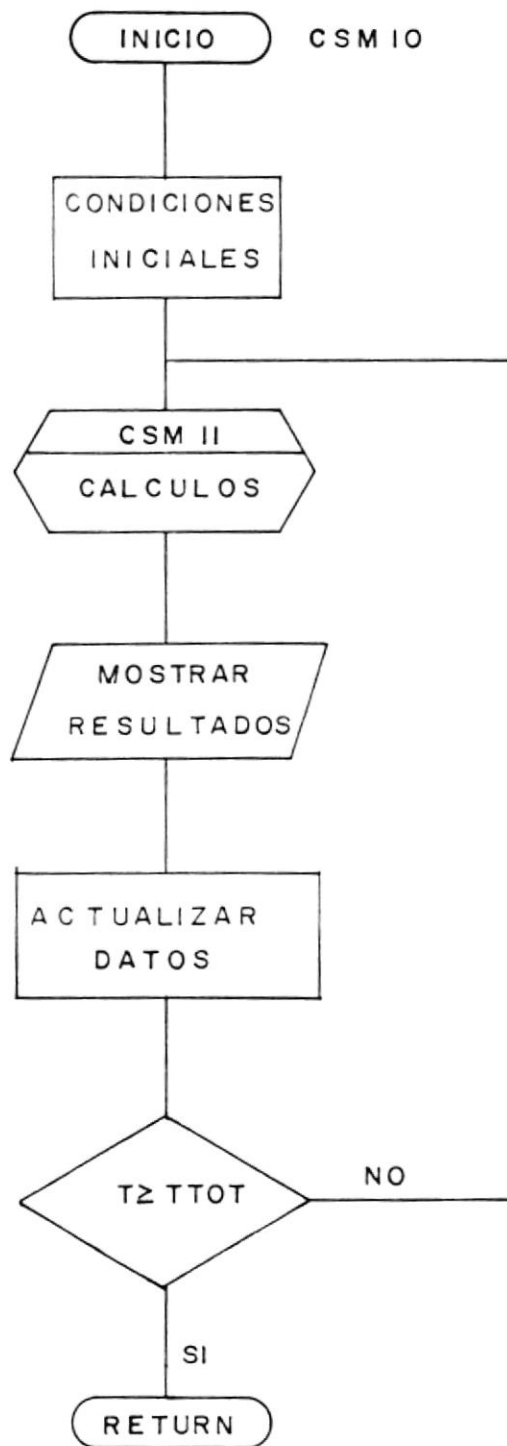


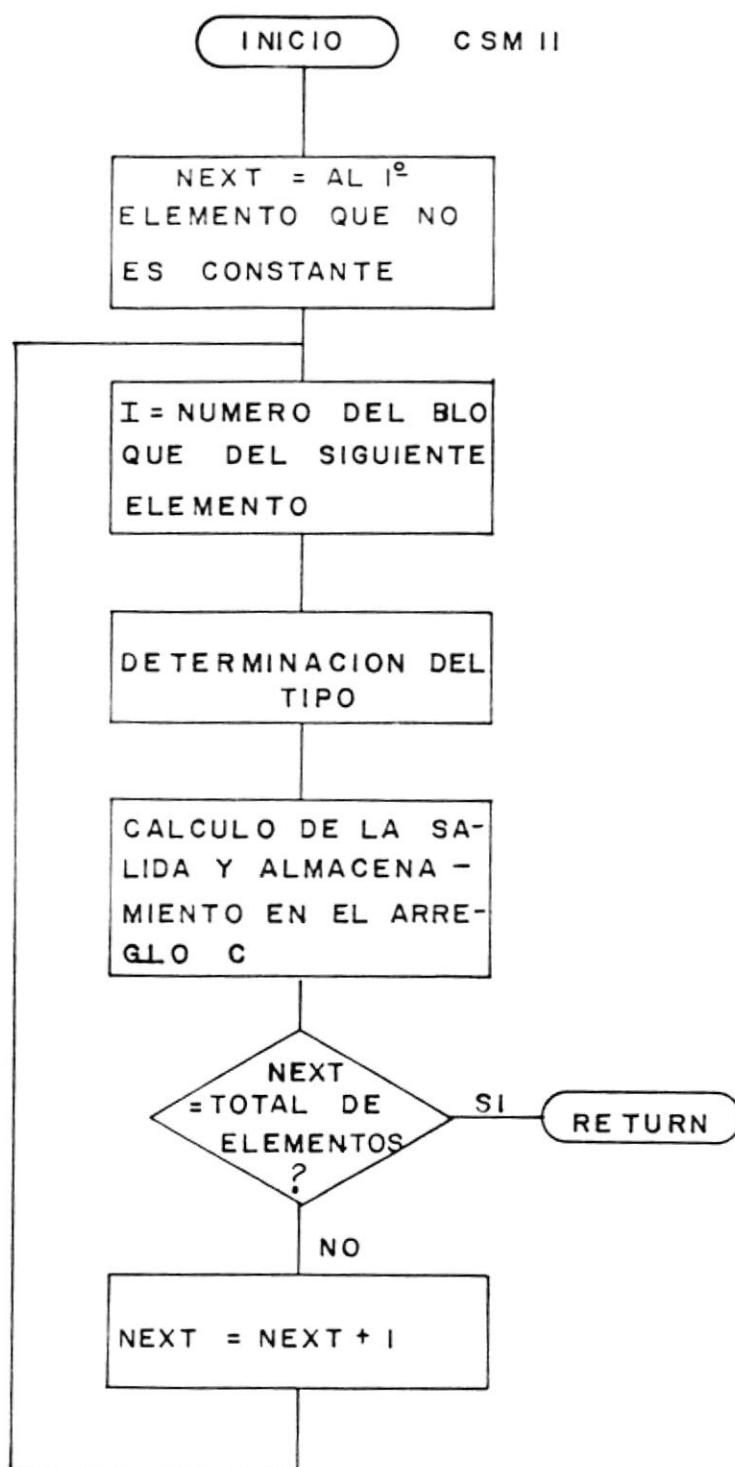












4.2 DESCRIPCION DE CADA UNA DE LAS SUBROUTINAS

CSMP.- es el programa principal. Este programa se basa en una serie de pruebas de interruptores internos que determinan la ejecución de una determinada subrutina. En este programa se coordina el llamado y corrida de cada una de las subrutinas que componen el sistema CSMP.

CSM0.- es la subrutina de inicialización de todas las variables de datos y de control que va a usar el sistema.

CSM1.- es la subrutina que permite el ingreso de los datos de configuración. Estos datos son leídos desde teclado o desde un archivo. Estos datos son chequeados para determinar si hay algún error, en cuyo caso se envía un mensaje. Si los datos están correctos, son almacenados en un arreglo llamado MTRX.

CSM2.- esta subrutina controla el número de unidades de retardo, constantes, generadores de funciones e integradores que han sido especificados, para que no se sobrepase el máximo de 75 elementos por juego de datos, de los cuales por lo menos uno, debe ser integrador, y no exceda el

máximo de 25 integradores, 25 unidades de retardo y 3 generadores de función. También, comprueba que todas las entradas requeridas por un bloque específico estén determinadas.

CSM3.- en esta subrutina se realiza el proceso de clasificación. Comprueba que cada bloque pueda ser calculado y ordena los bloques en la forma en que deben ser calculados.

CSM4.- permite el ingreso de los parámetros y condiciones iniciales asociados a cada bloque. Para cada bloque se puede asociar un máximo de tres parámetros. Para los elementos de memoria, el primer parámetro es la condición inicial. Esta subrutina chequea que los parámetros estén relacionados con un número de bloque permisible, esto es, que el número esté entre 1 y 75 y que haya sido ingresado dentro de los datos de configuración.

CSM5.- permite el ingreso de los datos para cada Generador de Funciones que se haya especificado. Solamente se pueden ingresar los datos de hasta tres Generadores de Funciones. Se comprueba, además, que el límite superior sea mayor que el inferior.

- CSM6.- permite grabar los datos ingresados desde el teclado, como un archivo en un diskette.
- CSM7.- esta subrutina permite el ingreso de la información necesaria para la realización de los cálculos propiamente dichos. Inquire sobre el intervalo de integración que va a ser usado en los cálculos de la subrutina de integración, y; también, sobre el tiempo total para el cual se van a realizar los cálculos.
- CSM10.- controla la realización de los cálculos y la presentación de los resultados. Esta subrutina llama a las subrutinas CSM11 y CSM8. A la primera, la llama para realizar la evaluación de las salidas en todos los bloques, en tanto que, a la segunda, la llama para la presentación de los resultados. Aquí, se procesa el vector derivativo Y, en lo que sería la integración propiamente dicha.
- CSM11.- contiene las características de operación de cada uno de los bloques funcionales. Calcula el valor de las salidas en todos los bloques. Para los integradores, hace los cálculos necesarios para evaluar el vector derivativo Y.

CSM8.- esta subrutina tiene dos funciones: Si la variable IARG es igual a 1, la subrutina CSM8A es llamada. Si es igual a 2, CSM8 presenta la información buscada, ya sea en la pantalla o en la impresora.

CSM8A.- esta subrutina averigua sobre el intervalo de presentación de los datos (intervalo de tiempo), y sobre los bloques, cuyas salidas se desea conocer.

CSM8B.- esta subrutina realiza los gráficos en la pantalla.

CSM12.- termina la ejecución: si ha existido error, por un elemento QUIT o porque así lo quiere el usuario.

CSM13.- averigua el valor de la salida en cualquier bloque.

SUB1..SUB5.- son subrutinas que el usuario desarrolla en FORTRAN o en ASSEMBLER, para representar relaciones funcionales de elementos tipo, que necesite en una simulación.

4.3 LISTADO Y DESCRIPCION DE LAS VARIABLES USADAS.

El programa almacena todas las variables de control y de datos en dos grandes listas:

REALS, lista de tipo real con 395 subíndices

INTS, lista de tipo entero con 587 subíndices

Estas variables son pasadas de una subrutina a otra por medio de una instrucción COMMON. Estas variables son llamados con nombres más significativos mediante el uso de la instrucción EQUIVALENCE.

VARIABLES REALES (LISTA REALS).-

Subíndice	Nombre	Nombre alternativo	Descripción
1	Blank		Constante cero, usada como C(0).
2-76	C(1-75)		Valor actual en la salida de los bloques.
77	T	C(76)	Valor de la variable tiempo.
78	DT		Intervalo de integración.
79	DTS2		Medio intervalo de integración.

80	TTOT		Tiempo total de la corrida.
81-155	Par(1-75,1)	Par1(1-75)	Primer parámetro o condición inicial de los bloques
156-230	Par(1-75,2)	Par2(1-75)	Segundo parámetro de los bloques.
231-305	Par(1-75,3)	Par3(1-75)	Tercer parámetro de los bloques.
306-340	F(1-3,1-11)		Intercepción de los generadores de funciones.
341-365	Y(1-25)		Valores del vector de integración (salida)
366-390	DYDT(1-25)		Valores del vector de integración (entrada)
391	TSAMP		Intervalo de impresión
392	VDEL		Diferencia entre el valor mínimo y el máximo del eje vertical.
393	HDEL		Diferencia entre el valor mínimo y máximo del eje

		horizontal.
394	VMIN	Valor mínimo del eje vertical
395	HMIN	Valor mínimo del eje horizontal

VARIABLES ENTERAS (LISTA INTS).-

Subíndice	Nombre	Nombre alternativo	Descripción
1-75	MTRX(1-75,1)	MTRX1(1-75)	Indicador del tipo de bloque
76-150	MTRX(1-75,2)	MTRX2(1-75)	Número del bloque de la primera entrada.
151-225	MTRX(1-75,3)	MTRX3(1-75)	Número del bloque de la segunda entrada.
226-300	MTRX(1-75,4)	MTRX4(1-75)	Número del bloque de la tercera entrada.
301-375	MTRX(1-75,5)	MTRX5(1-75)	El número de generador de función o del integrador.
376	I		Índice de la

			variable de salida
377	J		Indice de la primera entrada.
378	K		Indice de la segunda entrada.
379	L		Indice de la tercera entrada.
380-395	KEY(1-16)	KEY1...KEY16	Interruptores internos.
396-420	INTG(1-25)		Números de bloque de los integradores
421-423	NOFG(1-3)		Números de bloque de los generadores de funciones.
424-448	DELAY(1-25)		Números de bloque las unidades de retardo.
449-524	ORDER(1-76)		Orden en que se calculan los bloques.
525-533	TEST(1-9)	TEST1..TEST9	Interruptores internos.
534	NLIST		Número total de bloques.
535-539	PRINT(1-5)	K1...K5	Número de los bloques cuyas salidas se van a

		imprimir.
540	NCON	Número de elementos constantes.
541	NOD	Número de elementos unidades de retardo
542	NEQ	Número de elementos integradores.
543	NFG	Número de elementos generadores de función
546-585	TYPE(1-40)	El equivalente numérico de los tipos de bloque.
586	IR	Número interno de inicialización del generador de números aleatorios
587	IARG	Argumentos para las subrutinas de presentación de los resultados.

APENDICE

MANUAL DEL USUARIO

INTRODUCCION

El CSMP es un programa hecho en FORTRAN que permite realizar la simulación de sistemas continuos.

Esta diseñado para que se pueda modelar un sistema y se lo pruebe en línea en un microcomputador IBM o compatible.

Este programa emplea un lenguaje de bloques como entrada, lo que lo hace muy accesible dada la difusión que tiene este tipo de lenguaje. Estos bloques representan los elementos y organización de un computador analógico.

ELEMENTOS QUE PUEDE USAR EL PROGRAMA CON SU DESCRIPCION

El CSMP provee un total de 25 elementos standar para la simulación. Esto incluye los elementos funcionales comunes, como Integradores, Multiplicadores, Sumadores y Constantes, y elementos más especializados como Separadores, Generador de Funciones, Generador de Números

Aleatorios, etc. A más de estos elementos, el usuario puede definir 5 elementos más, desarrollando subrutinas en FORTRAN o ASSEMBLER para representar la relación funcional del mismo.

