

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA

“CONTROL DE MÍNIMA VARIANZA”

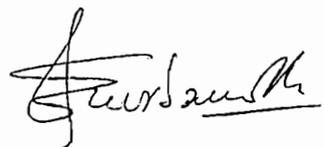
**TESIS PREVIA A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN ELECTRÓNICA Y CONTROL**

Edison Yépez Rojas

Quito, julio de 1996

CERTIFICACIÓN

Certifico que el presente trabajo de tesis fue realizado en su totalidad por el Sr. Edison Yépez Rojas, bajo mi supervisión y asesoramiento.

A handwritten signature in black ink, appearing to read 'Patricio Burbano R.', written in a cursive style.

Ing. Patricio Burbano R.

AGRADECIMIENTO

Mis más sinceros agradecimientos al Ing. Patricio Burbano R. por su invaluable colaboración en la realización de esta tesis; no solo por el aporte de sus conocimientos, sino también por su apoyo moral y su amistad.

DEDICATORIA

Dedico el presente trabajo a mis padres y a mi hermano, ya que sin su amor, paciencia y apoyo moral no me hubiera sido posible la culminación de mi carrera. Y a todas las personas que me han brindado su amistad a lo largo de estos años de estudio.

CONTENIDO

CAPÍTULO I: INTRODUCCIÓN

1.1	Introducción.....	1
1.2	Sistemas estocásticos.....	3
1.2.1	Conceptos y definiciones.....	3
1.2.2	Métodos de análisis de sistemas estocásticos.....	5
1.2.3	Clases de Procesos Estocásticos.....	8
1.3	El problema del Control Estocástico.....	13
1.3.1	El control de mínima varianza.....	13
1.3.2	Clases de controladores de mínima varianza.....	15
1.4	El problema de la estimación.....	16

CAPÍTULO II : CONTROL DE MÍNIMA VARIANZA

2.1.	El problema de la predicción.....	19
2.1.1	Predicción.....	19
2.1.2	Algoritmo para la predicción.....	24
2.1.3	Algoritmo básico del control de mínima varianza partiendo del algoritmo de predicción.....	26
2.2.	Control de mínima varianza.....	32
2.2.1	Control de mínima varianza generalizado.....	32
2.2.2	Control de mínima varianza generalizado para procesos sin tiempo muerto	33

2.2.3 Control de mínima varianza generalizado	
para procesos con tiempo muerto.....	46
2.2.4 Controladores de mínima varianza sin offset.....	51
2.3. Algoritmo para el control de mínima varianza.....	53
2.3.1 Algoritmo para el cálculo de los coeficientes	
de los polinomios F y G.....	53
2.3.2 Algoritmo para el cálculo de la señal de	
control $u(t)$ para el caso de regulación.....	55
2.3.3 Algoritmo para el cálculo de la señal de	
control $u(t)$ para el caso de seguimiento.....	58
2.4 Implementación y aplicaciones.....	61

CAPÍTULO III : PROGRAMA PARA EL CONTROL DE MÍNIMA VARIANZA

3.1 Programa principal.....	64
3.2 Rutinas para la simulación de plantas, salida y ruido.....	67
3.3 Rutinas para tiempo real (adquisición de datos).....	76
3.3.1 Tarjeta de adquisición de datos DAS - 128.....	77
3.3.2 Librería para utilización de la DAS - 128.....	78
3.4 Rutina de graficación.....	79
3.5 Rutina para el control de mínima varianza.....	81

CAPÍTULO IV : RESULTADOS Y CONCLUSIONES

4.1 Resultados de simulación.....	84
4.1.1 Ejemplo 1.....	84
4.1.2 Ejemplo 2.....	96
4.1.3 Ejemplo 3.....	102
4.1.4 Ejemplo 4.....	110
4.2 Resultados en tiempo real.....	117
4.2.1 Ejemplo 5.....	117
4.2.2 Ejemplo 6.....	126
4.3 Conclusiones.....	135
BIBLIOGRAFÍA.....	138

APÉNDICE A : MANUAL DEL USUARIO

A.1 Ingreso al programa y pantalla principal.....	A - 1
A.2 Pantallas de simulación y tiempo real.....	A - 3
A.3 Ingreso de datos.....	A - 7
A.4 Presentación de resultados.....	A - 10
A.5 Trabajando en tiempo real.....	A - 14

APÉNDICE B : LISTADO DEL PROGRAMA

B.1 Declaracion de variables globales.....	B - 1
B.2 Subrutinas de la pantalla de presentación.....	B - 2
B.3 Subrutinas de la pantalla principal.....	B - 2

B.4 Subrutinas de la pantalla de selección de modo de trabajo.....	B - 3
B.5 Subrutinas de la pantalla acerca del programa.....	B - 4
B.6 Subrutinas de la pantalla secundaria.....	B - 4
B.7 Subrutinas de la pantalla de selección de algoritmos.....	B - 6
B.8 Subrutinas de la pantalla para ingreso de datos.....	B - 9
B.9 Subrutinas de la pantalla de archivar datos.....	B - 15
B.10 Subrutinas de la pantalla cargar datos.....	B - 17
B.11 Subrutina simulación.....	B - 19
B.12 Subrutina para el cálculo de las varianzas.....	B - 22
B.13 Subrutinas de la pantalla de reporte de datos.....	B - 30
B.14 Subrutina para imprimir reporte de datos.....	B - 33
B.15 Subrutinas de la pantalla de menú de gráficos.....	B - 41
B.16 Subrutinas de la pantalla de gráficos.....	B - 43
B.17 Subrutinas de la pantalla cambiar tiempo de muestreo.....	B - 54
B.18 Subrutinas de la pantalla de cambio de escalas.....	B - 56
B.19 Subrutina para el cálculo de la señal de salida sin control de mínima varianza.....	B - 67
B.20 Subrutinas para graficación de las curvas de referencia.....	B - 70
B.21 Subrutina para mostrar los valores de las variables que están siendo graficadas.....	B - 70
B.22 Subrutina para realizar el desplazamiento en el tiempo de la variables.....	B - 71
B.23 Subrutina para encerrar las variables.....	B - 73

B.24 Rutina para calcular la señal de control

y la salida controlada.....B - 74

B.25 Rutina para limitar la señal de control en tiempo real.....B - 84

CAPÍTULO I: INTRODUCCIÓN

1.1 Introducción

1.2 Sistemas estocásticos

1.2.1 Conceptos y definiciones

1.2.2 Métodos de análisis de sistemas estocásticos

1.2.3 Clases de Procesos Estocásticos

1.3 El problema del Control Estocástico

1.3.1 El control de mínima varianza

1.3.2 Clases de controladores de mínima varianza

1.4 El problema de la estimación.

1. INTRODUCCIÓN

1.1 INTRODUCCIÓN.

En la práctica, muchos de los sistemas industriales son no determinísticos, es decir, son sistemas que son afectados por ruido de la planta, presencia de perturbaciones *aleatorias* e incertidumbre en las mediciones.

Al incluir variables aleatorias el control determinístico no permite el manejo de este tipo de sistemas, es por esta razón que se utiliza el *control estocástico*. En sistemas estocásticos se toma en cuenta las perturbaciones y ruido en la planta; y, la dinámica de las mediciones. Dentro del control estocástico se considera el *control de mínima varianza* en el cual se hace un tratamiento estadístico del sistema con el fin de que los valores obtenidos a la salida tengan la mínima varianza posible respecto a un valor esperado en la misma. En la actualidad se están implementando técnicas modernas de control para sistemas industriales con complejos y sofisticados esquemas, por lo que bien vale la pena profundizar en el desarrollo práctico del control estocástico.

El objetivo de esta tesis es el de implementar un programa computacional que permita realizar el control de sistemas no determinísticos utilizando un algoritmo generalizado de mínima varianza.

Se realizará un estudio teórico del problema de predicción, de los algoritmos del control de mínima varianza, de sus características y de las diferentes variantes a implementarse.

Se utilizará el lenguaje de programación *VISUAL BASIC* para la simulación y para la aplicación en tiempo real.

La simulación tendrá la posibilidad de manejar modelos de sistemas expresados en ecuaciones de diferencias y ruido blanco discreto. Para la

aplicación en tiempo real se utilizará la tarjeta de adquisición de datos DAS - 128 y software para identificación de sistemas que existe en el Laboratorio de Control.

En la simulación se utiliza la predicción para calcular el valor futuro de la salida afectada por las perturbaciones aleatorias; y, se aplica el algoritmo de mínima varianza para el cálculo de la señal requerida para que la salida predicha sea igual al valor deseado. Se abordarán los problemas de regulación y de seguimiento, considerando sistemas sin tiempo muerto y con tiempo muerto. Se incluye también el caso de tener controladores y diferentes procesos para la dinámica del ruido.

En la aplicación en tiempo real del programa, la única información que se dispone es la entrada y salida del sistema. Debido a esto, se estiman por separado los coeficientes de la ecuación de diferencias que describe matemáticamente al sistema (identificación del sistema). Una vez que los coeficientes del sistema han sido estimados se utilizan los algoritmos que fueron empleados durante la simulación para realizar el control del mismo.

Además de realizar el control de mínima varianza de un determinado sistema, el programa será capaz de mostrar de manera gráfica el comportamiento en el tiempo de la salida original (sin aplicar el control), de la salida después de realizar el control y de la señal de control, tanto para la simulación como para el; control en tiempo real.

Para el tratamiento del tema se lo ha dividido en cuatro capítulos. En el primero, se hace una introducción a los conceptos básicos sobre sistemas estocásticos que se utilizan para el tratamiento del problema. En el segundo, se desarrolla el algoritmo generalizado de los diferentes casos del control de mínima varianza y se hace un análisis de su implementación. En el tercero, se realiza propiamente la implementación del software que permite la aplicación práctica del control de mínima varianza. Y por último, en el cuarto capítulo, se presentan los resultados de la simulación y en

tiempo real del control desarrollado; y, se presentan las conclusiones y recomendaciones acerca de las experiencias obtenidas durante la realización de este trabajo.

1.2 SISTEMAS ESTOCÁSTICOS [2], [7]

1.2.1 CONCEPTOS Y DEFINICIONES

Procesos determinísticos. Son aquellos procesos en los cuales las señales o variables se pueden expresar a través de funciones analíticas o matemáticas del tipo exponencial o sinusoidal. Se puede predecir su valor futuro como función del tiempo. Al repetir un experimento controlado se tiene siempre la misma respuesta. No existe ruido o se lo desprecia por ser insignificante.

Procesos estocásticos. Son aquellos procesos en los cuales no se puede describir a las señales o variables a través de funciones analíticas o matemáticas. Tampoco es posible predecir el valor a un tiempo futuro puesto que este valor es aleatorio. Al repetir un experimento controlado se tienen siempre diferentes respuestas. En este tipo de procesos el ruido ya no es despreciable, siendo el ruido un proceso estocástico.

Definición. Un proceso estocástico, proceso aleatorio ó función aleatoria, es el conjunto de los posibles resultados de un proceso, en el cual, para cada experimento se tiene un resultado diferente. Cada resultado se lo conoce como realización o función de muestreo y se le asigna una función del tipo :

$$\{x(k, t)\} = \{x_k(t)\} \quad (1.1)$$

Donde: k es el espacio muestreado dado por todos los posibles resultados y t es una variable real, para sistemas dinámicos es el tiempo.

$$\{x_k(t)\} = \{x_1(t), x_2(t), \dots, x_k(t), \dots\} \quad (1.2)$$

Para k fijo el proceso estocástico se convierte en una *función de muestreo*; en una realización $x_k(t)$.

Si se fija t (para un valor $t = t_1$) se tiene una variable aleatoria; que puede tomar los valores $x_1(t_1), x_2(t_1), x_3(t_1), \dots, x_k(t_1), \dots$. La figura 1.1 ilustra gráficamente estos conceptos.

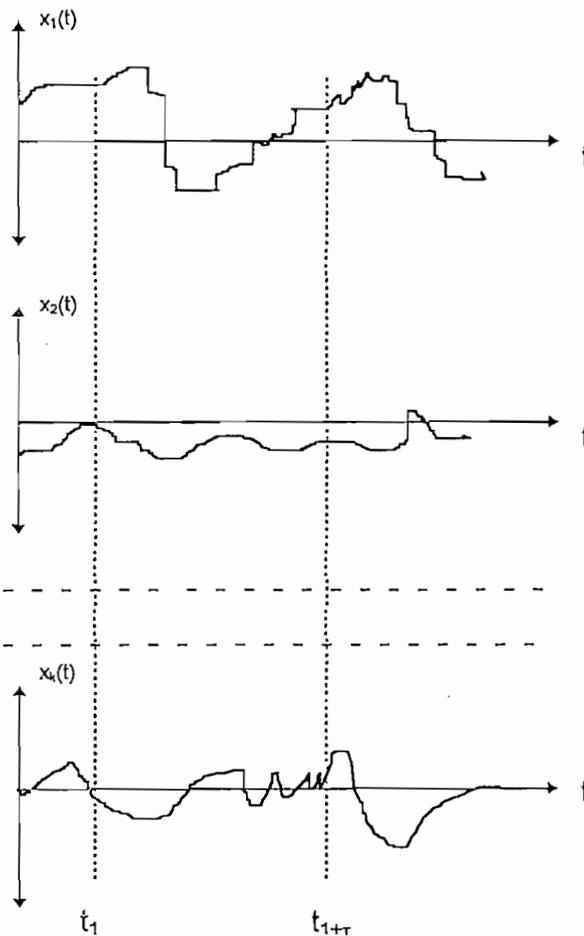


Figura 1.1. Ejemplos de procesos estocásticos.

Para su análisis se utilizan métodos estadísticos.

1.2.2 MÉTODOS DE ANÁLISIS DE SISTEMAS ESTOCÁSTICOS

Para la descripción de básica de las propiedades de un proceso estocástico se utilizan las siguientes funciones estadísticas.

Valor medio (μ_x). Proporciona una descripción básica de la intensidad de la información.

$$\mu_x = E\{x_k(t)\} \quad (1.3)$$

Para un conjunto de variables aleatorias en el espacio muestral :

$$\mu_x(t_1) = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=1}^N x_k(t_1) \quad (1.4)$$

Para una función de muestreo en el dominio del tiempo :

$$\mu_x(t) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^{\infty} x_k(t) dt \quad (1.5)$$

Función densidad de probabilidad ($f(x)$). Describe la probabilidad de que los datos asuman algún valor en un instante de tiempo, dentro de un rango definido.

El valor medio en base a la función de probabilidad $f(x)$ se expresa de la siguiente manera :

$$\mu_x = \int_{-\infty}^{\infty} x \cdot f(x) dx \quad (1.6)$$

Función de autocorrelación ($R_x(\tau, t_1)$). Describe la dependencia general de los datos en un instante de tiempo con los valores en otro instante de tiempo.

$$R_x(\tau, t_1) = E \{ x_k(t_1) \cdot x_k(t_1 + \tau) \} \quad (1.7)$$

Para un conjunto de variables aleatorias en el espacio muestral :

$$R_x(\tau, t_1) = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=1}^N x_k(t_1) \cdot x_k(t_1 + \tau) \quad (1.8)$$

Para una función de muestreo en el dominio del tiempo :

$$R_x(\tau, t_1) = \lim_{T \rightarrow \infty} \left[\frac{1}{T} \int_0^T x_k(t_1) x_k(t_1 + \tau) dt \right] \quad (1.9)$$

Aplicaciones:

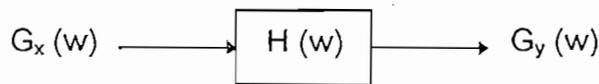
- Establece la influencia de valores en un instante de tiempo con otro instante de tiempo.
- Permite determinar en forma elemental si la información está enmascarada o corrupta con ruido.
- Su transformada de Fourier es la función densidad espectro de potencia.

Función densidad de espectro de potencia ($G_x(\omega)$). Describe la *composición en frecuencia* de los datos en términos de la densidad espectral de su *valor medio cuadrático*.

$$G_x(\omega) = \mathcal{F} \{R_x(\tau)\} = \int_{-\infty}^{\infty} e^{-j\omega\tau} \cdot R_x(\tau) \cdot d\tau \quad (1.10)$$

Aplicaciones:

- Determina las componentes en frecuencia del proceso estocástico.



$G_y(\omega) = |H(\omega)|^2 G_x(\omega)$, donde x es una señal de entrada y y es una señal de salida, $H(\omega)$ es una función de transferencia.

- Forma más idónea para distinguir el ruido presente en una señal.

Función covarianza ($C_x(\tau)$). Es la misma función de autocorrelación pero alrededor de la media.

$$C_x(\tau) = E \{ [x_k(t) - \mu_x] \cdot [x_k(t - \tau) - \mu_x] \} \quad (1.11)$$

En general se trabaja con procesos de media cero, es decir $\mu_x = 0$.

En ese caso : $C_x(\tau) = R_x(\tau)$.

1.2.3 CLASES DE PROCESOS ESTOCÁSTICOS

Proceso estacionario. Aquel en el cual las estadísticas que describen al sistema no dependen del instante inicial t_1 .

$$\mu_x(t_1) = \mu_x = \text{constante} \quad (1.12)$$

$$R_x(t_1, t_1 + \tau) = R_x(\tau) \quad (1.13)$$

En el estudio de los procesos estocásticos, las estadísticas pueden ser manipuladas en cualquier instante de tiempo.

Proceso ergódico. Es aquel proceso en el cual las estadísticas tomadas en un instante de tiempo sobre todas las funciones de muestreo son iguales a las estadísticas tomadas en una sola función de muestreo a lo largo de todo el tiempo. Es decir, el proceso estocástico es ergódico si es estacionario y existe una función de muestreo representativa del proceso.

Proceso Gaussiano. Es aquel en el cual su función densidad de probabilidad tiene la forma:

$$f(x) = \frac{1}{\sigma_x \sqrt{2\pi}} e^{-\frac{(x-\mu_x)^2}{2\sigma_x^2}} \quad (1.14)$$

Proceso Gaussiano normal : $f(x) = N(\mu, \sigma) = (0, 1)$.

La función densidad de probabilidad $f(x)$ está dada por dos valores μ_x y σ_x haciéndola fácil de manipular.

Los procesos industriales tienen aproximadamente una función densidad de probabilidad de Gauss, es decir, son procesos gaussianos, esto se debe al *Teorema Límite Central*.

El Teorema Límite Central dice que:

- La combinación lineal de procesos gaussianos es gaussiana.
- La combinación lineal de procesos gaussianos o no gaussianos tiende a ser gaussiana para un número elevado de muestras.

Procesos independientes. Son aquellos en que la función densidad de probabilidad conjunta entre X y Y se puede componer como el producto de las dos.

$$f(x,y) = f(x) \cdot f(y) \quad (1.15)$$

$$E[x(k) \cdot y(k)] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x \cdot y \cdot f(x,y) \cdot dx \cdot dy \quad (1.16)$$

Si los procesos X y Y son independientes :

$$E[x(k) \cdot y(k)] = \int_{-\infty}^{\infty} x \cdot f(x) \cdot dx \cdot \int_{-\infty}^{\infty} y \cdot f(y) \cdot dy \quad (1.17)$$

$$E[x(k) \cdot y(k)] = E[x(k)] \cdot E[y(k)] \quad (1.18)$$

Coefficiente de correlación :

$$\rho_{xy} = \frac{C_{xy}}{\sigma_x \cdot \sigma_y} = \frac{C_{xy}}{\sqrt{C_x} \cdot \sqrt{C_y}} \quad (1.19)$$

$$-1 \leq \rho_{xy} \leq 1$$

Si $\rho_{xy} = 0$ X y Y son procesos no correlacionados. Esto implica que $C_{xy} = 0$.

Si $\mu_x = \mu_y = 0$ entonces, $R_{xy} = C_{xy} = 0$.

donde :

$$R_{xy}(\tau) = E[x(k) \cdot y(k)] \quad (1.20)$$

$$C_{xy}(\tau) = E[(x(k) - \mu_x) \cdot (y(k) - \mu_y)] \quad (1.21)$$

Teorema de Factorización Espectral. El teorema de factorización está expresado, tanto para sistemas continuos como para sistemas discretos.

Procesos continuos:

$$\begin{aligned} G_y(\omega) &= H(e^{-j\omega})H(e^{j\omega})G_e(\omega) \\ G_y(\omega) &= |H(e^{j\omega})|^2 G_e(\omega) \end{aligned} \quad (1.22)$$

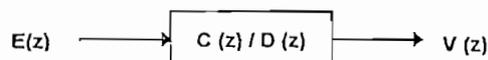
Procesos discretos:

$$G_y(z) = H(z^{-1})H(z)G_e(z) \quad (1.23)$$

$H(z) = C(z) / D(z)$ = función de transferencia discreta.

Consecuencias:

- $|H(e^{j\omega})|^2 = H(e^{-j\omega})H(e^{j\omega})$. Sistemas con función densidad de espectro de potencia racional.
- Se puede calcular $H(z)$ la función de transferencia discreta partir de la medición de $G_y(\omega)$ aplicando a la entrada ruido blanco.
- Se tiene que cualquier proceso $v(t)$ a la salida, se lo puede definir como la salida de un filtro a una entrada de ruido blanco $e(t)$.



Ruido Blanco . El ruido blanco es un proceso estocástico cuya función densidad de espectro de potencia es una constante para todo el rango de frecuencias desde menos infinito a más infinito, es decir, contiene componentes armónicas en todo el rango de frecuencias. Como se puede observar para el caso continuo en la figura 1.2.

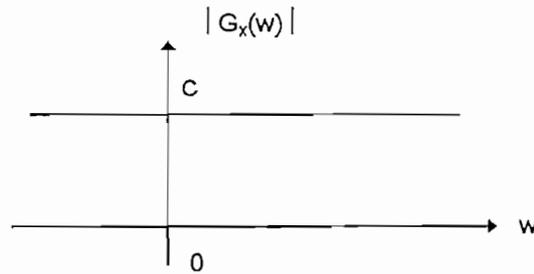


Figura 1.2 . Espectro de frecuencias del ruido blanco continuo.

Para el ruido blanco continuo se tiene la siguiente función idealizada de correlación :

$$R_x(\tau) = \frac{1}{2\pi} \cdot \int_{-\infty}^{\infty} e^{jw\tau} \cdot G_x(w) \cdot dw = c^2 \cdot \delta(\tau) \quad (1.24)$$

Para el ruido blanco discreto se tiene una función de correlación de la siguiente forma:

$$R_x(n) = \int_{-\pi}^{\pi} C \cdot e^{-jwn} \cdot dw = \frac{2 \cdot C \cdot \text{sen}(n \cdot \pi)}{n} \quad (1.25)$$

$$R_x(n) = \begin{cases} 2\pi \cdot C & \text{si } n = 0 \\ 0 & \text{si } n \neq 0 \end{cases}$$

Procesos estocásticos discretos :

Procesos con modelo ARMA. Autoregresivos de media móvil:

$$Y(z) = \frac{B(z)}{A(z)} \cdot U(z) \quad (1.26)$$

$$Y(z) \cdot A(z) = z^{-d} \cdot B(z) \cdot U(z) \quad (1.27)$$

$z^{-d} = \text{retardo}$

$$(1 + a_1 \cdot z^{-1} + \dots + a_n \cdot z^{-n}) \cdot Y(z) = (b_0 + b_1 \cdot z^{-1} + \dots + b_m \cdot z^{-m}) \cdot z^{-d} \cdot U(z) \quad (1.28)$$

$$y(t) + a_1 \cdot y(t-1) + \dots + a_n \cdot y(t-n) = b_0 \cdot u(t-d) + b_1 \cdot u(t-d-1) + \dots + b_m \cdot u(t-d-m) \quad (1.29)$$

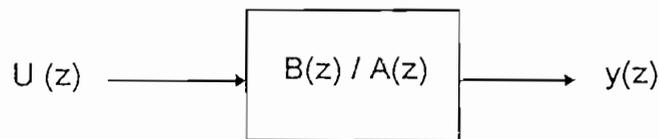


Figura 1.3. Diagrama de bloques del modelo ARMA.

Procesos con modelo ARMAX. Modelo ARMA más una entrada auxiliar (ruido blanco a la salida).

$$Y(z) = \frac{B(z)}{A(z)} \cdot U(z) + \frac{C(z)}{A(z)} \cdot E(z) \quad (1.30)$$

$$Y(z) \cdot A(z) = z^{-d} \cdot B(z) \cdot U(z) + C(z) \cdot E(z) \quad (1.31)$$

$z^{-d} = \text{retardo}$

$$(1 + a_1 \cdot z^{-1} + \dots + a_n \cdot z^{-n}) \cdot Y(z) = (b_0 + b_1 \cdot z^{-1} + \dots + b_m \cdot z^{-m}) \cdot z^{-d} \cdot U(z) + (c_0 + c_1 \cdot z^{-1} + \dots + c_p \cdot z^{-p}) \cdot E(z)$$

(1.32)

$$y(t) + a_1 \cdot y(t-1) + \dots + a_n \cdot y(t-n) = b_0 \cdot u(t-d) + b_1 \cdot u(t-d-1) + \dots + b_m \cdot u(t-d-m) + c_0 \cdot e(t-d) + c_1 \cdot e(t-d-1) + \dots + c_p \cdot e(t-d-p)$$

(1.33)

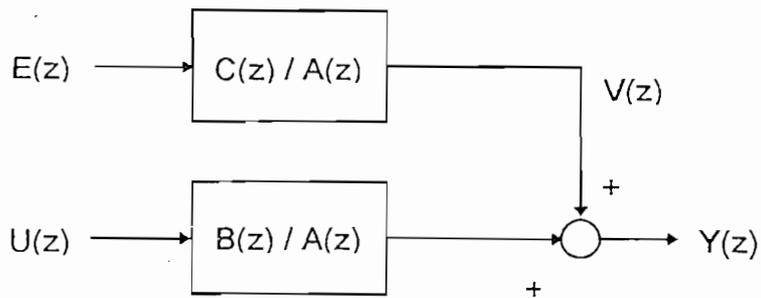


Figura 1.4. Diagrama de bloques del modelo ARMAX.

Donde $e(t)$ es ruido blanco discreto y $v(t)$ es un proceso estocástico cualquiera.

1.3 EL PROBLEMA DEL CONTROL ESTOCÁSTICO [3], [5]

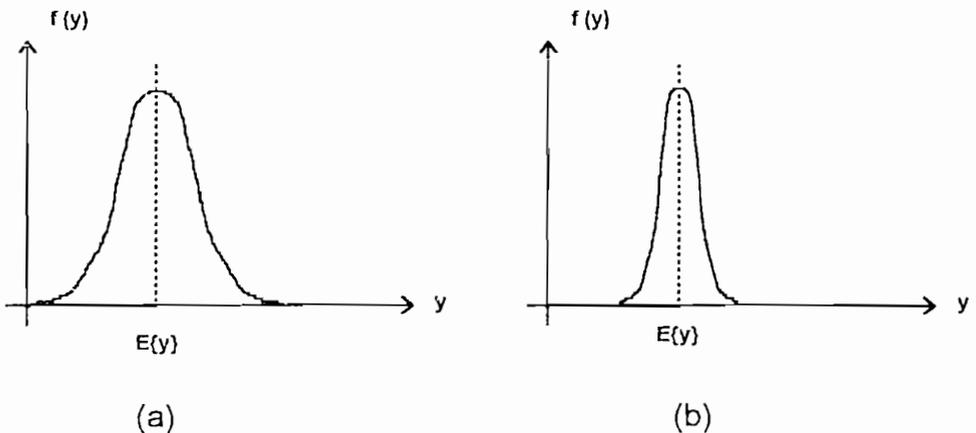
1.3.1 EL CONTROL DE MÍNIMA VARIANZA

El control determinístico no toma en cuenta las perturbaciones o ruido en la planta puesto que maneja señales y variables determinísticas. Esto

presupone que los niveles de ruido son despreciables y no se toman en cuenta las perturbaciones, ni los errores en las mediciones.

El control estocástico concibe al sistema como un proceso aleatorio en el cual se toman en cuenta las perturbaciones aleatorias, la dinámica de los procesos de ruido, ya sea en la planta o en la medición. En este caso no se puede realizar el mismo análisis matemático, ya que ahora se debe utilizar métodos estadísticos en el diseño del control.

Esto se debe al hecho de que en el control determinístico se tiene que la salida es un valor fijo, en cambio en el control estocástico la salida del sistema es un conjunto de valores agrupados en torno a un valor esperado de salida. Lo que se busca entonces es minimizar la desviación de los posibles valores de salida alrededor del valor esperado; esto implica minimizar la distribución de la salida como se muestra en la figura 1.5.



$E\{y\}$ = Valor esperado de la salida y .

Figura 1.5 . Curvas de distribución de los posibles valores de la salida y . (a) Sin control de mínima varianza. (b) Con control de mínima varianza.

El problema del control estocástico en este caso se refiere a un problema de mínima varianza de la salida.

1.3.2 CLASES DE CONTROLADORES DE MÍNIMA VARIANZA

El problema básico del control de mínima varianza es el de obtener una señal de regulación u que minimice la varianza en base a la predicción de la señal de salida.

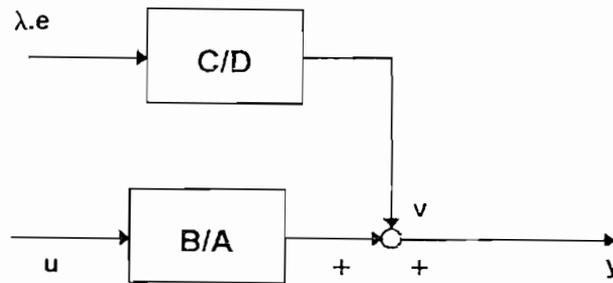


Figura 1.6 . Diagrama de bloques del Control de mínima varianza básico.

Donde la función de transferencia B/A representa la dinámica de la planta y la función de transferencia C/D la dinámica del proceso de ruido, como lo muestra la figura 1.6.

En este caso se toma como función de costo $I(t)$ la varianza de la salida :

$$I(t) = E[y^2(t)] \quad (1.34)$$

Del control de mínima varianza básico se llega al control de mínima varianza generalizado en el cual se abarcan los problemas de regulación y seguimiento, para sistemas con y sin tiempo muerto. El modelo del control de mínima varianza generalizado se puede observar en la figura 1.7 en donde d es el retardo y Q/P es el controlador para mínima varianza.

