

ESCUELA POLITECNICA NACIONAL

FACULTAD DE INGENIERIA ELECTRICA

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



HUGO RAMIRO ORTIZ TULCAN

MARZO - 1990

AGRADECIMIENTO

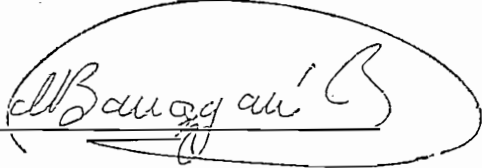
Mi sincero agradecimiento al Ing. Marco Barragán por sus valiosos consejos y tiempo dedicado a la dirección de esta tesis.

De igual forma a todas aquellas personas que colaboraron para la realización de la misma.

A MIS PADRES

CERTIFICACION

Certifico que el presente trabajo
ha sido realizado en su totalidad
por el Sr. Hugo Ramiro Ortiz Tulcán

A handwritten signature in cursive script, enclosed within an oval-shaped border. The signature appears to read "Marco Barragán".

Ing. Marco Barragán

CONTENIDO

CAPITULO I : INTRODUCCION

1.1	Introducción	1
1.2	Principales Técnicas de Control Adaptivo	5
1.3	Control Adaptivo con Modelo de Referencia	11

CAPITULO II : CONTROL LINEAL PARA SISTEMAS DISCRETOS DE FASE MINIMA

2.1	Representación del Sistema	17
2.2	Objetivos del Control Lineal	18
2.3	Estructura del Sistema de Control Lineal	21
2.4	Algoritmo de Control Lineal	27
2.5	Características	28

CAPITULO III : IDENTIFICACION PARAMETRICA

3.1	Principales Técnicas de Identificación	30
3.2	Identificación Paramétrica Discreta	33
3.3	Identificación Recursiva de Mínimos Cuadrados	37
3.4	Propiedades de la IRMC	43

CAPITULO IV : CONTROL ADAPTIVO CON MODELO DE REFERENCIA

4.1	Preliminares	47
4.2	Objetivos del Control	53
4.3	Ley de Control Adaptivo para el Sistema	54
4.4	Estructura e Implementación del Control Adaptivo	58
4.5	Algoritmo de Control Adaptivo MRAC	63

CAPITULO V : SIMULACION DIGITAL

5.1	Esquema para Simulación de Control Lineal	66
5.2	Esquema para Simulación de Control Adaptivo	71
5.3	Desarrollo del Programa de Simulación	77

CAPITULO VI : RESULTADOS Y CONCLUSIONES

6.1	Resultados	128
6.2	Conclusiones	186
6.3	Recomendaciones	189

APENDICES

APENDICE A :	Prueba de Identidad	191
APENDICE B :	Verificación de Equivalencia	193
APENDICE C :	Simbología	195
APENDICE D :	Manual de Usuario del Programa	196

BIBLIOGRAFIA	:	204
REFERENCIAS	:	205
LISTADO DE PROGRAMAS	:	206

CAPITULO I

INTRODUCCION

1.1 INTRODUCCION

1.2 PRINCIPALES TECNICAS DE CONTROL ADAPTIVO

1.3 CONTROL ADAPTIVO CON MODELO DE REFERENCIA

CAPITULO I

INTRODUCCION

1.1 INTRODUCCION

Un importante campo de aplicación del control automático considera aquellos procesos físicos que requieren sistemas de control realimentados capaces de funcionar para diferentes puntos de operación. En algunas circunstancias, los parámetros del modelo sobre el cual se basa el control en lazo cerrado toman durante la operación del proceso un amplio rango de valores (por ejemplo, debido a cambios en la masa y en las condiciones atmosféricas en un sistema de control de vuelo), de forma tal que un sistema de control convencional basado en parámetros fijos no está en capacidad de proporcionar una regulación adecuada.

Los cambios paramétricos son frecuentemente tratados mediante ciertos sistemas de control basados en parámetros fijos y acompañados de una apropiada lógica de conmutación; alternativamente, en algunos casos los valores que adquieren

los parámetros son precalculados y almacenados como funciones del punto de operación y se utiliza un sistema de control simple con ganancias funcionalmente dependientes de los parámetros almacenados.

No obstante si el número de parámetros del modelo es grande, si los parámetros no pueden ser calculados con suficiente precisión o si se requiere un control ajustado que cumpla con especificaciones rígidas, ningún tipo convencional de control podrá ser capaz de proporcionar una regulación adecuada.

Una alternativa promisoría, potencialmente aplicable en situaciones como éstas, es la de un sistema de control adaptivo. Un control de este tipo tiene la capacidad de ajustar sus propios parámetros para compensar un significativo cambio en las características del proceso como resultado de la transferencia de un punto de operación a otro. Intuitivamente, un regulador adaptivo podrá cambiar su comportamiento en respuesta a cambios en la dinámica del proceso y ante la presencia de perturbaciones.

Las investigaciones sobre control adaptivo se iniciaron a partir de los primeros años de la década del 50. Se motivaron principalmente por las dificultades presentadas por los sistemas realimentados linealmente con ganancias constantes para el diseño de controladores automáticos de vuelo. Sin una base teórica de sustentación se dieron los primeros pasos impulsados por el entusiasmo y la necesidad.

A partir de 1.960 se presentaron muchas contribuciones a la teoría del control automático, las cuales fueron importantes para el desarrollo del control adaptivo. Fueron introducidas por aquellos años asociativamente las teorías del espacio de estado y estabilidad. Se obtuvieron en los años subsiguientes importantes resultados en la teoría del control estocástico. Las técnicas de programación dinámica ayudaron en la comprensión de los procesos adaptivos. Significativas contribuciones constituyeron el desarrollo de técnicas para la identificación de sistemas y la estimación paramétrica, permitiendo así la presentación de múltiples esquemas de control adaptivo.

En los últimos años de la década del 70 y en los primeros de los 80 el trabajo estuvo dirigido especialmente al estudio de estabilidad en los esquemas adaptivos. Para ello se recurrió al planteamiento de algunas suposiciones restrictivas que en la actualidad se tratan de eliminar con el fin de establecer condiciones de estabilidad universal.

El rápido y revolucionario progreso de la microelectrónica ha hecho posible la implementación de reguladores adaptivos simples y económicos que en los días actuales han contribuido notablemente para un vigoroso desarrollo en la industria del control automático.

El presente trabajo considera el estudio de una de las técnicas de control adaptivo más utilizadas: la del control adaptivo con modelo de referencia (MRAC). Se realiza el

estudio teórico y una simulación numérica de esta técnica aplicada a sistemas discretos.

El desarrollo del tema contempla en el capítulo II el establecimiento de un algoritmo eficaz de control lineal para sistemas discretos. El esquema de control adaptivo estará basado en la ley de control lineal, pero en este caso los parámetros del controlador se deberán ajustar o actualizar en base a un proceso de identificación. Este proceso se lo efectúa mediante la aplicación de un algoritmo de identificación paramétrica discreta del tipo recursivo de mínimos cuadrados con factor de olvido.

El capítulo III considera el estudio de la técnica de identificación paramétrica utilizada y en el capítulo IV se acoplan los capítulos precedentes dando lugar a la técnica de control que permitirá al controlador adaptarse a los cambios que pudieran suceder en la planta.

En el capítulo V se establecen las simulaciones numéricas tanto para el control lineal como para el control adaptivo MRAC y en el capítulo VI se presentan los resultados obtenidos así como también las conclusiones pertinentes.

Se considera en este trabajo el estudio en un ambiente determinístico con plantas de fase mínima de hasta tercer orden y con retardos de uno o dos pasos, por considerar que dadas las características del control no se requiere en términos reales una precisión mayor en la modelación.

1.2 PRINCIPALES TECNICAS DE CONTROL ADAPTIVO

A través de los años han sido propuestos múltiples esquemas de control adaptivo, algunos de ellos orientados a satisfacer propósitos específicos y la mayor parte de ellos procurando establecer un marco teórico generalizado para su estudio.

Las diversas contribuciones presentadas podrían ser agrupadas en tres corrientes de fundamental importancia y que a su vez han dado lugar al establecimiento de tres esquemas básicos de control adaptivo: selección de ganancias, control adaptivo con modelo de referencia y regulación autosintonizable. Además revisten importancia por sus aplicaciones los esquemas de control adaptivo basados en la teoría de control estocástico.

A continuación se realiza una descripción básica de los esquemas señalados:

1.2.1 Control adaptivo mediante selección de ganancias

En algunos sistemas se presentan variables auxiliares que se relacionan adecuadamente con las características dinámicas del proceso. Si esas variables pueden ser medidas sería posible usarlas para cambiar los parámetros del regulador. Esta técnica es llamada de "selección de ganancias" debido a que el esquema fue originalmente usado para acomodar

cambios en las ganancias de un proceso. El diagrama de bloques de un sistema que utiliza esta técnica se observa a continuación en la FIG. 1.1 .

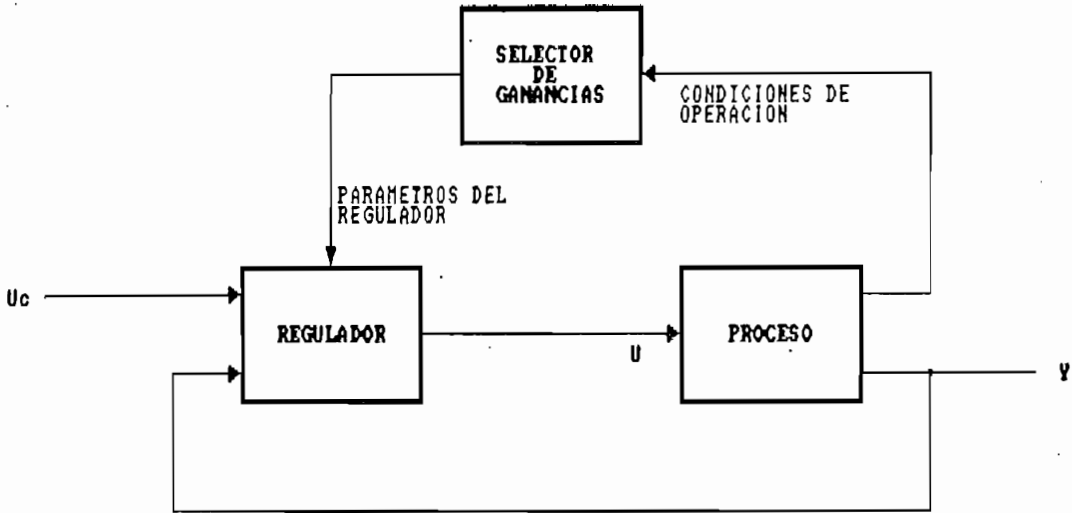


FIG. 1.1 DIAGRAMA DE BLOQUES DE CONTROL ADAPTIVO CON SELECCIÓN DE GANANCIAS

En este tipo de esquema se presenta una compensación en lazo abierto. No existe una realimentación que compense una selección incorrecta, pudiendo ser visto como un sistema de control realimentado donde la ganancia de realimentación es ajustada por una compensación en adelanto.

El problema clave en el diseño de sistemas con selección de ganancias es encontrar las variables de selección apropiadas. Esto es normalmente realizado en base al conocimiento de la física del sistema. Cuando las variables auxiliares de selección han sido obtenidas, los parámetros del regulador son determinados para diferentes condiciones de operación utilizando métodos de diseño adecuados.

La estabilidad y funcionamiento del sistema son evaluados generalmente por simulación, debiéndose poner especial atención en las transiciones entre los diferentes puntos de operación.

La selección de ganancias tiene la ventaja de que los parámetros pueden ser cambiados rápidamente en respuesta a cambios en el proceso. El factor limitante dependerá de cuan rápidamente las variables auxiliares respondan a los cambios.

Existe cierta controversia en nomenclatura con respecto a que si este esquema debería ser considerado como un sistema adaptivo o no debido a que los parámetros son cambiados en lazo abierto. Independientemente de esta controversia, cabe indicar que este esquema es una técnica muy útil para reducir las variaciones paramétricas. Es de fácil implementación mediante la utilización de moderno hardware en control distribuido de procesos.

1.2.2 Regulador autosintonizable (STR)

Fue propuesto por Kalman¹ en 1.958, quien construyó un computador de propósito especial para su implementación. El regulador autosintonizable es un esquema de control adaptivo en el cual al variar los parámetros del proceso se obtienen los parámetros del regulador a través de la solución de un

[1] KALMAN R.E.; "Design of a self-optimizing control system", Trans. ASME, vol 80, 1.958 .

problema de diseño. Un diagrama general de bloques se presenta a continuación en la FIG. 1.2 .

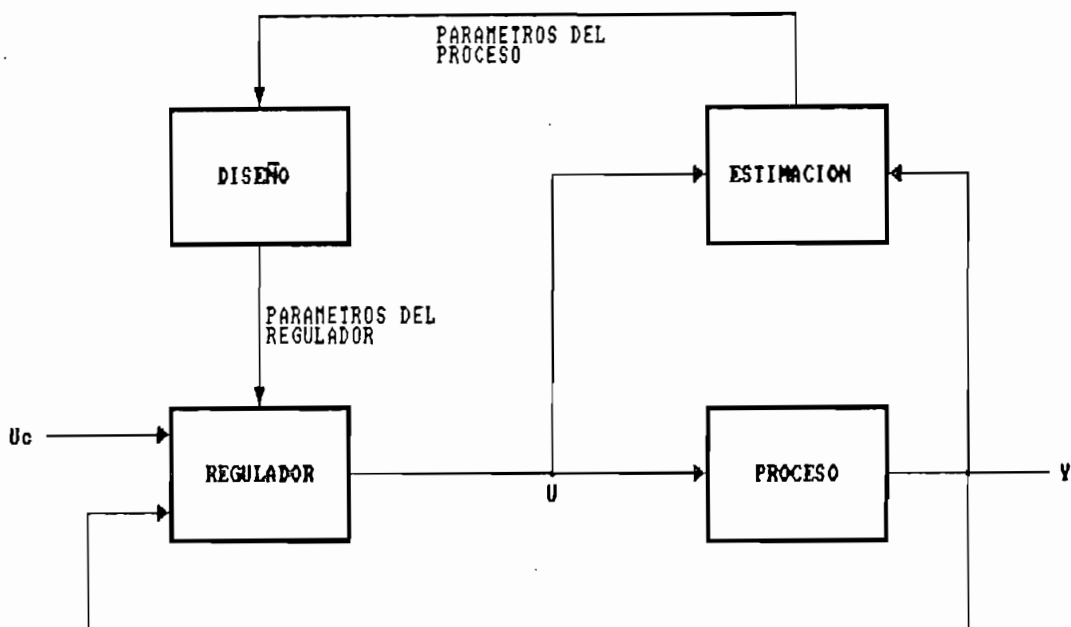


FIG. 1.2 DIAGRAMA DE BLOQUES DE UN REGULADOR AUTOSINTONIZABLE

El regulador adaptivo puede ser visto conformado por dos lazos. El lazo interno está constituido por el proceso y un regulador lineal realimentado ordinario. Los parámetros del regulador son ajustados por el lazo externo, el cual puede estar constituido por un estimador y los cálculos de diseño. El sistema podría ser visto como una automatización de la modelación del proceso y el diseño, donde, el modelo del proceso y el diseño del control son actualizados en cada período de muestreo.

El bloque indicado como de "diseño" representa una solución fuera de línea del problema de diseño del sistema

con parámetros conocidos. Este es conocido como "problema fundamental de diseño" y también puede ser asociado para los demás esquemas de control adaptivo. El bloque correspondiente a la estimación podría estar constituido por alguno de los esquemas existentes para el efecto, tales como: por aproximación estocástica, por mínimos cuadrados ordinarios o generalizados, a través de variables instrumentales, por filtración extendida de Kalman y otros.

Los reguladores autosintonizables han recibido considerable atención, presentando entre sus principales ventajas su flexibilidad, una sólida base teórica y, su relativamente fácil implementación a través de la utilización de microprocesadores. Han sido propuestas muchas modificaciones a los esquemas originales, habiéndose realizado extensiones a reguladores autosintonizables diseñados en base a la consideración de márgenes de fase y ganancia, a la asignación de polos y ceros, por lugar geométrico de las raíces, etc.

1.2.3 Control estocástico adaptivo

En los años precedentes las investigaciones se han orientado hacia la obtención de sistemas adaptivos basados en una estructura teórica unificada. Esto ha sido posible hacerlo usando la teoría de control estocástico no lineal, en la cual, tanto el sistema como los elementos circundantes son descritos mediante un modelo estocástico. De este modo, los parámetros son introducidos en el esquema como variables de

estado y aquellos parámetros inciertos son modelados estocásticamente. La estructura de un regulador óptimo adaptivo es mostrada en la FIG. 1.3 .

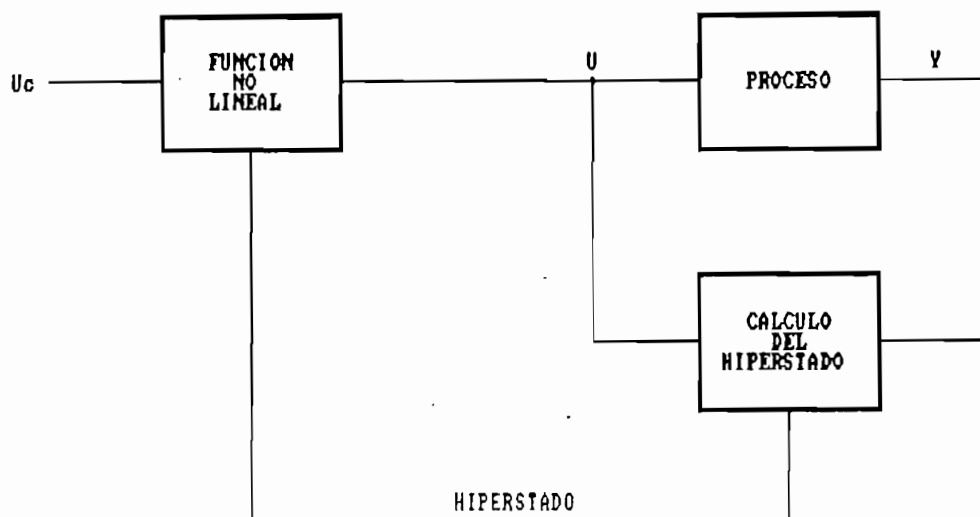


FIG. 1.3 DIAGRAMA DE BLOQUES DE UN REGULADOR ADAPTIVO ESTOCASTICO

El controlador puede ser visto conformado por dos partes: un estimador no lineal y un regulador realimentado. El estimador genera una distribución de probabilidad condicional del estado de medidas. Esta distribución es llamada el "hiperestado del problema". El regulador realimentado es una función lineal cuyos hiperestados están dentro del espacio de control de las variables. Esta función puede ser calculada fuera de línea. El hiperestado debe, sin embargo, ser actualizado en línea. La estructura aparentemente simple de la solución es obtenida al precio de introducir el hiperestado, el cual puede ser de muy alta dimensión. Además, la solución del hiperestado requiere en general la consideración de un problema complicado de filtrado no lineal.

En el esquema no existe una distinción entre los parámetros y las otras variables de estado. Esto significa que el regulador puede manejar rápidamente las variaciones de los parámetros, siendo necesario sin embargo tener cierta información precedente acerca de las propiedades estocásticas de las variaciones de estados y parámetros. La teoría de control estocástico óptimo proporciona otras posibilidades para la obtención de sofisticados algoritmos adaptivos.

1.3 CONTROL ADAPTIVO CON MODELO DE REFERENCIA (MRAC)

Los sistemas adaptivos de control con modelo de referencia fueron propuestos originalmente por Whitaker² a fines de la década del 50 como aplicaciones particulares para sistemas de tiempo continuo. Se consideró inicialmente el problema en el cual las especificaciones son dadas en términos de un modelo de referencia que indica como la salida del proceso debería responder idealmente.

El esquema original propuesto por Whitaker se puede observar en la FIG. 1.4. Sin embargo, es importante señalar que con el transcurrir de los años la filosofía del MRAC ha recibido nuevas e innovadoras contribuciones, que han hecho de ella una técnica "híbrida" ampliamente difundida en diferentes frentes y analizada desde diferentes puntos de vista. No existe realmente un planteamiento único e incontrastable que

[2] WHITAKER H.P.; "Design of model reference adaptive control systems for aircraft", MIT, 1.958 .

especifique a determinado esquema como de MRAC, sino más bien, como se verá en el capítulo IV, se presentan múltiples esquemas de MRAC definidos desde diferentes puntos de vista de la teoría de Control Automático.

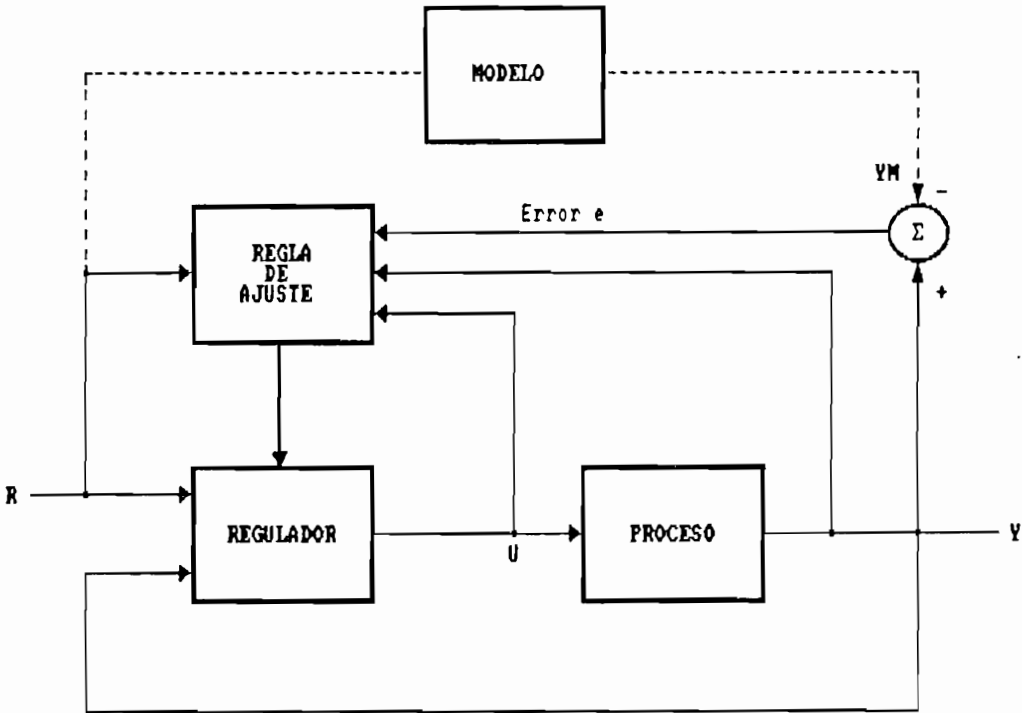


FIG. 1.4 DIAGRAMA DE BLOQUES DEL ESQUEMA ORIGINAL MRAC

En el diagrama de bloques de la FIG. 1.4 se observa la participación de un modelo de referencia en el sistema de control. Se puede considerar al sistema conformado por dos lazos. Un lazo interno de realimentación ordinaria constituido por el proceso y el regulador. Los parámetros del regulador son ajustados por el lazo externo de manera que el error e entre la salida y_M del modelo y la salida y del proceso tienda a ser pequeño. El lazo externo es así también un lazo regulador.

El problema clave es determinar el mecanismo de ajuste de modo que se obtenga un sistema estable que conduzca el error a cero. Este problema no es trivial y no puede ser resuelto con una simple realimentación lineal del error a los parámetros del controlador.

Es oportuno señalar que varios de los esquemas de control adaptivo mantienen una estrecha relación entre sí, tal es el caso particular del regulador autosintonizable y el MRAC. Los dos enfoques son muy similares a pesar de tener diferentes orígenes. Esta similitud puede ser apreciada mediante la comparación de los esquemas correspondientes a cada caso. Ambos sistemas tienen dos lazos de realimentación. El lazo interno es de realimentación ordinaria con el proceso y el regulador. El regulador tiene parámetros ajustables que son determinados por acción del lazo externo; los ajustes se basan en la realimentación de las entradas y salidas del proceso. Los métodos de diseño para el lazo interno y las técnicas de ajuste de los parámetros en el lazo externo presentan ciertas variaciones no fundamentales.

Los parámetros del regulador son actualizados directamente en el MRAC mientras que en el regulador autosintonizable son actualizados indirectamente y previo a cálculos de diseño. Cabe señalar sin embargo que cualquiera de los dos esquemas en consideración puede ser modificado y puesto en términos del otro; así por ejemplo, se presentan esquemas MRAC modificados en los cuales los parámetros son actualizados indirectamente y que son conocidos como esquemas MRAC

indirectos. Publicaciones técnicas especializadas³ utilizan equivalentemente las denominaciones MRAC directo y STR implícito, así como también MRAC indirecto y STR explícito.

En la actualidad se presenta una unificación de criterios y esquemas a través del planteamiento de modelos generales, que, dependiendo del punto de vista desde el cual se los considere, pueden ser tomados como pertenecientes a una u otra técnica.

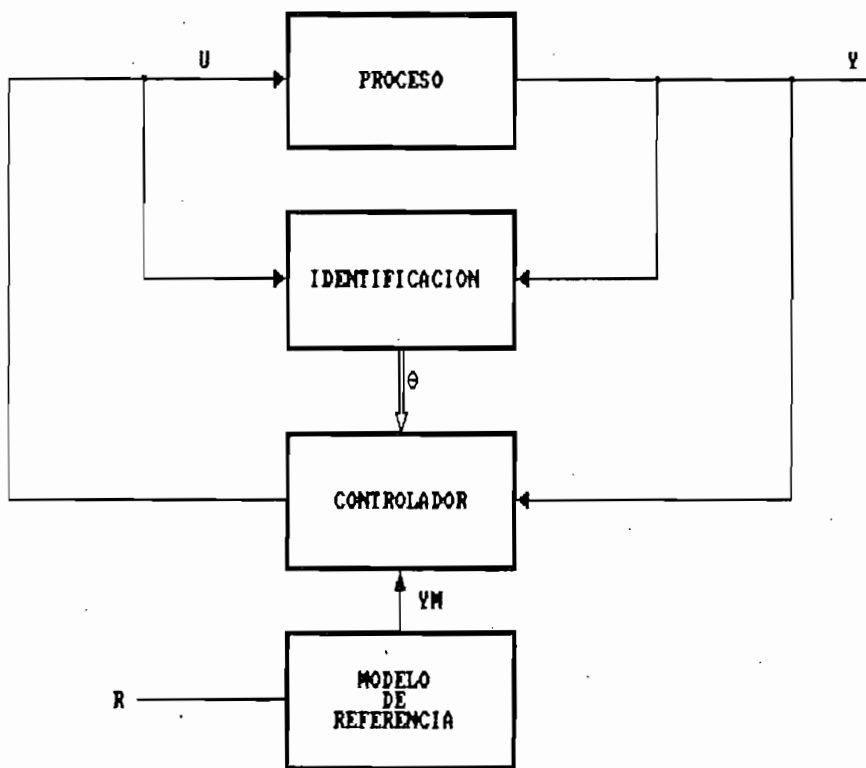


FIG. 1.5 ESQUEMA BASICO DE UNA TECNICA MRAC APLICADA EN CONTROL DISCRETO

Un importante campo de las técnicas MRAC considera los

[3] IEEE Transactions on Automatic Control Automatica (IFAC)

sistemas de control cuya estructura simplificada corresponde a la mostrada en el esquema básico de la FIG. 1.5 , siendo esta una de las estructuras más difundidas y aplicadas en control discreto.

Los sistemas adaptivos con modelo de referencia representan una importante línea en el desarrollo del control adaptivo. Estos han sido sujetos a muchos análisis teóricos que han llevado a su mejor comprensión. Los algoritmos han sido extendidos a sistemas de fase no mínima, a sistemas multivariables, a sistemas no lineales, etc.

Dada la amplitud y complejidad del campo correspondiente al control adaptivo, es oportuno mencionar que, en concordancia con lo señalado en la Introducción, en el presente trabajo se ha optado por el estudio de una de las técnicas MRAC, por tratarse de una técnica base en el control adaptivo, de mayor difusión que las restantes y de amplia aplicación en un considerable grupo de problemas diferentes de Control Automático.

CAPITULO II

CONTROL LINEAL PARA SISTEMAS DISCRETOS DE FASE MINIMA

- 2.1 REPRESENTACION DEL SISTEMA
 - 2.2 OBJETIVOS DEL CONTROL LINEAL
 - 2.3 ESTRUCTURA DEL SISTEMA DE CONTROL LINEAL
 - 2.4 ALGORITMO DE CONTROL LINEAL
 - 2.5 CARACTERISTICAS
-

CAPITULO II

CONTROL LINEAL PARA SISTEMAS DISCRETOS DE FASE MINIMA

2.1 REPRESENTACION DEL SISTEMA

Se considera un sistema discreto univariable de fase mínima descrito mediante una representación polinomial de la forma :

$$y(k) + a_1y(k-1) + a_2y(k-2) + a_3y(k-3) + \dots + a_nAy(k-nA) = b_0u(k-d) + b_1u(k-d-1) + b_2u(k-d-2) + \dots + b_nBu(k-d-nB) + p(k)$$

donde :

$y(k)$ es la salida

$u(k)$ es la entrada

$p(k)$ es una perturbación

d representa el retardo del sistema

a_1 y b_1 son coeficientes constantes

Con la finalidad de describir esta representación de una manera sucinta, se introduce el operador de desplazamiento regresivo q^{-1} . Si $y(k)$ denota el valor de la secuencia $\{y(k)\}$ al tiempo k , donde $k \in \{0, 1, \dots\}$, entonces $q^{-1}y(k)$ denota el valor de la secuencia al tiempo $(k-1)$. Esto es,

$$q^{-1}y(k) = y(k-1) \quad \text{para } k \geq 1$$

$$q^{-1}y(0) = 0$$

y consecuentemente,

$$q^{-i}y(k) = y(k-i) \quad \text{para } k \geq i$$

$$q^{-i}y(k) = 0 \quad \text{para } 0 \leq k < i$$

Estableciendo dos polinomios A y B en el operador de retardo regresivo q^{-1} el sistema puede ser representado de la siguiente forma :

$$A(q^{-1})y(k) = q^{-d}B(q^{-1})u(k) + p(k) \quad [2.1]$$

con :

$$A(q^{-1}) = 1 + a_1q^{-1} + a_2q^{-2} + \dots + a_nAq^{-nA}$$

$$B(q^{-1}) = b_0 + b_1q^{-1} + b_2q^{-2} + \dots + b_nBq^{-nB} \quad [2.2]$$

b_0 es diferente de 0

A continuación, en la FIG. 2.1 se presenta esquemáticamente el sistema en consideración, estructurado mediante diagramas de bloques.

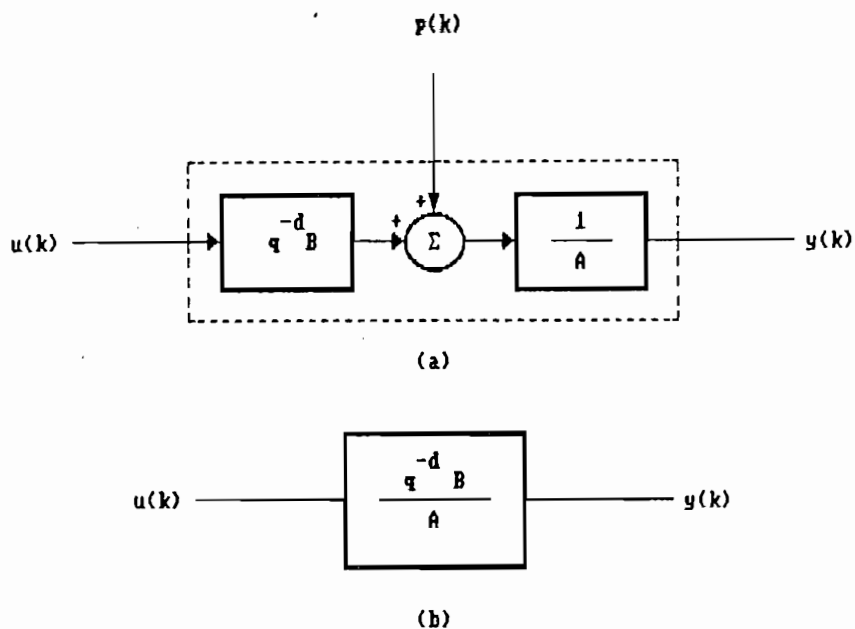


FIG. 2.1 DIAGRAMA DE BLOQUES DE UN SISTEMA DISCRETO
 (a) $p(k) \neq 0$ (b) $p(k) = 0$

2.2 OBJETIVOS DEL CONTROL LINEAL

El esquema de control lineal está orientado con miras hacia su posterior utilización en un esquema de control adaptivo con modelo de referencia. Por tal motivo, en primera instancia, el objetivo fundamental es el de que la salida de la planta siga a una secuencia, denominada de referencia y representada mediante $y^M(k)$, la misma que es generada por un cierto modelo.

El control a establecerse es de tipo lineal, lo cual implica que la señal de control $u(k)$ obtenida a través de una realimentación deberá ser implementada a partir de una combinación lineal de la información disponible hasta ese instante. Dicha información disponible estará dada por :

- las salidas de la planta en ese instante y en los anteriores:
 $y(k), y(k-1), y(k-2), \dots$
- las salidas del modelo de referencia en ese instante y en los anteriores:
 $y_M(k), y_M(k-1), y_M(k-2), \dots$
- las entradas de control anteriores:
 $u(k-1), u(k-2), u(k-3), \dots$ [2.3]

El sistema realimentado en base a una ley de control adecuada podría ser considerado en términos generales como un sistema que tiene por entradas a $y_M(k)$ y $p(k)$, y que dispone de una sola salida, $y(k)$.

Los objetivos del control pueden ser por tanto analizados en base a la consideración de las funciones de transferencia de la salida con respecto a cada una de las entradas señaladas.

2.2.1 Objetivo de seguimiento

La relación existente entre $y(k)$ y $y_M(k)$ en el transcurso del tiempo permite establecer un objetivo de seguimiento para el control; este objetivo puede ser planteado por tanto a través de una función de transferencia que relacione estas dos señales.

Como puede observarse en la FIG. 2.2, si se desea que la salida $y(k)$ siga a la referencia $y_M(k)$ será necesario que, idealmente, la función de transferencia de $y(k)$ a $y_M(k)$ sea la unidad, es decir :

$$y(k) = y_M(k)$$

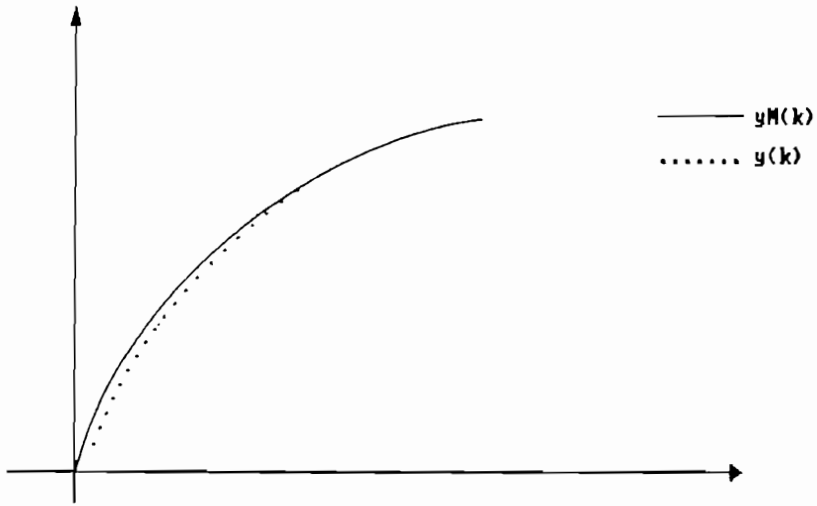


FIG. 2.2 SEGUIMIENTO DE $y(k)$ a $y_M(k)$

El retardo propio del sistema impide que la salida $y(k)$ siga a la referencia $y_M(k)$ sin ningún atraso. Lo que si se puede plantear como objetivo es que la salida siga a la referencia en el menor tiempo posible, que sería el dado por el retardo (d) del sistema. Esto puede expresarse como:

$$y(k) = y_M(k-d)$$

utilizando el operador de desplazamiento regresivo:

$$y(k) = q^{-d}y_M(k)$$

Y planteando como función de transferencia :

$$\frac{y(k)}{yM(k)} = q^{-d} \quad [2.4]$$

2.2.2 Objetivo de regulación

El esquema de control debe estar en capacidad de compensar las perturbaciones que impredeciblemente pueden presentarse en el transcurso del tiempo. Si la función de transferencia de la salida $y(k)$ con respecto a la perturbación $p(k)$ es expresada mediante una relación polinomial en el operador q^{-1} , es decir,

$$\frac{y(k)}{p(k)} = \frac{S(q^{-1})}{Cr(q^{-1})} \quad [2.5]$$

el objetivo de regulación será cumplido si esta función de transferencia es asintóticamente estable. Esto exige que $Cr(q^{-1})$ sea un polinomio normalizado cuyos ceros deben estar dentro del círculo unitario. El polinomio $S(q^{-1})$ depende de la planta y será definido oportunamente.

2.3 ESTRUCTURA DEL SISTEMA DE CONTROL LINEAL

Los objetivos de regulación y seguimiento propuestos son el punto de partida para el diseño e implementación del

sistema de control lineal. Estos dos objetivos pueden ser considerados simultáneamente en una sola expresión de la forma:

$$Cr(q^{-1})[y(k) - q^{-d}y_M(k)] = S(q^{-1})p(k) \quad [2.6]$$

El error de salida en cualquier instante k viene dado por,

$$e(k) = y(k) - q^{-d}y_M(k)$$

es decir:

$$e(k) = y(k) - y_M(k-d) \quad [2.7]$$

Equivalentemente, puede también definirse al error $e(k)$ como:

$$e(k) = y(k+d) - y_M(k)$$

Puede observarse que si en un instante dado k_0 se presenta un error de salida igual a,

$$e(k_0) = y(k_0) - y_M(k_0-d)$$

entonces, dado que el polinomio $Cr(q^{-1})$ es un polinomio asintóticamente estable, este error converge a cero conforme el tiempo crece siempre que:

$$S(q^{-1})p(k) = 0 \quad \text{para todo } k > k_0 \quad [2.8]$$

Por lo tanto, el control a implementarse es tal que el

error planta-modelo se elimina con una dinámica dada por:

$$Cr(q^{-1})e(k) = 0 \quad k > k_0 \quad [2.9]$$

Lo anterior se tendrá para el caso de una perturbación impulso. Si la perturbación es de tipo escalón será necesario agregar a $S(q^{-1})$ un factor $(1 - q^{-1})$, de tal forma que se obtendría:

$$S(q^{-1})(1 - q^{-1})p(k) = S(q^{-1})[p(k) - p(k-1)] \quad [2.10]$$

expresión que también es igual a cero para $k > k_0$.

2.3.1 Ley de control

Considérese la siguiente identidad polinomial a cumplirse para el polinomio $Cr(q^{-1})$:

$$Cr(q^{-1}) = A(q^{-1})S(q^{-1}) + q^{-d}R(q^{-1}) \quad [2.11]$$

donde:

$$\begin{aligned} Cr(q^{-1}) &= 1 + c_1q^{-1} + \dots + c_nCq^{-nC} \\ S(q^{-1}) &= 1 + s_1q^{-1} + \dots + s_nS q^{-nS} \\ R(q^{-1}) &= r_0 + r_1q^{-1} + \dots + r_nR q^{-nR} \end{aligned} \quad [2.12]$$

Cuando S es de grado $d-1$ la ecuación [2.11] tiene solución única y los grados de S y R son mínimos (ver Ap. A) e iguales a:

$$nS = d-1$$

$$nR = \max(nA-1, nC-d) \quad [2.13]$$

Al igualar los coeficientes q^{-1} correspondientes a cada miembro de la ecuación [2.11] se obtiene :

$$\begin{aligned} i=0 & : 1 = 1.s_0 \\ i=1 & : c_1 = 1.s_1 + a_1s_0 \\ i=2 & : c_2 = 1.s_2 + a_1s_1 + a_2s_0 \\ i=3 & : c_3 = 1.s_3 + a_1s_2 + a_2s_1 + a_3s_0 \\ i=4 & : c_4 = 1.s_4 + a_1s_3 + a_2s_2 + a_3s_1 + a_4s_0 \\ & \dots \\ i=d-1 & : c_{d-1} = 1.s_{d-1} + a_1s_{d-2} + a_2s_{d-3} + \dots + a_{d-1}s_0 \\ i=d & : c_d = a_1s_{d-1} + a_2s_{d-2} + \dots + a_d s_0 + r_0 \\ i=d+1 & : c_{d+1} = a_2s_{d-1} + a_3s_{d-2} + \dots + a_{d+1}s_0 + r_1 \\ i=d+2 & : c_{d+2} = a_3s_{d-1} + a_4s_{d-2} + \dots + a_{d+2}s_0 + r_2 \\ & \dots \end{aligned}$$



Las ecuaciones obtenidas permiten una resolución secuencial, ya que, de una ecuación a la siguiente se incrementa una sola incógnita.

La ecuación correspondiente a la dinámica de error puede ser escrita como:

$$Cr(q^{-1})e(k) = Cr(q^{-1})[y(k+d) - yM(k)] \quad k > k_0$$

Expresión que, utilizando la identidad [2.11] y la ecuación [2.1], puede ser desarrollada de la manera siguiente :

$$Cr(q^{-1})e(k) = [A(q^{-1})S(q^{-1}) + q^{-d}R(q^{-1})][y(k+d) - yM(k)]$$

$$Cr(q^{-1})e(k) = A(q^{-1})S(q^{-1})y(k+d) + R(q^{-1})y(k) - Cr(q^{-1})yM(k)$$

$$Cr(q^{-1})e(k) = B(q^{-1})S(q^{-1})u(k) + R(q^{-1})y(k) - Cr(q^{-1})yM(k)$$

[2.14]

Es claro de [2.9] que los objetivos de control serán alcanzados si la entrada de control es computada de modo que el lado derecho de la ecuación [2.14] es igual a cero, es decir:

$$B(q^{-1})S(q^{-1})u(k) + R(q^{-1})y(k) - Cr(q^{-1})yM(k) = 0 \quad k > k_0$$

[2.15]

Las consideraciones precedentes permiten establecer una ley de control dada por:

$$B(q^{-1})S(q^{-1})u(k) = Cr(q^{-1})yM(k) - R(q^{-1})y(k) \quad [2.16]$$

Sea $Bs(q^{-1})$ un polinomio definido como:

$$Bs(q^{-1}) = B(q^{-1})S(q^{-1}) - b_0$$

es decir,

$$Bs(q^{-1}) = (b_0s_1 + b_1)q^{-1} + \dots + b_nBs_{d-1}q^{-nB-d+1}$$

Entonces a partir de [2.16] la ley de control puede expresarse como:

$$[Bs(q^{-1}) + b_0]u(k) = Cr(q^{-1})yM(k) - R(q^{-1})y(k)$$

$$b_0u(k) = Cr(q^{-1})yM(k) - R(q^{-1})y(k) - Bs(q^{-1})u(k)$$

de donde,

$$u(k) = (1/b_0)[Cr(q^{-1})yM(k) - R(q^{-1})y(k) - Bs(q^{-1})u(k)]$$

El esquema correspondiente al control lineal se observa a continuación en la FIG. 2.3 .

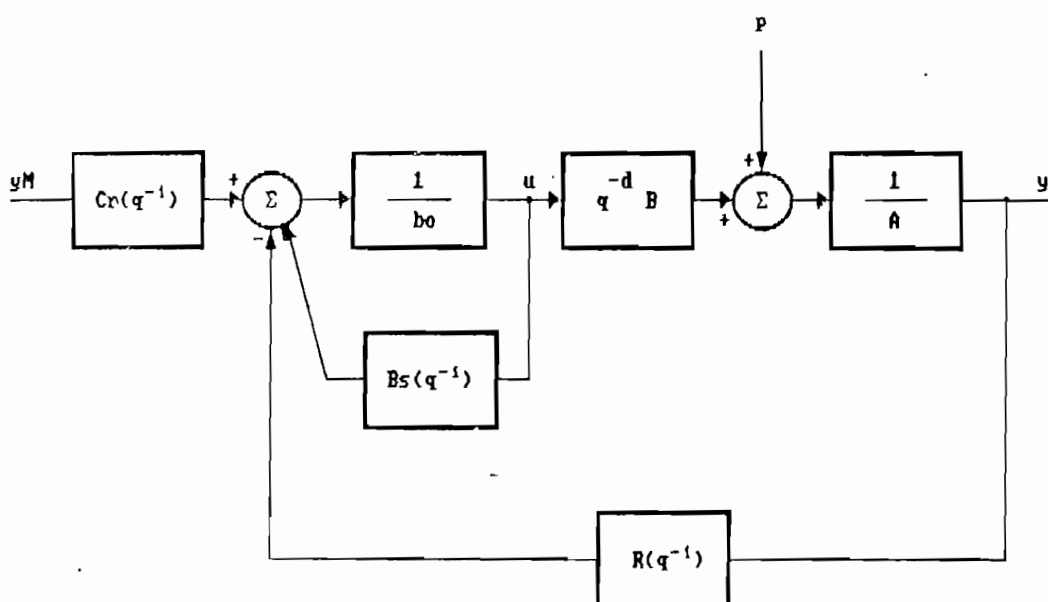
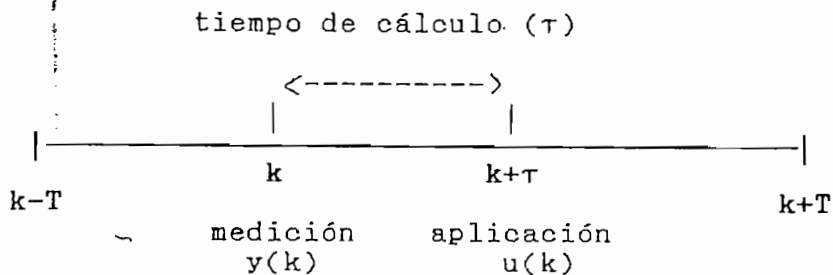


FIG. 2.3 DIAGRAMA DE BLOQUES DEL ESQUEMA DE CONTROL LINEAL

La señal de control en el instante k se calcula utilizando la salida en ese mismo instante, lo cual implica que para el funcionamiento correcto del control el tiempo de cálculo que emplea la computadora en el procesamiento (τ) deberá ser menor que el correspondiente período de muestreo utilizado (T).



Si el tiempo de cálculo τ no es menor que el período de muestreo T se deberá implementar una ley de control en base a las mediciones de las salidas en los instantes anteriores, es decir, $y(k-1)$, $y(k-2)$, etc. Para ello se deberá considerar al polinomio $S(q^{-1})$ de orden d en lugar de $d-1$ y se deberá definir $r_0 = 0$. Los objetivos dados por la expresión [2.6] serán así cumplidos.

2.4 ALGORITMO DE CONTROL LINEAL

La implementación del esquema de control lineal propuesto viene dado por intermedio de los siguientes pasos globales:

1. Se genera una secuencia de referencia $y_M(k)$ utilizando para ello un modelo.
2. Se considera plantas de fase mínima descritas mediante:

$$A(q^{-1})y(k) = q^{-d}B(q^{-1})$$

Son conocidos los coeficientes a_1 , b_1 y el retardo d .

3. Se asignan los coeficientes del polinomio $Cr(q^{-1})$. Estos coeficientes deberán ser tales que $Cr(q^{-1})$ sea un polinomio normalizado cuyos ceros estén dentro del círculo unitario.
4. Se calculan los coeficientes de los polinomios $S(q^{-1})$ y $R(q^{-1})$ resolviendo secuencialmente la identidad polinomial:

$$Cr(q^{-1}) = A(q^{-1})S(q^{-1}) + q^{-d}R(q^{-1})$$

Se calculan además los coeficientes del polinomio $Bs(q^{-1})$ mediante:

$$Bs(q^{-1}) = B(q^{-1})S(q^{-1}) - b_0$$

5. Se determina $u(k)$ mediante la ley de control dada por:

$$u(k) = (1/b_0)[Cr(q^{-1})yM(k) - R(q^{-1})y(k) - Bs(q^{-1})u(k)]$$

En el capítulo V se detalla el contenido de cada uno de estos pasos globales.

2.5 CARACTERISTICAS

El esquema de control presentado en la FIG. 2.3 puede ser aplicado a sistemas de fase mínima dado que los ceros de la planta son compensados. Cuando se trabaja con sistemas de

fase no mínima no se puede aplicar las técnicas descritas en esta sección debido a que si se intenta compensar ceros fuera del círculo unitario la entrada de control diverge. Mas aún, si se intenta compensar ceros dentro del círculo pero muy cerca de este y/o a la izquierda del plano z , se obtienen efectos indeseables como entradas de control muy grandes, o que oscilan.

En la FIG. 2.4 se presenta un diagrama equivalente que representa las características del control dadas en [2.6].

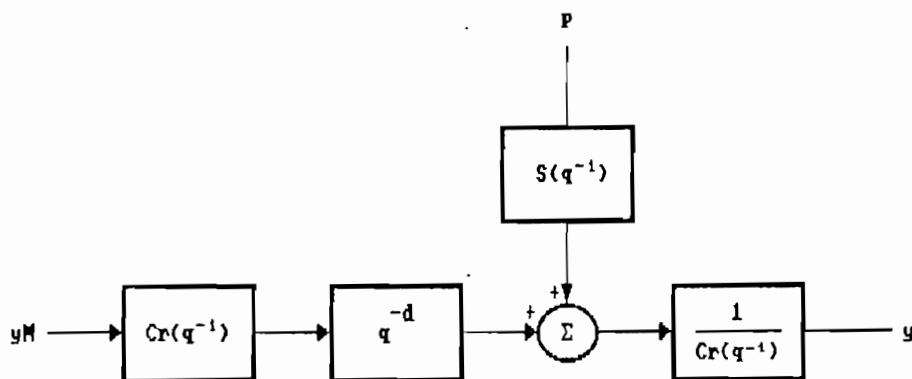


FIG. 2.4 CARACTERÍSTICAS DEL CONTROL LINEAL

Se puede observar que los ceros del sistema se han compensado y que los polos están ahora dados por el polinomio $Cr(q^{-1})$. El retardo q^{-d} aparece por cuanto no puede ser eliminado. Consecuentemente, la función cumplida por el polinomio $Cr(q^{-1})$ es de mucha importancia y de su selección dependerá en gran parte el comportamiento del control. En el capítulo V se efectúa una simulación de este esquema de control lineal discreto y con algunos ejemplos se determinan sus características principales.

CAPITULO III

IDENTIFICACION PARAMETRICA

- 3.1 PRINCIPALES TECNICAS DE IDENTIFICACION

- 3.2 IDENTIFICACION PARAMETRICA DISCRETA

- 3.3 IDENTIFICACION RECURSIVA DE MINIMOS CUADRADOS

- 3.4 PROPIEDADES DE LA IRMC

CAPITULO III

IDENTIFICACION PARAMETRICA

La implementación de un esquema de control adecuado requiere la utilización de un modelo dinámico que describa satisfactoriamente el comportamiento de la planta a controlarse. Convencionalmente la obtención de tales modelos se puede realizar a través de métodos basados en el estudio de las leyes físicas que gobiernan al sistema o, alternativamente, por algún procedimiento de análisis de datos.

Los métodos correspondientes al primer caso son conocidos como métodos de modelación. Tienen como objetivo la obtención de ecuaciones que describan las respuestas dinámicas de los sistemas, requiriendo para ello el planteamiento de ecuaciones intègro-diferenciales y de variables de estado en el dominio del tiempo, además de funciones de transferencia en el dominio de la frecuencia.

Los métodos que involucran análisis de datos son

conocidos como de identificación. A partir de la medición de datos experimentales de entrada y salida se determinan los mejores valores de los parámetros de un modelo asumido para el sistema. Esta técnica es particularmente apropiada cuando la estructura física del sistema es muy compleja o de difícil interpretación.

Dado el alto índice de complejidad de los procesos industriales, hoy en día, los métodos convencionales de modelación aún en variables de estado no resultan eficientes, más si se considera que las plantas industriales deberían ser tratadas como procesos estocásticos; de allí que, sea necesario realizar previo a la acción de control una adecuada identificación de la planta.

El control adaptivo en particular conlleva inherentemente la aplicación de una etapa de identificación ya sea explícita o implícita de la planta, siendo por tanto importante el estudio de las características básicas de esta etapa.

3.1 PRINCIPALES TECNICAS DE IDENTIFICACION

La identificación de la dinámica de procesos para propósitos de control computarizado ha recibido considerable atención en años recientes, de tal forma que se han desarrollado y aplicado de acuerdo a situaciones específicas diferentes algoritmos de identificación.

Las principales técnicas de identificación aplicables al control de sistemas pueden ser agrupadas como paramétricas y no paramétricas.

3.1.1 Técnicas de identificación no paramétricas

Las técnicas de identificación agrupadas bajo esta denominación se consideran a su vez desde dos puntos de vista: técnicas determinísticas y técnicas estocásticas.

Para el primero de los casos, los objetivos de identificación se logran a través del análisis de la respuesta de frecuencia ante señales de prueba sinusoidales o, alternativamente, mediante el análisis de la respuesta en el dominio del tiempo ante entradas impulso y/o funciones paso.

Análogamente, para el caso estocástico se consideran técnicas de análisis en el dominio de la frecuencia y en el dominio del tiempo, denominándose como análisis espectral y análisis de correlación respectivamente. Indistintamente las señales de prueba para estos análisis se conocen como rándómicas o aleatorias.

3.1.2 Técnicas de identificación paramétrica

Las técnicas convencionales de identificación paramétrica son de dos clases: las conocidas como de mínimos

cuadrados ordinarios y las de mínimos cuadrados estocásticos. Las técnicas de mínimos cuadrados ordinarios permiten establecer estimaciones paramétricas para sistemas de tiempo discreto o la obtención de modelos provenientes del análisis de respuesta de frecuencia y respuesta impulso. Los mínimos cuadrados estocásticos se orientan hacia la obtención de modelos más generalizados considerando la presencia de ruido.

El estudio de cada una de las técnicas de identificación señaladas no compromete los propósitos de este trabajo y está fuera del alcance del mismo. Una información detallada al respecto se puede encontrar en la tesis (FIE-EPN) correspondiente a "Identificación paramétrica discreta"⁴.

3.2 IDENTIFICACION PARAMETRICA DISCRETA

En la identificación paramétrica de sistemas se propone un modelo ya que se supone desconocida la estructura de tales sistemas. Considérese un modelo continuo en el tiempo dado por la función de transferencia siguiente:

$$G(s) = \frac{k(s-z_1)(s-z_2)\dots\dots(s-z_nB)}{(s-p_1)(s-p_2)\dots\dots(s-p_nA)} \quad [3.1]$$

o un modelo generalizado en base a variables de estado dado por :

[4] RACINES, Sylvia; Identificación Paramétrica Discreta Tesis FIE-EPN, 1988.

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

$$y = Cx + Du$$

donde x es el vector de estado, u es la entrada e y es la salida.

Los modelos así considerados se conocen como descripción paramétrica del sistema ya que quedarían completamente definidos al determinar un número finito de parámetros correspondientes, respectivamente, a los valores de los polos y ceros de la función de transferencia o a los elementos de las matrices A , B , C y D .

En la identificación paramétrica se observan y se miden las secuencias de datos de entrada y salida $\{u_0, u_1, \dots, u_n\}$ y $\{y_0, y_1, \dots, y_n\}$ y en base a estas mediciones se procura establecer un modelo que con buena aproximación represente estos datos.



FIG. 3.1 SISTEMA DISCRETO DE DATOS MUESTREADOS

Como puede observarse en la FIG. 3.1, la entrada y la salida se muestrean cada período T , proporcionando las secuencias $\{u(kT)\}$ y $\{y(kT)\}$. El período de muestreo T es usualmente omitido cuando se expresan las variables al

instante $t = kT$. Así, $x(t) = x(kT) = x(k)$. Para efectos de identificación y control en un computador digital es conveniente la utilización de un modelo discreto del sistema que, adicionalmente, permita un procesamiento adecuado. Para ello los modelos dinámicos discretos se expresan mediante una ecuación de diferencias o su respectiva función de transferencia discreta en base a la transformada Z:

$$G(z^{-1}) = \frac{b_0 + b_1z^{-1} + b_2z^{-2} + \dots + b_nBz^{-nB}}{1 + a_1z^{-1} + a_2z^{-2} + \dots + a_nAz^{-nA}} \quad [3.3]$$

Los parámetros $a_1, a_2, \dots, a_nA, b_0, b_1, \dots, b_nB$ son los que normalmente se deberán identificar o estimar adecuadamente.

Una descripción equivalente constituye la propuesta en el capítulo precedente mediante la ecuación de diferencias que aquí se escribe nuevamente:

$$\dot{y}(k) + a_1y(k-1) + \dots + a_nAy(k-nA) = b_0u(k-d) + b_1u(k-d-1) + \dots + b_nBu(k-d-nB) + p(k) \quad [3.4]$$

Este método tiene la ventaja de que se expresa o se actualiza un nuevo valor de salida en base a los valores anteriores de la salida y la entrada, lo cual permite desarrollar algoritmos ágiles y relativamente sencillos para la identificación de los parámetros.

Generalmente en los modelos discretos existen d

retardos entre la entrada y la salida debidos al proceso de discretización o a la dinámica misma de la planta, en la cual la salida no responde inmediatamente a la aplicación de la entrada .

La ecuación [3.4] puede reescribirse de la manera siguiente:

$$y(k) = -a_1y(k-1) - a_2y(k-2) - \dots - a_nAy(k-nA) + \\ + b_0u(k-d) + \dots + b_nBu(k-d-nB) + p(k) \quad [3.5]$$

Haciendo,

$$\theta^T = [-a_1 \quad -a_2 \quad \dots \quad -a_nA \quad b_0 \quad b_1 \quad \dots \quad b_nB]$$

y,

$$\phi(k-1) = [y(k-1) \quad \dots \quad y(k-nA) \quad u(k-1) \quad \dots \quad u(k-nB)]^T$$

donde T como exponente significa "transpuesta de", la ecuación [3.5] se transforma en,

$$y(k) = \theta^T\phi(k-1) + p(k) \quad [3.6]$$

A partir de este modelo se desarrollará un algoritmo de identificación de sus parámetros mediante la técnica de mínimos cuadrados recursivos y en el capítulo IV se establecerán las relaciones existentes entre éstos parámetros y los correspondientes a un controlador adaptivo.

En identificación paramétrica, para dar una idea de

cuán bueno es un conjunto de parámetros estimados con respecto a otro o cuán bien un modelo se ajusta a datos experimentales se introduce, generalmente, un criterio. Este criterio puede ser en base, por ejemplo, a la suma de los cuadrados de las diferencias de las salidas del proceso y del modelo, o cualquier criterio relacionado con una aplicación específica.

Se pueden distinguir dos clases de algoritmos de identificación: en línea y fuera de línea.

En el caso fuera de línea el análisis se hace con toda la información disponible y prácticamente sin restricción de tiempo. De otra manera, en el caso en línea hay disponible información secuencial requiriéndose que los parámetros sean actualizados dentro de un tiempo limitado, dado por el período de muestreo; entonces, en muchas aplicaciones es conveniente usar algoritmos relativamente simples o hacer el período de muestreo más largo para lograr una satisfactoria identificación en línea.

3.3 IDENTIFICACION RECURSIVA DE MINIMOS CUADRADOS IRMC

En esta sección se presenta el método de identificación de mínimos cuadrados en forma concisa, así como la versión recursiva más utilizada. Un análisis riguroso de este método se puede encontrar en la tesis (FIE-EPN) correspondiente a "Mínimos Cuadrado Generalizado"⁵.

[5] CORDERO, Patricio; Mínimos Cuadrados Generalizado
Tesis FIE-EPN, 1.985.

La técnica de estimación de parámetros por mínimos cuadrados básicamente trata del problema de minimización de la suma de los cuadrados de las diferencias entre los valores observados y computados.

De acuerdo con la sección precedente, se considera un sistema lineal discreto monovariable de tiempo invariante y cuya salida se puede expresar en la forma dada por [3.6], es decir,

$$y(k) = \theta^T \phi(k-1) + p(k)$$

θ representa al vector de parámetros

$\phi(k-1)$ es un vector de variables medibles

$p(k)$ es una perturbación

Sin pérdida de generalidad y por así convenir para la simplificación en el desarrollo del algoritmo se considera un retardo del sistema igual a 1 ($d=1$).

Al decir que el sistema es lineal hay que notar que implícitamente se está haciendo la hipótesis de que una variable del sistema se puede expresar como una combinación lineal de las demás variables.

El objetivo fundamental de esta técnica es la obtención de un vector de parámetros estimados, a representarse mediante $\hat{\theta}(k)$, utilizando para ello las mediciones de entrada y salida disponibles hasta el instante k .

El estimado $\hat{\theta}(k)$ será obtenido de forma tal que se minimice el criterio de error cuadrático con factor de olvido siguiente:

$$J[\hat{\theta}(k)] = \sum_{i=1}^k \left(\prod_{j=i}^k \alpha_j \right) [y(i) - \hat{\theta}^T(k)\phi(i-1)]^2 \quad 0 < \alpha_j \leq 1 \quad [3.7]$$

El error de estimación en el instante k vendrá dado por:

$$e(k) = y(k) - \hat{\theta}^T(k)\phi(k-1) \quad [3.8]$$

Nótese que este error se pondera en [3.7] por el factor α_k en el instante k ; en el instante $k-1$ se pondera por $\alpha_k\alpha_{k-1}$ y así sucesivamente. Si se toma $\alpha_i=1$ para todo i , todos los errores de estimación se ponderan igual, pero si $\alpha_i < 1$ los errores iniciales se ponderarán con coeficientes menores.

El gradiente de J con respecto a $\hat{\theta}(k)$ viene dado por:

$$J_{\hat{\theta}(k)} = 2 \sum_{i=1}^k \left(\prod_{j=i}^k \alpha_j \right) [y(i) - \hat{\theta}^T(k)\phi(i-1)] [-\phi(i-1)] \quad [3.9]$$

y el hessiano es:

$$J_{\hat{\theta}(k)\hat{\theta}(k)} = 2 \sum_{i=1}^k \left(\prod_{j=i}^k \alpha_j \right) \phi(i-1)\phi^T(i-1) \geq 0 \quad [3.10]$$

La expresión obtenida en [3.10] constituye una matriz semidefinida positiva, lo cual implica que igualando a cero la ecuación [3.9] se obtendrá un mínimo.

Resolviendo la ecuación así planteada y considerando que:

$$\hat{y}(k)\phi(i-1) = [\hat{\theta}^T(k)\phi(i-1)]\phi(i-1) = \phi(i-1)[\phi^T(i-1)\hat{\theta}(k)] \quad [3.11]$$

se obtiene,

$$\hat{\theta}(k) = \left[\sum_{i=1}^k \left(\sum_{j=i}^k \alpha_j \right) \phi(i-1)\phi^T(i-1) \right]^{-1} \sum_{i=1}^k \left(\sum_{j=i}^k \alpha_j \right) y(i)\phi(i-1) \quad [3.12]$$

Esta expresión permite calcular el estimado $\theta(k)$ si previamente se conocen todas las mediciones precedentes al tiempo k , en lo que consistiría una estimación fuera de línea (OFF-LINE).

Existen versiones de este algoritmo que permiten, como se indicó anteriormente, ir actualizando los estimados de los parámetros con cada medición. Estas versiones se denominan recursivas o en línea (ON-LINE). A diferencia de la versión fuera de línea las versiones recursivas no necesitan tener todos los datos de las mediciones en un instante dado, sino únicamente la medición en ese instante. Esto permite un ahorro considerable en memoria cuando se implementan en computadores digitales. En ambos casos se minimiza el mismo criterio de la ecuación [3.7] pero en la recursiva los estimados se aproximan paso a paso a la solución fuera de línea de la ecuación [3.12]. Por estas razones las versiones recursivas son más adecuadas para formar parte de un control de tipo adaptivo.

A partir de la ecuación dada en [3.12] se obtendrá a

continuación una de las versiones recursivas más utilizadas. Se define la matriz a invertir en la ecuación [3.12] como:

$$F^{-1}(k+1) = \sum_{i=1}^k \sum_{j=i}^k (\pi \alpha_j) \phi(i-1) \phi^T(i-1) \quad [3.13]$$

expresión que equivalentemente puede ser escrita como,

$$F^{-1}(k+1) = \alpha_k [F^{-1}(k) + \phi(k-1) \phi^T(k-1)] \quad [3.14]$$

reemplazando [3.13] en [3.12] se tiene,

$$\hat{\theta}(k) = F(k+1) \sum_{i=1}^k \sum_{j=i}^k (\pi \alpha_j) y(i) \phi(i-1) \quad [3.15]$$

equivalentemente:

$$\hat{\theta}(k) = F(k+1) [\alpha_k y(k) \phi(k-1) + \sum_{i=1}^{k-1} \sum_{j=i}^k (\pi \alpha_j) y(i) \phi(i-1)] \quad [3.16]$$

reemplazando [3.11] (escrita en $k-1$ en lugar de en k) en la ecuación [3.16] se obtiene,

$$\hat{\theta}(k) = F(k+1) [\alpha_k y(k) \phi(k-1) + \sum_{i=1}^{k-1} \sum_{j=i}^k (\pi \alpha_j) \phi(i-1) \phi^T(i-1) \hat{\theta}(k-1)] \quad [3.17]$$

Note que de [3.13] se puede establecer que:

$$F^{-1}(k+1) = \sum_{i=1}^{k-1} \sum_{j=i}^k (\pi \alpha_j) \phi(i-1) \phi^T(i-1) + \alpha_k \phi(k-1) \phi^T(k-1)$$

de donde:

$$\sum_{i=1}^{k-1} (\pi \alpha_j) \phi(i-1) \phi^T(i-1) = F^{-1}(k+1) - \alpha_k \phi(k-1) \phi^T(k-1) \quad [3.18]$$

reemplazando [3.18] en [3.17]

$$\hat{\theta}(k) = F(k+1) \{ \alpha_k y(k) \phi(k-1) + [F^{-1}(k+1) - \alpha_k \phi(k-1) \phi^T(k-1)] \hat{\theta}(k-1) \}$$

operando internamente en esta expresión y recordando que

$$FF^{-1} = I :$$

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \alpha_k F(k+1) \phi(k-1) [y(k) - \phi^T(k-1) \hat{\theta}(k-1)] \quad [3.19]$$

Note que,

$$\phi^T(k-1) \hat{\theta}(k-1) = \hat{\theta}^T(k-1) \phi(k-1)$$

reemplazando en [3.19] finalmente se tiene,

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \alpha_k F(k+1) \phi(k-1) [y(k) - \hat{\theta}^T(k-1) \phi(k-1)] \quad [3.20]$$

La determinación de $F(k+1)$ se la efectúa a partir de la ecuación [3.14] mediante la aplicación del siguiente lema de inversión de matrices:

Sea A una matriz de $n \times n$ y sean B y C n -vectores; se supone que:

A y $(A + BC^T)$ son matrices no singulares, y que

$$(1 + C^T A^{-1} B) = 0$$

entonces:

$$(A + BC^T)^{-1} = A^{-1} - \frac{A^{-1}BC^T A^{-1}}{1 + C^T A^{-1} B}$$

Haciendo en [3.14]:

$$F^{-1}(k) = A$$

$$\phi(k-1) = B$$

$$\phi^T(k-1) = C^T$$

se obtiene:

$$F(k+1) = \frac{1}{\alpha_k} \left[F(k) - \frac{F(k)\phi(k-1)\phi^T(k-1)F(k)}{1 + \phi^T(k-1)F(k)\phi(k-1)} \right] \quad [3.21]$$

Las expresiones [3.20] y [3.21] constituyen el algoritmo de identificación recursiva de mínimos cuadrados IRMC con factor de olvido que será implementado en el esquema de control adaptivo.