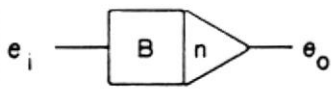
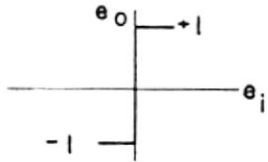
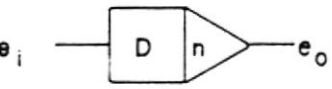
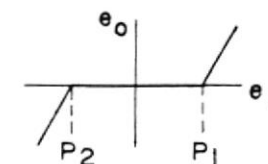
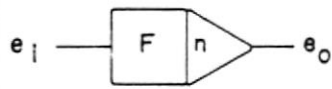
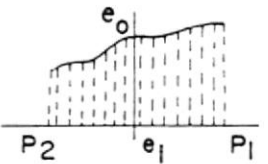
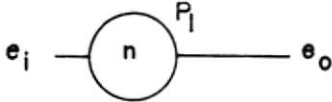
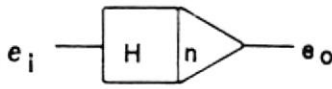
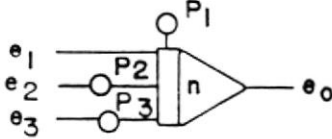
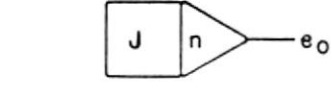
Cada uno de los elementos tipo tiene un máximo de 3 entradas y 3 parámetros y su salida es un escalar definido por la relación funcional del elemento. Estos elementos son identificados con un símbolo gráfico y un símbolo en el lenguaje. La tabla Tipos de Elementos para Simulación nos muestra todos los elementos tipo del CSMP.

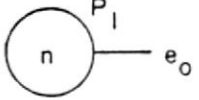
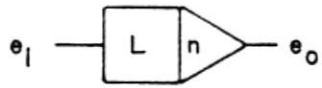
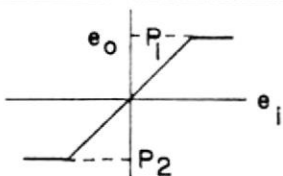
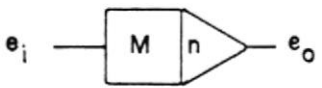
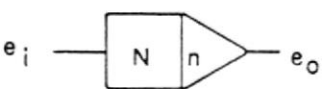
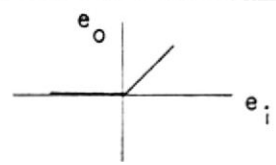
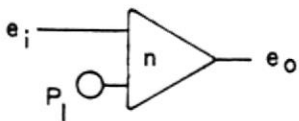
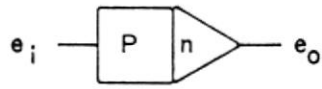
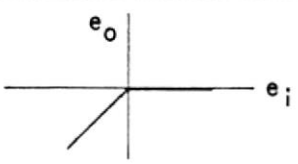
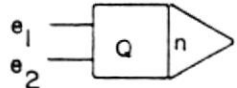
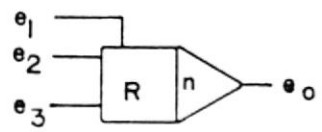
Algunos elementos no tienen parámetros ni condiciones iniciales, por lo que necesitarán solamente definir las especificaciones de configuración, por ejemplo: el Multiplicador, el Bang-Bang, el Valor Absoluto y el Relé.

Otros elementos en cambio, requieren que se especifique las condiciones iniciales y los parámetros. En este grupo se encuentran el Integrador, el Sumador con Peso, el Limitador, entre otros.

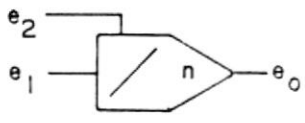
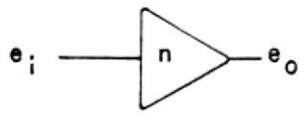

Finalmente, el Generador de Función requiere de los tres tipos de datos: especificaciones de configuración para indicar su interconexión; los parámetros sirven para especificar sus puntos terminales; y las especificaciones del generador de funciones sirven para indicar el valor

TIPOS DE ELEMENTOS PARA SIMULACION

TIPO ELEMENTO	SIMBOLO LENGUAJE	SIMBOLO DIAGRAMA	DESCRIPCION
BANG BANG	B		
ESPACIO MUERTO	D		
GENERADOR DE FUNCION	F		
AMPLIFICADOR	G		$e_o = P_1 e_i$
RAIZ CUADRADA	H		$e_o = \sqrt{e_i}$ RAIZ CUADRADA
INTEGRADOR	I		$e_o = P_1 \int (e_1 + e_2 P_2 + e_3 P_3) dt$
ALEATORIO	J		GENERADOR DE NUMEROS AL AZAR ENTRE ± 1

TIPO ELEMENTO	SIMBOLO LENGUAJE	SIMBOLO DIAGRAMA	DESCRIPCION
CONSTANTE	K		$e_o = P_1$
LIMITADOR	L		
MAGNITUD	M		$e_o = \text{VALOR ABSOLUTO}$ (e_i)
SEPARADOR NEGATIVO	N		
OFFSET	O		$e_o = e_i + P_1$
SEPARADOR POSITIVO	P		
FIN	Q		QUIT (CORRIDA TERMINADA) SI $e_1 > e_2$
RELE	R		$e_o = e_2$ PARA $e_1 \geq 0$ $e_o = e_3$ PARA $e_1 < 0$

TIPO ELEMENTO	SIMBOLO LENGUAJE	SIMBOLO DIAGRAMA	DESCRIPCION
GENERADOR DE PULSOS DE TIEMPO	T		GENERA UN TREN DE PULSOS CON PERIODO IGUAL A P_T COMIENZA CUANDO $e_i \neq 0$
UNIDAD DE RETARDO	U		$e_o = e_i(t - \Delta t/2)$
VACIO	V		$e \neq f(e_i)$ USADO EN CONJUNCION CON EL ELEMENTO WYE
SUMADOR CON PESO	W		$e_o = P_1 e_1 + P_2 e_2 + P_3 e_3$
MULTIPLICADOR	X		$e_o = e_1 e_2$
BIFURCACION	Y		ELEMENTO LOGICO DE BIFURCACION USADO PARA OPERACIONES IMPLICITAS
MANTENER ORDEN CERO	Z		$e_o = e_1$ PARA $e_2 > 0$ e_o NO CAMBIA PARA $e_2 \leq 0$
SUMADOR	+		$e_o = e_1 + e_2 + e_3$

TIPO ELEMENTO	SIMBOLO LENGUAJE	SIMBOLO DIAGRAMA	DESCRIPCION
DIVISOR	/		$e_0 = e_1 / e_2$
INVERSOR DE SIGNO	-		$e_0 = -e_i$
ESPECIAL	1-5		SUBROUTINAS SUMINISTRADAS POR EL USUARIO

n REPRESENTA EL NUMERO DE BLOQUE

e_i SEÑAL DE ENTRADA

e_o SEÑAL DE SALIDA

P_i PARAMETROS

de la variable de salida en cada punto de intercepción.

ENTRADA DE DATOS

Para poder simular un sistema en el CSMP, se debe preparar la información que se va a ingresar al computador; esto es, realizar los diagramas de bloques correspondientes del sistema y asignar a cada bloque un número arbitrario entre 1 y 75 y establecer el tipo del bloque que se trata. El número de bloque 76 es la variable tiempo.

Al preparar los datos se tendrá presente que los elementos que tienen una sola entrada, usan la entrada 1; los que tienen dos entradas usan las entradas 1 y 2; y, aquellos que tienen tres entradas usan las entradas 1,2 y 3. Si un elemento de múltiples entradas, sólo va a usar una entrada, ésta debe asignarse a la entrada 1, esto significa que no se puede utilizar una entrada si la anterior no ha sido usada. De esta regla, se exceptúan los Integradores, los Sumadores y los Sumadores con Peso. Los Sumadores (tipo +) son los únicos que admiten como entrada, número de bloques negativos.

Como se habría expresado antes, un elemento puede tener asignado hasta tres parámetros o no tener ninguno. Por ejemplo, el Sumador con Peso tiene asociado tres

parámetros, mientras que, el Sumador no tiene ninguno.

Si un elemento usa parámetros y algunos de ellos no son especificados, el programa los asume como cero.

Para elementos tipo que tienen memoria, el Integrador, la Unidad de retardo, etc., el primer parámetro se usa para especificar la condición inicial.

El elemento Generador de Funciones es usado para simular una relación funcional $y=f(x)$ que se encuentre en forma tabular o gráfica. Esta relación debe ser tal que pueda ser aproximada por 10 segmentos de línea espaciados igualmente en X sobre el rango de interés. Los valores máximos y mínimos de X deben ser especificados dentro de los parámetros del bloque y los puntos terminales de cada uno de los 10 segmentos, en las especificaciones del Generador de Funciones. La salida del Generador de Funciones es calculada usando interpolación lineal entre los 10 valores especificados.

Existen dos elementos, WYE y Vacío, que se usan en conjunto para simular relaciones implícitas del tipo $y=f(y,x)$. Este tipo de relaciones es calculada en base a un proceso iterativo en el cual un estimado inicial del valor de Y, es usado para una estimación mejorada de Y. El proceso se repite hasta que dos valores de Y sean

iguales. Como X cambia durante el proceso de simulación, el proceso iterativo es realizado en cada intervalo de integración numérica.

El valor estimado inicial de Y es ingresado como la condición inicial del elemento Vacío, después, en este elemento se irán almacenando los valores sucesivos de Y.

El elemento WYE sirve para determinar cuando dos valores sucesivos de Y cumplen con la condición de error máximo permitido, si no se cumple, el valor estimado último se almacena como salida del elemento Vacío. Cuando el valor de Y converge, este valor se presenta como la salida de WYE. El primer parámetro de WYE indica el error permitido y el parámetro 2 es un factor que es aplicado a la diferencia entre dos estimados sucesivos para obtener una corrección apropiada, este factor tiene un valor entre cero y la unidad. Se aconseja un valor de cero si la función implícita converge rápidamente.

Si la función comienza a oscilar se recomienda un valor de 0.5. Para incrementar la eficiencia del proceso iterativo, el número de bloque del elemento Vacío debe ser el más alto asignado.

Es preferible ingresar la información por medio de un archivo creado con un editor de texto. Si se decide por

esta opción, se podrá usar el EDLIN del sistema operativo MS-DOS. No es recomendable usar el Personal Editor (PE), pues graba ciertos caracteres de control que distorsionan los campos cuando los lee el programa.

Al crear el archivo se deben ingresar los datos de las especificaciones de configuración, una línea por bloque y al final una línea en blanco.

Si se ingresan los datos desde teclado, aparecerá una pantalla en la que habrá un encabezado que dirá ESPECIFICACION DE CONFIGURACION. Se debe ingresar una línea de datos para cada bloque y al final una línea en blanco.

La línea de datos de la especificación de configuración está estructurada de la siguiente forma:

Columnas 2-17	Nombre de la salida
Columnas 20-21	Nombre del bloque
Columna 31	Tipo del bloque
Columnas 41-43	Número de bloque de entrada 1
Columnas 53-55	Número de bloque de entrada 2
Columnas 65-67	Número de bloque de entrada 3

El nombre de la salida es opcional, sirve para identificar la salida. Puede dejarse en blanco pues no

afecta los cálculos.

Los datos numéricos deben ajustarse a la derecha.

Luego de ingresar las especificaciones de configuración, se deben ingresar los parámetros y condiciones iniciales. Si se ingresan los datos desde archivo, luego de la línea en blanco dejada anteriormente, se ingresan las líneas de datos de parámetros y condiciones iniciales de aquellos bloques que lo necesiten. Si se ingresan desde el teclado, se lo hará en el momento en que aparezca una pantalla con el encabezado CONDICIONES INICIALES Y PARAMETROS. La línea de datos está configurada de la siguiente forma:

COLUMNAS 2-17	Nombre del parámetro
COLUMNAS 20-21	Nombre del bloque
COLUMNAS 27-36	Parámetro 1
COLUMNAS 42-51	Parámetro 2
COLUMNAS 57-66	Parámetro 3

El nombre del parámetro es opcional y puede ser dejado en blanco. Los parámetros deben ser ingresados como cantidades reales con un punto decimal y de cero a siete decimales. También, las variables reales pueden ser escritas en notación exponencial usando el formato D.

Luego de ingresar todos los datos de parámetros para los bloques, se deja una línea en blanco.

Si se han especificado generadores de funciones se deben definir los puntos de intercepción. Por cada bloque generador de función son usadas 3 líneas, cuya estructura es la siguiente:

1a. línea	columnas 2-17	Nombre del Generador
1a. línea	columnas 20-21	Número del bloque
1a. línea	columnas 24-33	Intercepción para X=Parámetro 2
1a. línea	columnas 39-48	2da. intercepción
1a. línea	columnas 54-63	3ra. intercepción
1a. línea	columnas 69-78	4ta. intercepción
2a. línea	columnas 24-33	5ta. intercepción
2a. línea	columnas 39-48	6ta. intercepción
2a. línea	columnas 54-63	7ma. intercepción
2a. línea	columnas 69-78	8va. intercepción
3a. línea	columnas 24-33	9na. intercepción
3a. línea	columnas 39-48	10ma. intercepción
3a. línea	columnas 54-63	11va. intercepción

Al final se dejará una línea en blanco.

A continuación, se mostrará lo que aparece en la pantalla cuando se ingresan los datos desde teclado.