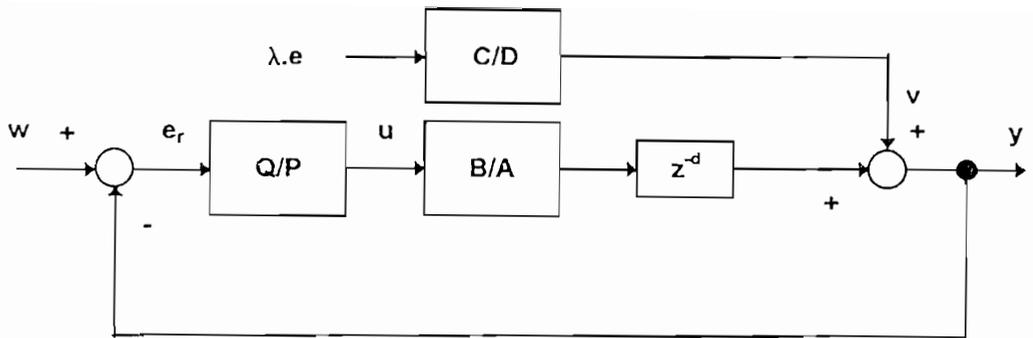


Figura 1.7 . Diagrama de bloques del Control de mínima varianza generalizado.

En el capítulo 2 se analiza el problema de predicción, luego el control de mínima varianza básico, para concluir con el algoritmo del control de mínima varianza generalizado con todas sus variantes. Para finalizar con esta introducción se aborda el problema de la estimación.

1.4 EL PROBLEMA DE LA ESTIMACIÓN [6]

Estimación óptima

Es un algoritmo computacional que procesa mediciones para obtener un estimador de mínimo error del estado de un sistema utilizando : conocimiento de la dinámica del sistema y de las mediciones, asumiendo estadísticas de los ruidos de la planta y errores en la medición, e información sobre la condición inicial. La figura 1.8 muestra los tres problemas de la estimación.

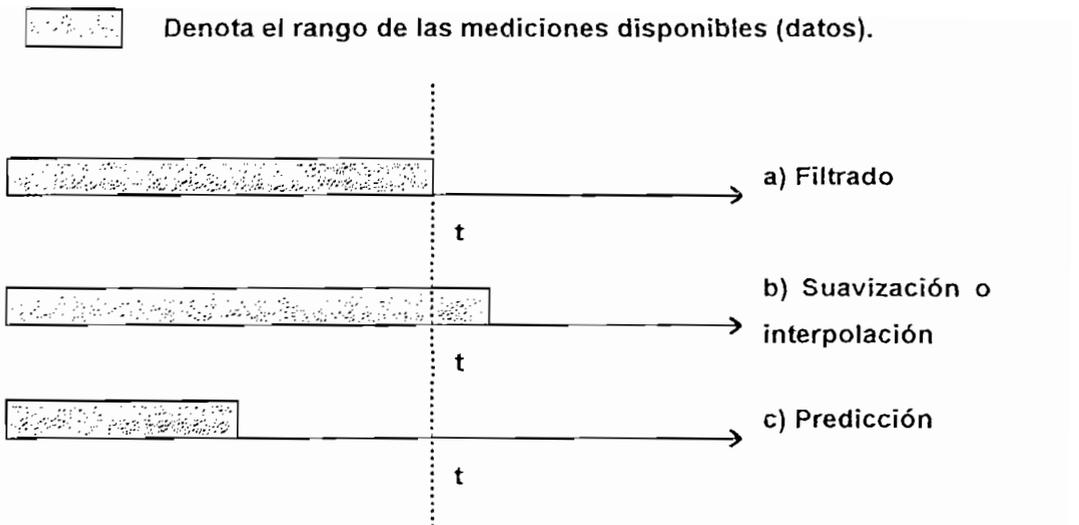


Figura 1.8 . Los tres problemas de la estimación.

- **Filtrado** . Se tiene filtrado cuando el tiempo al cual se desea la estimación coincide con la última medición.
- **Suavización**. Cuando el tiempo de interés cae dentro del rango de mediciones disponibles.
- **Predicción**. Cuando el tiempo de interés ocurre después de la obtención de la información disponible.

La suavización no es de interés en esta tesis. Lo que se utilizará es la predicción que se aborda en el siguiente capítulo.

En el filtrado se tienen dos problemas :

1. Dado el modelo, estimar o filtrar las variables (de estado) involucradas en el modelo.
2. Medidas las variables (entrada y salida), estimar o identificar los parámetros del modelo.

En simulación se conoce el modelo, por lo tanto se realiza el control de mínima varianza; en tiempo real, se mide la entrada y la salida por lo que es necesario previamente identificar el modelo y luego aplicar el control de mínima varianza.

El objetivo de esta tesis es el de realizar el control de mínima varianza. Por lo tanto, no se abordará los aspectos de identificación y se utilizarán los trabajos realizados anteriormente como tesis de grado para identificar el modelo de la planta. Para las pruebas experimentales se utilizarán modelos conocidos.

2. CONTROL DE MÍNIMA VARIANZA

2.1 EL PROBLEMA DE LA PREDICCIÓN

2.1.1 PREDICCIÓN [2]

Una vez revisados los conceptos fundamentales de lo que son los procesos estocásticos y el control estocástico se llega a la conclusión de que la solución del control estocástico está estrechamente relacionada con el problema de la predicción.

La teoría de la predicción puede ser establecida en diferentes formas las cuales difieren entre si en las asunciones hechas en el proceso, en el criterio, y los predictores admisibles.

Para el análisis del problema de la predicción que se desarrollará a continuación se harán las siguientes asunciones :

- El proceso a ser predicho es de tiempo discreto, estacionario, gaussiano, con densidad de espectro racional.
- El mejor predictor es aquel que minimiza la varianza del error de predicción.
- Un predictor admisible para $y(t+k)$ (donde t es la variable tiempo en el dominio discreto y k es un número entero mayor o igual a cero) es cualquier función de todas las observaciones pasadas. Por lo tanto para predecir $y(t+k)$ se pueden usar todas las observaciones $y(t)$, $y(t-1)$, $y(t-2)$, ...

El proceso estocástico será definido a través de ecuaciones de diferencias. Para facilitar el análisis se define el operador de desplazamiento en el tiempo "q" tal que:

$$q^k \cdot x(t) = x(t + k), \text{ donde } k = 0, \pm 1, \pm 2, \pm 3, \dots \quad (2.1)$$

Para determinar el predictor de "k" pasos hacia adelante se definirá el sistema de la manera que se muestra en la figura 2.1.

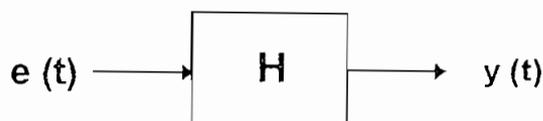


Figura 2.1 Diagrama de bloques del sistema estocástico a predecir.

De acuerdo al teorema de factorización espectral de los procesos de tiempo discreto, se puede representar a la función de transferencia "H" como el cociente entre dos polinomios "A" y "C" tales que :

$$H(z) = \frac{C(z)}{A(z)} \quad (2.2)$$

entonces

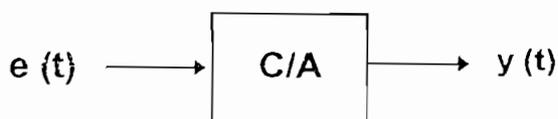


Figura 2.2 Diagrama de bloques del sistema estocástico a predecir con H expresada en función de los polinomios A y C.

Donde $A(z)$ es un polinomio que tiene todos los ceros dentro del círculo unitario y $C(z)$ tiene todos los ceros dentro o en el círculo unitario. Sin embargo, para todos los siguientes análisis se asumirá que el polinomio $C(z)$ no tiene ceros en el círculo unitario.

Los coeficientes de A y de C se los define de la siguiente manera :

$$A(z) = 1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_n z^{-n} \quad (2.3)$$

$$C(z) = 1 + c_1 z^{-1} + c_2 z^{-2} + \dots + c_n z^{-n} \quad (2.4)$$

Utilizando el operador q y las ecuaciones anteriores, el proceso estocástico de la figura 2.2 puede ser representado como :

$$A(q^{-1}) \cdot y(t) = \lambda \cdot C(q^{-1}) \cdot e(t) \quad (2.5)$$

donde λ es un escalar cualquiera. En este caso se considera un proceso normal de ruido (1, 0). El valor $[\lambda \cdot e(t)]$ modifica la desviación de $e(t)$ al valor λ y produce una varianza λ^2 .

Para determinar al mejor predictor se construye una función $\hat{y}(t+k/t)$ de $y(t)$, $y(t-1)$, $y(t-2)$, . . . , tal que :

$$E [y(t+k) - \hat{y}(t+k/t)]^2 = \text{error de predicción} \quad (2.6)$$

sea lo más pequeño posible. Donde $\hat{y}(t+k/t)$ es el valor estimado de la salida "k pasos adelante", en base de $y(t)$, $y(t-1)$, $y(t-2)$, . . . , que es justamente el predictor.

Para encontrar una expresión para el predictor se considera que al tiempo t las salidas $y(t)$, $y(t-1)$, $y(t-2)$, . . . han sido observadas y se desea predecir $y(t+k)$. De la ecuación 2.6 se obtiene :

$$y(t+k) = \lambda \cdot \frac{C(q^{-1})}{A(q^{-1})} \cdot e(t+k) \quad (2.7)$$

El término a la derecha de la ecuación 2.7 es una combinación lineal de $e(t+k)$, $e(t+k-1)$, . . . , $e(t+1)$, $e(t)$, $e(t-1)$, Las variables aleatorias

$e(t)$, $e(t - 1)$, ... pueden calcularse de los datos observados. Las variables aleatorias $e(t + 1)$, ..., $e(t + k)$ son independientes de las observaciones.

El miembro derecho consiste así de términos los cuales pueden ser calculados directamente de las observaciones, y de términos los cuales son independientes de las observaciones. Para separar estos grupos de términos se definen los polinomios "F" y "G" tales que :

$$\lambda \cdot \frac{C(q^{-1})}{A(q^{-1})} \cdot e(t+k) = \lambda \cdot \left[F(q^{-1}) \cdot e(t+k) + q^{-k} \cdot \frac{G(q^{-1})}{A(q^{-1})} \cdot e(t+k) \right] \quad (2.8)$$

siendo F y G de orden $k - 1$ y $n - 1$ respectivamente, así :

$$F(z) = 1 + f_1 z^{-1} + \dots + f_{k-1} z^{-k+1} \quad (2.9)$$

$$G(z) = g_0 + g_1 z^{-1} + \dots + g_{n-1} z^{-n+1} \quad (2.10)$$

La ecuación 2.7 puede reescribirse como :

$$y(t+k) = \lambda \cdot F(q^{-1}) \cdot e(t+k) + \lambda \cdot \frac{G(q^{-1})}{A(q^{-1})} \cdot e(t) \quad (2.11)$$

El segundo término del miembro derecho de la ecuación 2.11 es una función de $e(t)$, $e(t - 1)$, ... y puede ser calculado de las mediciones disponibles $y(t)$, $y(t - 1)$, ...

De la ecuación 2.7 se obtiene :

$$\lambda \cdot e(t) = \frac{A(q^{-1})}{C(q^{-1})} \cdot y(t) \quad (2.12)$$

Reemplazando en 2.11 se llega a :

$$y(t+k) = \lambda \cdot F(q^{-1}) \cdot e(t+k) + \frac{G(q^{-1})}{C(q^{-1})} \cdot y(t) \quad (2.13)$$

El primer término del miembro derecho es una función lineal de $e(t+1)$, $e(t+2)$, ..., $e(t+k)$, los cuales son independientes de las mediciones disponibles. El segundo término es una función lineal de las mediciones disponibles.

Siendo y una función arbitraria de $y(t)$, $y(t-1)$, ..., entonces :

$$\begin{aligned} E[y(t+k) - \hat{y}]^2 &= E[\lambda \cdot F(q^{-1}) \cdot e(t+k)]^2 + E\left[\hat{y} - \frac{G(q^{-1})}{C(q^{-1})} \cdot y(t)\right]^2 \\ &+ 2 \cdot E[\lambda \cdot F(q^{-1}) \cdot e(t+k)] \left[\hat{y} - \frac{G(q^{-1})}{C(q^{-1})} \cdot y(t)\right] \end{aligned} \quad (2.14)$$

El último término del miembro se anula ya que, $e(t+1)$, $e(t+2)$, ..., $e(t+k)$ son independientes de $y(t)$, $y(t-1)$, ..., y $e(t)$ tiene valor medio cero para todo t . Así se tiene :

$$\begin{aligned} E[y(t+k) - \hat{y}]^2 &= E[\lambda \cdot F(q^{-1}) \cdot e(t+k)]^2 + E\left[\hat{y} - \frac{G(q^{-1})}{C(q^{-1})} \cdot y(t)\right]^2 \\ E[y(t+k) - \hat{y}]^2 &\geq \lambda^2 \cdot [1 + f_1^2 + \dots + f_{k-1}^2] \end{aligned} \quad (2.15)$$

de donde el mínimo se obtiene para $E[y(t+k) - \hat{y}] = \lambda^2 \cdot [1 + f_1^2 + \dots + f_{k-1}^2]$, y entonces debe cumplirse que :

$$\hat{y} = \hat{y}(t+k/t) = \frac{G(q^{-1})}{C(q^{-1})} \cdot y(t) \quad (2.16)$$

El predictor de k pasos hacia adelante óptimo está dado por la siguiente ecuación de diferencias :

$$\hat{y}(t+k/t) + c_1 \cdot \hat{y}(t+k-1/t-1) + \dots + c_n \cdot \hat{y}(t+k-n/t-n) = g_0 \cdot y(t) + g_1 \cdot y(t-1) + \dots + g_{n-1} \cdot y(t-n+1) \quad (2.17)$$

El error de predicción está dado por :

$$\begin{aligned} \tilde{y}(t+k/t) &= y(t+k) - \hat{y}(t+k/t) = \lambda \cdot F(q^{-1}) \cdot e(t+k) \\ &= \lambda \cdot [e(t+k) + f_1 \cdot e(t+k-1) + \dots + f_{k-1} \cdot e(t+1)] \quad (2.18) \end{aligned}$$

Para obtener el predictor se debe determinar los coeficientes de los polinomios $F(z)$ y $G(z)$.

Despejando de la ecuación 2.8 se tiene :

$$C(q^{-1}) = A(q^{-1}) \cdot F(q^{-1}) + q^{-k} G(q^{-1}) \quad (2.19)$$

Si A y C son dos polinomios arbitrarios en q^{-1} , existen dos únicos polinomios F y G los cuales satisfacen la ecuación 2.19 .

2.1.2 ALGORITMO PARA LA PREDICCIÓN [2]

Los polinomios F y G pueden ser determinados dividiendo C/A . Así, el polinomio F es el cociente cuando se divide C para A y $q^{-k} G(q^{-1})$ es el residuo.

$$\frac{C(q^{-1})}{A(q^{-1})} = F(q^{-1}) + q^{-k} \cdot \frac{G(q^{-1})}{A(q^{-1})} \quad (2.20)$$

2.1.3 ALGORITMO BÁSICO DEL CONTROL DE MÍNIMA VARIANZA PARTIENDO DEL ALGORITMO DE PREDICCIÓN [2]

Habiendo obtenido la solución al problema de la predicción y partiendo de la misma se obtendrá una solución del problema de mínima varianza.

Se asume que el proceso a ser regulado es un sistema muestreado, lineal dinámico e invariante en el tiempo de orden m -ésimo con una entrada u y una salida y . La relación entrada salida puede ser descrita como una ecuación de diferencias de m -ésimo orden :

$$y(t) + a_1^1 \cdot y(t-1) + \dots + a_m^1 \cdot y(t-m) = b_0^1 \cdot u(t-k) + b_1^1 \cdot u(t-k-1) + \dots + b_m^1 \cdot u(t-k-m) \quad (2.24)$$

donde en este caso el intervalo de muestreo es la unidad de tiempo. Introduciendo el operador q , los polinomios A_T y B_T quedan definidos de la siguiente manera :

$$A_1(z) = 1 + a_1^1 z^{-1} + \dots + a_m^1 z^{-m} \quad (2.25)$$

$$B_1(z) = b_0^1 + b_1^1 z^{-1} + \dots + b_m^1 z^{-m} \quad (2.26)$$

La relación entrada - salida puede ser escrita así :

$$y(t) = \frac{B_1(q^{-1})}{A_1(q^{-1})} \cdot u(t-k) \quad (2.27)$$

También se asume que la influencia del ambiente en el proceso puede ser caracterizada por perturbaciones las cuales son procesos estocásticos.

Como el sistema es lineal, se puede hacer uso del *Teorema de Superposición* y representar a todas las posibles perturbaciones como a una sola actuando en la salida. El proceso y su ambiente puede entonces ser descrito por el siguiente modelo matemático :

$$y(t) = \frac{B_1(q^{-1})}{A_1(q^{-1})} \cdot u(t-k) + v(t) \quad (2.28)$$

Además se asume que la perturbación $v(t)$ es un proceso estacionario, gaussiano, con densidad espectral racional, y que puede ser representado como :

$$v(t) = \lambda \cdot \frac{C_1(q^{-1})}{A_2(q^{-1})} \cdot e(t) \quad (2.29)$$

donde $e(t)$ ($t = 0, \pm 1, \pm 2, \dots$) es una secuencia de variables aleatorias independientes normalmente distribuidas $(0,1)$, y C_1 y A_2 son polinomios. De las ecuaciones 2.28 y 2.29 se encuentra que el sistema y su ambiente pueden ser representados de la siguiente manera :

$$y(t) = \frac{B_1(q^{-1})}{A_1(q^{-1})} \cdot u(t-k) + \lambda \cdot \frac{C_1(q^{-1})}{A_2(q^{-1})} \cdot e(t) \quad (2.30)$$

Una representación en diagrama de bloques del sistema descrito en la ecuación 2.30 puede verse en la figura 2.3 .

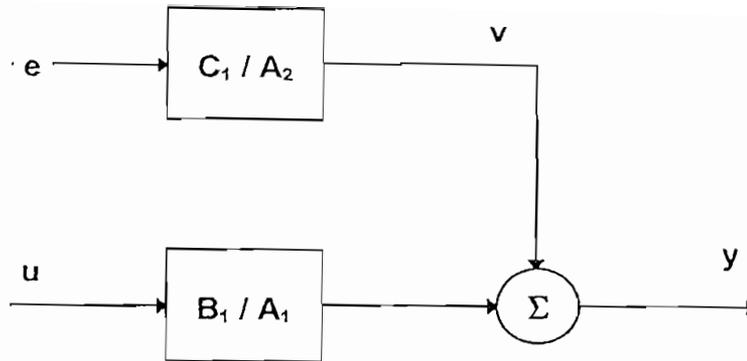


Figura 2.3 . Diagrama de bloques del sistema descrito en la ecuación 2.30 .

La ecuación 2.30 es así una forma canónica para un sistema dinámico, invariante en el tiempo con una entrada y una salida, con retardo en el tiempo que es un entero múltiplo del intervalo de muestreo, sujeto a perturbaciones las cuales son estacionarias con densidades de espectro racionales. Los polinomios $A_2(z)$ y $C_1(z)$ pueden ser escogidos siempre para tener sus ceros dentro del círculo unitario. Como la perturbación v fue asumida estacionaria, A_2 no puede tener ceros en el círculo unidad. Para simplificar el análisis se reescribirá la ecuación 2.30 de una forma ligeramente diferente, así :

$$A(q^{-1}) \cdot y(t) = B(q^{-1}) \cdot u(t - k) + \lambda \cdot C(q^{-1}) \cdot e(t) \quad (2.31)$$

donde los polinomios A , B y C están definidos de la siguiente manera :

$$A = A_1 \cdot A_2, \quad B = B_1 \cdot A_2 \quad \text{y} \quad C = C_1 \cdot A_1.$$

Estos polinomios tienen la siguiente forma :

$$A(z) = 1 + a_1 z^{-1} + \dots + a_n z^{-n} \quad (2.32)$$

$$B(z) = b_0 + b_1 z^{-1} + \dots + b_n z^{-n} \quad (2.33)$$

$$C(z) = 1 + c_1 z^{-1} + \dots + c_n z^{-n} \quad (2.34)$$

No se pierde generalidad al asumir que todos los polinomios son de orden n , debido a que siempre se podrá tener coeficientes igual a cero. El polinomio $C(z)$ siempre será escogido de tal manera que tenga todos sus ceros dentro del círculo unitario.

Ahora, el criterio para el problema del control es escoger el mismo tal que, se minimice la varianza de la salida y .

Se asume que la ley de control debería ser tal que el valor de u al tiempo t sea una función de las salidas observadas incluyendo la del tiempo t ($y(t)$, $y(t-1)$, $y(t-2)$, ...) y de todas las señales de control pasadas ($u(t-1)$, $u(t-2)$, ...).

A la ley de control óptima que hace que la salida del sistema descrito en la ecuación 2.31 tenga la mínima varianza posible se la llama *estrategia de mínima varianza*.

Para resolver el problema se considera la situación al tiempo t . Se han obtenido las mediciones $y(t)$, $y(t-1)$, ..., y se conocen todas las acciones de control pasadas $u(t-1)$, $u(t-2)$, El problema es determinar $u(t)$ de tal forma que la varianza de la salida sea lo más pequeña posible. De la ecuación 2.31 se observa que la señal de control $u(t)$ influenciará a $y(t+k)$ pero a ninguna otra salida anterior. Se considera que

$$y(t+k) = \frac{B(q^{-1})}{A(q^{-1})} u(t) + \lambda \cdot \frac{C(q^{-1})}{A(q^{-1})} e(t+k) \quad (2.35)$$

El último término es una función lineal de $e(t+k)$, $e(t+k-1)$, ..., $e(t+1)$, $e(t)$, $e(t-1)$, Siguiendo desde la ecuación 2.31 se tiene que $e(t)$, $e(t-1)$, ... pueden ser calculados de la información disponible al tiempo t . Para hacer esto se reescribe la ecuación 2.35 usando la identidad 2.19, es decir

$$C(q^{-1}) = A(q^{-1}) \cdot F(q^{-1}) + q^{-k} G(q^{-1})$$

donde F y G son polinomios de grados $k-1$ y $n-1$ definidos por las ecuaciones 2.9 y 2.10.

Comparando con 2.11 en la solución del problema de la predicción se tiene

$$y(t+k) = \lambda \cdot F(q^{-1}) \cdot e(t+k) + \frac{B(q^{-1})}{A(q^{-1})} \cdot u(t) + \lambda \cdot \frac{G(q^{-1})}{A(q^{-1})} \cdot e(t) \quad (2.36)$$

Resolviendo 2.31 para $\lambda \cdot e(t)$, se tiene

$$\lambda \cdot e(t) = \frac{A(q^{-1})}{C(q^{-1})} \cdot y(t) - \frac{B(q^{-1})}{C(q^{-1})} \cdot q^{-k} \cdot u(t) \quad (2.37)$$

Eliminando $e(t)$ entre 2.36 y 2.37 se encuentra :

$$y(t+k) = \lambda \cdot F(q^{-1}) \cdot e(t+k) + \left[\frac{B(q^{-1})}{A(q^{-1})} - q^{-k} \cdot \frac{B(q^{-1}) \cdot G(q^{-1})}{A(q^{-1})C(q^{-1})} \right] \cdot u(t) + \frac{G(q^{-1})}{C(q^{-1})} \cdot y(t) \quad (2.38)$$

Usando la identidad 2.19, el segundo término de la igualdad puede ser reducido y obtenerse

$$y(t+k) = \lambda \cdot F(q^{-1}) \cdot e(t+k) + \frac{G(q^{-1})}{C(q^{-1})} \cdot y(t) + \frac{B(q^{-1}) \cdot F(q^{-1})}{C(q^{-1})} \cdot u(t) \quad (2.39)$$

Ahora siendo $u(t)$ una función arbitraria de $y(t)$, $y(t-1)$, ... y $u(t-1)$, $u(t-2)$, ... Entonces

$$E[y^2(t+k)] = E\left[\lambda \cdot F(q^{-1}) \cdot e(t+k)\right]^2 + E\left[\frac{G(q^{-1})}{C(q^{-1})} \cdot y(t) + \frac{B(q^{-1}) \cdot F(q^{-1})}{C(q^{-1})} \cdot u(t)\right]^2 \quad (2.40)$$

Los términos cruzados se anulan debido a que $e(t+1)$, $e(t+1)$, ..., $e(t+k)$ son independientes de $y(t)$, $y(t-1)$, ... y de $u(t-1)$, $u(t-2)$, ... Por lo tanto

$$Ey^2(t+k) \geq \lambda^2 \cdot [1 + f_1^2 + f_2^2 + \dots + f_{k-1}^2] \quad (2.41)$$

donde la igualdad se obtiene para

$$B(q^{-1}) \cdot F(q^{-1}) \cdot u(t) + G(q^{-1}) \cdot y(t) = 0 \quad (2.42)$$

ecuación que permite calcular la ley de control deseada. Entonces :

$$u(t) = - \frac{G(q^{-1})}{B(q^{-1}) \cdot F(q^{-1})} \cdot y(t) \quad (2.43)$$

El error de regulación del sistema óptimo es de media variable y orden k .

$$\begin{aligned} y(t) &= \lambda \cdot F_0(q^{-1}) \cdot e(t) \\ y(t) &= \lambda \cdot [e(t) + f_1 \cdot e(t-1) + \dots + f_{k-1} \cdot e(t-k+1)] \end{aligned} \quad (2.44)$$

Se debe anotar que los polos del sistema en lazo cerrado son iguales a los ceros del polinomio $C(z)$.

2.2 CONTROL DE MÍNIMA VARIANZA

2.2.1 CONTROL DE MÍNIMA VARIANZA GENERALIZADO [1]

En los *controladores de mínima varianza* la varianza de la variable de salida es minimizada.

$$\text{var}[y(t)] = E[y^2(t)] \quad (2.45)$$

En el algoritmo básico desarrollado en la sección 2.1.3, la variable manipulada $u(t)$ no fue ponderada de manera que, en muchos casos se tendrán cambios excesivos en la entrada. Para evitar esta situación se introduce un término r que tiene como misión el ponderar las variaciones de la variable de entrada $u(t)$. Debido a esto el criterio a minimizar cambia, de la siguiente manera

$$E[y^2(t+k) + r u^2(t)] \quad (2.46)$$

Las perturbaciones aleatorias que afectan al sistema (procesos estocásticos) se modelan matemáticamente utilizando la ecuación 2.29, ligeramente cambiada, es decir :

$$v(t) = \lambda \cdot \frac{C(q^{-1})}{D(q^{-1})} \cdot e(t) \quad (2.47)$$

Como resultado de la introducción del término adicional r en la entrada, la varianza de la variable controlada ya no es la mínima posible, en su lugar es la varianza de la combinación obtenida la que es minimizada, en cuyo caso se llega a un *controlador de mínima varianza generalizado*.

En las siguientes secciones se hará un estudio de los controladores de mínima varianza generalizados para procesos con tiempo muerto y sin él; y, se abordarán los problemas de regulación y seguimiento.

2.2.2 CONTROL DE MÍNIMA VARIANZA GENERALIZADO PARA PROCESOS SIN TIEMPO MUERTO [1]

El proceso a ser controlado está descrito por la función de transferencia :

$$G_p(z) = \frac{Y(z)}{U(z)} = \frac{B(z)}{A(z)} = \frac{b_0 + b_1 \cdot z^{-1} + \dots + b_n \cdot z^{-n}}{1 + a_1 \cdot z^{-1} + \dots + a_n \cdot z^{-n}} \quad (2.48)$$

y por el proceso estocástico

$$G_{Pr}(z) = \frac{V(z)}{E(z)} = \lambda \cdot \frac{C(z)}{D(z)} = \frac{\lambda \cdot (1 + c_1 \cdot z^{-1} + \dots + c_n \cdot z^{-n})}{1 + d_1 \cdot z^{-1} + \dots + d_n \cdot z^{-n}} \quad (2.49)$$

donde $e(t)$ es una secuencia de variables aleatorias independientes normalmente distribuidas $(0, 1)$.

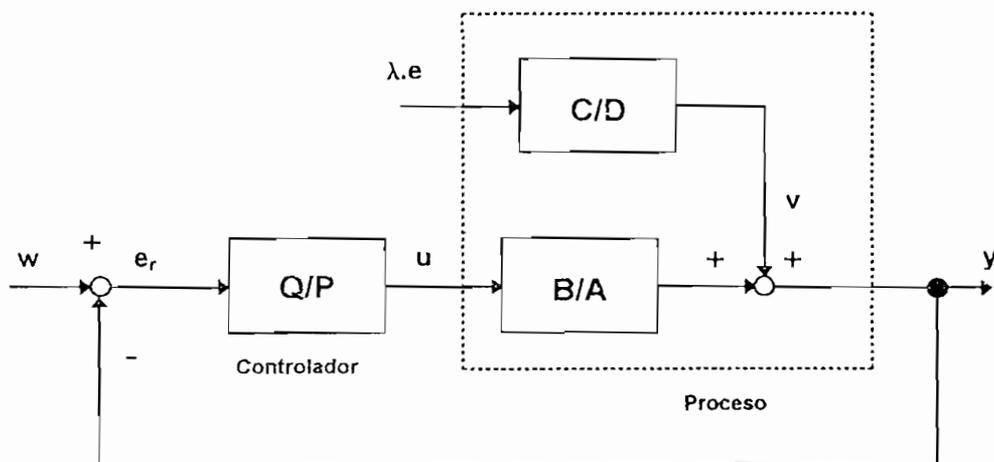


Figura 2.4 . Control de procesos sin tiempo muerto con controladores de mínima varianza .

Después de realizar el producto de los términos entre paréntesis, y transformar al dominio del tiempo se obtiene :

$$y(t+1) + (a_1 + d_1) \cdot y(t) + \dots + a_n \cdot d_n \cdot y(t-2n+1) = b_1 \cdot u(t) + (b_2 + b_1 \cdot d_1) \cdot u(t-1) + \dots + b_n \cdot d_n \cdot u(t-2n+1) + \lambda \cdot [e(t+1) + (a_u + d_1) \cdot e(t) + \dots + a_n \cdot d_n \cdot e(t-2n+1)] \quad (2.55)$$

Por lo que el criterio de costo de la ecuación 2.50 se convierte en :

$$I(t+1) = E \left\{ \left[\begin{aligned} &-(a_1 + d_1) \cdot y(t) - \dots - a_n \cdot d_n \cdot y(t-2n+1) + b_1 \cdot u(t) + (b_2 + b_1 \cdot d_1) \cdot u(t-1) + \\ &+ \dots + b_n \cdot d_n \cdot u(t-2n+1) + \lambda \cdot [(a_1 + c_1) \cdot e(t) + \dots + a_n \cdot c_n \cdot e(t-2n+1)] + \lambda \cdot e(t+1) \end{aligned} \right]^2 + r \cdot u^2(t) \right\} \quad (2.56)$$

Al instante de tiempo t todos los valores de las señales son conocidos con excepción de $u(t)$ y $v(t+1)$. Por lo tanto, solo el valor esperado de $v(t+1)$ debe ser calculado. Además, $v(t+1)$ es independiente de todos los valores de las señales.

$$I(t+1) = \left[\begin{aligned} &-(a_1 + d_1) \cdot y(t) - \dots - a_n \cdot d_n \cdot y(t-2n+1) + b_1 \cdot u(t) + (b_2 + b_1 \cdot d_1) \cdot u(t-1) + \\ &+ \dots + b_n \cdot d_n \cdot u(t-2n+1) + \lambda \cdot [(a_1 + c_1) \cdot e(t) + \dots + a_n \cdot c_n \cdot e(t-2n+1)] \end{aligned} \right]^2 + \lambda^2 \cdot E \{ e^2(t+1) \} + 2 \cdot \lambda \cdot \left[\begin{aligned} &-(a_1 + d_1) \cdot y(t) - \dots - a_n \cdot d_n \cdot y(t-2n+1) + \lambda \cdot [(a_1 + c_1) \cdot e(t) + \dots + a_n \cdot c_n \cdot e(t-2n+1)] \end{aligned} \right] E \{ e(t+1) \} + r \cdot u^2(t) \quad (2.57)$$

De donde, la condición para tener una señal de control óptima $u(t)$ se da para :

$$G_{RMV4}(z) = \frac{U(z)}{Y(z)} = \frac{Q(z)}{P(z)} = \frac{C(z) - D(z)}{B(z)} \quad (2.66)$$

Propiedades del controlador :

- Orden del controlador

	Orden del numerador	Orden del denominador
G_{RMV1}	$2n - 1$	$2n$
G_{RMV2}	$2n - 1$	$2n - 1$
G_{RMV3}	$n - 1$	n
G_{RMV4}	$n - 1$	$n - 1$

Tabla 2.1 Orden de las funciones de transferencia de los controles.