3.4 PROPIEDADES DE LA IRMC

En la presente sección se efectúa un análisis de la convergencia del algoritmo de identificación recursiva de mínimos cuadrados con factor de olvido ($\alpha_k < 1$).

El algoritmo de identificación para este caso se escribe, de acuerdo a lo establecido anteriormente, como:

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \alpha_k F(k+1) \phi(k-1) [y(k) - \hat{\theta}^T(k-1) \phi(k-1)]$$

$$F(k+1) = \frac{1}{\alpha_k} \left[F(k) - \frac{F(k) \phi(k-1) \phi^T(k-1) F(k)}{1 + \phi^T(k-1) F(k) \phi(k-1)} \right]$$

Para α_k se establecen las condiciones:

$$0 < \delta \leq \alpha_k \leq 1 - \epsilon \quad [3.22]$$

δ y ϵ son números positivos arbitrariamente pequeños pero distintos de cero, es decir:

$$0 < \delta \ll 1 \quad \text{y} \quad 0 < \epsilon \ll 1 \quad [3.23]$$

Para este algoritmo se demostrará que:

$$y(k) - \hat{y}(k) \xrightarrow[k \rightarrow \infty]{} 0 \quad [3.24]$$

o equivalentemente que,

$$y(k) - \hat{\theta}^T(k) \phi(k-1) \xrightarrow[k \rightarrow \infty]{} 0 \quad [3.25]$$

Se define la distancia paramétrica como:

$$\tilde{\theta}(k) = \theta - \hat{\theta}(k) \quad [3.26]$$

Por tanto la ecuación [3.24] a demostrarse puede escribirse como,

$$\theta^T \phi(k-1) - \hat{\theta}^T(k) \phi(k-1) \xrightarrow[k \rightarrow \infty]{} 0$$

$$[\theta^T - \hat{\theta}^T(k)]\phi(k-1) \xrightarrow[k \rightarrow \infty]{} 0$$

$$\tilde{\theta}^T(k)\phi(k-1) \xrightarrow[k \rightarrow \infty]{} 0 \quad [3.27]$$

De las ecuaciones [3.6], [3.20] y [3.26] se obtiene:

$$\tilde{\theta}(k) = [I - \alpha_k F(k+1)\phi(k-1)\phi^T(k-1)]\tilde{\theta}(k-1) \quad [3.28]$$

Multiplicando la ecuación [3.14] por $F(k+1)$ y combinándola con la [3.28] se tiene,

$$F^{-1}(k+1)\tilde{\theta}(k) = \alpha_k F^{-1}(k)\tilde{\theta}(k-1) \quad [3.29]$$

La demostración requiere la selección de una función de Lyapunov adecuada. Sea dicha función la propuesta a continuación:

$$V(k+1) = \tilde{\theta}^T(k)F^{-1}(k+1)\tilde{\theta}(k) > 0 \quad [3.30]$$

Entonces, utilizando la relación [3.29],

$$V(k+1) - \alpha_k V(k) = \tilde{\theta}^T(k)F^{-1}(k+1)\tilde{\theta}(k) - \alpha_k \tilde{\theta}^T(k-1)F^{-1}(k)\tilde{\theta}(k-1)$$

$$V(k+1) - \alpha_k V(k) = \alpha_k [\tilde{\theta}^T(k) - \tilde{\theta}^T(k-1)]^T F^{-1}(k)\tilde{\theta}(k-1) \quad [3.31]$$

Combinando [3.28] con [3.31],

$$V(k+1) - \alpha_k V(k) = -\alpha_k^2 [\phi^T(k-1)\tilde{\theta}(k-1)] [F(k+1)\phi(k-1)]^T F^{-1}(k)\tilde{\theta}(k-1)$$

De [3.21] se tiene que:

$$F(k+1)\phi(k-1) = \frac{1}{\alpha_k} \frac{F(k)\phi(k-1)}{1 + \phi^T(k-1)F(k)\phi(k-1)}$$

por lo tanto,

$$V(k+1) - \alpha_k V(k) = - \frac{\alpha_k [\theta^T(k-1)\phi(k-1)]^2}{1 + \phi^T(k-1)F(k)\phi(k-1)} \leq 0$$

de donde,

$$V(k+1) \leq \alpha_k V(k) \leq \alpha_k \alpha_{k-1} V(k-1) \leq \dots \leq \prod_{j=1}^k \alpha_j V(1)$$

Según [3.22] se tendría que,

$$V(k+1) \leq (1 - \epsilon)^k V(1) \quad [3.32]$$

de donde se concluye que:

$$V(k+1) = \tilde{\theta}^T(k) F^{-1}(k+1) \tilde{\theta}(k) \xrightarrow[k \rightarrow \infty]{} 0 \quad [3.33]$$

Por otro lado de la ecuación [3.14] se tiene,

$$\tilde{\theta}^T(k) F^{-1}(k+1) \tilde{\theta}(k) = \alpha_k \{ \tilde{\theta}^T(k) F^{-1}(k) \tilde{\theta}(k) + [\tilde{\theta}^T(k) \phi(k-1)]^2 \} \quad [3.34]$$

Ahora, dado que $\alpha_k \geq \delta > 0$ de [3.33] y [3.34] se concluye finalmente que:

$$\tilde{\theta}^T(k) \phi(k-1) \xrightarrow[k \rightarrow \infty]{} 0$$

CAPITULO IV

CONTROL ADAPTIVO CON MODELO DE REFERENCIA

4.1 PRELIMINARES

4.2 OBJETIVOS DEL CONTROL

4.3 LEY DE CONTROL ADAPTIVO PARA EL SISTEMA

4.4 ESTRUCTURA E IMPLEMENTACION DEL C. ADAPTIVO

4.5 ALGORITMO DE CONTROL ADAPTIVO MRAC

CAPITULO IV

CONTROL ADAPTIVO CON MODELO DE REFERENCIA

4.1 PRELIMINARES

El primer diseño de Control Adaptivo con Modelo de Referencia (MRAC) para sistemas discretos SISO (single-input, single output) fue propuesto en 1.977⁵. Algunas de las características básicas de este diseño fueron:

1. Se empleó un concepto de amplificación de error con la finalidad de contrarrestar los problemas relacionados con los retardos inherentes a los sistemas discretos.
2. El modelo de referencia empleado fue del tipo "paralelo".
3. Se utilizó un algoritmo de adaptación con ganancias de adaptación constantes.

IONESCU, I. and MONOPOLI R.; Discrete model reference adaptive control with an augmented error signal, 1977, AUTOMATICA 13.

4. En el algoritmo de adaptación se emplearon variables filtradas.
5. Se asumió conocido el signo correspondiente al primer coeficiente del polinomio del numerador de la planta.
6. El diseño se realizó desde un punto de vista de estabilidad utilizando funciones de Lyapunov.

Posteriormente se propusieron otros diseños igualmente basados en un punto de vista de estabilidad, los mismos que presentaban diferentes características y eran más apropiados para ciertos problemas específicos de control. Así, la configuración paralela propuesta inicialmente fue diseñada para cumplir con propósitos de seguimiento, no siendo consideradas sus propiedades con respecto a regulación. Por otro lado, otras configuraciones (serie, serie-paralelo por ejemplo) pueden ser consideradas más apropiadas para efectos de regulación.

Otros diseños consideraron, en lugar de la amplificación de error y variables filtradas, la utilización de un modelo de referencia con salida a priori y/o posteriori y un compensador lineal actuando sobre la señal de error existente entre planta y modelo.

Las simulaciones que se han realizado para investigar el comportamiento de los distintos diseños han demostrado que los algoritmos con ganancias de adaptación constantes a pesar

de su simplicidad y de sus propiedades en tiempo real tienen dos desventajas principales:

- a) no está claro como escoger los mejores valores de las ganancias de adaptación.
- b) el funcionamiento de los sistemas adaptivos es frecuentemente muy sensible con respecto a los valores de las ganancias de adaptación.

Por lo tanto, es de mucha utilidad el desarrollo de algoritmos de adaptación con ganancias variables en el tiempo e inspirados en los algoritmos usados en identificación recursiva, los cuales eliminan la necesidad de escoger las ganancias de adaptación y proporcionan un mejor funcionamiento. No obstante, la utilización de estos algoritmos trae complicaciones notables en los análisis correspondientes a estabilidad.

4.1.1 Tipos de MRAC

No es posible presentar una clasificación de los diseños de MRAC para sistemas discretos en términos absolutos ya que las consideraciones a tomarse en cuenta normalmente podrían resultar subjetivas por depender de un cierto punto de vista desde el cual son analizadas. No obstante de ello, y desde un punto de vista de estabilidad se acepta una clasificación en los siguientes términos:

MRAC para	[seguimiento
		regulación
		seguimiento y regulación (*)
MRAC con	[modelo de referencia paralelo
		modelo de referencia serie (*)
		modelo de referencia serie-paralelo
MRAC usando	[modelo de referencia a priori (*)
		modelo de referencia a posteriori
		modelo de referencia a priori y posteriori
		amplificación de error
MRAC usando	[variables filtradas
		compensador actuando sobre el error planta-
		modelo (*)
MRAC usando	[ganancias de adaptación constantes
		ganancias de adaptación variables en el
		tiempo (*)

Los trabajos de investigación concernientes a este campo se hallan empeñados en el estudio de diseños generales y unificados que abarquen todas o la mayoría de las posibilidades tanto de MRAC como también de las demás técnicas básicas de control adaptivo existentes. A partir de 1.980 se han presentado importantes contribuciones en este sentido a través de varias publicaciones técnicas especializadas en control automático.

El esquema de control adaptivo a considerarse en este trabajo participa básicamente de las características señaladas mediante (*) en la clasificación precedente.

4.1.2 Esquema básico

En el capítulo II se presentó un algoritmo de control lineal para sistemas discretos de fase mínima. Para el diseño de dicho control se requiere el conocimiento de los parámetros. Las características del controlador pueden cambiar drásticamente si los parámetros con los cuales se hizo el diseño sufren modificaciones.

El esquema básico del control adaptivo a considerarse se presenta en la FIG. 4.1. Este esquema está basado en la ley de control lineal, pero en este caso los parámetros del controlador se ajustan o actualizan usando los parámetros estimados de la planta que resultan de una identificación recursiva.

Se considera una planta lineal discreta SISO descrita mediante:

$$A(q^{-1})y(k) = q^{-d}B(q^{-1})u(k) \quad d > 0 \quad [4.1]$$

donde:

$$A(q^{-1}) = 1 + a_1q^{-1} + a_2q^{-2} + \dots + a_nq^{-n} \quad [4.2]$$

$$B(q^{-1}) = b_0 + b_1q^{-1} + b_2q^{-2} + \dots + b_nq^{-nB} \quad [4.3]$$

d representa el retardo del sistema y su valor es mayor que cero.

$u(k)$ y $y(k)$ son la entrada y salida de la planta, respectivamente.

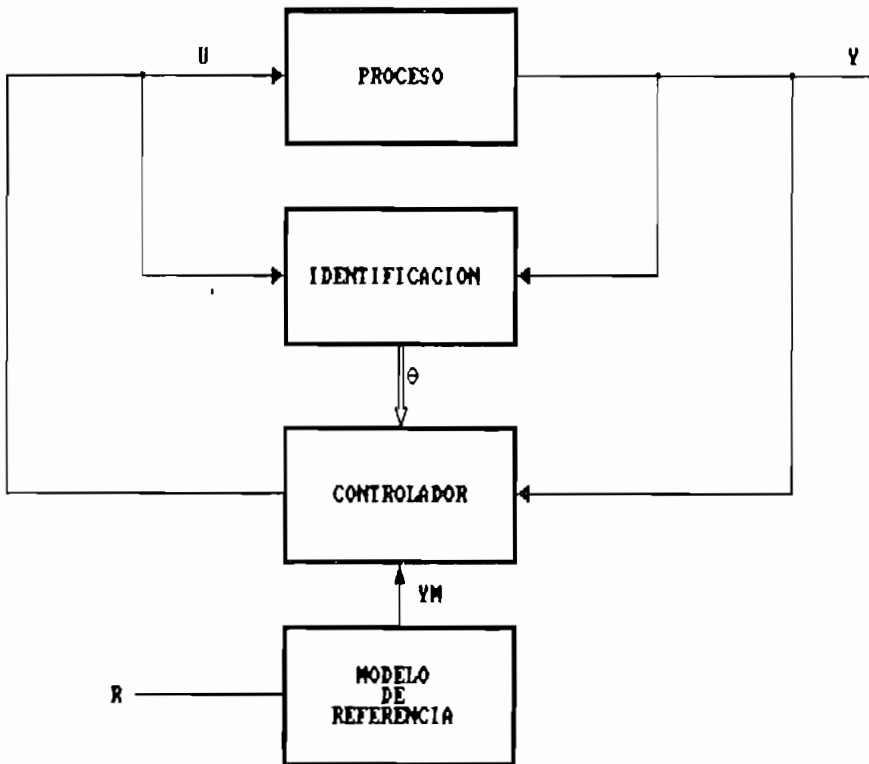


FIG. 4.1 ESQUEMA BASICO DE CONTROL ADAPTIVO

Se consideran las siguientes suposiciones:

a) el retardo d es conocido

b) n_A y n_B son conocidos

c) los ceros de $B(z^{-1})$ están en $|z| < 1$, lo cual permitirá que sean cancelados sin necesidad de recurrir a una entrada de control no acotada.

El esquema de control adaptivo a implementarse consiste por tanto en una extensión del control lineal discreto del capítulo II aplicado a plantas de fase mínima descritas mediante [4.1] y en las cuales los parámetros a_1 y b_1 de [4.2] y [4.3] son desconocidos.

El procedimiento global a seguirse viene dado en términos generales a través de los siguientes pasos: dado un proceso modelo (modelo de referencia) se define la estructura de un controlador tal que permita la igualación de la salida de la planta con la referencia para el caso de parámetros de la planta conocidos e invariantes. Entonces explícitamente se escriben las ecuaciones que describen la estructura del controlador y se nota la relación existente entre los parámetros del controlador y los de la planta. Se establecen por tanto los parámetros del controlador a partir de los parámetros de la planta. Se identifican los parámetros del controlador utilizando un método adecuado y finalmente se utilizan estos estimados en el controlador como si ellos fueran correctos (principio de la equivalencia de la certeza).

4.2. OBJETIVOS DEL CONTROL

Un problema presente desde hace algunos años en la

teoría de control automático ha sido la cuestión referente a la existencia de algoritmos de control adaptivos que reúnan características de simplicidad y convergencia.

Se desea algoritmos de control tales que para cualquier condición inicial del sistema y/o estado del algoritmo se consiga que la salida del sistema lineal dado siga asintóticamente a una secuencia de salida deseada, y alcance a ésta utilizando para tal efecto una secuencia de control acotada.

En consecuencia, la ley de control debe ser diseñada de tal manera que se obtenga un sistema estable en el cual la salida $y(k)$ siga a la referencia $y_M(k)$. Específicamente se requiere que $y(k)$ y $u(k)$ sean uniformemente acotadas en k , y que,

$$\lim_{k \rightarrow \infty} y(k) - y_M(k) = 0 \quad [4.4]$$

Se asume que la secuencia $y_M(k)$ es conocida a priori y que,

$$|y_M(k)| \leq m < \infty \text{ para todo } k \quad [4.5]$$

4.3 LEY DE CONTROL ADAPTIVO PARA EL SISTEMA

Considérese la expresión [2.15] obtenida en el capítulo II para el caso de control lineal y que aquí se escribe nuevamente:

$$B(q^{-1})S(q^{-1})u(k) + R(q^{-1})y(k) - Cr(q^{-1})yM(k) = 0 \quad [4.6]$$

donde según [2.12],

$$S(q^{-1}) = 1 + s_1q^{-1} + s_2q^{-2} + \dots + s_nsq^{-nS}$$

$$R(q^{-1}) = r_0 + r_1q^{-1} + r_2q^{-2} + \dots + r_nRq^{-nR}$$

La expresión [4.6] puede ser escrita (ver Apéndice B) como:

$$b_0u(k) + \theta_0^T \phi_0(k) - Cr(q^{-1})yM(k) = 0 \quad [4.7]$$

donde,

$$\phi_0^T(k) = [u(k-1), u(k-2), \dots, u(k-d-nB+1), y(k), \dots, y(k-nR)] \quad [4.8]$$

$$\theta_0^T = [b_0s_1+b_1, b_0s_2+b_1s_1+b_2, \dots, b_nBsa-1, r_0, r_1, \dots, r_nR] \quad [4.9]$$

A su vez [4.7] puede transformarse en,

$$\theta^T \phi(k) - Cr(q^{-1})yM(k) = 0 \quad [4.10]$$

donde,

$$\theta^T = [b_0; \theta_0^T] \quad [4.11]$$

$$\phi^T(k) = [u(k); \phi_0^T(k)]$$

es decir:

$$\phi^T(k) = [u(k), u(k-1), \dots, u(k-d-nB+1), y(k), \dots, y(k-nR)] \quad [4.12]$$

$$\theta^T = [b_0, b_0s_1+b_1, b_0s_2+b_1s_1+b_2, \dots, b_nB s_{d-1}, r_0, r_1, \dots, r_nR]$$

Note que θ^T es un vector de parámetros y $\phi^T(k)$ es un vector de variables medibles.

En consecuencia, los objetivos del control lineal podrán ser conseguidos mediante una entrada de control $u(k)$ tal que, según [4.7]:

$$u(k) = \frac{1}{b_0} [Cr(q^{-1})yM(k) - \theta_0^T \phi_0(k)] \quad [4.13]$$

Se puede observar que de la ecuación [2.1] y de la identidad polinomial [2.11] la planta se puede parametrizar como:

$$Cr(q^{-1})y(k) = A(q^{-1})S(q^{-1}) + q^{-d}R(q^{-1})y(k)$$

$$Cr(q^{-1})y(k) = q^{-d}B(q^{-1})S(q^{-1})u(k) + q^{-d}R(q^{-1})y(k) \quad [4.14]$$

Comparando esta expresión con la ley de control obtenida en [2.16] se observa que identificando la planta parametrizada como en la ecuación [4.14] se identificarán simultáneamente los parámetros del controlador.

Notar que la expresión [4.14] también puede expresarse como:

$$Cr(q^{-1})y(k) = \theta^T \phi(k-d) \quad [4.15]$$

En vista de que los coeficientes a_1 y b_1 de [4.2] y [4.3] son desconocidos es claro que los parámetros del controlador no pueden ser computados de acuerdo a lo establecido en el capítulo II. Sin embargo, si se asumen conocidos los valores de n_A , n_B y d , la estructura del controlador dada en [4.13] es también válida. Entonces, es natural el reemplazo de los parámetros desconocidos del controlador b_0 y θ_0 por parámetros ajustables $\hat{b}_0(k)$ y $\hat{\theta}_0(k)$ que serán actualizados mediante un mecanismo de adaptación.

Por lo tanto, la ley de control para el caso adaptivo viene dada por:

$$u(k) = \frac{1}{\hat{b}_0(k)} [Cr(q^{-1})yM(k) - \hat{\theta}_0^T(k)\phi_0(k)] \quad [4.16]$$

o, equivalentemente de [4.10],

$$\hat{\theta}^T(k)\phi(k) = Cr(q^{-1})yM(k) \quad [4.17]$$

donde,

$$\hat{\theta}^T(k) = [\hat{b}_0(k); \hat{\theta}_0^T(k)] \quad [4.18]$$

4.4 ESTRUCTURA E IMPLEMENTACION DEL CONTROL ADAPTIVO

El esquema de control adaptivo se estructura, de acuerdo a lo considerado anteriormente, asociando el esquema de control lineal con un mecanismo adecuado de adaptación. En la FIG. 4.2 se observa una aproximación inicial del esquema adaptivo.

El mecanismo de adaptación viene dado a través de la estimación y actualización de los parámetros del controlador. La ley de control propuesta en la sección precedente contempla los estimados $\hat{b}_0(k)$ y $\hat{\theta}_0(k)$ que, asociados según [4.18] conforman el vector de parámetros estimados $\hat{\theta}^T(k)$.

Entonces, en virtud de lo analizado en el capítulo III en relación a Identificación Paramétrica, el vector $\hat{\theta}^T(k)$ será estimado utilizando el siguiente algoritmo de identificación recursiva de mínimos cuadrados con factor de olvido:

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \alpha_k F(k+1) \phi(k-d) [C r(q^{-1}) y(k) - \hat{\theta}^T(k-1) \phi(k-d)] \quad [4.19]$$

F es la matriz adaptiva de ganancias calculada según:

$$F(k+1) = \frac{1}{\alpha_k} \left[F(k) - \frac{F(k) \phi(k-d) \phi^T(k-d) F(k)}{1 + \phi^T(k-d) F(k) \phi(k-d)} \right] \quad [4.20]$$

el valor correspondiente a α_k se denomina factor de olvido y su valor debe ser tal que:

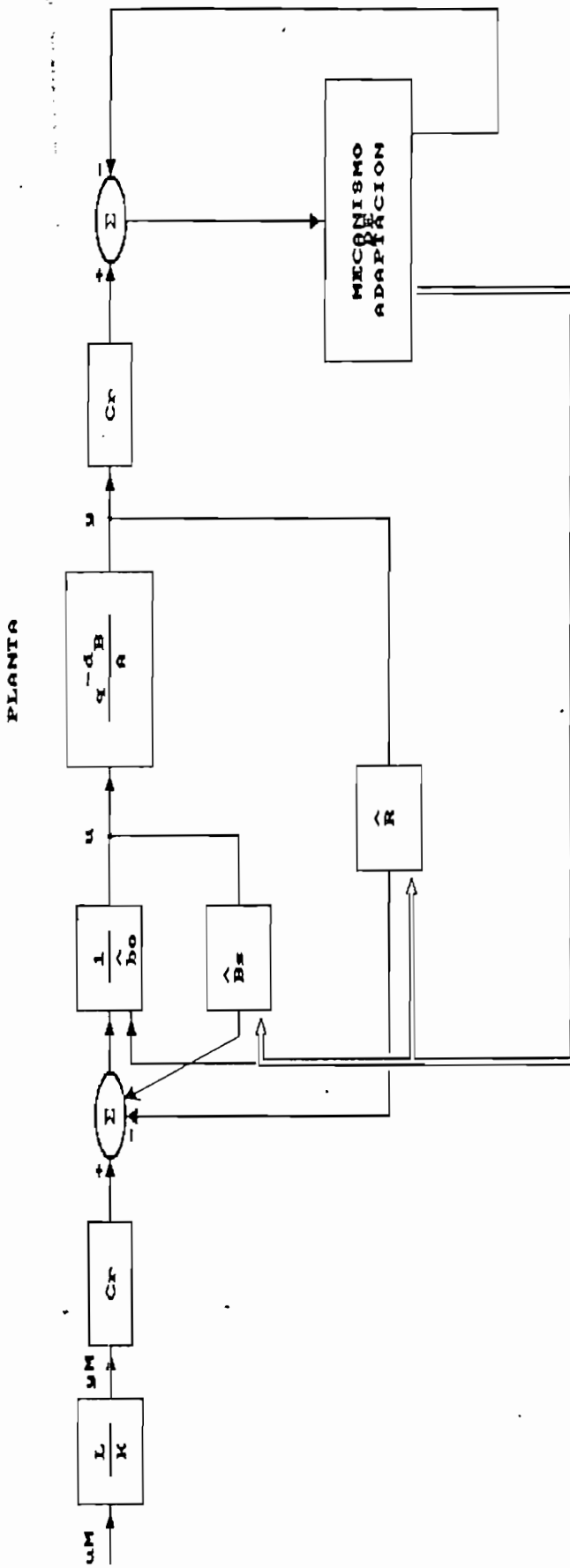


FIG. 4.2 ESQUEMA DE CONTROL ADAPTIVO

$$0 < \delta \leq \alpha_k \leq 1 - \epsilon \quad [4.21]$$

donde δ y ϵ son números positivos arbitrariamente pequeños pero distintos de cero.

$$0 < \epsilon < < 1 \quad 0 < \delta < < 1 \quad [4.22]$$

Según [4.12], el vector de parámetros θ^T y el vector de variables medibles $\phi^T(k-d)$ vienen dados por,

$$\theta^T = [b_0, b_0 s_1 + b_1, b_0 s_2 + b_1 s_1 + b_2, \dots, b_n s_{d-1}, r_0, \dots, r_n R] \quad [4.23]$$

$$\phi^T(k-d) = [u(k-d), \dots, u(k-2d-nB+1), y(k-d), \dots, y(k-d-nR)] \quad [4.24]$$

Considérese la ecuación [4.17] reescrita para el instante $k-d$

$$Cr(q^{-1})yM(k-d) = \hat{\theta}^T(k-d)\phi(k-d) \quad [4.25]$$

Restando [4.15] y [4.25] se obtiene,

$$Cr(q^{-1})[y(k) - yM(k-d)] = [\theta^T - \hat{\theta}^T(k-d)]\phi(k-d) \quad [4.26]$$

y recordando que:

$$\tilde{\theta}(k) = \theta - \hat{\theta}(k)$$

[4.26] puede ser escrita como:

$$\text{Cr}(q^{-1})[y(k) - yM(k-d)] = \tilde{\theta}^T(k-d)\phi(k-d) \quad [4.27]$$

o equivalentemente,

$$\text{Cr}(q^{-1})e(k) = \tilde{\theta}^T(k-d)\phi(k-d) \quad [4.28]$$

En el capítulo III se mostró que el algoritmo dado a través de [4.19] y [4.20] tiene la propiedad:

$$\tilde{\theta}^T(k)\phi(k-d) \xrightarrow[k \rightarrow \infty]{} 0 \quad [4.29]$$

Dado que $\tilde{\theta}(k)$ tiende a un vector constante, se puede concluir intuitivamente que,

$$\tilde{\theta}^T(k-d)\phi(k-d) \xrightarrow[k \rightarrow \infty]{} 0 \quad [4.30]$$

De las ecuaciones [4.27] y [4.30] se concluye que la salida converge a la salida deseada y es entonces acotada. Por otro lado dado que el sistema es de fase mínima, se concluye también que la entrada de control es acotada.

La FIG. 4.3 muestra el esquema de control adaptivo estructurado de acuerdo a las consideraciones anteriormente expuestas.

Lo expuesto hasta aquí sobre Control Adaptivo con Modelo de Referencia (MRAC) es resumido en el numeral 4.5 a través del establecimiento del correspondiente algoritmo de control.

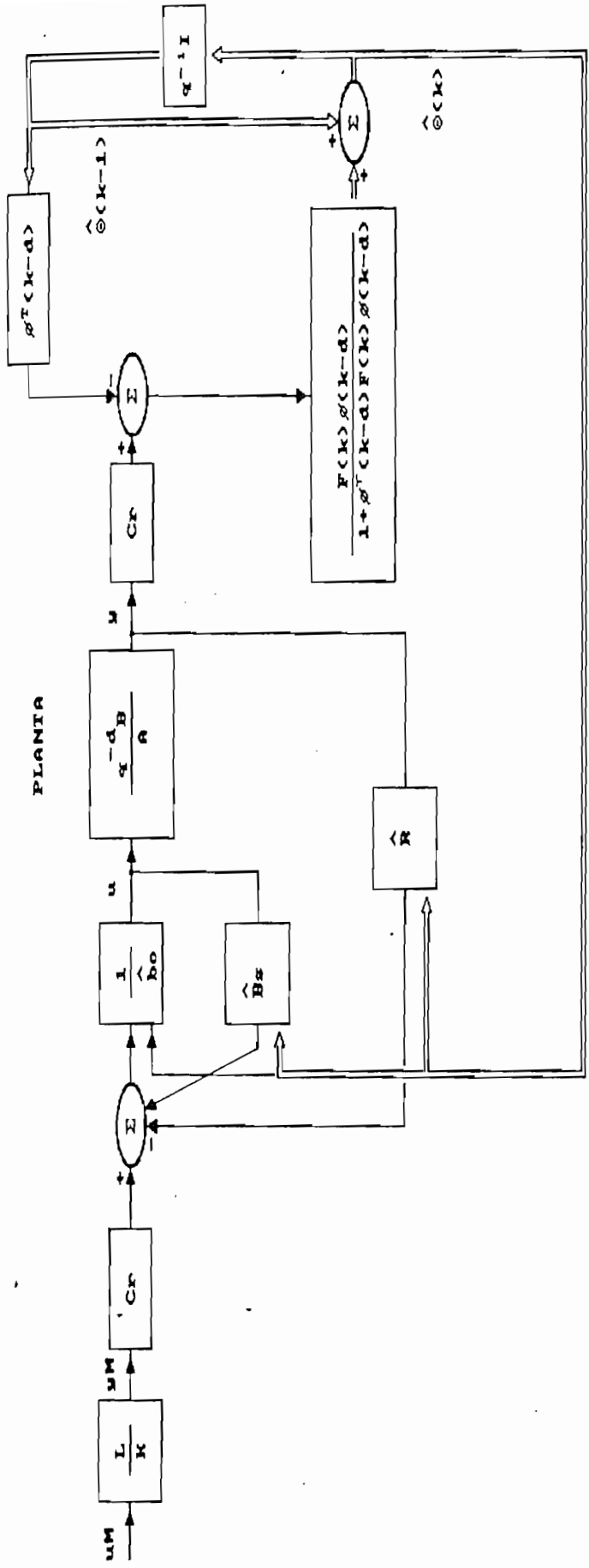


FIG. 4-3 ESQUEMA DE CONTROL ADAPTIVO

4.5 ALGORITMO DE CONTROL ADAPTIVO MRAC

La implementación del esquema de control adaptivo propuesto se realiza consecuentemente a través de los siguientes pasos globales:

1. Se determina una secuencia de referencia $\{y_M(k)\}$ utilizando para ello un modelo.
2. Se considera plantas de fase mínima descritas de la forma:

$$A(q^{-1})y(k) = q^{-d}B(q^{-1})u(k)$$

Se desconocen los coeficientes a_1, b_1

3. Se asumen conocidos los valores de n_A, n_B y d . Restrictivamente los ceros de $B(z^{-1})$ deberán estar en $|z| < 1$.
4. Se estima el vector de parámetros del controlador en cada instante k mediante:

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \alpha_k F(k+1) \phi(k-d) [C r(q^{-1})y(k) - \hat{\theta}^T(k-1) \phi(k-d)]$$

$$F(k+1) = \frac{1}{\alpha_k} \left[F(k) - \frac{F(k) \phi(k-d) \phi^T(k-d) F(k)}{1 + \phi^T(k-d) F(k) \phi(k-d)} \right]$$

$$0 < \alpha_k < 1 - \epsilon$$

$$\hat{\theta}^T(k) = [\hat{b}_0(k); \hat{\theta}_0^T(k)]$$

5. Se aplica la ley de control adaptivo dada por:

$$u(k) = \frac{1}{\hat{b}_0(k)} [Cr(q^{-1})yM(k) - \hat{\theta}_0^T(k)\phi_0(k)]$$

En el capítulo V se efectúa la simulación de este esquema de control y en ella se determina el contenido y el alcance de cada uno de estos pasos globales.

CAPITULO V

SIMULACION DIGITAL

- 5.1 ESQUEMA PARA SIMULACION DE CONTROL LINEAL
- 5.2 ESQUEMA PARA SIMULACION DE CONTROL ADAPTIVO
- 5.3 DESARROLLO DEL PROGRAMA DE SIMULACION

CAPITULO Y

SIMULACION DIGITAL

En este capítulo se efectúan las simulaciones digitales de los esquemas de control lineal y adaptivo MRAC propuestos en los capítulos precedentes. El análisis de los resultados a obtenerse mediante estas simulaciones permitirá establecer con mayor certeza el comportamiento de los sistemas sujetos a las acciones de control en consideración.

El soporte físico utilizado para procesamiento está conformado por un computador PC/XT IBM compatible operando a 10 Mhz. y con memoria RAM de 640 Kb.

El lenguaje de programación aplicado corresponde a la versión QuickBASIC 4.0 de Microsoft. La eficiencia de este compilador lo convierte en un instrumento idóneo para su utilización en trabajos como el presente.

El diseño y la programación se han estructurado tomando como punto referencial una orientación hacia el usuario

final, a efecto de lo cual se han explotado algunas de las principales disponibilidades ofrecidas tanto por el equipo como por el compilador en cuestión.

Las simulaciones de control lineal y adaptivo se desarrollan separadamente, procurando así obtener una mejor estructuración en la programación. Hacia la finalización de este trabajo se presenta el listado correspondiente a la programación así como también el manual de uso del programa.

5.1 ESQUEMA PARA LA SIMULACION DE CONTROL LINEAL

La simulación para el caso de control lineal se realiza tomando como base la siguiente secuencia directriz de procedimientos :

5.1.1 Generación de la secuencia de referencia

La secuencia de referencia $y_M(k)$ a la cual deberá ajustarse la salida $y(k)$ puede ser generada de dos formas alternativas :

- a. Asignando directamente sus valores para cada instante k , o
- b. Generando automáticamente sus valores, utilizando para ello una función de transferencia.

La primera forma implica una asignación manual de cada

valor correspondiente a $y_M(k)$, procedimiento a utilizarse en el supuesto caso de tener una secuencia de referencia especificada puntualmente.

La segunda forma es más general ya que permite la obtención secuencial de $y_M(k)$ mediante la disposición de una secuencia de entrada, definida como $u_M(k)$, y de una función de transferencia polinomial $GM(q^{-1})$ que las relacione; es decir :

$$GM(q^{-1}) = \frac{y_M(k)}{u_M(k)}$$

Considérese a $GM(q^{-1})$ dada por :

$$GM(q^{-1}) = \frac{L(q^{-1})}{K(q^{-1})}$$

donde,

$$L(q^{-1}) = l_0 + l_1q^{-1} + l_2q^{-2} + \dots + l_nLq^{-nL}$$

$$K(q^{-1}) = 1 + k_1q^{-1} + k_2q^{-2} + \dots + k_nKq^{-nK}$$

Entonces, $y_M(k)$ se determinará mediante :

$$y_M(k) = GM(q^{-1})u_M(k)$$

$$y_M(k) = \frac{L(q^{-1})}{K(q^{-1})} u_M(k)$$

Por lo tanto, esta forma de generación implica la asignación de los coeficientes l_1 y k_1 y de una secuencia $u_M(k)$ que será de tipo escalón y con posibles variaciones en sus niveles en el transcurso del tiempo.

Simultáneamente con la generación de los puntos de la secuencia de referencia se procede a su almacenamiento para la aplicación en las acciones de control respectivas.

5.1.2 Asignación de la planta del sistema a controlar

Para ello se considera la representación polinomial establecida con anterioridad :

$$A(q^{-1})y(k) = q^{-d}B(q^{-1})u(k) \quad d = 0$$

con,

$$A(q^{-1}) = 1 + a_1q^{-1} + a_2q^{-2} + \dots + a_nAq^{-nA}$$

$$B(q^{-1}) = b_0 + b_1q^{-1} + b_2q^{-2} + \dots + b_nBq^{-nB}$$

Se ingresan los coeficientes a_1 y b_1 respectivamente y el valor d correspondiente al retardo . Se verifica el cumplimiento de la condición de fase mínima en la planta.

5.1.3 Selección del polinomio de control $C_r(q^{-1})$

El esquema de control lineal involucra la selección de

un polinomio de control $Cr(q^{-1})$ normalizado y con sus ceros dentro del círculo unitario. Este polinomio fue representado mediante:

$$Cr(q^{-1}) = 1 + c_1q^{-1} + c_2q^{-2} + \dots + c_nq^{-n}$$

Se deberá por tanto proceder a la asignación de los coeficientes c_i respectivos.

5.1.4 Cálculo de los polinomios $S(q^{-1})$ y $R(q^{-1})$

De acuerdo a lo establecido en la sección 2.3.1 los objetivos de control se consiguen mediante el cumplimiento de la identidad polinomial [2.11], la misma que aquí nuevamente se la escribe:

$$Cr(q^{-1}) = A(q^{-1})S(q^{-1}) + q^{-d}R(q^{-1})$$

Habiéndose ya determinado los polinomios Cr y A , y siendo conocido el retardo d , es posible el cálculo de los polinomios S y R . Para ello se estableció también que :

$$n_S = d-1$$

$$n_R = \max(n_A-1, n_C-d)$$

Consecuentemente, los coeficientes de los polinomios S y R se calculan a partir de la igualdad de las potencias

q^{-1} correspondientes a ambos lados de la identidad en cuestión. Según lo expuesto en 2.3.1 la resolución es secuencial.

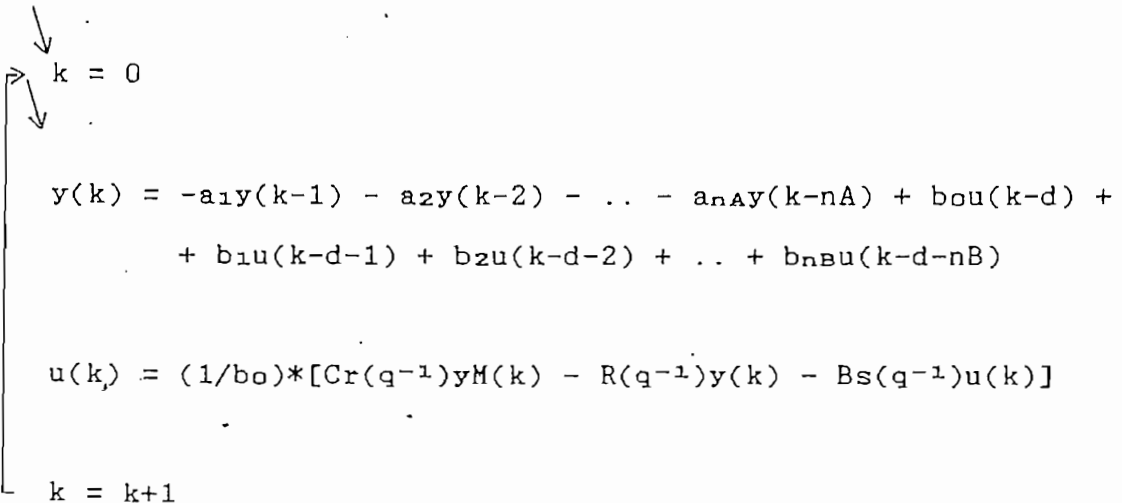
5.1.5 Cálculo del polinomio $Bs(q^{-1})$

Mediante la expresión :

$$Bs(q^{-1}) = B(q^{-1})S(q^{-1}) - b_0$$

5.1.6 Determinación de $y(k)$ y $u(k)$

Los procedimientos previos permiten el cálculo secuencial de las señales de control y salida del sistema como a continuación se indica:



$y(k) = -a_1y(k-1) - a_2y(k-2) - \dots - a_ny(k-n_A) + b_0u(k-d) +$
 $+ b_1u(k-d-1) + b_2u(k-d-2) + \dots + b_nu(k-d-n_B)$

$u(k) = (1/b_0) * [Cr(q^{-1})yM(k) - R(q^{-1})y(k) - Bs(q^{-1})u(k)]$

$k = k+1$

Los valores obtenidos para $y(k)$ y $u(k)$ son almacenados para su posterior presentación.

5.2 ESQUEMA PARA LA SIMULACION DE CONTROL ADAPTIVO MRAC

Una vez que en secciones anteriores se han propuesto y analizado consideraciones teóricas sobre el esquema MRAC en cuestión, en el presente capítulo se implementa un mecanismo de simulación digital que permite determinar en forma práctica y en base a la observación de resultados, el comportamiento de dicho esquema.

El mecanismo de simulación, implementado en función de una orientación demostrativa, está dado en los siguientes términos:

- Se considera una planta inicial de fase mínima, la misma que para efectos de aplicación del control se puede suponer conocida o desconocida, y se establece cierta dinámica de regulación mediante la asignación de $Cr(q^{-1})$. Si la planta inicial se considera conocida, con sus datos se calculan los parámetros respectivos del controlador (como en el caso de control lineal) y a partir de $k=0$ se continúa con el funcionamiento del sistema. Hay que notar que para este caso, si bien se utiliza desde $k=0$ el esquema de control adaptivo, los parámetros estimados del controlador para cada nueva iteración serán los mismos por cuanto se parte ya con los parámetros reales. Este caso propuesto en tales términos permitirá apreciar cómo la salida de una planta conocida y de parámetros fijos sigue a una referencia con cierta dinámica arbitraria de regulación.

Si la planta inicial se considera desconocida el esquema adaptivo se encargará de estimar recursivamente los parámetros del controlador con el fin de conseguir la igualación de la salida con la referencia propuesta.

- A continuación, el mecanismo de simulación consiste en la alteración arbitraria de la planta (cumpliendo con el condicionamiento de fase mínima) en cualquier instante k . Similarmente el algoritmo de control adaptivo se encargará entonces, a partir de dicho instante k , de la estimación paso a paso, recursiva, de los nuevos parámetros del controlador hasta conseguir que la salida del sistema en las nuevas condiciones retome la referencia inicial. Será apreciable la magnitud y duración del estado de transición que se presenta hasta conseguir dicho objetivo.

Este mecanismo de simulación es implementado en base a la siguiente secuencia directriz de procedimientos:

5.2.1 Generación de la secuencia de referencia

Se realiza en los mismos términos que los propuestos anteriormente para el caso de control lineal.

5.2.2 Asignación de la planta inicial del sistema

Utilizando la representación polinomial ya establecida,

la planta inicial es definida mediante:

$$A'(q^{-1})y(k) = q^{-d}B'(q^{-1})u(k) \quad d' \neq 0$$

con:

$$A'(q^{-1}) = 1 + a'_{1}q^{-1} + a'_{2}q^{-2} + \dots + a'_{nA}q^{-nA}$$

$$B'(q^{-1}) = b'_{0} + b'_{1}q^{-1} + b'_{2}q^{-2} + \dots + b'_{nB}q^{-nB}$$

Se ingresan los coeficientes a'_{1} , b'_{1} y d' y se verifica la condición de fase mínima.

5.2.3 Selección del instante de modificación

Se asigna cierto instante TMOD correspondiente al valor de k en el cual la planta original experimenta modificaciones. La selección del instante TMOD es arbitraria dentro del rango demostrativo de valores establecido para la simulación.

5.2.4 Asignación de la planta modificada

A partir de TMOD la planta del sistema adquiere otras características, pudiendo experimentar cambios que incluyan la variación en los valores de los coeficientes y/o en el grado de los polinomios representativos. La nueva planta viene dada por:

$$A(q^{-1})y(k) = q^{-d}B(q^{-1})u(k) \quad d = 0$$

donde los polinomios A y B corresponden a:

$$A(q^{-1}) = 1 + a_1q^{-1} + a_2q^{-2} + \dots + a_nAq^{-nA}$$

$$B(q^{-1}) = b_0 + b_1q^{-1} + b_2q^{-2} + \dots + b_nBq^{-nB}$$

Se ingresan los nuevos coeficientes a_1 y b_1 , además del retardo, y se verifica la condición de fase mínima. El ingreso de esta información permite conocer n_A , n_B y d , valores requeridos para su utilización en el algoritmo de MRAC.

5.2.5 Selección del polinomio de control $Cr(q^{-1})$

Se seleccionan los coeficientes c_1 respectivos correspondientes al polinomio Cr dado por:

$$Cr(q^{-1}) = 1 + c_1q^{-1} + c_2q^{-2} + \dots + c_nCq^{-nC}$$

Restrictivamente los ceros de Cr deben estar dentro del círculo unitario (polinomio asintóticamente estable).

5.2.6 Implementación del algoritmo

Con la información precedente se establece el funcionamiento del algoritmo MRAC en los siguientes términos:

Se calculan los parámetros iniciales del controlador en base a los datos de la planta original. Para ello se procede al cálculo de los polinomios S y R de acuerdo a lo establecido en el caso de control lineal para plantas con parámetros constantes, es decir, mediante la resolución secuencial de la identidad:

$$C_r(q^{-1}) = A'(q^{-1})S(q^{-1}) + q^{-d}R(q^{-1})$$

Según [4.8] y [4.9] se estableció que,

$$\phi_o^T(k) = [u(k-1), \dots, u(k-d-n_B), y(k), \dots, y(k-n_R)]$$

$$\theta_o^T = [b_0s_1+b_1, b_0s_2+b_1s_1+b_2, \dots, b_nBs_{d-1}, r_0, \dots, r_nR]$$

y por [4.11],

$$\phi^T(k) = [u(k), \phi_o^T(k)]$$

$$\theta^T = [b_0, \theta_o^T]$$

que en términos de estimación en el tiempo se escribe de la forma:

$$\hat{\theta}^T(k) = [\hat{b}_0(k), \hat{\theta}_o^T(k)]$$

En consecuencia, los parámetros iniciales (para $k=0$) que se establecen para el controlador como estimados de partida, suponiendo la planta conocida, están dados por:

$$\theta^{TINI} = [b'_0, b'_0s_1 + b'_1, \dots, b'_nB - s_1^{n-1}, r_0, r_1, \dots, r_nR']$$

$$\theta^{TINI} = [\hat{b}_0(0), \hat{\theta}_0^T(0)]$$

Los coeficientes b'_i son datos (planta inicial) y los coeficientes s_1 y r_1 se han calculado previamente.

A partir de $k=0$ se calcula $y(k)$ mediante:

$$y(k) = -a_1y(k-1) - a_2y(k-2) - \dots - a_nAy(k-nA) + \\ + b_0u(k-d) + b_1u(k-d-1) + \dots + b_nBu(k-d-nB)$$

donde los a_i y b_i corresponden a los a'_i y b'_i de la planta inicial mientras $k < T_{MOD}$ y a los a_i y b_i de la planta modificada para $k \geq T_{MOD}$ respectivamente.

De acuerdo con la ley de control adaptivo la secuencia de control $u(k)$ se calcula para $k=0,1,2,\dots$ como:

$$u(k) = [1/\hat{b}_0(k)] * [Cr(q^{-1})yM(k) - \hat{\theta}_0^T(k)\phi_0(k)]$$

El algoritmo contempla la estimación secuencial de los parámetros del controlador en base a:

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \alpha_k F(k+1) \phi(k-d) [Cr(q^{-1})y(k) - \hat{\theta}^T(k-1)\phi(k-d)]$$

$$F(k+1) = \frac{1}{\alpha_k} \left[F(k) - \frac{F(k)\phi(k-d)\phi^T(k-d)F(k)}{1 + \phi^T(k-d)F(k)\phi(k-d)} \right]$$

$$0 < \alpha_k < 1 - \epsilon$$

$F(k+1)$ es la matriz cuadrada de ganancias cuya dimensión es la correspondiente a la del vector $\phi(k-d)$. El factor de olvido α_k es calculado utilizando uno de los métodos recomendados para identificación recursiva, conocido como de "traza constante", y que consiste en determinar su valor de tal forma que la traza de la matriz de ganancias permanezca constante para todo k , es decir:

$$\text{tr}F(k) = \text{tr}F(0) \quad k = 1, 2, 3, \dots$$

Se establece para la matriz de partida $F(0)$ una matriz de tipo diagonal, es decir $F(0) = \text{diag}(X)$, debiéndose ingresar por consiguiente como dato inicial el valor de X para dicha diagonal.

5.3 DESARROLLO DEL PROGRAMA DE SIMULACION

Tomando como base las dos secuencias básicas de procedimientos anteriormente descritas, en esta sección se presenta el programa de simulación que las involucra, respectivamente.

Para ello se utiliza, como se indicó anteriormente, la versión QuickBASIC 4.0, compilador que se ha constituido al momento en uno de los medios más versátiles y adecuados para la realización de programas en microcomputadores. Se ha optado por esta versión por cuanto reúne características básicas que superan las facilidades de generación de

programas de los compiladores clásicos. Entre las principales de éstas características se pueden señalar las siguientes:

- Creación de programas en base a uno o varios módulos.
- Código fuente dotado de gran versatilidad, permitiendo una programación estructurada.
- Compilación inmediata en memoria.
- Uso automático de coprocesador aritmético.
- Mayor precisión en los números de punto flotante.
- Alta velocidad de ejecución, etc.

Para la organización del programa se aprovecha principalmente la característica de "modularidad" ofrecida por el compilador. Todo el trabajo se realiza en base a subprogramas que interactúan entre sí de acuerdo con cierta lógica funcional, considerándose como subprograma a un segmento de código independiente que está en capacidad de establecer comunicaciones con otras secciones de programa.

Inicialmente se presenta la organización del módulo principal del programa y posteriormente se analiza en forma separada la estructuración de las secciones correspondientes a la simulación de control lineal y de control adaptivo MRAC, respectivamente.

En el diseño de los diagramas representativos se hace uso de una Simbología estándar, la misma que consta en el APENDICE C.

5.3.1 Módulo Principal

Normalmente se asocia el concepto de programa al concepto de archivo, ya que muchas veces un programa está almacenado en un único archivo. En QuickBASIC se amplía esta noción en el sentido de que un programa fuente puede estar contenido en uno o varios archivos. Cada archivo de programa recibe el nombre de módulo y consta de una parte denominada "código a nivel de módulo" y de una o varias partes denominadas generalmente "procedimientos" (subprogramas y funciones). El programa de simulación se ha diseñado tomando como base un módulo principal y varios subprogramas que interactúan convenientemente entre sí.

La denominación de este módulo principal es CAD . Sus funciones básicas comprenden la presentación del trabajo y el direccionamiento lógico del flujo hacia las secciones correspondientes a las simulaciones de control lineal o adaptivo MRAC.

Su diagrama representativo se observa en la FIG. 5.1. Un módulo que utiliza subprogramas contiene instrucciones que los declara como tales. De ello se encarga el primer bloque asignado como de "Declaración de subprogramas" en la figura correspondiente.

Los subprogramas que incluye este módulo y que en consecuencia son declarados en este bloque se especifican a continuación :

CLINEAL
ESTADO1
FASEMIN
FASEMRAC
FIN
FUNCIONTRANSF.
GENERACION
GRAFYM
IMPRESION
INGRESOPLANTA
MENU.PRIN
MODIFICACIONES

MRAC
PRESENTACION
PROGRAMA.A
PROGRAMA.B
PROGRAMA.ADA1
PROGRAMA.ADA2
PUNTO.AP
RESULTADOS.1
RESULTADOS.1.GRAF
RESULTADOS.1.NUM
RETARDO
RETARDO.MRAC

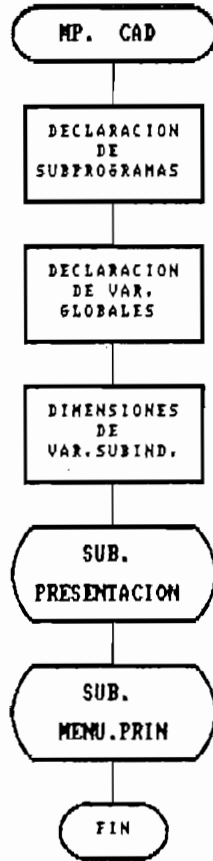


FIG. 5.1 MODULO PRINCIPAL

La programación modular y estructurada exige en ciertas circunstancias la compartición de variables entre módulos y subprogramas, siendo necesario para tales casos las correspondientes instrucciones de compartición. Una variable

dentro de un subprograma es "local" mientras no sea designada como "global" mediante declaración. Para tal efecto, el bloque indicado como "Declaración de variables" se encarga de declarar como tales a un grupo de variables cuyas funciones serán consideradas oportunamente en el desarrollo del programa.

El bloque de "Dimensionamiento" comprende la asignación de las dimensiones de las variables subindicadas globales. A continuación de este bloque se llama al subprograma PRESENTACION cuya función es la de visualizar una carátula informativa de la naturaleza del trabajo.

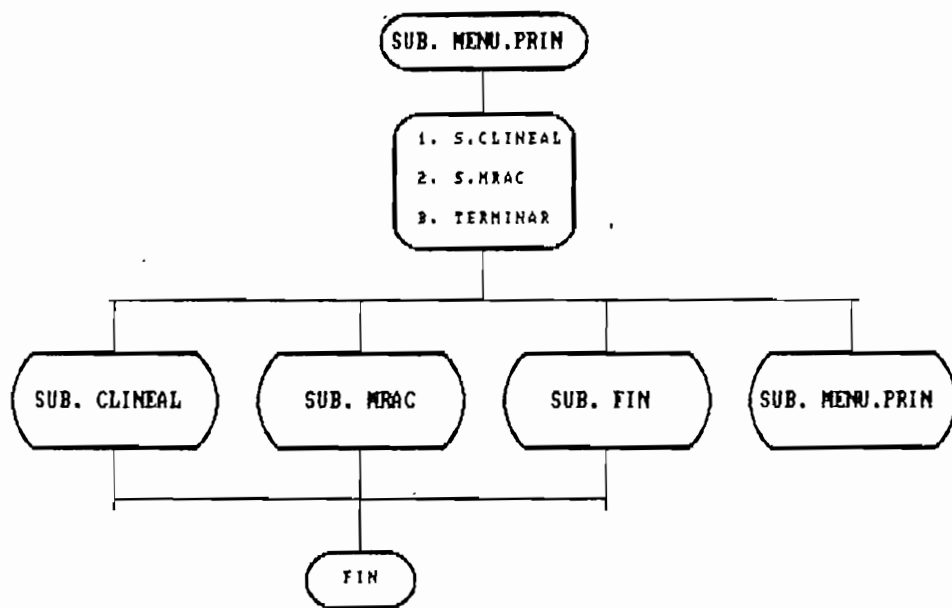


FIG. 5.2 SUBPROGRAMA MENU.PRIM

Finalmente, desde el módulo principal se efectúa el llamado al subprograma MENU.PRIN, cuyo esquema funcional se observa en la FIG. 5.2.

Se dispone de tres opciones: simulación de control lineal, simulación de MRAC y finalización de la sesión de trabajo. Para la autoverificación de errores en la selección se aprovecha las propiedades interactivas del compilador utilizado y se hace un autollamado al subprograma MENU.PRIN.

Los subprogramas CLINEAL y MRAC serán considerados posteriormente en forma extensiva en las secciones correspondientes a cada una de las simulaciones. El subprograma FIN lleva hacia la finalización de la sesión no sin antes presentar una alternativa de reinicio. Su esquema es:

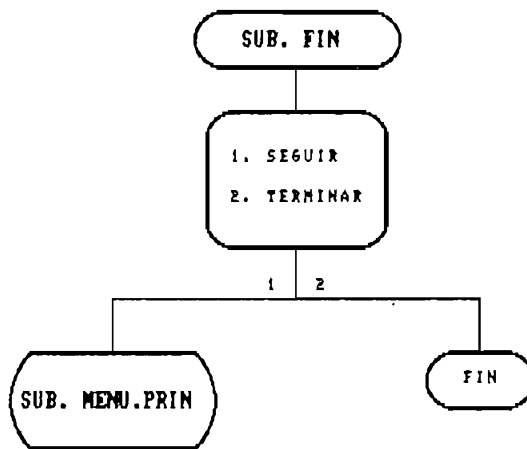


FIG. 5.3 SUBPROGRAMA FIN

5.3.2, Programa para simulación de Control Lineal

El desarrollo del programa para el caso de Control Lineal tiene su punto de partida en el subprograma CLINEAL. Esquemáticamente esta sección de programa está conformada según se muestra en la FIG. 5.4

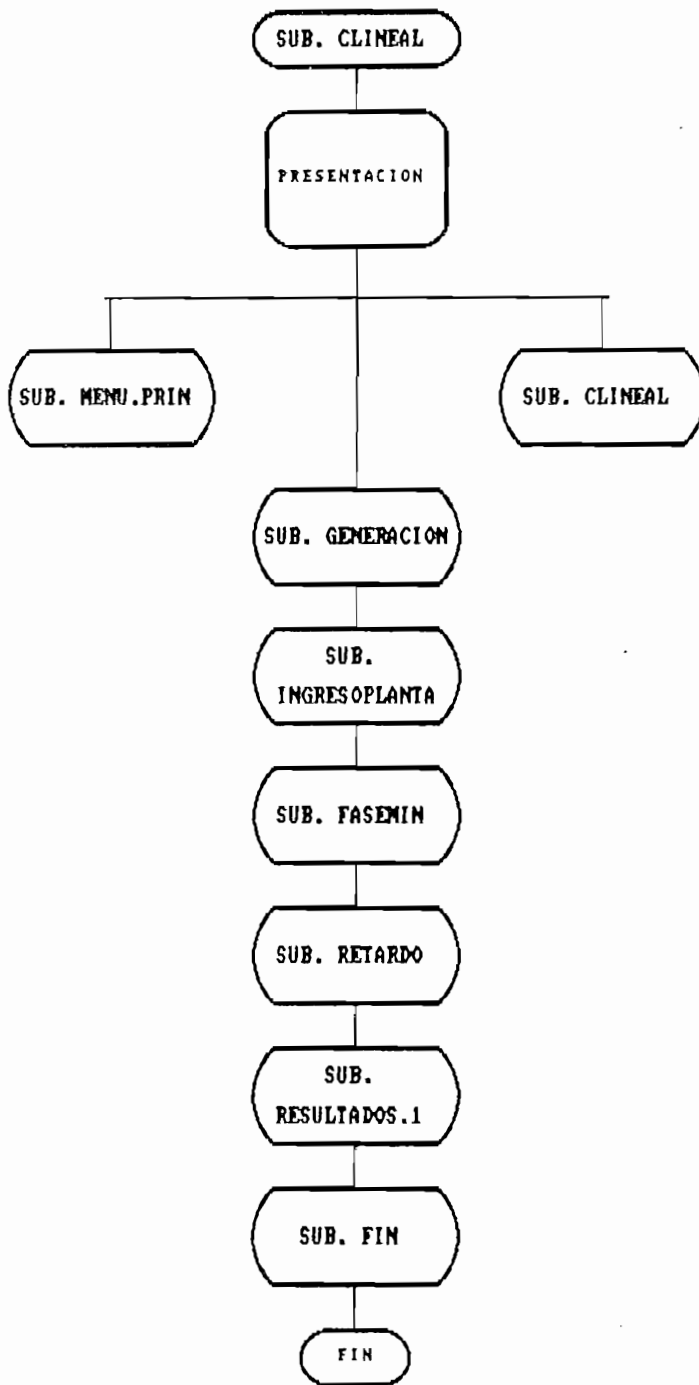


FIG. 5.4 SUBPROGRAMA CLINEAL

El bloque de "Presentación" se refiere al aparecimiento en pantalla de una carátula informativa y se incluye a continuación opciones de direccionamiento según se indica en el

esquema. El flujo normal lleva en primera instancia al llamado del subprograma denominado GENERACION.

5.3.2.1 Generación del modelo de referencia

El subprograma GENERACION cumple con la función básica de generar, almacenar y opcionalmente visualizar gráficamente la secuencia de referencia a seguirse. Su esquema es :

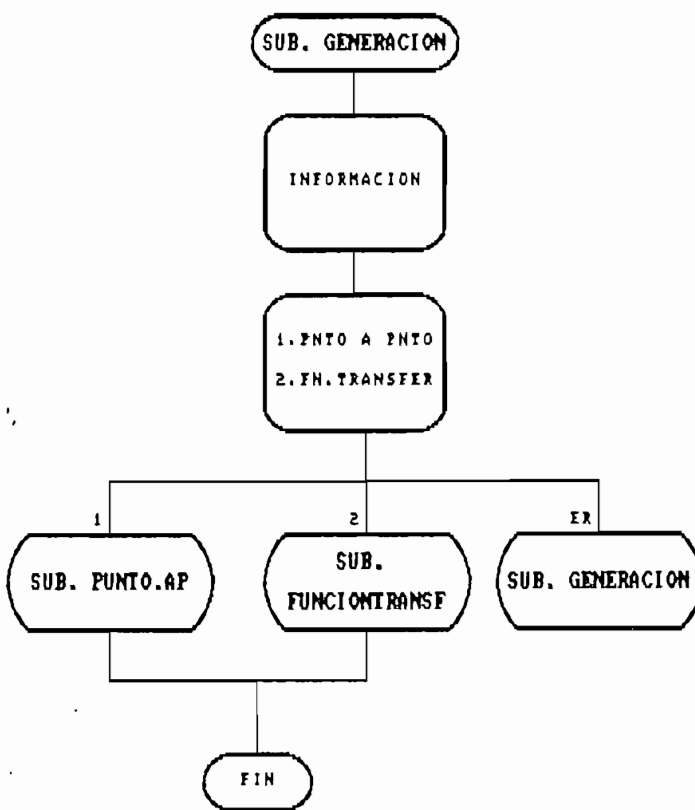


FIG. 5.5 SUBPROGRAMA GENERACION

Las secuencias de puntos correspondientes a la entrada y a la salida del modelo de referencia son almacenadas en los vectores UM() y YM() respectivamente. Estas variables son

tratadas como globales para todo el programa y disponen de un dimensionamiento máximo de 510 elementos.

El bloque "Información" establece la correspondencia entre las variables y su representación. Se presentan como opciones la generación de la secuencia de referencia asignando sus valores punto por punto o, alternativamente mediante el uso de una función de transferencia. Se autoverifica el error de digitación autollamando a GENERACION.

Para la generación punto por punto se direcciona el flujo hacia el subprograma PUNTO.AP. El esquema de este subprograma se observa a continuación en la FIG. 5.6.

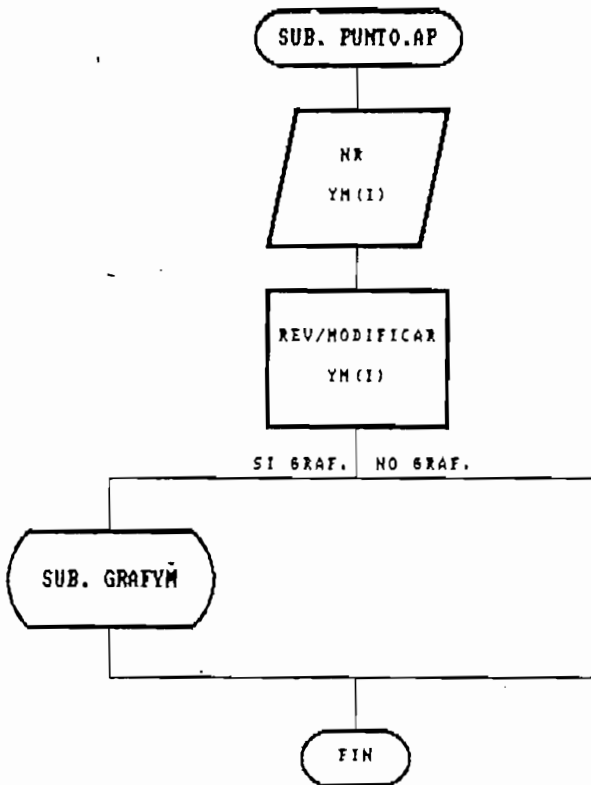


FIG. 5.6 SUBPROGRAMA PUNTO.AP

Por teclado se ingresa el número de puntos de la secuencia, NR, y los valores para cada uno de ellos. Es posible realizar a continuación una revisión de los datos ingresados y/o modificar cualquiera de ellos. Opcionalmente se puede visualizar gráficamente la secuencia ingresada a través del subprograma de gráficos GRAFYM. El esquema de este subprograma se observa en la FIG. 5.7.

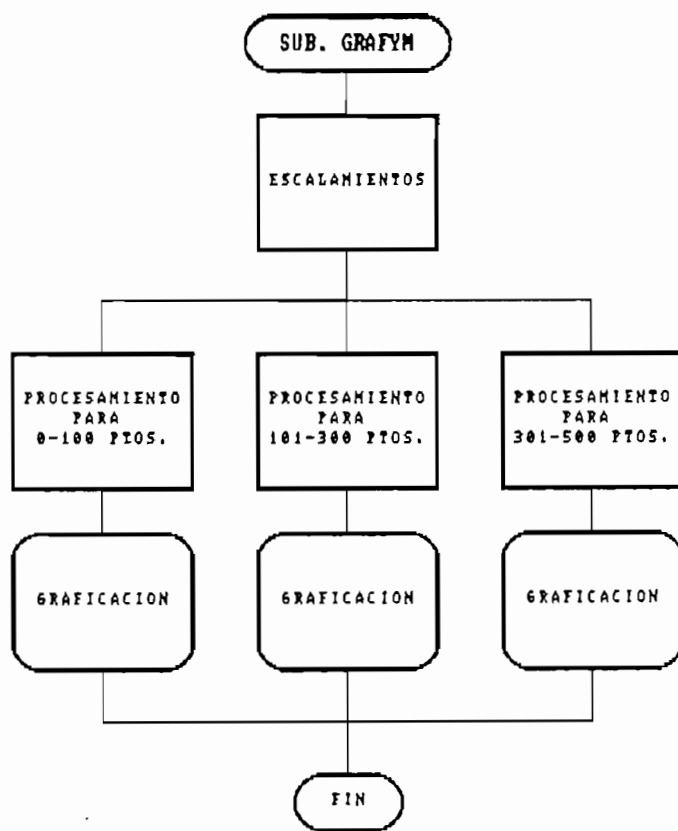


FIG. 5.7 SUBPROGRAMA GRAFYM

La función que cumple GRAFYM es acondicionar los datos para una mejor visualización gráfica en la pantalla. Los puntos a graficarse, en este caso los correspondientes a YM(), son transferidos al vector designado como GYM().