SIMULACION DE SISTEMAS CONTINUOS
Por Ruth Santana de Garcia

=====

ESPECIFICACION DE CONFIGURACION

SALIDA	BLOQUE	TIPO	ENTRADA 1	ENTRADA 2	ENTRADA 3
XXXXXXXXXXXXXXXXXX	99	t	999	999	999

SIMULACION DE SISTEMAS CONTINUOS
Por Ruth Santana de Garcia

=====

CONDICIONES INICIALES Y PARAMETROS

CI/PAR NOMBRE	BLOQUE	CI/PAR1	PAR2	PAR3
XXXXXXXXXXXXXXXXXX	99	9999999999.9999	9999999999.9999	9999999999.9999

SIMULACION DE SISTEMAS CONTINUOS
Por Ruth Santana de Garcia

=====

CONDICIONES INICIALES Y PARAMETROS
ESPECIFICACIONES DEL GENERADOR DE FUNCION

NOMBRE	BLOQUE	INTERCEPCIONES			
XXXXXXXXXXXXXXXXXX	99	99999.9999	99999.9999	99999.9999	99999.9999
		99999.9999	99999.9999	99999.9999	99999.9999
		99999.9999	99999.9999	99999.9999	99999.9999

En las líneas siguientes a la presentación de cada pantalla, se deben ingresar las líneas de datos, una por cada bloque definido. Se debe guiar, al ingresar los datos por los caracteres presentados en las pantallas correspondientes.

EJECUCION DEL PROGRAMA

Para poder ejecutar el programa CSMP, se debe digitar desde el drive donde se encuentra el CSMP, la palabra

CSMP y presionar la tecla ENTER.

Si su computador tiene tarjeta de gráficos, antes de cargar el CSMP, debe digitar GRAPHICS y presionar la tecla ENTER desde el drive donde esté el sistema operativo.

Cuando se ejecuta el CSMP, lo primero que pregunta es sobre el dispositivo de entrada de datos.

Si selecciona desde el disco, le pregunta por el nombre del archivo. Si es por teclado, le presenta la pantalla de ESPECIFICACION DE CONFIGURACION y le permite el ingreso de datos de configuración.

Luego, pregunta si desea modificar estos datos. Si desea modificar algún dato, lo puede hacer volviendo a digitar toda la línea del bloque específico. (Esto tanto para ingreso desde disco o desde teclado).

El programa realiza la clasificación de los bloques. Si hay algún error, le presenta el mensaje respectivo y le permite cambiar la línea de datos del bloque donde se produjo el error.

Si la clasificación es exitosa, lee los parámetros y condiciones iniciales desde el dispositivo de entrada

seleccionado al inicio de la corrida.

Luego, de leer todos los datos pregunta si desea modificar alguno. Si lo hace, bastará con modificar todas las líneas de los datos del bloque respectivo.

Si hay error en los datos ingresados, le enviará el mensaje correspondiente y le permitirá cambiar la línea de datos del bloque donde se produjo el error.

Si se han definido generadores de funciones, el programa lee las especificaciones de los mismos. Permite modificar estas especificaciones desde teclado para lo cual pregunta si las quiere modificar, si contesta afirmativamente le permitirá ingresar los nuevos datos.

Si hay error, envía el mensaje respectivo y permite cambiar los valores.

Una vez que el programa ha verificado la validez de todos los datos, le pregunta si los quiere grabar en disco, si responde afirmativamente, le pregunta el nombre del archivo.

A continuación, el programa pregunta sobre datos específicos para el cálculo de las salidas de los bloques.

Primero, pregunta sobre el intervalo de integración que puede tener hasta 3 decimales. Luego, el tiempo total de corrida del programa. El cálculo de las salidas en los bloques se realiza siempre desde un valor de tiempo igual a cero hasta este valor de tiempo total.

El programa inquiriere sobre el tipo de salida que desea. Existen tres tipos de salida: DATOS POR PANTALLA, GRAFICOS EN PANTALLA, SALIDA POR IMPRESORA. La primera es una presentación de los valores numéricos de las salidas de cuatro bloques del sistema simulado. La segunda es la graficación de la salida de un solo bloque y puede ser impreso. La tercera es una presentación de resultados numéricos de la salida en cuatro bloques del sistema y, además, un gráfico de una de las salidas.

Si selecciona por impresora, debe asegurarse que la impresora esté encendida y con suficiente papel. Para seleccionar gráficos por pantalla, su equipo debe tener una tarjeta adaptador de monitor para gráficos.

Luego, debe ingresar el incremento de tiempo, es decir, el paso con que se va a ir presentando la salida de la información.

Para el caso de gráficos por pantalla o por impresora, averigua el número de bloque de la variable a ser

graficada y su valor máximo y mínimo.

Para el caso de datos por pantalla y resultados por impresora, pregunta por los números de bloques de tres variables más, para presentar los valores de sus salidas.

Si pidió sacar resultados gráficos por pantalla, puede obtener ese mismo gráfico en la impresora, si al finalizar la elaboración del gráfico, presiona simultáneamente las teclas SHIFT y PRT-SC.

Como las opciones de salida por impresora y gráficos por pantalla, le piden los valores máximos y mínimos de la variable a ser graficada (Eje-Y), es conveniente que antes de pedir estas opciones, se seleccione la opción de datos por pantalla para conocer los valores que toma una variable específica (la que se quiere graficar).

Al finalizar la presentación de los resultados, se debe presionar la tecla ENTER.

El programa averigua si desea conocer la salida en algún bloque específico y finalmente, si termina la corrida.

Si no quiere terminar la corrida, puede seguir corriendo el programa con el juego de datos ingresados al inicio.

El CSMP tiene cinco elementos tipos, a saber, 1, 2, 3, 4 y 5, que pueden ser definidos por el usuario. Estos elementos corresponden a las subrutinas SUB1, SUB2, SUB3, SUB4 y SUB5. Estas subrutinas son subprogramas que sirven para generar una relación funcional especial: funciones trigonométricas, logarítmicas, exponenciales, etc.. Pueden ser escritas en FORTRAN o en ASSEMBLER. En el CSMP se ha dejado creadas estas subrutinas como subprogramas vacíos y se encuentran en el módulo CSMSUB.FOR. Cuando se realice una subrutina de este tipo, debe compilarse y luego, unirse con los demás módulos, se debe usar la biblioteca llamada UTIL.LIB. Los módulos objetos que se deben ser juntados son:

CSMP
CSM0
CSM1
CSM2
CSM3
CSM4
CSM5
CSM6
CSM7
CSM8
CSM8A
CSM8B
CSM10

CSM7
CSM8
CSM8A
CSM8B
CSM10
CSM11
CSM12
CSM13
CSMUTIL
CSMSUB

Este último puede dividirse en cinco módulos: SUB1, SUB2, SUB3, SUB4, SUB5

La biblioteca UTIL es una biblioteca que fue creada con todos los módulos realizados en ASSEMBLER, que son en su mayoría, subrutinas para manejo de pantallas y gráficos.

EJEMPLO PRACTICO

PLANTEAMIENTO DEL PROBLEMA

La figura 1, representa el modelo de una planta hidroeléctrica. Esto comprende: El regulador de velocidad, el efecto de la caída del agua, la apertura de la válvula esférica y la turbina hidráulica. Se trata de realizar la simulación de los componentes de la

planta. Se va a presentar la simulación del regulador de velocidad incluido el efecto del control de velocidad sobre la aceleración de la unidad y del agua, y el regulador de voltaje o excitatriz.

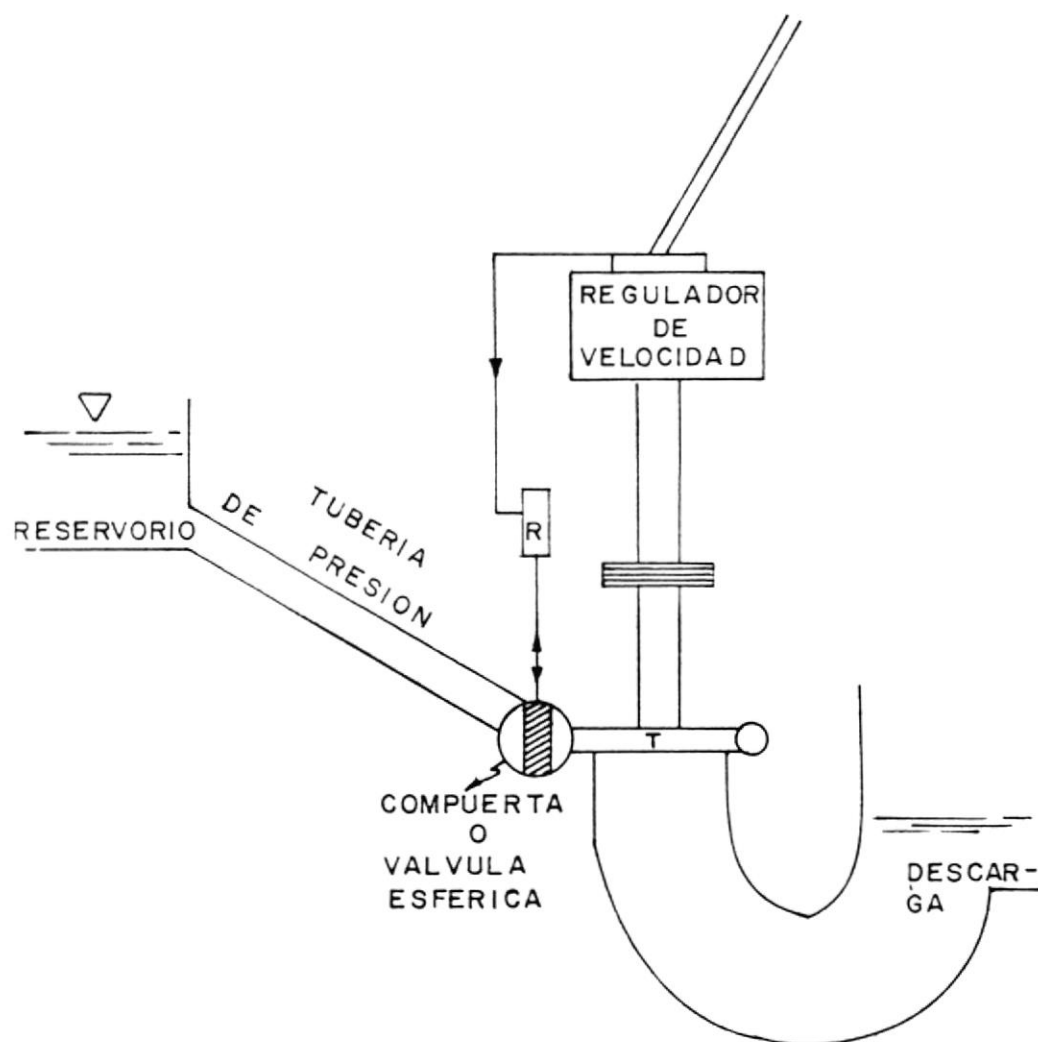
DEFINICION DE LAS ECUACIONES

En la Figura 2, podemos observar un modelo del Regulador de Velocidad.

El regulador de velocidad recibe la señal del eje de la turbina n_s y la compara con una señal de referencia n_r . Si esta señal es grande comparada con el efecto amortiguador (ζ), se origina un movimiento de palancas, las cuales mueven los pistones de los servo y válvulas pilotos, que a través de la válvula distribuidora desplazan la puerta del servomotor (g), abriendo o cerrando los inyectores, permitiendo el flujo del agua a la tubería.

El regulador de velocidad es mecánico y con caída de velocidad, esto último significa que para un cambio de velocidad existe un cambio en la apertura del servomotor. El efecto de caída de velocidad es dado por la realimentación de las palancas.

El efecto de control de velocidad viene dado por:



T = TUBERIA

FIG # 1

MODELO DE LA PLANTA

$$T_m - T_e = Mg \frac{d\bar{\delta}}{dt}$$

$$q = \frac{dq}{dh} h + \frac{dq}{d\bar{\delta}} \bar{\delta} + \frac{dq}{dg} g$$

$$T_m = \frac{dT_m}{dh} h + \frac{dT_m}{d\bar{\delta}} \bar{\delta} + \frac{dT_m}{dg} g$$

$$h = -T_w \frac{dq}{dt}$$

Para el Regulador de Voltaje se han simulado tres modelos que fueron establecidos por la IEEE y publicados en Agosto de 1969, en el IEEE Transaction vol. PAS-88.

Estos tres modelos son: Excitatriz D.C., Excitatriz A.C. y Excitatriz de Tiristores. Estos tres modelos están representados en las figuras 3, 4 y 5, respectivamente.

Todas las ecuaciones utilizadas en la resolución de este problema están desarrolladas en el Capítulo II de la Tesis para la Obtención del Título de Ingeniero Eléctrico del Sr. Miguel Vacas Quincha, titulada "Análisis de la Sincronización de un Generador Síncrono".