Debido a que las funciones de transferencia G_{RMV1} y G_{RMV2} tienen el más alto orden se prefiere asumir que $A = D$; es decir, G_{RMV3} y G_{RMV4} para modelar al ruido.

- Cancelación de polos y ceros

En base a la forma que tienen los controladores de mínima varianza anteriormente desarrollados se llega a establecer :

G_{RMV1} : Los polos del proceso ($A(z) = 0$) son cancelados. Por lo que el controlador no debería ser aplicado a plantas cuyos polos están cerca del círculo unitario o a procesos inestables.

G_{RMV2} : Los polos y ceros del proceso ($A(z) = 0$ y $B(z) = 0$) son cancelados. Por lo que el controlador no debe ser usado en procesos como el G_{RMV1} , ni en procesos con comportamiento de fase no mínima.

G_{RMV3} : En general no existe restricción.

G_{RMV4} : Los ceros de la planta ($B(z) = 0$) son cancelados. Por lo que este controlador no debería ser usado en procesos con comportamiento de fase no mínima.

Se puede observar que el controlador cuya aplicación es la más general, es aquel cuya función de transferencia es la G_{RMV3} , puesto que no existen restricciones para A y B.

- **Estabilidad**

Para el análisis de estabilidad se asume que las condiciones listadas en el numeral anterior han sido satisfechas. La ecuación característica en lazo cerrado del sistema es :

$$1 - G_R(z) \cdot G_p(z) = 0 \quad (2.67)$$

El controlador de mínima varianza expresado en la función de transferencia G_{RMV1} , permite expresar la ecuación 2.67 de la siguiente manera :

$$\left[\frac{r}{b_1} \cdot A(z) + z \cdot B(z) \right] \cdot C(z) \cdot z^{2n} = \left[\frac{r}{b_1} \cdot A(z) + z \cdot B(z) \right] \cdot C(z) = 0 \quad (2.68)$$

Continuando el análisis de estabilidad en lazo cerrado se tiene :

Para G_{RMV1} y G_{RMV3} ($r \neq 0$):

- Los ceros del filtro de ruido $C(z) = 0$ deben hallarse dentro del círculo unitario del plano z.

- Los ceros de

$$\left[\frac{r}{b_1} \cdot A(z) + z \cdot B(z) \right] = 0$$

deben hallarse dentro del círculo unitario. A mayor ponderación de r los ceros de lazo cerrado están más cercanos a los de la planta ($A = 0$)

Para G_{RMV2} y G_{RMV4} ($r = 0$):

- La ecuación característica para lazo cerrado se convierte en :

$$z \cdot B(z) \cdot C(z) = 0$$

- Por lo que los ceros del proceso $B(z) = 0$ y los del filtro de ruido $C(z) = 0$ deben hallarse dentro del círculo unidad.

- Los polos del filtro de ruido $D(z) = 0$ (para G_{RMV2}) no influyen la ecuación característica. Por lo que ellos pueden hallarse en cualquier lugar del plano z .

- **Variable controlada y (t)**

Para el comportamiento en lazo cerrado se utiliza para G_{RMV1} la siguiente función de transferencia :

$$\begin{aligned}
 G_V(z) &= \frac{Y(z)}{\lambda \cdot E(z)} = \frac{\frac{C}{D}}{1 + \frac{A(z) \cdot [C(z) - D(z)] \cdot z}{z \cdot B(z) \cdot D(z) + \frac{r}{b_1} \cdot A(z) \cdot C(z)} \cdot \frac{B(z)}{A(z)}} \\
 &= \frac{\frac{r}{b_1} \cdot A(z) \cdot C(z) + z \cdot B(z) \cdot D(z)}{\left[\frac{r}{b_1} \cdot A(z) + z \cdot B(z) \right] \cdot D(z)} \\
 &= 1 + \frac{\frac{r}{b_1} \cdot A(z) \cdot [C(z) - D(z)]}{\left[\frac{r}{b_1} \cdot A(z) + z \cdot B(z) \right] \cdot D(z)}
 \end{aligned} \tag{2.69}$$

La variable controlada $y(t)$ para $r \neq 0$ es un proceso autoregresivo de media móvil (ARMA) de orden $2n$ para G_{RMV1} y de orden n para G_{RMV3} . Cuando r tiende a cero, es decir, para G_{RMV2} y G_{RMV4} , se tiene que $Y(z)$ tiende a $\lambda \cdot E(z)$; es decir, la variable controlada se vuelve estadísticamente independiente (ruido blanco), proceso con varianza $\sigma_y^2 = \lambda^2 \cdot \sigma_e^2$. La más pequeña ponderación r de la entrada del proceso, da la varianza más pequeña de la variable controlada $y(t)$; y, la variable controlada $y(t)$ converge a la señal de ruido blanco $\lambda \cdot e(t)$. La mínima varianza la cual puede ser obtenida por un controlador de mínima varianza es por lo tanto :

$$\text{mín var} [y(t)] = \lambda^2 \quad (2.70)$$

Un caso especial se da cuando $C(z) = D(z)$; todos los controladores de mínima varianza son idénticamente igual a cero y a la salida se tiene ruido estadísticamente independiente $v(t) = \lambda \cdot e(t)$ y no se puede ejercer ningún control para disminuir la varianza. Únicamente para el ruido "coloreado" la varianza de la variable controlada puede ser reducida. Ruido coloreado es aquel cuya densidad de espectro de frecuencias existe para ciertos valores de frecuencia y no para todo el rango de frecuencias como es el caso del ruido blanco.

- **Comportamiento de los controladores de mínima varianza para perturbaciones constantes ($E\{v(t)\} \neq 0$)**

De la ecuación 2.63 se puede verificar que el comportamiento estático de G_{RMV1} satisface :

$$\lim_{z \rightarrow 1} G_{RMV1}(z) = \frac{A(1) \cdot [C(1) - D(1)]}{B(1) \cdot D(1) + \frac{r}{b_1} \cdot A(1) \cdot C(1)} = \frac{\sum a_i \cdot [\sum c_i - \sum d_i]}{\sum b_i \cdot \sum d_i - \frac{r}{b_1} \cdot \sum a_i \cdot \sum c_i} \quad (2.71)$$

Aquí el \sum se lee como $\sum_{i=0}^n$. Si el proceso $G_P(z)$ tiene un comportamiento de acción proporcional; es decir, $\sum a_i \neq 0$ y $\sum b_i \neq 0$, entonces el controlador G_{RMV1} en general tiene un comportamiento estático de acción proporcional. Para perturbaciones constantes se presentan *offsets*. Esto también se presenta en los controladores de mínima varianza G_{RMV2} , G_{RMV3} y G_{RMV4} . Para evitar estos *offsets* en controladores de mínima varianza se deben realizar algunas modificaciones, estas modificaciones serán discutidas más adelante en el numeral 2.2.4.

En la tabla 2.2 se hace un resumen de las propiedades típicas de los controladores de mínima varianza sin tiempo muerto. Se puede observar que las mejores propiedades se tienen para la función de transferencia G_{RMV3} .

Controlador		G_R	Peligro de inestabilidad para	Inestabilidad para
G_{RMV1}		$\frac{z \cdot A \cdot [C - D]}{z \cdot B \cdot D + \frac{r}{b_1} \cdot A \cdot C}$	$A^- = 0$	$C^- = 0$
G_{RMV2}	$r = 0$	$\frac{z \cdot A \cdot [C - D]}{z \cdot B \cdot D}$	$A^- = 0$ $B^- = 0$	$C^- = 0$ $B^- = 0$
G_{RMV3}	$A = D$	$\frac{z \cdot [C - A]}{z \cdot B + \frac{r}{b_1} \cdot C}$	-	$C^- = 0$
G_{RMV4}	$A = D$ $r = 0$	$\frac{z \cdot [C - A]}{z \cdot B}$	$B^- = 0$	$C^- = 0$ $B^- = 0$

Tabla 2.2 Características de estabilidad de los controladores de mínima varianza ($A^- = 0$ significa que los ceros de A están sobre o fuera del círculo unitario).

Por lo tanto, para la realización práctica de los controladores de mínima varianza se debería asumir que $D(z) = A(z)$. En la derivación de los

controladores de mínima varianza se asumió a $b_0 = 0$. Si $b_0 \neq 0$ solo se necesita reemplazar a b_1 por b_0 y escribir :

$$B(z) = b_0 + b_1 \cdot z^{-1} + \dots + b_n \cdot z^{-n}$$

- Elección del factor r

En un sistema determinístico (no afectado por perturbaciones aleatorias) cualquiera se tiene el siguiente diagrama de bloques :

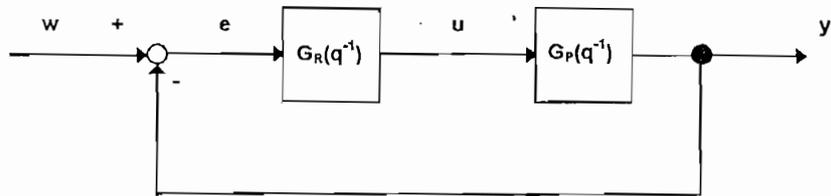


Figura 2.5. Diagrama de bloques de un sistema determinístico.

Donde :

$$G_p(q^{-1}) = \frac{y(t)}{u(t)} = \frac{B(q^{-1})}{A(q^{-1})} = \frac{b_0 + b_1 \cdot q^{-1} + \dots + b_n \cdot q^{-n}}{1 + a_1 \cdot q^{-1} + \dots + a_n \cdot q^{-n}} \quad (2.72)$$

$$G_R(q^{-1}) = \frac{u(t)}{e(t)} = \frac{Q(q^{-1})}{P(q^{-1})} = \frac{q_0 + q_1 \cdot q^{-1} + \dots + q_j \cdot q^{-j}}{1 + p_1 \cdot q^{-1} + \dots + p_i \cdot q^{-i}} \quad (2.73)$$

Ahora se encuentra la relación entre la señal de referencia $w(t)$ y la de control $u(t)$, entonces :

$$\frac{u(t)}{w(t)} = \frac{G_R(q^{-1})}{1 + G_R(q^{-1}) \cdot G_P(q^{-1})} \quad (2.74)$$

Desarrollando la función de transferencia anterior en función de los polinomios numeradores y denominadores se tiene :

$$\frac{u(t)}{w(t)} = \frac{A(q^{-1}) \cdot Q(q^{-1})}{P(q^{-1}) \cdot A(q^{-1}) + Q(q^{-1}) \cdot B(q^{-1})} \quad (2.75)$$

Para el tiempo $t = 0$ todos los operadores q con retardo mayor a uno se anulan, quedando la ecuación 2.75 de la siguiente manera :

$$\frac{u(0)}{w(0)} = \frac{q_0}{1 + q_0 \cdot b_0} \quad (2.76)$$

Si se asume que $b_0 = 0$ se tiene :

$$u(0) = q_0 \cdot w(0) \quad (2.77)$$

Esto quiere decir, que q_0 es una medida para el tamaño de la entrada del proceso y que la influencia del factor de ponderación en la variable manipulada $u(t)$ puede ser estimada observando la primera entrada $u(0)$ después de una entrada $w(t)$ de tipo escalón unitario ($w(t) = 1(t)$).

Sea ahora la función de transferencia $G_R(q^{-1})$ del controlador de la ecuación 2.73 igual a $G_{RMV1}(q^{-1})$ de la ecuación 2.63, entonces :

$$\frac{Q(q^{-1})}{P(q^{-1})} = - \frac{A(q^{-1}) \cdot [C(q^{-1}) - D(q^{-1})] \cdot q}{q \cdot B(q^{-1}) \cdot D(q^{-1}) + \frac{r}{b_1} \cdot A(q^{-1}) \cdot C(q^{-1})} \quad (2.78)$$

Para $t = 0$ todos los operadores con retardo mayor a uno se anulan dando como resultado :

$$q_0 = \frac{c_1 - d_1}{b_1 - \frac{r}{b_1}} \quad (2.79)$$

De la misma manera para G_{RMV3} :

$$q_0 = \frac{c_1 - a_1}{b_1 - \frac{r}{b_1}} \quad (2.80)$$

Para G_{RMV2} ($r = 0$) :

$$q_0 = \frac{c_1 - d_1}{b_1} \quad (2.81)$$

Para G_{RMV4} ($r = 0$) :

$$q_0 = \frac{c_1 - a_1}{b_1} \quad (2.82)$$

Por lo que, hay aproximadamente una relación hiperbólica entre q_0 y r / b_1 para $r / b_1 \gg b_1$. Una reducción de q_0 a la mitad se obtiene para $r = b_1^2$.

b_1 puede ser estimado de la respuesta transitoria del proceso, para una entrada paso u_0 la relación $b_1 = y(1) / u_0$ se mantiene.

Para un sistema con tiempo muerto para $GRMV1d$ y para $GRMV3d$ se tiene que $q_0 = g_0 / (b_1 + r / b_1)$.

2.2.3 CONTROL DE MÍNIMA VARIANZA GENERALIZADO PARA PROCESOS CON TIEMPO MUERTO [1]

El proceso a ser controlado puede ser descrito por la función de transferencia con tiempo muerto $G_p(z)$ como se muestra en la figura 2.6.

$$G_p(z) = \frac{Y(z)}{U(z)} = \frac{B(z)}{A(z)} \cdot z^{-d} = \frac{b_1 \cdot z^{-1} + \dots + b_n \cdot z^{-n}}{1 + a_1 \cdot z^{-1} + \dots + a_n \cdot z^{-n}} \cdot z^{-d} \quad (2.83)$$

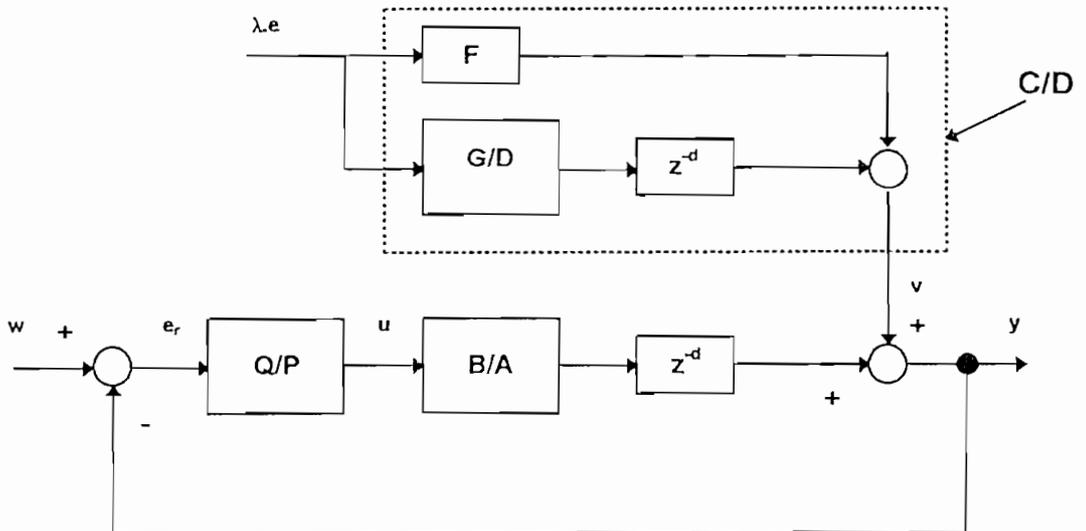


Figura 2.6. Control de procesos con tiempo muerto con controladores de mínima varianza.

El filtro de perturbaciones fue modelado en la ecuación 2.49. Como la entrada $u(t)$ para procesos con tiempo muerto d puede influenciar la variable controlada $y(t+d+1)$ el criterio de costo para este caso es el siguiente ($k = d+1$):

$$I(t+1) = E\{y^2(t+d+1) + r \cdot u^2(t)\} \quad (2.84)$$

De manera similar a lo hecho en la ecuación 2.52, para la predicción de $y(t + d + 1)$ resulta :

$$z^{d+1} \cdot Y(z) = \frac{B(z)}{A(z)} \cdot z \cdot U(z) + \lambda \cdot \frac{C(z)}{D(z)} \cdot z^{d+1} \cdot E(z) \quad (2.85)$$

Al tiempo t , para el cual $u(t)$ debe ser calculado, las señales de las perturbaciones $e(t + 1), \dots, e(t + d + 1)$ son desconocidas, esta parte del filtro de perturbaciones es separada de la manera siguiente :

$$z^{d+1} \cdot Y(z) = \frac{B(z)}{A(z)} \cdot z \cdot U(z) + \lambda \cdot \left[F(z) \cdot z^{d+1} + \frac{G(z)}{D(z)} \right] \cdot E(z) \quad (2.86)$$

Como también se puede apreciar en la figura 2.6, el filtro de perturbaciones es separado en un término $F(z)$ el cual describe las partes de $e(t)$ que no pueden ser controladas por $u(t)$, y un término $z^{-(d+1)} \cdot G(z) / D(z)$ que describe la parte de $e(t)$ en $y(t)$ que puede ser influenciada por $u(t)$. Los polinomios correspondientes son :

$$F(z) = 1 + f_1 \cdot z^{-1} + \dots + f_d \cdot z^{-d} \quad (2.87)$$

$$G(z) = g_0 + g_1 \cdot z^{-1} + \dots + g_{n-1} \cdot z^{-(n-1)} \quad (2.88)$$

Los coeficientes de estos polinomios pueden ser calculados resolviendo la siguiente identidad :

$$C(z) = F(z) \cdot D(z) + z^{-(d+1)} \cdot G(z) \quad (2.89)$$

Nótese que la ecuación 2.19 es igual a la 2.89 con $A = D$ y con $k = d + 1$. Desarrollando la ecuación 2.86 se tiene :

$$A(z) \cdot D(z) \cdot z^{d+1} \cdot Y(z) = B(z) \cdot D(z) \cdot z \cdot U(z) + \lambda \cdot F(z) \cdot A(z) \cdot D(z) \cdot z^{d+1} \cdot E(z) + \lambda \cdot G(z) \cdot A(z) \cdot E(z) \quad (2.90)$$

Realizando el mismo procedimiento que de las ecuaciones 2.55 a 2.57 se obtiene $I(t+1)$, y siguiendo también el de la ecuación 2.58 se obtiene $\frac{\partial I(t+1)}{\partial u(t)} = 0$. Por lo tanto:

$$\left[z^{d+h} \cdot Y(z) - \lambda \cdot F(z) \cdot z^{d+h} \cdot E(z) \right] \cdot b_1 + r \cdot U(z) = 0 \quad (2.91)$$

Despejando $\lambda \cdot E(z)$ en la ecuación 2.85 se obtiene:

$$\lambda \cdot E(z) = \frac{D(z)}{C(z)} \cdot Y(z) - \frac{B(z) \cdot D(z)}{A(z) \cdot C(z)} \cdot z^{-d} \cdot U(z) \quad (2.92)$$

Reemplazando $\lambda \cdot E(z)$ en la ecuación 2.91 con el término derecho de la ecuación 2.92 se tiene:

$$A(z) \cdot C(z) \cdot Y(z) + \frac{r}{b_1} \cdot z^{-(d+h)} \cdot A(z) \cdot C(z) \cdot U(z) = A(z) \cdot F(z) \cdot D(z) \cdot Y(z) - \quad (2.93)$$

$$- B(z) \cdot F(z) \cdot D(z) \cdot z^{-d} \cdot U(z)$$

Agrupando términos en torno a $U(z)$ y $Y(z)$ se tiene:

$$- A(z) \cdot [C(z) - F(z) \cdot D(z)] \cdot Y(z) = \left[B(z) \cdot D(z) \cdot F(z) \cdot z^{-d} + \frac{r}{b_1} \cdot z^{-(d+h)} \cdot A(z) \cdot C(z) \right] \cdot U(z) \quad (2.94)$$

Llegando de aquí a obtenerse la función de transferencia del Controlador de Mínima Varianza Generalizado para procesos con tiempo muerto

G_{RMV1d} :

$$G_{RMV1d}(z) = \frac{U(z)}{Y(z)} = - \frac{A(z) \cdot [C(z) - F(z) \cdot D(z)] \cdot z^{d+h}}{z \cdot B(z) \cdot D(z) \cdot F(z) + \frac{r}{b_1} \cdot A(z) \cdot C(z)} \quad (2.95)$$

Reemplazando la ecuación 2.89 en la ecuación 2.95 se tiene :

$$G_{RMV1d}(z) = \frac{U(z)}{Y(z)} = - \frac{A(z) \cdot G(z)}{z \cdot B(z) \cdot D(z) \cdot F(z) + \frac{r}{b_1} \cdot A(z) \cdot C(z)} \quad (2.96)$$

Al igual que para el caso de los controladores sin tiempo muerto se pueden reconocer los siguientes cuatro casos :

1. $A \neq D$ y $r \neq 0$, para este caso se tiene la función de transferencia G_{RMV1d} expresada en la ecuación 2.96.
2. $A \neq D$ y $r = 0$, para este caso se tiene la función de transferencia G_{RMV2d} , expresada en la ecuación 2.97.

$$G_{RMV2d}(z) = \frac{U(z)}{Y(z)} = \frac{Q(z)}{P(z)} = - \frac{A(z) \cdot G(z)}{z \cdot B(z) \cdot D(z) \cdot F(z)} \quad (2.97)$$

3. $A = D$ y $r \neq 0$, para este caso se tiene la función de transferencia G_{RMV3d} , expresada en la ecuación 2.98.

$$G_{RMV3d}(z) = \frac{U(z)}{Y(z)} = \frac{Q(z)}{P(z)} = - \frac{G(z)}{z \cdot B(z) \cdot F(z) + \frac{r}{b_1} \cdot C(z)} \quad (2.98)$$

4. $A = D$ y $r = 0$, para este caso se tiene la función de transferencia G_{RMV4d} , expresada en la ecuación 2.99. Que es el mismo caso que se analizó en la sección 2.1.3. De donde, se concluye que el algoritmo básico de mínima varianza es un caso particular del algoritmo de mínima varianza generalizado.

$$G_{RMV4d}(z) = \frac{U(z)}{Y(z)} = \frac{Q(z)}{P(z)} = - \frac{G(z)}{z \cdot B(z) \cdot F(z)} \quad (2.99)$$

Propiedades del controlador

- Orden del controlador

	Orden del numerador	Orden del denominador
G_{RMV1d} y G_{RMV2d}	$2n - 1$	$2n + d + 1$ ($d \geq 1$)
G_{RMV3d} y G_{RMV4d}	$n - 1$	$n + d - 1$ ($d \geq 1$)

Tabla 2.3 Orden del controlador.

- Cancelación de polos y ceros

Como para los controladores sin tiempo muerto.

- Estabilidad

La ecuación característica para G_{RMV1d} y G_{RMV3d} es :

$$\left[\frac{r}{b_1} \cdot A(z) + z \cdot B(z) \right] \cdot C(z) = 0 \quad (2.100)$$

y para G_{RMV2d} y G_{RMV4d} es :

$$z \cdot B(z) \cdot D(z) = 0 \quad (2.101)$$

Las ecuaciones anteriores son similares a las del caso del control de mínima varianza para sistemas sin tiempo muerto; por lo tanto, se llegan a las mismas conclusiones que para el caso anterior.

- Variable controlada y (t)

Para $r = 0$, es decir, para los controladores G_{RMV2d} y G_{RMV4d} se tiene :

$$\frac{Y(z)}{\lambda \cdot E(z)} = R(z) \cdot G_{P_r}(z) \cdot \frac{1}{\lambda} = F(z) \quad (2.102)$$

$Y(z)$ es por lo tanto un proceso de media móvil.

$$y(t) = [e(t) + f_1 \cdot e(t-1) + \dots + f_d \cdot e(t-d)] \cdot \lambda \quad (2.103)$$

y la varianza de $y(t)$ es

$$\text{var}[y(t)] = E\{y^2(t)\} = [1 + f_1^2 + \dots + f_d^2] \cdot \lambda^2 \quad (2.104)$$

De la anterior ecuación se puede concluir que; a mayor tiempo muerto, mayor es la varianza de la variable controlada.

2.2.4 CONTROLADORES DE MÍNIMA VARIANZA SIN OFFSET [1]

Para evitar offsets de la variable controlada ya sea para perturbaciones externas constantes o para cambios constantes en el valor de la señal de referencia, el controlador debe satisfacer la siguiente condición :

$$\lim_{z \rightarrow 1} G_r(z) = \infty \quad (2.105)$$

Como este no es el caso de los controladores de mínima varianza que se obtuvieron en las dos secciones anteriores, es decir, para procesos de acción proporcional, es conveniente modificar los controladores.

Existen varios métodos para realizar estas modificaciones, sin embargo se escoge el método de *minimización del error de control*, ya que es el que más se ajusta a los requerimientos del desarrollo que se ha venido haciendo de los controladores de mínima varianza.

Los controladores de mínima varianza de las secciones 2.2.2 y 2.2.3 se obtuvieron de la asunción de que la variable de referencia $w(t) = 0$, y por lo tanto $y(t) = -e(t)$. Ahora el criterio de costo se modifica de manera que se toma en cuenta a la variable de referencia $w(t)$ de la siguiente manera :

$$I(t+1) = E \left\{ \left[y(t+d+1) - w(t) \right]^2 + r \cdot \left[u(t) - u_w(t) \right]^2 \right\} \quad (2.106)$$

De tal forma que la varianza alrededor del punto de operación $[w(t); u_w(t)]$ diferente de cero es minimizada. En la ecuación 2.106 se tiene que :

$$u_w(t) = \frac{A(1)}{B(1)} \cdot w(t) = \frac{1}{K_p} \cdot w(t) \quad (2.107)$$

El valor de $u(t)$ para $y(t) = w(t)$, se presenta para el caso de offset cero. Partiendo de los resultados obtenidos en las secciones 2.2.2 y 2.2.3 se llega al controlador de mínima varianza modificado, en el cual la variable de control $u(t)$ es :

$$U(z) = - \frac{A(z) \cdot G(z)}{z \cdot B(z) \cdot D(z) \cdot F(z) + \frac{r}{b_1} \cdot A(z) \cdot C(z)} \cdot Y(z) + \frac{A(z) \cdot C(z)}{z \cdot B(z) \cdot D(z) \cdot F(z) + \frac{r}{b_1} \cdot A(z) \cdot C(z)} \cdot \left[1 + \frac{r}{b_1} \cdot \frac{1}{K_p} \right] \cdot W(z) \quad (2.108)$$

2.3 ALGORITMO PARA EL CONTROL DE MÍNIMA VARIANZA

2.3.1 ALGORITMO PARA EL CÁLCULO DE LOS COEFICIENTES DE LOS POLINOMIOS F Y G

Los valores de los coeficientes de los polinomios F y G se los encuentra resolviendo la identidad formulada en la ecuación 2.89; es decir :

$$C(z) = F(z) \cdot D(z) + z^{-(d+1)} \cdot G(z)$$

El orden de los polinomios F y G depende de los valores del retardo y del orden de los polinomios A , B , C y D respectivamente.

1. Si el valor del retardo d es cero, la ecuación 2.89 toma la siguiente forma:

$$C(z) = F(z) \cdot D(z) + z^{-1} \cdot G(z) \quad (2.109)$$

donde :

$$\begin{aligned} F(z) &= 1 \\ G(z) &= g_0 + g_1 \cdot z^{-1} + \dots + g_{n-1} \cdot z^{-(n-1)} \end{aligned} \quad (2.110)$$

A , B , C y D tienen la forma expresada en las ecuaciones 2.48 y 2.49. Por lo que, la identidad a resolver es :

$$C(z) = D(z) + z^{-1} \cdot G(z) \quad (2.111)$$

Se prevé que el máximo orden de los polinomios A , B , C y D a utilizarse durante el desarrollo del programa computacional será de cuarto orden ($n=4$), razón por la cual se calculan los valores de los coeficientes de $G(z)$

solo hasta el tercer orden ($n - 1 = 3$). Dando como resultado las siguientes ecuaciones :

$$\begin{aligned}g_0 &= c_1 - d_1 \\g_1 &= c_2 - d_2 \\g_2 &= c_3 - d_3 \\g_3 &= c_4 - d_4\end{aligned}\tag{2.112}$$

2. Si el valor del retardo d es uno, la ecuación 2.89 toma la siguiente forma:

$$C(z) = F(z) \cdot D(z) + z^{-2} \cdot G(z)\tag{2.113}$$

donde :

$$\begin{aligned}F(z) &= 1 + f_1 \cdot z^{-1} \\G(z) &= g_0 + g_1 \cdot z^{-1} + \dots + g_{n-1} \cdot z^{-(n-1)}\end{aligned}\tag{2.114}$$

Se tienen las siguientes ecuaciones :

$$\begin{aligned}f_1 &= c_1 - d_1 \\g_0 &= c_2 - d_2 - f_1 \cdot d_1 \\g_1 &= c_3 - d_3 - f_1 \cdot d_2 \\g_2 &= c_4 - d_4 - f_1 \cdot d_3 \\g_3 &= -f_1 \cdot d_4\end{aligned}\tag{2.115}$$

3. Si el valor del retardo d es dos, la ecuación 2.89 toma la siguiente forma :

$$C(z) = F(z) \cdot D(z) + z^{-3} \cdot G(z)\tag{2.116}$$

donde :

$$\begin{aligned} F(z) &= 1 + f_1 \cdot z^{-1} + f_2 \cdot z^{-2} \\ G(z) &= g_0 + g_1 \cdot z^{-1} + \dots + g_{n-1} \cdot z^{-(n-1)} \end{aligned} \quad (2.117)$$

Se tienen las siguientes ecuaciones :

$$\begin{aligned} f_1 &= c_1 - d_1 \\ f_2 &= c_2 - d_2 - f_1 \cdot d_1 \\ g_0 &= c_3 - d_3 - f_1 \cdot d_2 - f_2 \cdot d_1 \\ g_1 &= c_4 - d_4 - f_1 \cdot d_3 - f_2 \cdot d_2 \\ g_2 &= -f_1 \cdot d_4 - f_2 \cdot d_3 \\ g_3 &= -f_2 \cdot d_4 \end{aligned} \quad (2.118)$$

2.3.2 ALGORITMO PARA EL CÁLCULO DE LA SEÑAL DE CONTROL $u(t)$ PARA EL CASO DE REGULACIÓN

Para la regulación se tienen dos casos : sistemas sin tiempo muerto y sistemas con tiempo muerto.