Se realiza a continuación un escalamiento de los valores con fines de presentación y finalmente según sea el número de puntos en consideración se direcciona hacia diferentes bloques de procesamiento y graficación que se encargan a su vez de presentar adecuadamente y con la mejor resolución posible los gráficos en pantalla.

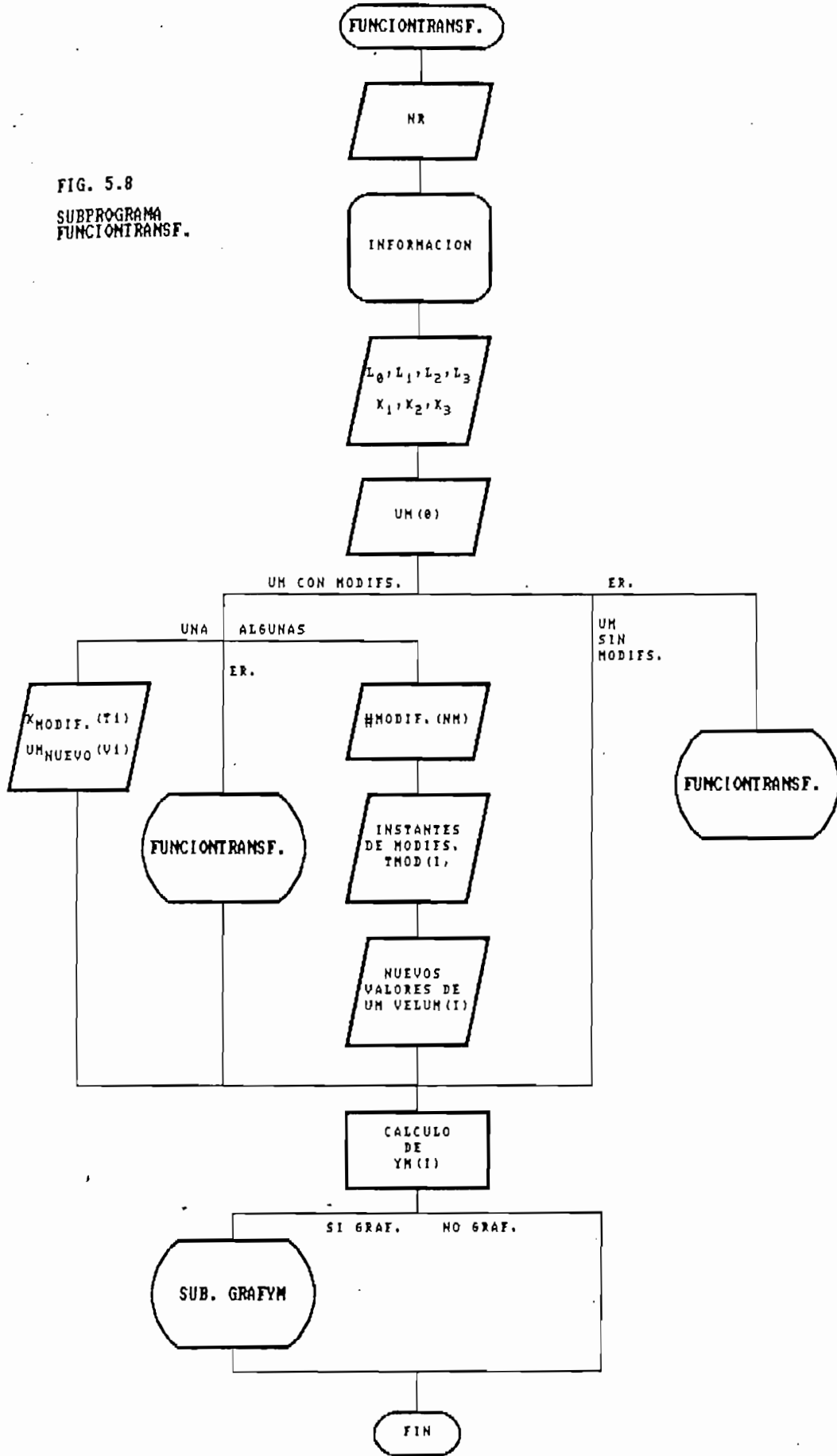
La generación de la secuencia de referencia mediante una función de transferencia se la realiza a través del subprograma denominado FUNCIONTRANSF., estructurado según la FIG. 5.8.

Se procede al ingreso del número de puntos NR de la secuencia de referencia (máx 500). El subprograma informa sobre la definición de la función de transferencia y solicita el ingreso de los coeficientes l_1 y k_1 correspondientes a los polinomios $L(q^{-1})$ y $K(q^{-1})$, respectivamente.

La secuencia de entrada $UM(K)$ como se indicó anteriormente es de tipo escalón. Todos sus puntos pueden tener un mismo valor ($UM(k)=cte, k=0,1,2,\dots$) o seccionadamente variar sus valores. Se preveé así esta posibilidad dando lugar a una o varias modificaciones, debiéndose indicar para ello los instantes en que ocurren (se almacenan en el vector $TMOD()$) y los nuevos valores de UM (se almacenan en el vector $VELUM()$).

La secuencia de referencia $YM(k)$ se calcula iterativamente desde $k=0$ hasta $k=NR$ utilizando para ello la expresión dada por:

FIG. 5.8
SUBPROGRAMA
FUNCIONTRANSF.



$$\begin{aligned}
 YM(k) = & -k_{1y}M(k-1) - k_{2y}M(k-2) - k_{3y}M(k-3) + l_{0u}M(k) + \\
 & + l_{1u}M(k-1) + l_{2u}M(k-2) + l_{3u}M(k-3) \quad k=0,1,\dots
 \end{aligned}$$

En el vector $YM()$ se almacenarán todos estos valores y si el usuario así lo decide se podrá visualizar gráficamente esta secuencia de referencia. Se provee de autoverificación y direccionamientos para los errores en la digitación.

5.3.2.2 Ingreso de la planta y verificación de fase mínima

Se realiza a través de los subprogramas **INGRESOPLANTA** y **FASEMIN**, respectivamente. El esquema del primero de ellos se observa en la FIG. 5.9.

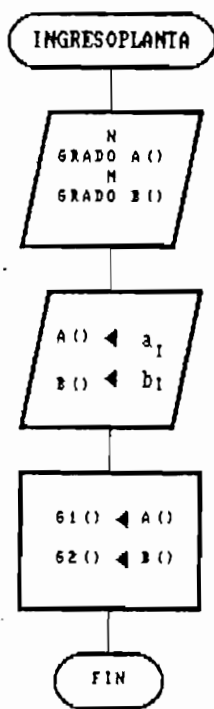


FIG. 5.9 SUBPROGRAMA INGRESOPLANTA

Los grados de los polinomios $A(q^{-1})$ y $B(q^{-1})$ corresponden respectivamente a las variables N y M . Tomando como referencia estos valores el subprograma solicita el ingreso de los coeficientes a_1 y b_1 respectivos. A continuación se transfieren dichos valores a los vectores $G1()$ y $G2()$, variables declaradas globales y que serán empleadas con fines de presentación de resultados posteriormente.

El subprograma FASEMIN tiene como objetivo básico el verificar si la planta ingresada cumple con el condicionamiento de fase mínima; de hacerlo, direcciona el flujo progresivamente, caso contrario notifica el incumplimiento y solicita nuevos datos.

La planta en consideración será de fase mínima si sus ceros se ubican dentro del círculo unitario. De la representación polinomial de la planta se concluye por tanto que será el polinomio $B(q^{-1})$ el que determine si la planta es o no de fase mínima en base al cálculo de las raíces respectivas.

Sea n_B el grado de $B(q^{-1})$, entonces:

$$b_0 + b_1q^{-1} + b_2q^{-2} + \dots + b_{n_B}q^{-n_B} = 0$$

$$(b_0q^{n_B} + b_1q^{n_B-1} + b_2q^{n_B-2} + \dots + b_{n_B})/q^{n_B} = 0$$

de donde:

$$b_0q^{n_B} + b_1q^{n_B-1} + b_2q^{n_B-2} + \dots + b_{n_B} = 0$$

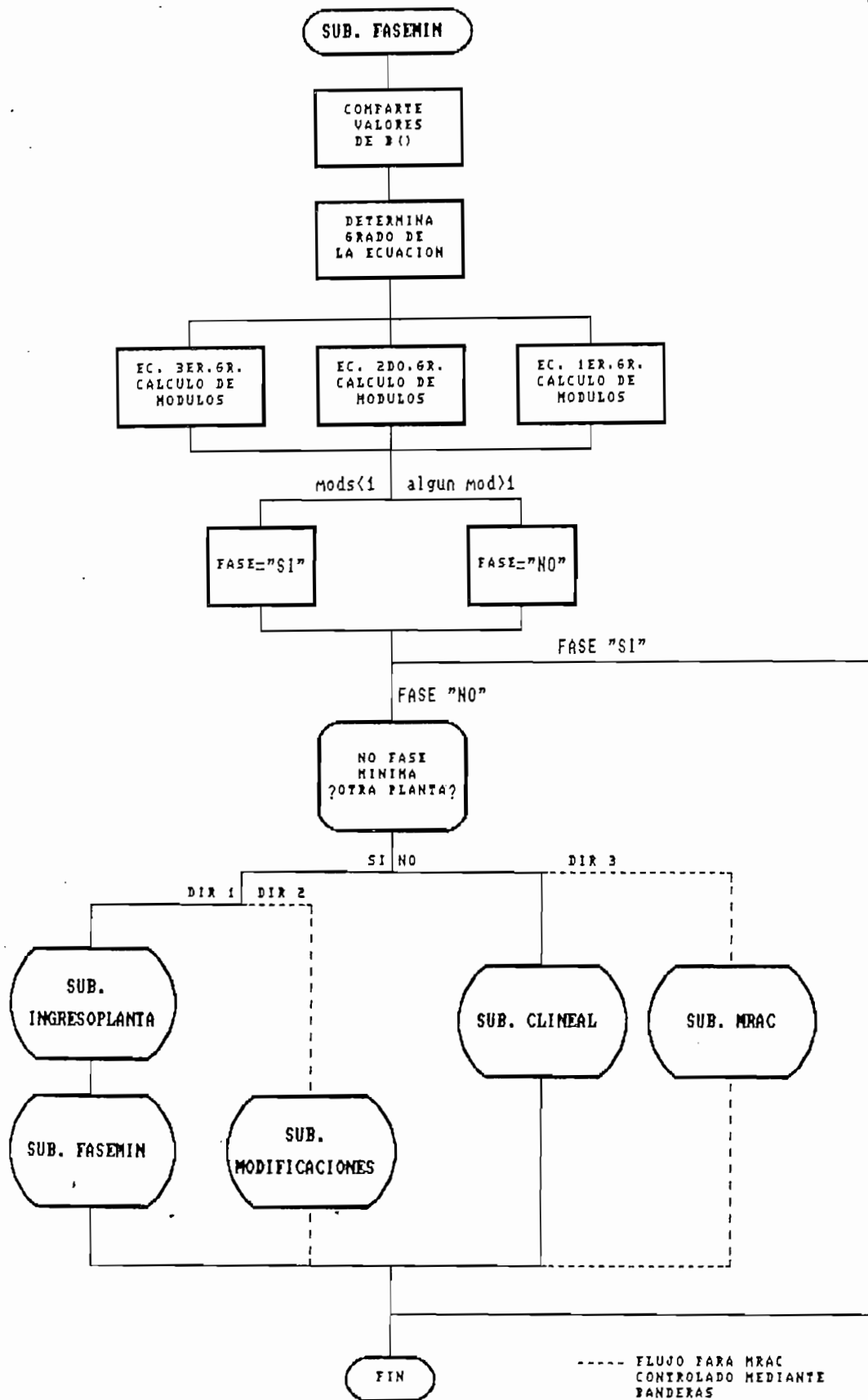


FIG. 5.10 SUBPROGRAMA FASEMIN

En el subprograma FASEMIN se calculan los módulos de las raíces de la ecuación precedente. Si tales módulos son menores que 1 implica que la planta cumple con la condición de fase mínima. Se contempla la resolución algebraica de ecuaciones de hasta nB grado, utilizando para el efecto aritmética compleja y funciones trigonométricas. El método de resolución es el denominado de Cardano.

Este subprograma se aplica también en la simulación para el caso MRAC, razón por la cual se incluye direccionamientos que oportunamente serán considerados y que son controlados mediante el uso de secuencias especificadas de caracteres.

5.3.2.3 Aplicación del algoritmo

Una vez que se ha generado el modelo de referencia, se ha ingresado la planta y se ha verificado que es de fase mínima se procede a la aplicación del algoritmo de control en su parte central. El flujo lleva hacia el llamado del subprograma RETARDO, estructurado según se observa en la FIG.5.11

El subprograma permite seleccionar el retardo del sistema y con esta referencia se direcciona hacia los subprogramas cuyas denominaciones son PROGRAMA.A y PROGRAMA.B. El primero de ellos tiene la estructura mostrada en las FIGS. 5.12, 5.13, y 5.14.

El subprograma recibe como datos los valores correspondientes a N, M, NR, A(), B() y YM(). Con ellos se calcula secuencialmente la señal de control, representada en el vector U(), de tal manera que la secuencia de salida, representada en el vector Y(), siga a la secuencia de referencia generada y almacenada en el vector YM(). Las variables y vectores señalados son de características globales.

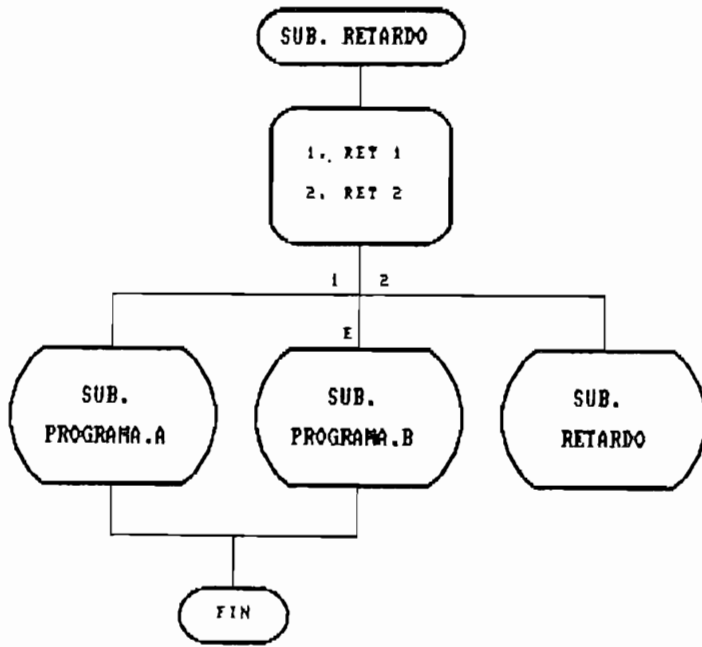


FIG. 5.11 SUBPROGRAMA RETARDO

Se ingresa por teclado el grado GC asignado para el polinomio de control $C_r(q^{-1})$ y con esta referencia el subprograma solicita los coeficientes de este polinomio, los mismos que son almacenados en el vector C1().

A través del vector auxiliar AUX3() se adelanta en 6 pasos referenciales de tiempo la secuencia YM().

FIG. 5.12
SUB. PROGRAMA.A
(PARTE I)

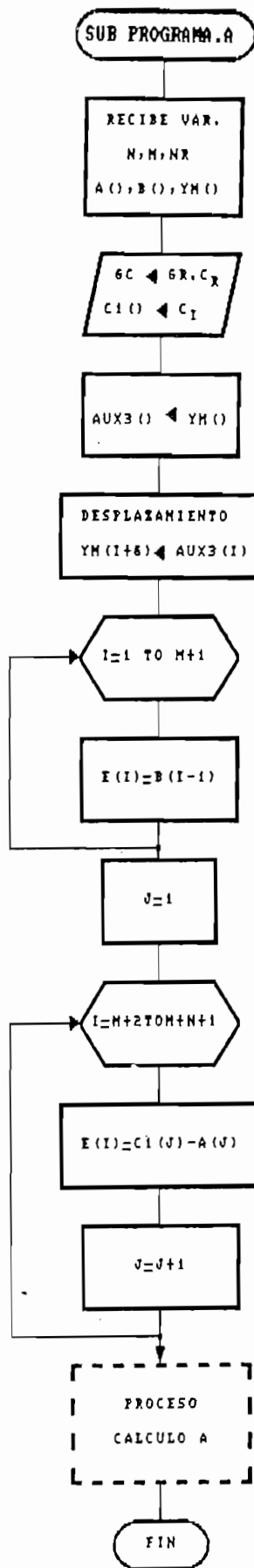


FIG. 5.13
SUB. PROGRAMA.A
(PARTE II)

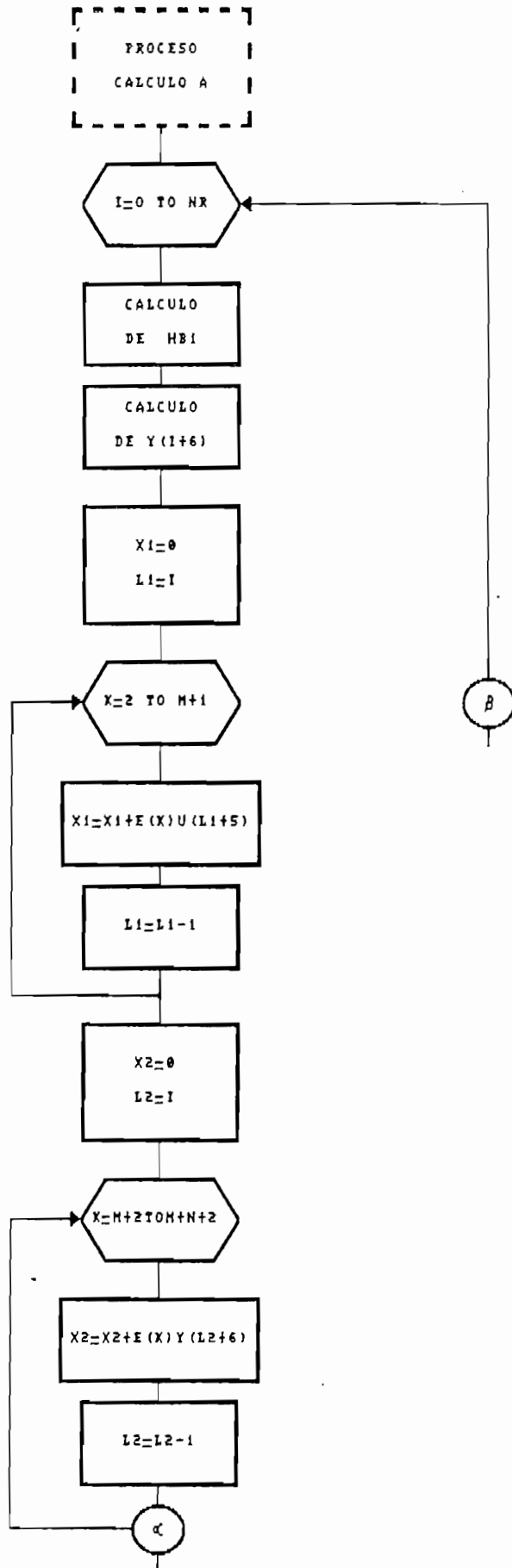
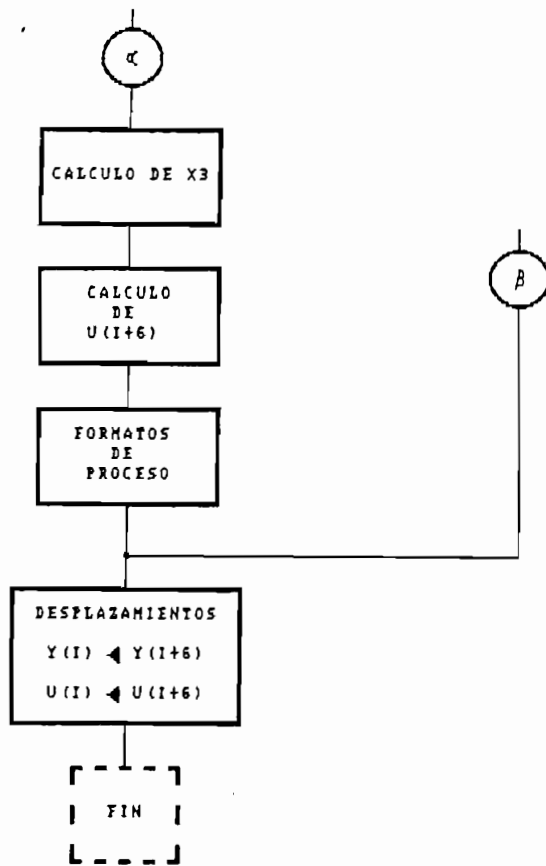


FIG. 5.14
SUB. PROGRAMA.A
(PARTE III)



Este adelantamiento es necesario por cuanto convencionalmente, y solo para fines de procesamiento en el computador, se efectuará el proceso de cálculo con un adelanto aparente de 6 pasos. Note que para determinar $y(k)$ se deberá emplear la expresión proveniente de la representación discreta del sistema con $d=1$ dada por:

$$y(k) = -a_1y(k-1) - a_2y(k-2) - \dots - a_nAy(k-nA) + \\ + b_0u(k-1) + b_1u(k-2) + \dots + b_nBu(k-nB-1)$$

Observe que para $k=0$ se requeriría los elementos de vector $Y(-1)$, $Y(-2)$, ..., $U(-2)$, ..., etc., que como tales (variables subíndicadas) no las acepta el computador dado que

los índices aceptables formalmente son 0,1,2,...etc. Para evitar esto se recurre al adelanto aparente en los subíndices (en aquellos vectores involucrados) en 6 pasos. Culminado el cálculo y una vez que se almacenan los valores se elimina el desplazamiento asumido. En conclusión, para fines de cálculo se determina $y(k+6)$ pero en términos reales esto representa $y(k)$.

A continuación se determinan los parámetros del controlador, los mismos que están representados en el vector $E()$, es decir :

$$\text{Parámetros del controlador} = \begin{bmatrix} E(1) \\ E(2) \\ E(3) \\ \dots \\ \dots \end{bmatrix}$$

La ley de control lineal establece que :

$$u(k) = (1/b_0) * (C r(q^{-1})yM(k) - R(q^{-1})y(k) - B s(q^{-1})u(k))$$

Los parámetros del controlador vienen dados por tanto a partir de los coeficientes de los polinomios R y Bs , donde,

$$B s(q^{-1}) = B(q^{-1})S(q^{-1}) - b_0$$

Se determinó que $n_S = d-1$, por tanto ya que $d = 1$,

el grado de $S(q^{-1})$ es 0, de donde :

$$S(q^{-1}) = 1$$

$$B_s(q^{-1}) = b_1q^{-1} + b_2q^{-2} + \dots + b_{nB}q^{-nB}$$

Si se considera el grado de C_r menor o igual que el de A entonces, de la relación [2.13] se establece que $n_R = n_A - 1$, caso contrario $n_R = n_C - 1$. El vector de parámetros a considerarse es por tanto el siguiente :

$$\theta = \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ \dots \\ b_{nB} \\ r_0 \\ r_1 \\ \dots \\ r_{nR} \end{bmatrix} = \begin{bmatrix} E(1) \\ E(2) \\ E(3) \\ \dots \\ E(nB+1) \\ E(nB+2) \\ E(nB+3) \\ \dots \\ E(nB+nR+2) \end{bmatrix}$$

La resolución secuencial de la identidad [2.1] establece que, para $d = 1$:

$$r_0 = c_1 - a_1$$

$$r_1 = c_2 - a_2$$

$$\dots$$

$$\dots$$

Por consiguiente PROGRAMA.A calcula los parámetros del controlador de la forma previamente establecida y como se indicó los almacena en el vector E(). A continuación, en el bloque interno denominado "Proceso/Cálculo A" se procede a determinar las secuencias de salida y control desde $k=0$ hasta $k=NR$, según se observa en la Parte II del esquema del PROGRAMA.A. Los bloques correspondientes a "Cálculos de HB1 y $Y(I+6)$ " involucran el cálculo de $Y()$ mediante la ecuación de diferencias proveniente de la representación polinomial de la planta, y contienen :

$$HB1 = B(0)*U(I+5)+B(1)*U(I+4)+B(2)*U(I+3)+B(3)*U(I+2)$$

$$Y(I+6) = -A(1)*Y(I+5)-A(2)*Y(I+4)-A(3)*Y(I+3)+HB1$$

A continuación se realizan las siguientes asignaciones para las variables X1, X2 y X3 :

$$X1 = -Bs(q^{-1})u(k)$$

$$X2 = -R(q^{-1})y(k)$$

$$X3 = Cr(q^{-1})yM(k)$$

y tomando en cuenta los desplazamientos respectivos se calculan sus valores. Seguidamente se determina la señal de control U aplicando la ley de control y reemplazando las variables respectivas, a efecto de lo cual se emplea la siguiente expresión:

$$U(I+6) = (1/E(1))*(X3-X2-X1)$$

El "Proceso A" se reinicia para el siguiente instante k , y así sucesivamente en forma iterativa hasta NR . Finalmente se eliminan los desplazamientos en $Y()$ y $U()$, respectivamente.

El subprograma denominado PROGRAM.A efectúa similar función que la del anterior pero para un retardo de la planta de dos pasos ($d=2$). Su estructuración se observa en las FIGS. 5.15, 5.16 y 5.17.

Las reglas de funcionamiento son similares a las ya consideradas para el caso de retardo unitario. El grado de $S(q^{-1})$ toma el valor de 1, por lo cual :

$$Bs(q^{-1}) = B(q^{-1})S(q^{-1}) - b_0$$

$$Bs(q^{-1}) = B(q^{-1})(1 + s_1q^{-1}) - b_0$$

de donde,

$$bs_1 = (b_1 + b_0s_1)$$

$$bs_2 = (b_2 + b_1s_1)$$

$$bs_3 = (b_3 + b_2s_1)$$

.....

FIG. 5.15
SUB. PROGRAMA.B
(PARTE I)

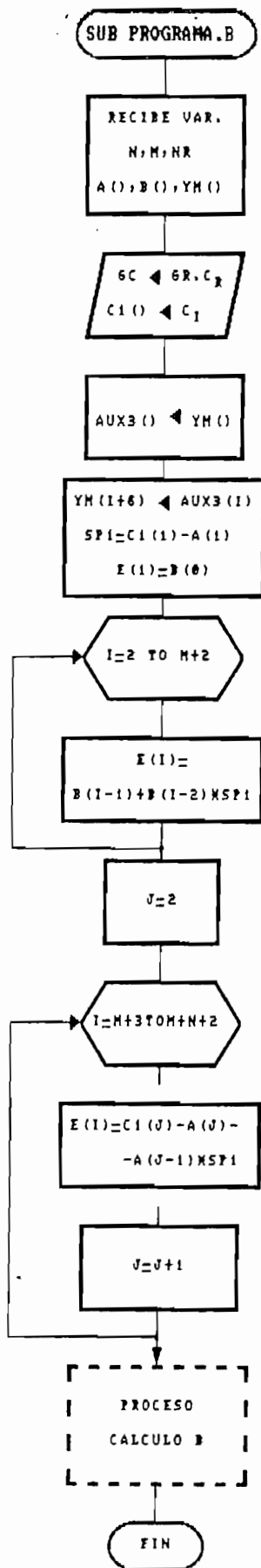


FIG. 5.16
SUB. PROGRAMA.B
(PARTE II)

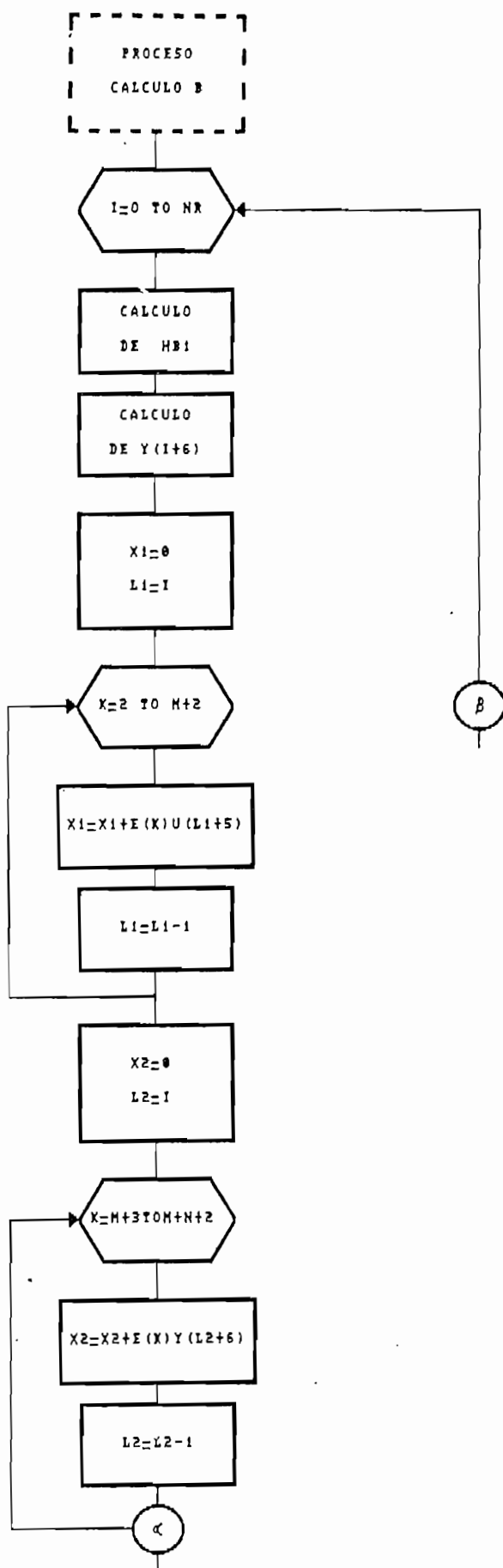
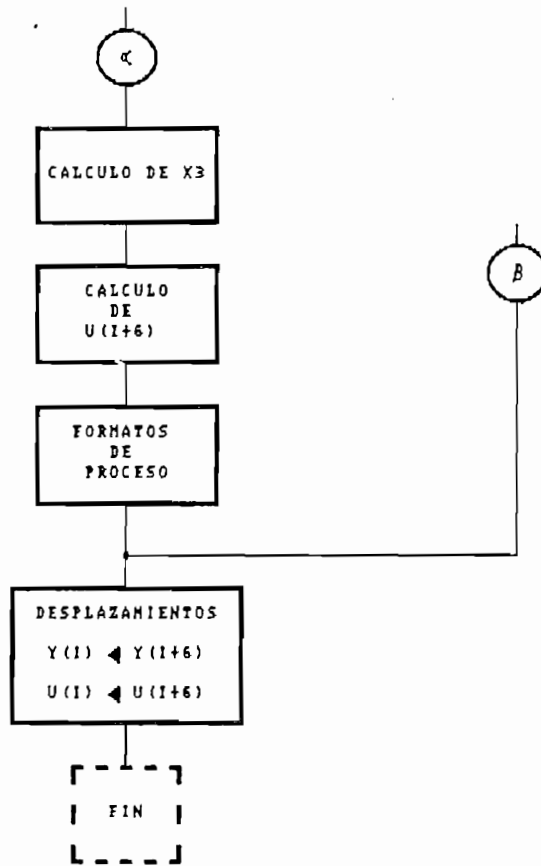


FIG. 5.17
SUB. PROGRAMA.B
(PARTE III)



De la resolución secuencial de la identidad polinomial [2.11] se establece que :

$$s_0 = 1$$

$$s_1 = c_1 - a_1 s_0$$

y para $R(q^{-1})$,

$$r_0 = c_2 - a_1 s_1 - a_2 s_0$$

$$r_1 = c_3 - a_2 s_1 - a_3 s_0$$

.....

En el subprograma se contempla por consiguiente el vector de parámetros definido por :

$$\theta = \begin{bmatrix} b_0 \\ b_{s1} \\ b_{s2} \\ \dots \\ b_{s_{nB+1}} \\ r_0 \\ r_1 \\ \dots \\ r_{nR} \end{bmatrix} \begin{bmatrix} E(1) \\ E(2) \\ E(3) \\ \dots \\ E(nB+2) \\ E(nB+3) \\ E(nB+4) \\ \dots \\ E(nB+nR+3) \end{bmatrix}$$

La variable SP1 representa al valor de s_1 . Se calculan consecuentemente los parámetros en la forma establecida y se almacenan en el vector E().

HB1 y Y(I+6) se calculan mediante :

$$HB1 = B(0)*U(I+4)+B(1)*U(I+3)+B(2)*U(I+2)+B(3)*U(I+1)$$

$$Y(I+6) = -A(1)*Y(I+5)-A(2)*Y(I+4)-A(3)*Y(I+3)+HB1$$

Las variables X1, X2 y X3 se determinan como :

$$X1 = -Bs(q^{-1})u(k)$$

$$X2 = -R(q^{-1})y(k)$$

$$X3 = Cr(q^{-1})yM(k)$$

Se calculan dichos valores con los ya establecidos y considerando los desplazamientos en los vectores $U(k)$, $Y(k)$ y $YM(k)$ se determina la señal de control mediante :

$$U(I+6) = (1/E(1))*(X3 - X2 - X1)$$

y se vuelve a reiniciar el "Proceso B" de cálculo para el siguiente instante k . Para terminar se eliminan los desplazamientos en $Y()$ y $U()$, respectivamente.

5.3.2.4 Presentación de resultados

El siguiente paso consiste en el llamado al subprograma RESULTADOS.1, estructurado según la FIG. 5.18.

Opcionalmente los resultados pueden ser visualizados en forma numérica, gráfica o de ambas a la vez. El diagrama indica el direccionamiento para cualquiera de estos casos. El subprograma ESTADO1 se encarga de presentar previamente en la pantalla los datos correspondientes al ejercicio que se está simulando y cuyos resultados se desean observar. El diagrama del subprograma ESTADO1 se observa en la FIG. 5.19.

La presentación numérica de los resultados se realiza por intermedio del subprograma denominado RESULTADOS.1.NUM, el cual establece formatos de salida adecuados para una apreciación correcta de los valores. El esquema de este subprograma se encuentra en la FIG. 5.20.

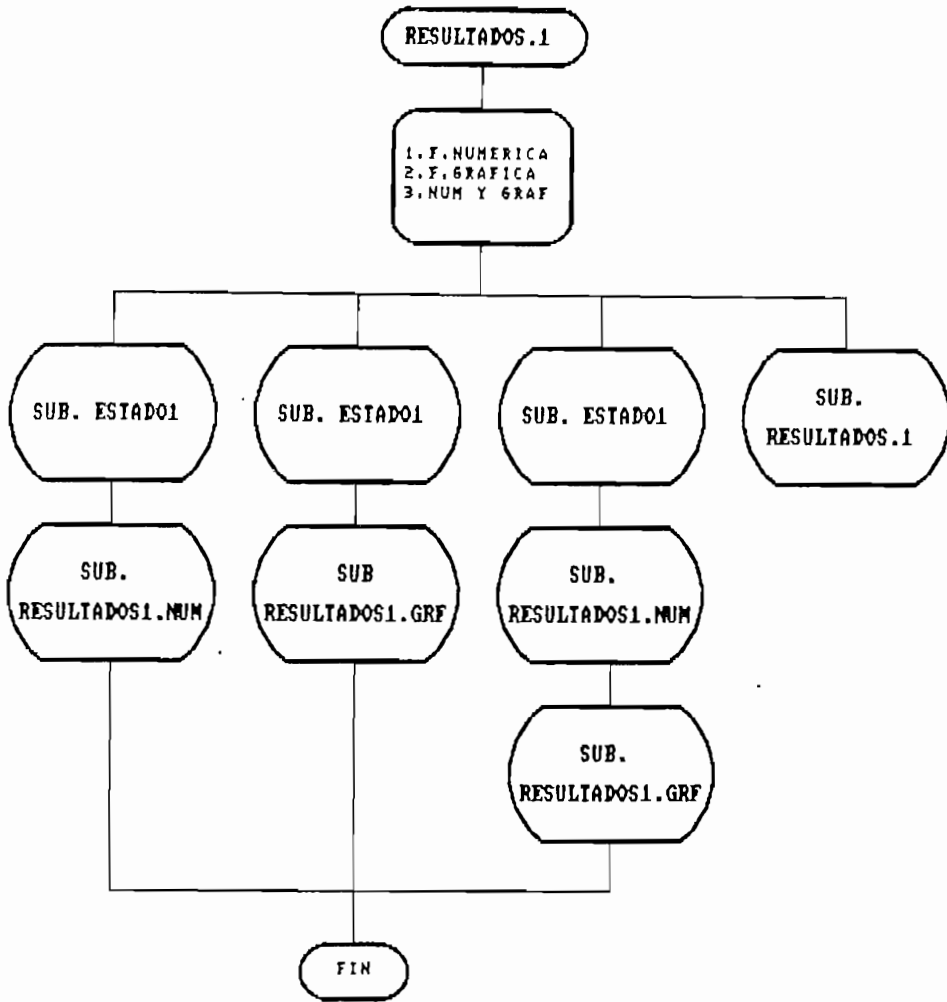


FIG. 5.18 SUBPROGRAMA RESULTADOS.1

Adicionalmente se presenta como una opción la impresión en papel de los resultados numéricos establecidos. Para ello el direccionamiento concurre hacia el llamado del subprograma definido como IMPRESION. Este subprograma se encarga de acondicionar los resultados numéricos de forma tal que sea factible su impresión adecuada en papel, procurando establecer formatos de salida que permitan apreciar correctamente los valores. Se proveé mensajes de orientación para efectuar la tarea.

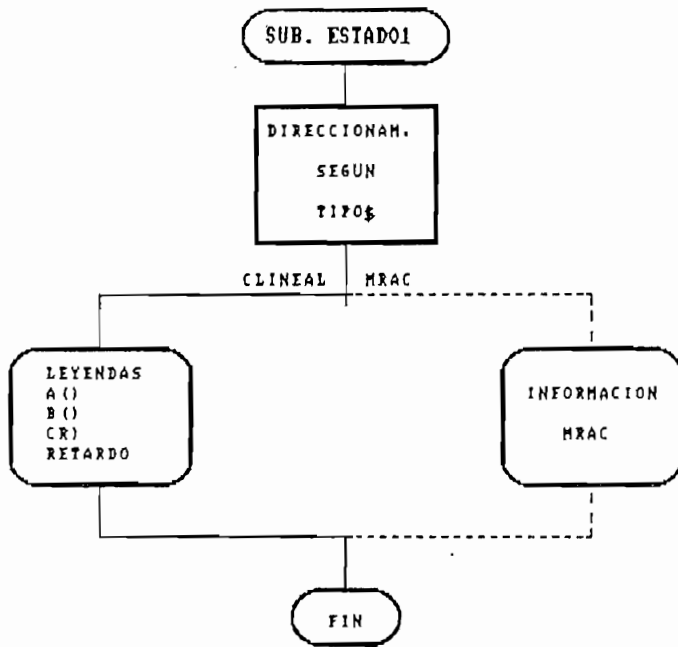


FIG. 5.19 SUBPROGRAMA ESTAD01

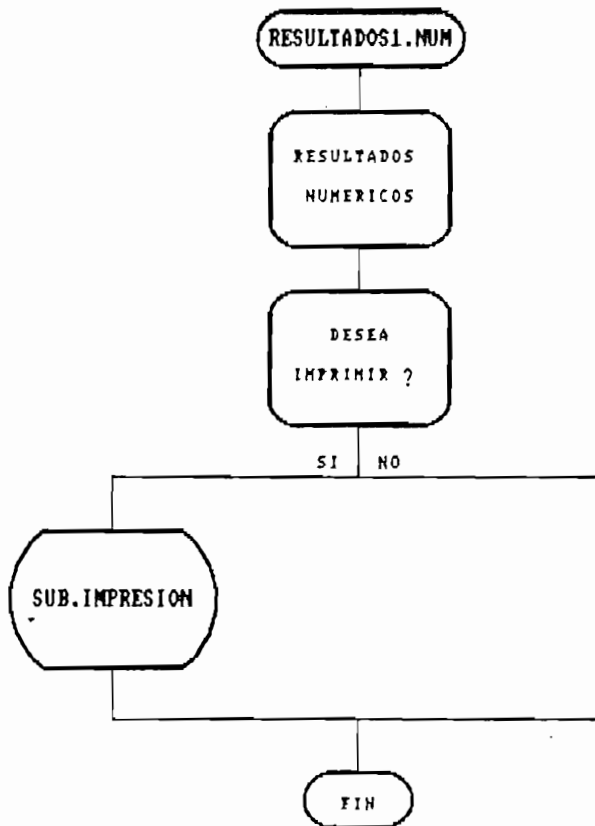


FIG. 5.20 SUBPROGRAMA RESULTADOS1.NUM

El subprograma RESULTADOS.1.GRAF a su vez cumple con la función de presentar en pantalla gráficamente los resultados obtenidos en cada ejercicio de simulación. Para ello se aprovecha la máxima resolución disponible en el equipo utilizado (720x348 pixels). Los esquemas de RESULTADOS.1.GRAF e IMPRESION se observan en las FIGS. 5.21 y 5.22 .

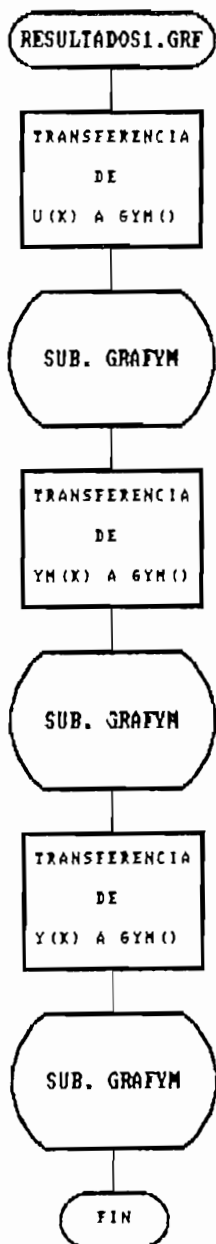


FIG. 5.21 SUBPROGRAMA RESULTADOS1.GRAF

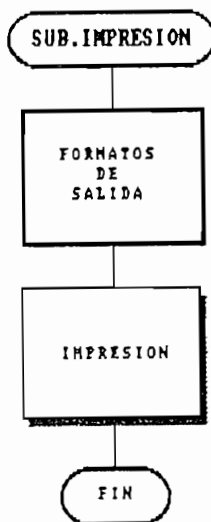


FIG. 5.22 SUBPROGRAMA IMPRESION

5.3.3 Programa para simulación de MRAC

El desarrollo de la sección de programa para la simulación de control adaptivo MRAC tiene su punto de partida en el direccionamiento desde el subprograma MENU.PRIN hacia el subprograma MRAC.

El subprograma MRAC está estructurado como se observa en la FIG. 5.23. El bloque indicado como "Presentación" se encarga de visualizar en la pantalla una carátula informativa y direcciona el flujo según lo indicado en el esquema.

5.3.3.1 Generación del modelo de referencia

Se realiza mediante el subprograma GENERACION y en los mismos términos que los considerados para el caso de Control Lineal (sección 5.3.3.1).

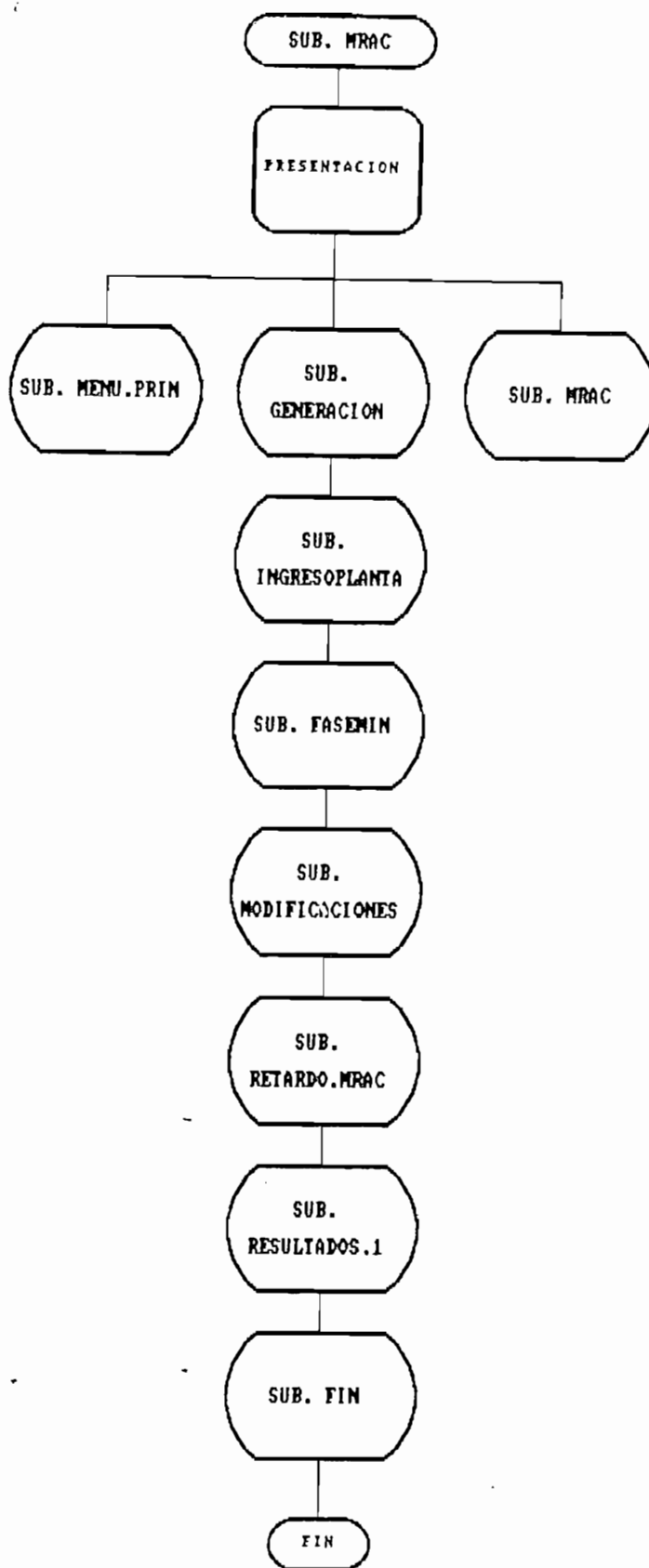


FIG. 5.23 SUBPROGRAMA MRAC

Como se indicó anteriormente, esta generación del modelo puede realizarse punto por punto o mediante una función de transferencia.

5.3.3.2 Ingreso de la planta inicial y verificación de fase mínima.

La planta inicial a considerarse fue propuesta en la sección 5.2.2 como :

$$A'(q^{-1})y(k) = q^{-d}B'(q^{-1})u(k)$$

Para el ingreso de sus datos se utiliza el subprograma INGRESOPLANTA, cuya estructura fue ya analizada en la sección 5.3.2.2. Por lo tanto, para el caso en consideración, las variables N y M representan a nA' y nB' , grados de $A'(q^{-1})$ y $B'(q^{-1})$, y, los vectores $A()$ y $B()$ almacenarán los coeficientes a'_1 y b'_1 , respectivamente. Con estos datos referenciales se procede a la verificación de fase mínima transfiriendo para ello a FASEMIN los coeficientes b'_1 .

Este subprograma fue ya revisado en la sección 5.3.2.2 y en su esquema se observan direccionamientos controlados para su utilización en el caso de MRAC. Si la planta inicial no es de fase mínima y se desea ingresar otra en su reemplazo, el flujo en FASEMIN se direcciona por la línea marcada como DIR1 (ver FIG. 5.10). Si no se desea ingresar otra planta se dirige por DIR3.

5.3.3.3 Ingreso de la planta modificada y verificación de fase mínima

Una vez que se ha ingresado una planta inicial de fase mínima es necesario proporcionar al programa otra planta diferente para que el esquema de control proceda a la adaptación respectiva. Esta nueva planta se ingresa por intermedio del subprograma MODIFICACIONES, presentado en la FIG. 5.24.

La nueva planta a ingresar puede tener o no modificaciones en los grados de sus respectivos polinomios representativos. El subprograma posibilita el direccionamiento en cualquiera de los dos sentidos.

Las variables N.1 y M.1 representan los nuevos grados n_A y n_B de los polinomios $A(q^{-1})$ y $B(q^{-1})$ y los vectores $A1()$ y $B1()$ contienen los nuevos coeficientes a_1 y b_1 , respectivamente.

Se ofrece un resguardo temporal de $A1()$ y $B1()$ para fines de presentación. Con los nuevos datos referenciales se procede al llamado del subprograma FASEMRAC que cumple con las mismas funciones que FASEMIN, es decir, verifica si la nueva planta ingresada es o no de fase mínima. Si el condicionamiento de fase mínima es cumplido se continúa el flujo normal del programa en la simulación, caso contrario se posibilita un nuevo ingreso de la planta o la finalización del ejercicio.

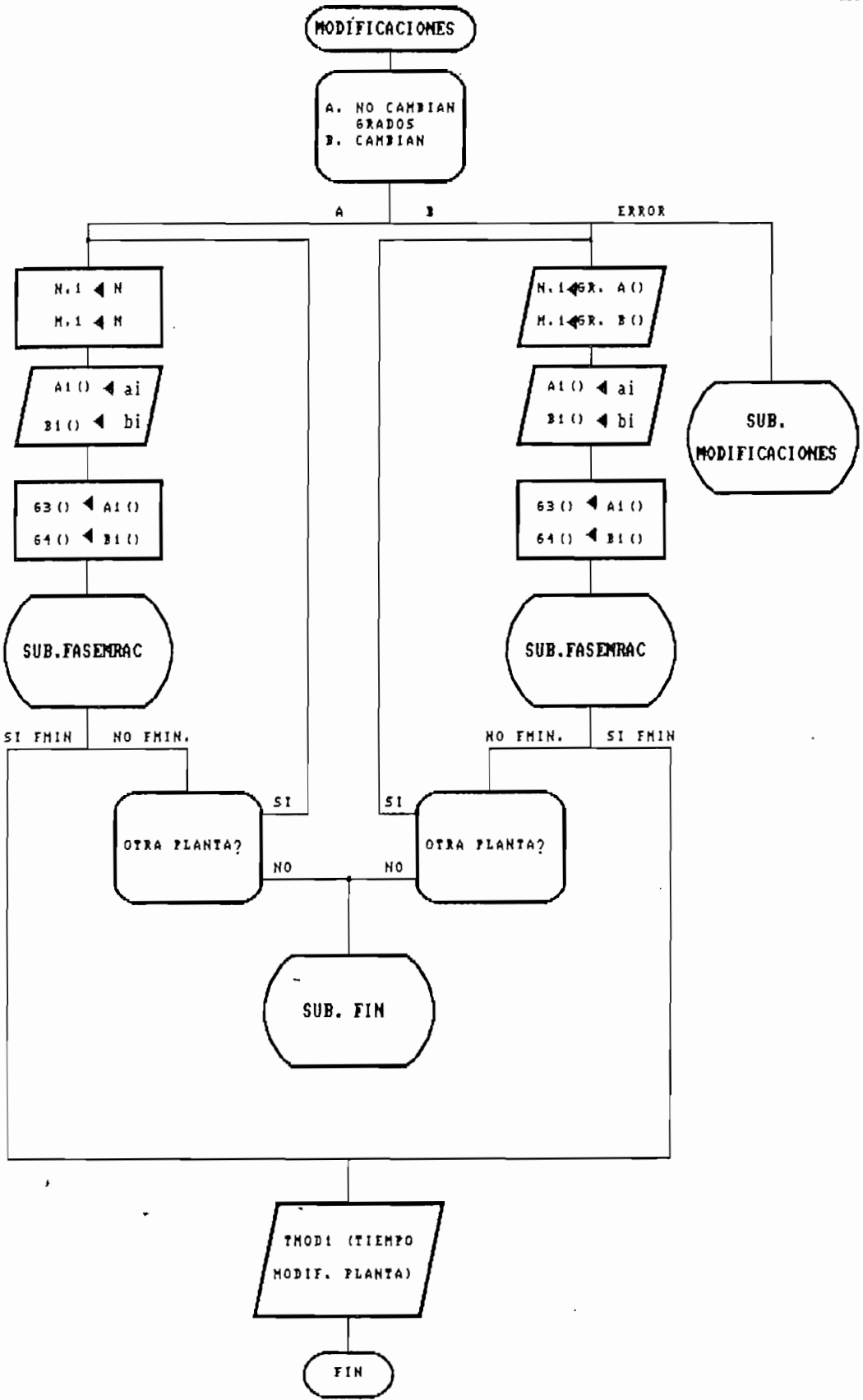


FIG. 5.24 SUBPROGRAMA MODIFICACIONES

Finalmente MODIFICACIONES solicita el ingreso del instante k en el cual ocurre la modificación. Este instante es arbitrario, siempre y cuando no exceda a NR (número máximo de puntos considerados).

5.3.3.4 Aplicación del algoritmo

Una vez que se ha generado una secuencia de referencia, se ha determinado una planta inicial y las modificaciones de ésta en cierto instante se procede a la aplicación del algoritmo de control. El diseño del flujo de programación lleva hacia el subprograma denominado RETARDO.MRAC, subprograma estructurado como a continuación se indica en la FIG. 5.25.

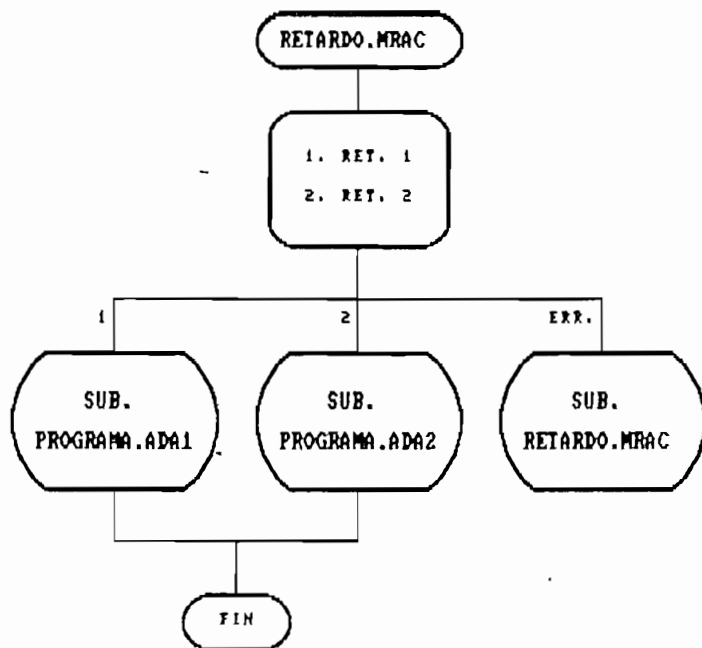


FIG. 5.25 SUBPROGRAMA RETARDO.MRAC

Se selecciona el retardo del sistema y con esta referencia se dirige hacia los subprogramas PROGRAMA.ADA1 o PROGRAMA.ADA2, respectivamente. La estructura general de funcionamiento es análoga para los dos subprogramas, presentándose en las FIGS. 5.26, 5.27 y 5.28 un diagrama general de bloques que involucra a los dos subprogramas mencionados.

A continuación se presenta una descripción del funcionamiento del esquema :

Se reciben como datos las siguientes variables :

N : grado de $A'(q^{-1})$ inicial, nA'
M : grado de $B'(q^{-1})$ inicial, nB'
A() : vector de coeficientes a'_1 de la planta inicial
B() : vector de coeficientes b'_1 de la planta inicial
N.1 : grado de $A(q^{-1})$ modificado, nA
M.1 : grado de $B(q^{-1})$ modificado, nB
A1() : vector de coeficientes a_1 de la planta modificada
B1() : vector de coeficientes b_1 de la planta modificada
YH() : vector de puntos de la secuencia de referencia

Se ingresa el grado GC y los coeficientes del polinomio de control $Cr(q^{-1})$, siendo representados por el vector C1().

El algoritmo contempla entre sus principales etapas de procesamiento las correspondientes al cálculo de los parámetros estimados del controlador y la aplicación de la ley de control.

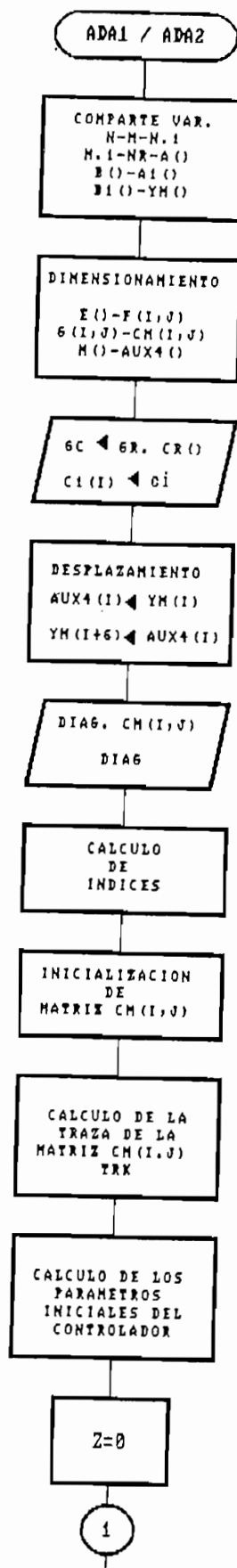


FIG. 5.26 PROGRAMA ADA1/ADA2 (PARTE I)

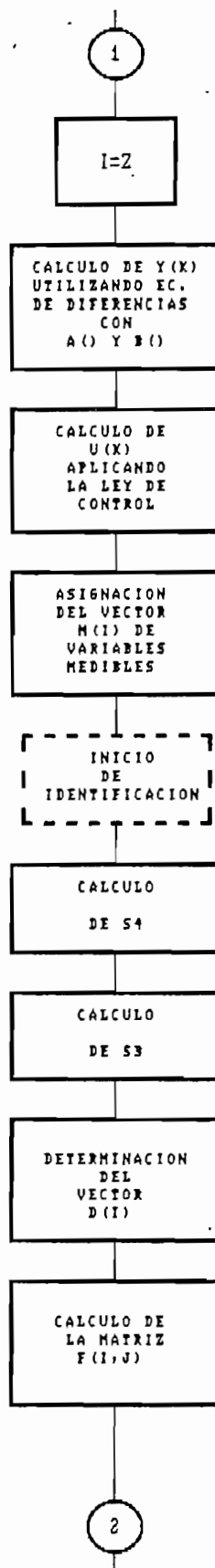


FIG 5.27 PROGRAMA ADAL/ADA2 (PARTE II)

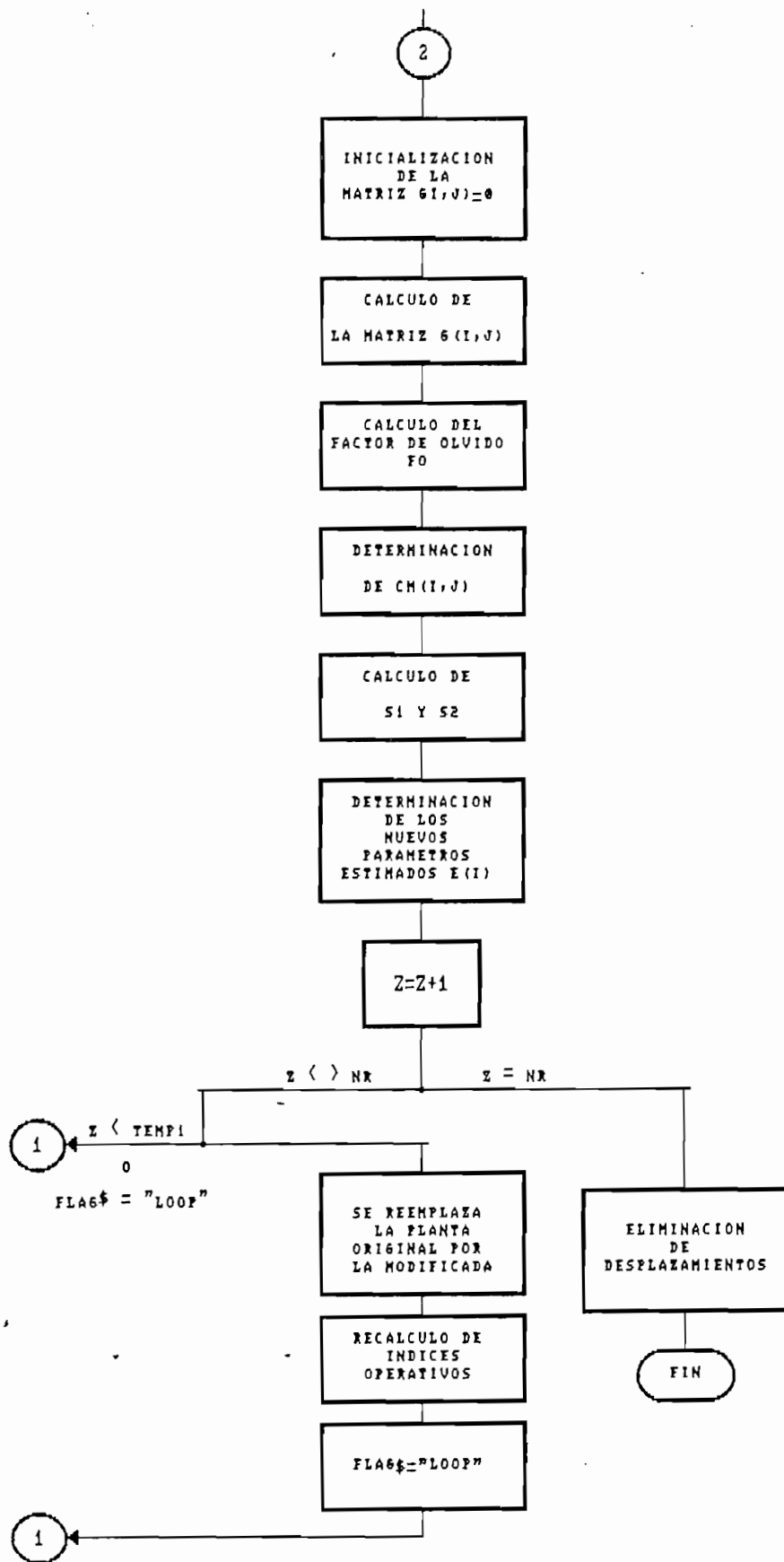


FIG 5.28 PROGRAMA ADAL/ADA2 (PARTE III)

Se ha establecido con anterioridad una ley de control de la forma:

$$u(k) = (1/\hat{b}_0) * (Cr(q^{-1})yM(k) - \hat{\theta}_0^T \phi_0(k))$$

y según [4.11] :

$$\hat{\theta}^T(k) = [\hat{b}_0(k), \hat{\theta}_0^T(k)]$$

$$\phi^T(k) = [u(k), \phi_0^T(k)]$$

Que suprimiendo el indicativo de estimados, según [4.23] y [4.24] equivale a tener :

$$\theta^T = [b_0, b_0s_1 + b_1, b_0s_2 + b_1s_1 + b_2, \dots, b_n s^{d-1} + b_{n-1}, r_0, r_1, \dots, r_{nR}]$$

$$\phi^T(k) = [u(k), u(k-1), \dots, u(k-d-nB+1), \dots, y(k-nR)]$$

La estimación paramétrica se establece mediante:

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \alpha_k F(k+1) \phi(k-d) [Cr(q^{-1})y(k) - \hat{\theta}^T(k-1)\phi(k-d)]$$

$$F(k+1) = \frac{1}{\alpha_k} \left[F(k) - \frac{F(k)\phi(k-d)\phi^T(k-d)F(k)}{1 + \phi^T(k-d)F(k)\phi(k-d)} \right]$$

$$0 < \alpha_k < 1$$

Consecuentemente, como el retardo considerado del sistema para este caso es unitario ($d=1$), entonces :

$$nS = d-1 = 0$$

$$S(q^{-1}) = 1$$

de donde $s_0 = 1$ y $s_i = 0$ para $i \geq 1$

Por lo tanto el vector θ^T en consideración es :

$$\theta^T = [b_0 , b_1 , b_2 , \dots , b_{nB} , r_0 , r_1 , \dots , r_{nR}] \quad [I]$$

y $\phi^T(k-1)$ viene dado por :

$$\phi^T(k-1) = [u(k-1), \dots, u(k-1-nB), y(k-1), \dots, y(k-1-nR)] \quad [II]$$

En consecuencia, los parámetros estimados se determinan mediante :

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \alpha_k F(k+1) \phi(k-1) [Cr(q^{-1})y(k) - \hat{\theta}^T(k-1)\phi(k-1)] \quad [III]$$

donde F viene dada por:

$$F(k+1) = \frac{1}{\alpha_k} \left[F(k) - \frac{F(k)\phi(k-1)\phi^T(k-1)F(k)}{1 + \phi^T(k-1)F(k)\phi(k-1)} \right] \quad [IV]$$

$$0 < \alpha_k < 1$$

Los parámetros iniciales del controlador ($\theta^T_{k=0}$) son calculados de manera similar a lo efectuado en la sección anterior para control lineal.

Estos parámetros del controlador son considerados en el vector $E()$, y, el vector de variables medibles es representado en el vector $M()$, es decir,

$$\theta = \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ \dots \\ b_{nB} \\ r_0 \\ r_1 \\ \dots \\ r_{nR} \end{bmatrix} = \begin{bmatrix} E(1) \\ E(2) \\ E(3) \\ \dots \\ E(nB+1) \\ E(nB+2) \\ E(nB+3) \\ \dots \\ E(nB+nR+2) \end{bmatrix}$$

$$\phi(k-1) = \begin{bmatrix} u(k-1) \\ u(k-2) \\ u(k-3) \\ \dots \\ u(k-1-nB) \\ y(k-1) \\ y(k-2) \\ \dots \\ y(k-1-nR) \end{bmatrix} = \begin{bmatrix} M(1) \\ M(2) \\ M(3) \\ \dots \\ M(nB+1) \\ M(nB+2) \\ M(nB+3) \\ \dots \\ M(nB+nR+3) \end{bmatrix}$$

La matriz de ganancias $F(k)$ (de orden igual al del vector $\phi(k-1)$) es representada en el programa por la matriz

$CM(i,j)$, la misma que es inicializada mediante una matriz diagonal ingresando el valor que toman los elementos de la diagonal principal (DIAG), es decir :

$$F(0) = CM_0(i,j) = \text{diag}(DIAG)$$

El factor de olvido se determina, como se indicó anteriormente, utilizando un algoritmo de traza constante. Se representa en el programa mediante la variable FO y su valor es tal que :

$$\text{TRAZA} [CM(i,j)_{\text{act}}] = \text{TRAZA} [CM(i,j)_{\text{ant}}]$$

Consiguientemente se lo calcula en la expresión [IV] de tal forma que la traza de la matriz $CM(i,j)$ actual sea igual a la de la matriz $CM(i,j)$ anterior.