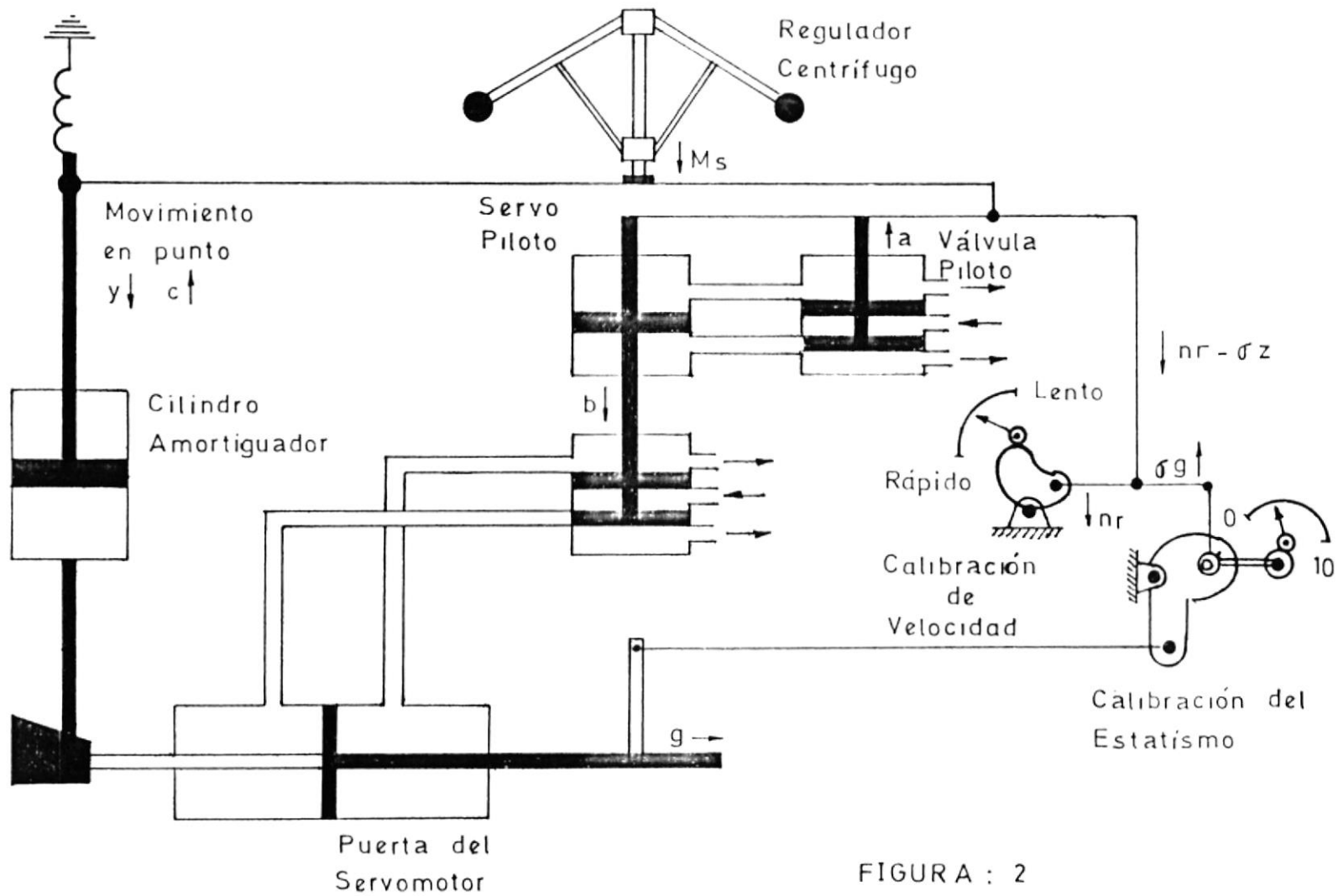


FIGURA : 2
REGULADOR DE VELOCIDAD

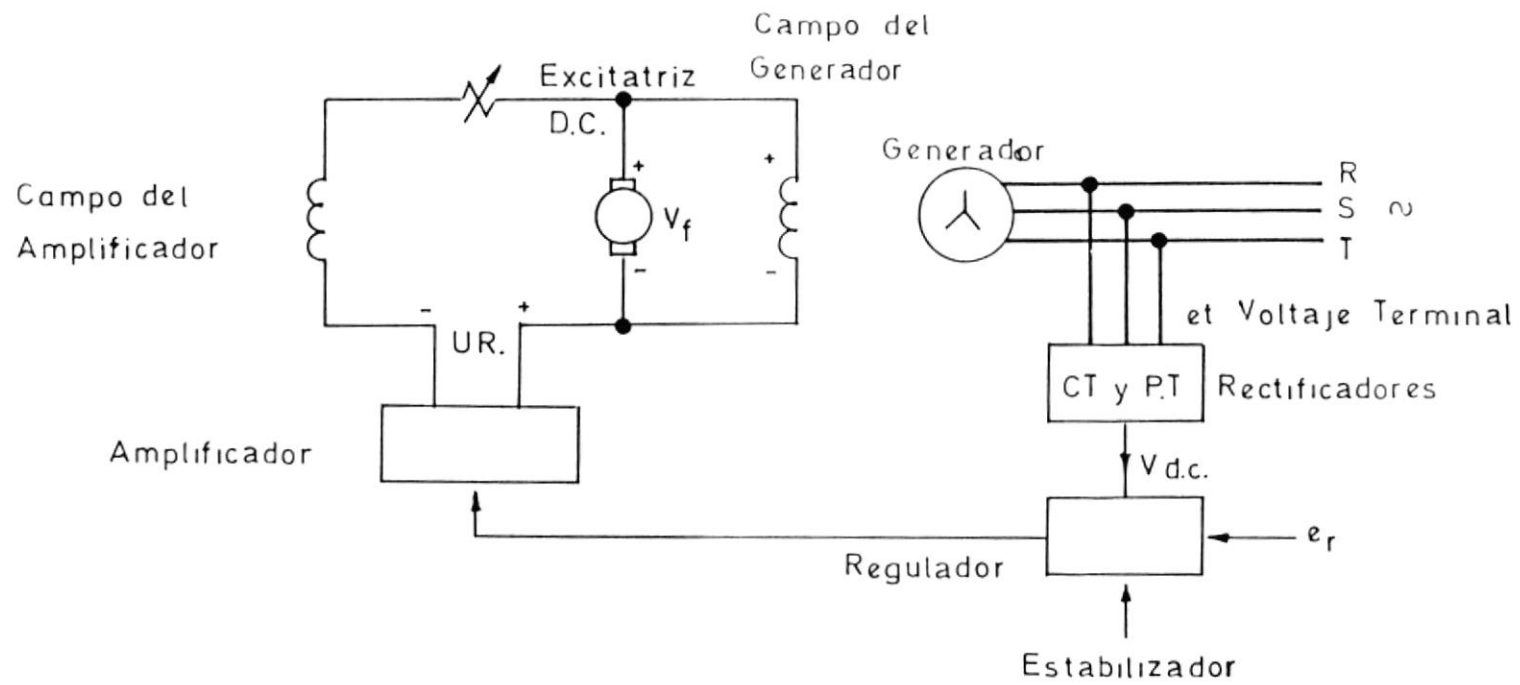


FIGURA : 3
EXCITATRIZ D.C.