De la misma manera, para cada uno de los tipos de sistemas anteriores se puede dar el caso de que los polinomios A y D sean iguales o que sean diferentes.

En las secciones 2.2.2 y 2.2.3 se encontraron las funciones de transferencia para cada caso. Estas funciones están expresadas en las ecuaciones 2.63 y 2.96 respectivamente.

Analizando ambas ecuaciones se llega a la conclusión de que la ecuación 2.96 es la más general; ya que, cubre todos los posibles casos que se presentan en regulación.

Reescribiendo la ecuación 2.96 se tiene :

$$\left[B(z) \cdot D(z) \cdot F(z) + \frac{r}{b_1} \cdot z^{-1} \cdot A(z) \cdot C(z) \right] \cdot U(z) = [z^{-1} \cdot A(z) \cdot G(z)] \cdot Y(z) \quad (2.119)$$

Escribiendo la ecuación anterior en función de los coeficientes de los polinomios y realizando el producto entre ellos (polinomios de orden $n = 4$), se tiene :

$$u(t) = \frac{1}{b_1} \cdot \left\{ \begin{array}{l} - \left[TU1 \cdot u(t-1) + TU2 \cdot u(t-2) + TU3 \cdot u(t-3) + TU4 \cdot u(t-4) + TU5 \cdot u(t-5) + \right. \\ \left. TU6 \cdot u(t-6) + TU7 \cdot u(t-7) + TU8 \cdot u(t-8) + TU9 \cdot u(t-9) + TU10 \cdot u(t-10) \right] \\ - \left[g_0 \cdot y(t-1) + AG1 \cdot y(t-2) + AG2 \cdot y(t-3) + AG3 \cdot y(t-4) + AG4 \cdot y(t-5) + \right. \\ \left. + AG5 \cdot y(t-6) + AG6 \cdot y(t-7) + AG7 \cdot y(t-8) \right] \end{array} \right\} \quad (2.120)$$

Donde, los coeficientes TU_n corresponden a los términos obtenidos del producto de $[B(z) \cdot D(z) \cdot F(z) + r/b_1 \cdot z^{-1} \cdot A(z) \cdot C(z)] \cdot U(z)$ y los coeficientes AG_n corresponden a los términos obtenidos del producto de $z^{-1} \cdot A(z) \cdot G(z) \cdot Y(z)$.

Los coeficientes TU_n se calculan de la siguiente manera :

$$\begin{aligned} TU1 &= (BDF1 + \text{factorrb}) \\ TU2 &= (BDF2 + \text{factorrb} * AC1) \\ TU3 &= (BDF3 + \text{factorrb} * AC2) \\ TU4 &= (BDF4 + \text{factorrb} * AC3) \\ TU5 &= (BDF5 + \text{factorrb} * AC4) \\ TU6 &= (BDF6 + \text{factorrb} * AC5) \\ TU7 &= (BDF7 + \text{factorrb} * AC6) \\ TU8 &= (BDF8 + \text{factorrb} * AC7) \\ TU9 &= (BDF9 + \text{factorrb} * AC8) \\ TU10 &= (BDF10) \end{aligned} \quad (2.121)$$

Donde los términos BDF_n , AC_n y el factor r_b se calculan así :

$$\begin{aligned}
 BDF1 &= BD1 + b0 * f1 \\
 BDF2 &= BD2 + BD1 * f1 + b0 * f2 \\
 BDF3 &= BD3 + BD2 * f1 + BD1 * f2 \\
 BDF4 &= BD4 + BD3 * f1 + BD2 * f2 \\
 BDF5 &= BD5 + BD4 * f1 + BD3 * f2 \\
 BDF6 &= BD6 + BD5 * f1 + BD4 * f2 \\
 BDF7 &= BD7 + BD6 * f1 + BD5 * f2 \\
 BDF8 &= BD8 + BD7 * f1 + BD6 * f2 \\
 BDF9 &= BD8 * f1 + BD7 * f2 \\
 BDF10 &= BD8 * f2
 \end{aligned}
 \tag{2.122}$$

$$\begin{aligned}
 BD1 &= b1 + b0 * d1 \\
 BD2 &= b2 + b1 * d1 + b0 * d2 \\
 BD3 &= b3 + b2 * d1 + b1 * d2 + b0 * d3 \\
 BD4 &= b4 + b3 * d1 + b2 * d2 + b1 * d3 + b0 * d4 \\
 BD5 &= b4 * d1 + b3 * d2 + b2 * d3 + b1 * d4 \\
 BD6 &= b4 * d2 + b3 * d3 + b2 * d4 \\
 BD7 &= b4 * d3 + b3 * d4 \\
 BD8 &= b4 * d4
 \end{aligned}
 \tag{2.123}$$

$$\begin{aligned}
 AC1 &= a1 + c1 \\
 AC2 &= a2 + a1 * c1 + c2 \\
 AC3 &= a3 + a2 * c1 + a1 * c2 + c3 \\
 AC4 &= a4 + a3 * c1 + a2 * c2 + a1 * c3 + c4 \\
 AC5 &= a4 * c1 + a3 * c2 + a2 * c3 + a1 * c4 \\
 AC6 &= a4 * c2 + a3 * c3 + a2 * c4 \\
 AC7 &= a4 * c3 + a3 * c4 \\
 AC8 &= a4 * c4
 \end{aligned}
 \tag{2.124}$$

$$\text{factorrb} = \frac{r}{b1} \quad \text{o} \quad \text{factorrb} = \frac{r}{b0} \quad (2.125)$$

Y los coeficientes AG_n se calculan de la siguiente manera :

$$\begin{aligned} AG1 &= a1 * g0 + g1 \\ AG2 &= a2 * g0 + a1 * g1 + g2 \\ AG3 &= a3 * g0 + a2 * g1 + a1 * g2 + g3 \\ AG4 &= a4 * g0 + a3 * g1 + a2 * g2 + a1 * g3 \\ AG5 &= a4 * g1 + a3 * g2 + a2 * g3 \\ AG6 &= a4 * g2 + a3 * g3 \\ AG7 &= a4 * g3 \end{aligned} \quad (2.126)$$

2.3.3 ALGORITMO PARA EL CÁLCULO DE LA SEÑAL DE CONTROL $u(t)$ PARA EL CASO DE SEGUIMIENTO

Para seguimiento se tienen los mismos casos que para regulación. En seguimiento se aumenta una variable al sistema que es la señal de referencia $w(t)$.

Con la inclusión de la variable $w(t)$ la función de transferencia del controlador de mínima varianza se transforma en la ecuación 2.108.

Para evitar el offset o error en estado estable de la salida del sistema se multiplica a la señal de referencia por el inverso de la ganancia de la planta (K_p). Este valor se lo calcula de la siguiente manera :

$$\frac{1}{Kp} = \frac{A(1)}{B(1)} = \frac{1 + \sum_{i=1}^4 a_i}{\sum_{i=0}^4 b_i} \quad (2.127)$$

Reescribiendo la ecuación 2.108 se tiene :

$$\left[z \cdot B(z) \cdot D(z) \cdot F(z) + \frac{r}{b_1} \cdot A(z) \cdot C(z) \right] \cdot U(z) = - A(z) \cdot G(z) \cdot Y(z) + \\ + A(z) \cdot C(z) \cdot \left[1 + \frac{r}{b_1} \cdot \frac{1}{Kp} \right] \cdot W(z) \quad (2.128)$$

Como se puede observar, el término de la izquierda la igualdad de la ecuación anterior se puede encontrar utilizando las ecuaciones 2.121, 2.122, 2.123, 2.124, y 2.125. Y el primer término de la derecha de la igualdad se puede calcular utilizando la ecuación 2.126. El segundo término de la parte izquierda de la igualdad se puede calcular de la siguiente manera :

$$\begin{aligned} AC1 &= a1 + c1 \\ AC2 &= a2 + a1 * c1 + c2 \\ AC3 &= a3 + a2 * c1 + a1 * c2 + c3 \\ AC4 &= a4 + a3 * c1 + a2 * c2 + a1 * c3 + c4 \\ AC5 &= a4 * c1 + a3 * c2 + a2 * c3 + a1 * c4 \\ AC6 &= a4 * c2 + a3 * c3 + a2 * c4 \\ AC7 &= a4 * c3 + a3 * c4 \\ AC8 &= a4 * c4 \end{aligned} \quad (2.129)$$

Por último, escribiendo la ecuación 2.128 en función de los coeficientes calculados anteriormente se tiene :

$$u(t) = \frac{1}{b_0} \cdot \left\{ \begin{array}{l} - \left[TU1 \cdot u(t-1) + TU2 \cdot u(t-2) + TU3 \cdot u(t-3) + TU4 \cdot u(t-4) + TU5 \cdot u(t-5) + \right. \\ \left. TU6 \cdot u(t-6) + TU7 \cdot u(t-7) + TU8 \cdot u(t-8) + TU9 \cdot u(t-9) + TU10 \cdot u(t-10) \right] \\ - \left[g_0 \cdot y(t-1) + AG1 \cdot y(t-2) + AG2 \cdot y(t-3) + AG3 \cdot y(t-4) + AG4 \cdot y(t-5) + \right. \\ \left. + AG5 \cdot y(t-6) + AG6 \cdot y(t-7) + AG7 \cdot y(t-8) \right] + \\ + \left[1 + \frac{r}{b1} \cdot \frac{1}{Kp} \right] \cdot \left[w(t-1) + AC1 \cdot w(t-2) + AC2 \cdot w(t-3) + AC3 \cdot w(t-4) + \right. \\ \left. AC4 \cdot w(t-5) + AC5 \cdot w(t-6) + AC6 \cdot w(t-7) + AC7 \cdot w(t-8) + \right. \\ \left. AC8 \cdot w(t-9) \right] \end{array} \right\}$$

(2.130)

2.4 IMPLEMENTACIÓN Y APLICACIONES

Para la implementación práctica del *Control de Mínima Varianza*, en primer lugar, se establecen los modos de funcionamiento. Como se dijo en la introducción (sección 1.1) se trabaja en el modo de simulación y en tiempo real. En segundo lugar, se determina la forma en que se realiza el control, es decir; regulación y seguimiento. En tercer lugar, como se puede observar en la sección 2.2, se tienen formas diferentes de las funciones de transferencia de la planta como del proceso de ruido. Debido a esto, se tienen principalmente los siguientes casos :

- Polinomios A y D diferentes, y polinomios A y D iguales.
- Sistemas sin tiempo muerto y sistemas con tiempo muerto.

Para las funciones de transferencia tanto de la planta como del proceso de ruido se decidió que las ecuaciones del sistema sean de cuarto orden como máximo debido a que para control se manejan modelos aproximados de bajo orden. Para los sistemas con tiempo muerto se decidió que el número máximo de retardos debe ser de dos ($d = 2$), debido a que para retardos mayores a dos, la varianza de la salida se hace tan grande que no tiene sentido el aplicar el control en estos casos, ya que no se aprecia una gran mejoría en la varianza de la salida, respecto de los sistemas sin control de mínima varianza.

Una vez que se han determinado las características generales del programa computacional que realiza el control de mínima varianza, se procede a su implementación, la misma que se realiza en lenguaje de programación *Visual Basic*, tanto para simulación como para tiempo real, con la excepción de que en tiempo real se utiliza una librería (DLL = Data Link Library) escrita en Lenguaje de programación C ++ para manejar la tarjeta de adquisición de datos DAS - 128. [10], [11]

Tanto para el modo de simulación como en tiempo real se deben ingresar los coeficientes de los polinomios A, B, C y D; así como la forma de control, el tipo de sistema, los retardos (si es el caso), el factor de ponderación y la varianza del ruido.

Para la aplicación en tiempo real adicionalmente se utilizan circuitos eléctricos para representar plantas de primero y segundo orden, sobre las cuales se hacen las pruebas. El proceso de ruido se lo genera en el computador y se lo aplica a la salida de la planta. El esquema de como se realiza la implementación en tiempo real se puede observar en la figura 2.7.

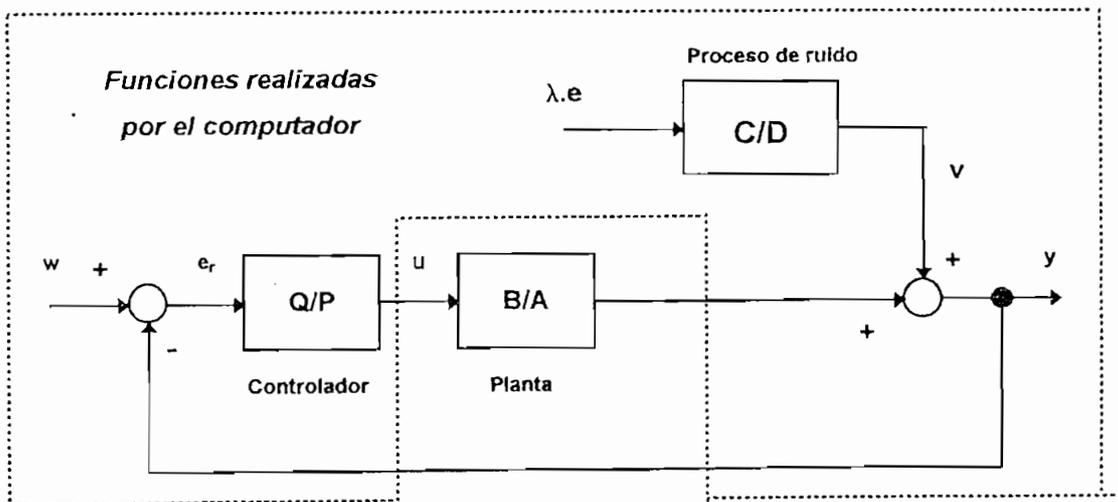


Figura 2.7 . Esquema de la implementación en tiempo real.

Puesto que la implementación en tiempo real se hace con prototipos circuitales conocidos no se requiere realizar la identificación de parámetros, sino más bien, la discretización del modelo continuo. Por esta razón se incluye en el programa una opción de trabajo para acceder al **CC** sin tener que salir del mismo. Si fuera necesario en algún caso estimar parámetros,

previamente es necesario utilizar un paquete de software para este propósito. [12]

La implementación debe ser capaz de manejar gráficos de la salida sin control, de la salida con control, de la señal de control, y del proceso estocástico. Siendo lo suficientemente versátil como para establecer comparaciones entre ellos. Así mismo, debe tener la posibilidad de imprimir los resultados tanto de los gráficos, como un reporte de los datos más importantes del sistema.

En cuanto a las aplicaciones en la presente tesis se ilustran las mismas con la utilización de circuitos electrónicos descritos anteriormente. Sin embargo esta técnica puede ser usada en procesos industriales con niveles de ruido elevados, aunque el objetivo de esta tesis no es trabajar con los mismos. Para mayor información de las aplicaciones industriales del control de mínima varianza referirse al libro de Karl Åstrom, "Introduction to Stochastic Control Theory". [2]

CAPÍTULO III : PROGRAMA PARA EL CONTROL DE MÍNIMA VARIANZA

3.1 Programa principal

3.2 Rutinas para la simulación de plantas, salida y ruido

3.3 Rutinas para tiempo real (adquisición de datos)

3.3.1 Tarjeta de adquisición de datos DAS - 128

3.3.2 Librería para utilización de la DAS - 128

3.4 Rutina de graficación

3.5 Rutina para el control de mínima varianza

3. PROGRAMA PARA EL CONTROL DE MÍNIMA VARIANZA

3.1 PROGRAMA PRINCIPAL

El programa para el control de mínima varianza está estructurado de la siguiente manera :

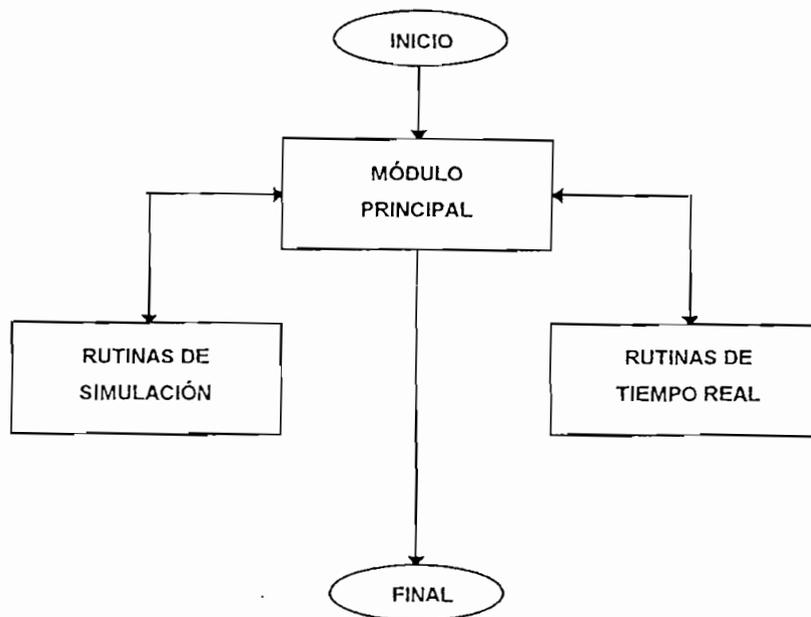


Figura 3.1 Esquema generalizado de la estructura del programa.

El módulo principal está formado por dos pantallas: la primera, la de presentación que aparece en el instante de iniciar el programa, y la segunda, la pantalla principal que aparece luego de la de presentación una vez que ha iniciado el programa.

La pantalla de presentación puede ser observada en la figura 3.2 y la pantalla principal puede ser observada en la figura 3.3.



Figura 3.2 Pantalla de presentación.

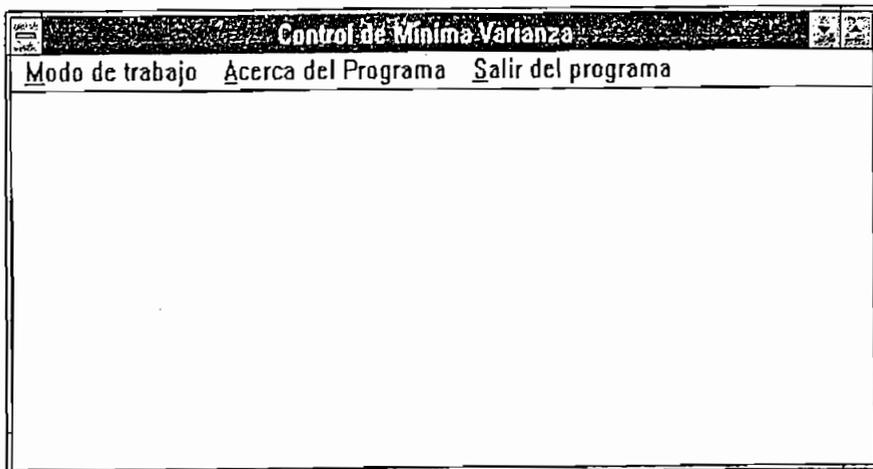


Figura 3.3 Pantalla principal.

La pantalla principal tiene un menú, el mismo que permite escoger el modo de trabajo, encontrar información acerca del programa, y salir del mismo si se desea.

En la figura 3.4 se puede observar la pantalla Modo de Trabajo que aparece al hacer clic en el ítem del mismo nombre.



Figura 3.4 .Pantalla para seleccionar el Modo de Trabajo.

Cuando se hace un clic en la opción Acerca del Programa se visualiza la pantalla mostrada en la figura 3.5.

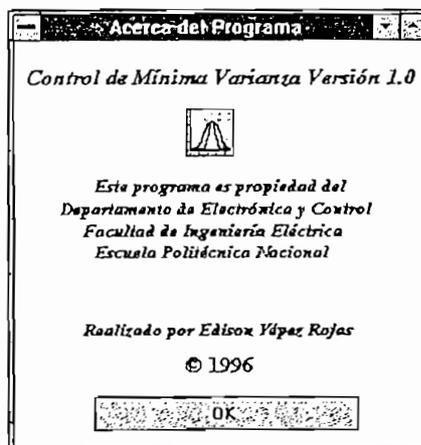


Figura 3.5. Pantalla de Información del Programa.

Cuando se selecciona la opción Simulación de la pantalla Modo de Trabajo se ingresa en el ambiente de **Modo de Simulación** que será discutido con más detalle en la sección 3.2. De la misma manera al seleccionar la opción Tiempo Real de la misma pantalla, se ingresa al **Modo de Tiempo Real** que será explicado en más detalle en la sección 3.3.

3.2 RUTINAS PARA LA SIMULACIÓN DE PLANTAS, SALIDA Y RUIDO

El Modo de Simulación está estructurado de una pantalla principal desde la cual se llaman a otras pantallas auxiliares que sirven para cargar datos, para el ingreso de datos nuevos, para guardar datos, para visualizar resultados y para invocar herramientas auxiliares.

Para dar una mejor idea de la estructura antes descrita se puede observar en la figura 3.6 un esquema del modo de simulación.

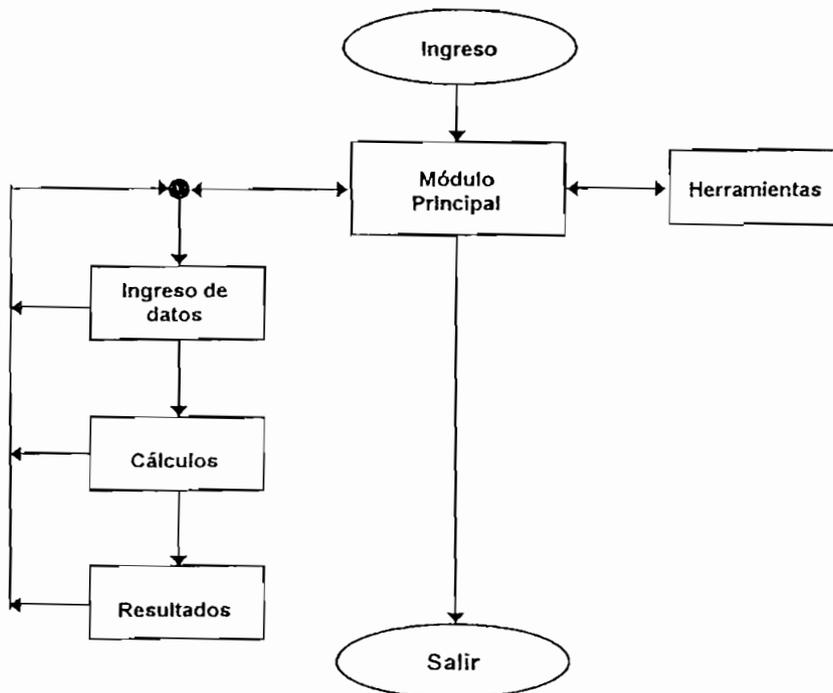


FIGURA 3.6 . Esquema de la estructura de funcionamiento del Modo de Simulación

Una vez que se ha ingresado al Modo de Simulación se visualiza la pantalla principal desde la cual se accede a través de su menú principal a todas las opciones del mismo. La pantalla principal se puede observar en la figura 3.7.

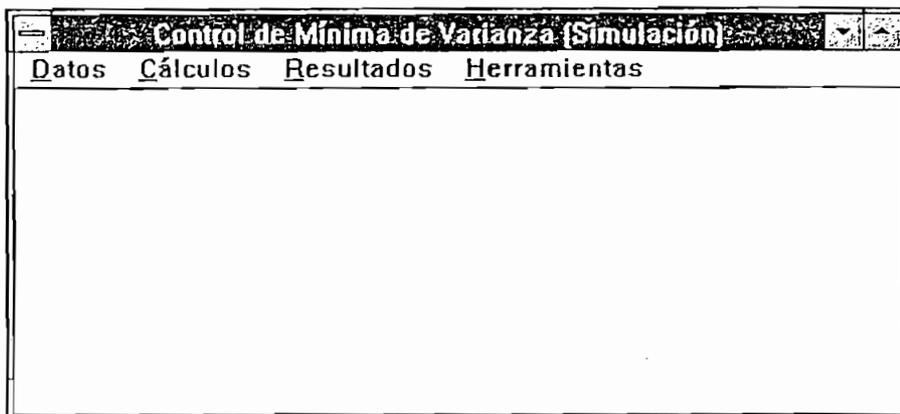


Figura 3.7. Pantalla principal del Modo de Simulación.

Esta pantalla contiene los siguientes submenús, el de Datos, que a su vez contiene el submenú Datos Nuevos que se pueden observar en la figura 3.8, el de Resultados que se muestra en la figura 3.9 y el de Herramientas que se muestra en la figura 3.10.

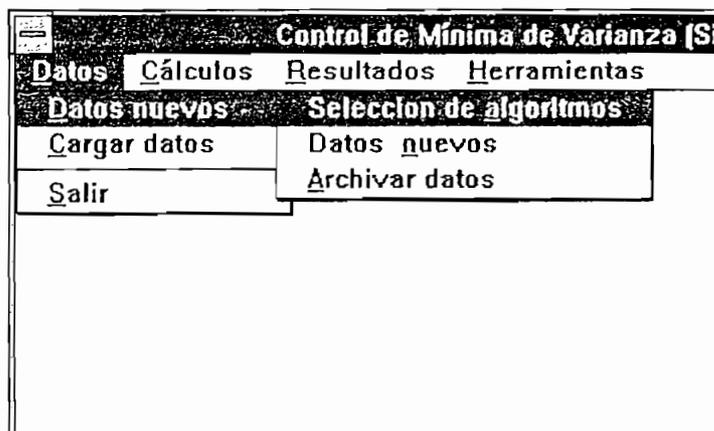


Figura 3.8. Submenús del ítem Datos y Datos Nuevos.



Figura 3.9. Submenú del ítem Resultados

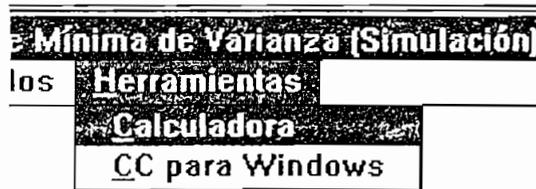


Figura 3.10. Submenú del ítem Herramientas

Cuando se hace un clic en la opción Selección de algoritmos del submenú Datos nuevos se visualiza la pantalla mostrada en la figura 3.11, la misma que permite escoger el tipo de algoritmo que se va a utilizar. Si se hace un clic en la siguiente opción; es decir, en Datos nuevos se visualiza la pantalla de ingreso de datos que se puede observar en la figura 3.12, a través de la cual se ingresa el orden del sistema, los valores de los coeficientes de los polinomios A, B, C, y D, el factor de ponderación, el retardo, y la varianza del ruido. Una vez que se han ingresado los datos del sistema con el cual se va a trabajar, esta información puede ser guardada en un archivo con extensión .CMV, esto se lo hace haciendo un clic en la tercera opción de este submenú; es decir, en Archivar datos la misma que muestra la pantalla de la figura 3.13.

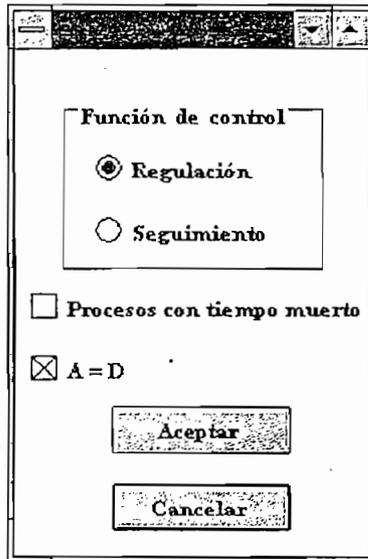


Figura 3.11. Pantalla de Selección de algoritmos

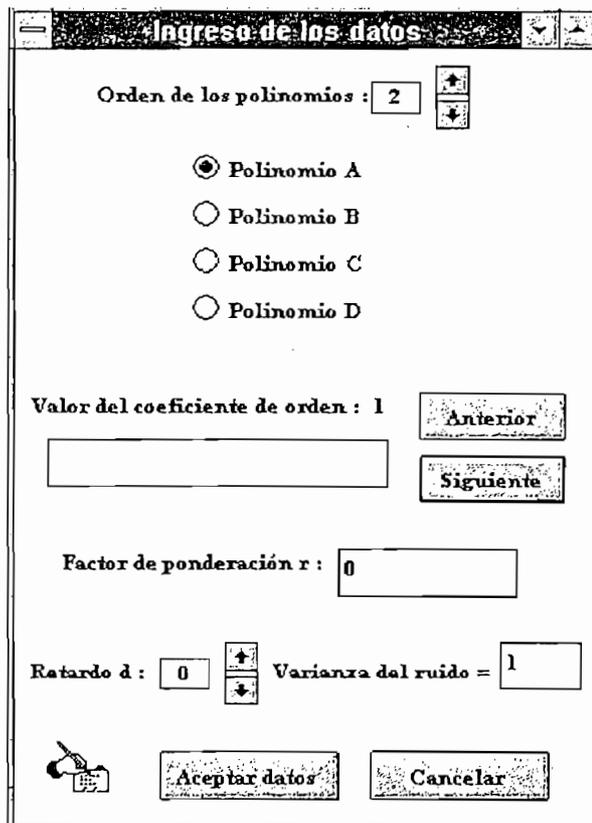


Figura 3.12. Pantalla de Ingreso de datos

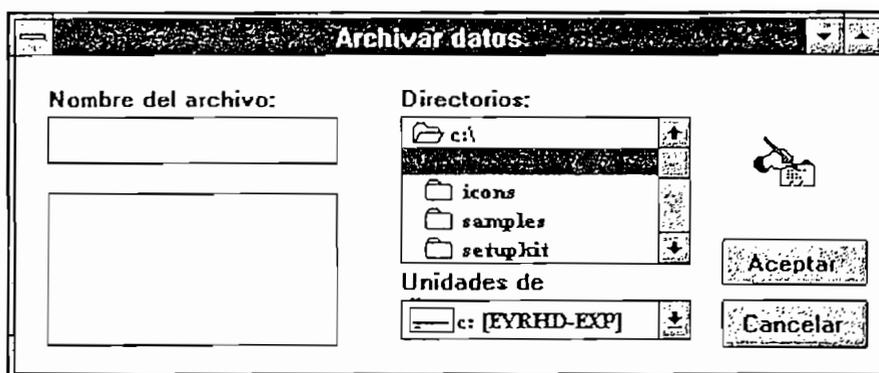


Figura 3.13. Pantalla para Archivar datos

Cuando se hace un clic en la opción Cargar datos del ítem Datos, se visualiza la pantalla de la figura 3.14, la misma que permite cargar en memoria los archivos con extensión .CMV generados anteriormente.