A su vez, la expresión [IV] para el cálculo en el programa es considerada como :

$$CM(i,j)_{k+1} = (1/FO) * \left[CM(i,j)_k - \frac{CM(i,j)_k M(i) M^T(j) CM(i,j)_k}{1 + M^T(i) CM(i,j)_k M(i)} \right]$$

Expresión que se calcula separadamente considerando los siguientes vectores y matrices :

$$D(i)_k = CM(i,j)_k M(i)$$

$$F(i,j)_k = D(i) M^T(i)$$

$$G(i,j)_k = F(i,j)_k CM(i,j)_k$$

y la variable SUM3 equivalente a,

$$SUM3 = 1 + M^T(i)CM(i,j)_k M(i)$$

De tal forma que,

$$CM(i,j)_{k+1} = (1/FO)*[CM(i,j)_k - G(i,j)_k/SUM3]$$

Todas estas acciones están representadas esquemáticamente en el diagrama.

El cálculo de los nuevos estimados, en las variables del programa, se representa como:

$$E(i)_{act} = E(i)_{ant} + FO * CM(i,j)_{k+1} M(i) [Cr(q^{-1})y(k) - E(i)_{ant} M(i)]$$

que se calcula separadamente considerando,

$$SUM2 = FO * CM(i,j)_{k+1} M(i)$$

$$SUM1 = [Cr(q^{-1})y(k) - E(i)_{ant} M(i)]$$

$$E(i)_{act} = E(i)_{ant} + SUM1 * SUM2$$

Una vez que se determinan los nuevos estimados se procede a calcular la salida para el siguiente instante y el nuevo valor de la señal de control aplicando en la ley de

control respectiva aquellos nuevos estimados. El proceso se repite secuencialmente hasta que $k=TMOD$.

Al variar la planta en $TMOD$, la salida de la planta es calculada con los nuevos valores de ella y a partir de este instante se comienzan a determinar nuevos estimados de los parámetros del controlador circulando en cada paso por el proceso anteriormente descrito. Si existe variación en los grados de los polinomios de la planta se recalculan los nuevos índices de los vectores y matrices involucrados en el proceso de cálculo.

El subprograma finaliza cuando se han completado los NR puntos propuestos.

Cuando el retardo es de dos pasos ($d=2$) se presenta una redefinición de los parámetros y de las variables medibles involucradas en el algoritmo. Para este caso, considerando que $nS = d-1 = 1$, se establece la presencia de los coeficientes s_0 y s_1 . Como $s_0=1$ entonces el vector de parámetros se establece según [4.23] como:

$$\theta^T = [b_0 , b_0s_1+b_1 , b_1s_1+b_2 , \dots , b_n s_1 , r_0 , r_1 , \dots , r_{nR}]$$

o equivalentemente:

$$\theta^T = [b_0 , b_{s1} , b_{s2} , \dots , b_{s_{nB+1}} , r_0 , r_1 , \dots , r_{nR}] \quad [I_B]$$

El vector de variables medibles, $\phi^T(k-d)$, que para este

caso es específicamente $\phi^T(k-2)$, según [4.24] está conformado por los siguientes elementos :

$$\phi^T(k-2) = [u(k-2), \dots, u(k-3-nB), y(k-2), \dots, y(k-2-nR)] \quad [\text{II}_B]$$

Las expresiones de estimación a evaluarse para este caso, según las expresiones [4.19] y [4.20], se presentan como :

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \alpha_k F(k+1) \phi(k-2) [Cr(q^{-1})y(k) - \hat{\theta}^T(k-1)\phi(k-2)] \quad [\text{III}_B]$$

$$F(k+1) = \frac{1}{\alpha_k} \left[F(k) - \frac{F(k)\phi(k-2)\phi^T(k-2)F(k)}{1 + \phi^T(k-2)F(k)\phi(k-2)} \right] \quad [\text{IV}_B]$$

$$0 < \alpha_k < 1$$

Los parámetros iniciales del controlador ($\theta^T_{k=0}$) son calculados en base a los datos de la planta inicial y de acuerdo a lo establecido en la acción de control lineal para $d=2$.

De manera análoga a lo efectuado para el caso de retardo unitario, los parámetros del controlador son considerados vectorialmente en el programa y se representan internamente por intermedio del vector $E()$. De igual forma, para el caso de las variables medibles se considera su representación en el vector $M()$.

Consecuentemente, los vectores $E()$ y $M()$ vienen dados por:

$$\theta = \begin{bmatrix} b_0 \\ b_{s1} \\ b_{s2} \\ \dots \\ b_{s_{nB+1}} \\ r_0 \\ r_1 \\ \dots \\ r_{nR} \end{bmatrix} = \begin{bmatrix} E(1) \\ E(2) \\ E(3) \\ \dots \\ E(nB+2) \\ E(nB+3) \\ E(nB+4) \\ \dots \\ E(nB+nR+3) \end{bmatrix}$$

$$\varnothing(k-2) = \begin{bmatrix} u(k-2) \\ u(k-3) \\ u(k-4) \\ \dots \\ u(k-3-nB) \\ y(k-2) \\ y(k-3) \\ \dots \\ y(k-2-nR) \end{bmatrix} = \begin{bmatrix} M(1) \\ M(2) \\ M(3) \\ \dots \\ M(nB+2) \\ M(nB+3) \\ M(nB+4) \\ \dots \\ M(nB+nR+3) \end{bmatrix}$$

Como puede observarse, los índices de los vectores y matrices involucrados en el algoritmo, para este caso, son diferentes. Considerando esta modificación el procedimiento de cálculo es análogo al realizado para el caso de $d-1$, y se

realiza en los mismos términos. Las variables del programa son las mismas y su secuencia de cálculo es la ya considerada.

5.3.3.5 Presentación de resultados

El siguiente paso en la secuencia consiste en el llamado al subprograma RESULTADOS.1 que fue ya considerado en la sección 5.3.2.4. Este a su vez recurre al subprograma ESTADO1 (ver FIG. 5.19), el cual para el caso de MRAC presenta en el bloque correspondiente a "Información" los datos de la simulación efectuada, esto es : datos de la planta inicial y de la modificada, el polinomio de control $Cr(q^{-1})$, el instante de modificación TMOD, el retardo y la inicialización de la matriz de ganancias $F(k)$. La secuencia de presentación incluye en su transcurso el llamado a todos los subprogramas de presentación e impresión considerados con anterioridad.

CAPITULO VI

RESULTADOS Y CONCLUSIONES

6.1 RESULTADOS

6.2 CONCLUSIONES

6.3 RECOMENDACIONES

CAPITULO VI

RESULTADOS Y CONCLUSIONES

6.1. Resultados

Se presentan a continuación los resultados obtenidos en las simulaciones numéricas efectuadas para los casos de control lineal y control adaptivo, respectivamente.

6.1.1 Simulaciones-Control Lineal

Los ejercicios numéricos propuestos para su simulación tienen como propósito fundamental el mostrar la dependencia del comportamiento del controlador propuesto en el capítulo II con respecto a las características de la planta y a la posición de los polos de lazo cerrado dados por el polinomio $C_r(q^{-1})$.

En las simulaciones CL-1, CL-2 y CL-3 el control lineal se aplica a una planta de 2do. orden con un cero y un retardo unitario. Los polinomios A y B respectivos están dados por :

$$A(q^{-1}) = 1 - q^{-1} + 0.24q^{-1}$$

$$B(q^{-1}) = 1 - 0.6q^{-1}$$

La planta así determinada es estable (ceros de A dentro del círculo unitario) y de fase mínima (ceros de B dentro del círculo unitario).

La simulación CL-1 se efectúa considerando $Cr(q^{-1}) = 1$, es decir, se asignan todos los polos al origen. Se observa que la salida $y(k)$ iguala a la referencia $yM(k)$ después del primer período de muestreo, eliminándose en consecuencia el error de salida en tiempo mínimo (d pasos).

En la simulación CL-2 se utiliza un polinomio Cr que tiene un cero en $q=0.5$, es decir, $Cr(q^{-1}) = 1 - 0.5q^{-1}$. En los resultados obtenidos se observa que la salida converge a la referencia establecida, pero, comparativamente con respecto al caso anterior la señal de control requerida es menor y la convergencia es más lenta.

Los resultados de la simulación CL-3 indican que el comportamiento del control empeora cuando los ceros de Cr se ubican en el semiplano izquierdo. Para este caso se escogió un polinomio Cr con un cero en $q=-0.5$, es decir, $Cr(q^{-1}) = 1 + 0.5q^{-1}$. Se observa una señal de control inicialmente oscilatoria y de mayor valor que en los casos anteriores. Existe convergencia en forma oscilatoria de la salida hacia la referencia .

En la simulación CL-4 se aprecia el comportamiento del control cuando éste es aplicado a una planta inestable (al menos un cero de A fuera del círculo unitario). Los polinomios A y B utilizados son respectivamente :

$$A(q^{-1}) = 1 + 1.5q^{-1} - q^{-2}$$

$$B(q^{-1}) = 1 - 0.7q^{-1}$$

Los ceros de A se ubican en $q=2$ y $q=-0.5$. Se observa que el control lineal en consideración cumple con su función consiguiendo que la salida converga a la referencia propuesta. Se aprecia sin embargo que la señal de control es de magnitud relativamente grande.

A continuación, en la simulación CL-5 se aplica el control a un sistema cuyos ceros se ubican en el semiplano izquierdo. Los polinomios empleados son :

$$A(q^{-1}) = 1 - 1.2q^{-1} + 0.35q^{-2}$$

$$B(q^{-1}) = 1 + 0.85q^{-1}$$

El polinomio B presenta un cero en $q=-0.85$. Es notoria en la señal de control la presencia de una marcada tendencia oscilatoria. La salida converge a la referencia según puede apreciarse en los gráficos correspondientes a este caso.

En la simulación CL-6 el control es aplicado a una planta de 3er. orden con un retardo de dos pasos ($d=2$), y, en la simulación CL-7 se utiliza una secuencia de referencia

compleja que es seguida eficientemente por la salida poniendo de relieve las características idóneas del control.

En el ejemplo CL-8 se investiga la respuesta del control cuando es aplicado a una planta de fase no mínima. En concordancia con lo determinado en el análisis teórico se observa que la señal de control diverge y no es acotada, no siendo por tanto aplicable este esquema para estos casos.

Finalmente, en la simulación CL-9 se observa la respuesta del control cuando la planta del sistema experimenta modificaciones apreciables. La planta inicial, en base a la cual se determinan los parámetros del controlador, está dada por los siguientes polinomios:

$$A(q^{-1}) = 1 - q^{-1} + 0.24q^{-2}$$

$$B(q^{-1}) = 1 - 0.2q^{-1}$$

En el instante $k=20$ la planta se modifica, pasando a constituirse por:

$$A(q^{-1}) = 1 - 0.9q^{-1} + 0.08q^{-2}$$

$$B(q^{-1}) = 1 - 0.8q^{-2}$$

Los resultados indican que el objetivo de seguimiento es incumplido a partir del instante en que se modifica la planta.

Posteriormente este mismo ejercicio CL-9 será implementado utilizando la técnica de control adaptivo.

Cabe indicar que las observaciones realizadas son de tipo general para cada uno de los casos considerados, es decir, al efectuarse simulaciones para varios ejemplos correspondientes a cada caso se obtuvieron resultados similares. Consecuentemente, por razones de espacio, se presenta un ejemplo representativo para cada circunstancia de interés.

A continuación aparecen los resultados de las simulaciones consideradas. Se presenta para cada una de ellas los datos respectivos del ejercicio, los resultados gráficos obtenidos para las secuencias de control, referencia y salida, y los resultados numéricos correspondientes :

SIMULACION CL-1

PLANTA

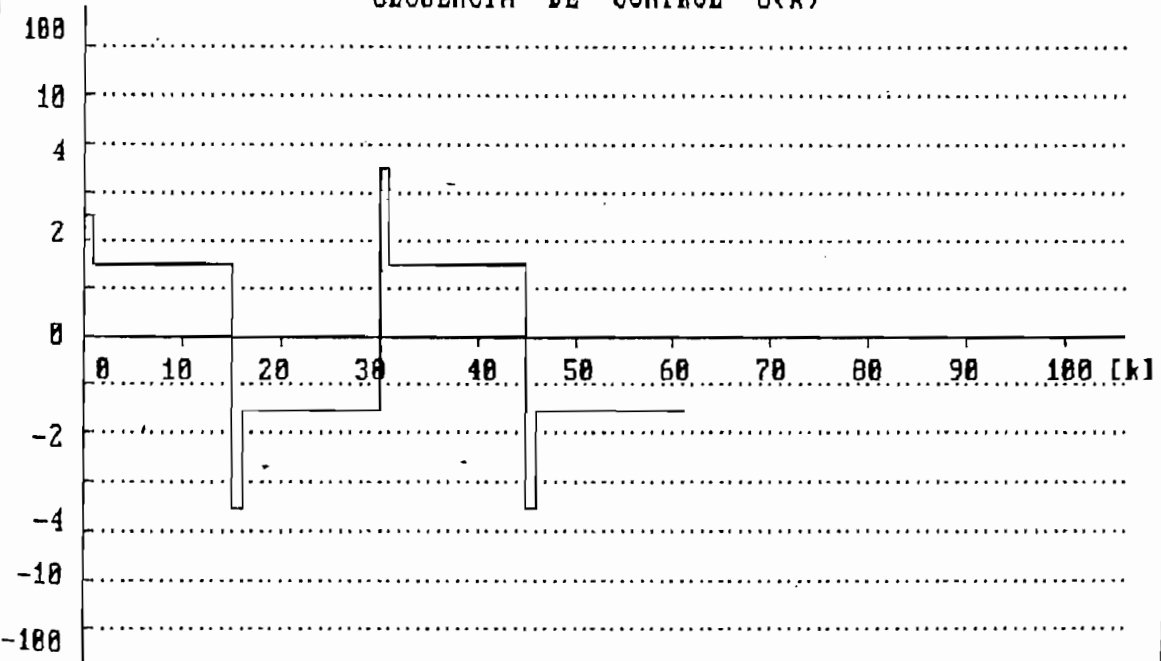
POLINOMIO A : a 1 = -1.000
 a 2 = 0.240

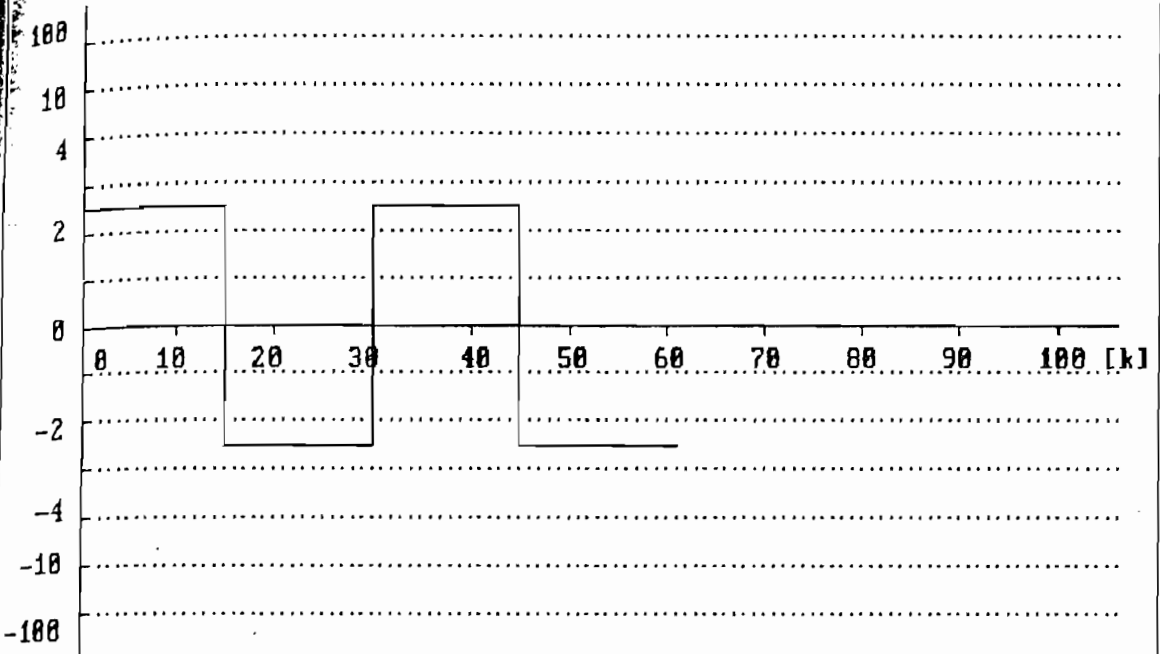
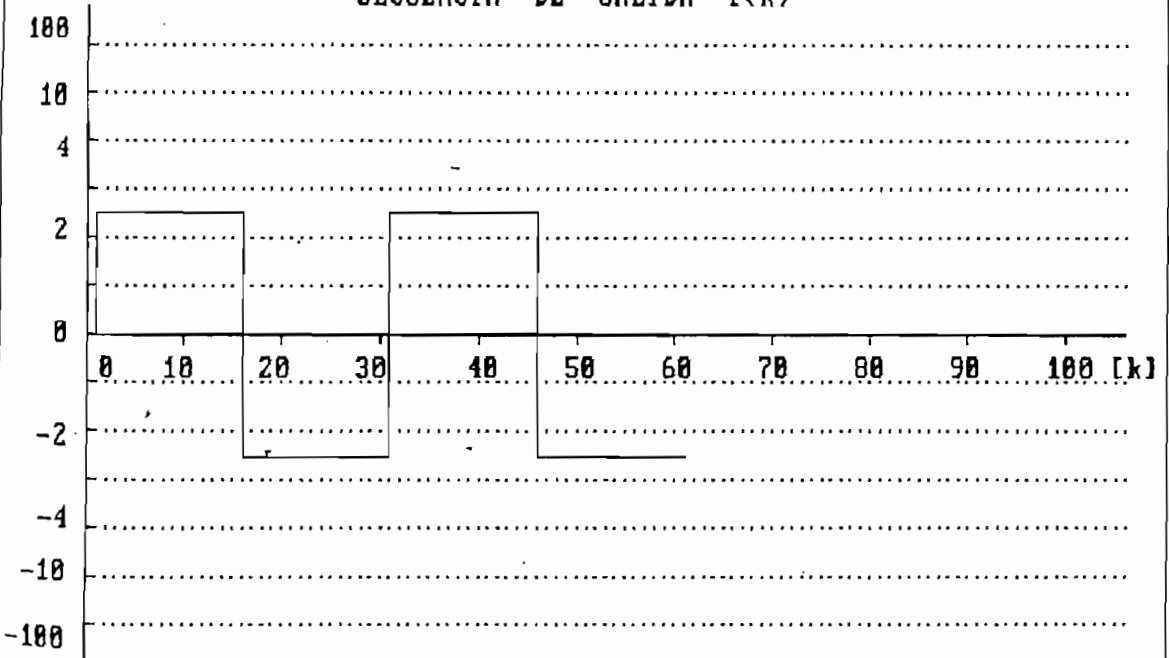
POLINOMIO B : b 0 = 1.000
 b 1 = -0.600

POLINOMIO DE CONTROL C_r

$$C_r = 1$$

$$\text{RETARDO } d = 1$$

SECUENCIA DE CONTROL $U(k)$ 

SECUENCIA DE REFERENCIA $Y_M(k)$ SECUENCIA DE SALIDA $Y(k)$ 

• RESULTADOS NUMERICOS •

YM(k)	U(k)	Y(k)
YM(0) = 2.500000	U(0) = 2.500000	Y(0) = 0.000000
YM(1) = 2.500000	U(1) = 1.500000	Y(1) = 2.500000
YM(2) = 2.500000	U(2) = 1.500000	Y(2) = 2.500000
YM(3) = 2.500000	U(3) = 1.500000	Y(3) = 2.500000
YM(4) = 2.500000	U(4) = 1.500000	Y(4) = 2.500000
YM(5) = 2.500000	U(5) = 1.500000	Y(5) = 2.500000
YM(6) = 2.500000	U(6) = 1.500000	Y(6) = 2.500000
YM(7) = 2.500000	U(7) = 1.500000	Y(7) = 2.500000
YM(8) = 2.500000	U(8) = 1.500000	Y(8) = 2.500000
YM(9) = 2.500000	U(9) = 1.500000	Y(9) = 2.500000
YM(10) = 2.500000	U(10) = 1.500000	Y(10) = 2.500000
YM(11) = 2.500000	U(11) = 1.500000	Y(11) = 2.500000
YM(12) = 2.500000	U(12) = 1.500000	Y(12) = 2.500000
YM(13) = 2.500000	U(13) = 1.500000	Y(13) = 2.500000
YM(14) = 2.500000	U(14) = 1.500000	Y(14) = 2.500000
YM(15) = -2.500000	U(15) = -3.500000	Y(15) = 2.500000
YM(16) = -2.500000	U(16) = -1.500000	Y(16) = -2.500000
YM(17) = -2.500000	U(17) = -1.500000	Y(17) = -2.500000
YM(18) = -2.500000	U(18) = -1.500000	Y(18) = -2.500000
YM(19) = -2.500000	U(19) = -1.500000	Y(19) = -2.500000
YM(20) = -2.500000	U(20) = -1.500000	Y(20) = -2.500000
YM(21) = -2.500000	U(21) = -1.500000	Y(21) = -2.500000
YM(22) = -2.500000	U(22) = -1.500000	Y(22) = -2.500000
YM(23) = -2.500000	U(23) = -1.500000	Y(23) = -2.500000
YM(24) = -2.500000	U(24) = -1.500000	Y(24) = -2.500000
YM(25) = -2.500000	U(25) = -1.500000	Y(25) = -2.500000
YM(26) = -2.500000	U(26) = -1.500000	Y(26) = -2.500000
YM(27) = -2.500000	U(27) = -1.500000	Y(27) = -2.500000
YM(28) = -2.500000	U(28) = -1.500000	Y(28) = -2.500000
YM(29) = -2.500000	U(29) = -1.500000	Y(29) = -2.500000
YM(30) = 2.500000	U(30) = 3.500000	Y(30) = -2.500000
YM(31) = 2.500000	U(31) = 1.500000	Y(31) = 2.500000
YM(32) = 2.500000	U(32) = 1.500000	Y(32) = 2.500000
YM(33) = 2.500000	U(33) = 1.500000	Y(33) = 2.500000
YM(34) = 2.500000	U(34) = 1.500000	Y(34) = 2.500000
YM(35) = 2.500000	U(35) = 1.500000	Y(35) = 2.500000
YM(36) = 2.500000	U(36) = 1.500000	Y(36) = 2.500000
YM(37) = 2.500000	U(37) = 1.500000	Y(37) = 2.500000
YM(38) = 2.500000	U(38) = 1.500000	Y(38) = 2.500000
YM(39) = 2.500000	U(39) = 1.500000	Y(39) = 2.500000
YM(40) = 2.500000	U(40) = 1.500000	Y(40) = 2.500000
YM(41) = 2.500000	U(41) = 1.500000	Y(41) = 2.500000
YM(42) = 2.500000	U(42) = 1.500000	Y(42) = 2.500000
YM(43) = 2.500000	U(43) = 1.500000	Y(43) = 2.500000
YM(44) = 2.500000	U(44) = 1.500000	Y(44) = 2.500000
YM(45) = -2.500000	U(45) = -3.500000	Y(45) = 2.500000
YM(46) = -2.500000	U(46) = -1.500000	Y(46) = -2.500000
YM(47) = -2.500000	U(47) = -1.500000	Y(47) = -2.500000
YM(48) = -2.500000	U(48) = -1.500000	Y(48) = -2.500000
YM(49) = -2.500000	U(49) = -1.500000	Y(49) = -2.500000
YM(50) = -2.500000	U(50) = -1.500000	Y(50) = -2.500000

SIMULACION CL-2

PLANTA

POLINOMIO A : a 1 = -1.000
 a 2 = 0.240

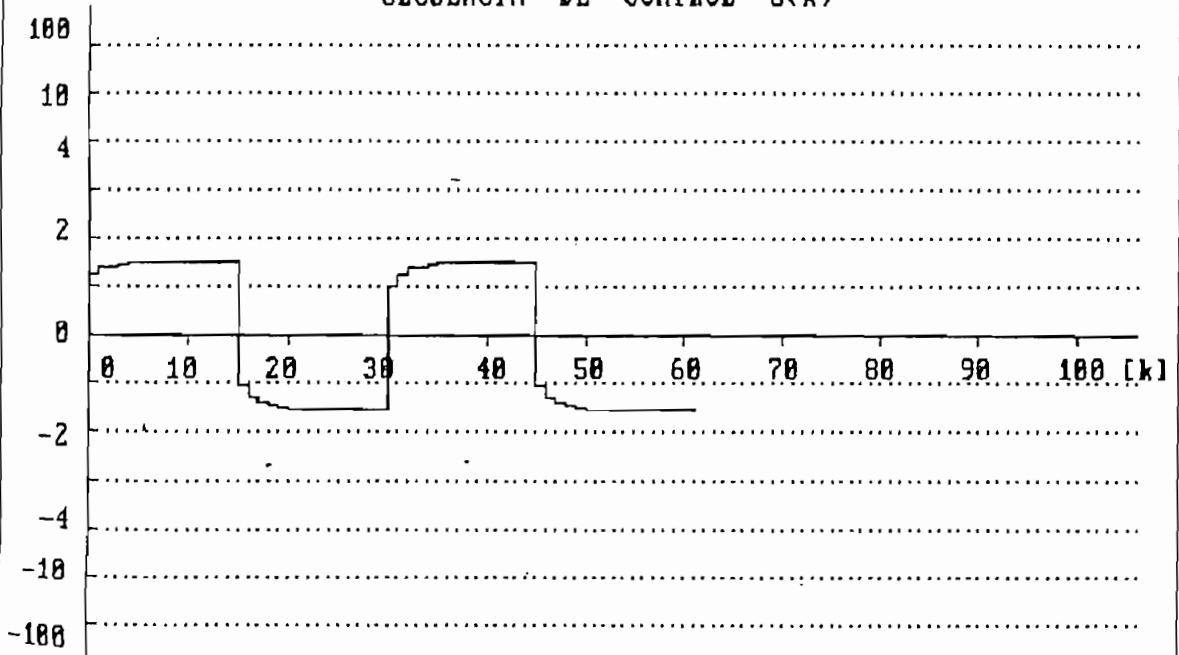
POLINOMIO B : b 0 = 1.000
 b 1 = -0.600

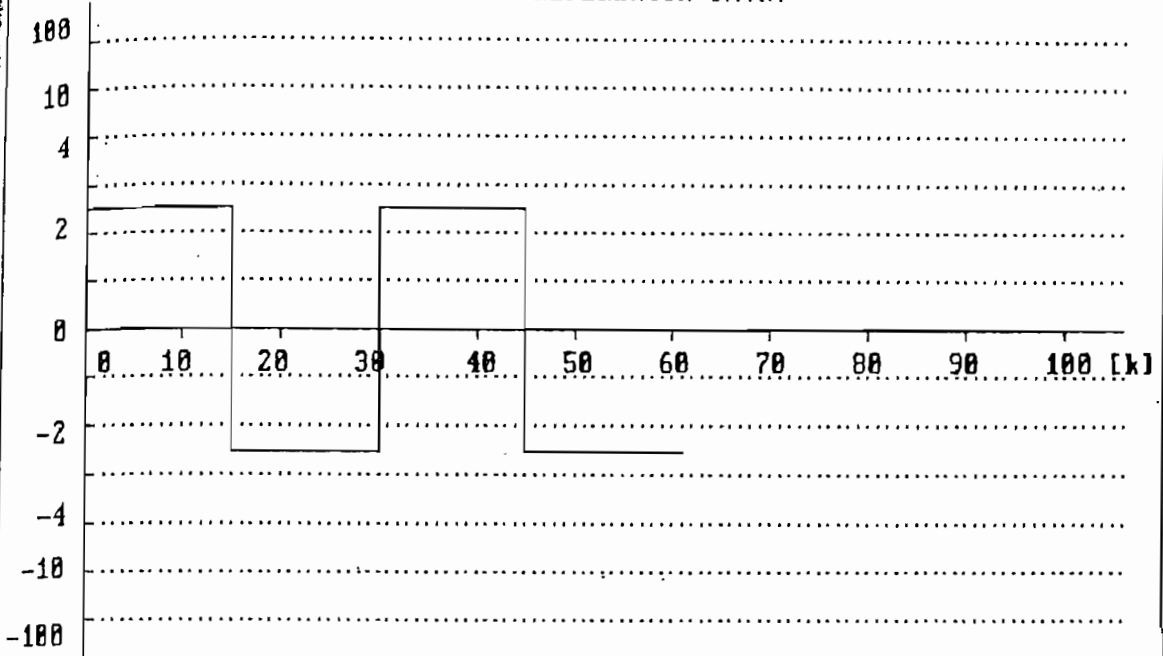
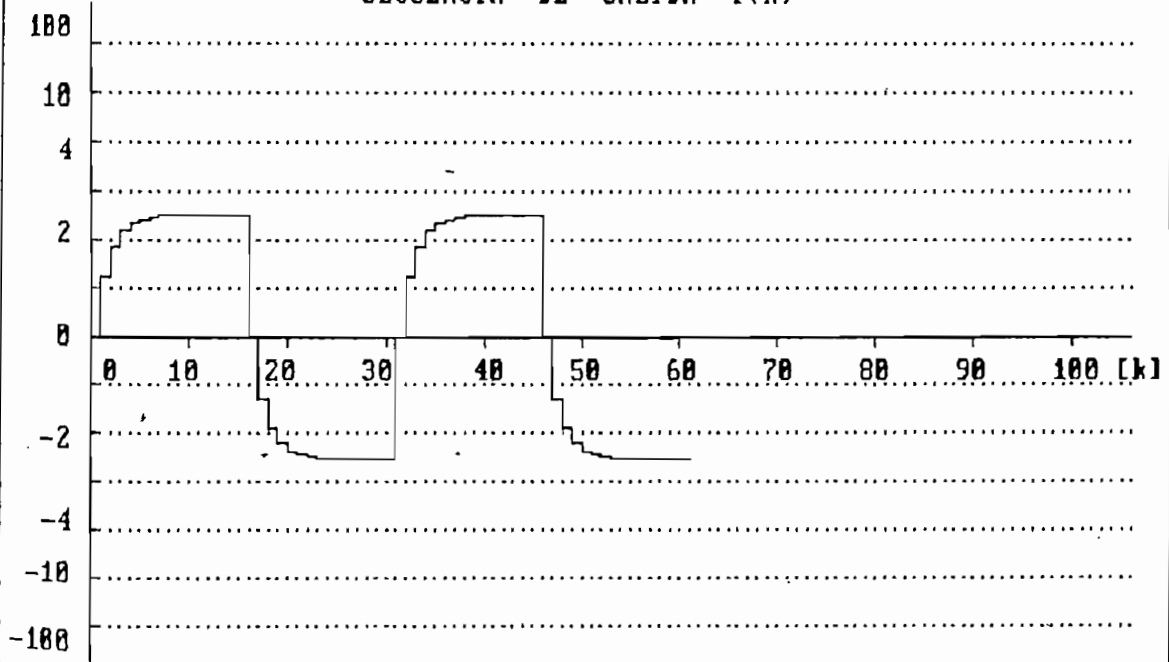
POLINOMIO DE CONTROL Cr

c 1 = -0.500

RETARDO d = 1

SECUENCIA DE CONTROL U(k)



SECUENCIA DE REFERENCIA $Y_M(k)$ SECUENCIA DE SALIDA $Y(k)$ 

• RESULTADOS NUMERICOS •

YM(k)	U(k)	Y(k)
YM(0) = 2.500000	U(0) = 1.250000	Y(0) = 0.000000
YM(1) = 2.500000	U(1) = 1.375000	Y(1) = 1.250000
YM(2) = 2.500000	U(2) = 1.437500	Y(2) = 1.875000
YM(3) = 2.500000	U(3) = 1.468750	Y(3) = 2.187500
YM(4) = 2.500000	U(4) = 1.484375	Y(4) = 2.343750
YM(5) = 2.500000	U(5) = 1.492188	Y(5) = 2.421875
YM(6) = 2.500000	U(6) = 1.496094	Y(6) = 2.460937
YM(7) = 2.500000	U(7) = 1.498047	Y(7) = 2.480469
YM(8) = 2.500000	U(8) = 1.499023	Y(8) = 2.490234
YM(9) = 2.500000	U(9) = 1.499512	Y(9) = 2.495117
YM(10) = 2.500000	U(10) = 1.499756	Y(10) = 2.497559
YM(11) = 2.500000	U(11) = 1.499878	Y(11) = 2.498779
YM(12) = 2.500000	U(12) = 1.499939	Y(12) = 2.499390
YM(13) = 2.500000	U(13) = 1.499969	Y(13) = 2.499695
YM(14) = 2.500000	U(14) = 1.499985	Y(14) = 2.499847
YM(15) = -2.500000	U(15) = -1.000008	Y(15) = 2.499924
YM(16) = -2.500000	U(16) = -1.250004	Y(16) = -0.000038
YM(17) = -2.500000	U(17) = -1.375002	Y(17) = -1.250019
YM(18) = -2.500000	U(18) = -1.437501	Y(18) = -1.875010
YM(19) = -2.500000	U(19) = -1.468750	Y(19) = -2.187505
YM(20) = -2.500000	U(20) = -1.484375	Y(20) = -2.343752
YM(21) = -2.500000	U(21) = -1.492188	Y(21) = -2.421876
YM(22) = -2.500000	U(22) = -1.496094	Y(22) = -2.460938
YM(23) = -2.500000	U(23) = -1.498047	Y(23) = -2.480469
YM(24) = -2.500000	U(24) = -1.499023	Y(24) = -2.490235
YM(25) = -2.500000	U(25) = -1.499512	Y(25) = -2.495117
YM(26) = -2.500000	U(26) = -1.499756	Y(26) = -2.497559
YM(27) = -2.500000	U(27) = -1.499878	Y(27) = -2.498779
YM(28) = -2.500000	U(28) = -1.499939	Y(28) = -2.499390
YM(29) = -2.500000	U(29) = -1.499969	Y(29) = -2.499695
YM(30) = 2.500000	U(30) = 1.000015	Y(30) = -2.499847
YM(31) = 2.500000	U(31) = 1.250008	Y(31) = 0.000076
YM(32) = 2.500000	U(32) = 1.375004	Y(32) = 1.250038
YM(33) = 2.500000	U(33) = 1.437502	Y(33) = 1.875019
YM(34) = 2.500000	U(34) = 1.468751	Y(34) = 2.187510
YM(35) = 2.500000	U(35) = 1.484375	Y(35) = 2.343755
YM(36) = 2.500000	U(36) = 1.492188	Y(36) = 2.421877
YM(37) = 2.500000	U(37) = 1.496094	Y(37) = 2.460939
YM(38) = 2.500000	U(38) = 1.498047	Y(38) = 2.480469
YM(39) = 2.500000	U(39) = 1.499023	Y(39) = 2.490235
YM(40) = 2.500000	U(40) = 1.499512	Y(40) = 2.495117
YM(41) = 2.500000	U(41) = 1.499756	Y(41) = 2.497559
YM(42) = 2.500000	U(42) = 1.499878	Y(42) = 2.498779
YM(43) = 2.500000	U(43) = 1.499939	Y(43) = 2.499390
YM(44) = 2.500000	U(44) = 1.499970	Y(44) = 2.499695
YM(45) = -2.500000	U(45) = -1.000015	Y(45) = 2.499848
YM(46) = -2.500000	U(46) = -1.250008	Y(46) = -0.000076
YM(47) = -2.500000	U(47) = -1.375004	Y(47) = -1.250038
YM(48) = -2.500000	U(48) = -1.437502	Y(48) = -1.875019
YM(49) = -2.500000	U(49) = -1.468751	Y(49) = -2.187509
YM(50) = -2.500000	U(50) = -1.484375	Y(50) = -2.343755

SIMULACION CL-3

PLANTA

POLINOMIO A : a 1 = -1.000

 a 2 = 0.240

POLINOMIO B : b 0 = 1.000

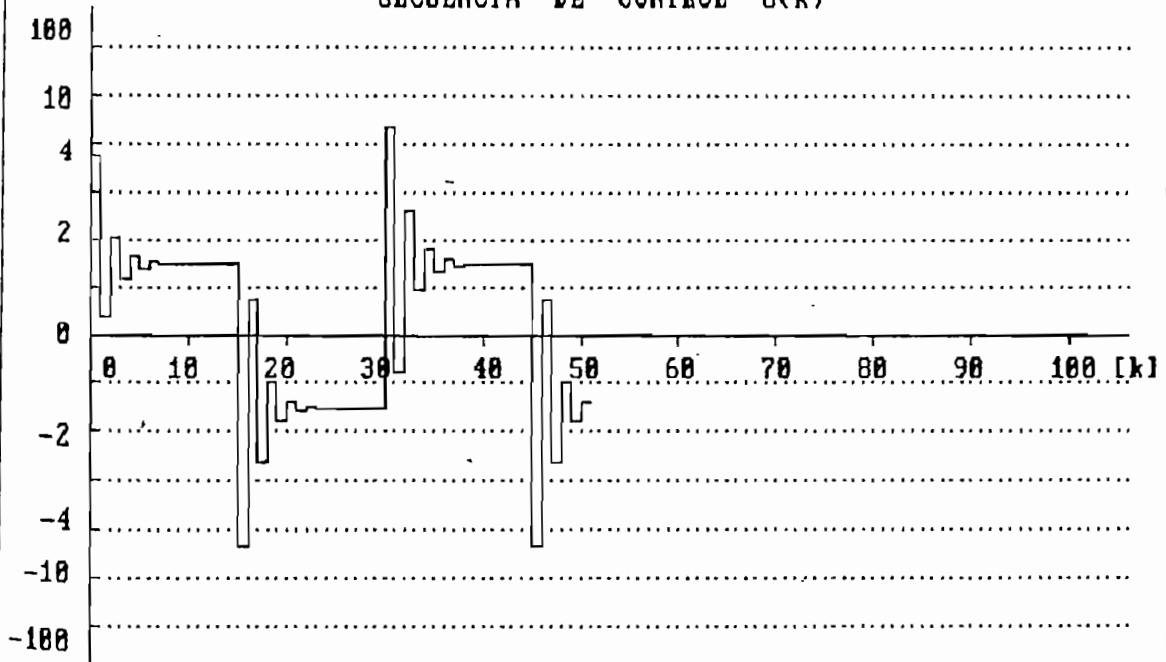
 b 1 = -0.600

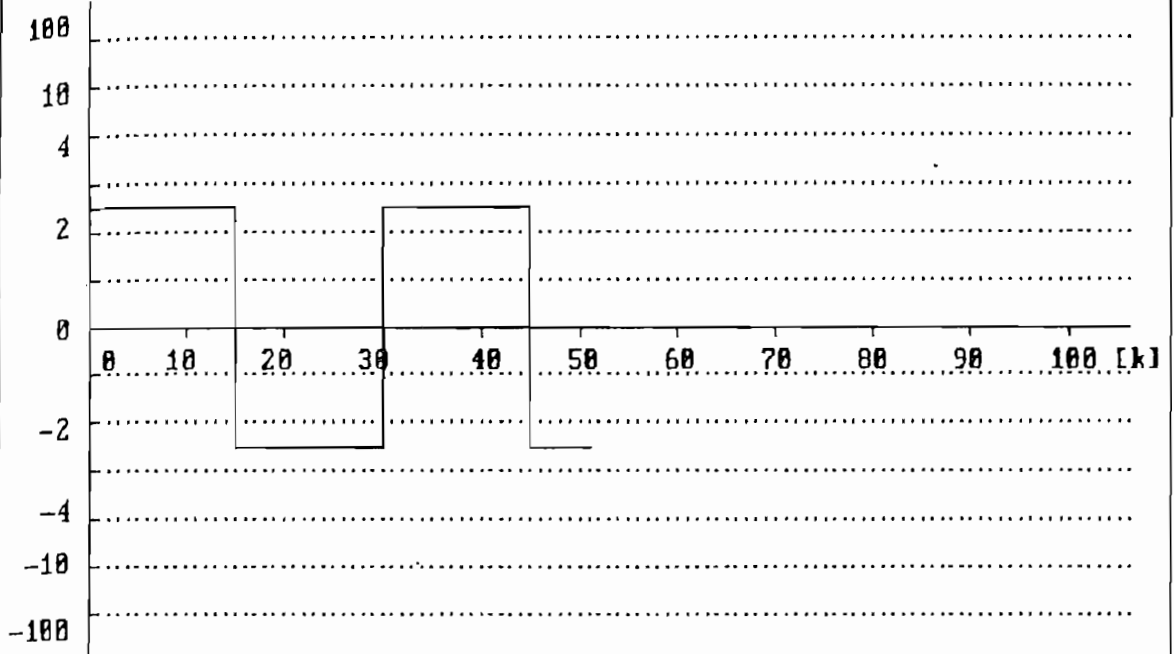
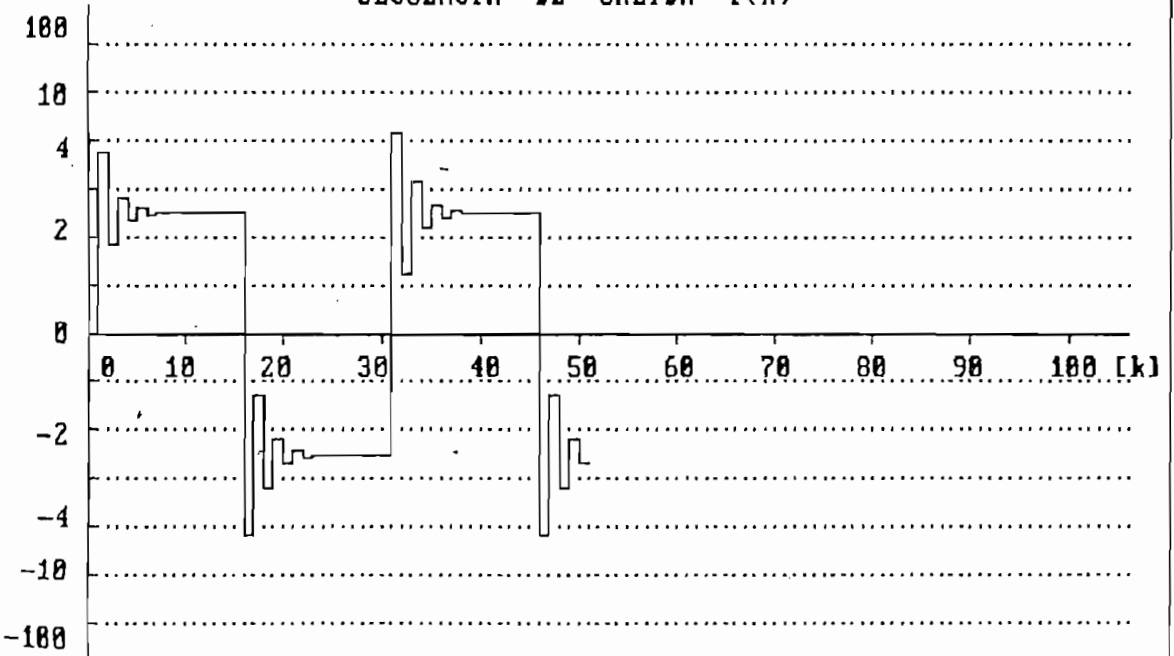
POLINOMIO DE CONTROL Cr

c 1 = 0.500

RETARDO d = 1

SECUENCIA DE CONTROL U(k)



SECUENCIA DE REFERENCIA $Y_M(k)$ SECUENCIA DE SALIDA $Y(k)$ 

• RESULTADOS NUMERICOS •

YM(k)	U(k)	Y(k)
YM(0) = 2.500000	U(0) = 3.750000	Y(0) = 0.000000
YM(1) = 2.500000	U(1) = 0.375000	Y(1) = 3.750000
YM(2) = 2.500000	U(2) = 2.062500	Y(2) = 1.875000
YM(3) = 2.500000	U(3) = 1.218750	Y(3) = 2.812500
YM(4) = 2.500000	U(4) = 1.640625	Y(4) = 2.343750
YM(5) = 2.500000	U(5) = 1.429688	Y(5) = 2.578125
YM(6) = 2.500000	U(6) = 1.535156	Y(6) = 2.460938
YM(7) = 2.500000	U(7) = 1.482422	Y(7) = 2.519531
YM(8) = 2.500000	U(8) = 1.508789	Y(8) = 2.490235
YM(9) = 2.500000	U(9) = 1.495606	Y(9) = 2.504883
YM(10) = 2.500000	U(10) = 1.502197	Y(10) = 2.497559
YM(11) = 2.500000	U(11) = 1.498901	Y(11) = 2.501221
YM(12) = 2.500000	U(12) = 1.500549	Y(12) = 2.499390
YM(13) = 2.500000	U(13) = 1.499725	Y(13) = 2.500305
YM(14) = 2.500000	U(14) = 1.500137	Y(14) = 2.499847
YM(15) = -2.500000	U(15) = -6.000069	Y(15) = 2.500076
YM(16) = -2.500000	U(16) = 0.750034	Y(16) = -5.000038
YM(17) = -2.500000	U(17) = -2.625017	Y(17) = -1.249981
YM(18) = -2.500000	U(18) = -0.937492	Y(18) = -3.125010
YM(19) = -2.500000	U(19) = -1.781254	Y(19) = -2.187495
YM(20) = -2.500000	U(20) = -1.359373	Y(20) = -2.656252
YM(21) = -2.500000	U(21) = -1.570313	Y(21) = -2.421874
YM(22) = -2.500000	U(22) = -1.464843	Y(22) = -2.539063
YM(23) = -2.500000	U(23) = -1.517578	Y(23) = -2.480468
YM(24) = -2.500000	U(24) = -1.491211	Y(24) = -2.509766
YM(25) = -2.500000	U(25) = -1.504395	Y(25) = -2.495117
YM(26) = -2.500000	U(26) = -1.497803	Y(26) = -2.502441
YM(27) = -2.500000	U(27) = -1.501099	Y(27) = -2.498779
YM(28) = -2.500000	U(28) = -1.499450	Y(28) = -2.500610
YM(29) = -2.500000	U(29) = -1.500275	Y(29) = -2.499695
YM(30) = 2.500000	U(30) = 6.000138	Y(30) = -2.500153
YM(31) = 2.500000	U(31) = -0.750069	Y(31) = 5.000077
YM(32) = 2.500000	U(32) = 2.625035	Y(32) = 1.249961
YM(33) = 2.500000	U(33) = 0.937482	Y(33) = 3.125019
YM(34) = 2.500000	U(34) = 1.781259	Y(34) = 2.187490
YM(35) = 2.500000	U(35) = 1.359371	Y(35) = 2.656255
YM(36) = 2.500000	U(36) = 1.570315	Y(36) = 2.421873
YM(37) = 2.500000	U(37) = 1.464843	Y(37) = 2.539064
YM(38) = 2.500000	U(38) = 1.517579	Y(38) = 2.480468
YM(39) = 2.500000	U(39) = 1.491211	Y(39) = 2.509766
YM(40) = 2.500000	U(40) = 1.504395	Y(40) = 2.495117
YM(41) = 2.500000	U(41) = 1.497803	Y(41) = 2.502441
YM(42) = 2.500000	U(42) = 1.501099	Y(42) = 2.498779
YM(43) = 2.500000	U(43) = 1.499451	Y(43) = 2.500610
YM(44) = 2.500000	U(44) = 1.500275	Y(44) = 2.499695
YM(45) = -2.500000	U(45) = -6.000137	Y(45) = 2.500153
YM(46) = -2.500000	U(46) = 0.750068	Y(46) = -5.000076
YM(47) = -2.500000	U(47) = -2.625034	Y(47) = -1.249962
YM(48) = -2.500000	U(48) = -0.937483	Y(48) = -3.125019
YM(49) = -2.500000	U(49) = -1.781259	Y(49) = -2.187491
YM(50) = -2.500000	U(50) = -1.359371	Y(50) = -2.656255

SIMULACION CL-4

PLANTA

POLINOMIO A : a 1 = 1.500
 a 2 = -1.000

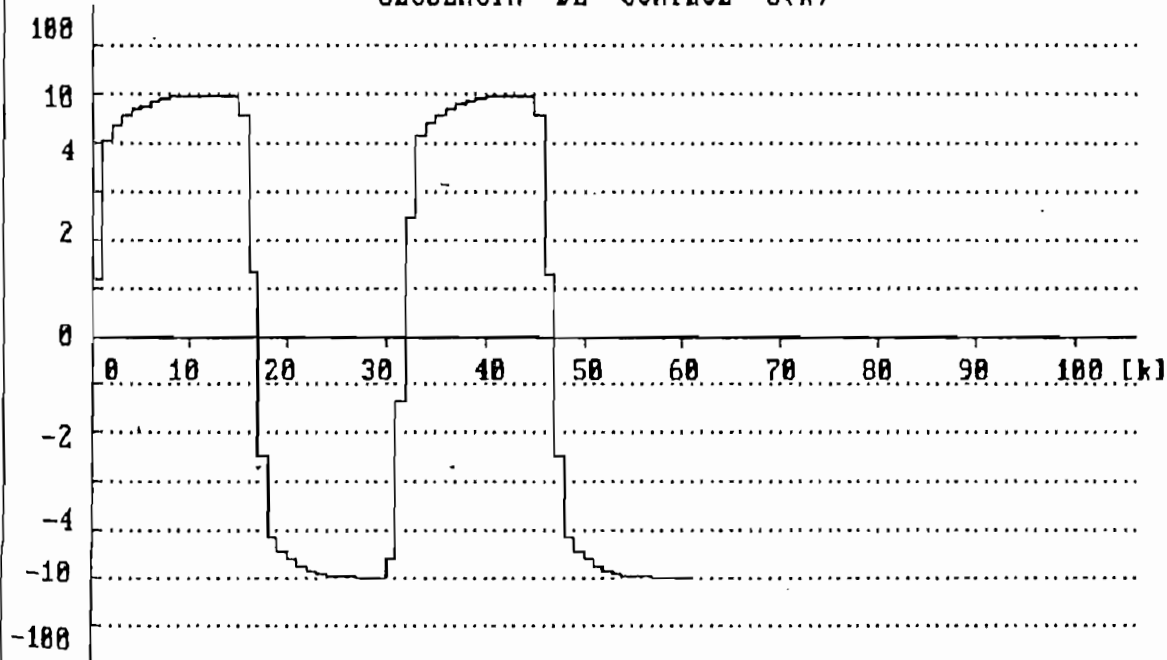
POLINOMIO B : b 0 = 1.000
 b 1 = -0.700

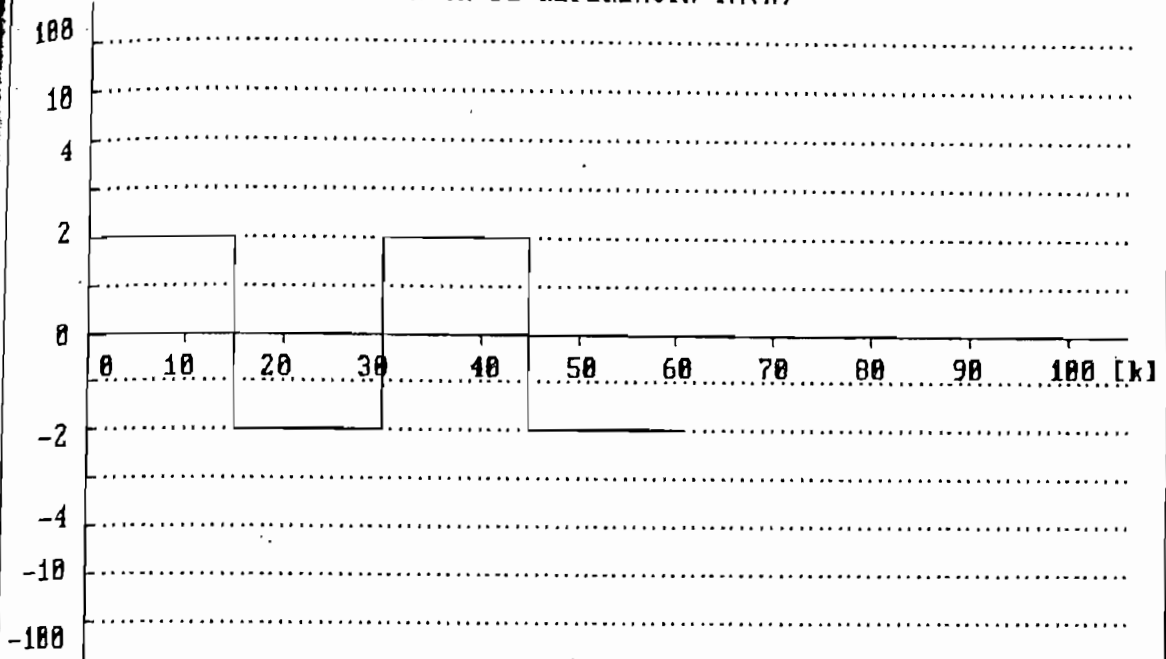
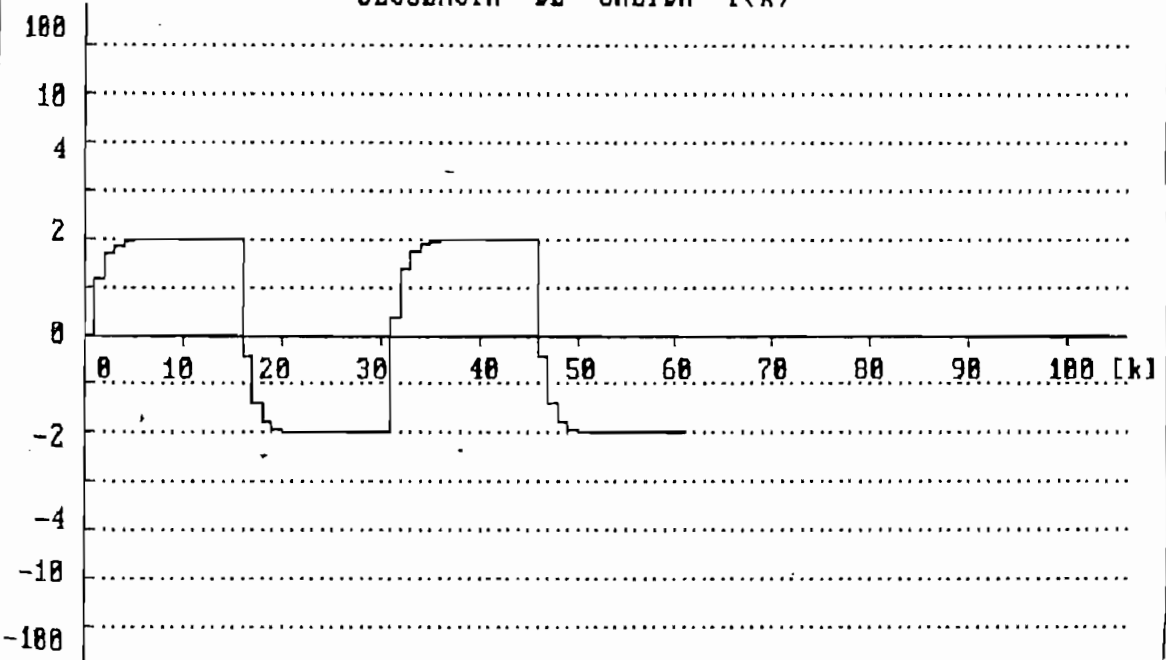
POLINOMIO DE CONTROL Cr

c 1 = -0.400

RETARDO d = 1

SECUENCIA DE CONTROL U(k)



SECUENCIA DE REFERENCIA $Y_M(k)$ SECUENCIA DE SALIDA $Y(k)$ 

• RESULTADOS NUMERICOS •

YM(k)	U(k)	Y(k)
YM(0) = 2.000000	U(0) = 1.200000	Y(0) = 0.000000
YM(1) = 2.000000	U(1) = 4.320000	Y(1) = 1.200000
YM(2) = 2.000000	U(2) = 6.216001	Y(2) = 1.680000
YM(3) = 2.000000	U(3) = 7.428000	Y(3) = 1.872000
YM(4) = 2.000000	U(4) = 8.230320	Y(4) = 1.948800
YM(5) = 2.000000	U(5) = 8.773512	Y(5) = 1.979520
YM(6) = 2.000000	U(6) = 9.146374	Y(6) = 1.991808
YM(7) = 2.000000	U(7) = 9.404427	Y(7) = 1.996723
YM(8) = 2.000000	U(8) = 9.583885	Y(8) = 1.998689
YM(9) = 2.000000	U(9) = 9.709034	Y(9) = 1.999475
YM(10) = 2.000000	U(10) = 9.796452	Y(10) = 1.999791
YM(11) = 2.000000	U(11) = 9.857567	Y(11) = 1.999917
YM(12) = 2.000000	U(12) = 9.900316	Y(12) = 1.999966
YM(13) = 2.000000	U(13) = 9.930230	Y(13) = 1.999987
YM(14) = 2.000000	U(14) = 9.951164	Y(14) = 1.999995
YM(15) = -2.000000	U(15) = 7.565817	Y(15) = 1.999998
YM(16) = -2.000000	U(16) = 1.336072	Y(16) = -0.400001
YM(17) = -2.000000	U(17) = -2.448750	Y(17) = -1.360001
YM(18) = -2.000000	U(18) = -4.867725	Y(18) = -1.744000
YM(19) = -2.000000	U(19) = -6.468847	Y(19) = -1.897600
YM(20) = -2.000000	U(20) = -7.552769	Y(20) = -1.959040
YM(21) = -2.000000	U(21) = -8.296768	Y(21) = -1.983616
YM(22) = -2.000000	U(22) = -8.811668	Y(22) = -1.993446
YM(23) = -2.000000	U(23) = -9.169741	Y(23) = -1.997378
YM(24) = -2.000000	U(24) = -9.419447	Y(24) = -1.998951
YM(25) = -2.000000	U(25) = -9.593863	Y(25) = -1.999580
YM(26) = -2.000000	U(26) = -9.715805	Y(26) = -1.999832
YM(27) = -2.000000	U(27) = -9.801104	Y(27) = -1.999933
YM(28) = -2.000000	U(28) = -9.860788	Y(28) = -1.999973
YM(29) = -2.000000	U(29) = -9.902557	Y(29) = -1.999989
YM(30) = 2.000000	U(30) = -7.531791	Y(30) = -1.999995
YM(31) = 2.000000	U(31) = -1.312255	Y(31) = 0.400002
YM(32) = 2.000000	U(32) = 2.465421	Y(32) = 1.360001
YM(33) = 2.000000	U(33) = 4.879394	Y(33) = 1.744000
YM(34) = 2.000000	U(34) = 6.477015	Y(34) = 1.897600
YM(35) = 2.000000	U(35) = 7.558487	Y(35) = 1.959040
YM(36) = 2.000000	U(36) = 8.300772	Y(36) = 1.983616
YM(37) = 2.000000	U(37) = 8.814473	Y(37) = 1.993447
YM(38) = 2.000000	U(38) = 9.171705	Y(38) = 1.997379
YM(39) = 2.000000	U(39) = 9.420824	Y(39) = 1.998952
YM(40) = 2.000000	U(40) = 9.594829	Y(40) = 1.999581
YM(41) = 2.000000	U(41) = 9.716480	Y(41) = 1.999832
YM(42) = 2.000000	U(42) = 9.801576	Y(42) = 1.999933
YM(43) = 2.000000	U(43) = 9.861119	Y(43) = 1.999973
YM(44) = 2.000000	U(44) = 9.902790	Y(44) = 1.999990
YM(45) = -2.000000	U(45) = 7.531954	Y(45) = 1.999995
YM(46) = -2.000000	U(46) = 1.312369	Y(46) = -0.400002
YM(47) = -2.000000	U(47) = -2.465341	Y(47) = -1.360001
YM(48) = -2.000000	U(48) = -4.879339	Y(48) = -1.744000
YM(49) = -2.000000	U(49) = -6.476977	Y(49) = -1.897600
YM(50) = -2.000000	U(50) = -7.558461	Y(50) = -1.959040

SIMULACION CL-5

PLANTA

POLINOMIO A : a 1 = -1.200
 a 2 = 0.350

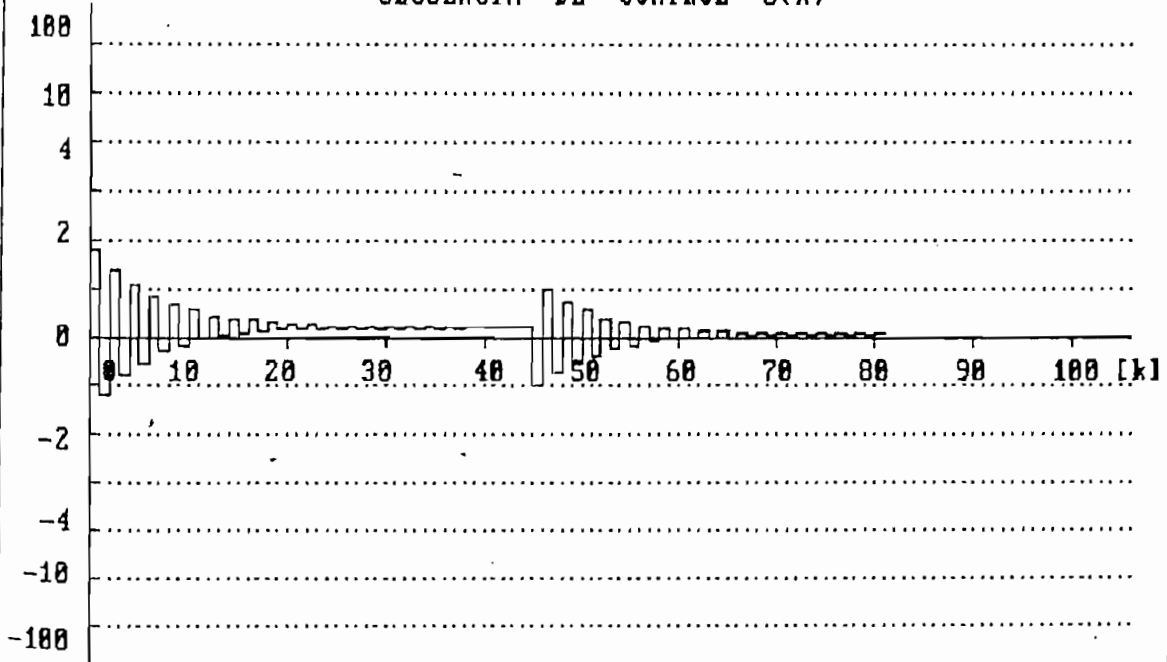
POLINOMIO B : b 0 = 1.000
 b 1 = 0.850

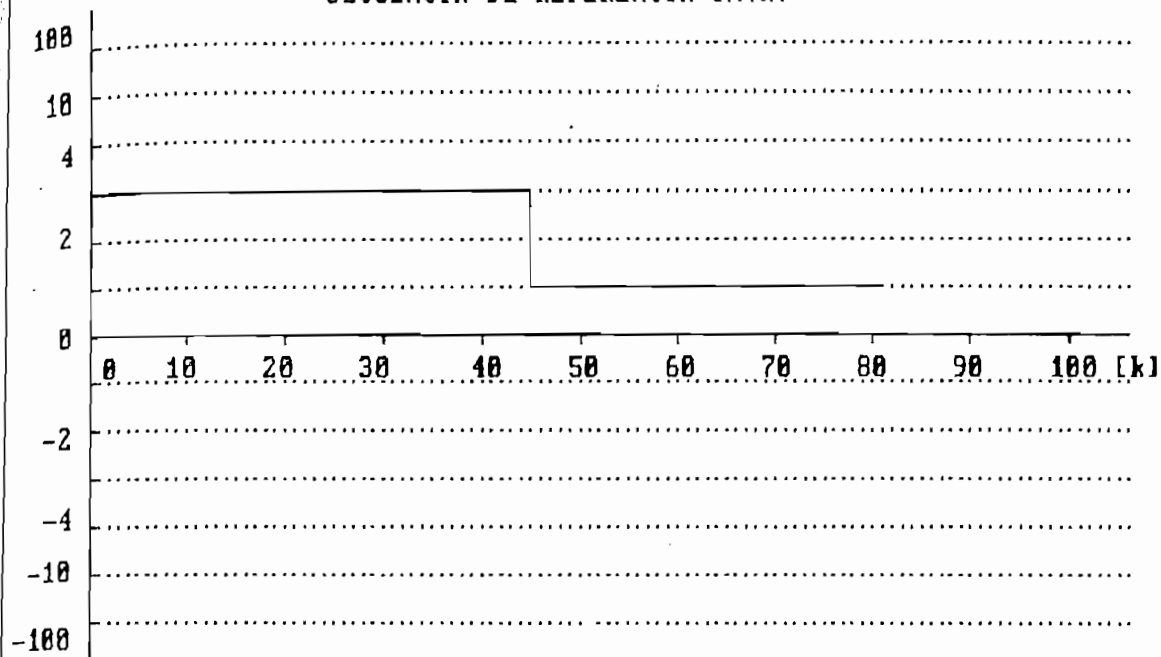
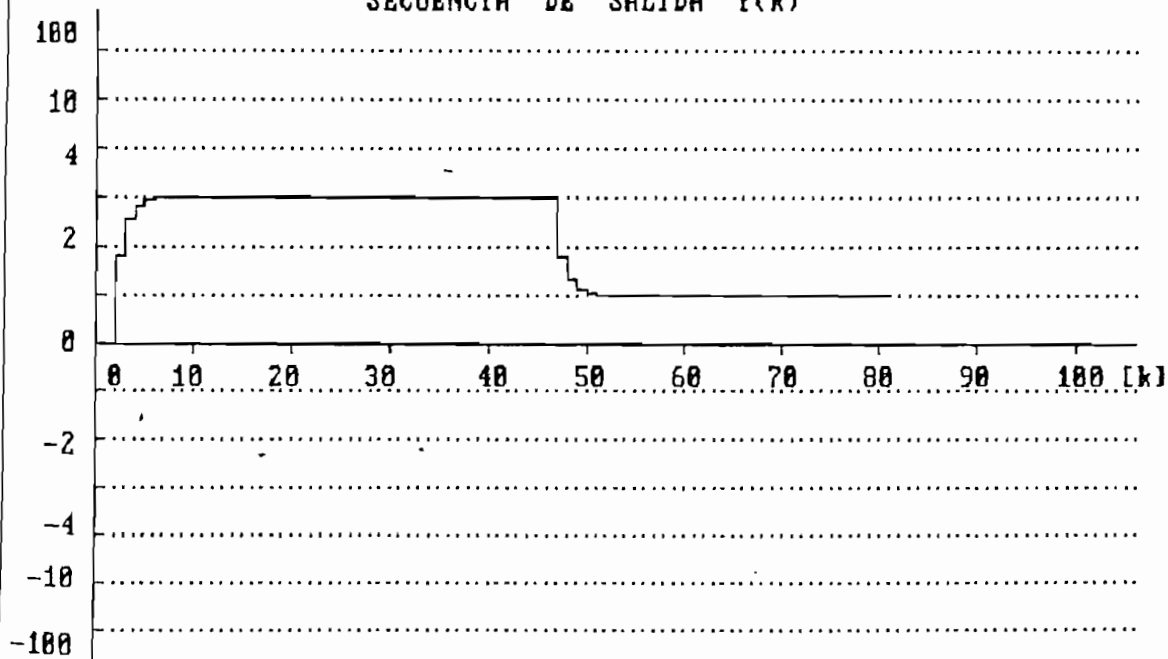
POLINOMIO DE CONTROL Cr

c 1 = -0.400

RETARDO d = 2

SECUENCIA DE CONTROL U(k)