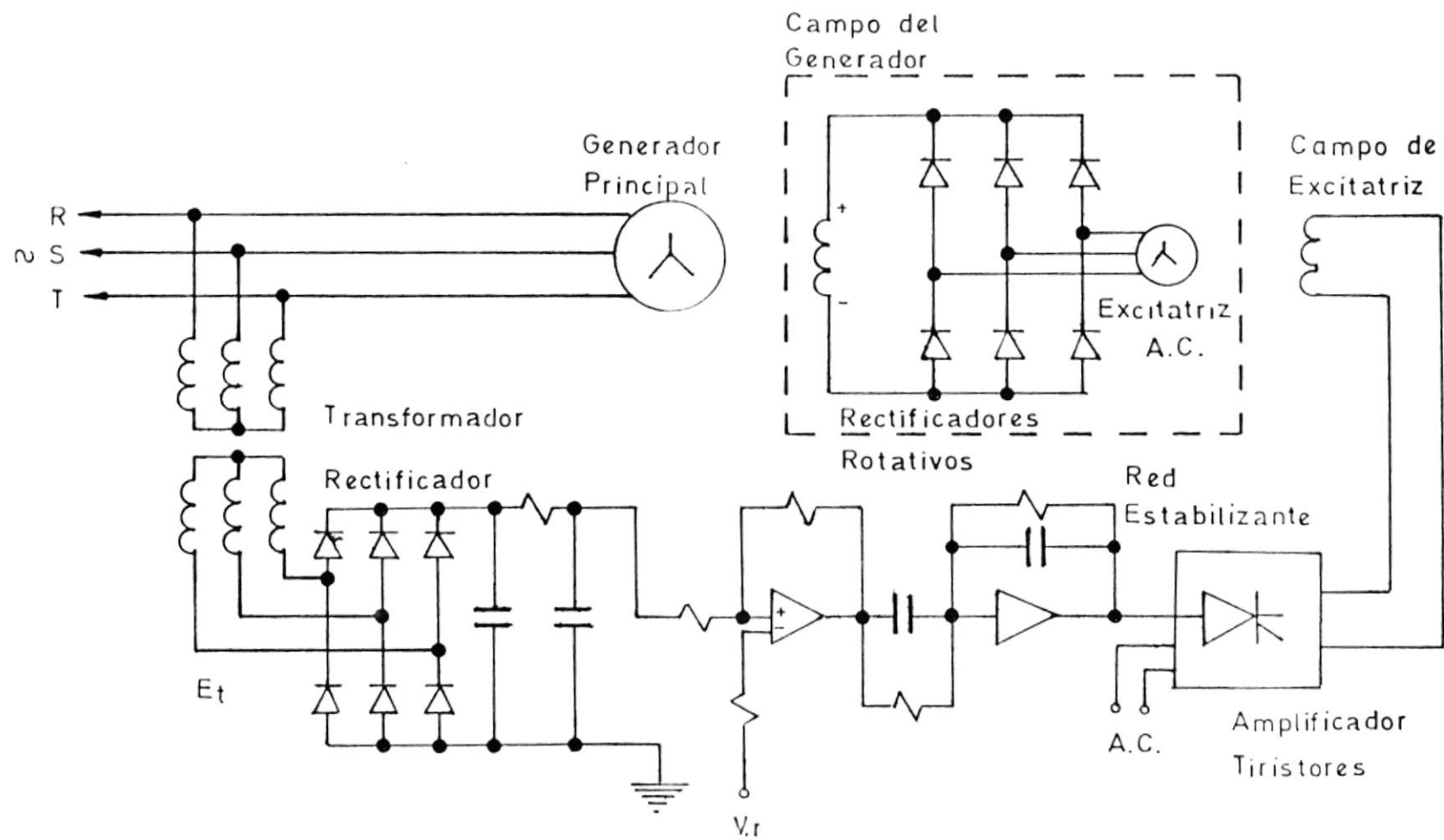


FIGURA : 4
EXCITATRIZ A.C. ROTATIVA

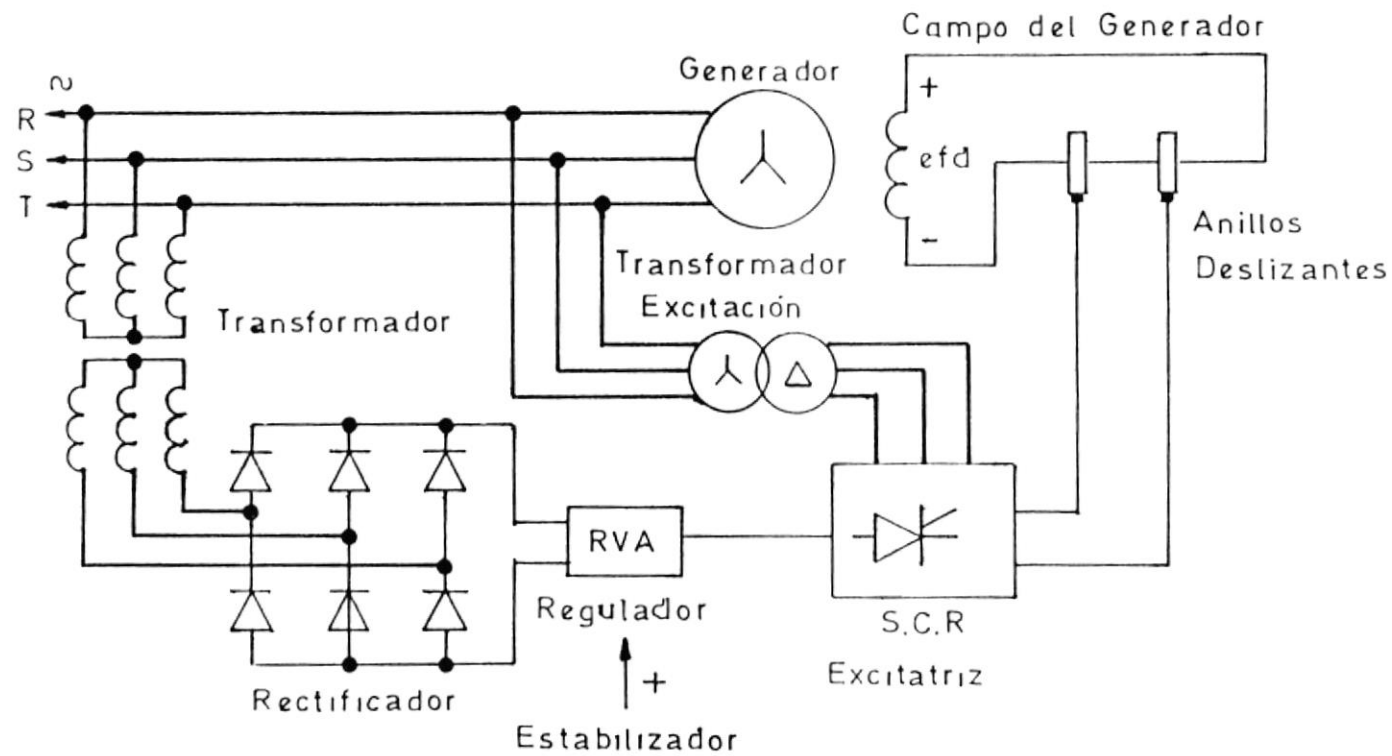


FIGURA : 5
EXCITATRIZ TIRISTORES

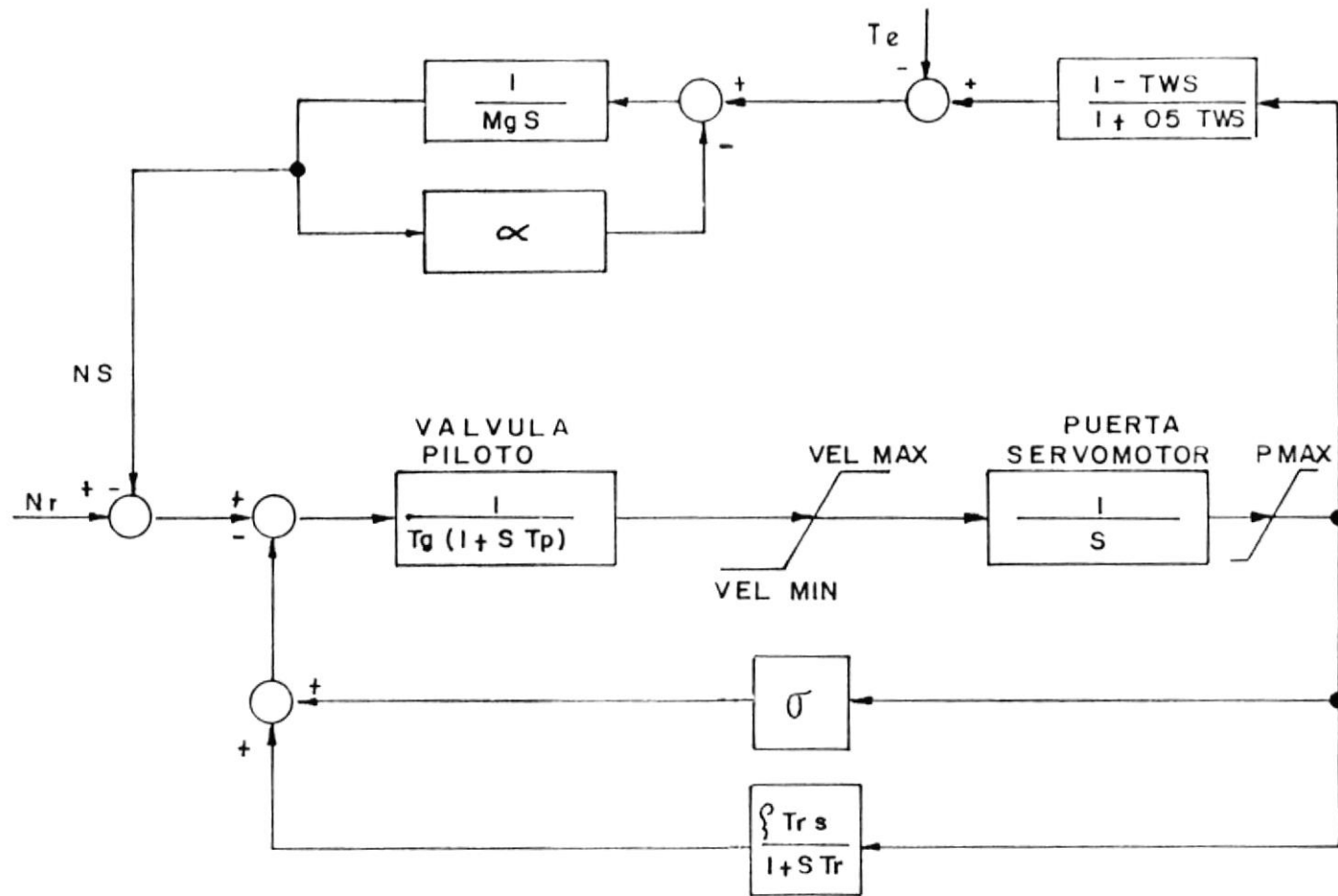


FIG # 2 a
 DIAGRAMA DE BLOQUES DEL REGULADOR DE VELOCIDAD

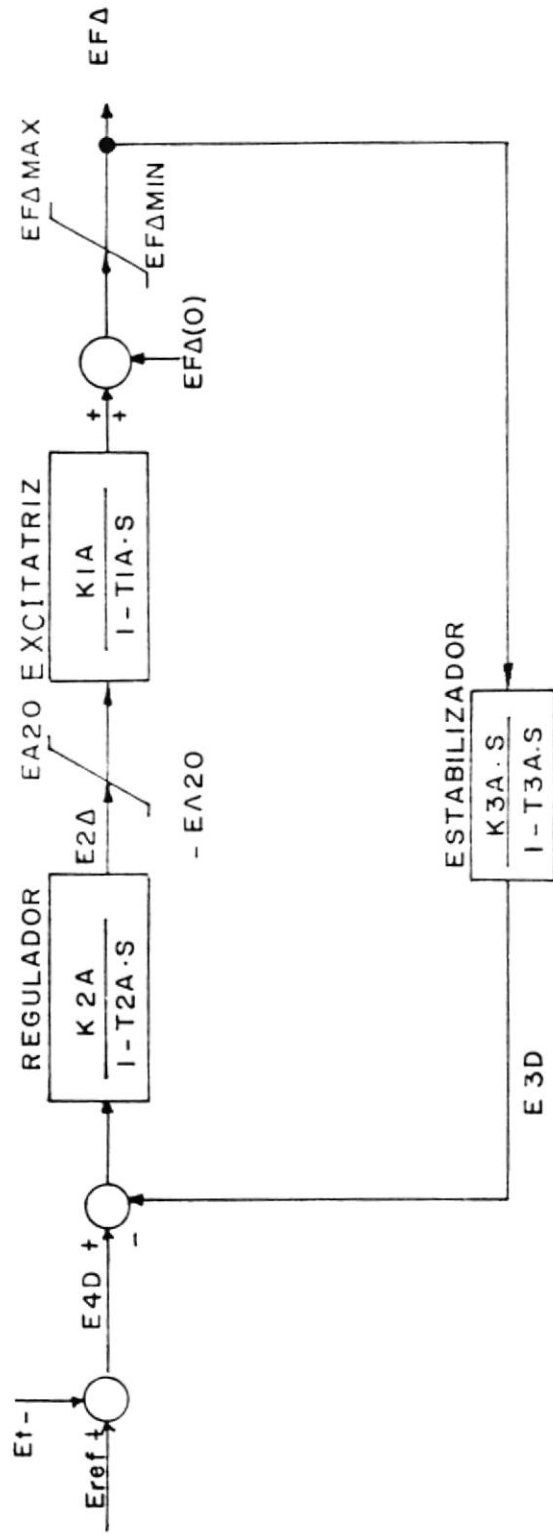


FIG : 30

EXCITATRIZ D.C.

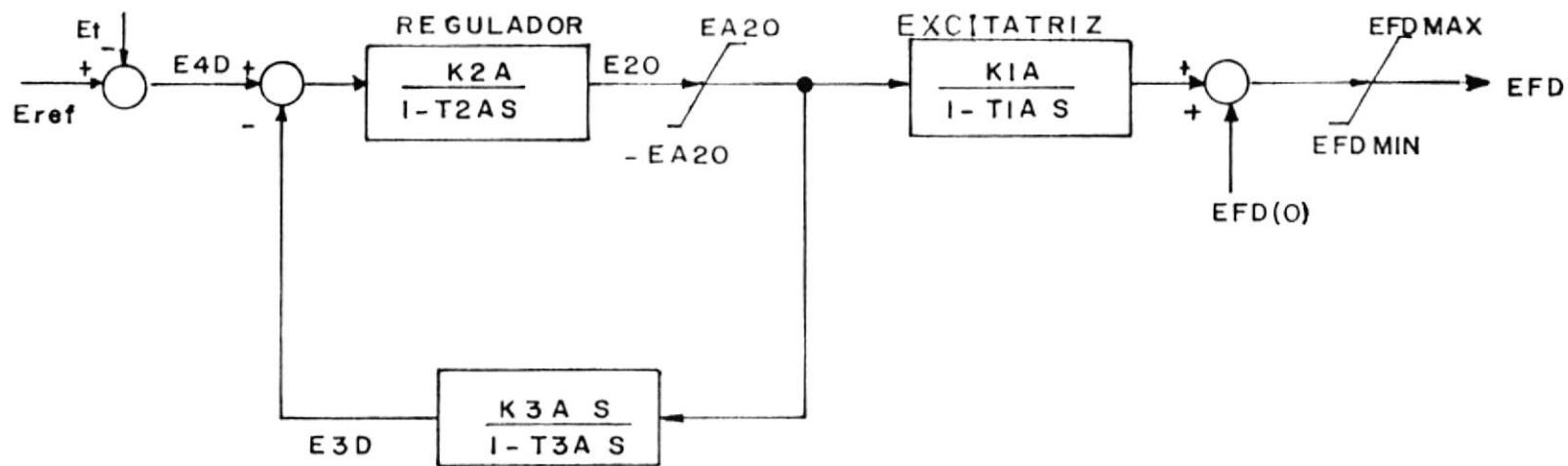
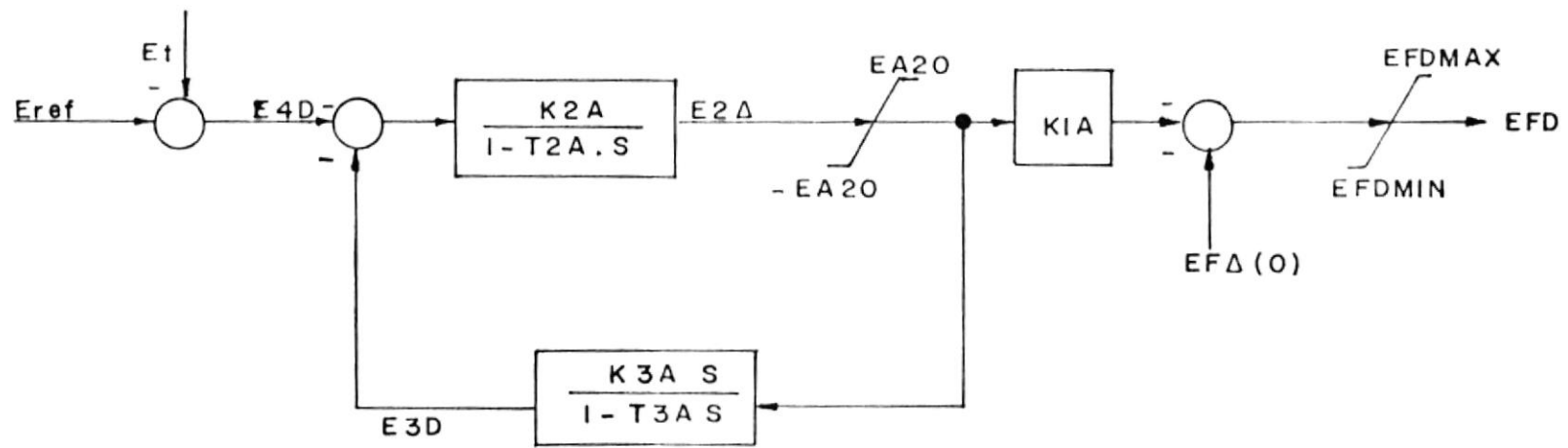


FIG : 4 a

EXCITATRIZ A.C. - ROTATIVA



T 1 A	T 2 A	T 3 A	K 1 A	K 2 A	K 3 A	E F Δ M A X	E F Δ M I N	E A 2 0
2.0 seg	0.1 seg	0.5 seg	80	1.0	0.1	4.0	0.0	100.0