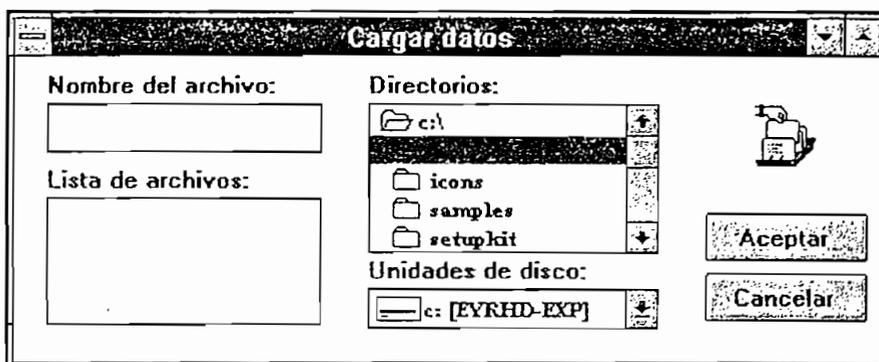


Figura 3.14. Pantalla para Cargar datos

Una vez que se encuentran cargados en memoria los valores de las variables del sistema, se procede a realizar los cálculos de los coeficientes de las ecuaciones de diferencias que permitirán implementar los algoritmos a implementarse. Esto se lo realiza haciendo un clic en el ítem Cálculos

del menú principal. Una vez realizados los cálculos se pueden ver los resultados. Esto se lo realiza haciendo un clic en el ítem Resultados, visualizándose el submenú de la figura 3.9. Al hacer un clic en la opción Reporte se visualiza la pantalla mostrada en la figura 3.15. La misma que permite verificar los datos del sistema y observar los resultados de los cálculos hechos anteriormente. En esta pantalla se encuentra un submenú de opciones el mismo que se muestra en la figura 3.16 y que permite salir de esta pantalla a la anterior y además permite la impresión del reporte a través de la impresora.

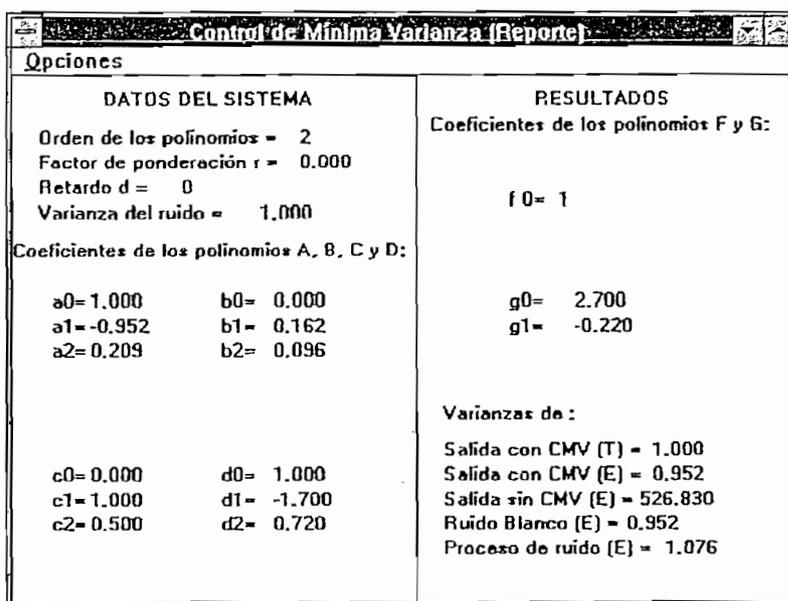


Figura 3.15. Pantalla de Reporte de datos

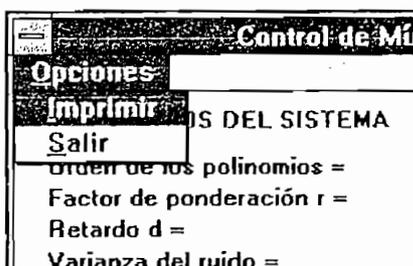


Figura 3.16. Submenú de Opciones de la pantalla de Reporte.

Si se desea ver gráficamente los resultados del control de mínima varianza se hace un clic en la opción Gráficos, visualizándose la pantalla de Menú de gráficos (figura 3.17) en la cual se escogen los gráficos que se desean ver. Una vez que se escogen los gráficos a visualizarse se procede a invocar a la pantalla de gráficos, la misma que se muestra en la figura 3.18.

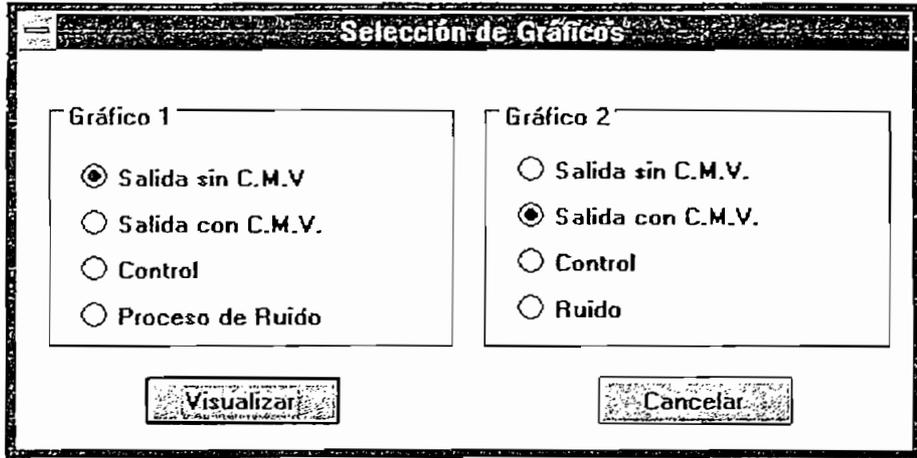


Figura 3.17. Pantalla de Menú de gráficos

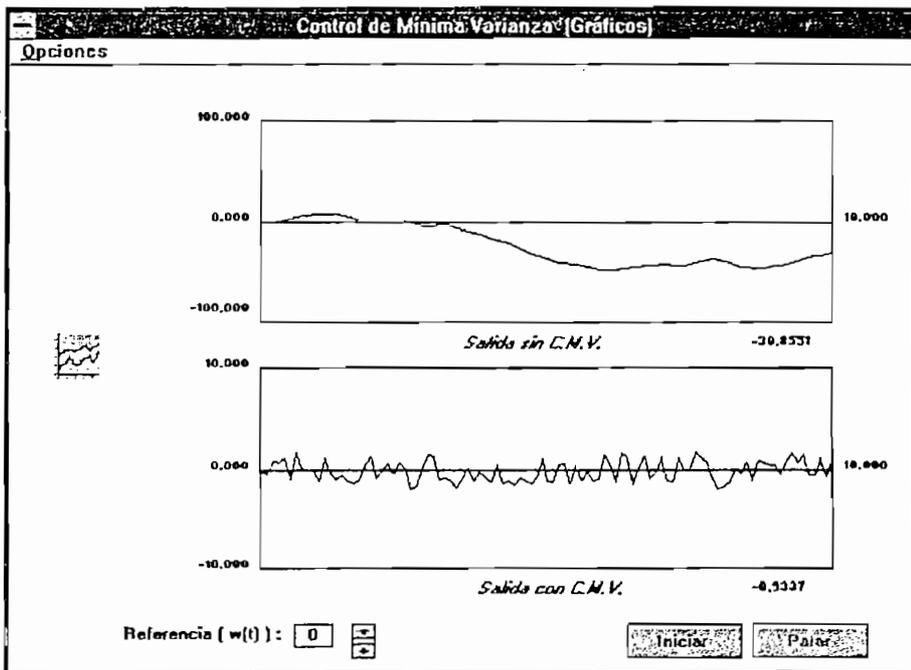


Figura 3.18. Pantalla de Gráficos

Desde la pantalla de Gráficos se accede a través de su menú a diferentes opciones las cuales se pueden observar en la figura 3.19

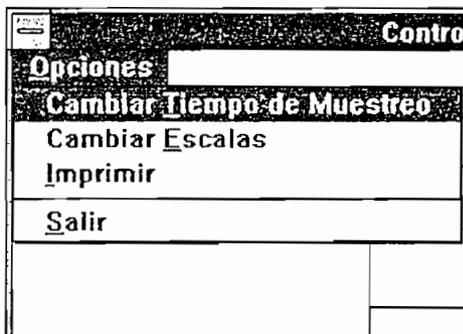


Figura 3.19. Submenú de Opciones de la pantalla de Gráficos

La opción Cambiar Tiempo de Muestreo permite modificar los intervalos de tiempo entre los cuales se hace el cálculo de los valores de las variables a graficarse y se dibuja una línea entre el punto anterior y el nuevo punto calculado. Esta pantalla se muestra en la figura 3.20.

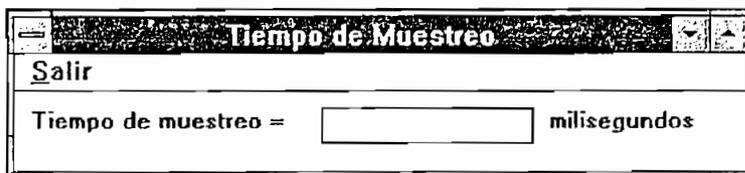


Figura 3.20. Pantalla de Cambio de Tiempo de muestreo

La opción Cambiar Escalas permite acceder a la pantalla mostrada en la figura 3.21, la misma que tiene dos opciones. La primera, de Escala automática que setea los límites máximos y mínimos en los ejes X y Y con valores predeterminados que serán recalculados cada vez que se de un barrido completo de la pantalla de gráficos, y la segunda, de Escala manual que permite al usuario fijar los límites de las escalas que no cambian cuando se barre completamente la pantalla de gráficos.

Cambio de Escalas

Salir

Valores de las escalas del gráfico 1 :

Máximo en X :

Mínimo en X :

Máximo en Y :

Mínimo en Y :

Valores de las escalas del gráfico 2 :

Máximo en X :

Mínimo en X :

Máximo en Y :

Mínimo en Y :

Escala automática Escala manual

Figura 3.21. Pantalla de Cambio de escalas

La opción Imprimir al igual que para el caso del reporte de datos permite realizar una impresión de los gráficos que se encuentran visualizándose en ese instante en la pantalla.

Las rutinas para el cálculo de las señales de salida sin control, las de la salida con control, las de la señal de control y la del ruido blanco se tratarán en las secciones 3.4 y 3.5.

3.3 RUTINAS PARA TIEMPO REAL (ADQUISICIÓN DE DATOS)

El Modo de Tiempo Real está estructurado de idéntica manera que el Modo de Simulación a nivel de pantallas y de subrutinas. Existen dos diferencias entre ambos modos de trabajo. La primera, es el título en la parte superior de la pantalla como se puede observar en la figura 3.22 que muestra la pantalla principal de este modo de trabajo.

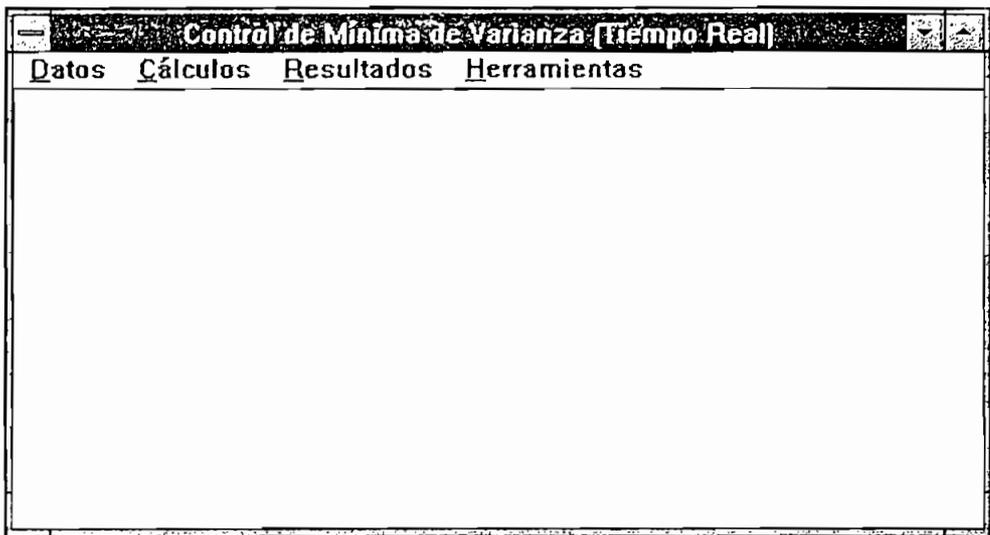


Figura 3.22. Pantalla principal del Modo de Tiempo Real.

La segunda y la más importante es que, en el modo de simulación la planta se la implementa a nivel de software; es decir, su salida sin control y con control se encuentran matemáticamente a partir de los modelos de la planta y del proceso de ruido. En cambio en tiempo real las salidas sin control y con control son el resultado del muestreo que se hace de la salida de los prototipos circuitales implementados, a los cuales ingresa la señal de control.

Todo el proceso de muestreo de la salida, y entrega de la señal de control se lo realiza a través de la tarjeta de adquisición de datos DAS - 128.

3.3.1 TARJETA DE ADQUISICIÓN DE DATOS DAS - 128 [11]

La tarjeta de adquisición de datos DAS - 128 tiene las siguientes características :

- Está instalada en un slot de ocho bits para tarjetas de extensión.
- Funciona en computadores personales de XT en adelante compatibles con IBM.
- Las fuentes de corriente continua necesarias para su operación se toman del slot, por lo tanto no se requiere de fuentes de alimentación externas. La tarjeta provee salidas de la fuente de + 5 VCD para usos externos.
- La tarjeta utiliza 15 direcciones consecutivas para direccionar los puertos disponibles, a partir de la dirección base asignada por medio del dipswitch existente. Las direcciones base que se le pueden dar a la tarjeta son : 200h, 210h, 220h, 230h, 300h, 310h, 320h y 330h.
- Tiene los siguientes puertos disponibles :
 - 8 entradas analógicas con 8 bits de resolución
 - 8 salidas analógicas con 8 bits de resolución
 - 7 puertos digitales de 8 bits de entradas
 - 7 puertos digitales de 8 bits de salidas
- Una vez definida la dirección base, las direcciones de acceso a los puertos de entradas y salidas analógicas disponibles en la tarjeta se ilustran a continuación :

<i>DIRECCIÓN</i>	<i>PUERTO DE ENTRADA</i>	<i>PUERTO DE SALIDA</i>
xx0h	Vi0	Vo0
xx1h	Vi1	Vo1
xx2h	Vi2	Vo2
xx3h	Vi3	Vo3
xx4h	Vi4	Vo4
xx5h	Vi5	Vo5
xx6h	Vi6	Vo6
xx7h	Vi7	Vo7

Tabla 3.1. Direcciones de los puertos de entrada y salida analógicos a partir de la dirección base xx0.

3.3.2 LIBRERÍA PARA UTILIZACIÓN DE LA DAS - 128 [10]

La librería para la adquisición de datos permite acceder a cualquier entrada o salida analógica de la tarjeta DAS - 128. Esta librería es una DLL; es decir, una Data Link Library. Es utilizada para medir los valores de la salida de la planta y para enviar los valores de las señales de control y del proceso de ruido. Dentro del programa de control de mínima varianza se encuentra declarada en el módulo global, las sentencias de declaración son:

Declare Function Medir Lib "loport.dll" (ByVal PUERTOE As Integer, ByVal MEDIDA As Integer) As Integer

Declare Function Enviar Lib "loport.dll" (ByVal PUERTOS As Integer, ByVal VCONTROL As Integer) As Integer

Como se puede observar, las instrucciones que realizan la lectura o escritura en la dirección de memoria de la entrada o salida utilizadas son declaradas como funciones. Por lo que, cuando se desea medir una entrada o escribir en una salida se invoca a la librería como una función cualquiera de Visual Basic.

A continuación se muestra un ejemplo de como se invocan a las funciones de la librería, tanto para lectura como para escritura.

m = Medir (&H300, Medida) (para leer un dato de la entrada Vi0 y asignarlo a la variable "m")

zz = Enviar (&H302, 0) (para escribir un cero en la salida Vo2)

3.4 RUTINA DE GRAFICACIÓN

La rutina de graficación se ejecuta cada vez que pasa un período de muestreo. Desde la misma se invoca a la rutina de cálculo de la señal de control y de la salida controlada, y a la rutina para generar la salida sin control dependiendo del caso. La señal del proceso de ruido se la genera desde la misma rutina de graficación.

Una descripción a manera de diagramas de bloques de como funciona la rutina de graficación se puede observar en la figura 3.23.

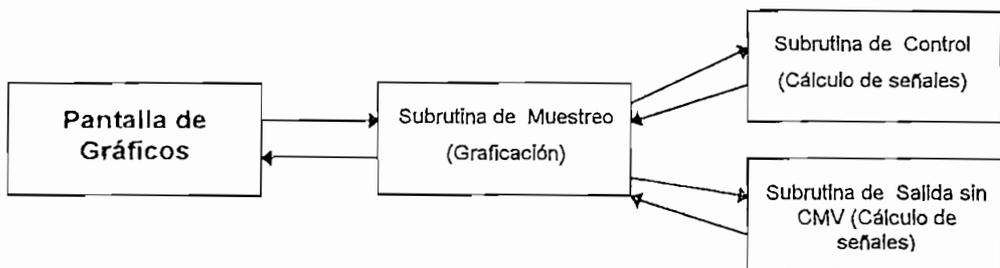


Figura 3.23. Esquema del proceso de graficación.

El diagrama de flujo de la rutina "Muestreo" (o de graficación) se puede observar en la figura 3.24.

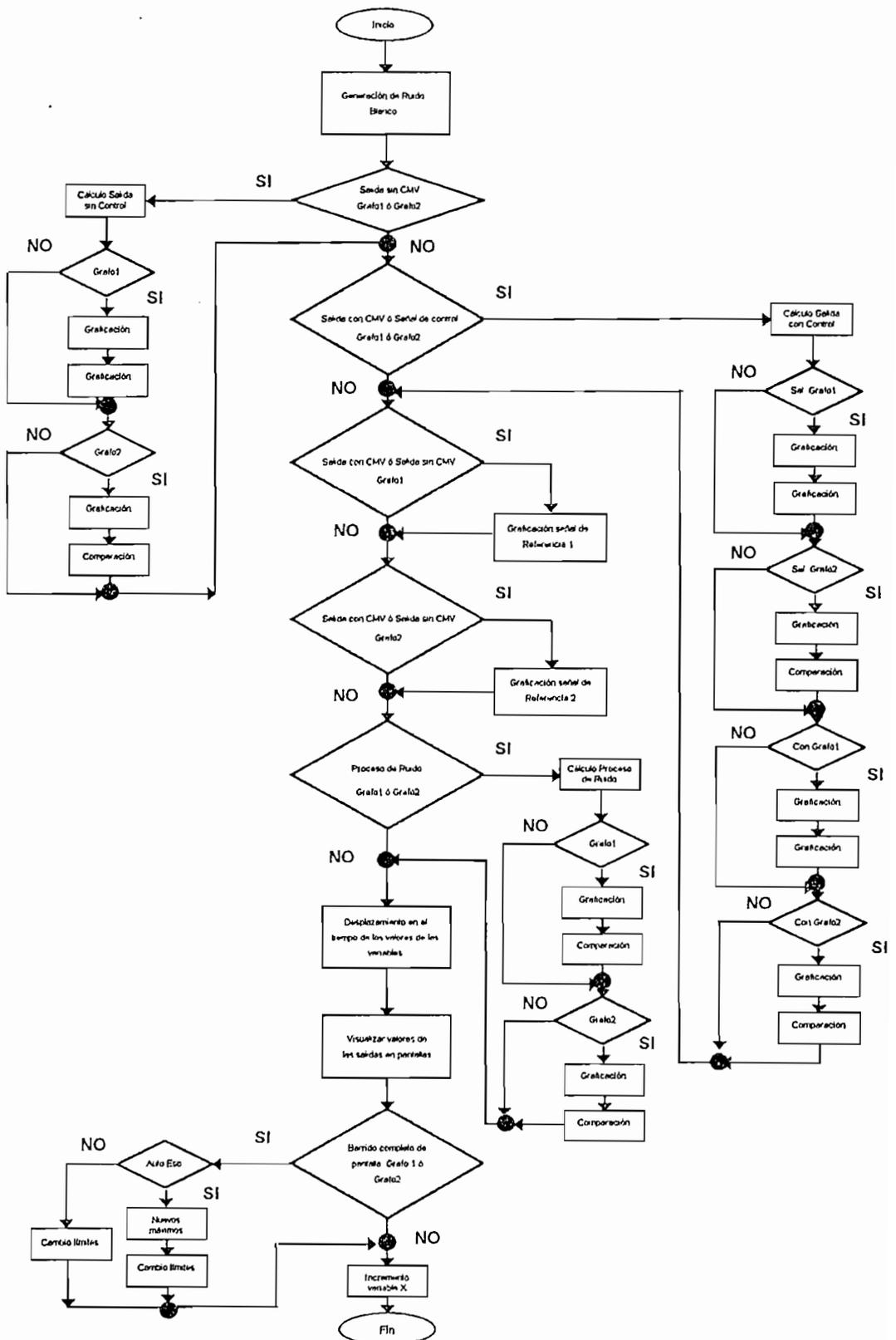


Figura 3.24 Flujograma de la rutina "Muestreo".

En la figura anterior se puede observar que la rutina Muestreo chequea primero las opciones que se escogieron en el Menú de Gráficos. Luego del chequeo de las opciones, dependiendo de cuales han sido seleccionadas se calcula la señal a ser graficada. Esto se lo hace invocando a la rutina Control para graficar la salida controlada o la señal de control; o a la rutina Salida sin CMV para graficar la salida sin control. La señal del proceso estocástico se calcula y grafica directamente desde la subrutina Muestreo. Una vez que se grafica una línea desde el punto anterior al nuevo que se ha calculado, se procede a comparar el valor actual con el anterior para encontrar el máximo entre los dos, con el fin de calcular el nuevo valor del máximo en el eje Y una vez que se haya barrido completamente la pantalla de gráficos.

Cuando se ha graficado y comparado los valores de las señales graficadas se incrementa la variable del eje X para cada gráfico y se sale de la subrutina. Todo este proceso se repite cada período de muestreo indefinidamente hasta que se presione el botón Parar o hasta que haga un clic en las Opciones del menú

3.5 RUTINA PARA EL CONTROL DE MÍNIMA VARIANZA

La rutina Control calcula la señal de control, simula la planta y produce la salida que será afectada por el proceso estocástico generado por la misma rutina. En simulación, la salida de la planta es generada a través del modelo matemático (B / A). En cambio en tiempo real, la salida es medida a través de la tarjeta de adquisición de datos, luego con esta señal se procede a calcular la señal de control, la misma que será aplicada a la planta a través de la tarjeta de adquisición de datos. En la figura 3.25 se observa el diagrama de flujo de la rutina Control.

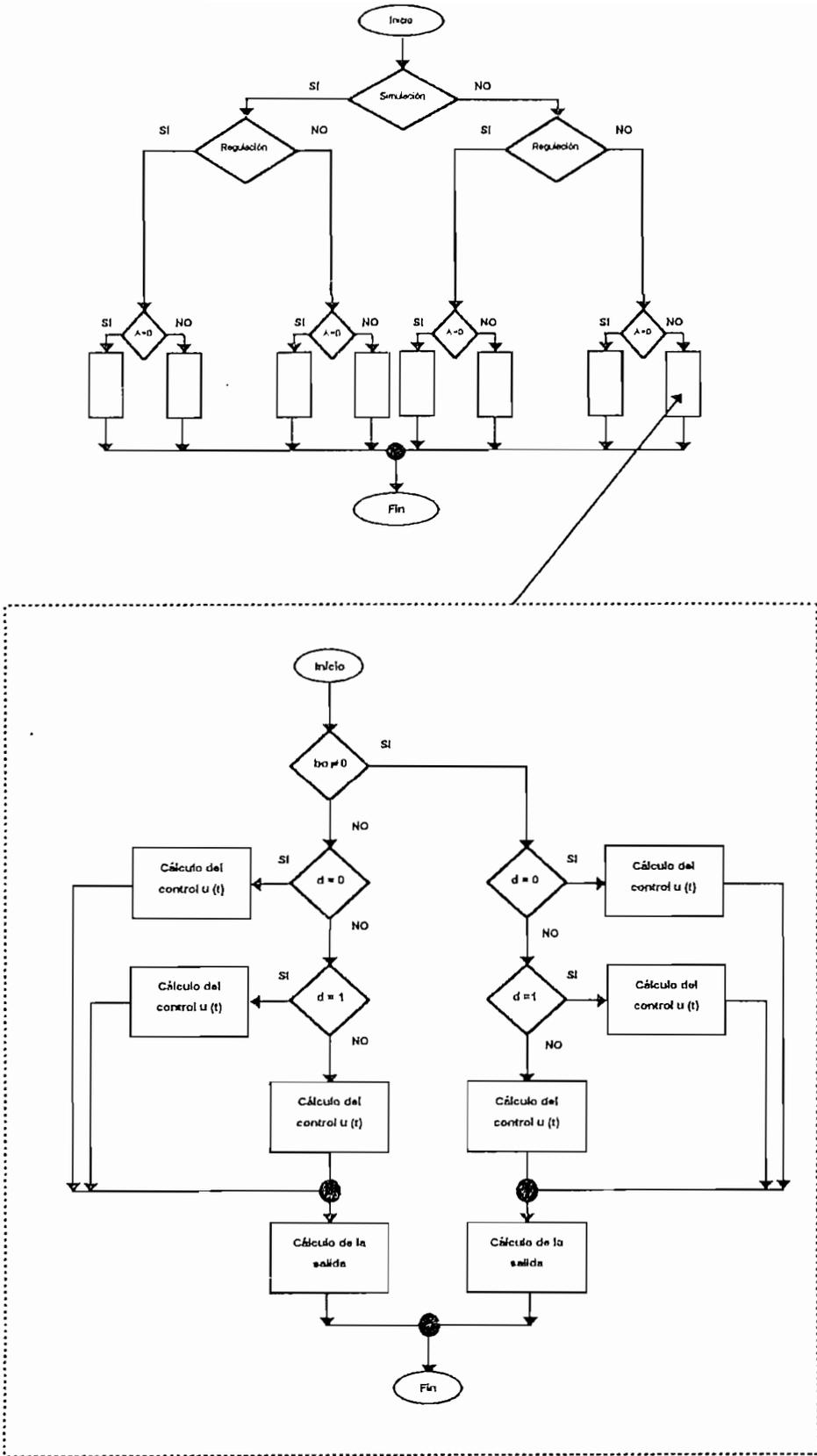


Figura 3.25 Flujo de la rutina Control.

De manera similar a la rutina de graficación, la rutina Control, trabaja verificando las diferentes opciones que se le pueden presentar para el cálculo de la señal de control como de la salida controlada. Estas opciones son :

- Modo de simulación o tiempo real.
- Regulación o seguimiento.
- Polinomio A igual al D o diferentes.
- Sistemas con tiempo muerto o sin él.
- Coeficiente b_0 igual o diferente de cero.

Dependiendo del caso la rutina calcula el control y la salida ajustando el algoritmo a cada necesidad.

En la rutina para la salida sin control simplemente se toma la salida de la planta cuya entrada es una señal constante y se le suma el proceso estocástico.

CAPÍTULO IV : RESULTADOS Y CONCLUSIONES

4.1 Resultados de simulación

4.1.1 Ejemplo 1

4.1.2 Ejemplo 2

4.1.3 Ejemplo 3

4.1.4 Ejemplo 4

4.2 Resultados en tiempo real

4.2.1 Ejemplo 5

4.2.2 Ejemplo 6

4.3 Conclusiones

4. RESULTADOS Y CONCLUSIONES

4.1 RESULTADOS DE SIMULACIÓN

Para comprobar el funcionamiento del programa *Control de Mínima Varianza 1.0*, se han seleccionado una serie de ejemplos con el fin de cubrir todos los casos que se podrían presentar en la utilización de este paquete computacional. Estos ejemplos cumplen con las condiciones impuestas en las secciones 2.2.2 y 2.2.3 (Propiedades del controlador). Todos los ejemplos se los ha analizado para un período de muestreo de 0.5 segundos.

4.1.1 EJEMPLO 1

Sea la planta con función de transferencia :

$$G_p(z) = \frac{0.2 + 0.1 \cdot z^{-1}}{1 - 0.7 \cdot z^{-1}} = \frac{0.2 \cdot (z + 0.5)}{z - 0.7} \quad (4.1)$$

Con la respuesta en el tiempo mostrada en la figura 4.1 (sin considerar el proceso de ruido).

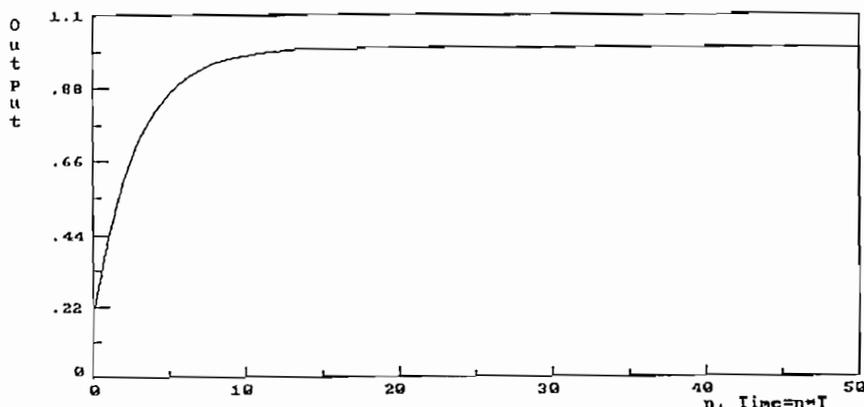


Figura 4.1. Respuesta en el tiempo en lazo abierto de la planta del ejemplo 1.

Y el proceso de ruido con la función de transferencia :

$$Gr(z) = \frac{1}{1 - 0.3 \cdot z^{-1}} = \frac{z}{z - 0.3} \quad (4.2)$$

La figura 4.2 muestra el ruido blanco de entrada y la figura 4.3 la salida del proceso de ruido.