SECUENCIA DE REFERENCIA $Y_M(k)$ SECUENCIA DE SALIDA $Y(k)$ 

• RESULTADOS NUMERICOS •

YM(k)	U(k)	Y(k)
YM(0) = 3.000000	U(0) = 1.800000	Y(0) = 0.000000
YM(1) = 3.000000	U(1) = -1.170000	Y(1) = 0.000000
YM(2) = 3.000000	U(2) = 1.408500	Y(2) = 1.800000
YM(3) = 3.000000	U(3) = -0.761625	Y(3) = 2.520000
YM(4) = 3.000000	U(4) = 1.091621	Y(4) = 2.808000
YM(5) = 3.000000	U(5) = -0.480182	Y(5) = 2.923200
YM(6) = 3.000000	U(6) = 0.857233	Y(6) = 2.969280
YM(7) = 3.000000	U(7) = -0.279017	Y(7) = 2.987712
YM(8) = 3.000000	U(8) = 0.687017	Y(8) = 2.995085
YM(9) = 3.000000	U(9) = -0.134023	Y(9) = 2.998034
YM(10) = 3.000000	U(10) = 0.563896	Y(10) = 2.999213
YM(11) = 3.000000	U(11) = -0.029321	Y(11) = 2.999685
YM(12) = 3.000000	U(12) = 0.474919	Y(12) = 2.999874
YM(13) = 3.000000	U(13) = 0.046317	Y(13) = 2.999949
YM(14) = 3.000000	U(14) = 0.410630	Y(14) = 2.999980
YM(15) = 3.000000	U(15) = 0.100964	Y(15) = 2.999992
YM(16) = 3.000000	U(16) = 0.364180	Y(16) = 2.999997
YM(17) = 3.000000	U(17) = 0.140447	Y(17) = 2.999999
YM(18) = 3.000000	U(18) = 0.330620	Y(18) = 2.999999
YM(19) = 3.000000	U(19) = 0.168972	Y(19) = 3.000000
YM(20) = 3.000000	U(20) = 0.306373	Y(20) = 3.000000
YM(21) = 3.000000	U(21) = 0.189583	Y(21) = 3.000000
YM(22) = 3.000000	U(22) = 0.288855	Y(22) = 3.000000
YM(23) = 3.000000	U(23) = 0.204473	Y(23) = 3.000000
YM(24) = 3.000000	U(24) = 0.276197	Y(24) = 3.000000
YM(25) = 3.000000	U(25) = 0.215232	Y(25) = 3.000000
YM(26) = 3.000000	U(26) = 0.267053	Y(26) = 3.000000
YM(27) = 3.000000	U(27) = 0.223005	Y(27) = 3.000000
YM(28) = 3.000000	U(28) = 0.260445	Y(28) = 3.000000
YM(29) = 3.000000	U(29) = 0.228621	Y(29) = 3.000000
YM(30) = 3.000000	U(30) = 0.255672	Y(30) = 3.000000
YM(31) = 3.000000	U(31) = 0.232679	Y(31) = 3.000000
YM(32) = 3.000000	U(32) = 0.252223	Y(32) = 3.000000
YM(33) = 3.000000	U(33) = 0.233610	Y(33) = 3.000000
YM(34) = 3.000000	U(34) = 0.249731	Y(34) = 3.000000
YM(35) = 3.000000	U(35) = 0.237728	Y(35) = 3.000000
YM(36) = 3.000000	U(36) = 0.247931	Y(36) = 3.000000
YM(37) = 3.000000	U(37) = 0.239259	Y(37) = 3.000000
YM(38) = 3.000000	U(38) = 0.246630	Y(38) = 3.000000
YM(39) = 3.000000	U(39) = 0.240365	Y(39) = 3.000000
YM(40) = 3.000000	U(40) = 0.245690	Y(40) = 3.000000
YM(41) = 3.000000	U(41) = 0.241163	Y(41) = 3.000000
YM(42) = 3.000000	U(42) = 0.245011	Y(42) = 3.000000
YM(43) = 3.000000	U(43) = 0.241741	Y(43) = 3.000000
YM(44) = 3.000000	U(44) = 0.244520	Y(44) = 3.000000
YM(45) = 1.000000	U(45) = -0.957842	Y(45) = 3.000000
YM(46) = 1.000000	U(46) = 1.024166	Y(46) = 3.000000
YM(47) = 1.000000	U(47) = -0.696541	Y(47) = 1.800000
YM(48) = 1.000000	U(48) = 0.751660	Y(48) = 1.320000
YM(49) = 1.000000	U(49) = -0.485071	Y(49) = 1.128000
YM(50) = 1.000000	U(50) = 0.563846	Y(50) = 1.051200

SIMULACION CL-6

PLANTA

POLINOMIO A : a 1 = 1.000
 a 2 = 1.000
 a 3 = 1.000

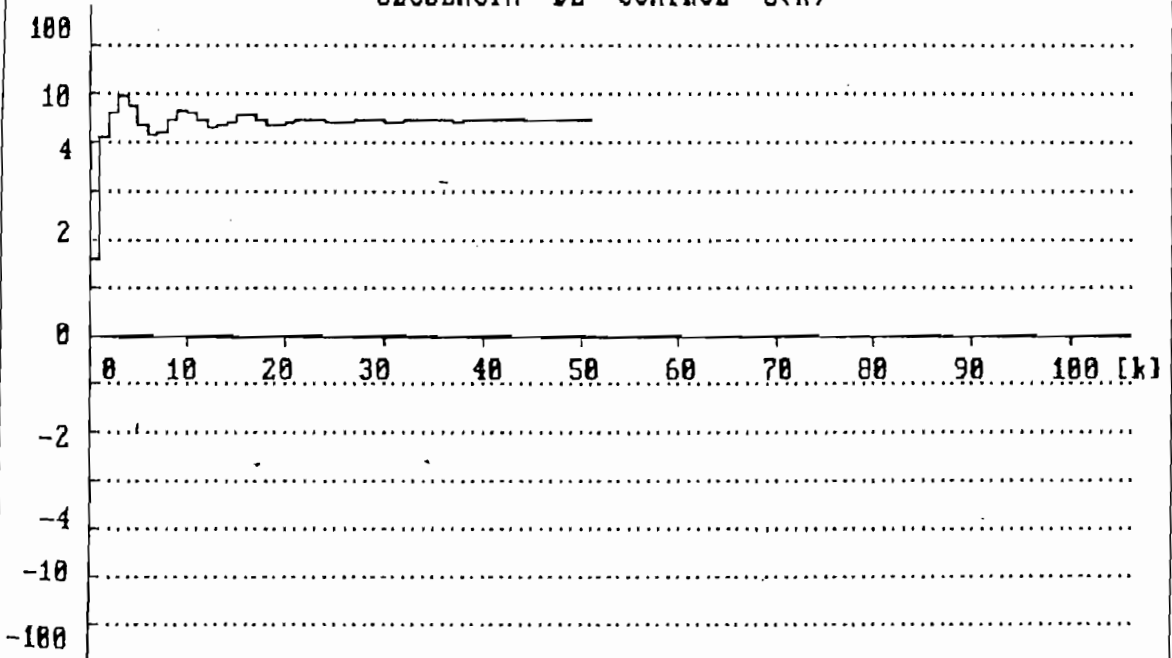
POLINOMIO B : b 0 = 1.000
 b 1 = -0.500
 b 2 = 0.400
 b 3 = 0.300

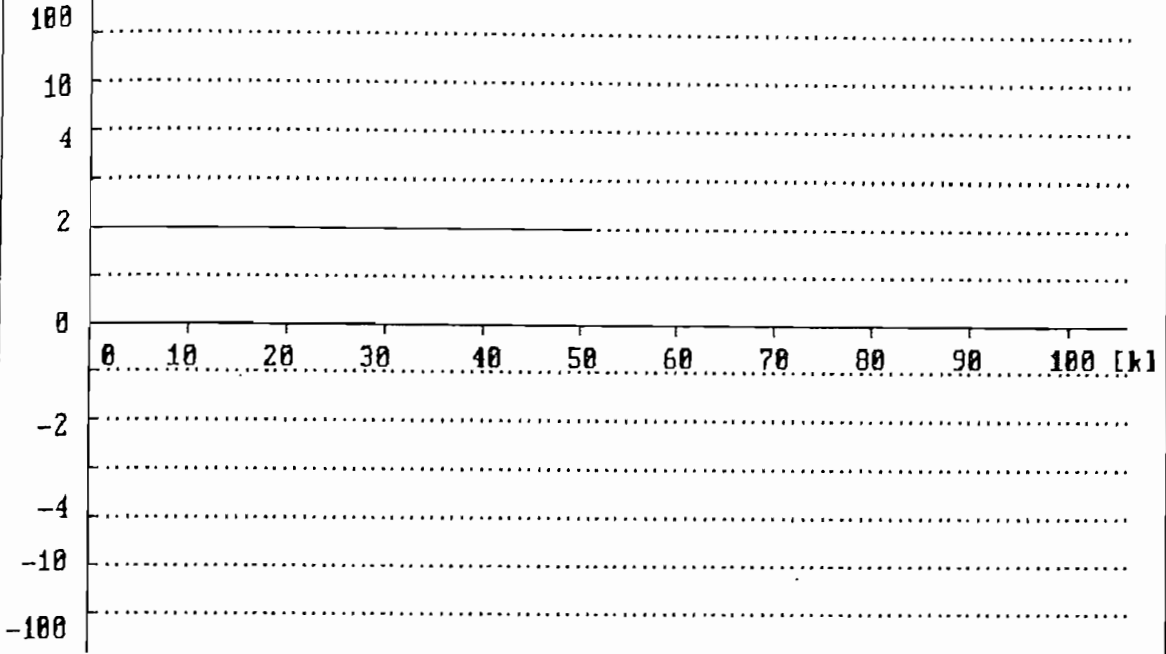
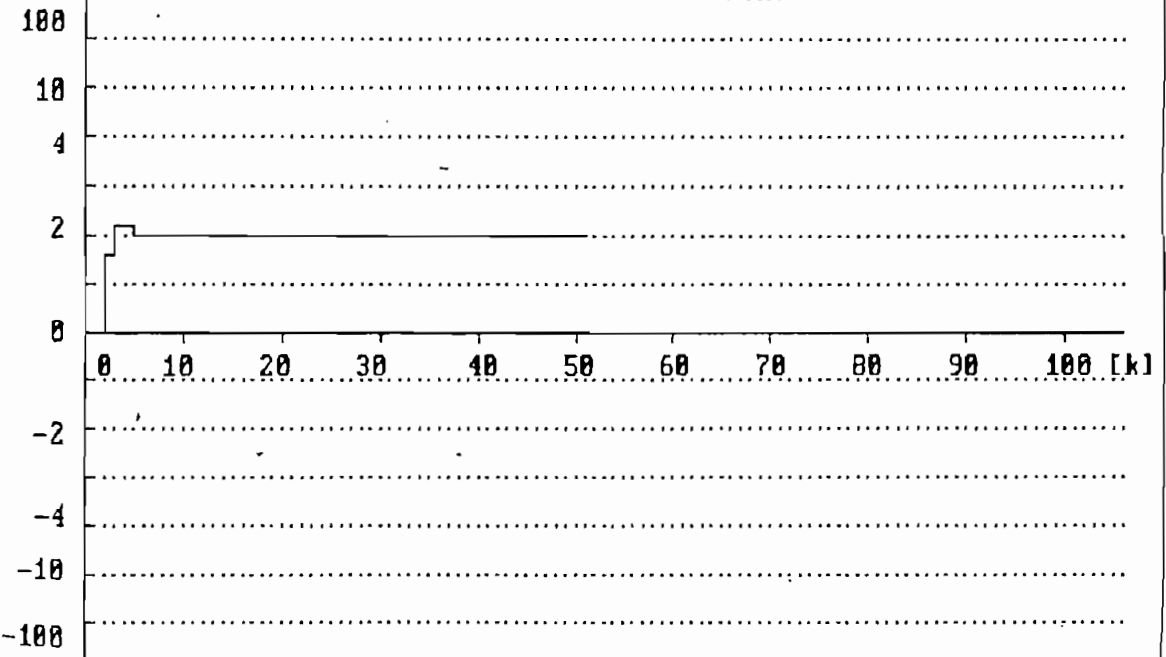
POLINOMIO DE CONTROL Cr

c 1 = -0.400
 c 2 = 0.200

RETARDO d = 2

SECUENCIA DE CONTROL U(k)



SECUENCIA DE REFERENCIA $Y_M(k)$ SECUENCIA DE SALIDA $Y(k)$ 

• RESULTADOS NUMERICOS •

YM(k)	U(k)	Y(k)
YM(0) = 2.000000	U(0) = 1.600000	Y(0) = 0.000000
YM(1) = 2.000000	U(1) = 4.640000	Y(1) = 0.000000
YM(2) = 2.000000	U(2) = 7.696000	Y(2) = 1.600000
YM(3) = 2.000000	U(3) = 9.550400	Y(3) = 2.240000
YM(4) = 2.000000	U(4) = 8.716960	Y(4) = 2.176000
YM(5) = 2.000000	U(5) = 6.386703	Y(5) = 2.022400
YM(6) = 2.000000	U(6) = 4.821888	Y(6) = 1.973760
YM(7) = 2.000000	U(7) = 5.201914	Y(7) = 1.985024
YM(8) = 2.000000	U(8) = 6.744397	Y(8) = 1.999257
YM(9) = 2.000000	U(9) = 7.848001	Y(9) = 2.002698
YM(10) = 2.000000	U(10) = 7.669281	Y(10) = 2.001228
YM(11) = 2.000000	U(11) = 6.672937	Y(11) = 1.999951
YM(12) = 2.000000	U(12) = 5.913959	Y(12) = 1.999735
YM(13) = 2.000000	U(13) = 5.986696	Y(13) = 1.999903
YM(14) = 2.000000	U(14) = 6.625832	Y(14) = 2.000014
YM(15) = 2.000000	U(15) = 7.144094	Y(15) = 2.000024
YM(16) = 2.000000	U(16) = 7.125732	Y(16) = 2.000008
YM(17) = 2.000000	U(17) = 6.717481	Y(17) = 1.999998
YM(18) = 2.000000	U(18) = 6.365214	Y(18) = 1.999997
YM(19) = 2.000000	U(19) = 6.357893	Y(19) = 1.999999
YM(20) = 2.000000	U(20) = 6.617616	Y(20) = 2.000001
YM(21) = 2.000000	U(21) = 6.856087	Y(21) = 2.000001
YM(22) = 2.000000	U(22) = 6.873628	Y(22) = 1.999999
YM(23) = 2.000000	U(23) = 6.709092	Y(23) = 1.999999
YM(24) = 2.000000	U(24) = 6.548266	Y(24) = 1.999999
YM(25) = 2.000000	U(25) = 6.528406	Y(25) = 2.000000
YM(26) = 2.000000	U(26) = 6.632166	Y(26) = 1.999999
YM(27) = 2.000000	U(27) = 6.740238	Y(27) = 1.999999
YM(28) = 2.000000	U(28) = 6.758728	Y(28) = 1.999999
YM(29) = 2.000000	U(29) = 6.693618	Y(29) = 1.999999
YM(30) = 2.000000	U(30) = 6.621245	Y(30) = 2.000000
YM(31) = 2.000000	U(31) = 6.605556	Y(31) = 2.000001
YM(32) = 2.000000	U(32) = 6.646194	Y(32) = 1.999999
YM(33) = 2.000000	U(33) = 6.694499	Y(33) = 2.000000
YM(34) = 2.000000	U(34) = 6.707103	Y(34) = 2.000000
YM(35) = 2.000000	U(35) = 6.681892	Y(35) = 2.000000
YM(36) = 2.000000	U(36) = 6.649753	Y(36) = 2.000000
YM(37) = 2.000000	U(37) = 6.639987	Y(37) = 2.000000
YM(38) = 2.000000	U(38) = 6.655523	Y(38) = 1.999999
YM(39) = 2.000000	U(39) = 6.676840	Y(39) = 2.000000
YM(40) = 2.000000	U(40) = 6.684214	Y(40) = 2.000000
YM(41) = 2.000000	U(41) = 6.674712	Y(41) = 2.000000
YM(42) = 2.000000	U(42) = 6.660616	Y(42) = 1.999999
YM(43) = 2.000000	U(43) = 6.655157	Y(43) = 2.000000
YM(44) = 2.000000	U(44) = 6.660917	Y(44) = 1.999999
YM(45) = 2.000000	U(45) = 6.670208	Y(45) = 2.000000
YM(46) = 2.000000	U(46) = 6.674189	Y(46) = 2.000000
YM(47) = 2.000000	U(47) = 6.670734	Y(47) = 1.999999
YM(48) = 2.000000	U(48) = 6.664628	Y(48) = 2.000000
YM(49) = 2.000000	U(49) = 6.661764	Y(49) = 2.000000
YM(50) = 2.000000	U(50) = 6.663810	Y(50) = 2.000001

SIMULACION CL-7

PLANTA

POLINOMIO A : a 1 = -1.200
 a 2 = 0.350

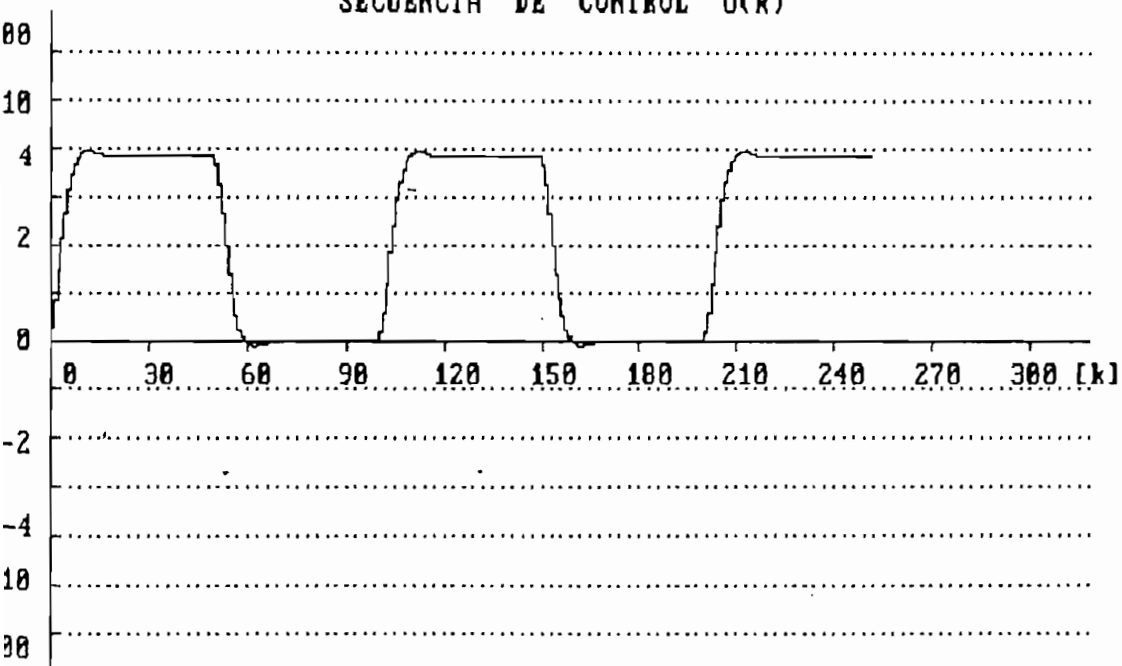
POLINOMIO B : b 0 = 1.000
 b 1 = -0.700

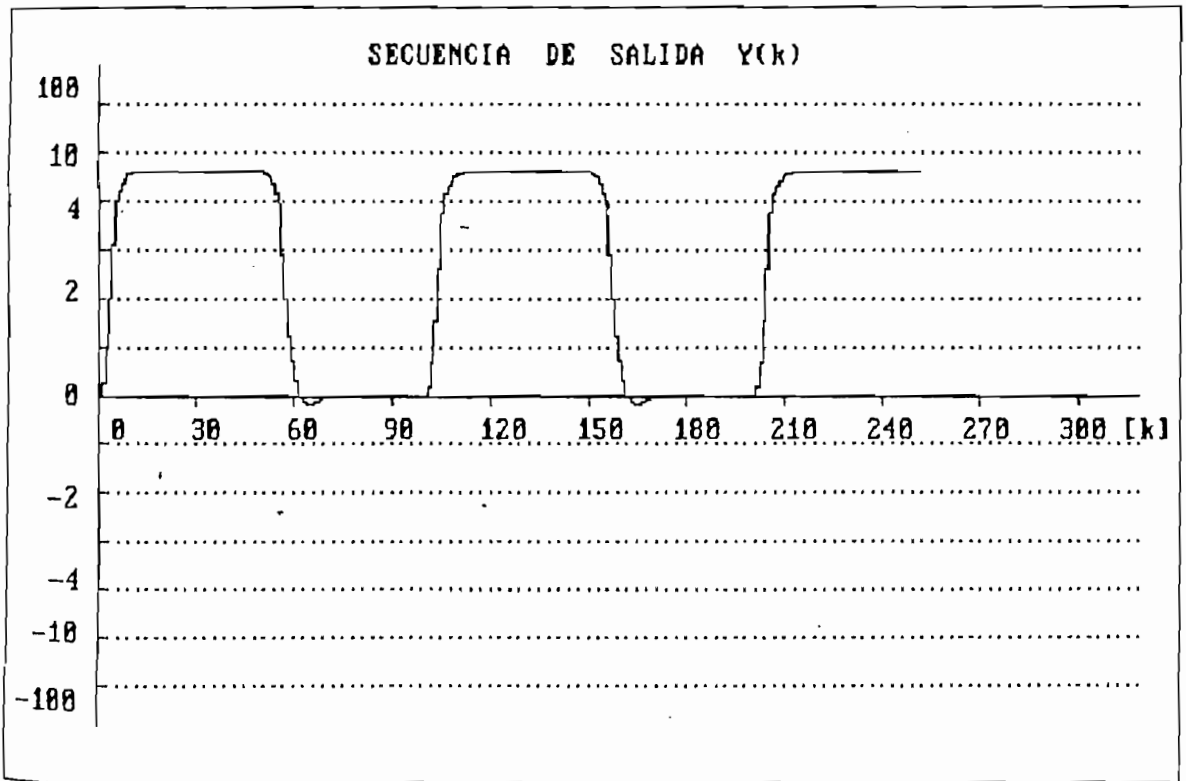
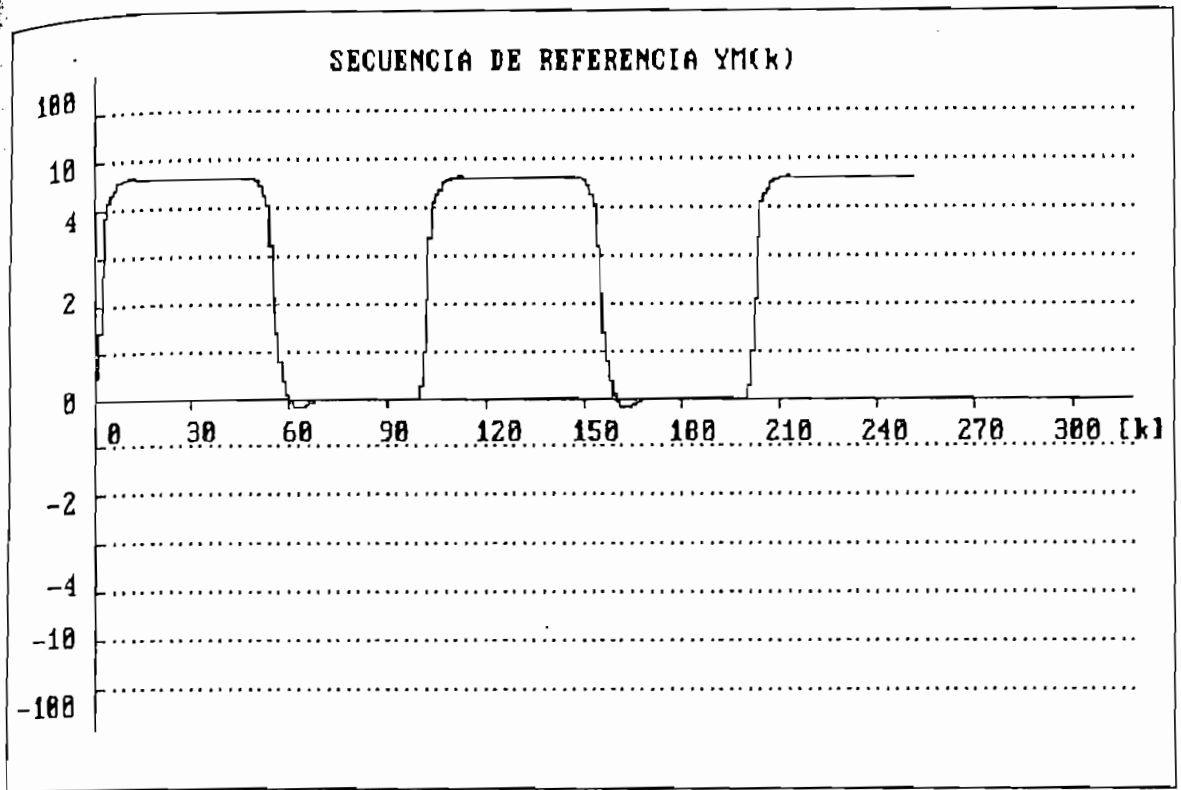
POLINOMIO DE CONTROL Cr

c 1 = -0.400

RETARDO d = 1

SECUENCIA DE CONTROL U(k)





• RESULTADOS NUMERICOS •

YM(k)	U(k)	Y(k)
YM(0) = 0.500000	U(0) = 0.300000	Y(0) = 0.000000
YM(1) = 1.450000	U(1) = 0.840000	Y(1) = 0.300000
YM(2) = 2.640000	U(2) = 1.485000	Y(2) = 0.990000
YM(3) = 3.865000	U(3) = 2.121000	Y(3) = 1.980000
YM(4) = 4.980550	U(4) = 2.677230	Y(4) = 3.111000
YM(5) = 5.908695	U(5) = 3.121943	Y(5) = 4.232730
YM(6) = 6.624668	U(6) = 3.450969	Y(6) = 5.238309
YM(7) = 7.139019	U(7) = 3.676399	Y(7) = 6.070124
YM(8) = 7.481598	U(8) = 3.817812	Y(8) = 6.711461
YM(9) = 7.689579	U(9) = 3.896392	Y(9) = 7.173544
YM(10) = 7.799674	U(10) = 3.931486	Y(10) = 7.483165
YM(11) = 7.843822	U(11) = 3.938984	Y(11) = 7.673070
YM(12) = 7.847400	U(12) = 3.930886	Y(12) = 7.775521
YM(13) = 7.829072	U(13) = 3.915576	Y(13) = 7.818648
YM(14) = 7.801547	U(14) = 3.898436	Y(14) = 7.824902
YM(15) = 7.772740	U(15) = 3.882553	Y(15) = 7.810889
YM(16) = 7.747008	U(16) = 3.869403	Y(16) = 7.787999
YM(17) = 7.726254	U(17) = 3.859411	Y(17) = 7.763404
YM(18) = 7.710839	U(18) = 3.852390	Y(18) = 7.741114
YM(19) = 7.700259	U(19) = 3.847859	Y(19) = 7.722949
YM(20) = 7.693616	U(20) = 3.845235	Y(20) = 7.709335
YM(21) = 7.689925	U(21) = 3.843963	Y(21) = 7.699904
YM(22) = 7.688278	U(22) = 3.843573	Y(22) = 7.693916
YM(23) = 7.687929	U(23) = 3.843702	Y(23) = 7.690533
YM(24) = 7.688313	U(24) = 3.844089	Y(24) = 7.688970
YM(25) = 7.689035	U(25) = 3.844562	Y(25) = 7.688575
YM(26) = 7.689843	U(26) = 3.845019	Y(26) = 7.688851
YM(27) = 7.690590	U(27) = 3.845408	Y(27) = 7.689446
YM(28) = 7.691208	U(28) = 3.845710	Y(28) = 7.690132
YM(29) = 7.691678	U(29) = 3.845927	Y(29) = 7.690778
YM(30) = 7.692008	U(30) = 3.846071	Y(30) = 7.691317
YM(31) = 7.692222	U(31) = 3.846158	Y(31) = 7.691731
YM(32) = 7.692346	U(32) = 3.846203	Y(32) = 7.692025
YM(33) = 7.692406	U(33) = 3.846220	Y(33) = 7.692217
YM(34) = 7.692426	U(34) = 3.846221	Y(34) = 7.692330
YM(35) = 7.692421	U(35) = 3.846213	Y(35) = 7.692388
YM(36) = 7.692403	U(36) = 3.846199	Y(36) = 7.692408
YM(37) = 7.692380	U(37) = 3.846186	Y(37) = 7.692404
YM(38) = 7.692358	U(38) = 3.846174	Y(38) = 7.692390
YM(39) = 7.692339	U(39) = 3.846165	Y(39) = 7.692371
YM(40) = 7.692325	U(40) = 3.846158	Y(40) = 7.692352
YM(41) = 7.692314	U(41) = 3.846153	Y(41) = 7.692335
YM(42) = 7.692307	U(42) = 3.846150	Y(42) = 7.692322
YM(43) = 7.692302	U(43) = 3.846148	Y(43) = 7.692313
YM(44) = 7.692299	U(44) = 3.846147	Y(44) = 7.692306
YM(45) = 7.692298	U(45) = 3.846148	Y(45) = 7.692302
YM(46) = 7.692298	U(46) = 3.846147	Y(46) = 7.692300
YM(47) = 7.692298	U(47) = 3.846148	Y(47) = 7.692298
YM(48) = 7.692299	U(48) = 3.846148	Y(48) = 7.692298
YM(49) = 7.692300	U(49) = 3.846148	Y(49) = 7.692298
YM(50) = 7.412301	U(50) = 3.678148	Y(50) = 7.692299

SIMULACION CL-B

PLANTA

POLINOMIO A : a 1 = -1.200
 a 2 = 0.350

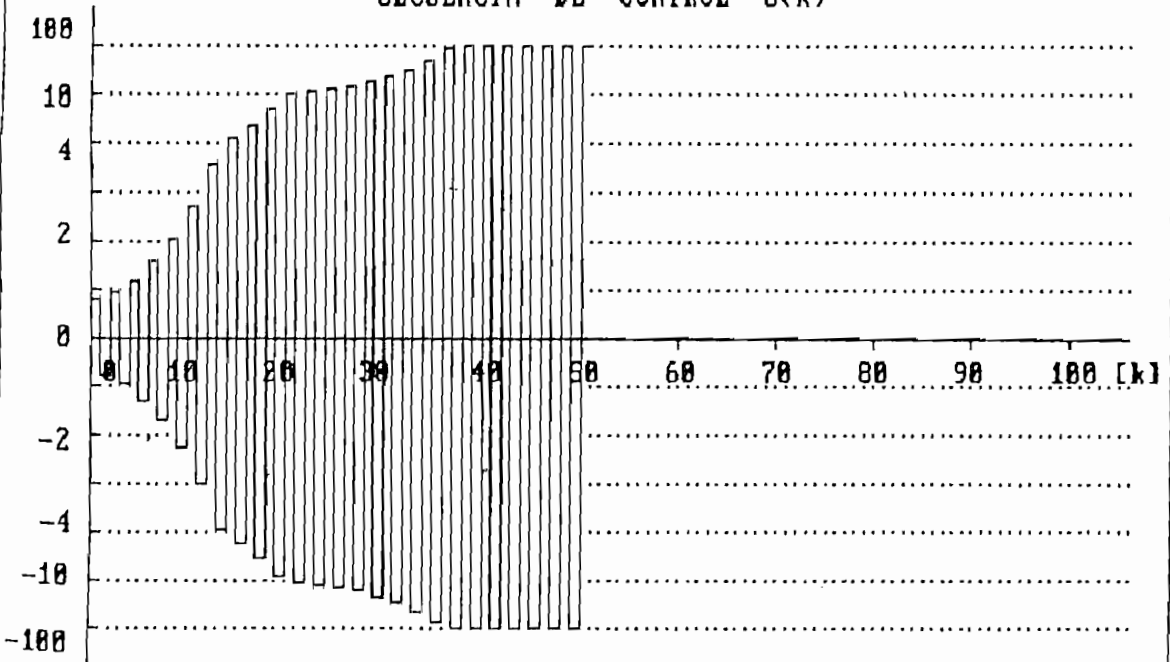
POLINOMIO B : b 0 = 1.000
 b 1 = 1.200

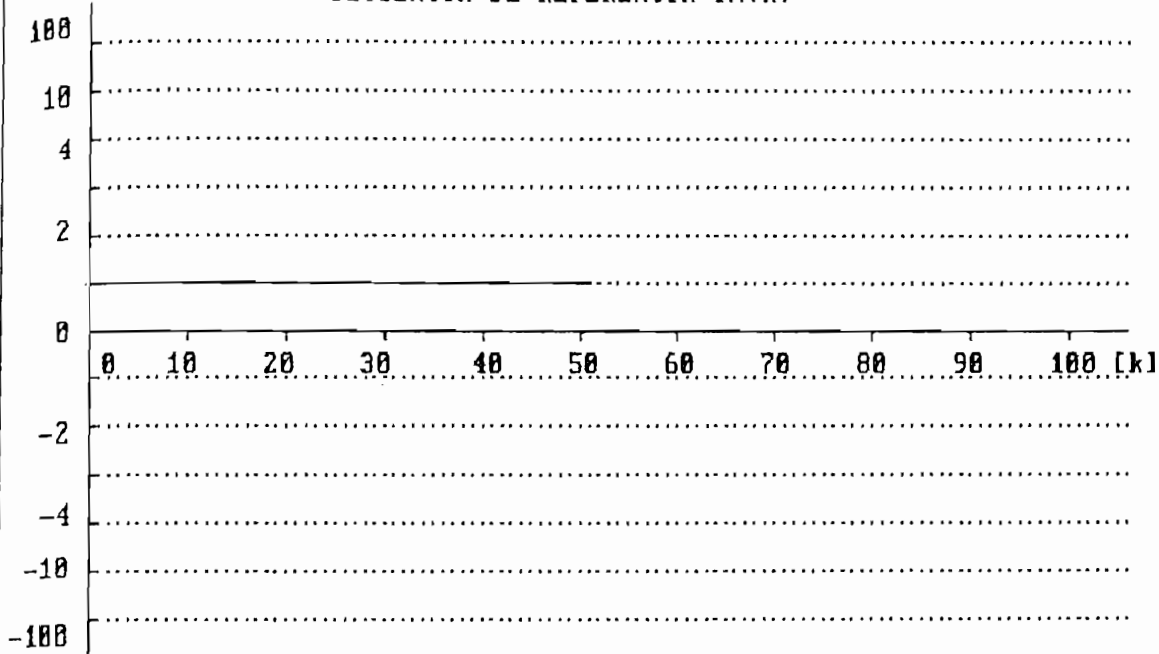
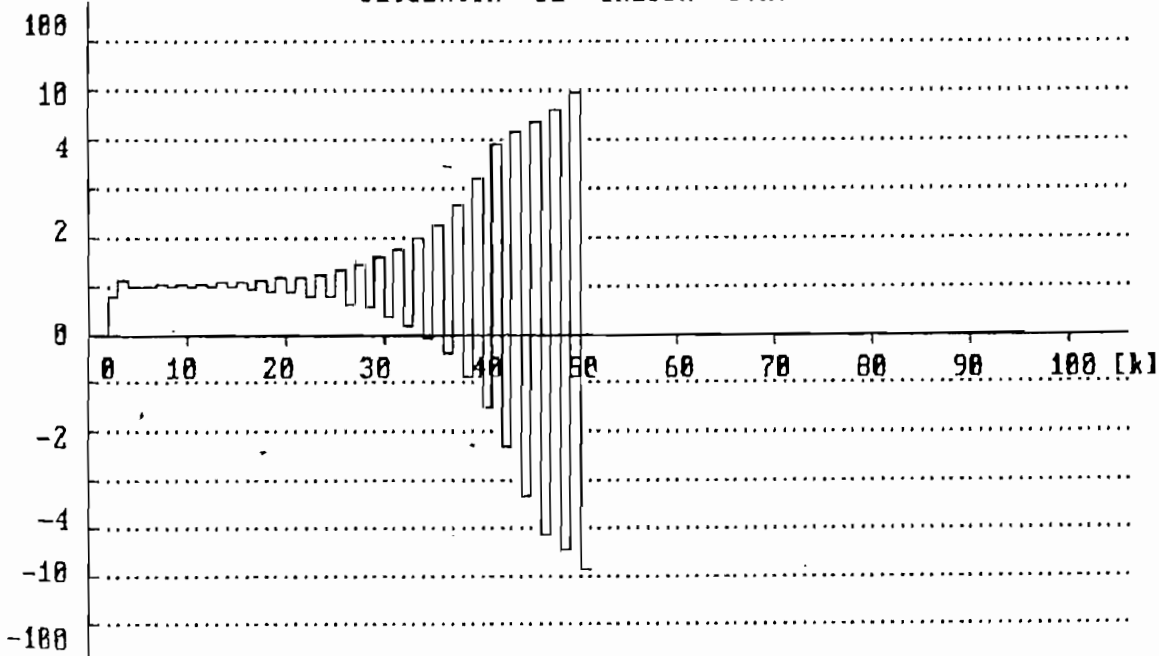
POLINOMIO DE CONTROL Cr

c 1 = -0.200

RETARDO d = 2

SECUENCIA DE CONTROL U(k)



SECUENCIA DE REFERENCIA $Y_M(k)$ SECUENCIA DE SALIDA $Y(k)$ 

• RESULTADOS NUMERICOS •

YM(k)	U(k)	Y(k)
YM(0) = 1.000000	U(0) = 0.800000	Y(0) = 0.000000
YM(1) = 1.000000	U(1) = -0.800000	Y(1) = 0.000000
YM(2) = 1.000000	U(2) = 0.920000	Y(2) = 0.800000
YM(3) = 1.000000	U(3) = -0.912000	Y(3) = 1.120000
YM(4) = 1.000000	U(4) = 1.225600	Y(4) = 1.024000
YM(5) = 1.000000	U(5) = -1.255280	Y(5) = 1.028800
YM(6) = 1.000000	U(6) = 1.588784	Y(6) = 1.007360
YM(7) = 1.000000	U(7) = -1.674275	Y(7) = 1.064192
YM(8) = 1.000000	U(8) = 2.076367	Y(8) = 1.006903
YM(9) = 1.000000	U(9) = -2.233911	Y(9) = 1.068081
YM(10) = 1.000000	U(10) = 2.718244	Y(10) = 0.996518
YM(11) = 1.000000	U(11) = -2.971559	Y(11) = 1.079722
YM(12) = 1.000000	U(12) = 3.566007	Y(12) = 0.984436
YM(13) = 1.000000	U(13) = -3.945593	Y(13) = 1.093754
YM(14) = 1.000000	U(14) = 4.685118	Y(14) = 0.968088
YM(15) = 1.000000	U(15) = -5.231444	Y(15) = 1.112507
YM(16) = 1.000000	U(16) = 6.162550	Y(16) = 0.946584
YM(17) = 1.000000	U(17) = -6.928991	Y(17) = 1.137221
YM(18) = 1.000000	U(18) = 8.113007	Y(18) = 0.918179
YM(19) = 1.000000	U(19) = -9.170038	Y(19) = 1.169857
YM(20) = 1.000000	U(20) = 10.687939	Y(20) = 0.880683
YM(21) = 1.000000	U(21) = -12.128599	Y(21) = 1.212940
YM(22) = 1.000000	U(22) = 14.087284	Y(22) = 0.831181
YM(23) = 1.000000	U(23) = -16.034399	Y(23) = 1.269817
YM(24) = 1.000000	U(24) = 18.574993	Y(24) = 0.765831
YM(25) = 1.000000	U(25) = -21.190714	Y(25) = 1.344904
YM(26) = 1.000000	U(26) = 24.499527	Y(26) = 0.679557
YM(27) = 1.000000	U(27) = -27.997921	Y(27) = 1.444031
YM(28) = 1.000000	U(28) = 32.320915	Y(28) = 0.565662
YM(29) = 1.000000	U(29) = -36.984589	Y(29) = 1.574896
YM(30) = 1.000000	U(30) = 42.646469	Y(30) = 0.415302
YM(31) = 1.000000	U(31) = -48.848507	Y(31) = 1.747660
YM(32) = 1.000000	U(32) = 56.277946	Y(32) = 0.216798
YM(33) = 1.000000	U(33) = -64.510880	Y(33) = 1.975734
YM(34) = 1.000000	U(34) = 74.273788	Y(34) = -0.045262
YM(35) = 1.000000	U(35) = -85.187851	Y(35) = 2.276837
YM(36) = 1.000000	U(36) = 98.031349	Y(36) = -0.391223
YM(37) = 1.000000	U(37) = -112.484962	Y(37) = 2.674337
YM(38) = 1.000000	U(38) = 129.395340	Y(38) = -0.847943
YM(39) = 1.000000	U(39) = -148.521744	Y(39) = 3.199112
YM(40) = 1.000000	U(40) = 170.801102	Y(40) = -1.450906
YM(41) = 1.000000	U(41) = -196.096390	Y(41) = 3.891894
YM(42) = 1.000000	U(42) = 225.463715	Y(42) = -2.246908
YM(43) = 1.000000	U(43) = -258.902985	Y(43) = 4.806488
YM(44) = 1.000000	U(44) = 297.627625	Y(44) = -3.297758
YM(45) = 1.000000	U(45) = -341.818298	Y(45) = 6.013903
YM(46) = 1.000000	U(46) = 392.896149	Y(46) = -4.685071
YM(47) = 1.000000	U(47) = -451.280487	Y(47) = 7.607915
YM(48) = 1.000000	U(48) = 518.666687	Y(48) = -6.516553
YM(49) = 1.000000	U(49) = -595.789063	Y(49) = 9.712276
YM(50) = 1.000000	U(50) = 684.704956	Y(50) = -8.934394

SIMULACION CL-9

PLANTA

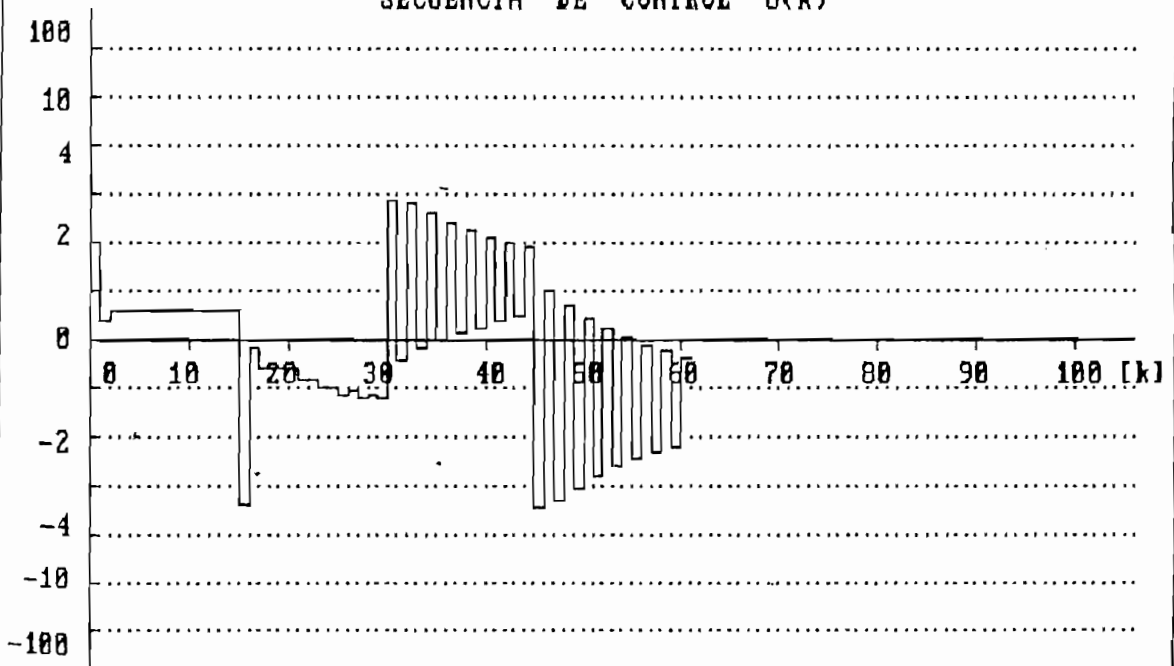
POLINOMIO A : $a_1 = -1.000$
 $a_2 = 0.240$

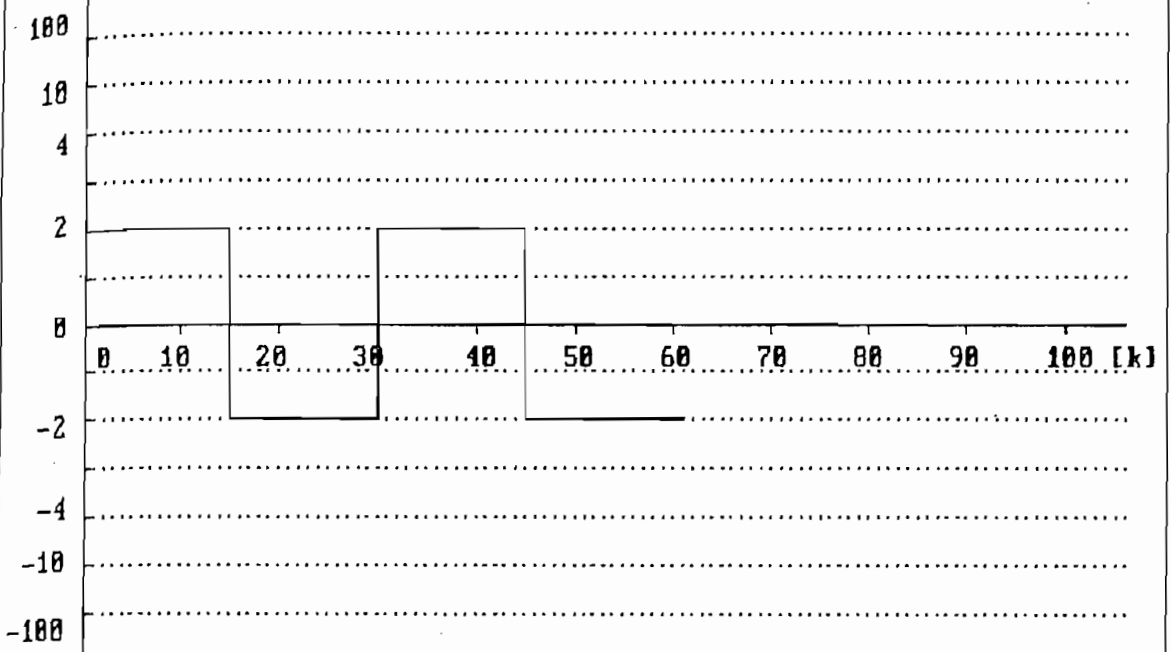
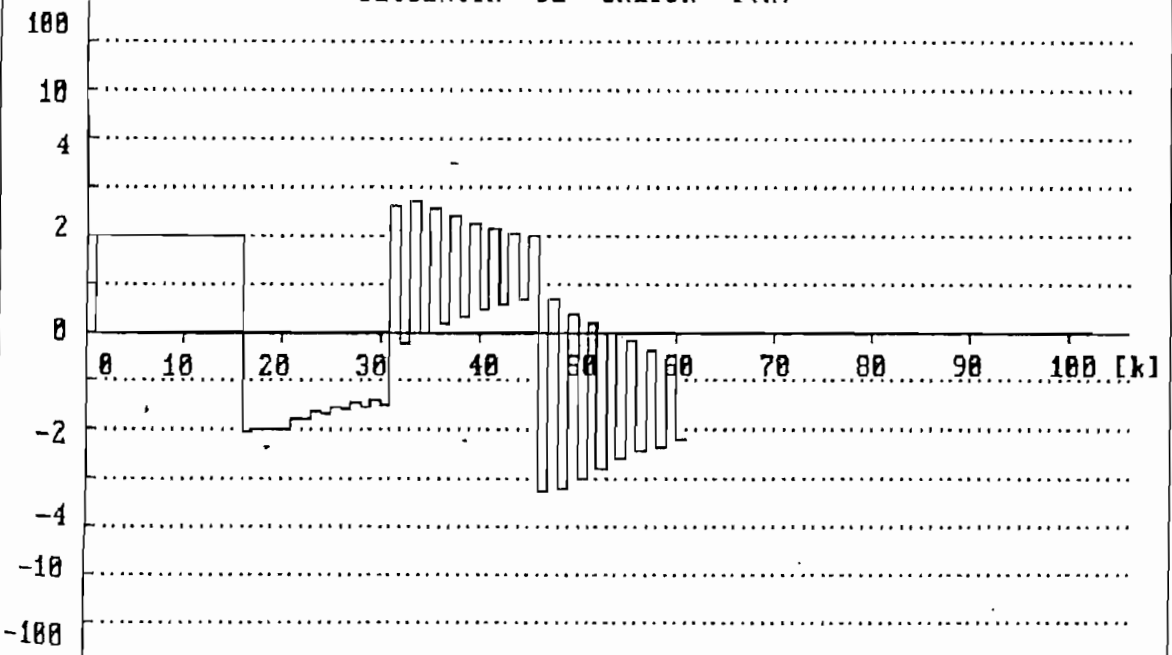
POLINOMIO B : $b_0 = 1.000$
 $b_1 = -0.200$

POLINOMIO DE CONTROL C_r

$C_r = 1$

RETARDO $d = 1$

SECUENCIA DE CONTROL $U(k)$ 

SECUENCIA DE REFERENCIA $Y_M(k)$ SECUENCIA DE SALIDA $Y(k)$ 

• RESULTADOS NUMERICOS •

YM(k)	U(k)	Y(k)
YM(0) = 2.000000	U(0) = 2.000000	Y(0) = 0.000000
YM(1) = 2.000000	U(1) = 0.400000	Y(1) = 2.000000
YM(2) = 2.000000	U(2) = 0.560000	Y(2) = 2.000000
YM(3) = 2.000000	U(3) = 0.592000	Y(3) = 2.000000
YM(4) = 2.000000	U(4) = 0.598400	Y(4) = 2.000000
YM(5) = 2.000000	U(5) = 0.599680	Y(5) = 2.000000
YM(6) = 2.000000	U(6) = 0.599936	Y(6) = 2.000000
YM(7) = 2.000000	U(7) = 0.599987	Y(7) = 2.000000
YM(8) = 2.000000	U(8) = 0.599997	Y(8) = 2.000000
YM(9) = 2.000000	U(9) = 0.599999	Y(9) = 2.000000
YM(10) = 2.000000	U(10) = 0.600000	Y(10) = 2.000000
YM(11) = 2.000000	U(11) = 0.600000	Y(11) = 2.000000
YM(12) = 2.000000	U(12) = 0.600000	Y(12) = 2.000000
YM(13) = 2.000000	U(13) = 0.600000	Y(13) = 2.000000
YM(14) = 2.000000	U(14) = 0.600000	Y(14) = 2.000000
YM(15) = -2.000000	U(15) = -3.400000	Y(15) = 2.000000
YM(16) = -2.000000	U(16) = -0.200000	Y(16) = -2.000000
YM(17) = -2.000000	U(17) = -0.520000	Y(17) = -2.000000
YM(18) = -2.000000	U(18) = -0.584000	Y(18) = -2.000000
YM(19) = -2.000000	U(19) = -0.596800	Y(19) = -2.000000
YM(20) = -2.000000	U(20) = -0.599360	Y(20) = -2.000000
YM(21) = -2.000000	U(21) = -0.837952	Y(21) = -1.761920
YM(22) = -2.000000	U(22) = -0.806259	Y(22) = -1.784192
YM(23) = -2.000000	U(23) = -0.988741	Y(23) = -1.600717
YM(24) = -2.000000	U(24) = -0.940277	Y(24) = -1.641643
YM(25) = -2.000000	U(25) = -1.083344	Y(25) = -1.498705
YM(26) = -2.000000	U(26) = -1.027732	Y(26) = -1.548627
YM(27) = -2.000000	U(27) = -1.142293	Y(27) = -1.434923
YM(28) = -2.000000	U(28) = -1.085191	Y(28) = -1.487649
YM(29) = -2.000000	U(29) = -1.178627	Y(29) = -1.395447
YM(30) = 2.000000	U(30) = 2.876732	Y(30) = -1.447364
YM(31) = 2.000000	U(31) = -0.400663	Y(31) = 2.628642
YM(32) = 2.000000	U(32) = 2.771223	Y(32) = -0.220482
YM(33) = 2.000000	U(33) = -0.181699	Y(33) = 2.683029
YM(34) = 2.000000	U(34) = 2.573901	Y(34) = 0.033686
YM(35) = 2.000000	U(35) = -0.012071	Y(35) = 2.534936
YM(36) = 2.000000	U(36) = 2.398415	Y(36) = 0.207556
YM(37) = 2.000000	U(37) = 0.137420	Y(37) = 2.392076
YM(38) = 2.000000	U(38) = 2.246630	Y(38) = 0.354952
YM(39) = 2.000000	U(39) = 0.269730	Y(39) = 2.264785
YM(40) = 2.000000	U(40) = 2.115159	Y(40) = 0.482336
YM(41) = 2.000000	U(41) = 0.386498	Y(41) = 2.152294
YM(42) = 2.000000	U(42) = 2.001001	Y(42) = 0.592849
YM(43) = 2.000000	U(43) = 0.489301	Y(43) = 2.053183
YM(44) = 2.000000	U(44) = 1.901687	Y(44) = 0.688937
YM(45) = -2.000000	U(45) = -3.420353	Y(45) = 1.966035
YM(46) = -2.000000	U(46) = 1.015164	Y(46) = -3.227386
YM(47) = -2.000000	U(47) = -3.261056	Y(47) = 0.689516
YM(48) = -2.000000	U(48) = 0.707704	Y(48) = -3.194432
YM(49) = -2.000000	U(49) = -3.011522	Y(49) = 0.386400
YM(50) = -2.000000	U(50) = 0.464803	Y(50) = -2.974372

6.1.2 Simulaciones-Control Adaptivo MRAC

A continuación se presentan ejemplos representativos de simulaciones numéricas que permiten establecer el comportamiento del control adaptivo MRAC propuesto en capítulos precedentes e implementado computacionalmente para su verificación. El mecanismo a seguirse en las simulaciones fue ya considerado en el capítulo anterior.

Para las simulaciones CA-1, CA-2 y CA-3 se utiliza una planta inicial de 2do. orden cuyos parámetros, para fines de control, se consideran desconocidos. Esta planta está dada por:

$$A(q^{-1}) = 1 - q^{-1} + 0.24q^{-2}$$

$$B(q^{-1}) = 1 - 0.2q^{-1}$$

Se tiene consecuentemente una planta estable y de fase mínima. El modelo de referencia determina a partir de $k=0$ una onda cuadrada simétrica cuyos niveles son $+2$ y -2 . En el instante $k=20$ la planta es alterada, transformándose a partir de ese instante en una planta dada por:

$$A(q^{-1}) = 1 - 0.9q^{-1} + 0.08q^{-2}$$

$$B(q^{-1}) = 1 - 0.8q^{-1}$$

Esta nueva planta es también estable y de fase mínima. Los ceros de A están dados en la planta original en $q=0.4$ y $q=0.6$ y en la alterada en $q=0.1$ y $q=0.8$. El cero de B pasa de $q=0.2$ a $q=0.8$.

Para la simulación CA-1 se ha escogido un polinomio de control Cr con sus ceros en el origen: $Cr(q^{-1}) = 1$. Se observa en los resultados la presencia de dos etapas de adaptación, una a partir de $k=0$ (ya que se supone no conocida la planta inicial) y otra a partir de $k=20$ (cuando se altera la planta). Los parámetros estimados para el controlador en el instante inicial ($k=0$) y necesarios para la inicialización del proceso de identificación recursiva son considerados nulos, a excepción de b_0 que por participar como denominador en el procesamiento se lo toma igual a la unidad. De los resultados en CA-1 se determina el cumplimiento de los objetivos generales del control: la salida converge a la referencia deseada y la secuencia de control es acotada.

En la simulación CA-2 se utiliza un polinomio Cr dado por: $Cr(q^{-1}) = 1 - 0.3q^{-1}$. Las demás condiciones son similares a las anteriores. Se observa una mejor respuesta con respecto al caso anterior; la señal de control requerida en la etapa inicial de transición es menor y con variaciones menos bruscas.

Para la simulación CA-3 igualmente se mantienen las condiciones anteriores, a excepción del polinomio Cr , al que se le asigna un cero en el semiplano izquierdo y que está dado por $Cr(q^{-1}) = 1 + 0.6q^{-1}$. Si bien se consigue el seguimiento a la referencia, los resultados obtenidos son notoriamente desfavorables con respecto a los casos precedentes. Se observa en la señal de control una etapa transitoria de adaptación más irregular y con valores extremos de mayor magnitud

Iguales resultados se reflejan en la secuencia de salida.

En la simulación CA-4 se contempla una transición entre una planta inicial estable hacia una inestable. La planta original está dada por:

$$A'(q^{-1}) = 1 - 1.2q^{-1} + 0.35q^{-2}$$

$$B'(q^{-1}) = 1 - 0.7q^{-1}$$

y la inestable esta representada por :

$$A(q^{-1}) = 1 + 1.5q^{-1}$$

$$B(q^{-1}) = 1 - 0.4q^{-1}$$

El polinomio de control C_r utilizado es: $C_r(q^{-1}) = 1 - 0.2q^{-1}$. Los resultados obtenidos son satisfactorios, particularmente por la rápida convergencia conseguida. Cabe anotar que para este caso se utilizó $F_{1n1} = \text{diag}(100)$ mientras que para los precedentes $F_{1n1} = \text{diag}(10)$.

En la simulación CA-5 se genera una secuencia de referencia con varios niveles y la planta experimenta a partir del instante k de alteración un aumento en su orden. Así, la planta, inicial está dada por:

$$A'(q^{-1}) = 1 + q^{-1}$$

$$B'(q^{-1}) = 2 + q^{-1}$$

y la planta modificada por:

$$A(q^{-1}) = 1 + q^{-1} + q^{-2} + q^{-3}$$

$$B(q^{-1}) = 1 + 0.2q^{-1} + 0.3q^{-2}$$

El polinomio de control escogido es de 2do. grado y esta descrito por: $Cr(q^{-1}) = 1 - 0.4q^{-1} + 0.1q^{-2}$. Se supone conocida la planta inicial al instante $k=0$, de manera que, los parámetros iniciales del controlador son determinados en base a dichos datos. La etapa transitoria de adaptación comienza a partir del instante de modificación ($k=15$) observándose que para $k=28$ se retoma nuevamente la referencia y a partir de ese instante se presenta un adecuado seguimiento a la referencia.

Las simulaciones CA-6 y CA-7 utilizan secuencias de referencia más complejas, permitiendo observar la respuesta en seguimiento del control. En el ejemplo CA-6 se presenta además una disminución en el orden de la planta, estando la planta inicial dada por :

$$A'(q^{-1}) = 1 + q^{-1} + q^{-2} + q^{-3}$$

$$B'(q^{-1}) = 2 + q^{-1} + q^{-2}$$

y la modificada por :

$$A(q^{-1}) = 1 - 1.2q^{-1} + 0.35q^{-2}$$

$$B(q^{-1}) = 4 + q^{-1} + q^{-2}$$

El polinomio de control es: $Cr(q^{-1}) = 1 - 0.4q^{-1} + 0.1q^{-2}$. Se asume conocida la planta inicial. Se observa que

el período de adaptación va desde $k=30$ hasta $k=46$ presentando valores extremos que se los puede considerar grandes.

En la simulación CA-7 se aprecia un seguimiento satisfactorio a la referencia y una rápida convergencia en el período de adaptación ($k=26$ a $k=38$).