FIG 5a

EXCITATRIZ TIRISTORES

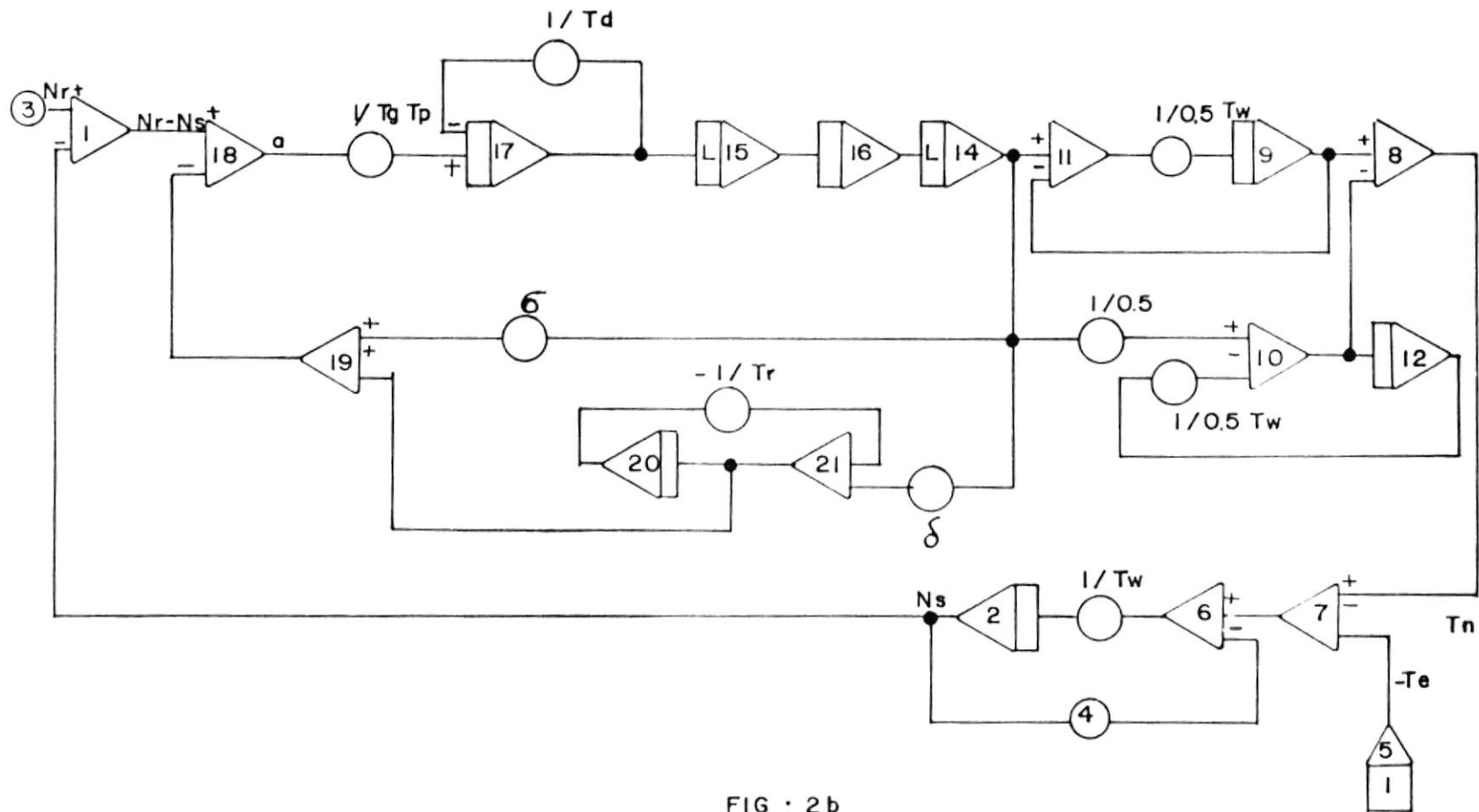


FIG : 2b
REGULADOR DE VELOCIDAD

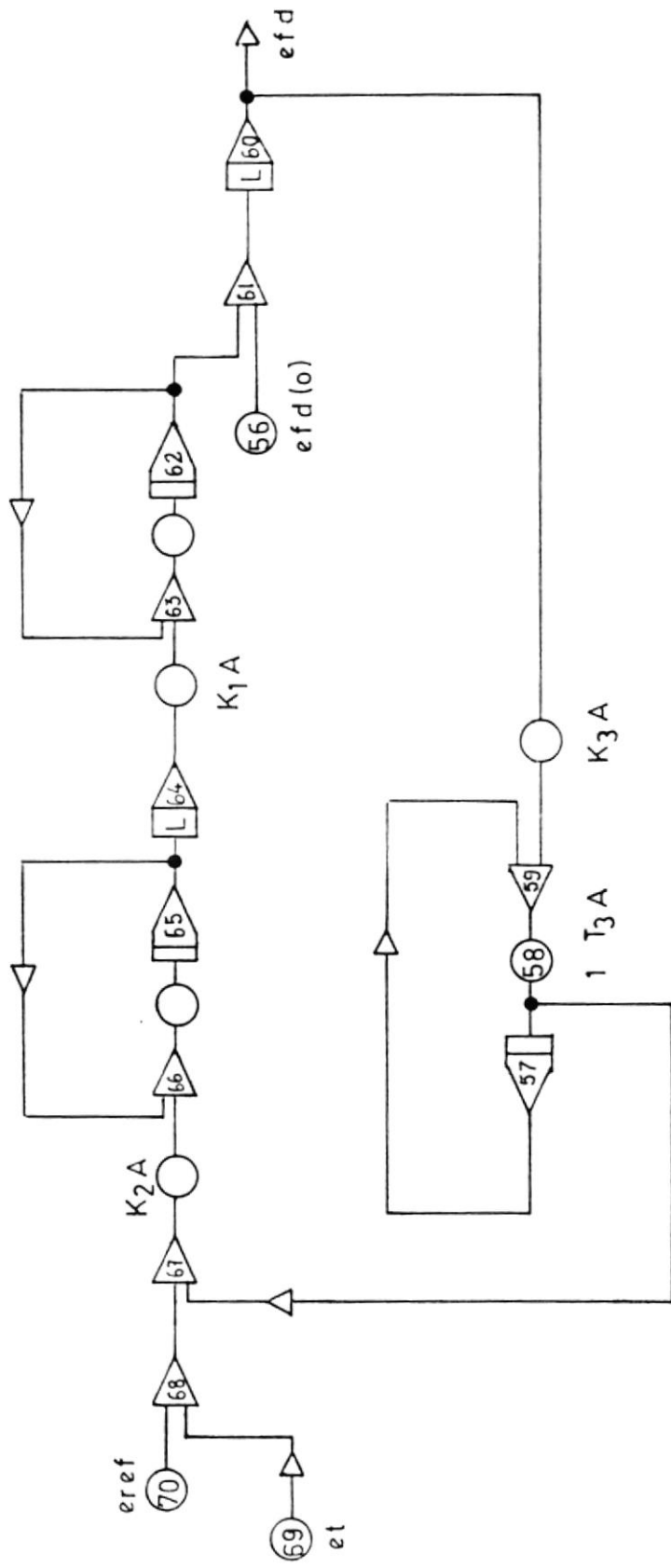


FIGURA : 3 b
SIMULACION DE LA EXCITATRIZ D.C.

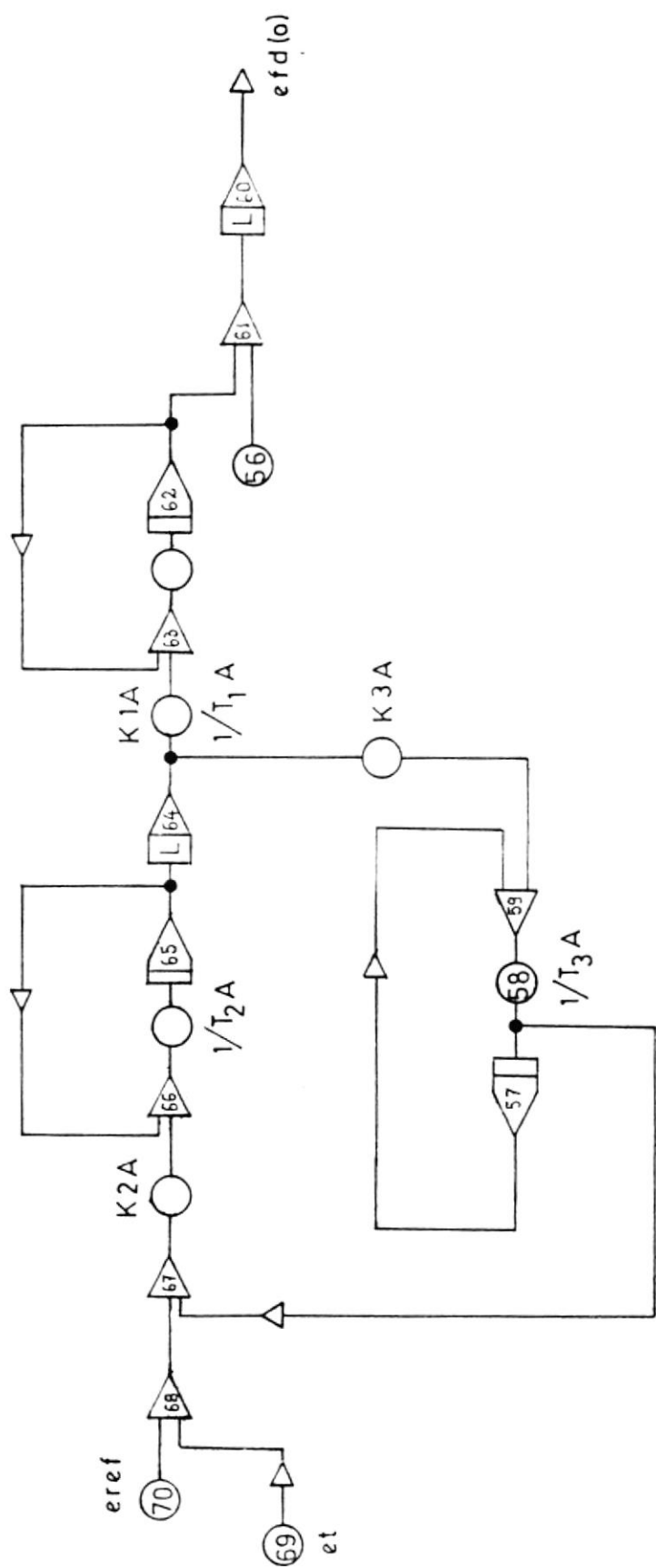


FIGURA : 4.b

SIMULACION DE LA EXCITATRIZ AC ROTATIVA

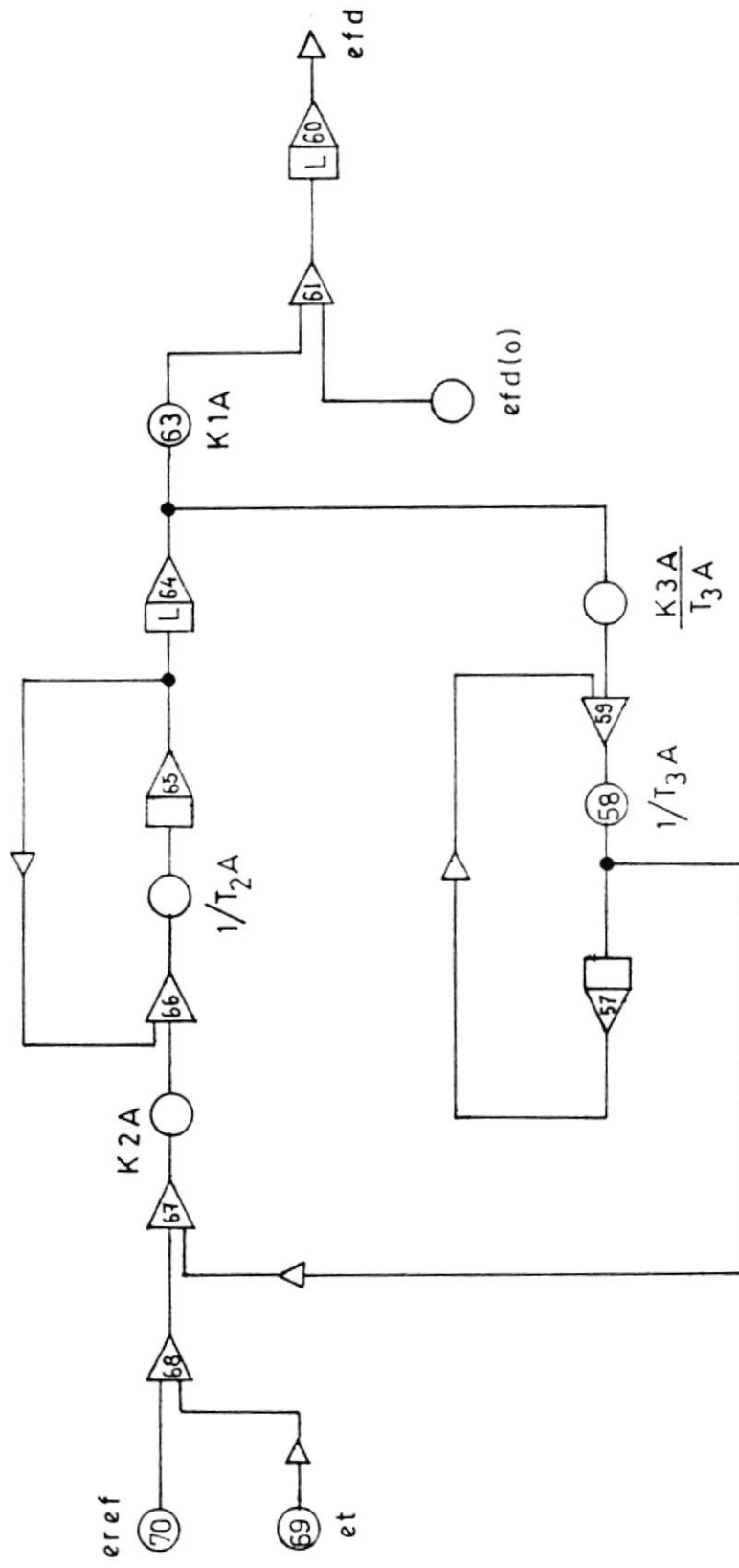


FIGURA : 5.b
SIMULACION DE LA EXCITATRIZ DE TIRISTORES

DATOS PARA SIMULACION

REGULADOR DE VELOCIDAD

BLOQUE	TIPO	ENT. 1	ENT. 2	ENT. 3
1	+	3	-2	0
2	I	0	6	0
3	K	0	0	0
4	G	2	0	0
5	1	0	0	0
6	+	-4	7	0
7	+	-5	8	0
8	+	9	-10	0
9	I	0	11	0
10	W	14	12	0
11	+	14	-9	0
12	I	10	0	0
14	L	16	0	0
15	L	17	0	0
16	I	15	0	0
17	I	0	17	18
18	+	1	-19	0
19	W	14	21	0
20	I	21	0	0
21	W	20	14	0

BLOQUE	PAR. 1	PAR. 2	PAR. 3
2	0.0000	0.2000	.0000
3	0.0028	0.0000	.0000
4	0.9000	0.0000	.0000
5	0.0050	0.0000	.0000
9	0.0000	2.0000	.0000
10	2.0000	-2.0000	.0000
14	2.0000	-1.0000	.0000
15	2.0000	-1.0000	.0000
17	0.0000	-10.0000	50.0000
19	0.0400	1.0000	.0000
21	-0.2000	0.3100	.0000

EXCITRATIZ D.C.