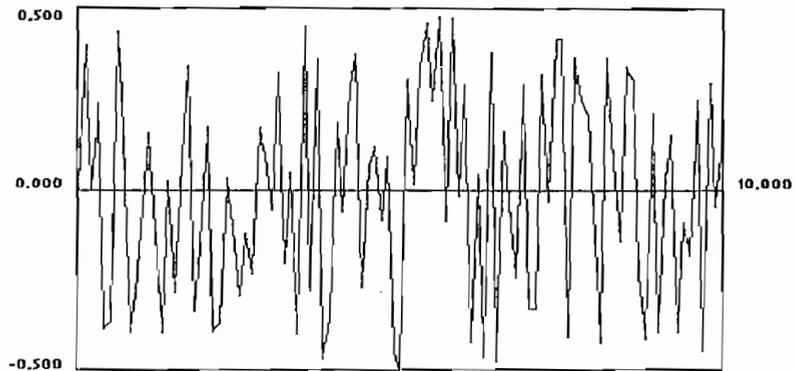


Figura 4.2. Ruido blanco de entrada al proceso de ruido del ejemplo 1.

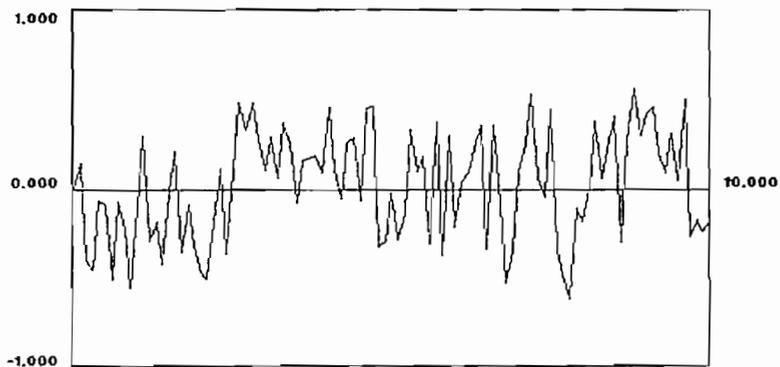


Figura 4.3. Salida del proceso de ruido del ejemplo 1.

Para este sistema se realizaron las pruebas de seguimiento para los retardos desde $d = 0$ a $d = 2$, con el factor de ponderación igual a uno y varianza del ruido igual a uno. Los resultados de este ejemplo se encuentran impresos en las páginas siguientes.

CONTROL DE MINIMA VARIANZA - REPORTE DE DATOS

DATOS DEL SISTEMA

Orden de los polinomios = 1
Factor de ponderación = 1.00000
Retardo $d = 0$
Varianza del ruido = 1.00000

Coefficientes de los polinomios :

$a_0 = 1.00000$ $b_0 = 0.20000$
 $a_1 = -0.70000$ $b_1 = 0.10000$

$c_0 = 1.00000$ $d_0 = 1.00000$
 $c_1 = 0.00000$ $d_1 = -0.30000$

RESULTADOS

Coefficientes de los polinomios F y G :

$f_0 = 1.00000$ $g_0 = 0.30000$
 $f_1 = 0.00000$
 $f_2 = 0.00000$

Varianza de la salida controlada (teórica) = 1.00000
Varianza de la salida controlada (experimental) = 1.04610
Varianza de la salida sin control (experimental) = 1.15614
Varianza del ruido blanco (experimental) = 1.04610
Varianza del proceso de ruido (experimental) = 1.55329

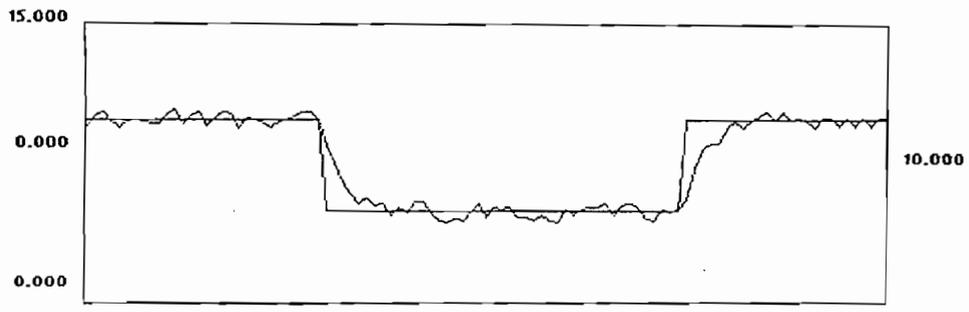


Figura EJE 1.1

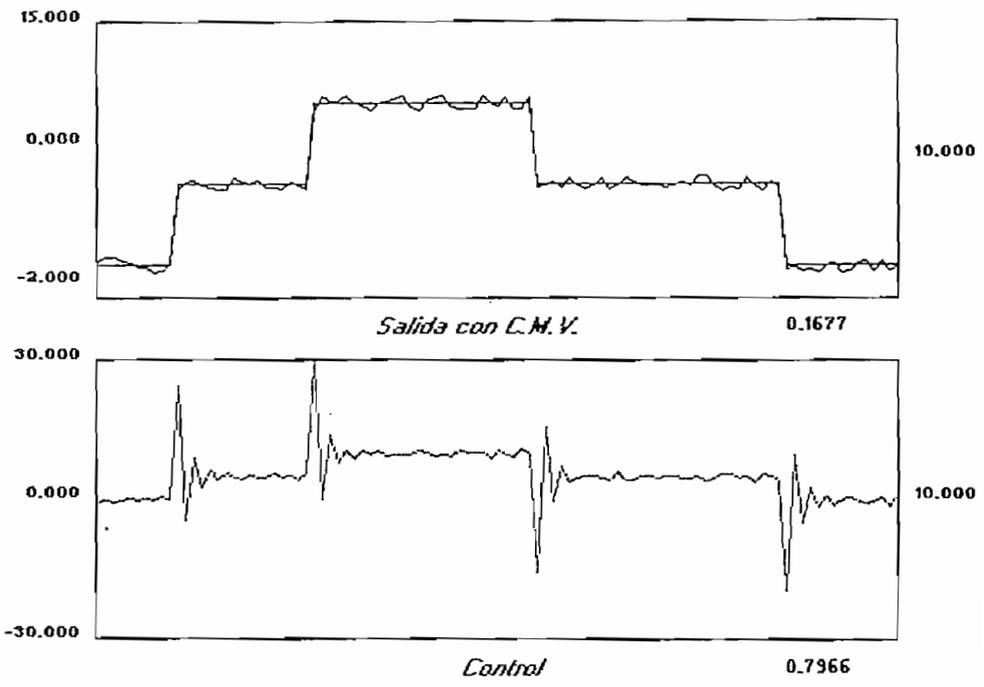


Figura EJE 1.2

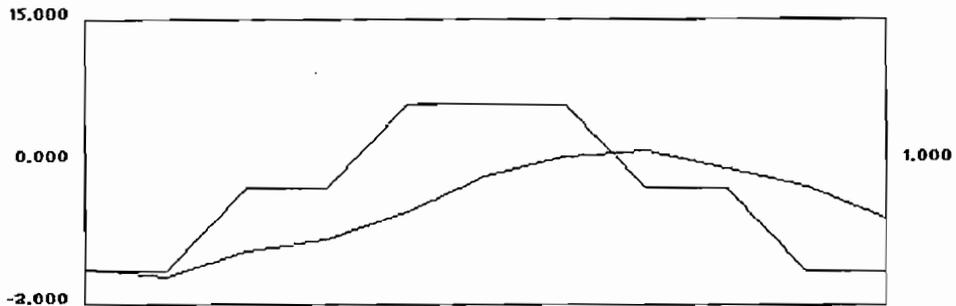
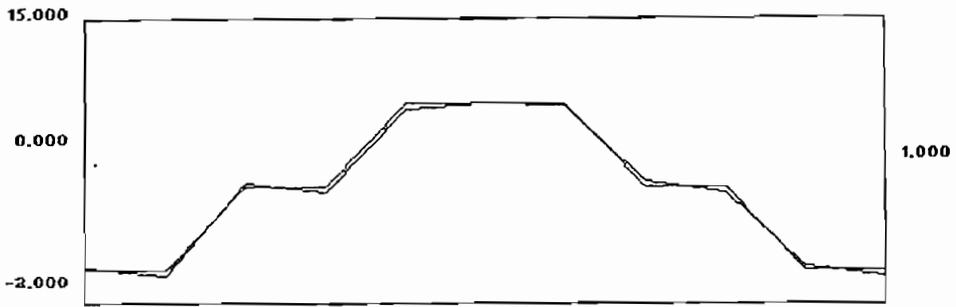


Figura EJE 1.3

CONTROL DE MINIMA VARIANZA - REPORTE DE DATOS

DATOS DEL SISTEMA

Orden de los polinomios = 1
Factor de ponderación = 1.00000
Retardo $d = 1$
Varianza del ruido = 1.00000

Coefficientes de los polinomios :

$a_0 = -0.70000$ $b_0 = 0.20000$
 $a_1 = -0.70000$ $b_1 = 0.10000$

$c_0 = 1.00000$ $d_0 = 1.00000$
 $c_1 = 0.00000$ $d_1 = -0.30000$

RESULTADOS

Coefficientes de los polinomios F y G :

$f_0 = 1.00000$ $g_0 = 0.09000$
 $f_1 = 0.30000$
 $f_2 = 0.00000$

Varianza de la salida controlada (teórica) = 1.09000
Varianza de la salida controlada (experimental) = 1.08618
Varianza de la salida sin control (experimental) = 1.10394
Varianza del ruido blanco (experimental) = 0.99069
Varianza del proceso de ruido (experimental) = 1.46084

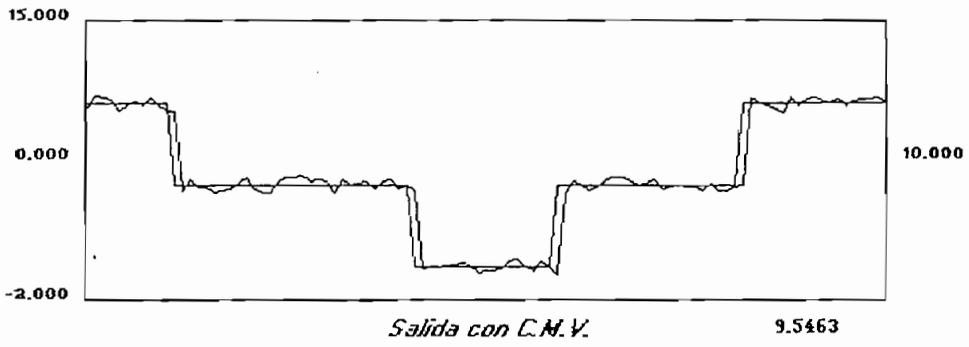
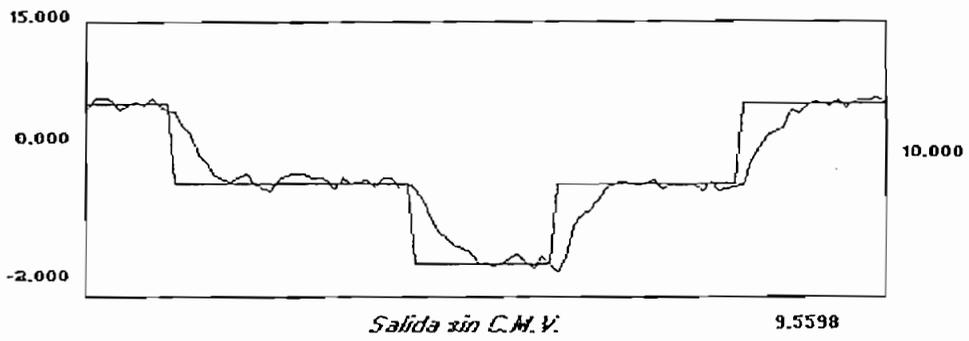


Figura EJE 1.4

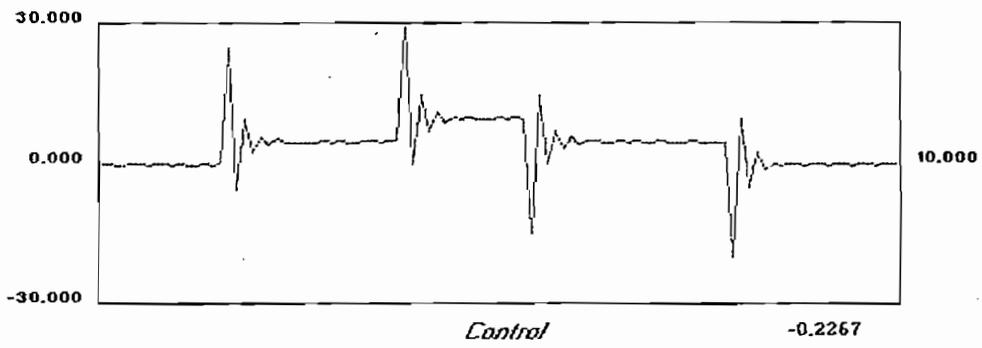


Figura EJE 1.5

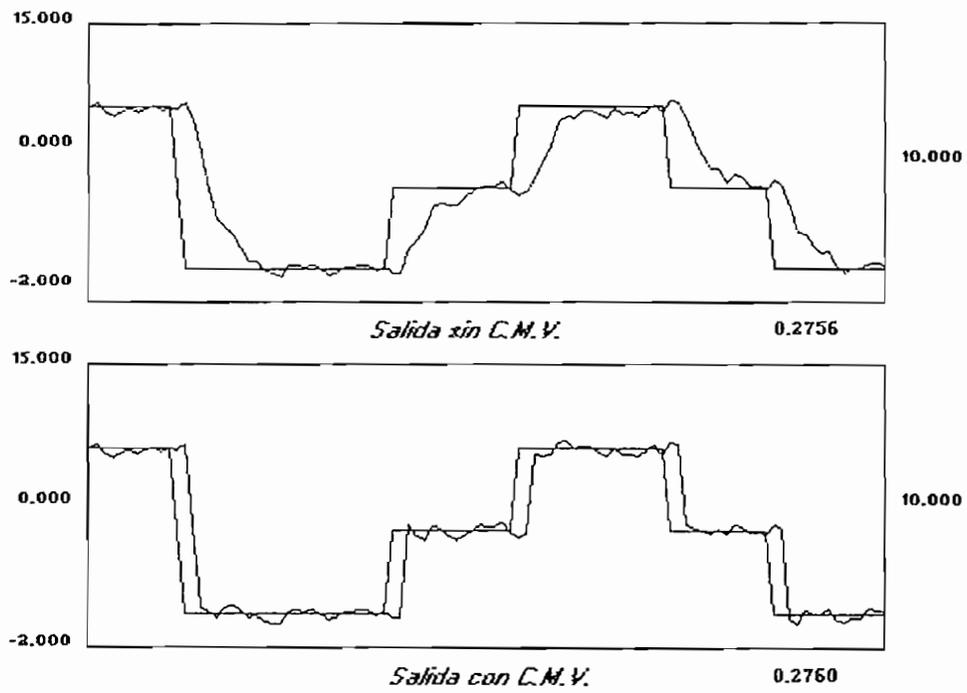


Figura EJE 1.6

CON

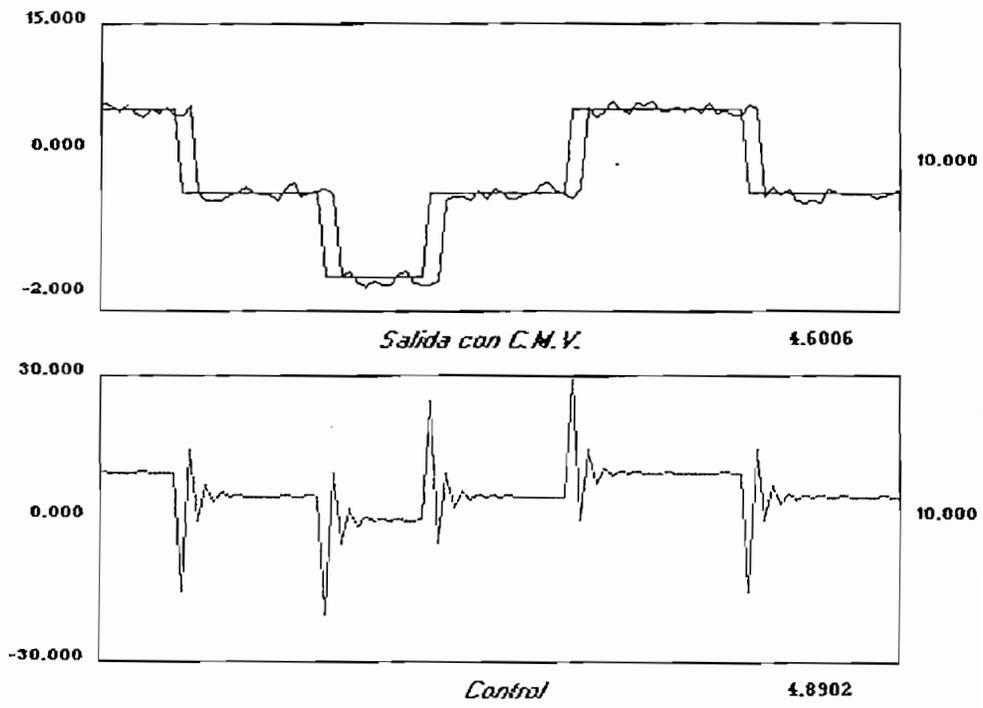


Figura EJE 1.7

4.1.2 EJEMPLO 2

Sea la planta con función de transferencia :

$$Gp(z) = \frac{z^{-1} + 0.5 \cdot z^{-2}}{1 - 1.7 \cdot z^{-1} + 0.7 \cdot z^{-2}} = \frac{z + 0.5}{(z - 1) \cdot (z - 0.7)} \quad (4.3)$$

Con la respuesta en el tiempo mostrada en la figura 4.4 (sin considerar el proceso de ruido).

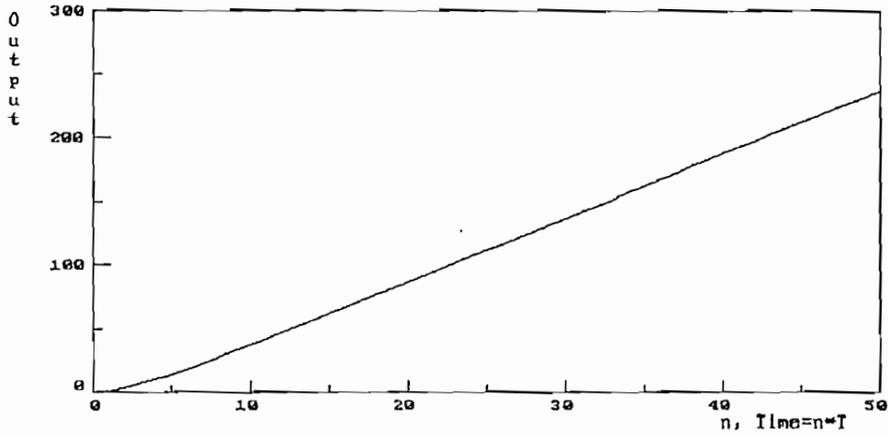


Figura 4.4. Respuesta en el tiempo en lazo abierto de la planta del ejemplo 2.

Y el proceso de ruido con la función de transferencia :

$$Gr(z) = \frac{1 + 1.5 \cdot z^{-1} + 0.9 \cdot z^{-2}}{1 - 1.7 \cdot z^{-1} + 0.7 \cdot z^{-2}} = \frac{(z + 0.75 + j0.58) \cdot (z + 0.75 - j0.58)}{(z - 1) \cdot (z - 0.7)} \quad (4.4)$$

La figura 4.5 muestra el ruido blanco de entrada y la figura 4.6 la salida del proceso de ruido.

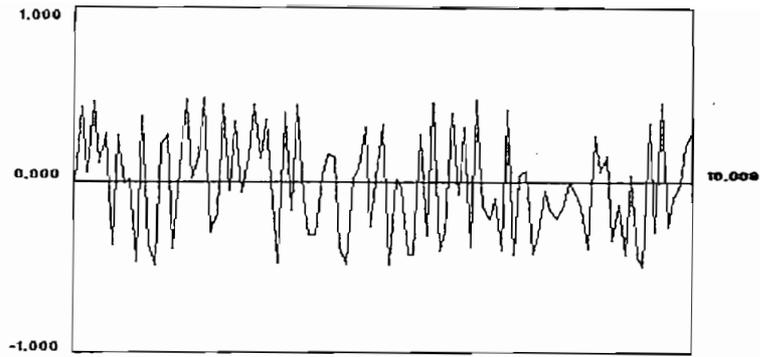


Figura 4.5. Ruido blanco de entrada al proceso de ruido del ejemplo 2.

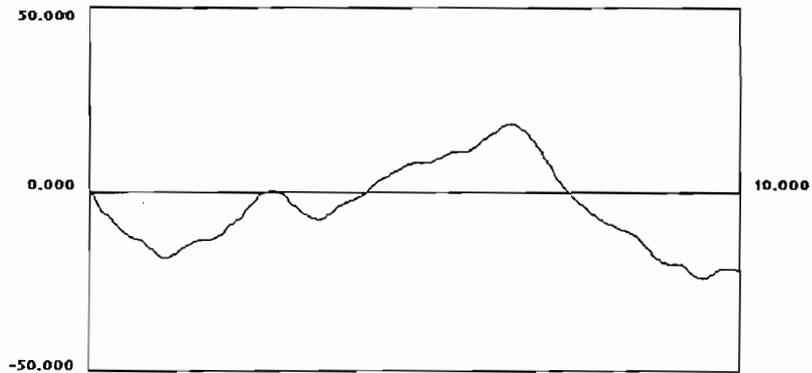


Figura 4.6. Salida del proceso de ruido del ejemplo 2.

Para este sistema se realizaron las pruebas de regulación para los retardos $d = 0$ y $d = 2$, con el factor de ponderación igual a cero y la varianza del ruido igual a uno. Los resultados de este ejemplo se encuentran impresos en las páginas siguientes. [2]

CONTROL DE MINIMA VARIANZA - REPORTE DE DATOS

DATOS DEL SISTEMA

Orden de los polinomios = 2
Factor de ponderación = 0.00000
Retardo $d = 0$
Varianza del ruido = 1.00000

Coefficientes de los polinomios :

$a_0 = 1.00000$	$b_0 = 0.00000$
$a_1 = -1.70000$	$b_1 = 1.00000$
$a_2 = 0.70000$	$b_2 = 0.50000$

$c_0 = 1.00000$	$d_0 = 1.00000$
$c_1 = 1.50000$	$d_1 = -1.70000$
$c_2 = 0.90000$	$d_2 = 0.70000$

RESULTADOS

Coefficientes de los polinomios F y G :

$f_0 = 1.00000$	$g_0 = 3.20000$
$f_1 = 0.00000$	$g_1 = 0.20000$
$f_2 = 0.00000$	

Varianza de la salida controlada (teórica) = 1.00000
Varianza de la salida controlada (experimental) = 1.03386
Varianza de la salida sin control (experimental) = 4725.74446
Varianza del ruido blanco (experimental) = 1.03386
Varianza del proceso de ruido (experimental) = 4.54174

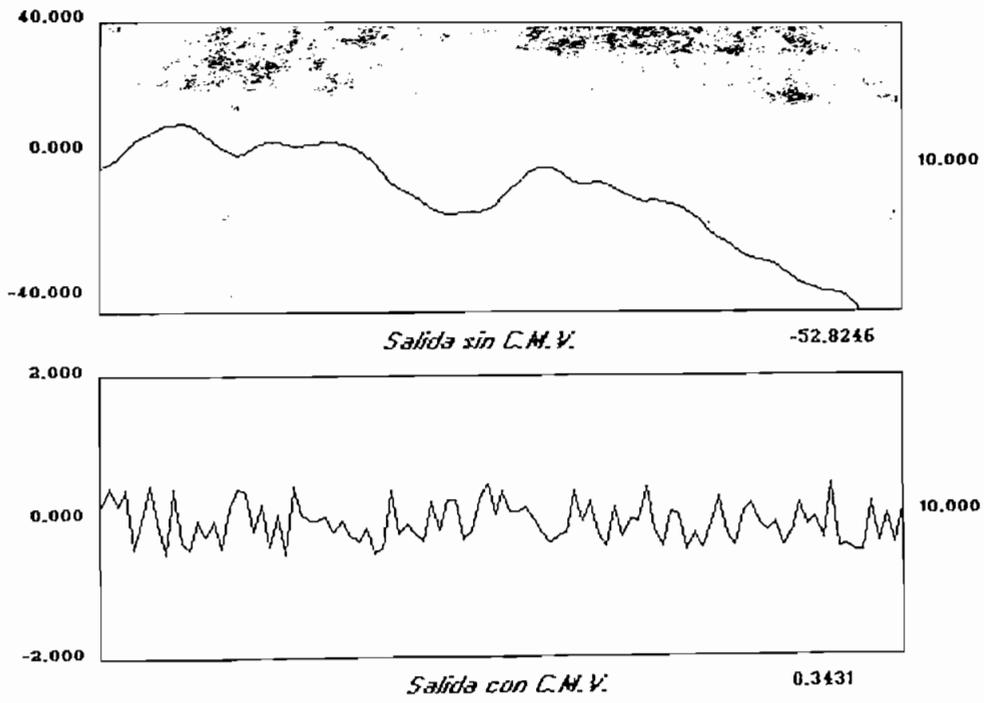


Figura EJE 2.1

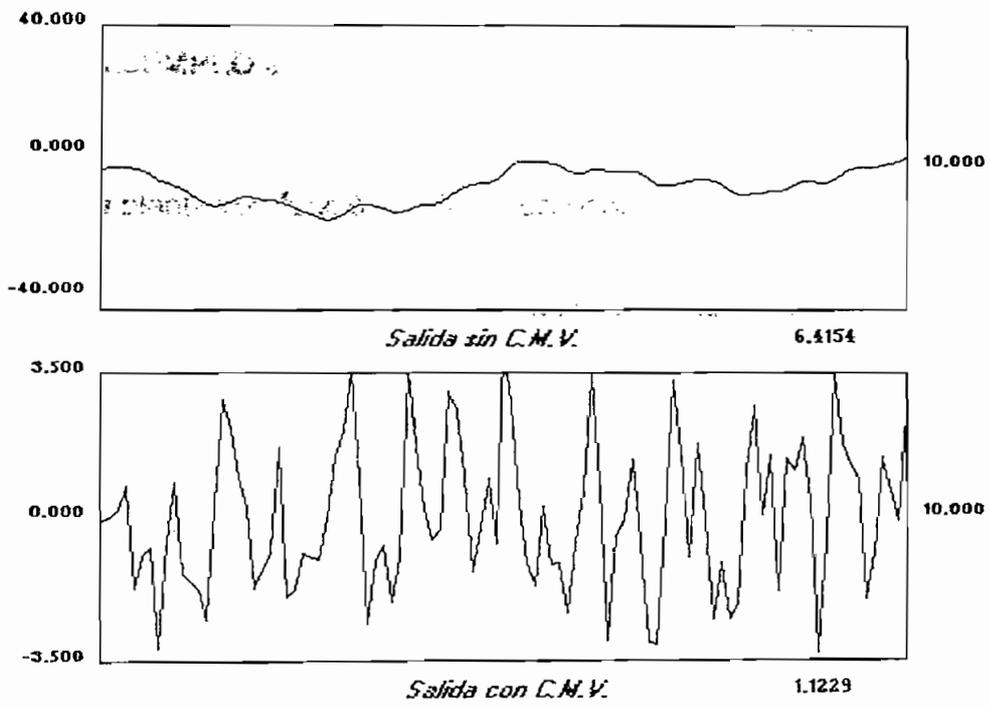


Figura EJE 2.2

CONTROL DE MINIMA VARIANZA - REPORTE DE DATOS

DATOS DEL SISTEMA

Orden de los polinomios = 3
Factor de ponderación = 0.00000
Retardo $d = 0$
Varianza del ruido = 1.00000

Coefficientes de los polinomios :

$a_0 = 1.00000$	$b_0 = 0.00000$
$a_1 = 0.40000$	$b_1 = 1.00000$
$a_2 = -0.37000$	$b_2 = 0.60000$
$a_3 = -0.04000$	$b_3 = -0.27000$

$c_0 = 1.00000$	$d_0 = 1.00000$
$c_1 = 0.00000$	$d_1 = 0.20000$
$c_2 = 0.30000$	$d_2 = 0.01000$
$c_3 = 0.00000$	$d_3 = 0.00000$

RESULTADOS

Coefficientes de los polinomios F y G :

$f_0 = 1.00000$	$g_0 = -0.20000$
$f_1 = 0.00000$	$g_1 = 0.29000$
$f_2 = 0.00000$	$g_2 = 0.00000$

Varianza de la salida controlada (teórica) = 1.00000
Varianza de la salida controlada (experimental) = 1.00121
Varianza de la salida sin control (experimental) = 1.18985
Varianza del ruido blanco (experimental) = 1.00162
Varianza del proceso de ruido (experimental) = 1.16425

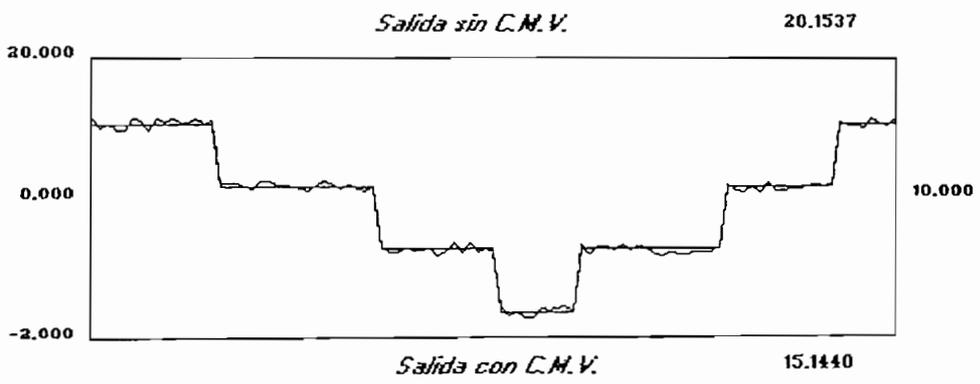
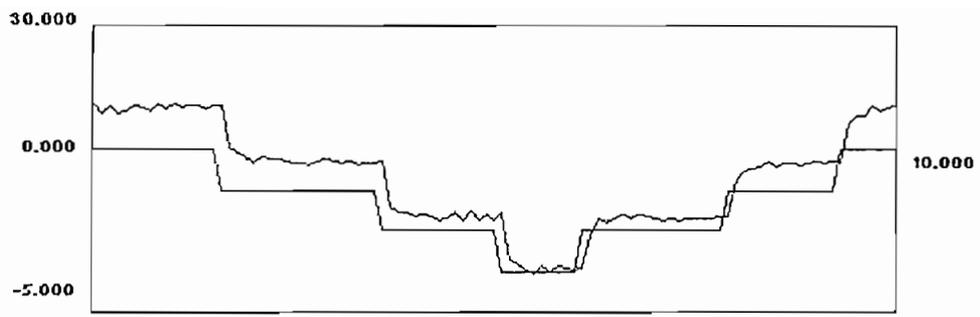