A continuación se presentan los resultados gráficos y numéricos obtenidos en los ejercicios correspondientes de simulación:

SIMULACION CA-1

PLANTA

POLINOMIO A: $a_1 = -1.000$
 $a_2 = 0.240$

POLINOMIO B: $b_0 = 1.000$
 $b_1 = -0.200$

MODIFICACIONES EN LA PLANTA

POLINOMIO A: $a_1 = -0.900$
 $a_2 = 0.000$

POLINOMIO B: $b_0 = 1.000$
 $b_1 = -0.000$

POLINOMIO DE CONTROL C_r

$c_1 = 0.000$

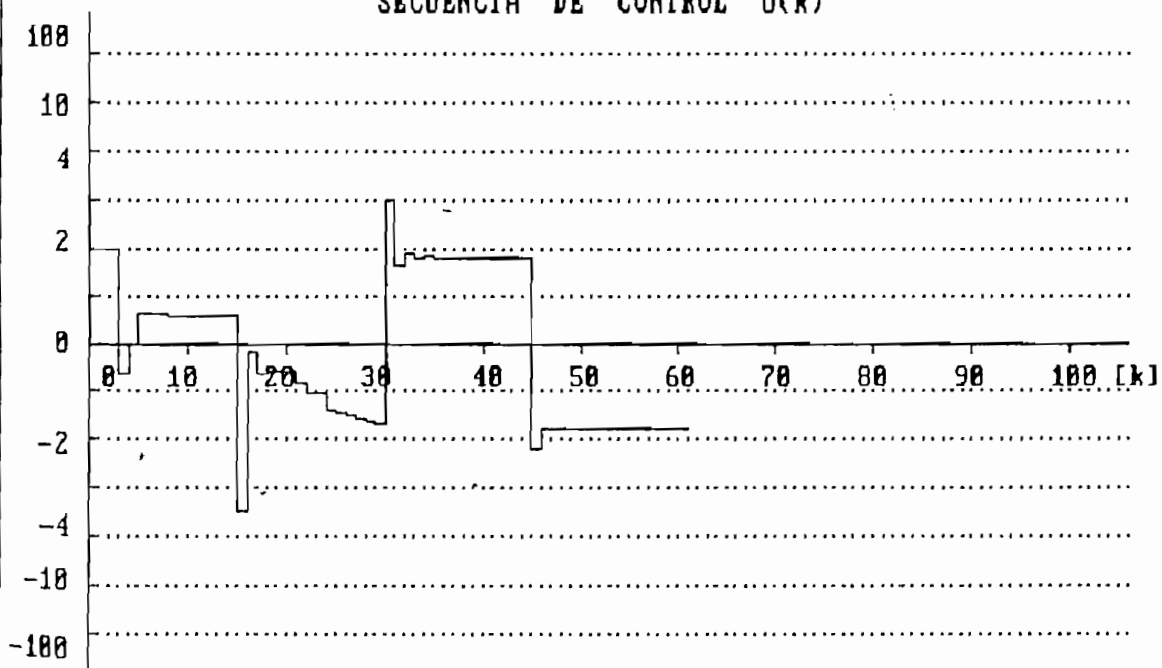
$c_2 = 0.000$

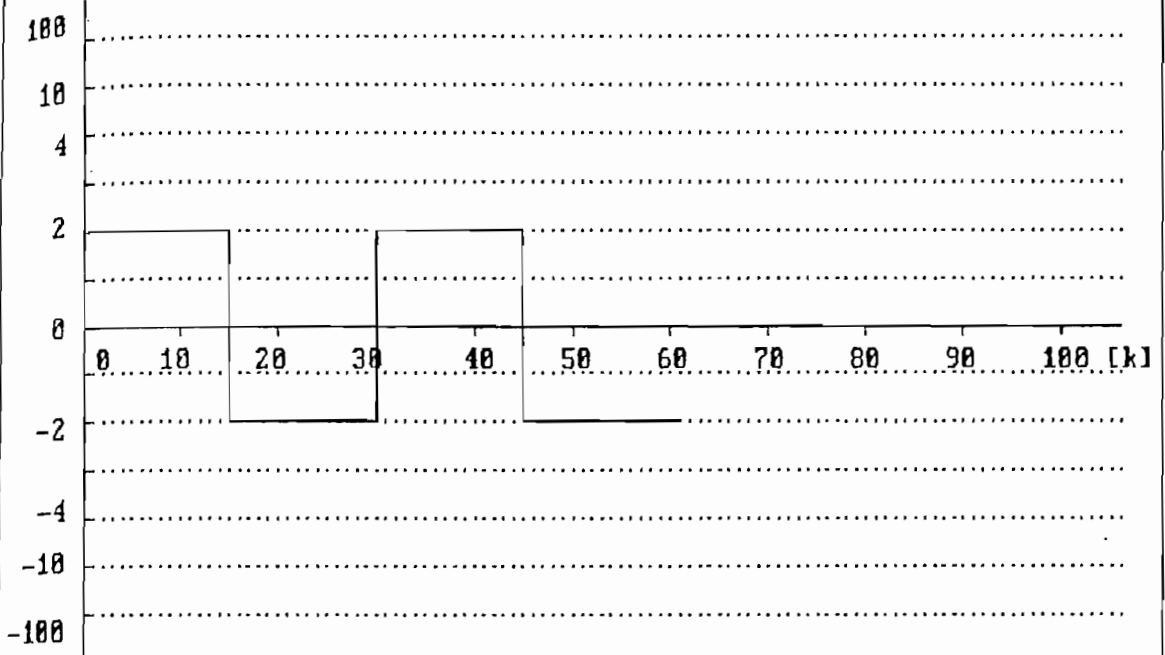
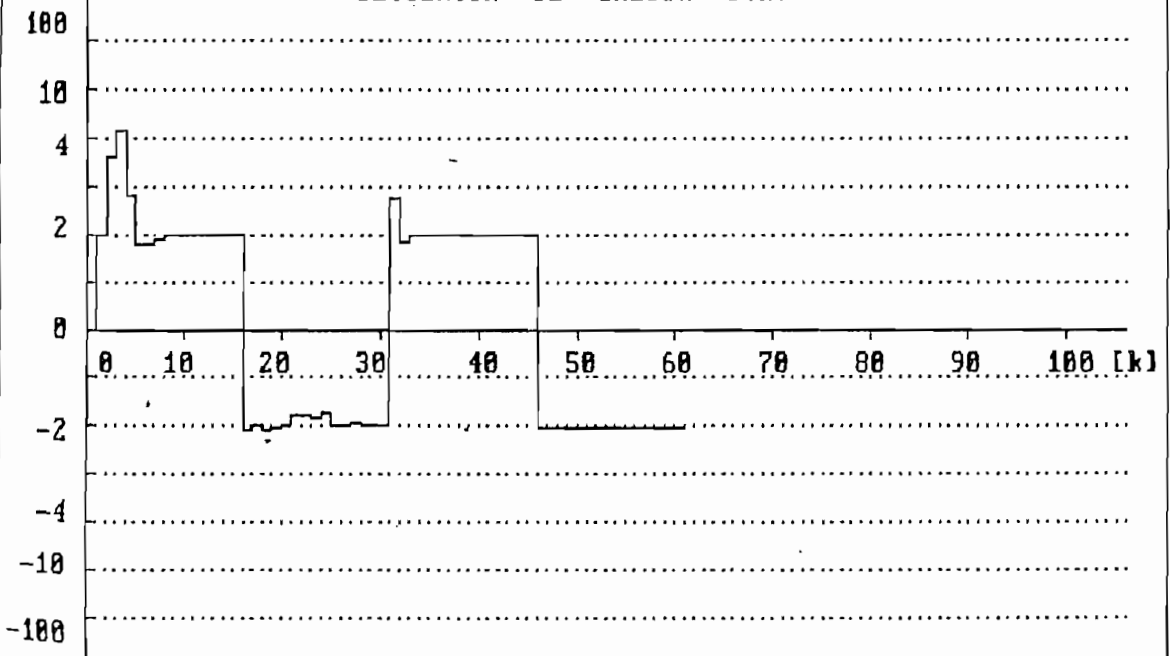
$c_3 = 0.000$

T.MODIFICACION = 20

$F_1 = \text{diag}(10)$

RETARDO $d = 1$

SECUENCIA DE CONTROL $U(k)$ 

SECUENCIA DE REFERENCIA $Y_M(k)$ SECUENCIA DE SALIDA $Y(k)$ 

• RESULTADOS NUMERICOS •

YM(k)	U(k)	Y(k)
YM(0) = 2.000000	U(0) = 2.000000	Y(0) = 0.000000
YM(1) = 2.000000	U(1) = 2.000000	Y(1) = 2.000000
YM(2) = 2.000000	U(2) = 2.000000	Y(2) = 3.600000
YM(3) = 2.000000	U(3) = -0.625079	Y(3) = 4.720000
YM(4) = 2.000000	U(4) = -0.000529	Y(4) = 2.830921
YM(5) = 2.000000	U(5) = 0.643516	Y(5) = 1.822608
YM(6) = 2.000000	U(6) = 0.665121	Y(6) = 1.786809
YM(7) = 2.000000	U(7) = 0.653487	Y(7) = 1.885801
YM(8) = 2.000000	U(8) = 0.603631	Y(8) = 1.977429
YM(9) = 2.000000	U(9) = 0.598713	Y(9) = 1.997771
YM(10) = 2.000000	U(10) = 0.598075	Y(10) = 2.001175
YM(11) = 2.000000	U(11) = 0.599131	Y(11) = 2.000042
YM(12) = 2.000000	U(12) = 0.599647	Y(12) = 1.999277
YM(13) = 2.000000	U(13) = 0.599805	Y(13) = 1.999088
YM(14) = 2.000000	U(14) = 0.599836	Y(14) = 1.999137
YM(15) = -2.000000	U(15) = -3.458324	Y(15) = 1.999231
YM(16) = -2.000000	U(16) = -0.122884	Y(16) = -2.058853
YM(17) = -2.000000	U(17) = -0.620699	Y(17) = -1.969888
YM(18) = -2.000000	U(18) = -0.536245	Y(18) = -2.071885
YM(19) = -2.000000	U(19) = -0.592155	Y(19) = -2.011218
YM(20) = -2.000000	U(20) = -0.599482	Y(20) = -1.998872
YM(21) = -2.000000	U(21) = -0.821165	Y(21) = -1.763845
YM(22) = -2.000000	U(22) = -1.019248	Y(22) = -1.769130
YM(23) = -2.000000	U(23) = -1.019591	Y(23) = -1.813425
YM(24) = -2.000000	U(24) = -1.363490	Y(24) = -1.694744
YM(25) = -2.000000	U(25) = -1.417520	Y(25) = -1.928013
YM(26) = -2.000000	U(26) = -1.467320	Y(26) = -1.926361
YM(27) = -2.000000	U(27) = -1.568598	Y(27) = -1.912788
YM(28) = -2.000000	U(28) = -1.617433	Y(28) = -1.962142
YM(29) = -2.000000	U(29) = -1.650957	Y(29) = -1.975459
YM(30) = 2.000000	U(30) = 3.026377	Y(30) = -1.977952
YM(31) = 2.000000	U(31) = 1.655891	Y(31) = 2.725023
YM(32) = 2.000000	U(32) = 1.896048	Y(32) = 1.845546
YM(33) = 2.000000	U(33) = 1.839506	Y(33) = 2.014325
YM(34) = 2.000000	U(34) = 1.842570	Y(34) = 1.987916
YM(35) = 2.000000	U(35) = 1.833840	Y(35) = 1.998943
YM(36) = 2.000000	U(36) = 1.827154	Y(36) = 1.999800
YM(37) = 2.000000	U(37) = 1.821788	Y(37) = 1.999987
YM(38) = 2.000000	U(38) = 1.817480	Y(38) = 2.000068
YM(39) = 2.000000	U(39) = 1.814025	Y(39) = 2.000112
YM(40) = 2.000000	U(40) = 1.811257	Y(40) = 2.000137
YM(41) = 2.000000	U(41) = 1.809038	Y(41) = 2.000151
YM(42) = 2.000000	U(42) = 1.807261	Y(42) = 2.000158
YM(43) = 2.000000	U(43) = 1.805837	Y(43) = 2.000160
YM(44) = 2.000000	U(44) = 1.804696	Y(44) = 2.000159
YM(45) = -2.000000	U(45) = -2.199681	Y(45) = 2.000157
YM(46) = -2.000000	U(46) = -1.797396	Y(46) = -2.003309
YM(47) = -2.000000	U(47) = -1.797803	Y(47) = -2.000642
YM(48) = -2.000000	U(48) = -1.798309	Y(48) = -2.000199
YM(49) = -2.000000	U(49) = -1.798660	Y(49) = -2.000194
YM(50) = -2.000000	U(50) = -1.798942	Y(50) = -2.000172

SIMULACION CA-2

PLANTA

POLINOMIO A: $a_1 = -1.000$
 $a_2 = 0.240$

POLINOMIO B: $b_0 = 1.000$
 $b_1 = -0.200$

MODIFICACIONES EN LA PLANTA

POLINOMIO A: $a_1 = -0.900$
 $a_2 = 0.000$

POLINOMIO B: $b_0 = 1.000$
 $b_1 = -0.000$

POLINOMIO DE CONTROL C_r

$c_1 = -0.300$

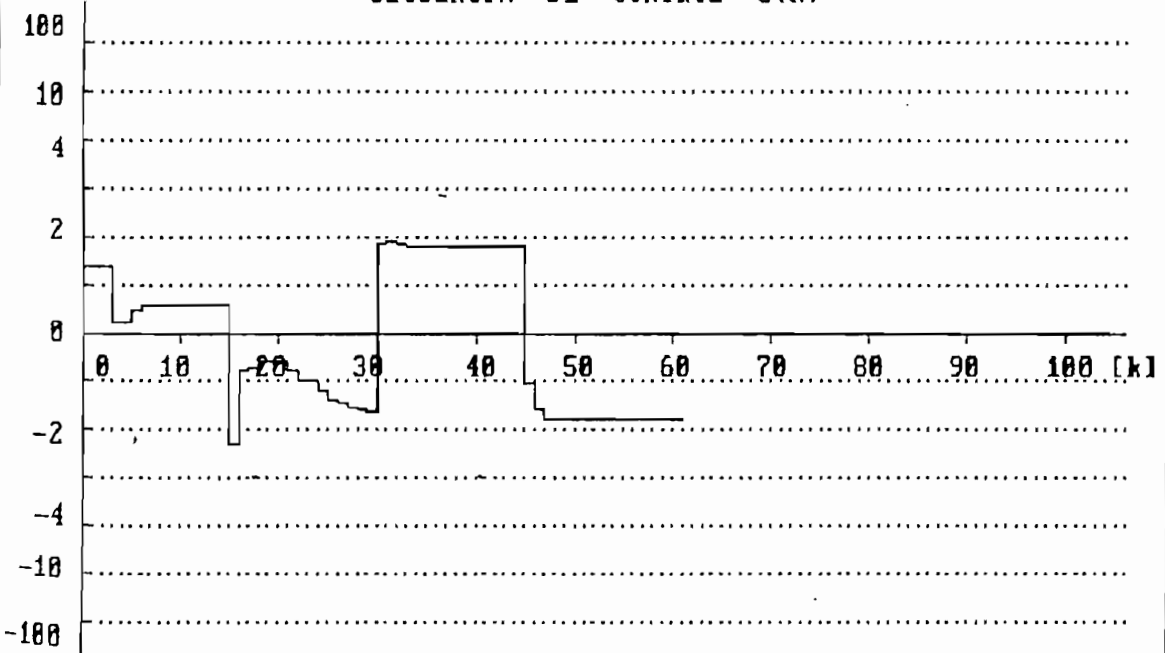
$c_2 = 0.000$

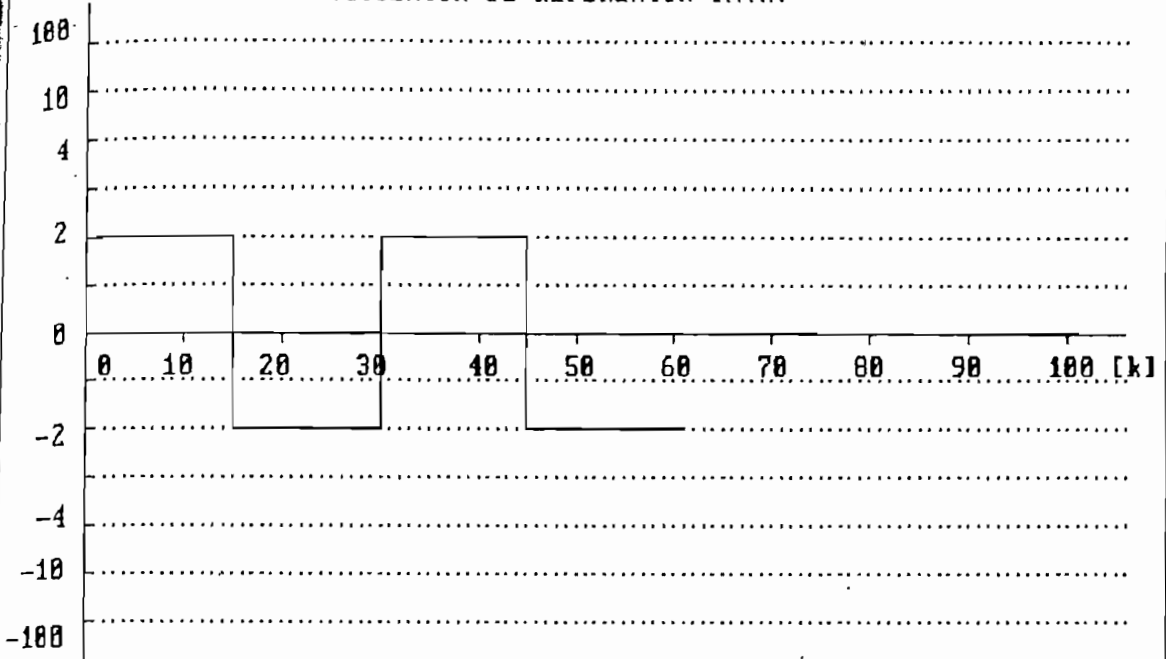
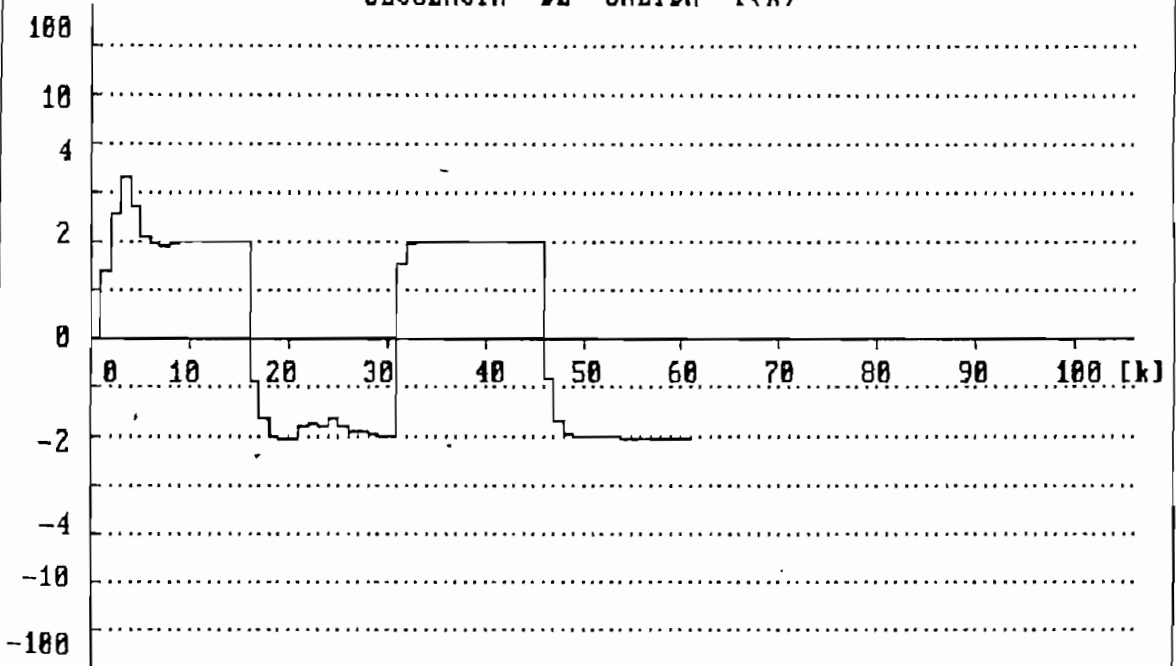
$c_3 = 0.000$

T.MODIFICACION = 20

$F_1 = \text{diag}(10)$

RETARDO $d = 1$

SECUENCIA DE CONTROL $U(k)$ 

SECUENCIA DE REFERENCIA $Y_M(k)$ SECUENCIA DE SALIDA $Y(k)$ 

▪ RESULTADOS NUMERICOS ▪

YM(k)	U(k)	Y(k)
YM(0) = 2.000000	U(0) = 1.400000	Y(0) = 0.000000
YM(1) = 2.000000	U(1) = 1.400000	Y(1) = 1.400000
YM(2) = 2.000000	U(2) = 1.400000	Y(2) = 2.520000
YM(3) = 2.000000	U(3) = 0.270113	Y(3) = 3.304000
YM(4) = 2.000000	U(4) = 0.266515	Y(4) = 2.689313
YM(5) = 2.000000	U(5) = 0.516851	Y(5) = 2.108845
YM(6) = 2.000000	U(6) = 0.597184	Y(6) = 1.926958
YM(7) = 2.000000	U(7) = 0.613051	Y(7) = 1.914649
YM(8) = 2.000000	U(8) = 0.608629	Y(8) = 1.945794
YM(9) = 2.000000	U(9) = 0.602803	Y(9) = 1.972297
YM(10) = 2.000000	U(10) = 0.599875	Y(10) = 1.986383
YM(11) = 2.000000	U(11) = 0.598832	Y(11) = 1.992346
YM(12) = 2.000000	U(12) = 0.598673	Y(12) = 1.994471
YM(13) = 2.000000	U(13) = 0.598786	Y(13) = 1.995215
YM(14) = 2.000000	U(14) = 0.598930	Y(14) = 1.995593
YM(15) = -2.000000	U(15) = -2.259156	Y(15) = 1.995914
YM(16) = -2.000000	U(16) = -0.724587	Y(16) = -0.861970
YM(17) = -2.000000	U(17) = -0.691455	Y(17) = -1.613745
YM(18) = -2.000000	U(18) = -0.589507	Y(18) = -1.953410
YM(19) = -2.000000	U(19) = -0.578507	Y(19) = -2.017327
YM(20) = -2.000000	U(20) = -0.590820	Y(20) = -2.009115
YM(21) = -2.000000	U(21) = -0.729276	Y(21) = -1.774832
YM(22) = -2.000000	U(22) = -0.955832	Y(22) = -1.693239
YM(23) = -2.000000	U(23) = -0.946985	Y(23) = -1.754340
YM(24) = -2.000000	U(24) = -1.197012	Y(24) = -1.625766
YM(25) = -2.000000	U(25) = -1.380521	Y(25) = -1.762267
YM(26) = -2.000000	U(26) = -1.424618	Y(26) = -1.878890
YM(27) = -2.000000	U(27) = -1.499092	Y(27) = -1.870221
YM(28) = -2.000000	U(28) = -1.576731	Y(28) = -1.892285
YM(29) = -2.000000	U(29) = -1.625593	Y(29) = -1.930896
YM(30) = 2.000000	U(30) = 1.844964	Y(30) = -1.950632
YM(31) = 2.000000	U(31) = 1.887416	Y(31) = 1.544342
YM(32) = 2.000000	U(32) = 1.842532	Y(32) = 1.957403
YM(33) = 2.000000	U(33) = 1.832232	Y(33) = 1.970714
YM(34) = 2.000000	U(34) = 1.836443	Y(34) = 1.975257
YM(35) = 2.000000	U(35) = 1.832404	Y(35) = 1.990732
YM(36) = 2.000000	U(36) = 1.826991	Y(36) = 1.996887
YM(37) = 2.000000	U(37) = 1.821994	Y(37) = 1.999008
YM(38) = 2.000000	U(38) = 1.817777	Y(38) = 1.999757
YM(39) = 2.000000	U(39) = 1.814326	Y(39) = 2.000042
YM(40) = 2.000000	U(40) = 1.811536	Y(40) = 2.000162
YM(41) = 2.000000	U(41) = 1.809291	Y(41) = 2.000218
YM(42) = 2.000000	U(42) = 1.807488	Y(42) = 2.000245
YM(43) = 2.000000	U(43) = 1.806040	Y(43) = 2.000258
YM(44) = 2.000000	U(44) = 1.804879	Y(44) = 2.000263
YM(45) = -2.000000	U(45) = -1.001332	Y(45) = 2.000262
YM(46) = -2.000000	U(46) = -1.560022	Y(46) = -0.805020
YM(47) = -2.000000	U(47) = -1.726789	Y(47) = -1.643496
YM(48) = -2.000000	U(48) = -1.777022	Y(48) = -1.893515
YM(49) = -2.000000	U(49) = -1.792346	Y(49) = -1.968275
YM(50) = -2.000000	U(50) = -1.797116	Y(50) = -1.990695

SIMULACION CA-3

PLANTA

POLINOMIO A: $a_1 = -1.000$
 $a_2 = 0.240$

POLINOMIO B: $b_0 = 1.000$
 $b_1 = -0.200$

MODIFICACIONES EN LA PLANTA

POLINOMIO A: $a_1 = -0.900$
 $a_2 = 0.000$

POLINOMIO B: $b_0 = 1.000$
 $b_1 = -0.000$

POLINOMIO DE CONTROL C_r

$c_1 = 0.600$

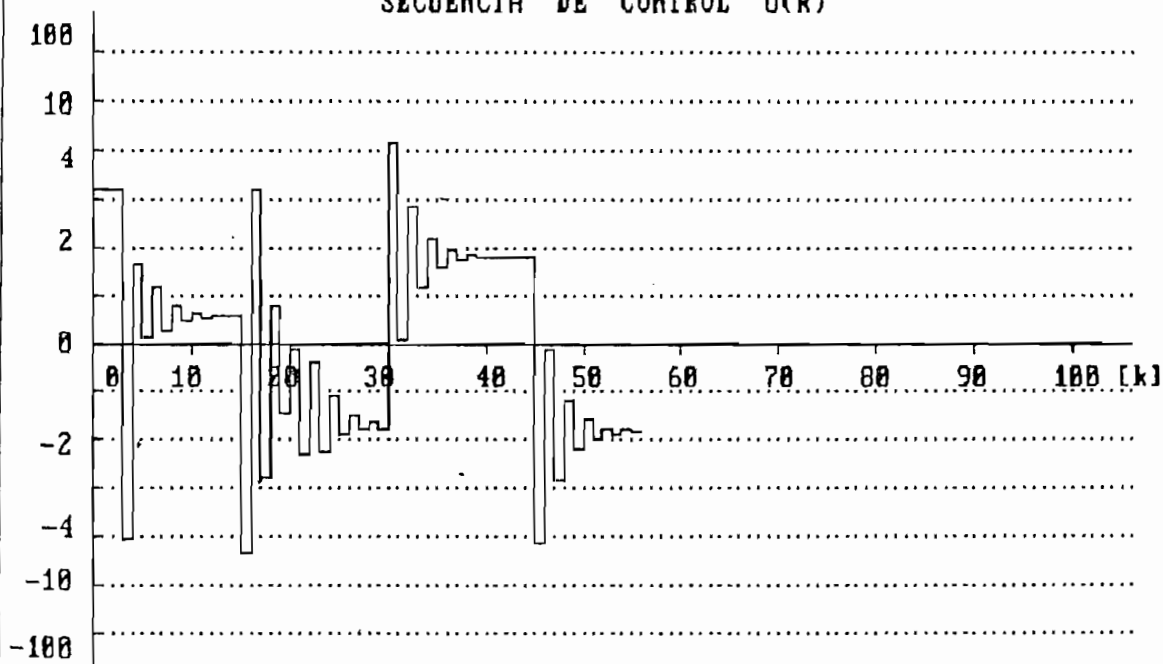
$c_2 = 0.000$

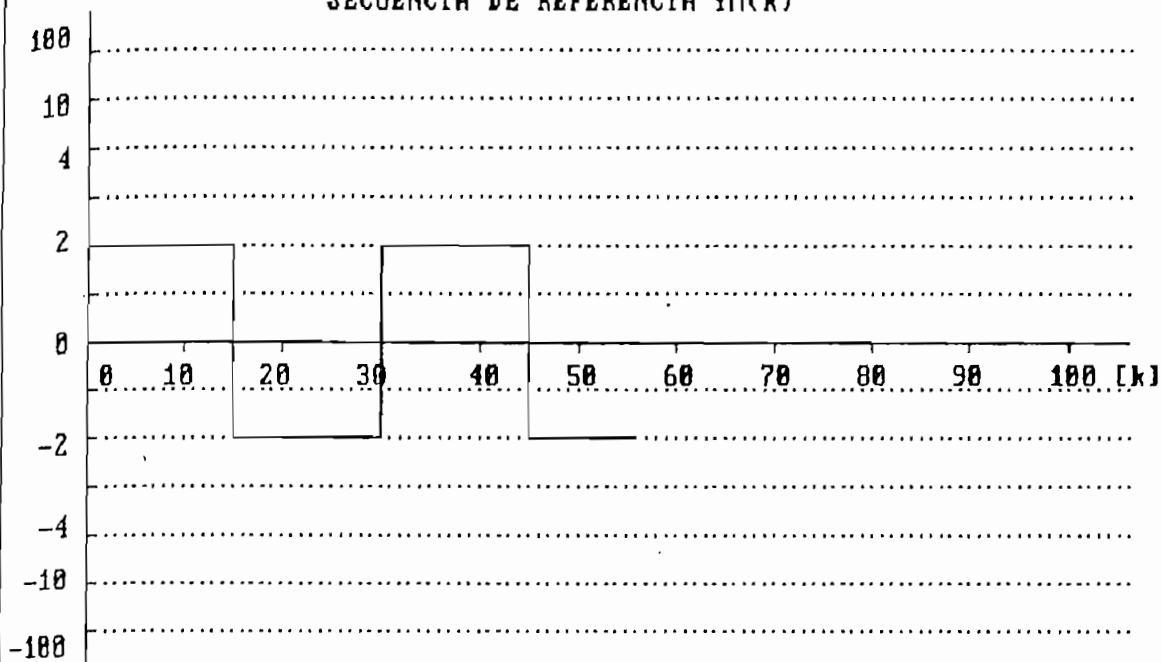
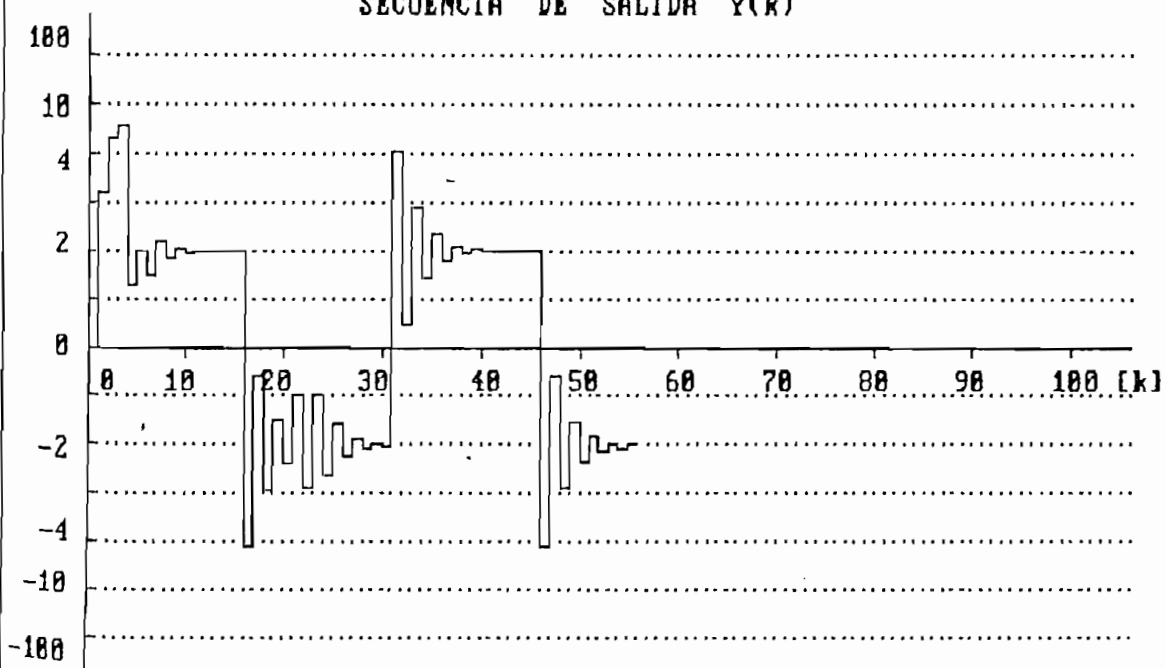
$c_3 = 0.000$

T.MODIFICACION = 20

$F_1 = \text{diag}(10)$

RETARDO $d = 1$

SECUENCIA DE CONTROL $U(k)$ 

SECUENCIA DE REFERENCIA $Y_M(k)$ SECUENCIA DE SALIDA $Y(k)$ 

• RESULTADOS NUMERICOS •

YM(k)	U(k)	Y(k)
YM(0) = 2.000000	U(0) = 3.200000	Y(0) = 0.000000
YM(1) = 2.000000	U(1) = 3.200000	Y(1) = 3.200000
YM(2) = 2.000000	U(2) = 3.200000	Y(2) = 5.760000
YM(3) = 2.000000	U(3) = -4.234475	Y(3) = 7.552000
YM(4) = 2.000000	U(4) = 1.671765	Y(4) = 1.295125
YM(5) = 2.000000	U(5) = 0.153335	Y(5) = 2.001305
YM(6) = 2.000000	U(6) = 1.192221	Y(6) = 1.509458
YM(7) = 2.000000	U(7) = 0.286993	Y(7) = 2.190698
YM(8) = 2.000000	U(8) = 0.783418	Y(8) = 1.876977
YM(9) = 2.000000	U(9) = 0.482436	Y(9) = 2.077229
YM(10) = 2.000000	U(10) = 0.671262	Y(10) = 1.952507
YM(11) = 2.000000	U(11) = 0.555833	Y(11) = 2.028747
YM(12) = 2.000000	U(12) = 0.627164	Y(12) = 1.981726
YM(13) = 2.000000	U(13) = 0.583136	Y(13) = 2.010823
YM(14) = 2.000000	U(14) = 0.610347	Y(14) = 1.992713
YM(15) = -2.000000	U(15) = -5.824898	Y(15) = 2.004035
YM(16) = -2.000000	U(16) = 3.178728	Y(16) = -4.421231
YM(17) = -2.000000	U(17) = -2.769504	Y(17) = -0.558492
YM(18) = -2.000000	U(18) = 0.774222	Y(18) = -2.902646
YM(19) = -2.000000	U(19) = -1.437586	Y(19) = -1.440485
YM(20) = -2.000000	U(20) = -0.094441	Y(20) = -2.336280
YM(21) = -2.000000	U(21) = -2.278680	Y(21) = -0.931786
YM(22) = -2.000000	U(22) = -0.322165	Y(22) = -2.854832
YM(23) = -2.000000	U(23) = -2.208006	Y(23) = -0.994027
YM(24) = -2.000000	U(24) = -1.068010	Y(24) = -2.616512
YM(25) = -2.000000	U(25) = -1.852962	Y(25) = -1.576943
YM(26) = -2.000000	U(26) = -1.468880	Y(26) = -2.208482
YM(27) = -2.000000	U(27) = -1.762653	Y(27) = -1.647989
YM(28) = -2.000000	U(28) = -1.635664	Y(28) = -2.074060
YM(29) = -2.000000	U(29) = -1.749778	Y(29) = -1.944357
YM(30) = 2.000000	U(30) = 4.738646	Y(30) = -2.025243
YM(31) = 2.000000	U(31) = 0.116729	Y(31) = 4.471299
YM(32) = 2.000000	U(32) = 2.876987	Y(32) = 0.512001
YM(33) = 2.000000	U(33) = 1.207764	Y(33) = 2.886700
YM(34) = 2.000000	U(34) = 2.200524	Y(34) = 1.463244
YM(35) = 2.000000	U(35) = 1.596203	Y(35) = 2.320296
YM(36) = 2.000000	U(36) = 1.951631	Y(36) = 1.806991
YM(37) = 2.000000	U(37) = 1.732592	Y(37) = 2.115337
YM(38) = 2.000000	U(38) = 1.859330	Y(38) = 1.930531
YM(39) = 2.000000	U(39) = 1.779544	Y(39) = 2.041507
YM(40) = 2.000000	U(40) = 1.824402	Y(40) = 1.974994
YM(41) = 2.000000	U(41) = 1.795080	Y(41) = 2.014941
YM(42) = 2.000000	U(42) = 1.810740	Y(42) = 1.991005
YM(43) = 2.000000	U(43) = 1.799798	Y(43) = 2.005385
YM(44) = 2.000000	U(44) = 1.805124	Y(44) = 1.996773
YM(45) = -2.000000	U(45) = -4.600240	Y(45) = 2.001950
YM(46) = -2.000000	U(46) = -0.116919	Y(46) = -4.402326
YM(47) = -2.000000	U(47) = -2.807290	Y(47) = -0.558876
YM(48) = -2.000000	U(48) = -1.193657	Y(48) = -2.864637
YM(49) = -2.000000	U(49) = -2.162209	Y(49) = -1.481288
YM(50) = -2.000000	U(50) = -1.581414	Y(50) = -2.311272

SIMULACION CA-4

PLANTA

POLINOMIO A: a 1 = -1.200
a 2 = 0.350

POLINOMIO B: b 0 = 1.000
b 1 = -0.700

MODIFICACIONES EN LA PLANTA

POLINOMIO A: a 1 = 1.500
a 2 = -1.000

POLINOMIO B: b 0 = 1.000
b 1 = -0.400

POLINOMIO DE CONTROL Cr

c1 = -0.200

c2 = 0.000

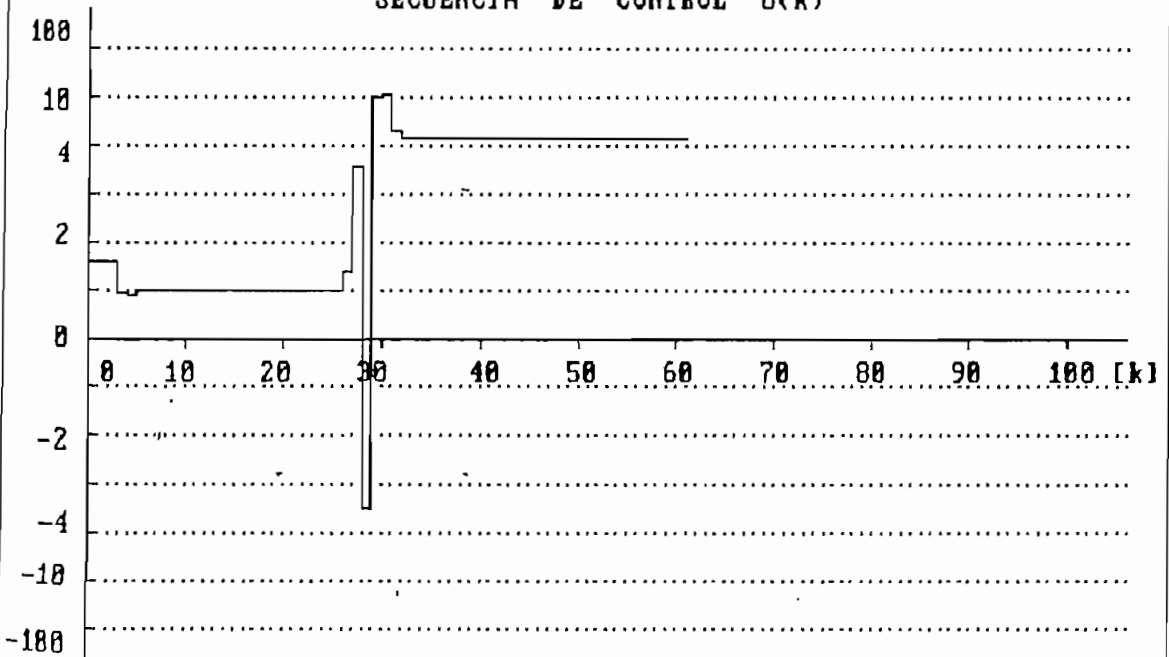
c3 = 0.000

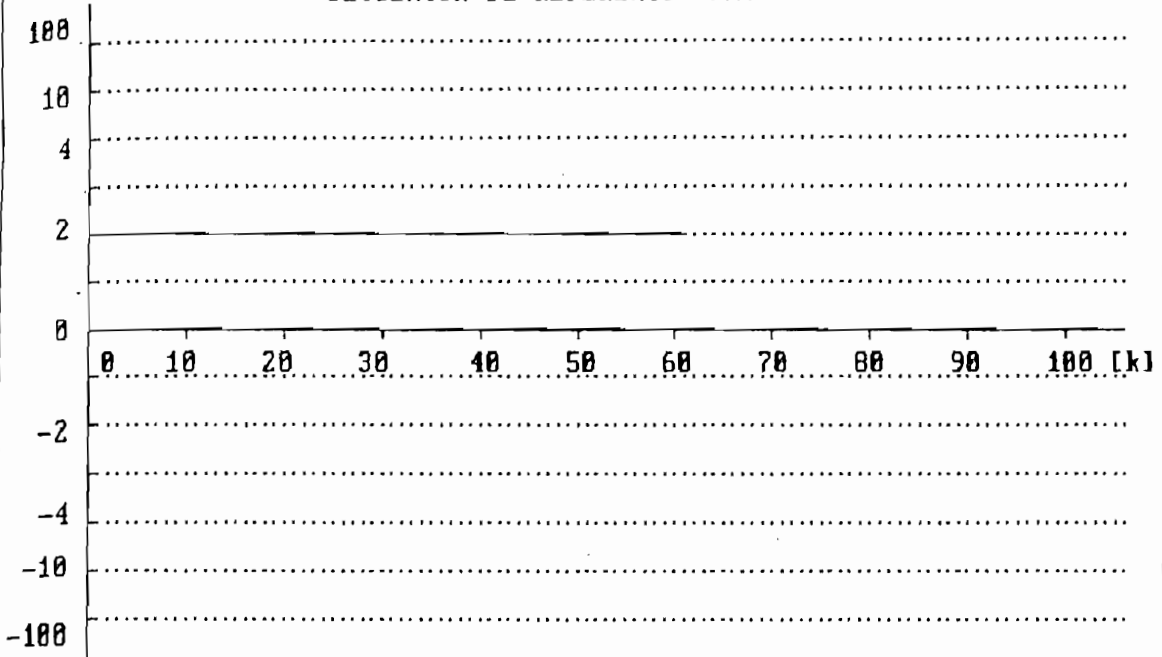
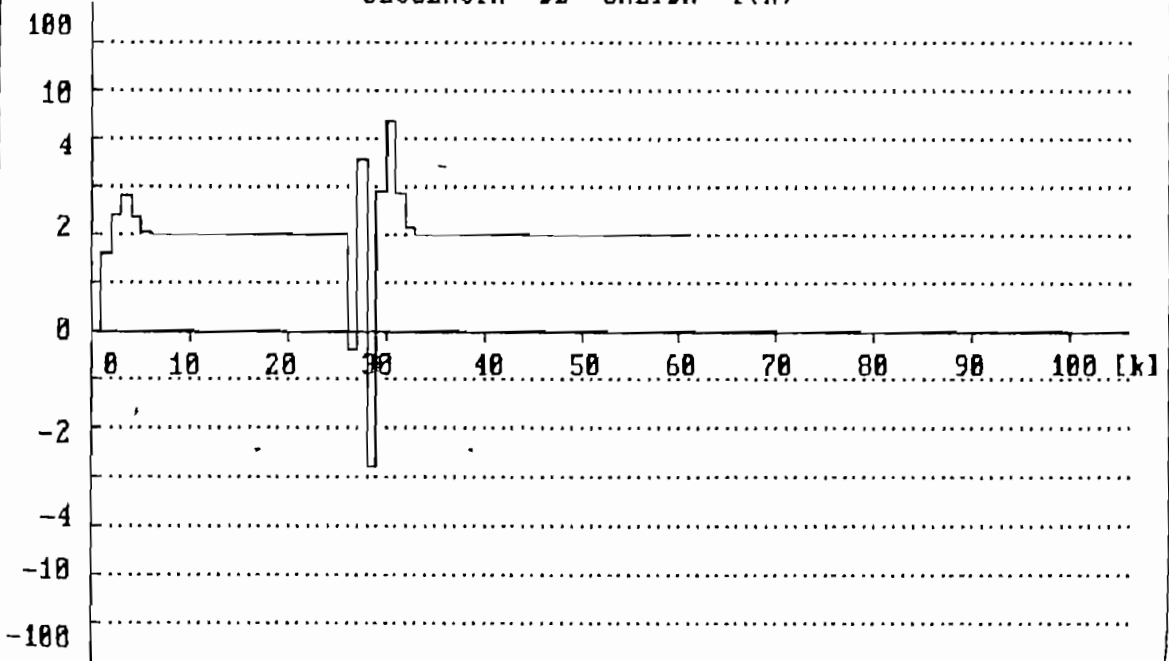
T.MODIFICACION = 25

F1 = diag(100)

RETARDO d = 1

SECUENCIA DE CONTROL U(k)



SECUENCIA DE REFERENCIA $Y_M(k)$ SECUENCIA DE SALIDA $Y(k)$ 

• RESULTADOS NUMERICOS •

YM(k)	U(k)	Y(k)
YM(0) = 2.000000	U(0) = 1.600000	Y(0) = 0.000000
YM(1) = 2.000000	U(1) = 1.600000	Y(1) = 1.600000
YM(2) = 2.000000	U(2) = 1.600000	Y(2) = 2.400000
YM(3) = 2.000000	U(3) = 0.941706	Y(3) = 2.800000
YM(4) = 2.000000	U(4) = 0.897509	Y(4) = 2.341706
YM(5) = 2.000000	U(5) = 0.979064	Y(5) = 2.068362
YM(6) = 2.000000	U(6) = 0.995954	Y(6) = 2.013246
YM(7) = 2.000000	U(7) = 0.999200	Y(7) = 2.002578
YM(8) = 2.000000	U(8) = 0.999835	Y(8) = 2.000489
YM(9) = 2.000000	U(9) = 0.999961	Y(9) = 2.000080
YM(10) = 2.000000	U(10) = 0.999987	Y(10) = 2.000001
YM(11) = 2.000000	U(11) = 0.999992	Y(11) = 1.999988
YM(12) = 2.000000	U(12) = 0.999994	Y(12) = 1.999987
YM(13) = 2.000000	U(13) = 0.999994	Y(13) = 1.999987
YM(14) = 2.000000	U(14) = 0.999995	Y(14) = 1.999989
YM(15) = 2.000000	U(15) = 0.999995	Y(15) = 1.999990
YM(16) = 2.000000	U(16) = 0.999996	Y(16) = 1.999990
YM(17) = 2.000000	U(17) = 0.999996	Y(17) = 1.999991
YM(18) = 2.000000	U(18) = 0.999996	Y(18) = 1.999992
YM(19) = 2.000000	U(19) = 0.999996	Y(19) = 1.999992
YM(20) = 2.000000	U(20) = 0.999996	Y(20) = 1.999993
YM(21) = 2.000000	U(21) = 0.999997	Y(21) = 1.999993
YM(22) = 2.000000	U(22) = 0.999997	Y(22) = 1.999994
YM(23) = 2.000000	U(23) = 0.999997	Y(23) = 1.999994
YM(24) = 2.000000	U(24) = 0.999997	Y(24) = 1.999994
YM(25) = 2.000000	U(25) = 0.999997	Y(25) = 1.999995
YM(26) = 2.000000	U(26) = 1.399495	Y(26) = -0.399999
YM(27) = 2.000000	U(27) = 3.564534	Y(27) = 3.599489
YM(28) = 2.000000	U(28) = -3.451432	Y(28) = -2.794498
YM(29) = 2.000000	U(29) = 11.975686	Y(29) = 2.913991
YM(30) = 2.000000	U(30) = 14.012986	Y(30) = 6.190775
YM(31) = 2.000000	U(31) = 5.848305	Y(31) = 2.850540
YM(32) = 2.000000	U(32) = 4.757842	Y(32) = 2.158076
YM(33) = 2.000000	U(33) = 4.799019	Y(33) = 2.031945
YM(34) = 2.000000	U(34) = 4.897928	Y(34) = 2.006041
YM(35) = 2.000000	U(35) = 4.955178	Y(35) = 2.001205
YM(36) = 2.000000	U(36) = 4.981276	Y(36) = 2.000241
YM(37) = 2.000000	U(37) = 4.992352	Y(37) = 2.000048
YM(38) = 2.000000	U(38) = 4.996909	Y(38) = 2.000010
YM(39) = 2.000000	U(39) = 4.998758	Y(39) = 2.000002
YM(40) = 2.000000	U(40) = 4.999502	Y(40) = 2.000000
YM(41) = 2.000000	U(41) = 4.999801	Y(41) = 2.000000
YM(42) = 2.000000	U(42) = 4.999920	Y(42) = 2.000000
YM(43) = 2.000000	U(43) = 4.999968	Y(43) = 2.000000
YM(44) = 2.000000	U(44) = 4.999987	Y(44) = 2.000000
YM(45) = 2.000000	U(45) = 4.999995	Y(45) = 2.000000
YM(46) = 2.000000	U(46) = 4.999998	Y(46) = 2.000000
YM(47) = 2.000000	U(47) = 4.999999	Y(47) = 2.000000
YM(48) = 2.000000	U(48) = 4.999999	Y(48) = 2.000000
YM(49) = 2.000000	U(49) = 5.000000	Y(49) = 2.000000
YM(50) = 2.000000	U(50) = 5.000000	Y(50) = 2.000000

SIMULACION CA-5

PLANTA

POLINOMIO A: $a_1 = 1.000$
 $a_2 = 1.000$

POLINOMIO B: $b_0 = 2.000$
 $b_1 = 1.000$

MODIFICACIONES EN LA PLANTA

POLINOMIO A: $a_1 = 1.000$
 $a_2 = 1.000$
 $a_3 = 1.000$

POLINOMIO B: $b_0 = 1.000$
 $b_1 = 0.200$
 $b_2 = 0.300$

POLINOMIO DE CONTROL Cr

$c_1 = -0.400$

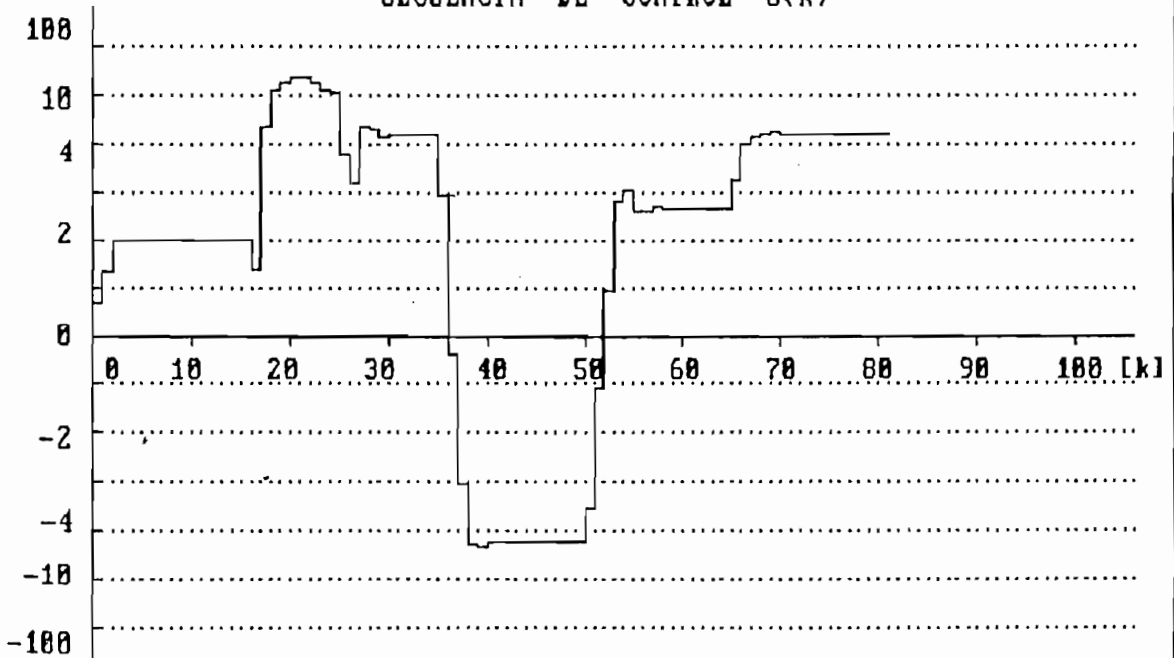
$c_2 = 0.100$

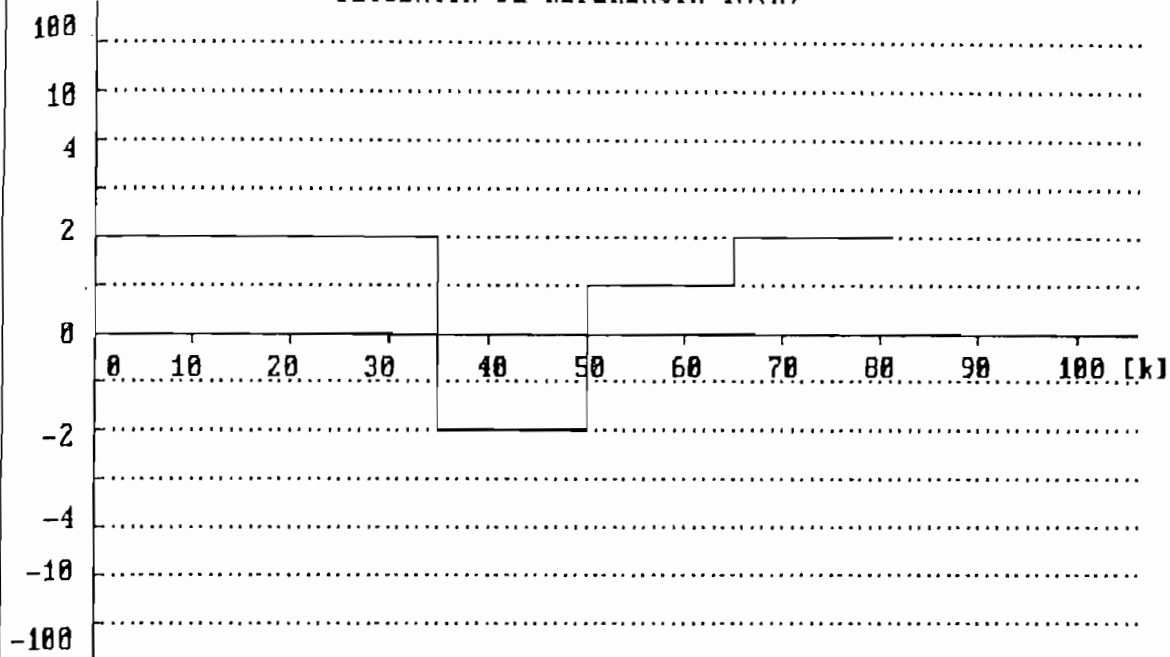
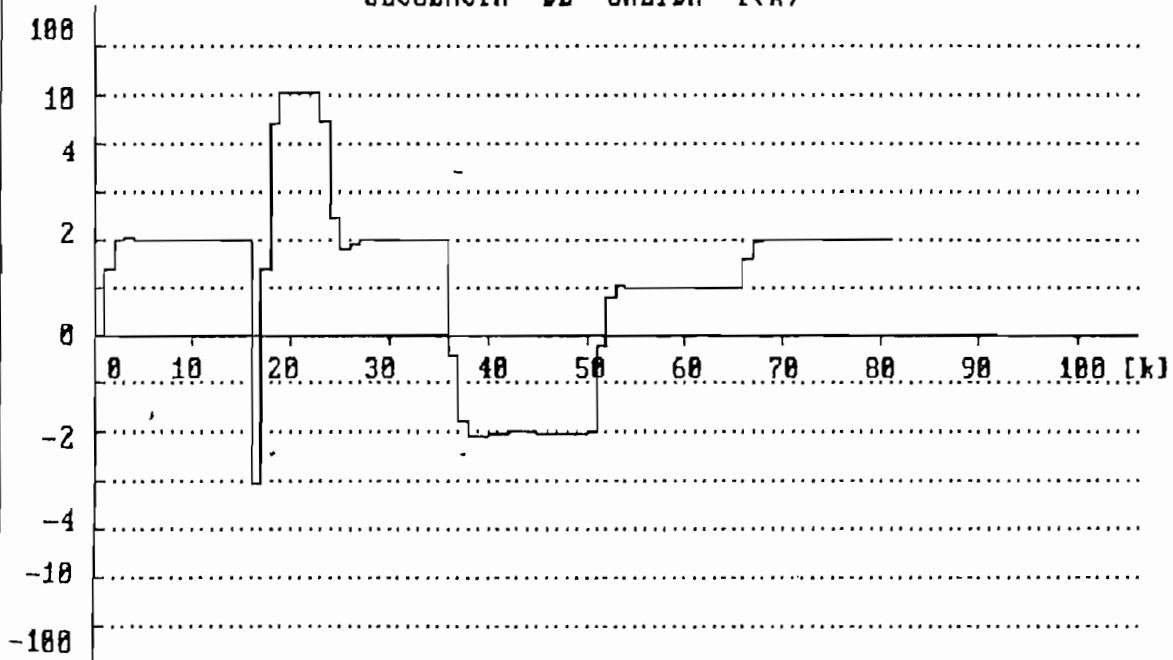
$c_3 = 0.000$

T.MODIFICACION = 15

$F_1 = \text{diag}(100)$

RETARDO $d = 1$

SECUENCIA DE CONTROL $U(k)$ 

SECUENCIA DE REFERENCIA $Y_M(k)$ SECUENCIA DE SALIDA $Y(k)$ 

• RESULTADOS NUMERICOS •

YM(k)	U(k)	Y(k)
YM(0) = 2.000000	U(0) = 0.700000	Y(0) = 0.000000
YM(1) = 2.000000	U(1) = 1.330000	Y(1) = 1.400000
YM(2) = 2.000000	U(2) = 2.037000	Y(2) = 1.960000
YM(3) = 2.000000	U(3) = 1.994300	Y(3) = 2.044000
YM(4) = 2.000000	U(4) = 2.037770	Y(4) = 2.021600
YM(5) = 2.000000	U(5) = 1.993803	Y(5) = 2.004240
YM(6) = 2.000000	U(6) = 2.004682	Y(6) = 1.999536
YM(7) = 2.000000	U(7) = 1.997024	Y(7) = 1.999391
YM(8) = 2.000000	U(8) = 2.001076	Y(8) = 1.999803
YM(9) = 2.000000	U(9) = 1.999361	Y(9) = 1.999983
YM(10) = 2.000000	U(10) = 2.000321	Y(10) = 2.000013
YM(11) = 2.000000	U(11) = 1.999851	Y(11) = 2.000008
YM(12) = 2.000000	U(12) = 2.000079	Y(12) = 2.000002
YM(13) = 2.000000	U(13) = 1.999961	Y(13) = 2.000000
YM(14) = 2.000000	U(14) = 2.000017	Y(14) = 2.000000
YM(15) = 2.000000	U(15) = 1.999991	Y(15) = 2.000000
YM(16) = 2.000000	U(16) = 1.400000	Y(16) = -3.000018
YM(17) = 2.000000	U(17) = 5.946998	Y(17) = 1.400021
YM(18) = 2.000000	U(18) = 19.140070	Y(18) = 6.426991
YM(19) = 2.000000	U(19) = 32.797654	Y(19) = 15.922475
YM(20) = 2.000000	U(20) = 38.999273	Y(20) = 14.660280
YM(21) = 2.000000	U(21) = 44.160229	Y(21) = 14.291077
YM(22) = 2.000000	U(22) = 31.995481	Y(22) = 16.925549
YM(23) = 2.000000	U(23) = 20.687443	Y(23) = 6.650403
YM(24) = 2.000000	U(24) = 14.124997	Y(24) = 2.467580
YM(25) = 2.000000	U(25) = 3.811425	Y(25) = 1.817597
YM(26) = 2.000000	U(26) = 3.168352	Y(26) = 1.907077
YM(27) = 2.000000	U(27) = 5.920233	Y(27) = 1.975882
YM(28) = 2.000000	U(28) = 5.746232	Y(28) = 1.996774
YM(29) = 2.000000	U(29) = 5.049056	Y(29) = 2.001050
YM(30) = 2.000000	U(30) = 5.264952	Y(30) = 2.000665
YM(31) = 2.000000	U(31) = 5.434142	Y(31) = 2.000144
YM(32) = 2.000000	U(32) = 5.334466	Y(32) = 1.999990
YM(33) = 2.000000	U(33) = 5.302971	Y(33) = 1.999981
YM(34) = 2.000000	U(34) = 5.339029	Y(34) = 1.999993
YM(35) = -2.000000	U(35) = 2.940991	Y(35) = 2.000000
YM(36) = -2.000000	U(36) = -0.350506	Y(36) = -0.400286
YM(37) = -2.000000	U(37) = -3.036903	Y(37) = -1.760306
YM(38) = -2.000000	U(38) = -5.561788	Y(38) = -2.064115
YM(39) = -2.000000	U(39) = -5.864043	Y(39) = -2.049614
YM(40) = -2.000000	U(40) = -5.286233	Y(40) = -2.013436
YM(41) = -2.000000	U(41) = -5.245825	Y(41) = -2.000413
YM(42) = -2.000000	U(42) = -5.377121	Y(42) = -1.998821
YM(43) = -2.000000	U(43) = -5.349461	Y(43) = -1.999486
YM(44) = -2.000000	U(44) = -5.315208	Y(44) = -1.999913
YM(45) = -2.000000	U(45) = -5.331552	Y(45) = -2.000017
YM(46) = -2.000000	U(46) = -5.339078	Y(46) = -2.000016
YM(47) = -2.000000	U(47) = -5.332758	Y(47) = -2.000005
YM(48) = -2.000000	U(48) = -5.331748	Y(48) = -2.000001
YM(49) = -2.000000	U(49) = -5.333830	Y(49) = -2.000000
YM(50) = 1.000000	U(50) = -3.533712	Y(50) = -2.000000

SIMULACION CA-6

PLANTA

POLINOMIO A: a 1 = 1.000
 a 2 = 1.000
 a 3 = 1.000

POLINOMIO B: b 0 = 2.000
 b 1 = 1.000
 b 2 = 1.000

MODIFICACIONES EN LA PLANTA

POLINOMIO A: a 1 = -1.200
 a 2 = 0.350

POLINOMIO B: b 0 = 4.000
 b 1 = 1.000
 b 2 = 1.000

POLINOMIO DE CONTROL Cr

c1 = -0.400

c2 = 0.200

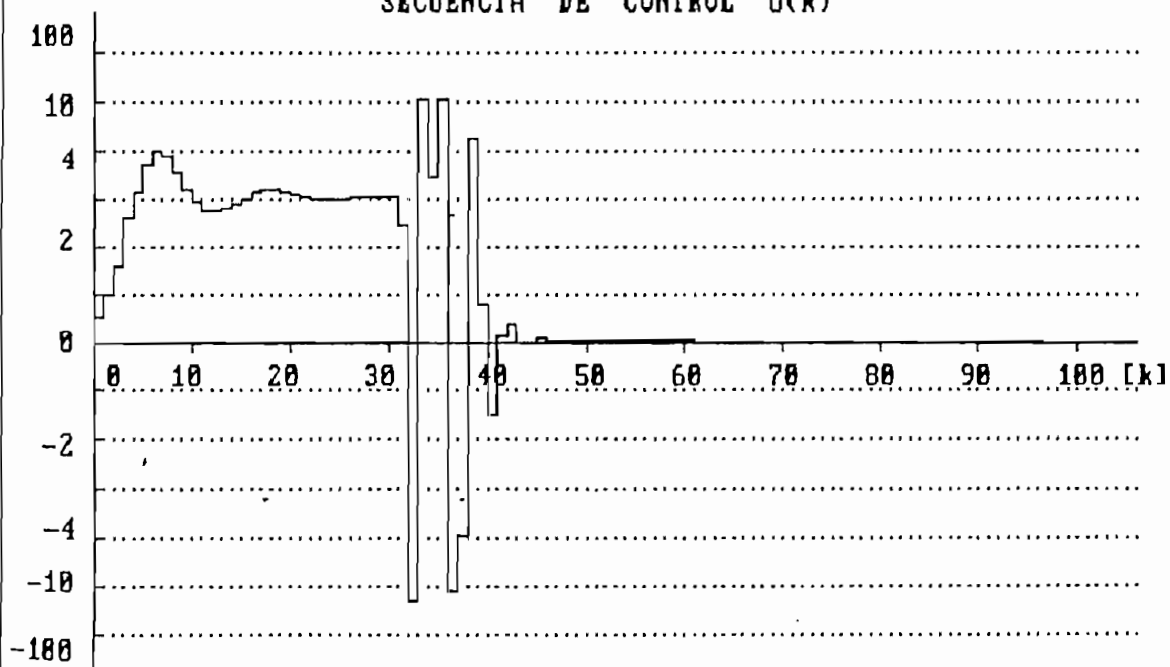
c3 = 0.000

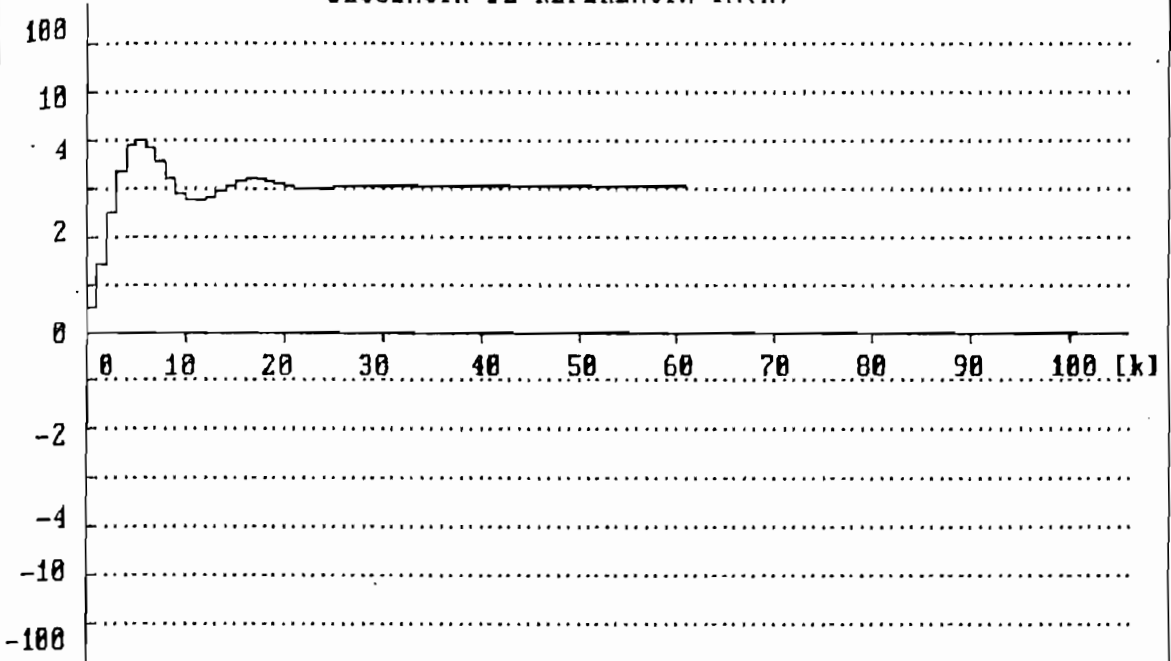
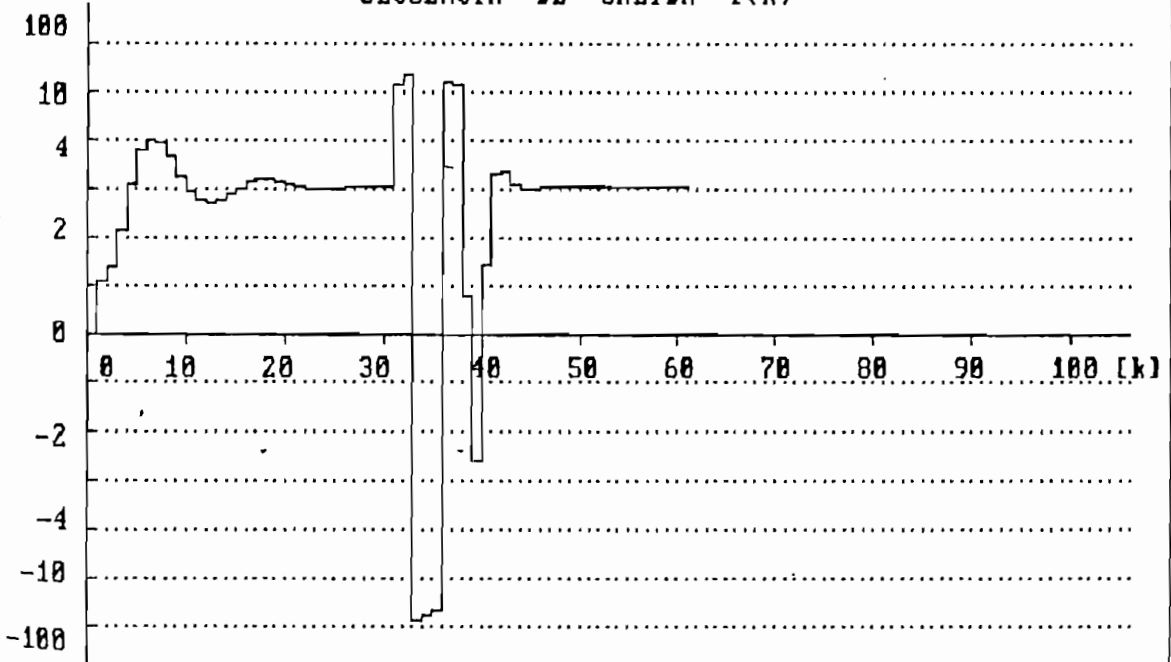
T.MODIFICACION = 30

F1 = diag(100)

RETARDO d = 1

SECUENCIA DE CONTROL U(k)