NOMBRE	BLOQUE	TIPO	ENT. 1	ENT. 2	ENT. 3
	70	K	0	0	0
	69	K	0	0	0
	68	+	70	-69	0
	67	+	68	-58	0
	66	W	67	65	0
	65	I	0	66	0
	64	L	65	0	0
	63	W	64	62	0
	62	I	0	63	0
	61	+	62	56	0
Voltaje de Campo	60	L	61	0	0
	59	W	60	57	0
	58	G	59	0	0
	57	I	58	0	0
	56	K	0	0	0

BLOQUE	PAR. 1	PAR. 2	PAR. 3
70	1.0000	0.0000	.0000
69	0.9900	0.0000	.0000
66	1.0000	-1.0000	.0000
65	0.0000	10.0000	.0000
64	100.0000	-100.0000	.0000
63	80.0000	-1.0000	.0000
62	0.0000	0.5000	.0000
60	4.0000	0.0000	.0000
59	0.1000	-1.0000	.0000
58	2.0000	0.0000	.0000
56	0.0000	0.0000	.0000

EXCITRATIZ A.C. ROTATIVA

NOMBRE	BLOQUE	TIPO	ENT. 1	ENT. 2	ENT. 3
	70	K	0	0	0
	69	K	0	0	0
	68	+	70	-69	0
	67	+	68	-58	0
	66	W	67	65	0
	65	I	0	66	0
	64	L	65	0	0
	63	W	64	62	0
	62	I	0	63	0
	61	+	62	56	0
Voltaje de Campo	60	L	61	0	0
	59	W	64	57	0
	58	G	59	0	0
	57	I	58	0	0
	56	K	0	0	0

BLOQUE	PAR. 1	PAR. 2	PAR. 3
70	1.0000	0.0000	.0000
69	0.9900	0.0000	.0000
66	1.0000	-1.0000	.0000
65	0.0000	10.0000	.0000
64	100.0000	-100.0000	.0000
63	80.0000	-1.0000	.0000
62	0.0000	0.5000	.0000
60	4.0000	0.0000	.0000
59	0.1000	-1.0000	.0000
58	2.0000	0.0000	.0000
56	0.0000	0.0000	.0000

EXCITRATIZ TIRISTORES

NOMBRE	BLOQUE	TIPO	ENT. 1	ENT. 2	ENT. 3
	70	K	0	0	0
	69	K	0	0	0
	68	+	70	-69	0
	67	+	68	-58	0
	66	W	67	65	0
	65	I	0	66	0
	64	L	65	0	0
	63	G	64	0	0
	61	+	63	56	0
Voltaje de Campo	60	L	61	0	0
	59	W	64	57	0
	58	G	59	0	0
	57	I	58	0	0
	56	K	0	0	0

BLOQUE	PAR. 1	PAR. 2	PAR. 3
70	1.0000	0.0000	.0000
69	0.9900	0.0000	.0000
66	1.0000	-1.0000	.0000
65	0.0000	10.0000	.0000
64	100.0000	-100.0000	.0000
63	80.0000	0.0000	.0000
60	4.0000	0.0000	.0000
59	0.1000	-1.0000	.0000
58	2.0000	0.0000	.0000
56	0.0000	0.0000	.0000

OBTENCION DE LOS RESULTADOS

REGULADOR DE VELOCIDADES

SALIDA 14	SALIDA 2	SALIDA 7	TIEMPO	- .0060	.0180	SALIDA 8
.0000	.0000	.0000	.000	I-----+	I	.0000
.0092	-.0015	-.0073	1.000	I-----+	I	-.0023
.0143	-.0017	.0021	2.000	I-----+	I	.0071
.0147	-.0003	.0091	3.000	I-----+	I	.0141
.0115	.0017	.0110	4.000	I-----+	I	.0160
.0073	.0032	.0083	5.000	I-----+	I	.0133
.0043	.0038	.0036	6.000	I-----+	I	.0086
.0035	.0034	-.0004	7.000	I-----+	I	.0046
.0047	.0026	-.0020	8.000	I-----+	I	.0030
.0068	.0018	-.0012	9.000	I-----+	I	.0038
.0085	.0015	.0011	10.000	I-----+	I	.0061
.0092	.0017	.0032	11.000	I-----+	I	.0082
.0098	.0021	.0043	12.000	I-----+	I	.0093
.0078	.0026	.0042	13.000	I-----+	I	.0092
.0068	.0026	.0032	14.000	I-----+	I	.0082
.0064	.0028	.0020	15.000	I-----+	I	.0070
.0064	.0027	.0013	16.000	I-----+	I	.0063
.0069	.0024	.0013	17.000	I-----+	I	.0063
.0074	.0023	.0017	18.000	I-----+	I	.0067
.0077	.0023	.0022	19.000	I-----+	I	.0072
.0077	.0024	.0027	20.000	I-----+	I	.0077
.0075	.0025	.0028	21.000	I-----+	I	.0078
.0073	.0026	.0026	22.000	I-----+	I	.0076
.0071	.0026	.0023	23.000	I-----+	I	.0073
.0070	.0026	.0021	24.000	I-----+	I	.0071
.0071	.0025	.0020	25.000	I-----+	I	.0070

EXCITATRIZ D.C.

SALIDA 58	SALIDA 62	SALIDA 65	TIEMPO	.0000	.8000	SALIDA 60
.0000	.0000	.0000	.000	+	I	.0000
.0070	.0996	.0029	1.000	I-----+	I	.0996
.0064	.1634	.0036	2.000	I-----+	I	.1634
.0058	.2213	.0041	3.000	I-----+	I	.2213
.0053	.2740	.0047	4.000	I-----+	I	.2740
.0048	.3218	.0052	5.000	I-----+	I	.3218
.0044	.3653	.0056	6.000	I-----+	I	.3653
.0040	.4049	.0060	7.000	I-----+	I	.4049
.0036	.4408	.0064	8.000	I-----+	I	.4408
.0033	.4735	.0067	9.000	I-----+	I	.4735
.0030	.5032	.0070	10.000	I-----+	I	.5032
.0027	.5302	.0073	11.000	I-----+	I	.5302
.0025	.5548	.0075	12.000	I-----+	I	.5548
.0022	.5771	.0077	13.000	I-----+	I	.5771
.0020	.5974	.0080	14.000	I-----+	I	.5974
.0018	.6158	.0081	15.000	I-----+	I	.6158
.0017	.6326	.0083	16.000	I-----+	I	.6326
.0015	.6478	.0085	17.000	I-----+	I	.6478
.0014	.6617	.0086	18.000	I-----+	I	.6617
.0013	.6742	.0087	19.000	I-----+	I	.6742
.0011	.6857	.0088	20.000	I-----+	I	.6857
.0010	.6961	.0089	21.000	I-----+	I	.6961
.0009	.7055	.0090	22.000	I-----+	I	.7055
.0009	.7141	.0091	23.000	I-----+	I	.7141
.0008	.7220	.0092	24.000	I-----+	I	.7220
.0007	.7291	.0093	25.000	I-----+	I	.7291

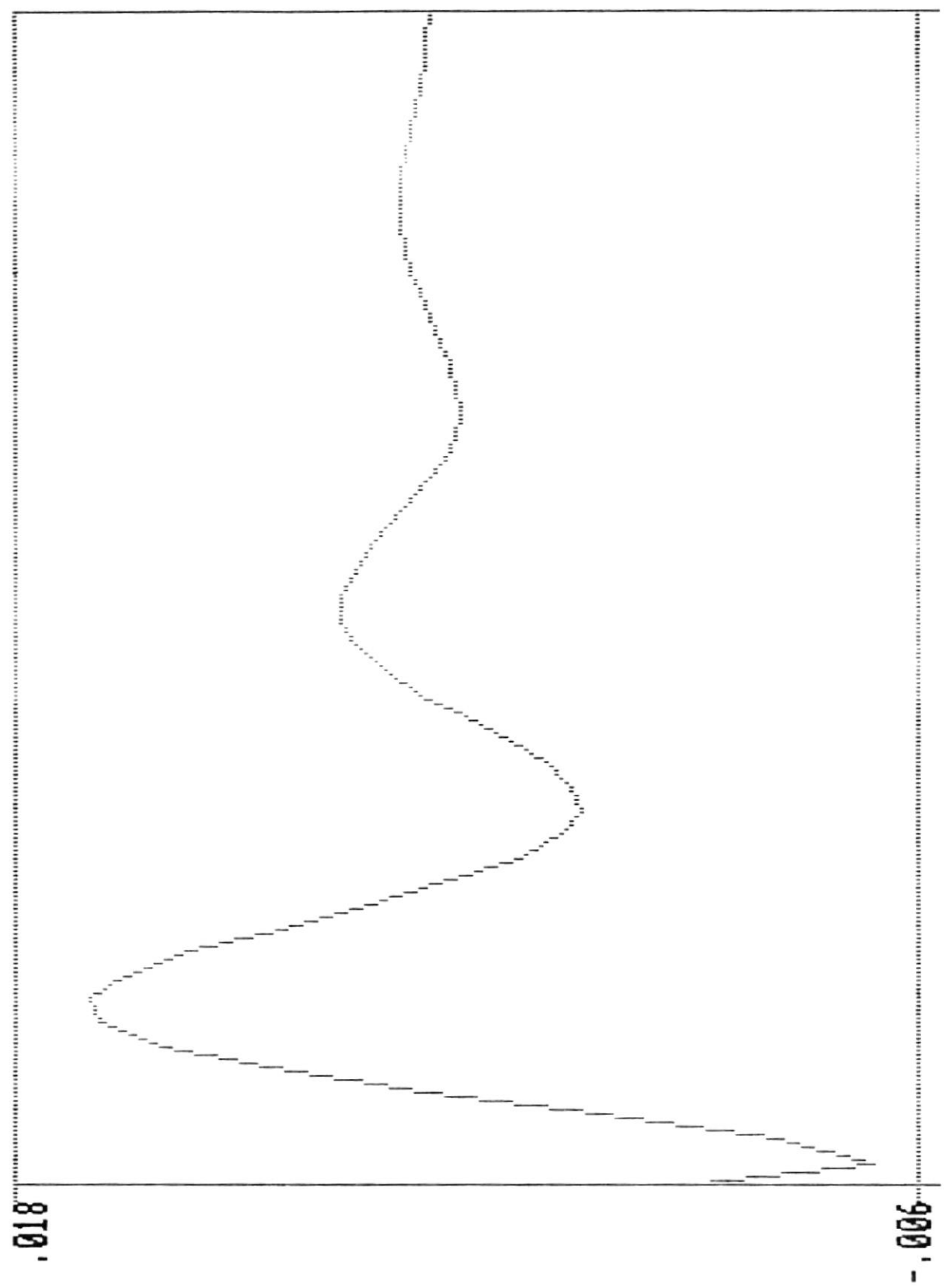
EXCITATRIZ A.C. ROTATIVA

SALIDA 59	SALIDA 62	SALIDA 65	TIEMPO	.0000	.8000	SALIDA 60
.0000	.0000	.0000	.000	+	I	.0000
.0002	.2665	.0095	1.000	I----->	I	.2665
.0000	.4699	.0099	2.000	I----->	I	.4699
.0000	.5984	.0100	3.000	I----->	I	.5984
.0000	.6775	.0100	4.000	I----->	I	.6775
.0000	.7256	.0100	5.000	I----->	I	.7256
.0000	.7549	.0100	6.000	I----->	I	.7549
.0000	.7726	.0100	7.000	I----->	I	.7726
.0000	.7834	.0100	8.000	I----->	I	.7834
.0000	.7899	.0100	9.000	I----->	I	.7899
.0000	.7939	.0100	10.000	I----->	I	.7939
.0000	.7963	.0100	11.000	I----->	I	.7963
.0000	.7977	.0100	12.000	I----->	I	.7977
.0000	.7986	.0100	13.000	I----->	I	.7986
.0000	.7992	.0100	14.000	I----->	I	.7992
.0000	.7995	.0100	15.000	I----->	I	.7995
.0000	.7997	.0100	16.000	I----->	I	.7997
.0000	.7998	.0100	17.000	I----->	I	.7998
.0000	.7999	.0100	18.000	I----->	I	.7999
.0000	.7999	.0100	19.000	I----->	I	.7999
.0000	.8000	.0100	20.000	I----->	I	.8000
.0000	.8000	.0100	21.000	I----->	I	.8000
.0000	.8000	.0100	22.000	I----->	I	.8000
.0000	.8000	.0100	23.000	I----->	I	.8000
.0000	.8000	.0100	24.000	I----->	I	.8000
.0000	.8000	.0100	25.000	I----->	I	.8000

EXCITATRIZ TIRISTORES

SALIDA 58	SALIDA 61	SALIDA 65	TIEMPO	.0000	.8000	SALIDA 60
.0000	.0000	.0000	.000	+	I	.0000
.0004	.7634	.0095	1.000	I-----→	I	.7634
.0001	.7929	.0099	2.000	I-----→	I	.7929
.0000	.7986	.0100	3.000	I-----→	+	.7986
.0000	.7997	.0100	4.000	I-----→	+	.7997
.0000	.7999	.0100	5.000	I-----→	+	.7999
.0000	.8000	.0100	6.000	I-----→	+	.8000
.0000	.8000	.0100	7.000	I-----→	+	.8000
.0000	.8000	.0100	8.000	I-----→	+	.8000
.0000	.8000	.0100	9.000	I-----→	+	.8000
.0000	.8000	.0100	10.000	I-----→	+	.8000
.0000	.8000	.0100	11.000	I-----→	+	.8000
.0000	.8000	.0100	12.000	I-----→	+	.8000
.0000	.8000	.0100	13.000	I-----→	+	.8000
.0000	.8000	.0100	14.000	I-----→	+	.8000
.0000	.8000	.0100	15.000	I-----→	+	.8000
.0000	.8000	.0100	16.000	I-----→	+	.8000
.0000	.8000	.0100	17.000	I-----→	+	.8000
.0000	.8000	.0100	18.000	I-----→	+	.8000
.0000	.8000	.0100	19.000	I-----→	+	.8000
.0000	.8000	.0100	20.000	I-----→	+	.8000
.0000	.8000	.0100	21.000	I-----→	+	.8000
.0000	.8000	.0100	22.000	I-----→	+	.8000
.0000	.8000	.0100	23.000	I-----→	+	.8000
.0000	.8000	.0100	24.000	I-----→	+	.8000
.0000	.8000	.0100	25.000	I-----→	+	.8000