Figura EJE 3.1

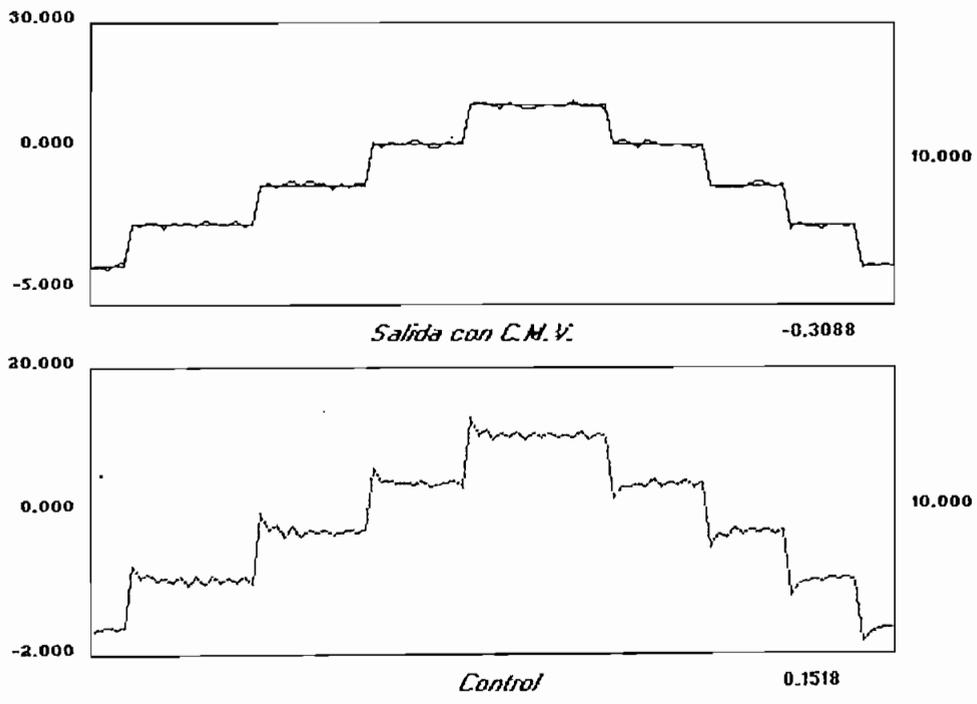


Figura EJE 3.2

CONTROL DE MINIMA VARIANZA - REPORTE DE DATOS

DATOS DEL SISTEMA

Orden de los polinomios = 3
Factor de ponderación = 0.00000
Retardo $d = 2$
Varianza del ruido = 1.00000

Coeficientes de los polinomios :

$a_0 = 1.00000$	$b_0 = 0.00000$
$a_1 = 0.40000$	$b_1 = 1.00000$
$a_2 = -0.37000$	$b_2 = 0.60000$
$a_3 = -0.04000$	$b_3 = -0.27000$

$c_0 = 1.00000$	$d_0 = 1.00000$
$c_1 = 0.00000$	$d_1 = 0.20000$
$c_2 = 0.30000$	$d_2 = 0.01000$
$c_3 = 0.00000$	$d_3 = 0.00000$

RESULTADOS

Coeficientes de los polinomios F y G :

$f_0 = 1.00000$	$g_0 = -0.06400$
$f_1 = -0.20000$	$g_1 = -0.00330$
$f_2 = 0.33000$	$g_2 = 0.00000$

Varianza de la salida controlada (teórica) = 1.14890
Varianza de la salida controlada (experimental) = 1.13906
Varianza de la salida sin control (experimental) = 1.14167
Varianza del ruido blanco (experimental) = 0.97699
Varianza del proceso de ruido (experimental) = 1.15008

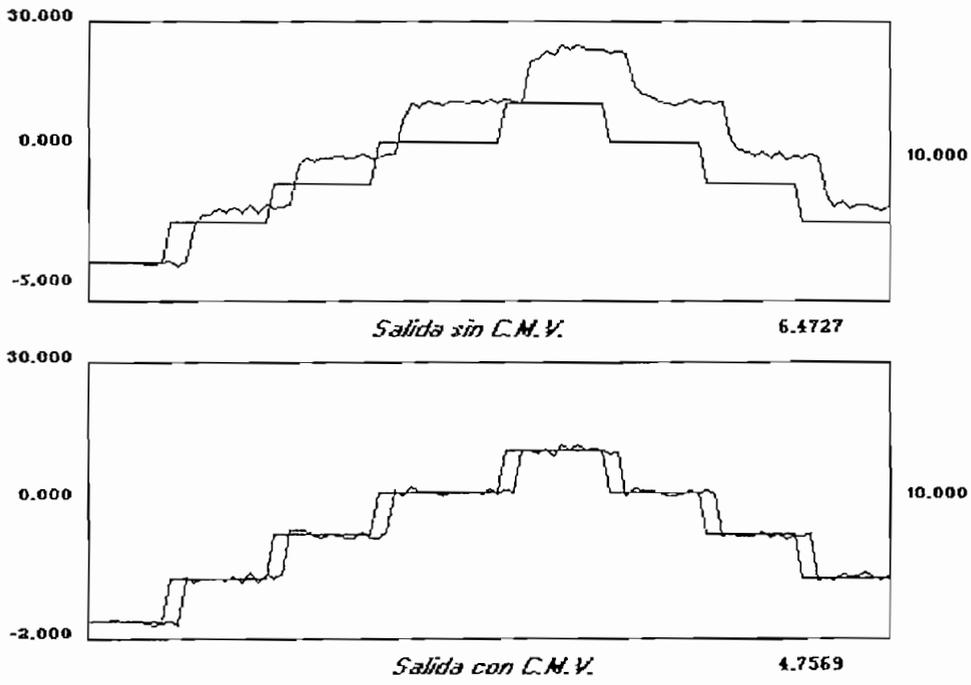
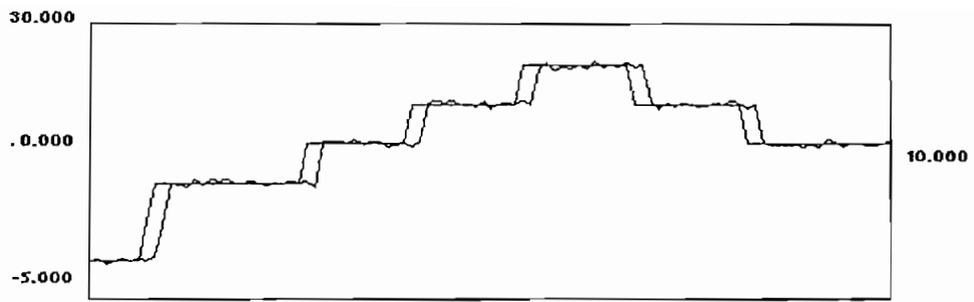
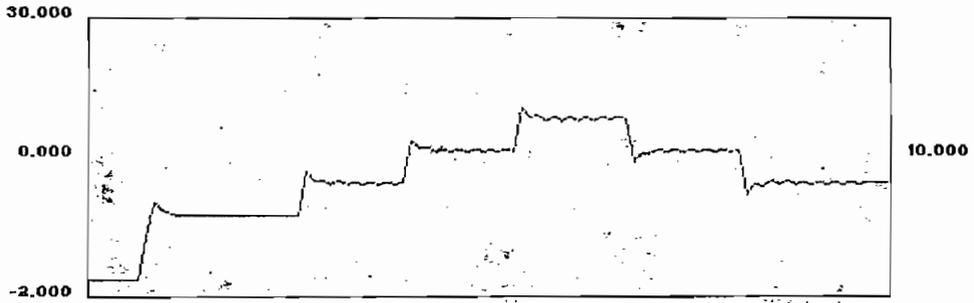


Figura EJE 3.3



Salida con C.M.V.

14.7211



Control

11.2271

Figura EJE 3.4

4.1.4 EJEMPLO 4

Sea la planta con función de transferencia :

$$G_P(z) = \frac{1 + z^{-3}}{1 - 0.41 \cdot z^{-2} - 0.045 \cdot z^{-4}} \quad (4.7)$$
$$= \frac{z \cdot (z+1) \cdot (z - 0.5 + j0.86) \cdot (z - 0.5 - j0.86)}{(z - \sqrt{0.5}) \cdot (z + \sqrt{0.5}) \cdot (z - j0.3) \cdot (z + j0.3)}$$

Con la respuesta en el tiempo mostrada en la figura 4.10 (sin tomar en cuenta el proceso de ruido).

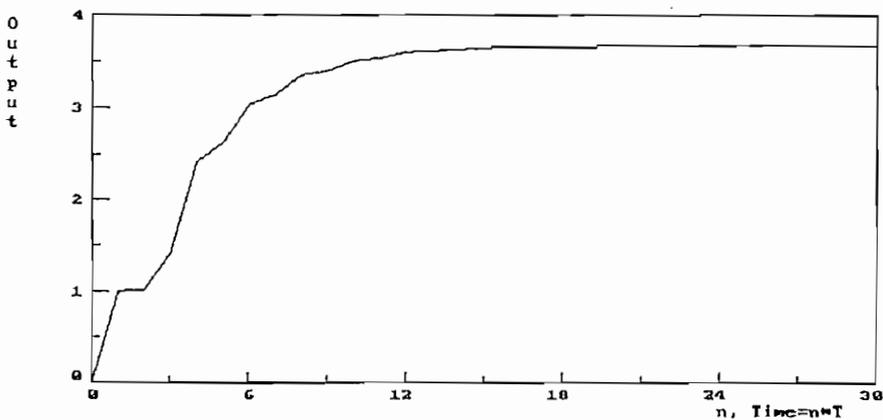


Figura 4.10. Respuesta en el tiempo en lazo abierto de la planta del ejemplo 4.

Y el proceso de ruido con la función de transferencia :

$$G_R(z) = 1 \quad (4.8)$$

Para este sistema se realizaron las pruebas de seguimiento para los retardos $d = 1$ y $d = 2$, con el factor de ponderación igual a cero y la varianza del ruido igual a uno. Los resultados de este ejemplo se encuentran impresos en las páginas siguientes.

CONTROL DE MINIMA VARIANZA - REPORTE DE DATOS

DATOS DEL SISTEMA

Orden de los polinomios = 4
Factor de ponderación = 0.00000
Retardo $d = 1$
Varianza del ruido = 1.00000

Coefficientes de los polinomios :

$a_0 = 1.00000$	$b_0 = 1.00000$
$a_1 = 0.00000$	$b_1 = 0.00000$
$a_2 = -0.41000$	$b_2 = 0.00000$
$a_3 = 0.00000$	$b_3 = 1.00000$
$a_4 = -0.04500$	$b_4 = 0.00000$

$c_0 = 1.00000$	$d_0 = 1.00000$
$c_1 = 0.00000$	$d_1 = 0.00000$
$c_2 = 0.00000$	$d_2 = 0.00000$
$c_3 = 0.00000$	$d_3 = 0.00000$
$c_4 = 0.00000$	$d_4 = 0.00000$

RESULTADOS

Coefficientes de los polinomios F y G :

$f_0 = 1.00000$	$g_0 = 0.00000$
$f_1 = 0.00000$	$g_1 = 0.00000$
$f_2 = 0.00000$	$g_2 = 0.00000$
	$g_3 = 0.00000$

Varianza de la salida controlada (teórica) = 1.00000
Varianza de la salida controlada (experimental) = 1.02273
Varianza de la salida sin control (experimental) = 1.02272
Varianza del ruido blanco (experimental) = 1.02273
Varianza del proceso de ruido (experimental) = 1.20235

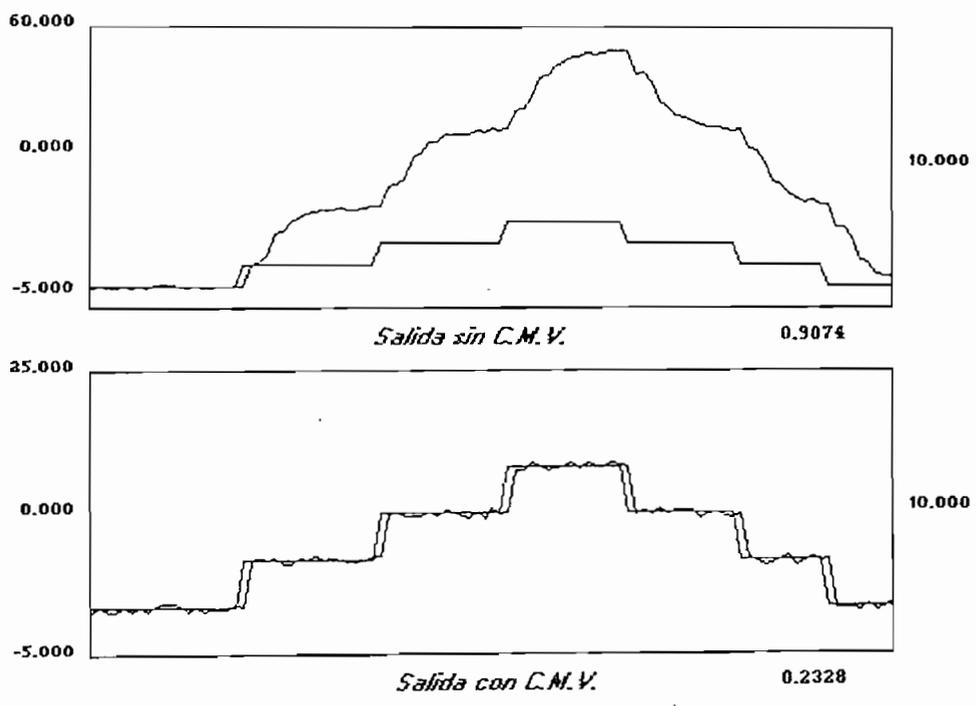


Figura EJE 4.1

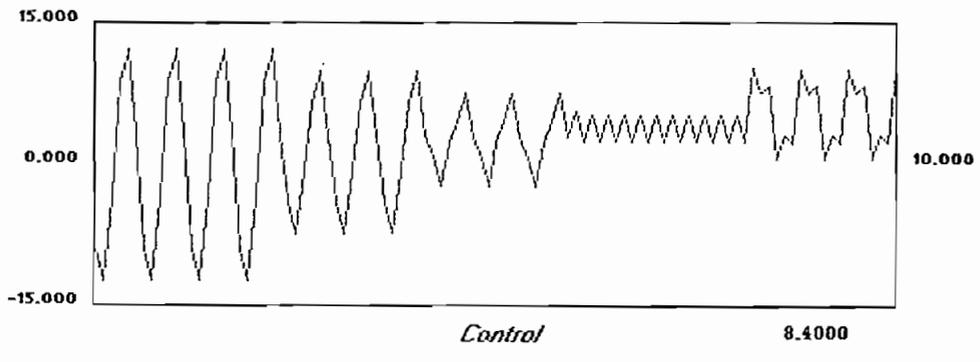
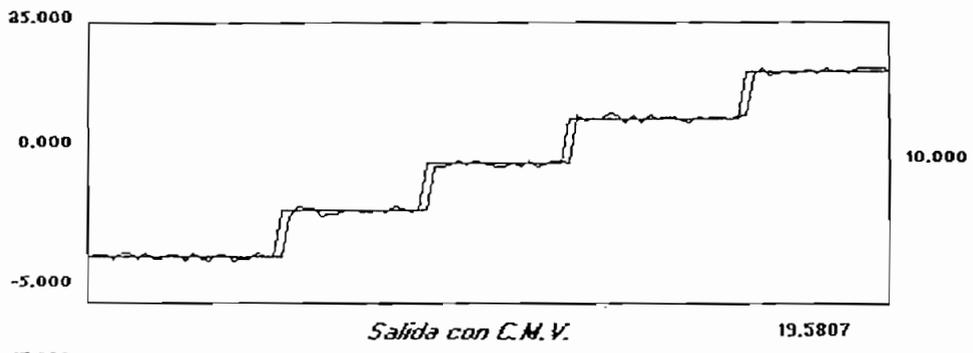


Figura EJE 4.2

CONTROL DE MINIMA VARIANZA - REPORTE DE DATOS

DATOS DEL SISTEMA

Orden de los polinomios = 4
Factor de ponderación = 0.00000
Retardo $d = 2$
Varianza del ruido = 1.00000

Coefficientes de los polinomios :

$a_0 = 1.00000$	$b_0 = 1.00000$
$a_1 = 0.00000$	$b_1 = 0.00000$
$a_2 = -0.41000$	$b_2 = 0.00000$
$a_3 = 0.00000$	$b_3 = 1.00000$
$a_4 = -0.04500$	$b_4 = 0.00000$

$c_0 = 1.00000$	$d_0 = 1.00000$
$c_1 = 0.00000$	$d_1 = 0.00000$
$c_2 = 0.00000$	$d_2 = 0.00000$
$c_3 = 0.00000$	$d_3 = 0.00000$
$c_4 = 0.00000$	$d_4 = 0.00000$

RESULTADOS

Coefficientes de los polinomios F y G :

$f_0 = 1.00000$	$g_0 = 0.00000$
$f_1 = 0.00000$	$g_1 = 0.00000$
$f_2 = 0.00000$	$g_2 = 0.00000$
	$g_3 = 0.00000$

Varianza de la salida controlada (teórica) = 1.00000
Varianza de la salida controlada (experimental) = 1.00370
Varianza de la salida sin control (experimental) = 1.00371
Varianza del ruido blanco (experimental) = 1.00370
Varianza del proceso de ruido (experimental) = 1.14954

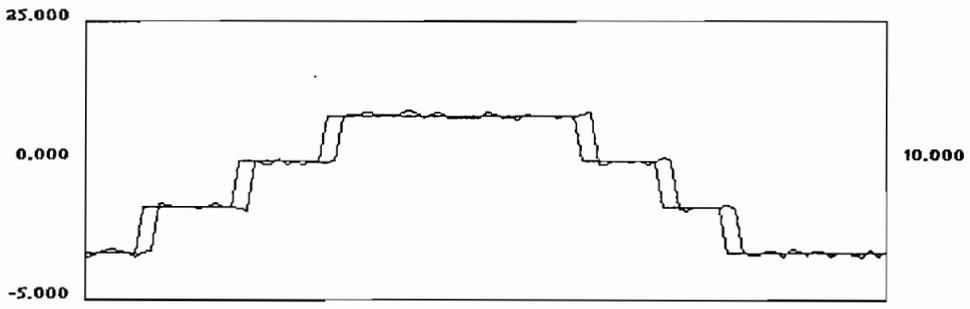
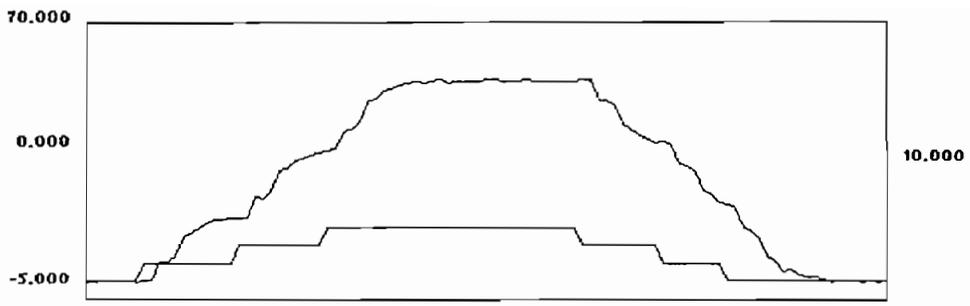


Figura EJE 4.3

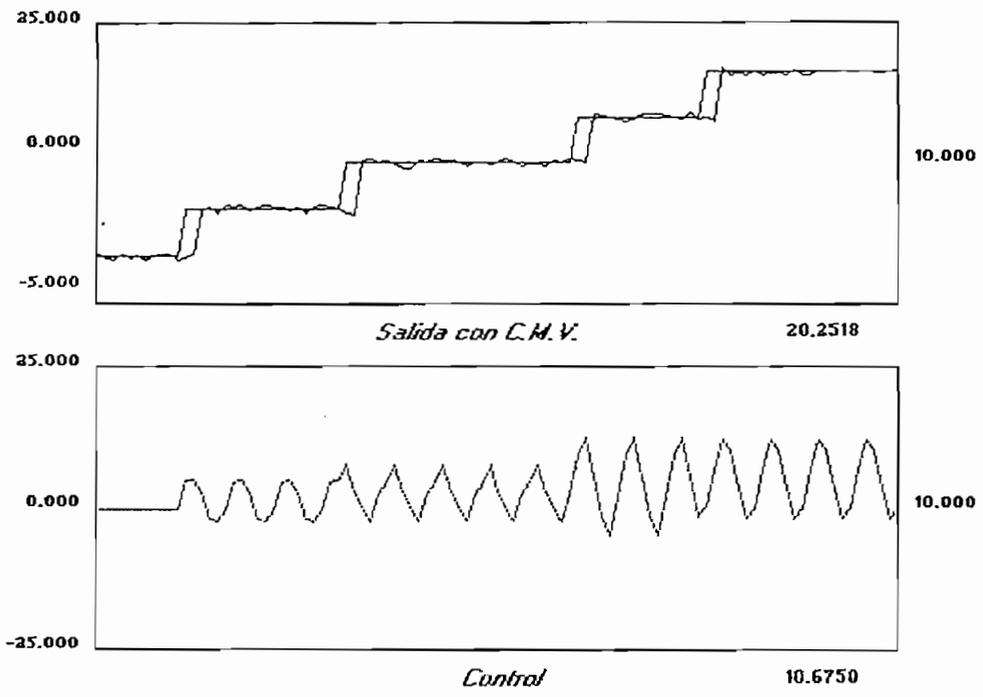


Figura EJE 4.4

4.2 RESULTADOS EN TIEMPO REAL

Para comprobar el funcionamiento del programa *Control de Mínima Varianza 1.0* en tiempo real se procedió a la construcción de prototipos circuitales que simulan plantas de primero y segundo orden a las cuales se les aplica ruido correlacionado generado desde el computador. Al conocer las funciones de transferencia de estos prototipos no fue necesaria la identificación paramétrica [12], tan solo la discretización del modelo continuo con un tiempo de muestreo de 0.5 segundos para todos los ejemplos siguientes.

4.2.1 EJEMPLO 5

Sea la planta con función de transferencia :

$$G_p(s) = \frac{1}{s+1} = \frac{Y_p(s)}{U(s)} \quad (4.9)$$

El prototipo circuitual para la implementación de este sistema se muestra en la figura 4.11.

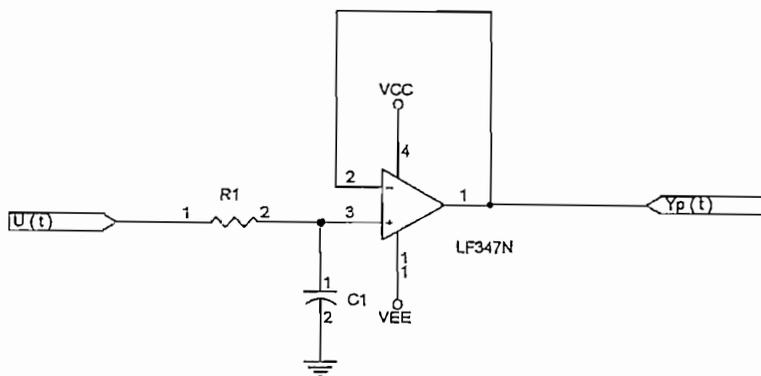


Figura 4.11. Circuito RC de primer orden.

Donde $C_1 = 10 \mu F$ y $R_1 = 100 K\Omega$. Obteniéndose una constante de tiempo τ igual a :

$$\tau = R_1 \cdot C_1 = 1 \text{ segundos}$$

El seguidor de voltaje conectado a la salida del circuito permite desacoplar el circuito de la entrada a la tarjeta que tiene muy baja impedancia.

Discretizando la función de transferencia de la planta se tiene :

$$G_p(z) = \frac{0.3934}{z - 0.6065} = \frac{0.3934 \cdot z^{-1}}{1 - 0.6065 \cdot z^{-1}} \quad (4.10)$$

La respuesta en el tiempo para esta planta sin considerar el proceso de ruido se observa en la figura 4.12.

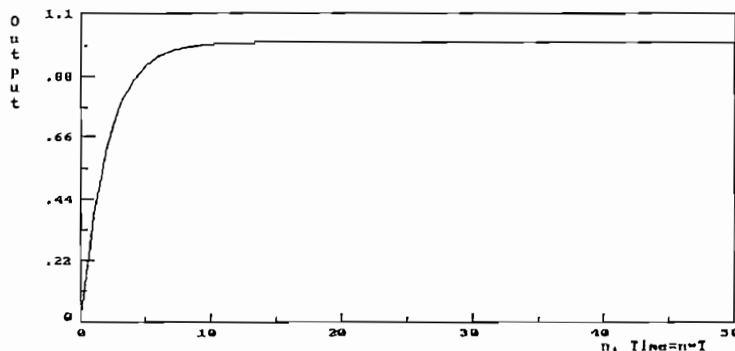


Figura 4.12. Respuesta en el tiempo en lazo abierto de la planta del ejemplo 5.

Y el proceso de ruido con la función de transferencia :

$$G_r(z) = \frac{1}{1 - 0.3 \cdot z^{-1}} = \frac{z}{z - 0.3} \quad (4.11)$$

CONTROL DE MINIMA VARIANZA - REPORTE DE DATOS

DATOS DEL SISTEMA

Orden de los polinomios = 1
Factor de ponderación = 0.00000
Retardo $d = 0$
Varianza del ruido = 1.00000

Coefficientes de los polinomios :

$a_0 = 1.00000$ $b_0 = 0.00000$
 $a_1 = -0.60653$ $b_1 = 0.39347$

$c_0 = 1.00000$ $d_0 = 1.00000$
 $c_1 = 0.00000$ $d_1 = -0.30000$

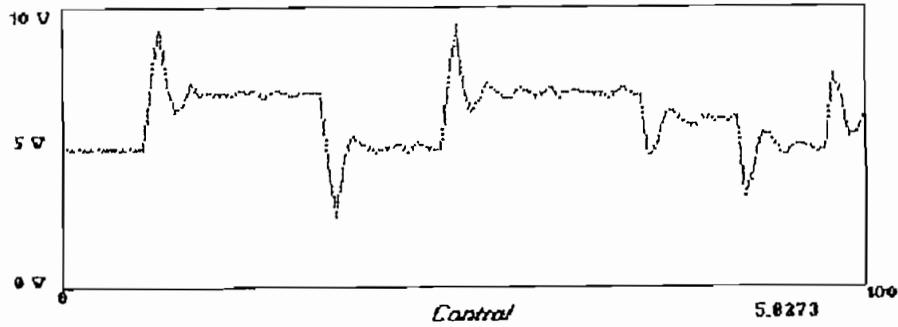
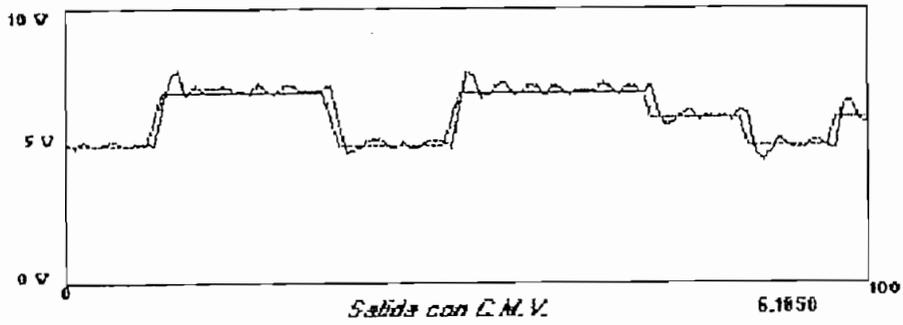
RESULTADOS

Coefficientes de los polinomios F y G :

$f_0 = 1.00000$ $g_0 = 0.30000$
 $f_1 = 0.00000$
 $f_2 = 0.00000$

Varianza de la salida controlada (teórica) = 1.00000
Varianza de la salida controlada (experimental) = 1.02205
Varianza de la salida sin control (experimental) = 1.13936
Varianza del ruido blanco (experimental) = 1.02205
Varianza del proceso de ruido (experimental) = 1.37782

CONTROL DE MINIMA VARIANZA - REPORTE DE GRAFICOS



CONTROL DE MINIMA VARIANZA - REPORTE DE GRAFICOS

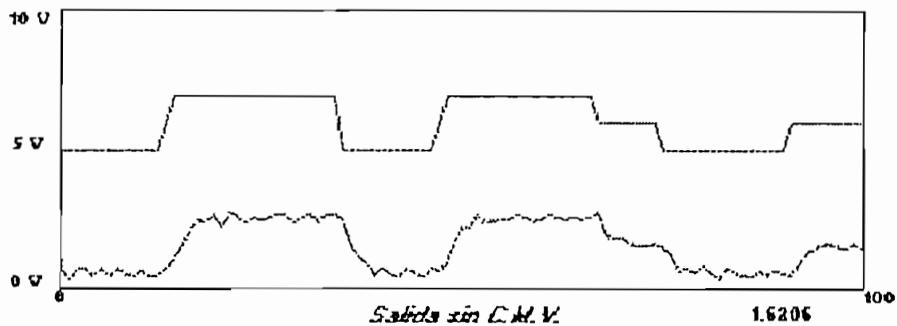
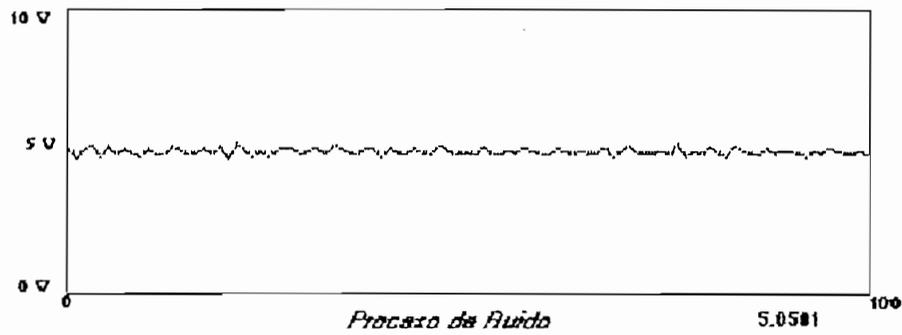


Figura EJE 5.1

CONTROL DE MINIMA VARIANZA - REPORTE DE DATOS

DATOS DEL SISTEMA

Orden de los polinomios = 1
Factor de ponderación = 0.00000
Retardo $d = 1$
Varianza del ruido = 1.00000