SECUENCIA DE REFERENCIA $Y_M(k)$ SECUENCIA DE SALIDA $Y(k)$ 

• RESULTADOS NUMERICOS •

YH(k)	U(k)	Y(k)
YH(0) = 0.520000	U(0) = 0.557669	Y(0) = 0.000000
YH(1) = 1.456000	U(1) = 0.990702	Y(1) = 1.115338
YH(2) = 2.501200	U(2) = 1.564524	Y(2) = 1.423735
YH(3) = 3.366480	U(3) = 2.606456	Y(3) = 2.138346
YH(4) = 3.881748	U(4) = 3.146392	Y(4) = 3.090719
YH(5) = 4.016688	U(5) = 3.689892	Y(5) = 3.810963
YH(6) = 3.850772	U(6) = 4.073325	Y(6) = 4.092604
YH(7) = 3.520119	U(7) = 3.894080	Y(7) = 3.988647
YH(8) = 3.164102	U(8) = 3.520715	Y(8) = 3.659162
YH(9) = 2.886838	U(9) = 3.220948	Y(9) = 3.268421
YH(10) = 2.739693	U(10) = 2.935034	Y(10) = 2.940460
YH(11) = 2.723303	U(11) = 2.743944	Y(11) = 2.743686
YH(12) = 2.802701	U(12) = 2.725268	Y(12) = 2.691303
YH(13) = 2.927519	U(13) = 2.800129	Y(13) = 2.754064
YH(14) = 3.050271	U(14) = 2.908704	Y(14) = 2.880417
YH(15) = 3.138340	U(15) = 3.033225	Y(15) = 3.017020
YH(16) = 3.178334	U(16) = 3.130034	Y(16) = 3.123783
YH(17) = 3.173913	U(17) = 3.172946	Y(17) = 3.180778
YH(18) = 3.139661	U(18) = 3.173302	Y(18) = 3.187571
YH(19) = 3.093843	U(19) = 3.144629	Y(19) = 3.157452
YH(20) = 3.052352	U(20) = 3.099684	Y(20) = 3.109704
YH(21) = 3.025118	U(21) = 3.057022	Y(21) = 3.062571
YH(22) = 3.015218	U(22) = 3.028646	Y(22) = 3.028629
YH(23) = 3.020109	U(23) = 3.016623	Y(23) = 3.013093
YH(24) = 3.034008	U(24) = 3.019365	Y(24) = 3.014621
YH(25) = 3.050437	U(25) = 3.032293	Y(25) = 3.027657
YH(26) = 3.064184	U(26) = 3.048485	Y(26) = 3.045203
YH(27) = 3.072337	U(27) = 3.062443	Y(27) = 3.061147
YH(28) = 3.074372	U(28) = 3.071302	Y(28) = 3.071657
YH(29) = 3.071584	U(29) = 3.074143	Y(29) = 3.075524
YH(30) = 3.066180	U(30) = 3.071920	Y(30) = 3.073703
YH(31) = 3.060412	U(31) = 2.449483	Y(31) = 21.045135
YH(32) = 3.055953	U(32) = -32.590543	Y(32) = 40.122363
YH(33) = 3.053614	U(33) = 16.768655	Y(33) = -84.059726
YH(34) = 3.053391	U(34) = 3.445354	Y(34) = -77.980943
YH(35) = 3.054705	U(35) = 15.751774	Y(35) = -66.196702
YH(36) = 3.056738	U(36) = -14.461069	Y(36) = 31.078384
YH(37) = 3.058722	U(37) = -3.953824	Y(37) = 21.815759
YH(38) = 3.060135	U(38) = 5.643304	Y(38) = 0.776885
YH(39) = 3.060767	U(39) = 0.773576	Y(39) = -2.544931
YH(40) = 3.060681	U(40) = -1.440747	Y(40) = 1.457958
YH(41) = 3.060118	U(41) = 0.144790	Y(41) = 3.294169
YH(42) = 3.059374	U(42) = 0.385647	Y(42) = 3.354709
YH(43) = 3.058704	U(43) = -0.018146	Y(43) = 3.119325
YH(44) = 3.058268	U(44) = 0.031571	Y(44) = 3.026898
YH(45) = 3.058114	U(45) = 0.114831	Y(45) = 3.034300
YH(46) = 3.058198	U(46) = 0.077896	Y(46) = 3.054497
YH(47) = 3.058426	U(47) = 0.065758	Y(47) = 3.061380
YH(48) = 3.058692	U(48) = 0.078543	Y(48) = 3.060340
YH(49) = 3.058913	U(49) = 0.078705	Y(49) = 3.058750
YH(50) = 3.059043	U(50) = 0.075490	Y(50) = 3.058504

SIMULACION CA-7

PLANTA

POLINOMIO A: $a_1 = -1.200$
 $a_2 = 0.350$

POLINOMIO B: $b_0 = 1.000$
 $b_1 = -0.700$

MODIFICACIONES EN LA PLANTA

POLINOMIO A: $a_1 = -1.000$
 $a_2 = 0.800$

POLINOMIO B: $b_0 = 0.700$
 $b_1 = -0.200$

POLINOMIO DE CONTROL C_r

$c_1 = -0.250$

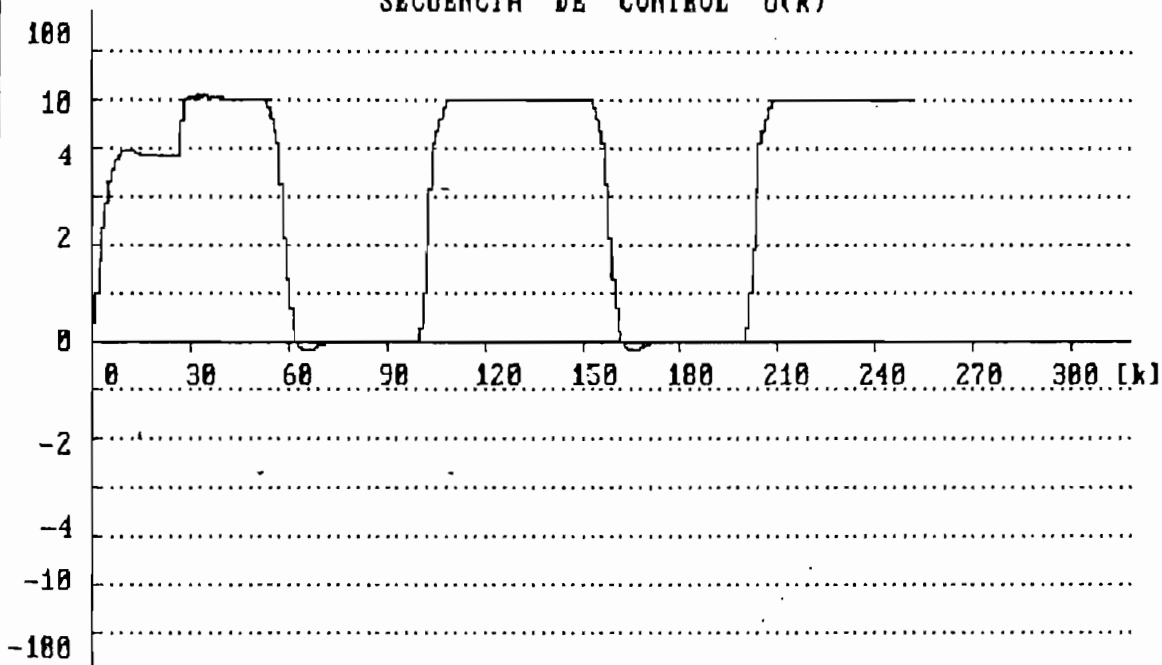
$c_2 = 0.800$

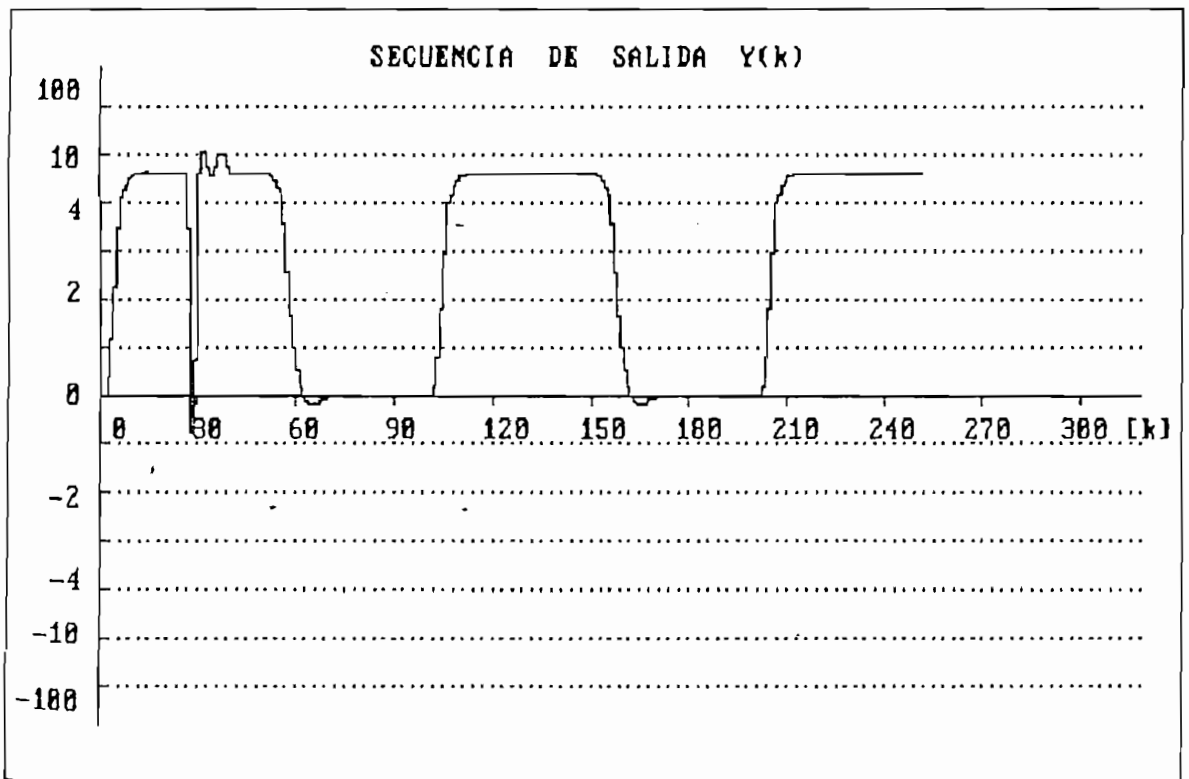
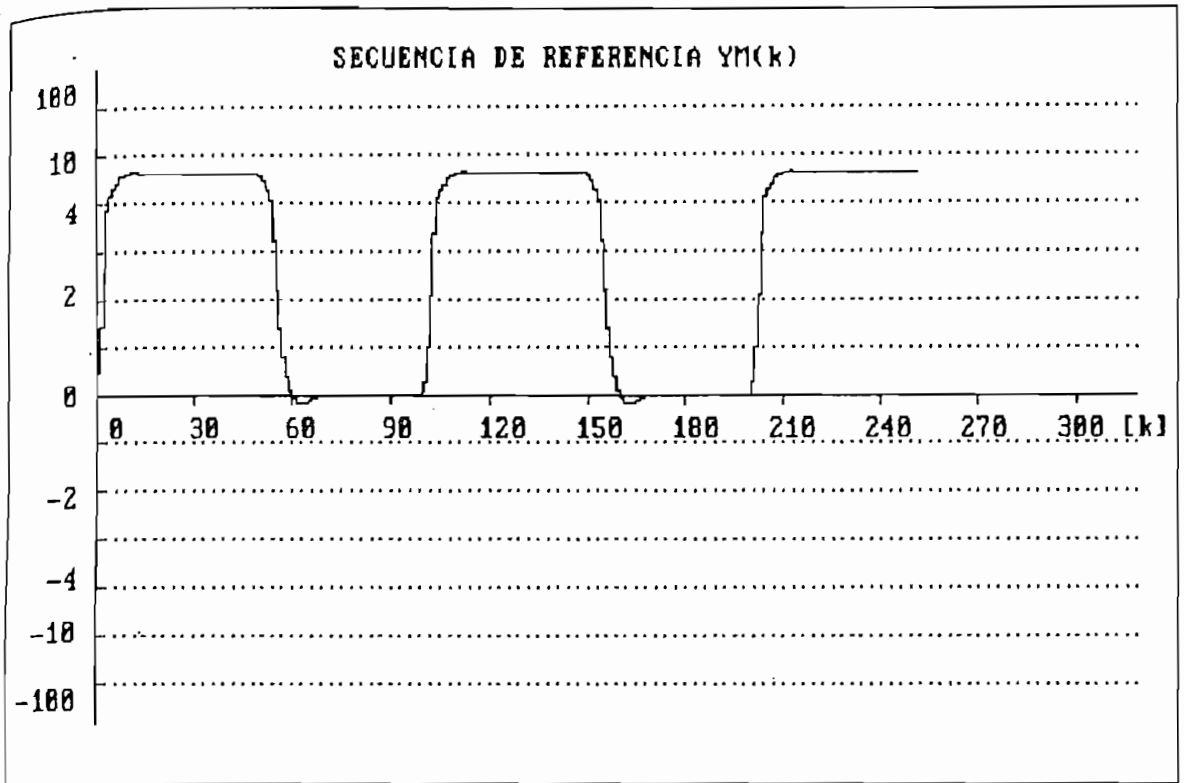
$c_3 = 0.000$

T.MODIFICACION = 25

$F_1 = \text{diag}(100)$

RETARDO $d = 2$

SECUENCIA DE CONTROL $U(k)$ 



• RESULTADOS NUMERICOS •

YM(k)	U(k)	Y(k)
YM(0) = 0.500000	U(0) = 0.375000	Y(0) = 0.000000
YM(1) = 1.450000	U(1) = 0.993750	Y(1) = 0.000000
YM(2) = 2.640000	U(2) = 1.684687	Y(2) = 0.375000
YM(3) = 3.865000	U(3) = 2.329922	Y(3) = 1.181250
YM(4) = 4.980550	U(4) = 2.868518	Y(4) = 2.275313
YM(5) = 5.908695	U(5) = 3.280944	Y(5) = 3.467578
YM(6) = 6.624668	U(6) = 3.572976	Y(6) = 4.602307
YM(7) = 7.139019	U(7) = 3.763255	Y(7) = 5.582098
YM(8) = 7.481598	U(8) = 3.874880	Y(8) = 6.364026
YM(9) = 7.689579	U(9) = 3.930304	Y(9) = 6.945269
YM(10) = 7.799674	U(10) = 3.948739	Y(10) = 7.347515
YM(11) = 7.843822	U(11) = 3.945172	Y(11) = 7.604063
YM(12) = 7.847400	U(12) = 3.930409	Y(12) = 7.750772
YM(13) = 7.829072	U(13) = 3.911630	Y(13) = 7.820559
YM(14) = 7.801547	U(14) = 3.893164	Y(14) = 7.840690
YM(15) = 7.772740	U(15) = 3.877265	Y(15) = 7.831977
YM(16) = 7.747008	U(16) = 3.864794	Y(16) = 7.809154
YM(17) = 7.726254	U(17) = 3.855760	Y(17) = 7.781844
YM(18) = 7.710839	U(18) = 3.849723	Y(18) = 7.755717
YM(19) = 7.700259	U(19) = 3.846059	Y(19) = 7.733621
YM(20) = 7.693616	U(20) = 3.844129	Y(20) = 7.716535
YM(21) = 7.689925	U(21) = 3.843369	Y(21) = 7.704328
YM(22) = 7.688278	U(22) = 3.843328	Y(22) = 7.696294
YM(23) = 7.687929	U(23) = 3.843673	Y(23) = 7.691517
YM(24) = 7.688313	U(24) = 3.844178	Y(24) = 7.689088
YM(25) = 7.689035	U(25) = 3.844703	Y(25) = 7.688219
YM(26) = 7.689843	U(26) = 3.845169	Y(26) = 7.688289
YM(27) = 7.690590	U(27) = 7.186200	Y(27) = 3.460171
YM(28) = 7.691208	U(28) = 11.028114	Y(28) = -0.767783
YM(29) = 7.691678	U(29) = 14.649335	Y(29) = 0.725386
YM(30) = 7.692008	U(30) = 14.287647	Y(30) = 7.622053
YM(31) = 7.692222	U(31) = 10.804282	Y(31) = 15.090655
YM(32) = 7.692346	U(32) = 19.420426	Y(32) = 16.064498
YM(33) = 7.692406	U(33) = 16.986001	Y(33) = 8.697442
YM(34) = 7.692426	U(34) = 17.603820	Y(34) = 7.279285
YM(35) = 7.692421	U(35) = 13.259577	Y(35) = 8.327446
YM(36) = 7.692403	U(36) = 16.467409	Y(36) = 11.429491
YM(37) = 7.692380	U(37) = 13.990912	Y(37) = 10.528475
YM(38) = 7.692358	U(38) = 15.002945	Y(38) = 10.260152
YM(39) = 7.692339	U(39) = 13.660278	Y(39) = 8.337529
YM(40) = 7.692325	U(40) = 12.808040	Y(40) = 7.833285
YM(41) = 7.692314	U(41) = 12.489208	Y(41) = 7.724868
YM(42) = 7.692307	U(42) = 12.358598	Y(42) = 7.691811
YM(43) = 7.692302	U(43) = 12.322645	Y(43) = 7.692755
YM(44) = 7.692299	U(44) = 12.312046	Y(44) = 7.692482
YM(45) = 7.692298	U(45) = 12.309024	Y(45) = 7.692410
YM(46) = 7.692298	U(46) = 12.308087	Y(46) = 7.692327
YM(47) = 7.692298	U(47) = 12.307803	Y(47) = 7.692306
YM(48) = 7.692299	U(48) = 12.307716	Y(48) = 7.692300
YM(49) = 7.692300	U(49) = 12.307690	Y(49) = 7.692289
YM(50) = 7.412301	U(50) = 12.007688	Y(50) = 7.692299

6.2 CONCLUSIONES

El desarrollo y análisis de los contenidos de la presente tesis, así como también el estudio de los resultados obtenidos permiten establecer las siguientes conclusiones :

- En primera instancia cabe indicar que en términos generales el presente trabajo es una contribución que tiene el carácter de complementaria para la formación teórico-práctica en el área del Control Automático. Se han investigado técnicas relativamente actuales en el control de procesos y se ha establecido una solución idónea para el control de aquellos procesos cuyos parámetros son desconocidos o varían eventualmente en el tiempo.
- Se ha demostrado que un proceso cuyos parámetros no sean conocidos o cambien durante el funcionamiento dentro de un rango amplio puede ser controlado eficientemente utilizando para el efecto un controlador de tipo adaptivo; de las múltiples técnicas existentes, la correspondiente a MRAC se presenta como una de las potencialmente aplicables por su versatilidad y eficiencia.
- Los esquemas de control lineal y adaptivo considerados en el presente trabajo permiten especificar independientemente los objetivos de seguimiento y regulación. Esta particularidad no es común en la mayoría de los esquemas de control adaptivo existentes.

- El objetivo de seguimiento se lo determina a través de la secuencia de referencia $y_M(k)$ y el objetivo de regulación se especifica con el polinomio $C_r(q^{-1})$. La independencia se consigue debido a que la entrada de control $u(k)$ se calcula utilizando la salida de referencia $y_M(k)$ y no la entrada a un modelo de referencia que genera $y_M(k)$. Al utilizar un modelo de referencia para generar $y_M(k)$ la dinámica de ese modelo puede especificarse independientemente del polinomio $C_r(q^{-1})$.

- El papel que desempeña el polinomio $C_r(q^{-1})$ es de fundamental importancia. Las simulaciones realizadas demuestran que entre más próximos estén sus ceros del origen más rápido será cero el error de salida, pero siempre a expensas de más energía en la entrada de control.

- Consecuentemente, en la selección del polinomio $C_r(q^{-1})$ debe existir un compromiso entre la rapidez con que se anula el error y la energía que se desea emplear en ello.

- Cuando la planta del sistema a controlar tiene sus ceros dentro del círculo unitario pero muy próximos a este, o a la izquierda del plano z , se presentan efectos indeseables tales como entradas de control oscilatorias y/o de muy alta magnitud.

- Las simulaciones realizadas demuestran que los esquemas propuestos pueden ser aplicados a sistemas de fase mínima, pero no deben ser aplicados a sistemas de fase no mínima.

- En el algoritmo de mínimos cuadrados recursivo con factor de olvido la matriz de ganancias $F(k)$ no converge a cero como en el caso del algoritmo de mínimos cuadrados recursivo sin factor de olvido ($\alpha_k=1$). Consecuentemente, la introducción del factor de olvido proporciona características deseables para su implementación con un esquema de control adaptivo. Dado que la ganancia nunca es cero, el algoritmo está siempre en potencia de identificar un cambio eventual en los parámetros del proceso.

- El tiempo de ejecución depende del número de operaciones que debe realizar el algoritmo, principalmente en la tarea de estimación de los parámetros del controlador. Mientras mayor sea la cantidad de parámetros a estimarse mayor será la cantidad de operaciones por cuanto las dimensiones matriciales se incrementan. A su vez, el número de parámetros se relaciona directamente con el orden de la planta.

- La implementación del programa mediante el compilador QuickBasic 4.0 ofrece varias ventajas, pudiéndose señalar entre las principales la portabilidad del programa para ser ejecutado en muchos computadores personales sin mayores limitaciones de hardware y la alta velocidad de ejecución. Aunque el QuickBasic 4.0 no es fuertemente estructurado se ha procurado mantener en el desarrollo de todo el trabajo un diseño modular mediante la aplicación de las técnicas de programación estructurada. De esta forma, el programa podría ser codificado sin mayores dificultades en cualquier lenguaje estructurado moderno.

6.3 RECOMENDACIONES

- La importancia de la técnica de control adaptivo considerada podría ser vista con mayor objetividad mediante su utilización en alguna aplicación en tiempo real. Para ello se deberán realizar los acondicionamientos necesarios en el programa de forma tal que éste pueda funcionar adecuadamente en base a la correspondiente etapa de adquisición de datos.
- En la ley de control adaptivo aplicada aparece una división por el estimado $\hat{b}_0(k)$. Es necesario tomar precauciones cuando $|\hat{b}_0(k)|$ adquiere valores menores que un cierto valor mínimo. La forma más sencilla es la de reemplazar el estimado por un valor constante cuando $|\hat{b}_0(k)|$ sea muy pequeño; no obstante, podrían investigarse mejores alternativas.
- En vista de que las técnicas de control presentadas no pueden ser aplicadas a sistemas de fase no mínima se deberá tener en cuenta que al discretizar un sistema continuo se puede encontrar un sistema de fase no mínima a pesar de que la planta en continuo de la cual se parte sea de fase mínima.
- Se puede mejorar sustancialmente la rapidez en el procesamiento computacional, particularmente cuando se trabaja con sistemas de orden alto, utilizando un sistema

informático que disponga de coprocesador matemático. Para aplicaciones en tiempo real esta disposición sería fundamental.

- La etapa de identificación paramétrica, efectuada mediante la IRMC en el presente trabajo, puede también ser implementada recurriendo a otros métodos de identificación. Para problemas específicos de control podrían obtenerse mejores resultados utilizando alguno de éstos otros métodos.
- Se observa que la importancia del control adaptivo radica en sus características de versatilidad y eficiencia, no obstante, debe tenerse en cuenta que no todos los procesos requieren de un control adaptivo sofisticado y que en muchos casos un controlador convencional (PI, PID por ejemplo) puede satisfacer los objetivos deseados.
- Se recomienda complementar este trabajo mediante el estudio de temas correspondientes a: control adaptivo de sistemas de fase no mínima, control adaptivo multivariable, control adaptivo estocástico y temas específicos de aplicación tales como controladores PI y PID autoajustables, ya que en la actualidad la incidencia del control adaptivo en el control automático de procesos (principalmente industriales) es notable.

APENDICES

APENDICE B

VERIFICACION DE EQUIVALENCIA ENTRE LAS
EXPRESIONES [4.6] Y [4.7]

Considérese la expresión [4.6] respectivamente dada por:

$$B(q^{-1})S(q^{-1})u(k) + R(q^{-1})y(k) - Cr(q^{-1})yM(k) = 0$$

donde :

$$B(q^{-1}) = b_0 + b_1q^{-1} + b_2q^{-2} + \dots + b_nBq^{-nB}$$

$$S(q^{-1}) = 1 + s_1q^{-1} + s_2q^{-2} + \dots + s_nSq^{-nS}$$

$$R(q^{-1}) = r_0 + r_1q^{-1} + r_2q^{-2} + \dots + r_nRq^{-nR}$$

$$Cr(q^{-1}) = 1 + c_1q^{-1} + c_2q^{-2} + \dots + c_nCq^{-nC}$$

,Según [2.13] $nS = d-1$. Entonces [4.6] puede ser escrita como :

$$(b_0 + b_1q^{-1} + \dots + b_nBq^{-nB})(1 + s_1q^{-1} + \dots + s_{d-1}q^{-d+1})u(k) +$$

$$+ (r_0 + r_1q^{-1} + \dots + r_nRq^{-nR})y(k) - Cr(q^{-1})yM(k) = 0$$

Operando internamente se obtiene :

$$[b_0 + (b_{0s1} + b_1)q^{-1} + (b_{0s2} + b_1s_1 + b_2)q^{-2} + \dots + (b_{nBSd-1})q^{-d-nB+1}] \cdot$$

$$u(k) + (r_0 + r_1q^{-1} + \dots + r_{nR}q^{-nR})y(k) - Cr(q^{-1})y(k) = 0$$

Expresión equivalente a :

$$\begin{aligned} & b_0u(k) + (b_{0s1} + b_1)u(k-1) + (b_{0s2} + b_1s_1 + b_2)u(k-2) + \dots + \\ & + (b_{nBSd-1})u(k-d-nB+1) + r_0y(k) + r_1y(k-1) + r_2y(k-2) + \dots + \\ & + r_{nR}y(k-nR) - Cr(q^{-1})y(k) = 0 \end{aligned}$$

y que puede ser escrita como :

$$b_0u(k) + \theta_0^T \phi_0(k) - Cr(q^{-1})y(k) = 0 \quad [4.7]$$

donde :

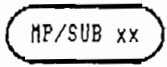
$$\theta_0^T = [b_{0s1} + b_1, b_{0s2} + b_1s_1 + b_2, \dots, b_{nBSd-1}, r_0, r_1, \dots, r_{nR}]$$

$$\phi_0^T = [u(k-1), u(k-2), \dots, u(k-d-nB+1), y(k), y(k-1), \dots, y(k-nR)]$$

APENDICE C

SIMBOLOGIA

La simbología utilizada en la implementación de los diagramas es la siguiente:



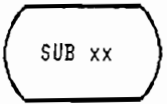
al inicio de un diagrama indica el punto de partida del diagrama correspondiente al módulo o subprograma cuya denominación es xx.



finalización de un módulo o subprograma



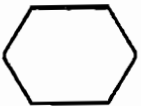
ingreso de datos por teclado



en el contexto de un diagrama indica "llamada" al subprograma cuya denominación es xx



acción de procesamiento



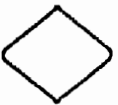
procesamiento en lazo iterativo



presentación en pantalla de información



presentación impresa de información



bifurcación lógica



conector

APENDICE DMANUAL DE USUARIO DEL PROGRAMA

En este apéndice se anotan las características del equipo en el que se pueden ejecutar los programas de simulación así como también una guía de operación de los mismos.

El usuario deberá estar familiarizado con el manejo de computadores personales y con el sistema operativo DOS.

D.1 CONFIGURACION DEL EQUIPO

En vista de que los programas fueron desarrollados utilizando un computador personal, su empleo tendrá lugar en un equipo compatible que reúna las siguientes características:

CPU	microprocesador 8088/286 640 KB RAM (mínimo 512KB RAM) adaptador de gráficos (720x350 pixels) 1 FDD + 1 HDD (opcional) coprocesador matemático 8087 (opcional)
MONITOR	con capacidad para gráficos
SISTEMA OPERATIVO	DOS versión 3.X o más reciente
IMPRESORA (opc.)	con capacidad de impresión para gráficos

D.2 ARCHIVOS

Los programas de simulación y el software necesario para su ejecución se presentan en tres discos flexibles 5¹/₄, DS, DD. El contenido de cada uno de ellos se especifica a continuación :

DISCO #1

TITULO : INSTALACION PZ
ARCHIVOS : INSTALL.COM
INSTALL.EDM
INSTALL.CMP
PZ.COM
PZ.PZO
PZ.PZS

DISCO #2

TITULO : OPERATIVO CAD
ARCHIVOS : COMMAND.COM
CONFIG.SYS
AUTOEXEC.BAT
QB.INI
QBHERC.EXE
QB.EXE
CAD.BAS

DISCO #3

TITULO : SIMULACION CAD
ARCHIVOS : CAD.EXE
BRUN40.EXE
LINK.EXE
BC.EXE
BRUN40.LIB
QBHERC1.EXE

La disponibilidad de un HDD permitirá mantener en un solo directorio todos los archivos en cuestión.

D.3 EJECUCION DE LOS PROGRAMAS

Una vez que el sistema entre en funcionamiento mediante una versión adecuada del sistema operativo DOS se efectúan los siguientes pasos :

1. SI DESEA TRABAJAR EN AMBIENTE DOS

- > Coloque el DISCO #1 en el drive A y ejecute el programa PZ.EXE (opc.)
- > Coloque el DISCO #3 en el drive A y ejecute secuencialmente los programas :
 - QBHERC.EXE (instalación adaptador Hércules-opc.)
 - CAD.EXE
- El programa PZ es un utilitario autoejecutable que permite efectuar adecuadamente las tareas de impresión.
- CAD.EXE es el archivo compilado y autoejecutable que contiene los programas de simulación.

2. SI DESEA TRABAJAR EN AMBIENTE QUICKBASIC

- > Coloque el DISCO #1 en el drive A y ejecute el programa PZ.EXE (opc.)
- > Coloque el DISCO #2 en el drive A y ejecute secuencialmente los programas :
 - QBHERC.EXE
 - QB.EXE
- Se visualizará entonces en la pantalla la ventana de trabajo de QuickBasic. Seleccione la opción File presionando la tecla <Alt>.
- Seleccione la opción Open a program.
- Seleccione el programa CAD.BAS y presione <Enter>.
- Ejecute CAD.BAS presionando <Shift>-<F5>.

Cualquiera de los dos procedimientos anteriores le

permiten ejecutar los programas de simulación. Cuando se trabaja en ambiente QuickBasic se tendrá además acceso directo a la codificación de los programas.

Inicialmente se visualiza una carátula informativa y a continuación el siguiente menú principal :

MENU PRINCIPAL

1. Simulación de Control Lineal
2. Simulación de Control Adaptivo MRAC
3. Terminar

D.4 SIMULACION DE CONTROL LINEAL

El procedimiento a efectuarse es el siguiente :

- > En el MENU PRINCIPAL seleccione la opción 1. Para seleccionar una opción de cualquier menú presione únicamente el número o carácter indicativo de la misma, no se requiere presionar seguidamente la tecla <Enter>.
- > Se visualiza una carátula informativa. Presione <C> para continuar.
- > Aparece en pantalla información básica sobre la generación de la secuencia de referencia. Espere.
- > Se presenta el menú para la generación de la secuencia de referencia.

GENERACION DE LA SEC. DE REFERENCIA

1. PUNTO POR PUNTO
2. MEDIANTE FUNCION DE TRANSFERENCIA

- > Si desea generar PUNTO POR PUNTO seleccione la opción respectiva.
 - Indique el número de puntos de la secuencia (máx. 300)
 - Asigne los valores para c/punto según el programa le vaya requiriendo.
 - Al finalizar seleccione <S> o <N> para revisar y/o modificar los valores ingresados. Presione <M> para modificar el valor correspondiente.
 - Seleccione <S> o <N> según desee observar o no gráficamente la secuencia generada. Luego presione cualquier tecla para continuar.

 - > Si desea generar mediante una FUNCION DE TRANSFERENCIA seleccione la opción respectiva.
 - Indique el # de puntos de la secuencia (máx. 500). Espere.
 - Ingrese los coeficientes de los polinomios L y K según le requiere el programa.
 - $UM(k)$ es la secuencia de entrada al modelo de referencia. Puede ser una secuencia de tipo escalón con varios niveles. Asigne un valor inicial para ella y conteste <S> o <N> en el caso de tener o no modificaciones en sus valores.
Si tiene modificaciones indique el número de ellas y asigne sus valores e instantes de ocurrencia según le vaya requiriendo el programa.
 - Seleccione <S> o <N> según desee observar o no gráficamente la secuencia generada. Presione cualquier tecla para continuar y seleccione <S> o <N> para confirmar o no la validez de la referencia generada.

 - > Indique los grados de los polinomios A y B de la planta y asigne los valores de sus coeficientes según le requiera el programa.

 - > Si la planta ingresada es de fase mínima el programa sigue su curso normal, caso contrario notifica el incumplimiento de la condición y pide ingresar una nueva planta o terminar la sesión de trabajo. Seleccione la opción correspondiente.

 - > Seleccione a continuación el retardo del sistema escogiendo en el menú respectivo la opción adecuada.
-

- > Indique el grado del polinomio Cr y asigne los valores de sus coeficientes.
- > A partir de este instante el programa efectúa el procesamiento de simulación. Espere.
- > Se visualiza a continuación el siguiente menú para presentación de resultados.

PRESENTACION DE RESULTADOS

1. EN FORMA NUMERICA
2. EN FORMA GRAFICA
3. EN FORMAS NUMERICA Y GRAFICA

Seleccione la opción correspondiente.

- Cuando se visualiza numéricamente los resultados se puede imprimirlos directamente presionando <S> cuando el programa le consulte sobre este propósito. Cerciónese que la impresora este activada y en condiciones de operar.
- Los resultados gráficos pueden imprimirse presionando simultáneamente las teclas <Shift>/<Print Screen> Este procedimiento le da acceso al utilitario PIZAZZ que presenta en pantalla un menú con varias opciones. Se recomienda utilizar:

```
Color      : _Reversed
Width      : 7.0 inch.
Height     : 4.7 inch.
Form feed  : No
Density    : Medium
```

Una vez que se han establecido las condiciones adecuadas seleccione para la impresión la opción Print. Cerciónese que la impresora esté en condiciones de operar.

Si la impresora que se utiliza es de 132 o más columnas pueden seleccionarse otras dimensiones para los gráficos a imprimirse.

- Al seleccionar la forma de presentación gráfica y numérica se tiene acceso a todas las disponibilidades señaladas anteriormente.

- > Al finalizar la presentación de resultados se visualiza

en pantalla opciones para efectuar otra simulación (va al menú principal) o terminar la sesión de trabajo.

D.5 SIMULACION CONTROL ADAPTIVO MRAC

El procedimiento a efectuarse es el siguiente :

- > En el MENU PRINCIPAL seleccione la opción 2.
- > Se visualiza una carátula informativa. Presione <C> para continuar.
- > Se presenta el menú para la generación de la secuencia de referencia.
 - Para la generación de la secuencia de referencia proceda de manera análoga a lo establecido para el efecto en la sección D.4 precedente.
- > A continuación el programa solicita los DATOS DE LA PLANTA INICIAL.
 - Indique los grados de los polinomios A y B.
 - Ingrese los coeficientes de A y B según le requiera el programa.
 - Si la planta no es de fase mínima se notifica el incumplimiento de la condición y se solicita ingresar una nueva planta o terminar la sesión de trabajo.
- > Se visualizan a continuación las siguientes opciones:

ALTERACIONES EN LA PLANTA

- A. GRADOS DE A y B SIN MODIFICACIONES
- B. GRADOS DE A y/o B CON MODIFICACIONES

- Si se selecciona A el programa pasará inmediatamente a solicitarle los datos modificados de la planta.
 - Si se selecciona B el programa primero le requiere los nuevos grados de los polinomios A y B.
-

- > El programa solicita los DATOS DE LA NUEVA PLANTA.
 - Ingrese los coeficientes de A y B según le vaya requiriendo el programa.
 - Análogamente el programa verifica el cumplimiento de la condición de fase mínima.

 - > Asigne cierta respuesta a la siguiente interrogante que le hace el programa:
LA PLANTA SE MODIFICA AL INSTANTE (máx. xx) k = ?

 - > Seleccione una opción en el menú para la determinación del retardo del sistema.

 - > Se solicita a continuación información sobre el polinomio Cr de control. De una respuesta a :
EL GRADO DEL POLINOMIO Cr ES (máx x).....?
Asigne sus coeficientes según se le requiera.

 - > Ingrese el valor de la diagonal de inicialización de la matriz F[i,j].

 - > Seguidamente el programa le consulta :
SE ASUME CONOCIDA LA PLANTA INICIAL ?...<S>/<N> :
Seleccione una de las opciones de acuerdo a su interés.

 - > Con toda la información precedente el programa de simulación comienza a trabajar en la etapa de procesamiento. Espere.

 - > Se visualiza finalmente el menú para presentación de resultados.
 - Proceda de acuerdo a lo establecido para el efecto en la sección D.4 precedente.

 - > Seleccione una de las opciones correspondientes a la ejecución de una nueva simulación o terminación de la sesión de trabajo.
-

BIBLIOGRAFIA

GOODWIN G.C. and SIN S.K., Adaptive Filtering Prediction and Control, Prentice-Hall, NJ., 1984.

KAILATH T., Linear Systems, Prentice-Hall, NJ., 1982.

KUO B.C., Digital Control Systems, NY., McGraw-Hill, 1981.

KUO B.C., Sistemas Automáticos de Control, NJ., Prentice-Hall, 1979.

LANDAU I.D., Adaptive Control - The Model Reference Approach, NY., M. Dekker, 1979.

NARENDRA K.S., Adaptive and Learning Systems - Theory and Applications, NY., Plenum Press, 1986.

REFERENCIAS

- ASTROM K.J., Adaptive Feedback Control, Proc. IEEE, Vol-75, pp. 185-209, 1987.
- ASTROM K.J., Model Reference Adaptive Controllers, Automática, Vol-19, pp. 467-471, 1983.
- BARRAGAN B.M., Identificación y Control, FIE-EPN, Dpto. Electrónica y Control, 1986.
- CORDERO P., Mínimos Cuadrados Generalizado, Tesis FIE-EPN, 1985.
- EGART B.I., Unification of some discrete time Adaptive Control Schemes, IEEE-TAC, Vol AC-25, pp. 693-697, 1979.
- EGART B.I., A Unified Approach to Model Reference Adaptive Systems and STR Regulators, Automatica, Vol-17, pp. 521-525, 1981.
- ELLIOT H. and WOLOWICH W., Parameter Adaptive Identification and Control, IEEE-TAC, Vol AC-24, pp. 592-599, 1979.
- FEUER A. and MORSE S., Adaptive Control of single input-single output systems, IEEE-TAC, Vol AC-23, pp. 557-569, 1978.
- GOODWING G., RAMADGE P. and CAINES E., Discrete Time Adaptive Control, IEEE-TAC, Vol AC-25, pp. 449-456, 1980.
- ISERMANN R., Practical aspects of Process Identification, Automatica, Vol-16, pp. 575-587, 1980.

LANDAU I. and LOZANO R., Unification of Discrete Time Explicit Model Reference Adaptive Control Designs, Automatica, Vol-17, pp. 593-611, 1981.

LANDAU I. and LOZANO R., Redesign of Explicit and Implicit Discrete Time Model Reference Adaptive Control Schemes, Int. J. Control, Vol-33, pp. 245-251, 1983.

LOZANO R.L., Independent Tracking and Regulation Adaptive Control with Forgetting Factor, Automatica, Vol-18, pp. 455-459, 1982.

MARTINEZ G.R., Técnicas Convencionales de Identificación, Tesis FIE-EPN, 1986.

MORSE S.D., Global Stability of Parameter-Adaptive Control Systems, IEEE-TAC, Vol AC-25, pp. 433-439, 1980.

NARENDRA K. and VALAVANI L., Direct and Indirect Adaptive Control, Automatica, Vol-15, pp. 653-664, 1979.

NARENDRA K. and VALAVANI L., Stable Adaptive Design-Direct Control, IEEE-TAC, Vol AC-23, pp. 570-582, 1978.

RACINES S., Identificación Paramétrica Discreta, Tesis FIE-EPN, 1988.

VALLEJO J.C., Control Adaptivo de un Proceso Pneumático, Simposio en Control Automático, FIE-EPN, 1987.

```
=====
MODULO PRINCIPAL
=====
```

```
' Llamado a subprograma de presentacion
```

```
CALL PRESENTACION
```

```
1: 'Referencia de inicio
```

```
CLEAR
```

```
DIM A(4), B(4), C1(4)
```

```
DIM UM(510), YM(510)
```

```
DIM Y(510) AS DOUBLE, U(510) AS DOUBLE
```

```
DIM A1(4), B1(4), CAMBIO1(3), D(7)
```

```
DIM G1(4), G2(4), G3(4), G4(4)
```

```
DIM VBAUX(510)
```

```
' Llamado al subprograma del menu principal
```

```
CALL MENU.PRIN
```

```
IF NF$ = "NO" THEN GOTO 1
```

```
2: 'Referencia de finalizacion
```

```
END
```

```
'*****
' SUBROUTINAS PARA VERIFICACION DE ERRORES
'*****
```

```
IMPR:
```

```
CLS
```

```
LOCATE 3, 2
```

```
PRINT STRING$(22, " ")
```

```
LOCATE 4, 2
```

```
PRINT "VERIFIQUE LA IMPRESORA"
```

```
LOCATE 5, 2
```

```
PRINT STRING$(22, " ")
```

```
RESUME
```

```
OVER:
```

```
CLS
```

```
PRINT
```

```
PRINT STRING$(22, " ")
```

```
PRINT "SIMULACION SUSPENDIDA"
```

```
PRINT STRING$(22, " ")
```

```
PRINT
```

```
PRINT "DATOS INCOMPATIBLES"
```

```
PRINT
```

```
PRINT
```

```
PRINT "## Presione <K> ##"
```

```
OVER$ = INPUT$(1)
RESUME NEXT
```

```
HGR:
```

```
CLS
PRINT
PRINT STRING$(22, "T")
PRINT "RANGO GRAFICO INCOMPATIBLE"
PRINT STRING$(22, "I")
PRINT
PRINT
PRINT "##Presione <K>##"
HGR$ = INPUT$(1)
RESUME NEXT
```

```
=====
SUBPROGRAMA CONTROL LINEAL : CLINEAL
=====
```

```
SUB CLINEAL
'Presentacion
ON ERROR GOTO OVER
CLS
'Transferencia a modo grafico
SCREEN 3
LINE (0, 0)-(650, 320), , B
LINE (40, 40)-(610, 280), , B
LINE (43, 43)-(607, 277), , B
LINE (46, 46)-(604, 274), , BF
'
'Leyendas
LOCATE 10, 13
PRINT " # P R O G R A M A   D E   S I M U L A C I O N # "
LOCATE 14, 20
PRINT " # C O N T R O L   L I N E A L # "
LOCATE 22, 5
PRINT "Presione [C] para cont.,[Q] para regresar al menu principal : ";
OPTION2$ = INPUT$(1)
'
'Seleccion
SELECT CASE OPTION2$
CASE "Q", "q"
'Llamado a MENU.PRIN
=====
CALL MENU.PRIN
=====
CASE "C", "c"
'Llamado a GENERACION de Modelo de Ref.
=====
CALL GENERACION
=====
IF NF$ = "NO" THEN GOTO CLA1
```

```

CASE ELSE
'Autollamado a CLINEAL
PLAY "E30D40"

```

```

-----
CALL CLINEAL

```

```

-----
END SELECT

```

```

'Vuelve el control a CLINEAL
'Llamado a INGRESO de la planta

```

```

PLAN$ = "¡ INGRESE LOS DATOS DE LA PLANTA !"

```

```

-----
CALL INGRESOPLANTA

```

```

-----
CALL FASEMIN

```

```

-----
CALL RETARDO

```

```

-----
CALL RESULTADOS.1

```

```

-----
CALL FIN

```

```

CLAL: 'Referencia.
END SUB

```

```

=====
SUBPROGRAMA ESTAD01
=====

```

```

SUB ESTAD01

```

```

ON ERROR GOTO OVER
SCREEN 3

```

```

'Este subprograma visualiza los datos

```

```

CLS
LINE (0, 0)-(660, 345), , B
VIEW PRINT

```

```

'Seleccion de acuerdo a TIPO$

```

```

SELECT CASE TIPO$

```

```

CASE "CLINEAL"

```

```

LINE (205, 11)-(469, 30), , BF

```

```

LOCATE 2, 25

```

```

PRINT " SIMULACION CONTROL LINEAL "

```

```

LOCATE 5, 35

```



```

LINE (296, 55)-(370, 70), , B
PRINT "PLANTA"
LOCATE 8, 15
PRINT "POLINOMIO A : "
FOR I = 1 TO N
  LOCATE I + 7, 45
  PRINT "a"; I; " = ";
  PRINT USING "###.###"; a(I)
NEXT I
LOCATE 12, 15
PRINT "POLINOMIO B : "
FOR I = 0 TO M
  LOCATE I + 12, 45
  PRINT "b"; I; " = ";
  PRINT USING "###.###"; b(I)
NEXT I
LINE (225, 223)-(446, 238), , B
LOCATE 17, 27
PRINT "POLINOMIO DE CONTROL Cr"

SELECT CASE GC

CASE 0
LOCATE 21, 36
PRINT "Cr = 1"

CASE IS > 0
FOR I = 1 TO GC
  LOCATE I + 19, 32
  PRINT "c"; I; " = ";
  PRINT USING "###.###"; C1(I) 'C1() coef. de Cr
NEXT I

END SELECT
LOCATE 24, 32
PRINT "RETARDO d = "; RET$

CASE "MRAC"
CLS
LINE (0, 0)-(660, 340), , B

LINE (205, 11)-(486, 30), , BF
LOCATE 2, 25
PRINT " SIMULACION CONTROL ADAPTIVO "
LOCATE 5, 35
LINE (296, 55)-(370, 70), , B
PRINT "PLANTA"
LOCATE 7, 4
PRINT "POLINOMIO A:"
FOR I = 1 TO GK1
  LOCATE 6 + I, 17
  PRINT "a"; I; " = ";
  PRINT USING "###.###"; G1(I)
NEXT I
LOCATE 7, 43
PRINT "POLINOMIO B:"
FOR I = 0 TO GK2
  LOCATE 7 + I, 56
  PRINT "b"; I; " = ";
  PRINT USING "###.###"; G2(I)
NEXT I

```

```

LINE (187, 153)-(475, 168), , B
LOCATE 12, 23
PRINT "MODIFICACIONES EN LA PLANTA"

```

```

LOCATE 13, 23
LOCATE 14, 4
PRINT "POLINOMIO A:"
FOR I = 1 TO GK3
  LOCATE 13 + I, 17
  PRINT "a"; I; " = ";
  PRINT USING "###.###"; G3(I)
NEXT I
LOCATE 14, 43
PRINT "POLINOMIO B:"
FOR I = 0 TO GK4
  LOCATE 14 + I, 56
  PRINT "b"; I; " = ";
  PRINT USING "###.###"; G4(I)
NEXT I

```

```

LINE (207, 251)-(457, 266), , B
LOCATE 19, 25
PRINT "POLINOMIO DE CONTROL Cr"

```

```

LOCATE 21, 4
PRINT "c1 = ";
PRINT USING "###.###"; C1(1)
LOCATE 21, 31
PRINT "c2 = ";
PRINT USING "###.###"; C1(2)
LOCATE 21, 58
PRINT "c3 = ";
PRINT USING "###.###"; C1(3)
LOCATE 23, 4
PRINT "T.MODIFICACION = "; TEMPI
LOCATE 23, 32
PRINT "F1 = diag("; DIAG; ")"
LOCATE 23, 57
PRINT "RETARDO d = "; RETd

```

```

END SELECT

```

```

OPd = INPUT$(1)

```

```

END SUB

```

```
=====
SUBPROGRAMA FASEMIN
=====
```

```
SUB FASEMIN
```

```
'Variables compartidas
```

```
SHARED B(), D
```

```
ON ERROR GOTO OVER
```

```
IF B(3) <> 0 THEN 'Ecuacion de 3er.Grado
```

```
F = B(1) / B(0)
```

```
G = B(2) / B(0)
```

```
H = B(3) / B(0)
```

```
F = F / 3
```

```
D1 = G / 3 - F * F
```

```
E = H - F * G + 2 * F * F * F
```

```
C = 4 * D1 * D1 * D1 + E * E
```

```
'C es el discriminante...raices reales e imaginarias
```

```
IF C > 0 THEN
```

```
C = SQR(C)
```

```
a = .5 * (C - E)
```

```
B = -.5 * (C + E)
```

```
C = 1 / 3
```

```
a = ABS(a) ^ C * SGN(a)
```

```
B = ABS(B) ^ C * SGN(B)
```

```
C = SQR(3) / 2
```

```
R1 = a + B - F
```

```
'R1 es una raiz real
```

```
'PRINT "R1=";R1
```

```
R2 = -.5 * (a + B) - F
```

```
'R2 es la parte real de las otras dos raices
```

```
I2 = C * ABS(a - B)
```

```
'I2 es la parte imag. de las otras dos raices
```

```
'PRINT "P.REAL=";R2;"P.IMAG=";I2
```

```
M23 = SQR(R2 ^ 2 + I2 ^ 2)
```

```
IF ABS(R1) <= 1 AND M23 <= 1 THEN
```

```
FASE$ = "SI"
```

```
'PRINT "FASE="; FASE$
```

```
ELSE
```

```
FASE$ = "NO"
```

```
END IF
```

```
ELSE
```

```
a = 2 * SQR(-D1)
```

```
X = E / (2 * D1 * SQR(-D1))
```

```
B = (-ATN(X / SQR(-X * X + 1)) + 1.5707633#) / 3
```

```
D1 = 1.5707963#
```

```
E = .5235987#
```

```
G = a * SIN(D1 - B)
```

```
H = -a * SIN(E + B)
```

```
I = -a * SIN(E - B)
```

```
G = G - F
```

```
H = H - F
```

```
I = I - F
```

```
'G,H,I tres raices reales
```

```
'PRINT G,H,I
```

```

IF ABS(G) <= 1 AND ABS(H) <= 1 AND ABS(I) <= 1 THEN
    FASE$ = "SI"
ELSE
    FASE$ = "NO"
END IF

END IF

ELSEIF B(2) <> 0 THEN 'Ecuacion de 2do. grado
    D1 = B(1) ^ 2 - 4 * B(0) * B(2)
    IF D1 >= 0 THEN

        R = -B(1) / (2 * B(0))
        R1 = R + SQR(D1) / (2 * B(0))
        R2 = R - SQR(D1) / (2 * B(0))
        'R1 y R2 son dos raices reales

        IF ABS(R1) <= 1 AND ABS(R2) <= 1 THEN
            FASE$ = "SI"
            PRINT "FASE="; FASE$
        ELSE
            FASE$ = "NO"
            PRINT "FASE="; FASE$
        END IF
    ELSE
        R = -B(1) / (2 * B(0))
        I = SQR(-D1) / (2 * B(0))
        'R e I son la parte real e imaginaria
        'PRINT "P.REAL="; R, "P.IMAG.="; I
        MD = SQR(R ^ 2 + I ^ 2)

        IF MD <= 1 THEN
            FASE$ = "SI"
            'PRINT "FASE="; FASE$
        ELSE
            FASE$ = "NO"
            'PRINT "FASE="; FASE$
        END IF
    END IF

ELSEIF B(1) <> 0 THEN 'Ecuacion de 1er grado
    R = -B(1) / B(0)
    IF ABS(R) <= 1 THEN
        FASE$ = "SI"
        'PRINT "FASE="; FASE$
    ELSE
        FASE$ = "NO"
        'PRINT "FASE="; FASE$
    END IF

END IF

'Chequeo de fase minima.....

INPUT a$

IF FASE$ = "NO" THEN
FOR I = 1 TO 1000: NEXT
CLS
SCREEN 3
LINE (0, 0)-(700, 320), , B

```

```
LINE (3, 3)-(697, 317), , B
LOCATE 4, 20
PLAY "E30040"
PRINT "††LA PLANTA NO ES DE FASE MINIMA††"
LOCATE 6, 20
PRINT "† DESEA INGRESAR OTRA PLANTA...<S> o <N> ?:";
OPTION8$ = INPUT†(1)

SELECT CASE OPTION8$

CASE "S", "s"
IF ACON$ = "MRAC" THEN
B(0) = 0: B(1) = 0: B(2) = 0: B(3) = 0
-----
CALL MODIFICACIONES
-----
ELSE
B(0) = 0: B(1) = 0: B(2) = 0: B(3) = 0
-----
CALL INGRESOPLANTA
-----
CALL FASEMIN
-----

END IF
CASE "N", "n"

IF PLAN$ = "† INGRESE LOS DATOS DE LA PLANTA †" THEN
-----
CALL CLINEAL
-----
ELSE
-----
CALL MRAC
-----

END IF

CASE ELSE

END SELECT

ELSE
END IF

END SUB
```

=====
 SUBPROGRAMA FASEMRAC
 =====

```

SUB FASEMRAC
ON ERROR GOTO OVER
IF B1(3) <> 0 THEN
  F = B1(1) / B1(0)
  G = B1(2) / B1(0)
  H = B1(3) / B1(0)
  F = F / 3
  D1 = G / 3 - F * F
  E = H - F * G + 2 * F * F * F
  C = 4 * D1 * D1 * D1 + E * E
  IF C > 0 THEN
    C = SQR(C)
    a = .5 * (C - E)
    B = -.5 * (C + E)
    a = ABS(a) ^ C * SGN(a)
    B = ABS(B) ^ C * SGN(B)
    C = SQR(3) / 2
    R1 = a + B - F
    R2 = -.5 * (a + B) - F
    I2 = C * ABS(a - B)
    'PRINT "R.REAL="; R1; PRINT "P.REAL="; R2, "IMAGINARIA="; I2
    M23 = SQR(R2 ^ 2 + I2 ^ 2)
    IF ABS(R1) <= 1 AND M23 <= 1 THEN
      FLAG1$ = "SI"
      'PRINT "FASE="; FLAG1$
    ELSE
      FLAG1$ = "NO"
      'PRINT "FASE="; FLAG1$
    END IF
  ELSE
    a = 2 * SQR(-D1)
    X = E / (2 * D1 * SQR(-D1))
    B = (-ATN(X / SQR(-X * X + 1)) + 1.5707633#) / 3
    D1 = 1.5707963#
    E = .5235987#
    G = a * SIN(D1 - B)
    H = -a * SIN(E + B)
    I = -a * SIN(E - B)
    G = G - F
    H = H * F
    I = I - F
    'PRINT "Tres raices reales"; G, H, I
    IF ABS(G) <= 1 AND ABS(H) <= 1 AND ABS(I) <= 1 THEN
      FLAG1$ = "SI"
      'PRINT "FASE="; FLAG1$
    ELSE
      FLAG1$ = "NO"
      'PRINT "FASE="; FLAG1$
    END IF
  END IF

```

```

ELSEIF B1(2) <> 0 THEN
D1 = B1(1) ^ 2 - 4 * B1(0) * B1(2)
IF D1 >= 0 THEN
    R = -B1(1) / (2 * B1(0))
    R1 = R + SQR(D1) / (2 * B1(0))
    R2 = R - SQR(D1) / (2 * B1(0))
    PRINT "DOS RAICES REALES"; R1, R2

    IF ABS(R1) <= 1 AND ABS(R2) <= 1 THEN
        FLAG1$ = "SI"
        PRINT "FASE="; FLAG1$
    ELSE
        FLAG1$ = "NO"
        PRINT "FASE="; FLAG1$
    END IF
ELSE
    R = -B1(1) / (2 * B1(0))
    I = SQR(-D1) / (2 * B1(0))
    PRINT "P.REAL="; R, "P.IMAG="; I
    MO = SQR(R ^ 2 + I ^ 2)

    IF MO <= 1 THEN
        FLAG1$ = "SI"
        PRINT "FASE="; FLAG1$
    ELSE
        FLAG1$ = "NO"
        PRINT "FASE="; FLAG1$
    END IF
END IF

ELSEIF B1(1) <> 0 THEN
    R = -B1(1) / B1(0)
    IF ABS(R) <= 1 THEN
        FLAG1$ = "SI"
        PRINT "FASE="; FLAG1$
    ELSE
        FLAG1$ = "NO"
        PRINT "FASE="; FLAG1$
    END IF
END IF

'Se entrega.....FLAG1$
END SUB

```

```
=====
SUBPROGRAMA FIN
=====
```

```
SUB FIN
ON ERROR GOTO OVER
CLS
'Transferencia a modo grafico
LINE (0, 0)-(700, 320), , B
LINE (3, 3)-(697, 317), , B
LINE (6, 6)-(694, 314), , B
LOCATE 12, 25
PRINT "  ** SIMULACION TERMINADA **  "
LOCATE 21, 5
PRINT " Presione : <S> para efectuar otra simulacion ...."
LOCATE 22, 5
PRINT "                <T> para terminar ....      "
FIN1$ = INPUT$(1)

SELECT CASE FIN1$
CASE "S", "s"
-----
RUN 1
-----
CASE "T", "t"
CLS
-----
RUN 2
-----
CASE ELSE
PLAY "E30D40"
-----
RUN 1
-----
END SELECT

END SUB
```

```
=====
SUBPROGRAMA FUNCIONTRANSF.
=====
```

```
SUB FUNCIONTRANSF.
CLS
'Transferencia a modo grafico
ON ERROR GOTO OVER
SCREEN 3
LINE (0, 0)-(700, 320), , B
LINE (3, 3)-(697, 317), , B
LINE (3, 158)-(697, 158)
LOCATE 4, 12
LINE (80, 37)-(620, 61), , BF
PRINT " * SECUENCIA DE REFERENCIA CON FUNCION DE TRANSFERENCIA * "

'Ingreso de valores para YM
```



```

'NR es el # de puntos de la secuencia de referencia
LOCATE 9, 8
PRINT "INGRESE EL # DE PUNTOS DE LA SECUENCIA DE REFERENCIA (max.500): ";
INPUT NR
'Dimensionamiento dinamico
DIM AUX1(NR + 10), UM(500)
IF NR > 500 THEN
  PLAY "E30D40"
  LOCATE 10, 8
  PRINT "Se asume 500 puntos..."
ELSE
END IF

LOCATE 13, 15
PRINT " LA SALIDA DE REFERENCIA SE OBTIENE MEDIANTE : "
LOCATE 18, 30
PRINT "YM(k) =      UM(k)"
LOCATE 17, 38
PRINT "L(q)"
LOCATE 19, 38
PRINT "K(q)"
LINE (330, 244)-(370, 244)
LOCATE 22, 65
PRINT "espere"

FOR IFSH = 1 TO 7
LOCATE 22, 72
PRINT " "
FOR JFSH = 1 TO 100: NEXT
LOCATE 22, 72
PRINT ":"
FOR JFSH = 1 TO 100: NEXT
NEXT IFSH

CLS
LINE (0, 0)-(700, 320), , B
LINE (3, 3)-(697, 317), , B
LOCATE 5, 14
PRINT "## INGRESE LOS COEFICIENTES DE LOS POLINOMIOS L y K ##"
'Proteccion iterativa
L0 = 0: L1 = 0: L2 = 0: L3 = 0: K1 = 0: K2 = 0: K3 = 0

LOCATE 9, 20
PRINT "POLINOMIO L(q) : "
LOCATE 9, 40
PRINT "L0="; : INPUT L0
LOCATE 10, 40
PRINT "L1="; : INPUT L1
LOCATE 11, 40
PRINT "L2="; : INPUT L2
LOCATE 12, 40
PRINT "L3="; : INPUT L3
LOCATE 16, 20
PRINT "POLINOMIO K(q) : "
LOCATE 16, 40
PRINT "K1="; : INPUT K1
LOCATE 17, 40
PRINT "K2="; : INPUT K2
LOCATE 18, 40
PRINT "K3="; : INPUT K3

'Ingreso de UM(k)

```

```

CLS
LINE (0, 0)-(700, 320), , B
LINE (3, 3)-(697, 317), , B
LOCATE 4, 22
PRINT "### UM(k) ES UNA FUNCION ESCALONADA ###"
LOCATE 6, 24
PRINT "EL VALOR INICIAL DE UM(k) ES : ";
INPUT U1
LOCATE 8, 20
PRINT "TIENE MODIFICACIONES UM(k) ?.....<S> o <N> : "
TAB4$ = INPUT$(1)

SELECT CASE TAB4$
CASE "S", "s"
LOCATE 11, 26
PRINT "CUANTAS MODIFICACIONES.....?"
PRINT
LOCATE 13, 26
PRINT "<U> ..... UNA"
LOCATE 14, 26
PRINT "<A> ..... ALGUNAS"
LOCATE 16, 26
PRINT "SELECCIONE <U> o <A> ..... : "
TAB5$ = INPUT$(1)

SELECT CASE TAB5$
CASE "A", "a"
'UM sufre algunas modificaciones
LOCATE 19, 24
PRINT "INDIQUE EL # DE MODIFICACIONES : ";
INPUT NM
'Dimensionamiento dinamico
DIM TMOD(NM + 10), VELUM(NM + 10), t(NM + 10)
CLS
LINE (0, 0)-(700, 90), , B
LINE (3, 3)-(697, 87), , B
LOCATE 4, 16

PRINT " EN QUE INSTANTES OCURREN LAS MODIFICACIONES..?"
VIEW PRINT 8 TO 13
FOR I = 1 TO NM
PRINT TAB(17); "La modificacion "; I; " ocurre al instante k= ";
INPUT TMOD(I)
NEXT I
CLS 2
PRINT
PRINT TAB(25); " ## NUEVOS VALORES DE UM(k) ##"
PRINT
VIEW PRINT 12 TO 17
FOR I = 1 TO NM
PRINT TAB(17); "El valor que toma UM(k) al instante "; TMOD(I); " es: ";
INPUT VELUM(I)
NEXT I

VIEW PRINT
LOCATE 24, 65: PRINT "espere...."

'Asignacion de UM(k)
FOR I = 1 TO NM
t(I) = TMOD(I) - 1
NEXT I
t(NM + 1) = NR

```

```

FOR I = 0 TO t(1)
UM(I) = U1
NEXT I
FOR J = 1 TO NM
FOR I = TMOD(J) TO t(J + 1)
UM(I) = VELUM(J)
NEXT I
NEXT J

```

```

CASE "U", "u"
LOCATE 19, 20
PRINT "‡ En que instante ocurre la modificacion...: ";
INPUT T1
LOCATE 21, 20
PRINT "‡ El valor de UM(k) al instante "; T1; " es: ";
INPUT V1

```

```

LOCATE 22, 67: PRINT "espere...."

```

```

'Asignacion de valores

```

```

FOR I = 0 TO (T1 - 1)
UM(I) = U1
NEXT I
FOR I = T1 TO NR.
UM(I) = V1
NEXT I

```

```

CASE ELSE
PLAY "E30D40"

```

```

-----
CALL FUNCIONTRANSF.
-----

```

```

END SELECT

```

```

CASE "N", "n"
'No existen modificaciones para UM(k)
LOCATE 22, 60: PRINT "espere : "
FOR I = 0 TO NR
UM(I) = U1
NEXT I

```

```

CASE ELSE
PLAY "E30D40"

```

```

-----
CALL FUNCIONTRANSF.
-----

```

```

END SELECT

```

```

'CALCULO DE YM(k) CON FUNCION DE TRANSFRENCIA

```

```

FOR I = 0 TO NR
AUX1(I) = UM(I)
NEXT I
FOR I = 0 TO NR
UM(I + 3) = AUX1(I)
NEXT I
FOR I = 3 TO (NR + 3)
A1 = K1 * YM(I - 1) + K2 * YM(I - 2) + K3 * YM(I - 3)
YM(I) = L0 * UM(I) + L1 * UM(I - 1) + L2 * UM(I - 2) + L3 * UM(I - 3) - A1
NEXT I

```

```
FOR I = 0 TO NR
  YN(I) = YN(I + 3)
NEXT I
```

```
'Impresion
VIEW PRINT
```

```
CLS
LINE (30, 30)-(670, 70), , B
LOCATE 4, 14
PRINT "¿Desea observar graficamente YN(k)...<S> o <N> ?";
OPTION$ = INPUT$(1)
SELECT CASE OPTION$
CASE "S", "s"
```

```
GRAF$ = "SECUENCIA DE REFERENCIA YN(k)"
FARB$ = "AX1"
```

```
-----
CALL GRAFYM
-----
```

```
CASE ELSE
END SELECT
CLS
GOTO 100
PRINT "LA SECUENCIA DE REFERENCIA ES :"
```

```
FOR I = 0 TO NR
  PRINT "YN(" ; I ; ")=" ; YN(I)
  INPUT a$
NEXT I
```

```
100 EX = 1
END SUB
```

```
=====
SUBPROGRAMA GENERACION
=====
```

```
SUB GENERACION
```

```
ON ERROR GOTO DVER
'UH(I) es la entrada de control para el modelo de referencia
'YN(I) es la secuencia de salida del modelo de referencia
CLS
'Transferencia a modo grafico
SCREEN 3
LINE (0, 0)-(700, 320), , B
LINE (3, 3)-(697, 317), , B
LINE (135, 50)-(573, 75), , BF
LOCATE 5, 18
PRINT " * GENERACION DEL MODELO DE REFERENCIA * "
LOCATE 11, 8
PRINT " * UH(k) ES LA ENTRADA DE CONTROL PARA EL MODELO DE REFERENCIA"
LOCATE 16, 8
PRINT " * YN(k) ES LA SECUENCIA DE SALIDA DEL MODELO DE REFERENCIA"
PRINT : PRINT : PRINT
```

```
FOR I = 200 TO 500 STEP 2
  PSET (I, 280)
NEXT I
LOCATE 22, 65
```

```
PRINT "espere"
```

```
FOR IFSH = 1 TO 7
LOCATE 22, 72
PRINT " "
FOR JFSH = 1 TO 100: NEXT
LOCATE 22, 72
PRINT ":"
FOR JFSH = 1 TO 100: NEXT
NEXT IFSH
```

```
CLS
```

```
LINE (0, 0)-(700, 320), , B
LINE (3, 3)-(697, 317), , B
LOCATE 3, 10
LINE (68, 51)-(623, 74), , BF
LOCATE 5, 10
PRINT "SELECCIONE COMO DESEA ESTABLECER LA SECUENCIA DE REFERENCIA"
LOCATE 7, 10
LOCATE 12, 19
LINE (152, 152)-(179, 169), , BF
PRINT "1"
LOCATE 12, 24
PRINT "PUNTO POR PUNTO"
LOCATE 16, 19
LINE (152, 208)-(179, 225), , BF
PRINT "2"
LOCATE 16, 24
PRINT "MEDIANTE FUNCION DE TRANSFERENCIA"
LOCATE 21, 24
PRINT "Seleccione una opcion :"
```

```
OPTION3$ = INPUT$(1)
```

```
'Selecion
```

```
SELECT CASE OPTION3$
```

```
CASE "1"
```

```
'Llamado a subprograma PUNTO.AP
```

```
-----
CALL PUNTO.AP
-----
```

```
CASE "2"
```

```
'Llamado a subprograma F.TRANSF.
```

```
-----
CALL FUNCIONTRANSF.
-----
```

```
CASE ELSE
```

```
PLAY "E30040"
```

```
'Autollamado a subprograma GENERACION
```

```
-----
CALL GENERACION
-----
```

```
END SELECT
```

```
'En este punto se dispone la secuencia YM(k)
```

```
'Presentacion grafica de YM(k)
```

```
'Llamado a subprograma GRAF.GEN
```

```
-----
CALL GRAF.GEN
```

```
END SUB
```

```
=====
SUBPROGRAMA GRAFYM
=====
```

```
SUB GRAFYM
```

```
ON ERROR GOTO HGR
SHARED YM(), NR
DIM GYM(610)
```

```
'Escalamiento
```

```
FOR I = 0 TO NR
SELECT CASE YM(I)
CASE -3.999999# TO 4.000001#
GYM(I + 100) = INT(25 * YM(I))

CASE 4.000001# TO 10.000001#
GYM(I + 100) = INT(100 + (YM(I) - 4) * 4.166666)

CASE -9.999999 TO -4#
GYM(I + 100) = INT(-100 + (YM(I) + 4) * 4.166666)

CASE 10.001 TO 100
GYM(I + 100) = INT(125 + (YM(I) - 10) * .277777)

CASE -100 TO -10.00001
GYM(I + 100) = INT(-125 + (YM(I) + 10) * .277777)

CASE IS > 100
GYM(I + 100) = 150

CASE IS < -100
GYM(I + 100) = -150

CASE ELSE
PLAY "E30040"
```

```
END SELECT
NEXT I
```

```
SELECT CASE NR
```

```
CASE IS <= 100
CLS
VIEW PRINT
```

```
'Impresion de numeros en ejes
```

```
LOCATE 2, 21: PRINT GRAF*
```

```
'LOCATE 2, 8: PRINT IMPGRAF*
```

```
LOCATE 13, 8
```

```
PRINT "0 10 20 30 40 50 60 70 80 90 100 [k]"
```

```
LOCATE 3, 3: PRINT "100"
```

```
FOR IJ = 81 TO 87
```

```

LINE (42, IJ)-(43, IJ)
NEXT IJ
PSET (43, 80)
LINE (41, 88)-(44, 88)
LINE (38, 85)-(44, 85)
PSET (41, 82): PSET (40, 83): PSET (39, 84)
LOCATE 5, 4: PRINT "10"
LOCATE 7, 5: PRINT "2"
LOCATE 12, 5: PRINT "0"
LOCATE 15, 4: PRINT "-2"

```

```

PSET (43, 240)

```

```

FOR IJ = 233 TO 239
LINE (42, IJ)-(43, IJ)
NEXT IJ
PSET (43, 232)
PSET (41, 234): PSET (40, 235): PSET (39, 236)
LINE (41, 240)-(44, 240)
LINE (38, 237)-(44, 237)
LINE (28, 237)-(35, 237)
LOCATE 19, 3: PRINT "-10"
LOCATE 21, 2: PRINT "-100"
LOCATE 23, 50: PRINT "Presione cualquier tecla para continuar...."