REGULADOR DE VELOCIDADES

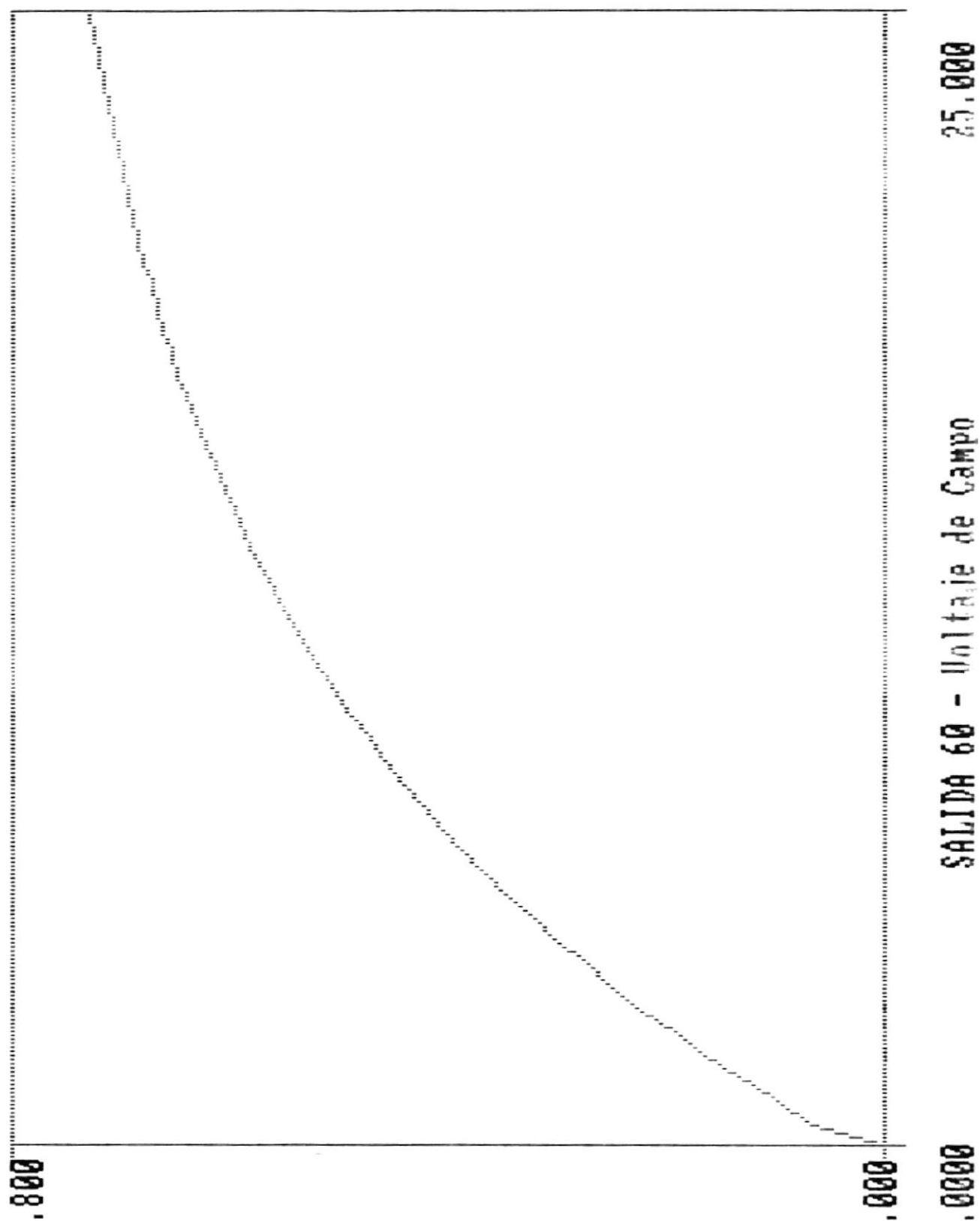


25,000

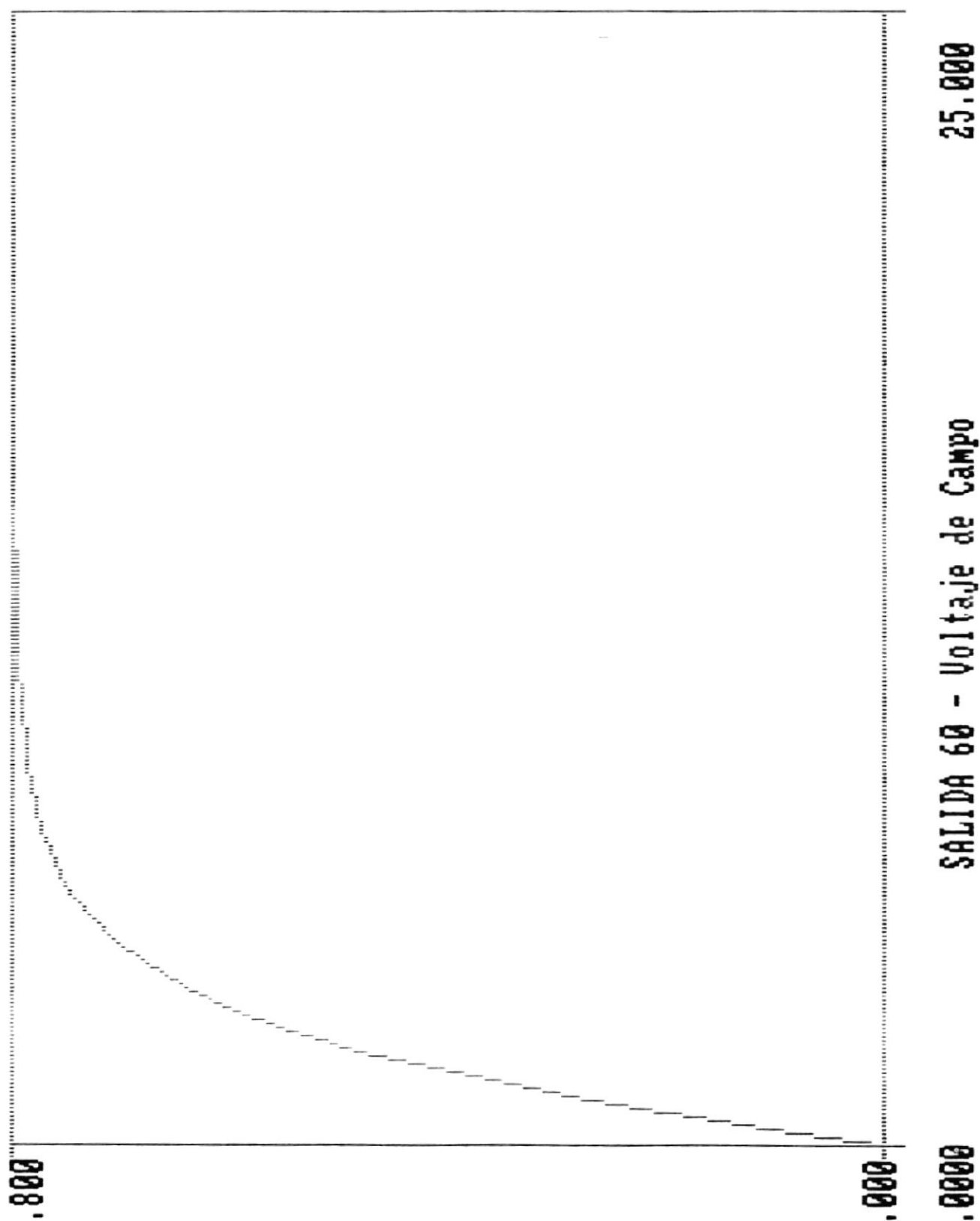
SALIDA 8 - Torque Mecanico

0000

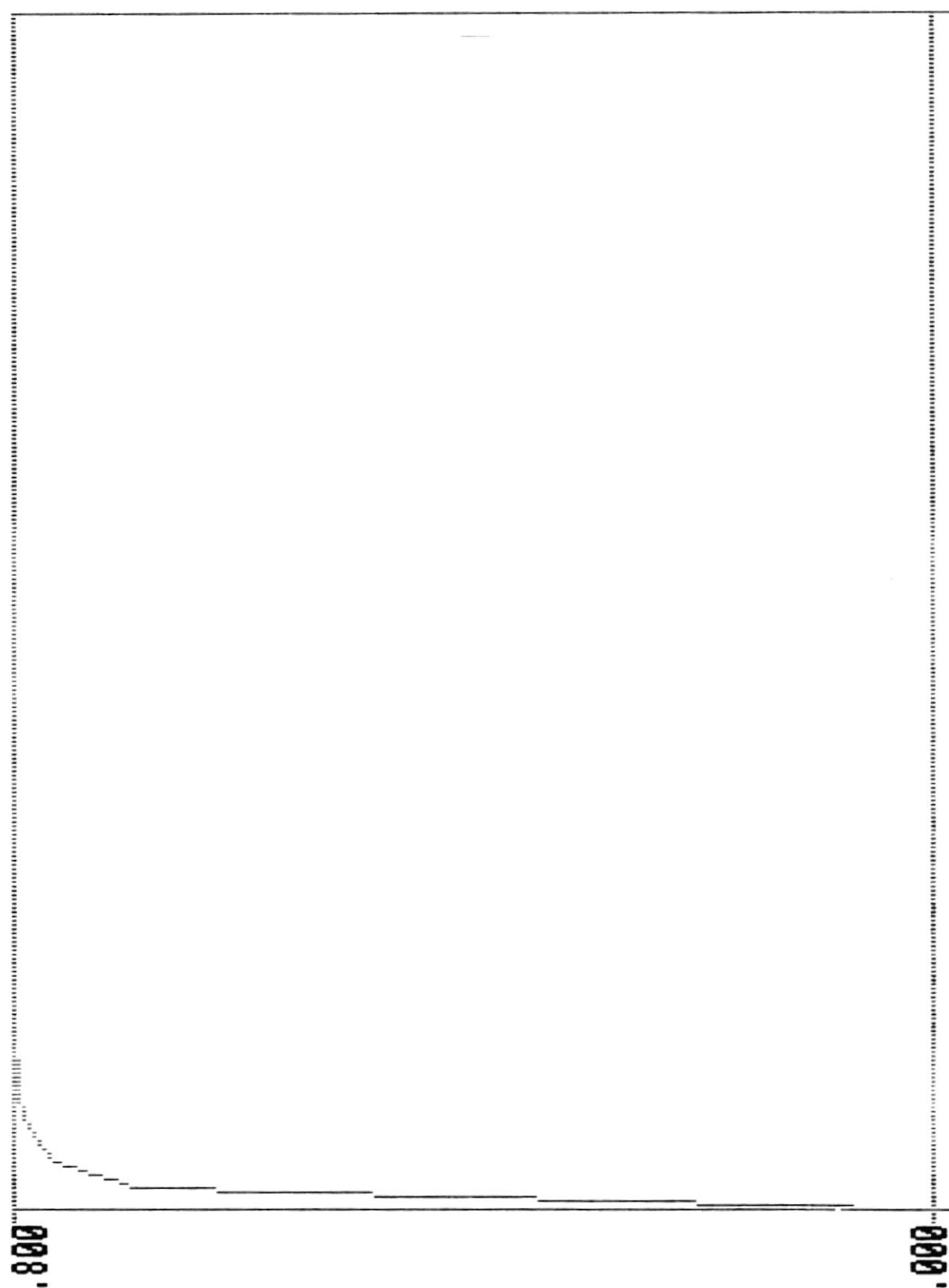
EXCITATRIZ D.C.



EXCITATRIZ A.C. ROTATIVA



EXCITATRIZ TIRISTORES



CONCLUSIONES

Los sistemas dinámicos y la teoría de control han llegado a ser respetables no sólo como una herramienta aguda sino como un medio realístico de percibir una serie de eventos fuera del dominio común de la ingeniería.

Cuando el investigador obtiene las ecuaciones del sistema, se enfrenta al trabajo de resolver esas ecuaciones para obtener resultados numéricos de las mismas. Es aquí, cuando interviene el computador, pues, su potencia radica en poder entregar soluciones rápidas y eficientes.

Inicialmente, todas las simulaciones se realizaban en computadores analógicos que pueden resolver ecuaciones diferenciales por comparaciones con circuitos electrónicos. El corazón de un computador analógico es el amplificador operacional. La facilidad que presenta este computador para programar la solución de ecuaciones diferenciales, hizo que su uso se difundiera rápidamente, y más aún, las técnicas de simulación analógicas estaban bastante desarrolladas cuando todavía no se hacían los primeros esfuerzos para desarrollar la simulación digital. Sin embargo, con la evolución en los últimos años de los

computadores digitales, la simulación digital ha ganado mucho terreno frente a la simulación analógica.

En sus primeros pasos, la simulación digital trató de transferir todos los programas de computadores analógicos a computadores digitales surgiendo los programas de simulación analógica-digital. Estos programas fueron perfeccionándose. Bloques más versátiles fueron añadidos y, poco a poco, se llegó a los programas de simulación de sistemas continuos. El CSMP fue desarrollado en 1965 por IBM para su computador 1130. En este sistema se basó la implementación del presente trabajo.

Este trabajo fue hecho con la finalidad de ayudar en las actividades docentes, especialmente, del Laboratorio de Control Automático, de la Facultad de Ingeniería Eléctrica. Su uso en microcomputadores PC esta orientado a ese fin.

Al ser escrito, para microcomputadores personales, tiene la gran ventaja de ser transportable y por lo tanto, tener una difusión mayor. Si es corrido en un microcomputador modelo AT, su rapidez va a ser bastante elevada.

Se ha mejorado notablemente el programa, respecto del original, en lo que es entrada y salida de información principalmente. Esta versión tiene la ventaja de poder

realizar gráficos de media resolución por pantalla, lo que lo convierte en una excelente herramienta de estudio.

Sin embargo, para el futuro, el programa puede ser notablemente mejorado si se lo vuelve a juntar con las bibliotecas necesarias para activar un coprocesador matemático .8087. Esto incrementaría su velocidad porque el microcomputador sería ayudado en todas las operaciones matemáticas (el CSMP se basa en un elevado número de cálculos matemáticos) por el coprocesador.

También, se podría incluir en el programa nuevos bloques para simulación actualmente posee 30 bloques, pero podría aumentarse el número de ellos.

Finalmente, podría crearse una subrutina que realice la graficación de la salida de los bloques en un graficador.

BIBLIOGRAFIA

1. NAYLOR, BALINTFY, BURDICK, CHU. Técnicas de Simulación en Computadoras. Limusa, México, 1980.
2. TORRES, CZITROM. Métodos para la Solución de Problemas con Computadora Digital. Representaciones y Servicios de Ingeniería S.A., México, 1980.
3. AUSLANDER, TAKAHASHI, RABINES. Introducción a Sistemas y Control. McGraw-Hill, México, 1976.
4. SMITH M. JON. Mathematical Modeling and Digital Simulation for Engineers and Scientists. John Wiley & Sons, Inc., U.S.A., 1977.
5. McCALLA R. THOMAS. Introduction to Numerical Methods and FORTRAN Programming. John Wiley & Sons, Inc., U.S.A., 1967.
6. CARNAHAN, LUTHER, WILKES. Applied Numerical Methods. John Wiley & Sons, Inc., U.S.A., 1969.



A.F. 142245