```

```

'Transferencia a modo grafico

```

```

SCREEN 3

```

```

VIEW (1, 1)-(660, 320), , 1
WINDOW (50, -200)-(650, 200)

```

```

'Ejes

```

```

LINE (100, 0)-(630, 0)
LINE (100, 170)-(100, -170)
FOR I = 150 TO 600 STEP 50

```

```

-----
LINE (I, 0)-(I, -4)
NEXT I
FOR I = 25 TO 150 STEP 25
LINE (100, I)-(103, I)
FOR J = 100 TO 630 STEP 5
PSET (J, I)
NEXT J
NEXT I

```

```

FOR I = -25 TO -150 STEP -25

```

```

-----
LINE (100, I)-(103, I)
FOR J = 100 TO 630 STEP 5
PSET (J, I)
NEXT J
NEXT I

```

```

'Graficacion

```

```

J = 100
PGE = 1
FOR I = 100 TO (NR + 100)
PSET (J, GYM(I))
LINE (J, GYM(I))-(J, GYM(I - 1))
LINE (J, GYM(I))-(J + 5, GYM(I))
J = J + 5

```


J = J + 1.666666666#

NEXT I

NR entre 300 y 500

CASE 301 TO 500

CLS

VIEW PRINT

LOCATE 2, 21: PRINT GRAF\$

LOCATE 13, 8

PRINT "0 50 100 150 200 250 300 350 400 450 500 [k]"

LOCATE 3, 3: PRINT "100": LOCATE 5, 4: PRINT "10"

FOR IJ = 81 TO 87

LINE (42, IJ)-(43, IJ)

NEXT IJ

PSET (43, 80)

LINE (41, 88)-(44, 88)

LINE (38, 85)-(44, 85)

PSET (41, 82): PSET (40, 83): PSET (39, 84)

LOCATE 9, 5: PRINT "2"

LOCATE 12, 5: PRINT "0": LOCATE 15, 4: PRINT "-2"

FOR IJ = 233 TO 239

LINE (42, IJ)-(43, IJ)

NEXT IJ

PSET (43, 232)

PSET (41, 234): PSET (40, 235): PSET (39, 236)

LINE (41, 240)-(44, 240)

LINE (38, 237)-(44, 237)

LINE (28, 237)-(35, 237)

LOCATE 19, 3: PRINT "-10"

LOCATE 21, 2: PRINT "-100"

LOCATE 23, 50

PRINT "Presione cualquier tecla para continuar"

SCREEN 3

VIEW (1, 1)-(660, 320), , 1

WINDOW (50, -200)-(650, 200)

LINE (100, 0)-(630, 0)

LINE (100, 170)-(100, -170)

FOR I = 150 TO 600 STEP 50

LINE (I, 0)-(I, -4)

NEXT I

FOR I = 25 TO 150 STEP 25

LINE (100, I)-(103, I)

FOR J = 160 TO 630 STEP 5

PSET (J, I)

NEXT J

NEXT I

FOR I = -25 TO -150 STEP -25

LINE (100, I)-(103, I)

FOR J = 100 TO 630 STEP 5

PSET (J, I)

NEXT J

NEXT I

Graficacion

J = 100

FOR I = 100 TO (NR + 100)

PSET (J, GYM(I))

```

LINE (J, GYM(I))-(J, GYM(I - 1))
LINE (J, GYM(I))-(J + 1, GYM(I))
J = J + 1
NEXT I

```

```

CASE ELSE
END SELECT
WINDOW
VIEW
CONT$ = INPUT$(1)

```

```

W1$ = "SECUENCIA DE REFERENCIA YM(k)"
IF GRAF$ = W1$ AND FARG$ = "AX1" THEN
CLS
LINE (30, 30)-(670, 70), , B
LOCATE 4, 14
PRINT " APRUEBA LA SECUENCIA DE REFERENCIA...<S> o <N> ?";
OPTION$ = INPUT$(1)

```

```

SELECT CASE OPTION$
CASE "S", "s"
FOR I = 1 TO 10: NEXT I
CASE "N", "n"

```

```

-----
NF$ = "NO"

```

```

CASE ELSE
FOR I = 1 TO 10: NEXT I
END SELECT
ELSE
FOR I = 1 TO 10: NEXT I
END IF
FARG$ = " "
END SUB

```

```

=====
SUBPROGRAMA IMPRESION
=====

```

```

SUB IMPRESION

```

```

'Control de ejecucion

```

```

ON ERROR GOTO INPR

```

```

LPRINT : LPRINT

```

```

'Codigo de condensacion

```

```

WIDTH "LPT1:", 255

```

```

LPRINT CHR$(27); CHR$(15)

```

```

'Codigo de marginacion

```

```

LPRINT CHR$(27); CHR$(108); CHR$(8)

```

```

'LPRINT CHR$(27); CHR$(69)

```

```

LPRINT TAB(48); " * RESULTADOS NUMERICOS * "

```

```

LPRINT

```

```

LPRINT STRING$(112, "-")

```

```

LPRINT TAB(16); "YM(k)"; TAB(58); "U(k)"; TAB(100); "Y(k)"

```

```

LPRINT STRING$(112, "-")

```

```

'LPRINT CHR$(27); CHR$(70)

```

```

LPRINT

```

```

FOR I = 0 TO NR
SELECT CASE I
CASE 0 TO 9
SX1 = 2
CASE 10 TO 99
SX1 = 1
CASE IS >= 100
SX1 = 0
END SELECT
LPRINT TAB(7); "YM("; I; ")"; SPC(SX1); "=";
LPRINT USING "#####.#####"; YM(I);
LPRINT TAB(49); "U("; I; ")"; SPC(SX1); "=";
LPRINT USING "#####.#####"; U(I);
LPRINT TAB(91); "Y("; I; ")"; SPC(SX1); "=";
LPRINT USING "#####.#####"; Y(I)
NEXT I
END SUB

```

```

=====
SUBPROGRAMA INGRESOPLANTA
=====

```

```

SUB INGRESOPLANTA
CLS
ON ERROR GOTO OVER
SCREEN 3
LINE (0, 0)-(700, 320), , B
LINE (3, 3)-(697, 317), , B
LINE (176, 37)-(507, 61), , BF

LOCATE 4, 22
PRINT PLAN$
LINE (3, 80)-(697, 80)
LOCATE 8, 16
PRINT "‡ EL GRADO DEL POLINOMIO A(q) ES (max.3).... :";

INPUT N

'Se guarda este dato para presentacion
BK1 = N

LOCATE 10, 16
PRINT "‡ EL GRADO DEL POLINOMIO B(q) ES (max.3).... :";
INPUT M

'Se guarda este dato para presentacion
BK2 = M

LOCATE 11, 30
LOCATE 13, 20
PRINT "POLINOMIO A(q) : "
'Proteccion iterativa
a(1) = 0: a(2) = 0: a(3) = 0
B(0) = 0: B(1) = 0: B(2) = 0: B(3) = 0
SELECT CASE N
CASE 1
LOCATE 13, 40
PRINT "a1="; ; INPUT a(1)
CASE 2

```

```

LOCATE 13, 40
PRINT "a1="; : INPUT a(1)
LOCATE 14, 40
PRINT "a2="; : INPUT a(2)
CASE 3
LOCATE 13, 40
PRINT "a1="; : INPUT a(1)
LOCATE 14, 40
PRINT "a2="; : INPUT a(2)
LOCATE 15, 40
PRINT "a3="; : INPUT a(3)

```

```

CASE ELSE
PLAY "E30D40"

```

```

-----
CALL INGRESOPLANTA
-----

```

```

END SELECT

```

```

'Se guarda datos de A() para la presentacion
FOR I = 1 TO N
G1(I) = a(I)
NEXT I

```

```

LOCATE 17, 20
PRINT "POLINOMIO B(q) :"
SELECT CASE M
CASE 0
LOCATE 17, 40
PRINT "b0="; : INPUT B(0)
CASE 1
LOCATE 17, 40
PRINT "b0="; : INPUT B(0)
LOCATE 18, 40
PRINT "b1="; : INPUT B(1)
CASE 2
LOCATE 17, 40
PRINT "b0="; : INPUT B(0)
LOCATE 18, 40
PRINT "b1="; : INPUT B(1)
LOCATE 19, 40
PRINT "b2="; : INPUT B(2)
CASE 3
LOCATE 17, 40
PRINT "b0="; : INPUT B(0)
LOCATE 18, 40
PRINT "b1="; : INPUT B(1)
LOCATE 19, 40
PRINT "b2="; : INPUT B(2)
LOCATE 20, 40
PRINT "b3="; : INPUT B(3)
CASE ELSE
END SELECT

```

```

'Se guarda datos para presentacion
FOR I = 0 TO M
B2(I) = B(I)
NEXT I

```

```

END SUB

```

```
=====
SUBPROGRAMA MENU PRINCIPAL : MENU.PRIN
=====
```

```
SUB MENU.PRIN
```

```
SHARED OPTION1$
```

```
CLS
```

```
ON ERROR GOTO OVER
```

```
'Transferencia a modo grafico
```

```
SCREEN 3
```

```
'Diseño de marco externo
```

```
STYLE% = &HFFFF
```

```
STYLE1% = &HFFFF
```

```
LINE (1, 1)-(650, 1), , , STYLE%
```

```
LINE (650, 1)-(650, 320), , , STYLE%
```

```
LINE (1, 320)-(650, 320), , , STYLE%
```

```
LINE (1, 1)-(1, 320), , , STYLE%
```

```
LINE (4, 4)-(647, 4), , , STYLE1%
```

```
LINE (647, 4)-(647, 317), , , STYLE1%
```

```
LINE (4, 317)-(647, 317), , , STYLE1%
```

```
LINE (4, 4)-(4, 317), , , STYLE1%
```

```
LINE (7, 7)-(644, 314), , B
```

```
'Leyendas
```

```
LOCATE 3, 25
```

```
LINE (180, 20)-(465, 47), , BF
```

```
PRINT "**** MENU PRINCIPAL ****"
```

```
LOCATE 5, 27
```

```
PRINT "....."
```

```
LOCATE 8, 19
```

```
LINE (152, 96)-(179, 113), , BF
```

```
PRINT "1"
```

```
LOCATE 8, 24
```

```
PRINT "SIMULACION DE CONTROL LINEAL"
```

```
LOCATE 12, 19
```

```
LINE (152, 152)-(179, 169), , BF
```

```
PRINT "2"
```

```
LOCATE 12, 24
```

```
PRINT "SIMULACION DE CONTROL ADAPTIVO MRAC"
```

```
LOCATE 16, 19
```

```
LINE (152, 208)-(179, 225), , BF
```

```
PRINT "3"
```

```
LOCATE 16, 24
```

```
PRINT "TERMINAR"
```

```
'Selección de una opción
```

```
LOCATE 20, 25
```

```
PRINT "Selecione un # : ";
```

```
OPTION1$ = INPUT$(1)
```

```
'Selección
```

```
'OPTION1$ = 1...Simulación de Control Lineal
```

```

'OPTION1$ = 2...Simulacion de Control Adaptivo MRAC
'OPTION1$ = 3...Terminar
SELECT CASE OPTION1$
CASE "1"
TIPO$ = "CLINEAL" 'Bandera de referencia posterior
'Llamado a subprograma de Control Lineal
-----
CALL CLINEAL
-----
CASE "2"
TIPO$ = "MRAC" 'Bandera de referencia posterior
'Llamado a subprograma de Control Adaptivo MRAC
-----
CALL MRAC
-----
CASE "3"
'Llamado a subprograma de finalizacion
-----
CALL FIN
-----
CASE ELSE
PLAY "E30D40"
'Autollamado a subprograma Menu Principal
-----
CALL MENU.PRIN
-----
END SELECT
END SUB

```

```

=====
SUBPROGRAMA MODIFICACIONES
=====

```

```

SUB MODIFICACIONES
CLS
ON ERROR GOTO OVER
ACON$ = "MRAC"
'Transferencia a modo grafico
SCREEN 3
LINE (0, 0)-(700, 320), , B
LINE (3, 3)-(697, 317), , B
LOCATE 4, 21
LINE (130, 37)-(565, 61), , BF
PRINT " * ALTERACIONES EN LA PLANTA * "
LINE (3, 90)-(697, 90)
LOCATE 12, 19
LINE (152, 152)-(179, 169), , BF
PRINT "A"

```

```

LOCATE 12, 24
PRINT "GRADOS DE A(q) Y B(q) SIN MODIFICACIONES"
LOCATE 16, 19
LINE (152, 208)-(179, 225), , BF
PRINT "B"
LOCATE 16, 24
PRINT "GRADOS DE A(q) y/o B(q) CON MODIFICACIONES"
LOCATE 21, 24
PRINT "Seleccione una opcion : "
OPTION$ = INPUT$(1)

'Seleccion
SELECT CASE OPTION$
CASE "A", "a" 'No se modifican N y M
30  CLS
    LINE (0, 0)-(700, 320), , B
    LINE (3, 3)-(697, 317), , B
    LOCATE 4, 24
    LINE (130, 37)-(565, 61), , BF
    PRINT "* DATOS DE LA NUEVA PLANTA *"

    N.1 = N: M.1 = M

'Se guardan estos valores para presentacion
    GK3 = N.1
    GK4 = M.1

'Proteccion iterativa
    A1(1) = 0: A1(2) = 0: A1(3) = 0
    B1(0) = 0: B1(1) = 0: B1(2) = 0: B1(3) = 0

    FOR I = 1 TO N
        LOCATE (6 + I), 34
        PRINT "a"; I; "="; : INPUT A1(I)
    'Resguardo
        B3(I) = A1(I)
    NEXT I

    FOR I = 0 TO M
        LOCATE (13 + I), 34
        PRINT "b"; I; "="; : INPUT B1(I)
    'Resguardo
        B4(I) = B1(I)
    NEXT I

'Resguardo temporal de B(I) de la planta inicial
    FOR I = 0 TO M
        CAMBIO1(I) = B(I)
    NEXT I

    ACON$ = "MRAC"

-----
    CALL FASEMRAC
-----

'Se recibe...FLAG1$
IF FLAG1$ = "NO" THEN 'Proceso cuando NOFASEMIN
CLS
LINE (0, 0)-(700, 320), , B
LINE (3, 3)-(697, 317), , B
LOCATE 4, 10

```

```

PRINT "LA NUEVA PLANTA INGRESADA NO ES DE FASE MINIMA"
LOCATE 6, 10 :
PRINT "? DESEA INGRESAR OTRA PLANTA ?....<S> o <N> :";
OPCION1$ = INPUT$(1)

```

```

SELECT CASE OPCION1$

```

```

CASE "S", "s"

```

```

GOTO 30

```

```

CASE "N", "n"

```

```

-----
CALL FIN
-----

```

```

CASE ELSE

```

```

PLAY "E30D40"

```

```

GOTO 30

```

```

END SELECT

```

```

ELSE

```

```

END IF

```

```

CASE "B", "b" 'Se modifican N y/o M

```

```

CLS

```

```

LINE (0, 0)-(700, 320), , B

```

```

LINE (3, 3)-(697, 317), , B

```

```

LINE (176, 37)-(507, 61), , BF

```

```

LOCATE 4, 22

```

```

PRINT " * DATOS DE LA NUEVA PLANTA * "

```

```

LINE (3, 80)-(697, 80)

```

```

'Proteccion iterativa

```

```

A1(1) = 0: A1(2) = 0: A1(3) = 0

```

```

B1(0) = 0: B1(1) = 0: B1(2) = 0: B1(3) = 0

```

```

LOCATE 8, 16

```

```

PRINT " * EL GRADO DEL POLINOMIO A(q) ES (max.3).... :";

```

```

INPUT N.1

```

```

'Resguardo

```

```

GK3 = N.1

```

```

LOCATE 10, 16

```

```

PRINT " * EL GRADO DEL POLINOMIO B(q) ES (max.3).... :";

```

```

INPUT M.1

```

```

'Resguardo

```

```

GK4 = M.1

```

```

IF N.1 > 3 OR M.1 > 3 THEN

```

```

GOTO 40

```

```

ELSE

```

```

END IF

```

```

FOR I = 1 TO N.1

```

```

LOCATE (I + 12), 40

```

```

PRINT "a"; I; "="; : INPUT A1(I)

```

```

'Resguardo

```

```

G3(I) = A1(I)

```

```

NEXT I

```

```

FOR I = 0 TO M.1

```

```

LOCATE (I + 16), 40

```

```

PRINT "b"; I; "="; : INPUT B1(I)

```

```

'Resguardo

```

```

G4(I) = B1(I)

```

```

NEXT I

```

```

ACON$ = "MRAC"

```



```
-----
CALL FASEMRAC
-----
```

```
'Se recibe...FLAG1$
IF FLAG1$ = "NO" THEN
CLS
LINE (0, 0)-(700, 320), , B
LINE (3, 3)-(697, 317), , B
LOCATE 4, 10
PRINT "LA NUEVA PLANTA INGRESADA NO ES DE FASE MINIMA"
LOCATE 6, 10
PRINT "? DESEA INGRESAR UNA NUEVA PLANTA ?...<S> o <N>:";
OPCION2$ = INPUT$(1)
```

```
SELECT CASE OPCION2$
```

```
  CASE "S", "s"
```

```
    GOTO 40
```

```
  CASE "N", "n"
```

```
    -----
    'CALL FIN
    -----
```

```
  CASE ELSE
```

```
    PLAY "E30D40"
```

```
    GOTO 40
```

```
  END SELECT
```

```
ELSE
```

```
END IF
```

```
CASE ELSE
```

```
  PLAY "E30D40"
```

```
  -----
  CALL MODIFICACIONES
  -----
```

```
END SELECT
```

```
'Ingreso del tiempo al cual se modifica la planta
```

```
120 :
```

```
CLS
```

```
LINE (0, 0)-(700, 320), , B
```

```
LINE (3, 3)-(697, 317), , B
```

```
LOCATE 10, 15
```

```
LOCATE 12, 15
```

```
PRINT " LA PLANTA SE MODIFICA AL INSTANTE (max.; NR; ") k="; :
```

```
INPUT TEMP1
```

```
IF TEMP1 > NR THEN
```

```
  GOTO 120
```

```
ELSE
```

```
END IF
```

```
END SUB
```

```
=====
SUBPROGRAMA MRAC
=====
```

```
SUB MRAC
```

```
'Presentacion
```

```
CLS
```

```
ON ERROR GOTO OVER
```

```
'Transferencia a modo grafico
```

```
SCREEN 3
```

```
LINE (0, 0)-(650, 320), , B
```

```
LINE (40, 40)-(610, 280), , B
```

```
LINE (43, 43)-(607, 277), , B
```

```
LINE (46, 46)-(604, 274), , BF
```

```
'Leyendas
```

```
LOCATE 10, 13
```

```
PRINT " * P R O G R A M A   D E   S I M U L A C I O N * "
```

```
LOCATE 14, 13
```

```
PRINT " * C O N T R O L   A D A P T I V O   M.R.A.C. * "
```

```
LOCATE 22, 5
```

```
PRINT "Presione [C] para cont.,[Q] para regresar al menu principal : ";
```

```
OPTION21$ = INPUT$(1)
```

```
'Seleccion
```

```
SELECT CASE OPTION21$
```

```
CASE "Q", "q"
```

```
'Llamado a MENU.PRIN
```

```
CALL MENU.PRIN
```

```
CASE "C", "c"
```

```
'Llamado a GENERACION de Modelo de Referencia
```

```
CALL GENERACION
```

```
CASE ELSE
```

```
'Autollamado a MRAC
```

```
PLAY "E30D40"
```

```
CALL MRAC
```

```
END SELECT
```

```
'Vuelve el control a MRAC
```

```
'Ingreso de los datos de la planta inicial
```

```
PLAN$ = " * D A T O S   D E   L A   P L A N T A   I N I C I A L * "
```

```
CALL INGRESOPLANTA
```

```
CALL FASEMIN
```

```
-----
CALL MODIFICACIONES
-----
```

```
-----
CALL RETARDO.MRAC
-----
```

```
-----
CALL RESULTADOS.1
-----
```

```
-----
CALL FIN
-----
```

```
END SUB
```

```
=====
SUBPROGRAMA DE PRESENTACION DE CARATULA
=====
```

```
SUB PRESENTACION
```

```
CLS
```

```
'Transferencia a modo grafico
```

```
SCREEN 3
```

```
PLAY "E30040": FOR I = 1 TO 1000: NEXT I:
FOR I = 10 TO 650: PSET (I, 1): NEXT
FOR I = 1 TO 311: PSET (650, I): NEXT
FOR I = 7 TO 647: PSET (I, 4): NEXT
FOR I = 4 TO 314: PSET (647, I): NEXT
FOR I = 4 TO 644: PSET (I, 7): NEXT
FOR I = 7 TO 317: PSET (644, I): NEXT
FOR I = 1 TO 641: PSET (I, 10): NEXT
FOR I = 10 TO 320: PSET (641, I): NEXT
FOR I = 641 TO 1 STEP -1: PSET (I, 320): NEXT
FOR I = 320 TO 10 STEP -1: PSET (I, 1): NEXT
FOR I = 6 TO 636 STEP 2: PSET (I, 15): NEXT
FOR I = 15 TO 315 STEP 2: PSET (636, I): NEXT
FOR I = 636 TO 6 STEP -2: PSET (I, 315): NEXT
FOR I = 315 TO 15 STEP -2: PSET (6, 1): NEXT
```

```
LINE (10, 19)-(632, 311), , BF
```

```
'Leyendas
```

```
LOCATE 6, 21'
```

```
PRINT "ESCUELA POLITECNICA NACIONAL"
```

```
LOCATE 10, 19
```

```
PRINT "FACULTAD DE INGENIERIA ELECTRICA"
```

```
LOCATE 15, 14
```

```
PRINT "CONTROL ADAPTIVO CON MODELO DE REFERENCIA"
```

```
LOCATE 17, 24
```

```
PRINT "PARA SISTEMAS DISCRETOS"
```

```
LOCATE 21, 4
```

```
PRINT "AUTOR: H. Ortiz T."
```

```
LOCATE 21, 43
```

```
PRINT "DIRECTOR: Ing. M. Barragan"
```

```
LOCATE 22, 26
```

```

FOR I = 1 TO 10000
IF INKEY$ <> "" THEN EXIT FOR
NEXT I

END SUB

```

```

=====
PROGRAMA.A
=====

```

```

SUB PROGRAMA.A

```

```

'Variables compartidas

```

```

SHARED N, M, NR, a(), B(), YM()

```

```

OVER$ = " "

```

```

'Ingreso del polinomio de control Cr(q)

```

```

'Los coeficientes de Cr(q) estan en C1()

```

```

DIM E(10), AUX3(510)

```

```

CLS

```

```

'Transferencia a modo grafico

```

```

SCREEN 3

```

```

LINE (0, 0)-(700, 320), , B

```

```

LINE (3, 3)-(697, 317), , B

```

```

LOCATE 4, 17

```

```

LINE (80, 37)-(620, 61), , BF

```

```

PRINT "* INGRESO DEL POLINOMIO Cr(q) DE CONTROL *"

```

```

LINE (3, 158)-(697, 158)

```

```

LOCATE 9, 8

```

```

PRINT "EL GRADO DEL POLINOMIO Cr(q) ES (max.; N; ").....:";

```

```

INPUT GC

```

```

IF GC > N THEN

```

```

PLAY "E30D40"

```

```

'Se asume que GC es igual a N-1

```

```

GC = N

```

```

ELSE

```

```

END IF

```

```

SELECT CASE GC

```

```

CASE 0

```

```

LOCATE 15, 35

```

```

PRINT "C(q)=1"

```

```

C1(1) = 0: C1(2) = 0: C1(3) = 0

```

```

CASE 1

```

```

LOCATE 15, 40

```

```

PRINT "c1="; : INPUT C1(1)

```

```

C1(2) = 0: C1(3) = 0

```

```

CASE 2

```

```

LOCATE 15, 40

```

```

PRINT "c1="; : INPUT C1(1)

```

```

LOCATE 16, 40

```

```

PRINT "c2="; : INPUT C1(2)

```

```

C1(3) = 0

```

```

CASE 3

```

```

LOCATE 15, 40
PRINT "c1="; : INPUT C1(1)
LOCATE 16, 40
PRINT "c2="; : INPUT C1(2)
LOCATE 17, 40
PRINT "c3="; : INPUT C1(3)

CASE ELSE
PLAY "E30D40"
-----
CALL PROGRAMA.A
-----

END SELECT

'Calculos del subprograma

FOR I = 0 TO NR
AUX3(I) = YM(I)
NEXT I
FOR I = 0 TO NR
YM(I + 6) = AUX3(I)
NEXT I

FOR I = 1 TO (M + 1)
E(I) = B(I - 1)
NEXT I
J = 1
FOR I = (M + 2) TO (M + N + 1)
E(I) = C1(J) - a(J)
J = J + 1
NEXT I

'Comienza el lazo principal

CLS
LOCATE 1, 10
PRINT "Procesando ....."
LOCATE 3, 8
PRINT STRING$(65, " ")
LOCATE 24, 8
PRINT STRING$(65, " ")
VIEW PRINT 5 TO 22
'Comienza el lazo principal

FOR I = 0 TO NR

DN ERROR GOTO OVER
IF OVER$ = "K" OR OVER$ = "k" THEN EXIT FOR

HB1 = B(0) * U(I + 5) + B(1) * U(I + 4) + B(2) * U(I + 3) + B(3) * U(I + 2)
Y(I + 6) = -a(1) * Y(I + 5) - a(2) * Y(I + 4) - a(3) * Y(I + 3) + HB1
X1 = 0
L1 = I
FOR K = 2 TO (M + 1)
X1 = X1 + E(K) * U(L1 + 5)
L1 = L1 - 1
NEXT K
X2 = 0
L2 = I
FOR K = (M + 2) TO (M + N + 2)
X2 = X2 + E(K) * Y(L2 + 6)

```

```

L2 = L2 - 1
NEXT K
X3 = YM(I + 6) + C1(1) * YM(I + 6) + C1(2) * YM(I + 5) + C1(3) * YM(I + 4)
U(I + 6) = (1 / E(1)) * (X3 - X2 - X1)

SELECT CASE I
CASE 0 TO 9
SX1 = 2
CASE 10 TO 99
SX1 = 1
CASE IS >= 100
SX1 = 0
END SELECT

PRINT TAB(10); "U("; I; ")"; SPC(SX1); "=";
PRINT USING "#####.#####"; U(I + 6);
PRINT TAB(50); "Y("; I; ")"; SPC(SX1); "=";
PRINT USING "#####.#####"; Y(I + 6)
.....
IF I >= 20 THEN
A(1) = -.9; A(2) = .08; B(0) = 1; B(1) = -.8
ELSE
END IF
.....
NEXT I

VIEW PRINT
'Regreso en el tiempo 6 pasos
FOR I = 0 TO NR
Y(I) = Y(I + 6)
U(I) = U(I + 6)
YM(I) = YM(I + 6)
NEXT I

END SUB

```

```

=====
PROGRAMA.B
=====

```

```
SUB PROGRAMA.B
```

```

'Programa para C.Lineal con retardo d=2
'Variables compartidas
SHARED N, M, NR, a(), B(), YM()
'PRINT "N="; N; "M="; M; "NR="; NR: INPUT K3

```

```
OVER$ = " "
```

```

'Ingreso del polinomio de control Cr(q)
'Los coeficientes de Cr(q) estan en C1()
DIM E(8), AUX3(510)

```

```
CLS
```

```
'Transferencia a modo grafico
```

```
SCREEN 3
```

```
LINE (0, 0)-(700, 320), , B
```

```
LINE (3, 3)-(697, 317), , B
```

```
LOCATE 4, 17
```

```
LINE (80, 37)-(620, 61), , BF
```

```
PRINT "* INGRESO DEL POLINOMIO Cr(q) DE CONTROL *"
```

```

LINE (3, 158)-(697, 158)
LOCATE 9, 8
PRINT "EL GRADO DEL POLINOMIO Cr(q) ES (max."; N; ").....";
INPUT GC
IF GC > N THEN
PLAY "E30040"
GC = N
ELSE
END IF

```

```

SELECT CASE GC

```

```

CASE 0

```

```

LOCATE 15, 35
PRINT "C(q)=1"
C1(1) = 0: C1(2) = 0: C1(3) = 0

```

```

CASE 1

```

```

LOCATE 15, 40
PRINT "c1="; : INPUT C1(1)
C1(2) = 0: C1(3) = 0

```

```

CASE 2

```

```

LOCATE 15, 40
PRINT "c1="; : INPUT C1(1)
LOCATE 16, 40
PRINT "c2="; : INPUT C1(2)
C1(3) = 0

```

```

CASE 3

```

```

LOCATE 15, 40
PRINT "c1="; : INPUT C1(1)
LOCATE 16, 40
PRINT "c2="; : INPUT C1(2)
LOCATE 17, 40
PRINT "c3="; : INPUT C1(3)

```

```

CASE ELSE

```

```

PLAY "E30040"

```

```

-----
CALL PROGRAMA.B
-----

```

```

END SELECT

```

```

'Calculos del subprograma

```

```

FOR I = 0 TO NR
AUX3(I) = YH(I)
NEXT I
FOR I = 0 TO NR
YH(I + 6) = AUX3(I)
NEXT I

```

```

'SP1 representa a s1

```

```

SP1 = C1(1) - a(1)
E(1) = B(0)
FOR I = 2 TO (M + 2)
E(I) = B(I - 1) + B(I - 2) * SP1
NEXT I

```

```

J = 2
FOR I = (M + 3) TO (M + N + 2)
E(I) = C1(J) - a(J) - a(J - 1) * SP1
J = J + 1
NEXT I

```

```

CLS
LOCATE 1, 10
PRINT "Procesando ....."
LOCATE 3, 8
PRINT STRING$(65, " ")
LOCATE 24, 8
PRINT STRING$(65, " ")
VIEW PRINT 5 TO 22

```

Comienza el lazo principal

```

FOR I = 0 TO NR
  ON ERROR GOTO OVER
  IF OVER$ = "K" OR OVER$ = "k" THEN EXIT FOR

  HB1 = B(0) * U(I + 4) + B(1) * U(I + 3) + B(2) * U(I + 2) + B(3) * U(I + 1)
  Y(I + 6) = -a(1) * Y(I + 5) - a(2) * Y(I + 4) - a(3) * Y(I + 3) + HB1
  X1 = 0
  L1 = I
  FOR K = 2 TO (M + 2)
    X1 = X1 + E(K) * U(L1 + 5)
    L1 = L1 - 1
  NEXT K
  X2 = 0
  L2 = I
  FOR K = (M + 3) TO (M + N + 2)
    X2 = X2 + E(K) * Y(L2 + 6)
    L2 = L2 - 1
  NEXT K
  X3 = YM(I + 6) + C1(1) * YM(I + 6) + C1(2) * YM(I + 5) + C1(3) * YM(I + 4)

  U(I + 6) = (1 / E(1)) * (X3 - X2 - X1)

  SELECT CASE I
  CASE 0 TO 9
    SX1 = 2
  CASE 10 TO 99
    SX1 = 1
  CASE IS >= 100
    SX1 = 0
  END SELECT

  PRINT TAB(10); "U("; I; ")"; SPC(SX1); "=";
  PRINT USING "#####.#####"; U(I + 6);
  PRINT TAB(50); "Y("; I; ")"; SPC(SX1); "=";
  PRINT USING "#####.#####"; Y(I + 6)

```

```

NEXT I

```

```

VIEW PRINT

```

Regreso en el tiempo 6 pasos

```

FOR I = 0 TO NR
Y(I) = Y(I + 6)
U(I) = U(I + 6)
YM(I) = YM(I + 6)

```


END SUB

```

=====
PROGRAMA.ADA1
=====

```

SUB PROGRAMA.ADA1

'Comparte variables:N,M,N.1,M.1,NR,A(),B(),A1(),B(1)

OVER# = " "

'Ingreso del polinomio de control Cr(q)

'Los coeficientes de Cr(q) estan en C(1)

DIM AUX4(NR + 10), M(10)

DIM E(10) AS DOUBLE

DIM F(10, 10) AS DOUBLE, G(10, 10) AS DOUBLE, CM(10, 10) AS DOUBLE

CLS

'Transferencia a modo grafico

SCREEN 3

LINE (0, 0)-(700, 320), , B

LINE (3, 3)-(697, 317), , B

LOCATE 4, 17

LINE (80, 37)-(620, 61), , BF

PRINT "¡ INGRESO DEL POLINOMIO Cr(q) DE CONTROL !"

LINE (3, 158)-(697, 158)

LOCATE 9, 8

PRINT "EL GRADO DEL POLINOMIO Cr(q) ES (max."; N; ").....";

INPUT GC

IF GC > N THEN

PLAY "E30D40"

'Se asume que CG es igual a N

GC = N

ELSE

END IF

SELECT CASE GC

CASE 0

LOCATE 15, 35

PRINT "Cr(q)=1"

C1(1) = 0: C1(2) = 0: C1(3) = 0

CASE 1

LOCATE 15, 40

PRINT "c1="; : INPUT C1(1)

C1(2) = 0: C1(3) = 0

CASE 2

LOCATE 15, 40

PRINT "c1="; : INPUT C1(1)

LOCATE 16, 40

PRINT "c2="; : INPUT C1(2)

C1(3) = 0

CASE 3

```

LOCATE 15, 40
PRINT "c1="; : INPUT C1(1)
LOCATE 16, 40
PRINT "c2="; : INPUT C1(2)
LOCATE 17, 40
PRINT "c3="; : INPUT C1(3)

```

```

CASE ELSE
PLAY "E30040"

```

```

-----
CALL PROGRAMA.ADA1
-----

```

```

END SELECT

```

```

'Calculos del subprograma

```

```

'Desplazamiento de YM()

```

```

FOR I = 0 TO NR
AUX4(1) = YM(I)
NEXT I
FOR I = 0 TO NR
YM(I + 6) = AUX4(I)
NEXT I

```

```

LOCATE 19, 3
PRINT STRING$(74, "-")
LOCATE 21, 5
PRINT "INGRESE EL VALDR DE LA DIAGONAL DE F[I,J] :";
INPUT DIAG

```

```

.....
LOCATE 21, 5
PRINT "SE ASUME CONOCIDA LA PLANTA INICIAL ?...<S>/<N> :";
OPTIONP$ = INPUT$(1)
SELECT CASE OPTIONP$
CASE "N", "n"
PAR1$ = "E51"
CASE ELSE
PAR1$ = "NES1"
END SELECT

```

```

G = N + M + 1

```

```

'Proteccion iterativa

```

```

FOR I = 1 TO G
FOR J = 1 TO G
CM(I, J) = 0
NEXT J
NEXT I

```

```

FOR I = 1 TO G
CM(I, I) = DIAG
NEXT I

```

```

TRK = 0
FOR I = 1 TO G
TRK = TRK + CM(I, I)
NEXT I

```

```

IF PAR1$ = "NES1" THEN
FOR I = 1 TO (M + 1)
E(I) = B(I - 1)

```

```

NEXT I
J = 1
FOR I = (M + 2) TO 6
E(I) = C1(J) - a(J)
J = J + 1
NEXT I
ELSE
E(I) = 1: FOR I = 2 TO 6: E(I) = 0: NEXT I
END IF
FLAG$ = " "

CLS
LOCATE 1, 10
PRINT "Procesando ....."
LOCATE 3, 8
PRINT STRING$(65, " ")
LOCATE 24, 8
PRINT STRING$(65, " ")
VIEW PRINT 5 TO 22

'Comienza el lazo principal
z = 0
500 I = z
ON ERROR GOTO OVER
HB1 = B(0) * U(I + 5) + B(1) * U(I + 4) + B(2) * U(I + 3) + B(3) * U(I + 2)
Y(I + 6) = -a(1) * Y(I + 5) - a(2) * Y(I + 4) - a(3) * Y(I + 3) + HB1
IF OVER$ = "K" OR OVER$ = "k" THEN GOTO 200
X1 = 0
L1 = I
FOR K = 2 TO (M + 1)
X1 = X1 + E(K) * U(L1 + 5)
IF OVER$ = "K" OR OVER$ = "k" THEN GOTO 200
L1 = L1 - 1
NEXT K
X2 = 0
L2 = I
FOR K = (M + 2) TO (M + N + 2)
X2 = X2 + E(K) * Y(L2 + 6)
IF OVER$ = "K" OR OVER$ = "k" THEN GOTO 200
L2 = L2 - 1
NEXT K
X3 = YM(I + 6) + C1(1) * YM(I + 6) + C1(2) * YM(I + 5) + C1(3) * YM(I + 4)
IF OVER$ = "K" OR OVER$ = "k" THEN GOTO 200
U(I + 6) = (1 / E(I)) * (X3 - X2 - X1)
IF OVER$ = "K" OR OVER$ = "k" THEN GOTO 200
SELECT CASE I
CASE 0 TO 9
SX1 = 2
CASE 10 TO 99
SX1 = 1
CASE IS >= 100
SX1 = 0
END SELECT

```

```

PRINT TAB(10); "U("; I; ")"; SPC(SX1); "=";
PRINT USING "#####"; U(I + 6);
PRINT TAB(50); "Y("; I; ")"; SPC(SX1); "=";
PRINT USING "#####"; Y(I + 6)

```

'Asignacion del vector M()

```

FOR K = 1 TO (M + 1)
M(K) = U(z - K + 6)
NEXT K
L3 = 1
FOR K = (M + 2) TO (M + N + 2)
M(K) = Y(z - L3 + 6)
L3 = L3 + 1
NEXT K

```

'P.Id.

```

S4 = 0
S5 = 0
FOR I = 1 TO 6
FOR J = 1 TO 6
S4 = S4 + M(J) * CM(J, I)
NEXT J
S5 = S5 + M(I) * S4
S4 = 0
NEXT I
S3 = 1 + S5
'Aqui se obtuvo S3
S6 = 0
FOR I = 1 TO 6
FOR J = 1 TO 6
S6 = S6 + CM(I, J) * M(J)
NEXT J
D(I) = S6
S6 = 0
NEXT I
FOR I = 1 TO 6
FOR J = 1 TO 6
F(I, J) = D(I) * M(J)
NEXT J
NEXT I
FOR I = 1 TO 6
FOR J = 1 TO 6
G(I, J) = 0
NEXT J
NEXT I
FOR I = 1 TO 6
FOR J = 1 TO 6
FOR K = 1 TO 6
G(I, J) = G(I, J) + F(I, K) * CM(K, J)
NEXT K
NEXT J
NEXT I
FOR I = 1 TO 6
FOR J = 1 TO 6
CM(I, J) = CM(I, J) - G(I, J) / S3
NEXT J
NEXT I
SUMATRAZA = 0

```

```

FOR I = 1 TO G
SUMATRAZA = SUMATRAZA + CM(I, I)
NEXT I
FO = SUMATRAZA / TRK      'Factor de olvido
FOR I = 1 TO G
FOR J = 1 TO G
CM(I, J) = CM(I, J) / FO 'Es la nueva [CM(I,J)]
NEXT J
NEXT I

S7 = 0
FOR I = 1 TO G
S7 = S7 + E(I) * M(I)
NEXT I
S1 = Y(z + 6) + C1(1) * Y(z + 5) + C1(2) * Y(z + 4) + C1(3) * Y(z + 3) - S7
S2 = 0
FOR I = 1 TO G
FOR J = 1 TO G
S2 = S2 + CM(I, J) * M(J)
NEXT J
'Se obtienen los nuev
E(I) = E(I) + FO * S1 * S2

IF E(1) < .01 THEN E(1) = 1

S2 = 0
NEXT I
'PRINT "e(1)="; E(1)
z = z + 1

IF z > NR THEN GOTO 200

CTEMP1 = TEMP1 + 1
IF z < CTEMP1 OR FLAG$ = "LOOP" THEN
GOTO 500
ELSE
IF N = N.1 AND M = M.1 THEN
a(1) = 0: a(2) = 0: a(3) = 0
FOR I = 1 TO N.1
a(I) = A1(I)
NEXT I
B(0) = 0: B(1) = 0: B(2) = 0: B(3) = 0
FOR I = 0 TO M.1
B(I) = B1(I)
NEXT I

FOR I = 1 TO G
FOR J = 1 TO G
IF I = J THEN
CM(I, I) = DIAG
ELSE
CM(I, J) = 0
END IF
NEXT J
NEXT I
TRK = 0
FOR I = 1 TO G
TRK = TRK + CM(I, I)
NEXT I

FLAG$ = "LOOP"

```

```

GOTO 500
ELSE
  H1 = N + M: H2 = N.1 + M.1
  IF H1 >= H2 THEN
    G = N + M + 1
  ELSE
    G = N.1 + M.1 + 1
    N = N.1: M = M.1
  END IF

  FOR I = 1 TO G
    FOR J = 1 TO G
      IF I = J THEN
        CM(I, I) = DIAG
      ELSE
        CM(I, J) = 0
      END IF
    NEXT J
  NEXT I
  TRK = 0
  FOR I = 1 TO G
    TRK = TRK + CM(I, I)
  NEXT I

  a(1) = 0: a(2) = 0: a(3) = 0
  FOR I = 1 TO N.1
    a(I) = A1(I)
  NEXT I
  B(0) = 0: B(1) = 0: B(2) = 0: B(3) = 0
  FOR I = 0 TO M.1
    B(I) = B1(I)
  NEXT I
  FLAG$ = "LOOP"
  'Comentario al proc.
  'FOR i = 1 TO (M.1 + 1)
  'e(i) = B(i - 1)
  'NEXT i
  'j = 1
  'FOR i = (M.1 + 2) TO g
  'e(i) = C1(j) - A(j)
  'j = j + 1
  'NEXT i

  'FOR j = 2 TO 5: U(z + j) = 0: NEXT j
  'FOR j = 3 TO 5: Y(z + j) = 0: NEXT j
  'FOR j = 4 TO 5: YM(z + j) = 0: NEXT j
  E(1) = 1: FOR J = 2 TO 6: E(J) = 0: NEXT J

  GOTO 500
END IF
END IF

```

200 VIEW PRINT

'Regreso en el tiempo 6 pasos

FOR I = 0 TO NR

Y(I) = Y(I + 6)

U(I) = U(I + 6)

YM(I) = YM(I + 6)

NEXT I

END SUB

```
=====
PROGRAMA.ADA2
=====
```

```
SUB PROGRAMA.ADA2
'Comparte variables:N,M,N.1,M.1,NR,A(),B(),A1(),B1()

OVER$ = " "

DIM AUX4(NR + 10), M(10)
DIM E(10) AS DOUBLE
DIM F(10, 10) AS DOUBLE, G(10, 10) AS DOUBLE, CH(10, 10) AS DOUBLE
CLS
'Transferencia a modo grafico
SCREEN 3
LINE (0, 0)-(700, 320), , B
LINE (3, 3)-(697, 317), , B
LOCATE 4, 17
LINE (80, 37)-(620, 61), , BF
PRINT "¡ INGRESO DEL POLINOMIO Cr(q) DE CONTROL !"
LINE (3, 158)-(697, 158)
LOCATE 9, 8
PRINT "EL GRADO DEL POLINOMIO Cr(q) ES (max."; N; ")....:";
INPUT GC
IF GC > N THEN
PLAY "E30D40"
'Se asume que GC es igual a N
GC = N
ELSE
END IF

SELECT CASE GC

CASE 0
LOCATE 15, 35
PRINT "Cr(q)=1"
C1(1) = 0: C1(2) = 0: C1(3) = 0

CASE 1
LOCATE 15, 40
PRINT "c1="; : INPUT C1(1)
C1(2) = 0: C1(3) = 0

CASE 2
LOCATE 15, 40
PRINT "c1="; : INPUT C1(1)
LOCATE 16, 40
PRINT "c2="; : INPUT C1(2)
C1(3) = 0

CASE 3
LOCATE 15, 40
PRINT "c1="; : INPUT C1(1)
LOCATE 16, 40
PRINT "c2="; : INPUT C1(2)
LOCATE 17, 40
PRINT "c3="; : INPUT C1(3)

CASE ELSE
PLAY "E30D40"
```

CALL PROGRAMA.ADA2

END SELECT

'Calculos del subprograma

'Desplazamiento de YM()

FOR I = 0 TO NR

AUX4(I) = YM(I)

NEXT I

FOR I = 0 TO NR

YM(I + 6) = AUX4(I)

NEXT I

LOCATE 19, 3

PRINT STRING\$(74, "-")

LOCATE 21, 5

PRINT "INGRESE EL VALOR DE LA DIAGONAL DE F[I,J] :";

INPUT DIAG

LOCATE 21, 5

PRINT "SE ASUME CONOCIDA LA PLANTA INICIAL ?...<S>/<N> :";

OPTIONF\$ = INPUT\$(1)

SELECT CASE OPTIONF\$

CASE "N", "n"

PAR1\$ = "ES1"

CASE ELSE

PAR1\$ = "NES1"

END SELECT

G = N + M + 2

'Proteccion iterativa

FOR I = 1 TO G

FOR J = 1 TO G

CM(I, J) = 0

NEXT J

NEXT I

FOR I = 1 TO G

CM(I, I) = DIAG

NEXT I

TRK = 0

FOR I = 1 TO G

TRK = TRK + CM(I, I)

NEXT I

IF PAR1\$ = "NES1" THEN

'SP1 representa a s1

SP1 = C1(1) - a(1)

E(1) = B(0)

FOR I = 2 TO (M + 2)

E(I) = B(I - 1) + B(I - 2) * SP1

NEXT I

J = 2

FOR I = (M + 3) TO (M + N + 2)

E(I) = C1(J) - a(J) - a(J - 1) * SP1

J = J + 1


```

NEXT I
ELSE
E(1) = 1: FOR I = 2 TO 6: E(I) = 0: NEXT I
END IF
FLAG$ = " "
CLS
LOCATE 1, 10
PRINT "Procesando ....."
LOCATE 3, 8
PRINT STRING$(65, " ")
LOCATE 24, 8
PRINT STRING$(65, " ")
VIEW PRINT 5 TO 22
'Comienza el lazo principal
z = 0
600 I = z
ON ERROR GOTO OVER
IF OVER$ = "K" OR OVER$ = "k" THEN GOTO 700
HB1 = B(0) * U(I + 4) + B(1) * U(I + 3) + B(2) * U(I + 2) + B(3) * U(I + 1)
Y(I + 6) = -a(1) * Y(I + 5) - a(2) * Y(I + 4) - a(3) * Y(I + 3) + HB1
IF OVER$ = "K" OR OVER$ = "k" THEN GOTO 700
X1 = 0
L1 = I
FOR K = 2 TO (M + 2)
X1 = X1 + E(K) * U(L1 + 5)
IF OVER$ = "K" OR OVER$ = "k" THEN GOTO 700
L1 = L1 - 1
NEXT K
X2 = 0
L2 = I
FOR K = (M + 3) TO (M + N + 2)
X2 = X2 + E(K) * Y(L2 + 6)
IF OVER$ = "K" OR OVER$ = "k" THEN GOTO 700
L2 = L2 - 1
NEXT K
X3 = YH(I + 6) + C1(1) * YH(I + 6) + C1(2) * YH(I + 5) + C1(3) * YH(I + 4)
IF OVER$ = "K" OR OVER$ = "k" THEN GOTO 700
U(I + 6) = (1 / E(1)) * (X3 - X2 - X1)
IF OVER$ = "K" OR OVER$ = "k" THEN GOTO 700
SELECT CASE I
CASE 0 TO 9
SX1 = 2
CASE 10 TO 99
SX1 = 1
CASE IS >= 100
SX1 = 0
END SELECT
PRINT TAB(10); "U("; I; ")"; SPC(SX1); "=";
PRINT USING "#####.#####"; U(I + 6);
PRINT TAB(50); "Y("; I; ")"; SPC(SX1); "=";
PRINT USING "#####.#####"; Y(I + 6)

```

'Asignacion del vector M()

```
J = 2
FOR K = 1 TO (M + 2)
M(K) = U(z - J + 6)
J = J + 1
NEXT K
J = 2
FOR K = (M + 3) TO (M + N + 2)
M(K) = Y(z - J + 6)
J = J + 1
NEXT K
```

'P.Id.

```
S4 = 0
S5 = 0
FOR I = 1 TO 6
FOR J = 1 TO 6
S4 = S4 + M(J) * CM(J, I)
NEXT J
S5 = S5 + M(I) * S4
S4 = 0
NEXT I
S3 = 1 + S5
'Se obtiene S3
S6 = 0
FOR I = 1 TO 6
FOR J = 1 TO 6
S6 = S6 + CM(I, J) * M(J)
NEXT J
D(I) = S6
S6 = 0
NEXT I
FOR I = 1 TO 6
FOR J = 1 TO 6
F(I, J) = D(I) * M(J)
NEXT J
NEXT I
FOR I = 1 TO 6
FOR J = 1 TO 6
G(I, J) = 0
NEXT J
NEXT I
FOR I = 1 TO 6
FOR J = 1 TO 6
FOR K = 1 TO 6
G(I, J) = G(I, J) + F(I, K) * CM(K, J)
NEXT K
NEXT J
NEXT I
FOR I = 1 TO 6
FOR J = 1 TO 6
CM(I, J) = CM(I, J) - G(I, J) / S3
NEXT J
NEXT I

SUMATRAZA = 0
FOR I = 1 TO 6
SUMATRAZA = SUMATRAZA + CM(I, I)
NEXT I
```

```

FD = SUMATRAZA / TRK      'Factor de olvido
FOR I = 1 TO 6
FOR J = 1 TO 6
CM(I, J) = CM(I, J) / FD
NEXT J
NEXT I

S7 = 0
FOR I = 1 TO 6
S7 = S7 + E(I) * M(I)
NEXT I
S1 = Y(z + 6) + C1(1) * Y(z + 5) + C1(2) * Y(z + 4) + C1(3) * Y(z + 3) - S7
S2 = 0
FOR I = 1 TO 6
FOR J = 1 TO 6
S2 = S2 + CM(I, J) * M(J)
NEXT J
'Se obtienen los nuevos parametros estimados
E(I) = E(I) + FD * S1 * S2
IF E(I) < .01 THEN E(I) = B(0)
S2 = 0
NEXT I
z = z + 1

IF z > NR THEN GOTO 700

C2TEMP1 = TEMP1 + 2
IF z < C2TEMP1 OR FLAG# = "LOOP" THEN
  GOTO 600
ELSE
  IF N = N.1 AND M = M.1 THEN
    N = N.1: M = M.1
    a(1) = 0: a(2) = 0: a(3) = 0
    FOR I = 1 TO N.1
      a(I) = A1(I)
    NEXT I
    B(0) = 0: B(1) = 0: B(2) = 0: B(3) = 0
    FOR I = 0 TO M.1
      B(I) = B1(I)
    NEXT I
    FOR I = 1 TO 6
      FOR J = 1 TO 6
        IF I = J THEN
          CM(I, I) = DIAG
        ELSE
          CM(I, J) = 0
        END IF
      NEXT J
    NEXT I
    TRK = 0
    FOR I = 1 TO 6
      TRK = TRK + CM(I, I)
    NEXT I
    FLAG# = "LOOP"
    GOTO 600
  ELSE
    H11 = N + M: H12 = N.1 + M.1
    IF H11 >= H12 THEN
      G = N + M + 2
    ELSE
      G = N.1 + M.1 + 2
    N = N.1: M = M.1
  
```

```
END IF
```

```
FOR I = 1 TO G
```

```
FOR J = 1 TO G
```

```
IF I = J THEN
```

```
    CM(I, I) = DIAG
```

```
ELSE
```

```
    CM(I, J) = 0
```

```
END IF
```

```
NEXT J
```

```
NEXT I
```

```
TRK = 0
```

```
FOR I = 1 TO G
```

```
TRK = TRK + CM(I, I)
```

```
NEXT I
```

```
a(1) = 0: a(2) = 0: a(3) = 0
```

```
FOR I = 1 TO N.1
```

```
a(I) = A1(I)
```

```
NEXT I
```

```
B(0) = 0: B(1) = 0: B(2) = 0: B(3) = 0
```

```
FOR I = 0 TO M.1
```

```
B(I) = B1(I)
```

```
NEXT I
```

```
FLAG# = "LOOP"
```

```
GOTO 600
```

```
END IF
```

```
END IF
```

```
700 VIEW PRINT
```

```
'Regreso en el tiempo 6 pasos
```

```
FOR I = 0 TO NR
```

```
Y(I) = Y(I + 6)
```

```
U(I) = U(I + 6)
```

```
YM(I) = YM(I + 6)
```

```
NEXT I
```

```
END SUB
```

```
=====
SUBPROGRAMA PUNTO.AP
=====
```

```
SUB PUNTO.AP
```

```
CLS
```

```
ON ERROR GOTO OVER
```

```
'Transferencia a modo grafico
```

```
SCREEN 3
```

```
LINE (0, 0)-(700, 155), , B
```

```
LINE (3, 3)-(697, 152), , B
```

```
LOCATE 4, 18
```

```
LINE (100, 37)-(600, 61), , BF
```

```
PRINT "*" SECUENCIA DE REFERENCIA PUNTO POR PUNTO *
```

```
'Ingreso de valores para YM
```

```
'NR es el # de puntos de la secuencia de referencia
```

```
LOCATE 9, 8
```

```

PRINT "INGRESE EL # DE PUNTOS DE LA SECUENCIA DE REFERENCIA (max.300) :";
INPUT NR
'Dimensionamiento dinamico de YM(I)
IF NR > 300 THEN
  PLAY "E30D40"
  LOCATE 10, 8
  PRINT "Se asume 300 ptos. ..."
ELSE
END IF

'Ventana de ingreso
VIEW PRINT 14 TO 20
FOR I = 0 TO NR
PRINT TAB(33); "YM("; I; ")=";
INPUT YM(I)
NEXT I
CLS 2
VIEW
'Revision
PRINT TAB(10); "Desea revisar y/o modificar YM(t) ?....Presione <S> o <N>"
TAB1$ = INPUT$(1)

SELECT CASE TAB1$
  CASE "S", "s"
    PRINT
    PRINT TAB(10); "* Presione cualquier tecla para observar YM(t)"
    PRINT TAB(10); "* Presione <M> para modificar el ultimo YM(t)"
  'Correccion
    VIEW PRINT 19 TO 24
    FOR I = 0 TO NR
    PRINT TAB(33); "YM("; I; ")="; YM(I)
    TAB2$ = INPUT$(1)
    SELECT CASE TAB2$
      CASE "M", "m"
        PLAY "E30D40"
        PRINT TAB(33); "YM("; I; ")=";
        INPUT YM(I)
      CASE ELSE
    END SELECT
    NEXT I
  CASE "N", "n"
    PRINT
    PRINT TAB(25); "YM(t) SIN MODIFICACIONES...."
  CASE ELSE
    PLAY "E30D40"
END SELECT

PRINT
PRINT TAB(10); "* Desea observar graficamente YM(t) ?...Presione <S> o <N>"
TAB3$ = INPUT$(1)
SELECT CASE TAB3$
  CASE "S", "s"
    'Llamada a subprograma de graficos YM
    GRAF$ = "SECUENCIA DE REFERENCIA YM(t)"
    -----
    CALL GRAFYM
    -----
  CASE ELSE
    PLAY "E30D40"
END SELECT

```

VIEW PRINT

END SUB

```

=====
SUBPROGRAMA: RESULTADOS.1
=====

```

```

SUB RESULTADOS.1
ON ERROR GOTO OVER

```

```

'Variables compartidas

```

```

CLS

```

```

'Transferencia a modo grafico
SCREEN 3

```

```

LINE (0, 0)-(700, 320), , B
LINE (3, 3)-(697, 317), , B
LOCATE 3, 24
LINE (180, 20)-(520, 47), , BF
PRINT "¡ PRESENTACION DE RESULTADOS !"
LOCATE 5, 27
PRINT "....."
LOCATE 8, 19
LINE (152, 96)-(179, 113), , BF
PRINT "1"
LOCATE 8, 24
PRINT "EN FORMA NUMERICA"
LOCATE 12, 19
LINE (152, 152)-(179, 169), , BF
PRINT "2"
LOCATE 12, 24
PRINT "EN FORMA GRAFICA"
LOCATE 16, 19
LINE (152, 208)-(179, 225), , BF
PRINT "3"
LOCATE 16, 24
PRINT "EN FORMAS NUMERICA Y GRAFICA"

```

```

'Seleccion de una opcion
LOCATE 20, 25
PRINT "Seleccione un #: ";
OPTION$ = INPUT$(1)

```

```

'Seleccion
SELECT CASE OPTION$

```

```

CASE "1"

```

```

-----
CALL ESTADO1
-----

```

```

-----
CALL RESULTADOS.1.NUM
-----

```

```

CASE "2"

```

```

-----
CALL ESTADD1
-----
CALL RESULTADOS.1.GRAF
-----
CASE "3"
-----
CALL ESTADD1
-----
CALL RESULTADOS.1.NUM
-----
CALL RESULTADOS.1.GRAF
-----
CASE ELSE
PLAY "E30D40"
-----
CALL RESULTADOS.1
-----
END SELECT
END SUB

```

```

=====
SUBPROGRAMA: RESULTADOS.1.GRAF
=====

```

```

SUB RESULTADOS.1.GRAF
-----
ON ERROR GOTO OVER
'Resguardo de YN(I) en el VGAUX(I)
FOR I = 0 TO NR
VGAUX(I) = YN(I)
NEXT I

'Grificacion de la secuencia de control
'Intercambio
FOR I = 0 TO NR
YN(I) = U(I)
NEXT I

GRAF$ = " SECUENCIA DE CONTROL U(k)"
.....
IMPGRAF$ = "U(k)"
-----
CALL GRAFYM
-----
'Grificacion de la secuencia de referencia
'Recuperacion
FOR I = 0 TO NR
YN(I) = VGAUX(I)
NEXT I

```

```
GRAF$ = " SECUENCIA DE REFERENCIA YH(k) "
```

```
.....
IMPGRAF$ = "YM(k)"
-----
```

```
CALL GRAFYH
-----
```

```
'Graficacion de la secuencia de salida
```

```
'Intercambio
```

```
FOR I = 0 TO NR
```

```
YH(I) = Y(I)
```

```
NEXT I
```

```
GRAF$ = " SECUENCIA DE SALIDA Y(k) "
```

```
.....
IMPGRAF$ = "Y(K)"
-----
```

```
CALL GRAFYH
-----
```

```
END SUB
```

```
=====
SUBPROGRAMA: RESULTADOS.1.NUM
=====
```

```
SUB RESULTADOS.1.NUM
```

```
ON ERROR GOTO OVER
```

```
'Variables compartidas
```

```
CLS
```

```
'Transferencia a modo grafico
```

```
SCREEN 3
```

```
LOCATE 3, 24
```

```
LINE (175, 24)-(465, 45), , BF
```

```
PRINT "¡ RESULTADOS NUMERICOS ¡"
```

```
PRINT
```

```
PRINT TAB(10); "YM(k)"; TAB(36); "U(k)"; TAB(64); "Y(k)"
```

```
LOCATE 6, 1
```

```
PRINT STRING$(80, "-")
```

```
LOCATE 22, 1
```

```
PRINT STRING$(80, "-")
```

```
LOCATE 23, 20
```

```
PRINT "¡ Presione cualquier tecla para avanzar ..."
```

```
LOCATE 24, 20
```

```
PRINT "¡ Presione <T> para terminar ..."
```

```
'Presentacion de resultados
```

```
VIEW PRINT 7 TO 21
```

```
FOR I = 0 TO NR
```

```
SELECT CASE I
```

```
CASE 0 TO 9
```

```
SX1 = 2
```

```
CASE 10 TO 99
```

```
SX1 = 1
```

```
CASE IS >= 100
```

```
SX1 = 0
```

```
END SELECT
```



```

PRINT TAB(4); "YM("; I; ")"; SPC(SX1); "=";
PRINT USING "#####.#####"; YM(I);
PRINT TAB(31); "U("; I; ")"; SPC(SX1); "=";
PRINT USING "#####.#####"; U(I);
PRINT TAB(57); "Y("; I; ")"; SPC(SX1); "=";
PRINT USING "#####.#####"; Y(I)
OPTION$ = INPUT$(1)

IF OPTION$ = "T" OR OPTION$ = "t" THEN EXIT FOR
NEXT I
VIEW PRINT
LOCATE 25, 15
PRINT "Desea observar los resultados otra vez..?.<S> o <N> :";
OPTION1$ = INPUT$(1)

IF OPTION1$ = "S" OR OPTION1$ = "s" THEN
-----
CALL RESULTADOS.1.NUM
-----
ELSE
PLAY "E30D40"
END IF

CLS
LINE (30, 30)-(670, 70), , B
LOCATE 4, 14
PRINT "Desea imprimir los resultados numericos...<S> o <N> ?";
IMP1$ = INPUT$(1)

SELECT CASE IMP1$
CASE "S", "s"
CLS
LINE (30, 30)-(670, 70), , BF
LOCATE 4, 12
PRINT "##PREPARE LA IMPRESORA Y LUEGO PRESIONE CUALQUIER TECLA##"
IMP2$ = INPUT$(1)
'Llamado a subprograma de impresion
-----
CALL IMPRESION
-----
CASE ELSE
END SELECT

END SUB

'
'
'=====
'
'                -SUBPROGRAMA RETARDO
'
'=====

SUB RETARDO

CLS
ON ERROR GOTO OVER
SCREEN 3
LINE (0, 0)-(700, 320), , B
LINE (3, 3)-(697, 317), , B
LOCATE 4, 21
LINE (130, 37)-(565, 61), , BF
PRINT "## INGRESO DEL RETARDO DEL SISTEMA ##"

```

```

LINE (3, 90)-(697, 90)
LOCATE 12, 19
LINE (152, 152)-(179, 169), , BF
PRINT "A"
LOCATE 12, 24
PRINT "RETARDO DEL SISTEMA d=1 "
LOCATE 16, 19
LINE (152, 208)-(179, 225), , BF
PRINT "B"
LOCATE 16, 24
PRINT "RETARDO DEL SISTEMA d=2 "
LOCATE 21, 24
PRINT "SELECCIONE UNA OPCION : "
OPTION$ = INPUT$(1)

```

```

'Seleccion

```

```

SELECT CASE OPTION$
CASE "A", "a"

```

```

RET$ = "1" 'Referencia

```

```

-----
CALL PROGRAMA.A
-----

```

```

CASE "B", "b"

```

```

RET$ = "2" 'Referencia

```

```

-----
CALL PROGRAMA.B
-----

```

```

CASE ELSE
PLAY "E30040"

```

```

-----
CALL RETARDO
-----

```

```

END SELECT
END SUB

```

```

=====
SUBPROGRAMA RETARDO.MRAC
=====

```

```

SUB RETARDO.MRAC

```

```

CLS
SCREEN 3
LINE (0, 0)-(700, 320), , B
LINE (3, 3)-(697, 317), , B
LOCATE 4, 21
LINE (130, 37)-(565, 61), , BF
PRINT "*** INGRESO DEL RETARDO DEL SISTEMA **"
LINE (3, 90)-(697, 90)
LOCATE 12, 19
LINE (152, 152)-(179, 169), , BF

```

```
PRINT "A"
LOCATE 12, 24
PRINT "RETARDO DEL SISTEMA d=1"
LOCATE 16, 19
LINE (152, 208)-(179, 225), , BF
PRINT "B"
LOCATE 16, 24
PRINT "RETARDO DEL SISTEMA d=2"
LOCATE 21, 24
PRINT "SELECCIONE UNA OPCION : "
OPTION$ = INPUT$(1)
'Seleccion
SELECT CASE OPTION$
CASE "A", "a"
RET$ = "1" 'Referencia
-----
CALL PROGRAMA.ADA1
-----
CASE "B", "b"
RET$ = "2" 'Referencia
-----
CALL PROGRAMA.ADA2
-----
CASE ELSE
PLAY "E30D40"
-----
CALL RETARDO.MRAC
-----
END SELECT
PRINT a(1), a(2), a(3)
PRINT B(0), B(1), B(2), B(3)
PRINT "DATOS PLANTA MODIFICADA"
PRINT A1(1), A1(2), A1(3)
PRINT B1(0), B1(1), B1(2), B1(3)
PRINT "TIEMPO MODIFICACION ="; TEMP1
END SUB
```