Coefficientes de los polinomios :

$a_0 = 1.00000$ $b_0 = 0.00000$
 $a_1 = -0.60653$ $b_1 = 0.39347$

$c_0 = 1.00000$ $d_0 = 1.00000$
 $c_1 = 0.00000$ $d_1 = -0.30000$

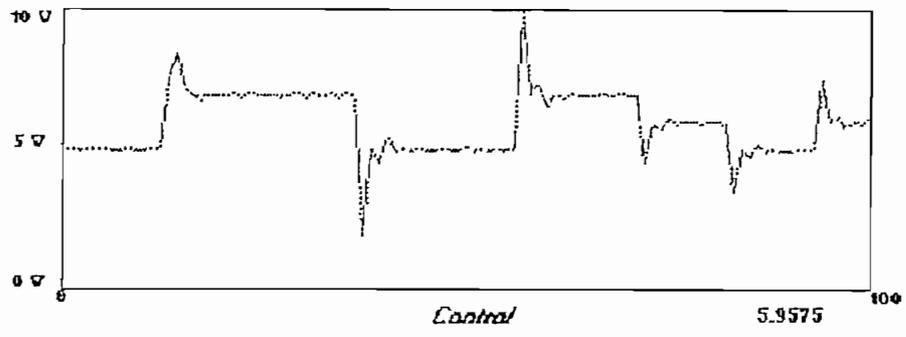
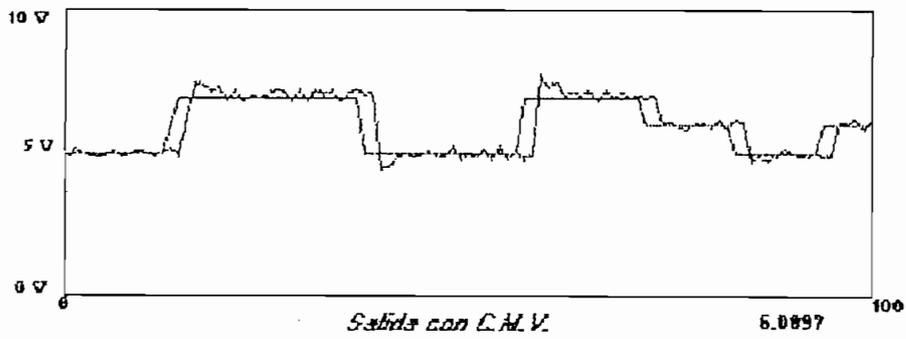
RESULTADOS

Coefficientes de los polinomios F y G :

$f_0 = 1.00000$ $g_0 = 0.09000$
 $f_1 = 0.30000$
 $f_2 = 0.00000$

Varianza de la salida controlada (teórica) = 1.09000
Varianza de la salida controlada (experimental) = 1.09701
Varianza de la salida sin control (experimental) = 1.10920
Varianza del ruido blanco (experimental) = 0.99517
Varianza del proceso de ruido (experimental) = 1.33742

CONTROL DE MINIMA VARIANZA - REPORTE DE GRAFICOS



CONTROL DE MINIMA VARIANZA - REPORTE DE GRAFICOS

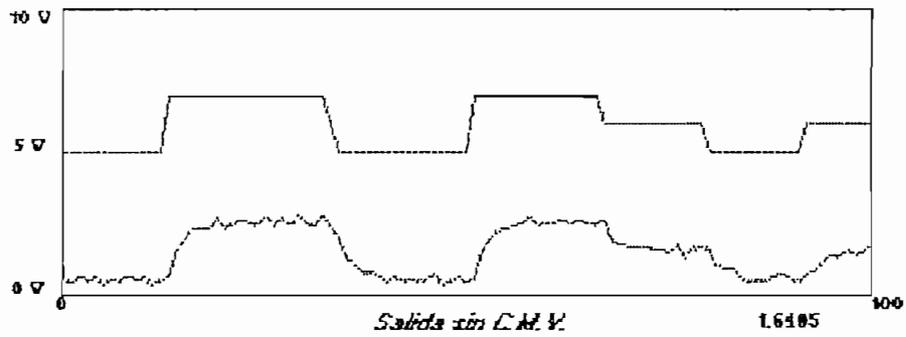
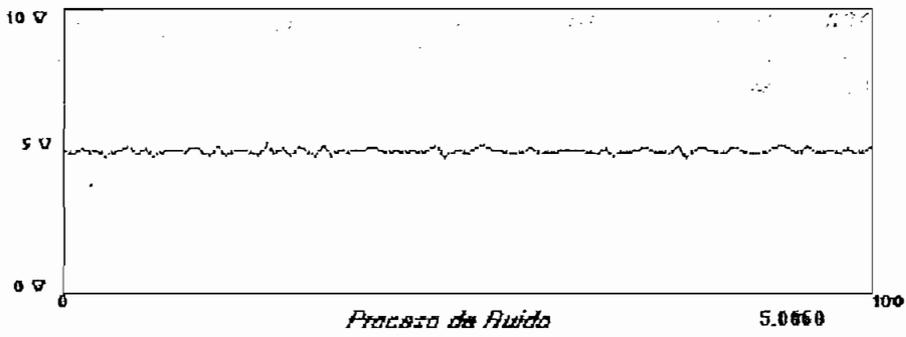


Figura EJE 5.2

4.2.2 EJEMPLO 6

Sea la planta con función de transferencia :

$$Gp(s) = \frac{1}{(s + 1) \cdot (0.47 \cdot s + 1)} = \frac{Y_p(s)}{U(s)} \quad (4.12)$$

El prototipo circuitual para la implementación de este sistema se muestra en la figura 4.15.

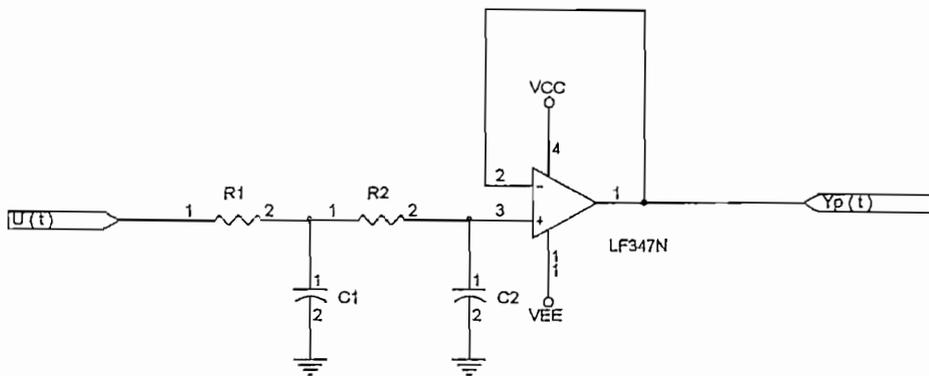


Figura 4.15. Circuito RC de segundo orden.

Donde $C_1 = 10 \mu F$, $R_1 = 100 K\Omega$, $C_2 = 100 \mu F$ y $R_2 = 4.7 K\Omega$.

Obteniéndose las constantes de tiempo τ_1 y τ_2 iguales a :

$$\tau_1 = R_1 \cdot C_1 = 1 \text{ segundos} \quad \text{y} \quad \tau_2 = R_2 \cdot C_2 = 0.47 \text{ segundos}$$

El seguidor de voltaje conectado a la salida del circuito permite desacoplar el circuito de la entrada a la tarjeta que tiene muy baja impedancia.

Discretizando la función de transferencia de la planta se tiene :

$$G_p(z) = \frac{0.1616 \cdot z^{-1} + 0.0960 \cdot z^{-2}}{1 - 0.9516 \cdot z^{-1} + 0.2093 \cdot z^{-2}} = \frac{0.1616 \cdot z + 0.0960}{z^2 - 0.9516 \cdot z + 0.2093} \quad (4.13)$$

La respuesta en el tiempo para esta planta sin considerar el proceso de ruido se observa en la figura 4.16.

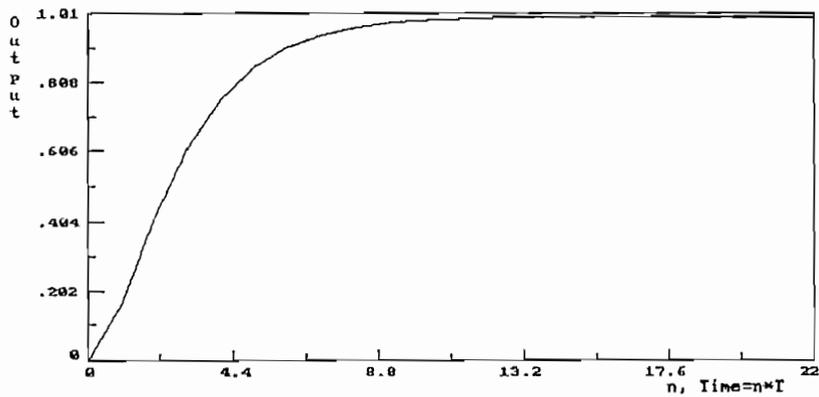


Figura 4.16. Respuesta en el tiempo en lazo abierto de la planta del ejemplo 6.

Y el proceso de ruido con la función de transferencia :

$$G_r(z) = \frac{1}{1 - 0.5 \cdot z^{-1}} = \frac{z}{z - 0.5} \quad (4.14)$$

La figura 4.17 muestra el ruido blanco de entrada y la figura 4.18 la salida del proceso de ruido.

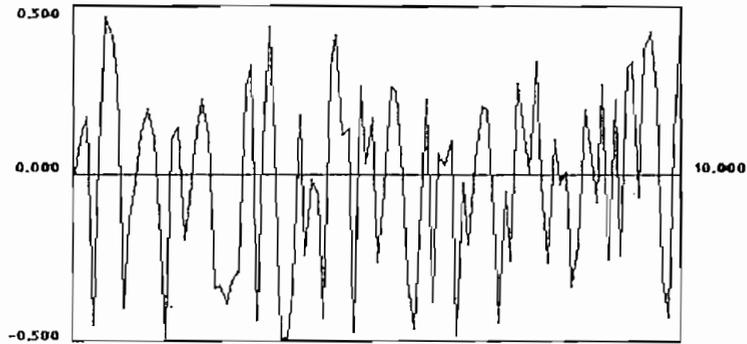


Figura 4.17. Ruido blanco de entrada al proceso de ruido del ejemplo 6.

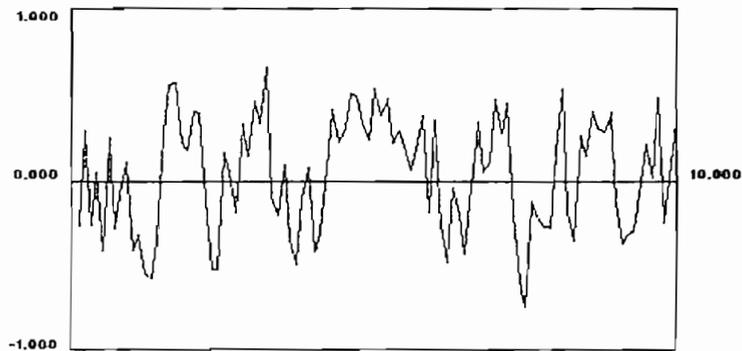


Figura 4.18. Salida del proceso de ruido del ejemplo 6.

Para este sistema se realizaron las pruebas de seguimiento para un retardo $d = 0$, $d = 1$ y $d = 2$; con factor de ponderación igual a cero y varianza del ruido igual a uno. Los resultados de este ejemplo se encuentran impresos en las páginas siguientes.

CONTROL DE MINIMA VARIANZA - REPORTE DE DATOS

DATOS DEL SISTEMA

Orden de los polinomios = 2
Factor de ponderación = 0.00000
Retardo $d = 0$
Varianza del ruido = 1.00000

Coefficientes de los polinomios :

$a_0 = 1.00000$	$b_0 = 0.00000$
$a_1 = -0.95166$	$b_1 = 0.16166$
$a_2 = 0.20933$	$b_2 = 0.09601$

$c_0 = 1.00000$	$d_0 = 1.00000$
$c_1 = 0.00000$	$d_1 = -0.50000$
$c_2 = 0.00000$	$d_2 = 0.00000$

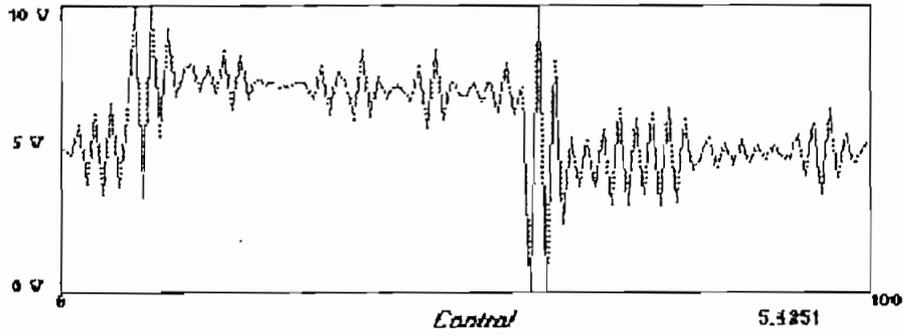
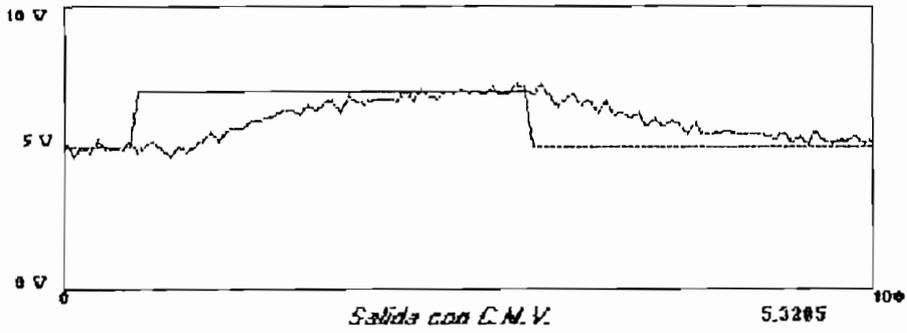
RESULTADOS

Coefficientes de los polinomios F y G :

$f_0 = 1.00000$	$g_0 = 0.50000$
$f_1 = 0.00000$	$g_1 = 0.00000$
$f_2 = 0.00000$	

Varianza de la salida controlada (teórica) = 1.00000
Varianza de la salida controlada (experimental) = 0.97765
Varianza de la salida sin control (experimental) = 1.26909
Varianza del ruido blanco (experimental) = 0.97765
Varianza del proceso de ruido (experimental) = 1.90149

CONTROL DE MINIMA VARIANZA - REPORTE DE GRAFICOS



CONTROL DE MINIMA VARIANZA - REPORTE DE GRAFICOS

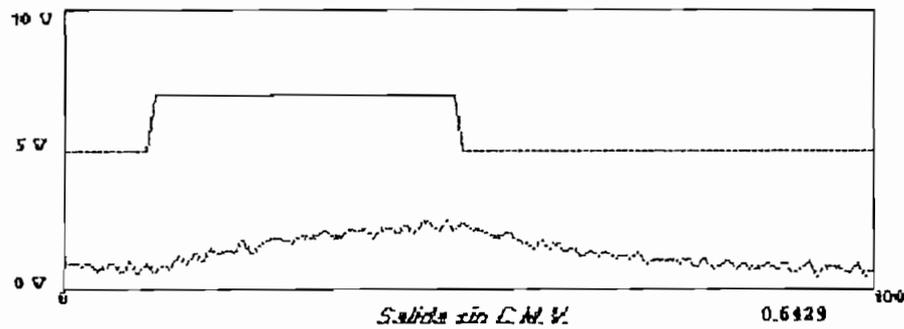
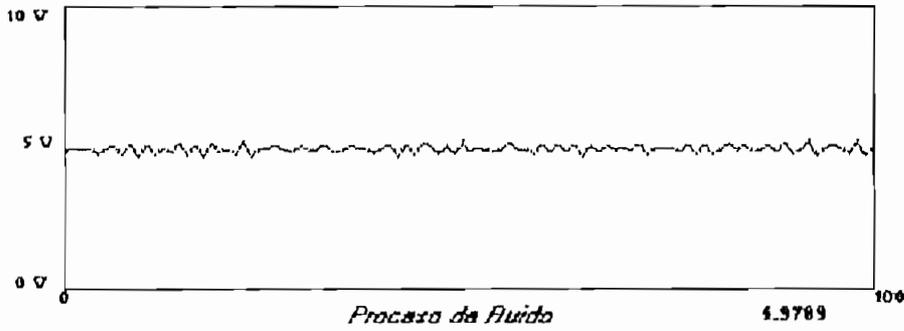


Figura EJE 6.2

CONTROL DE MINIMA VARIANZA - REPORTE DE DATOS

DATOS DEL SISTEMA

Orden de los polinomios = 2
Factor de ponderación = 0.00000
Retardo $d = 2$
Varianza del ruido = 1.00000

Coefficientes de los polinomios :

$a_0 = 1.00000$	$b_0 = 0.00000$
$a_1 = -0.95166$	$b_1 = 0.16166$
$a_2 = 0.20933$	$b_2 = 0.09601$

$c_0 = 1.00000$	$d_0 = 1.00000$
$c_1 = 0.00000$	$d_1 = -0.50000$
$c_2 = 0.00000$	$d_2 = 0.00000$

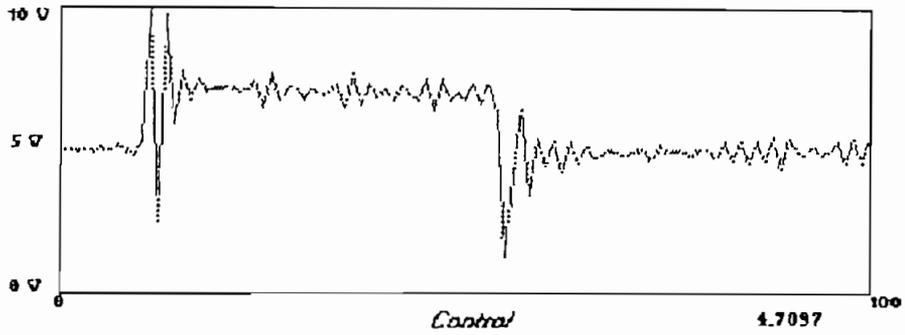
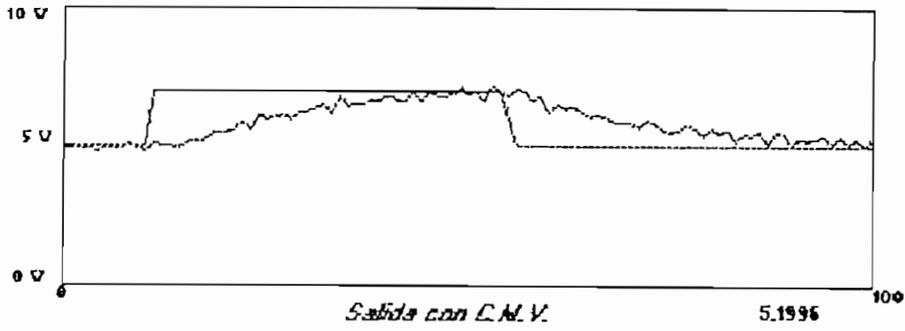
RESULTADOS

Coefficientes de los polinomios F y G :

$f_0 = 1.00000$	$g_0 = 0.12500$
$f_1 = 0.50000$	$g_1 = 0.00000$
$f_2 = 0.25000$	

Varianza de la salida controlada (teórica) = 1.31250
Varianza de la salida controlada (experimental) = 1.28962
Varianza de la salida sin control (experimental) = 1.31756
Varianza del ruido blanco (experimental) = 0.99924
Varianza del proceso de ruido (experimental) = 2.04620

CONTROL DE MINIMA VARIANZA - REPORTE DE GRAFICOS



CONTROL DE MINIMA VARIANZA - REPORTE DE GRAFICOS

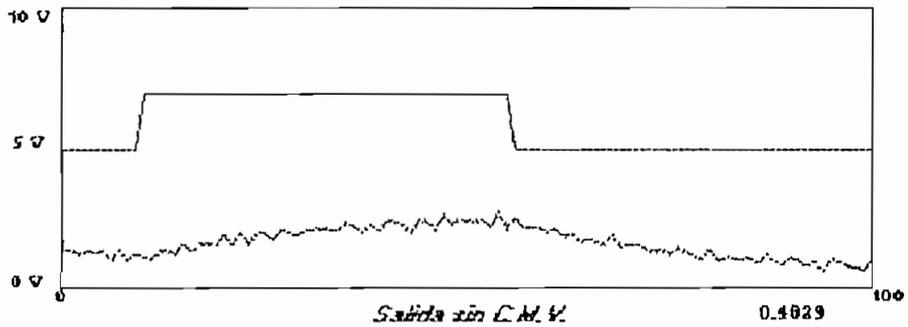


Figura EJE 6.3

4.3 CONCLUSIONES

El programa implementado entrega resultados satisfactorios y se han cumplido con los objetivos planteados al inicio del trabajo.

El controlador de mínima varianza obtenido no solo mejora la varianza de la salida, sino que también mejora la respuesta en el tiempo de las plantas, como se puede observar en los resultados.

Se mejora el tiempo de establecimiento, el mismo que depende del retardo de la planta. Si se trabaja en simulación y con sistemas sin tiempo muerto ($d=0$), se verifica en la figura EJE 1.3 que la salida controlada sigue a la señal de referencia con una rapidez tal que se puede decir que se tiene un tiempo de establecimiento prácticamente cero; en contraste con la salida sin control, que tiene un tiempo de establecimiento diferente de cero. Para sistemas con tiempo muerto ($d=1$ y $d=2$), se puede observar que el tiempo de establecimiento es aproximadamente igual al retardo d por el período de muestreo, siendo en todo caso más rápida la respuesta que para la planta sin control. Cuando se trabaja en tiempo real, la respuesta transitoria de la salida controlada es más lenta que para simulación, aunque mejor al caso de la salida sin control en tiempo real.

Como se puede observar en los resultados de las secciones 4.1 y 4.2, no existe sobreimpulsos en la respuesta de la salida controlada ante cambios de tipo escalón en la variable de referencia. Los picos que se observan se deben al proceso de ruido.

Por último, el error en estado estable es eliminado con la utilización del algoritmo de la sección 2.2.4. Esto se lo puede observar también en las secciones 4.1 y 4.2.

La varianza mínima de la salida aumenta con el retardo de la planta. A mayor retardo, mayor será la varianza de la salida (como se puede observar en los gráficos de las secciones 4.1). La varianza aumenta al incrementarse el retardo debido a que, para sistemas con tiempo muerto se debe predecir los valores futuros de la salida del proceso de ruido con el fin de calcular la señal de control óptima. Al aumentar el retardo se debe predecir un mayor número de pasos hacia adelante, teniéndose resultados menos aceptables.

El control de mínima varianza no elimina las perturbaciones aleatorias, sino que las minimiza; siendo la varianza mínima posible la varianza del ruido blanco de entrada al proceso de ruido (λ^2). En el caso de tener ruido correlacionado, el algoritmo de mínima varianza permite reducir la varianza de la salida al mínimo del ruido blanco. Si el ruido que afecta a la planta es ruido blanco (ver ejemplo 4) no es

posible minimizar la variable de la salida, puesto que es la mínima varianza posible de las perturbaciones aleatorias.

Para la implementación práctica, la modelación matemática del ruido blanco debería hacerse con los polinomios A y D iguales, ya que para este caso no existen restricciones respecto a la ubicación de polos y ceros de la planta, por lo tanto del tipo de sistema; y además el orden de los polinomios numerador y denominador del controlador son más bajos que para el caso en que los polinomios A y D sean diferentes, lo cual hace que las ecuaciones de diferencias para la implementación del control sean más fáciles de manejar.

En tiempo real, la respuesta transitoria de las plantas implementadas en los ejemplos 5 y 6 no es tan satisfactoria como para los ejemplos de simulación. Debido a que en simulación se calcula la salida de la planta a nivel de software, en cambio en tiempo real es el prototipo de la planta el que proporciona la salida.

En tiempo real, los valores de las señales de salida controlada y del control son limitados por el voltaje de trabajo de la tarjeta DAS - 128 el cual es de cero a diez voltios. Esto implica que al estar limitada la señal de control no se obtenga la forma de onda de control requerida por la planta para su óptimo funcionamiento, ocasionando una respuesta transitoria más lenta de la que se tiene en simulación.

BIBLIOGRAFÍA

1. ISERMANN ROLF, "Digital Control Systems" , Editorial Springer - Verlag, Alemania, 1981.
2. ÅSTRÖM KARL J. , "Introduction to Stochastic Control Theory", Editorial Academic Press, U.S.A. , 1970.
3. EDITADO POR WARWICK K. Y REES D. , "Industrial Digital Control Systems", Editorial Peter Peregrinus Ltd., Inglaterra, 1988.
4. DAVIS H.M.A. Y VINTER R.B. , "Stochastic Modelling and Control" , Editorial Chapman and Hall, Inglaterra, 1984.
5. EDITADO POR HARRIS C.J. Y BILLINGS S.A. , "Self - Tuning and Adaptive Control : Theory and Applications" , Editorial Peter Peregrinus Ltd., Inglaterra, 1981.
6. EDITADO POR GELB ARTHUR, "Applied Optimal Estimation" , Editorial The M.I.T. Press, U.S.A., 1974.
7. BENDAT S. JULIUS Y PIERSOL G. ALLAN, "Random Data: Analysis and Measurement Procedures", Editorial Wiley - Interscience, U.S.A., 1971.

8. MICROSOFT VISUAL BASIC Version 3.0, "Programmer's Guide", Microsoft Corporation, U.S.A., 1993.
9. MICROSOFT VISUAL BASIC Version 3.0, "Lenguaje Reference", Microsoft Corporation, U.S.A., 1993.
10. HINOJOSA RAZA MIGUEL, "Control computarizado de temperatura de un caldero asociado a un intercambiador de calor", Escuela Politécnica Nacional - Universidad Andina Simón Bolívar Subsede Ecuador, 1995.
11. CEVALLOS FAUSTO, "Diseño y construcción de una tarjeta de adquisición de datos para computadores personales", Escuela Politécnica Nacional, 1992.
12. ZÁRATE MORA BYRON, "Identificación paramétrica discreta", Escuela Politécnica Nacional, 1996.

APÉNDICE A : MANUAL DEL USUARIO

El programa *Control de Mínima Varianza 1.0* fue escrito en lenguaje *Visual Basic 3.0*, por lo que el ambiente de trabajo es de *Windows*; es decir, en base a pantallas y cajas de diálogo. Está hecho de tal forma que no requiere ayuda adicional a este *Manual del usuario*.

Los requerimientos de hardware y software para este programa son :

- Cualquier computadora compatible con IBM con un procesador 80286 o superior.
- Un monitor EGA, VGA, 8514, Hercules, o compatible.
- Un megabyte de memoria RAM.
- Un mouse.
- Microsoft MS - DOS 3.1 o superior.
- Windows 3.0 o versiones superiores.

A.1 INGRESO AL PROGRAMA Y PANTALLA PRINCIPAL

Para ingresar al programa se hace un clic en el icono mostrado en la figura A1.



Figura A1. Icono correspondiente al Control de Mínima Varianza.

El menú de la pantalla principal está compuesto por los siguientes ítems (ver figura A.2) :

- Modo de trabajo
- Acerca del programa, y
- Salir del programa

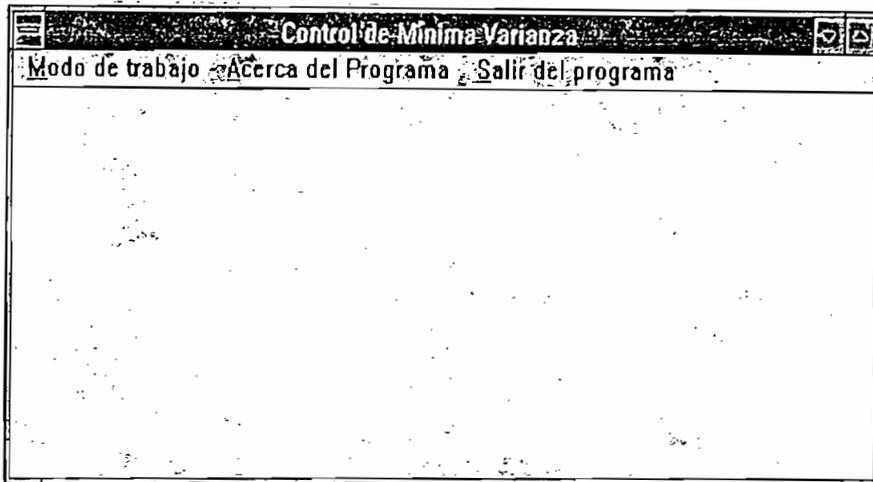


Figura A.2. Pantalla principal del *Modo de Tiempo Real*.

El ítem *Acerca del programa* invoca una pantalla con información del programa (ver figura A.3).

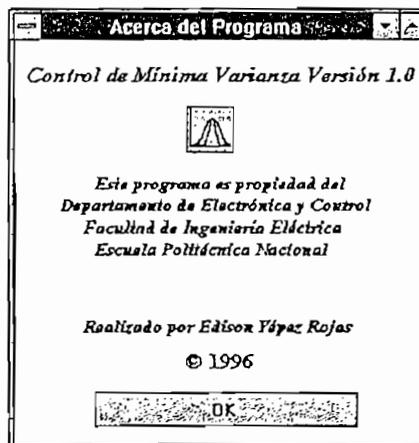


Figura A.3. Pantalla de Información del Programa.

El ítem Salir del programa finaliza la sesión de trabajo con el programa.

El ítem Modo de trabajo invoca una pantalla del mismo nombre que consta de las siguientes opciones (ver figura A.4) :

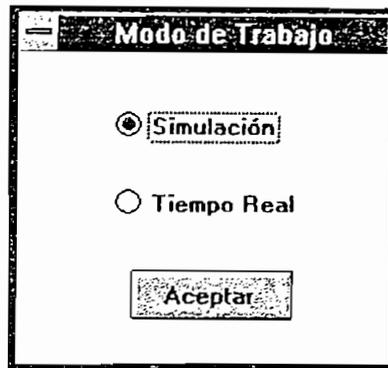


Figura A.4 .Pantalla para seleccionar el Modo de Trabajo

- **Simulación, y**
- **Tiempo real**

Al escoger alguna de estas opciones se ingresa al modo de trabajo respectivo. Debido a que tanto para simulación, como para tiempo real se utilizan exactamente las mismas pantallas y subrutinas se hará una única descripción de las mismas.

A.2 PANTALLAS DE SIMULACIÓN Y TIEMPO REAL

Una vez que se ha ingresado al modo de simulación o de tiempo real, se tiene una pantalla cuyo menú principal tiene los siguientes ítems (ver figura A.5 y figura A.6) :

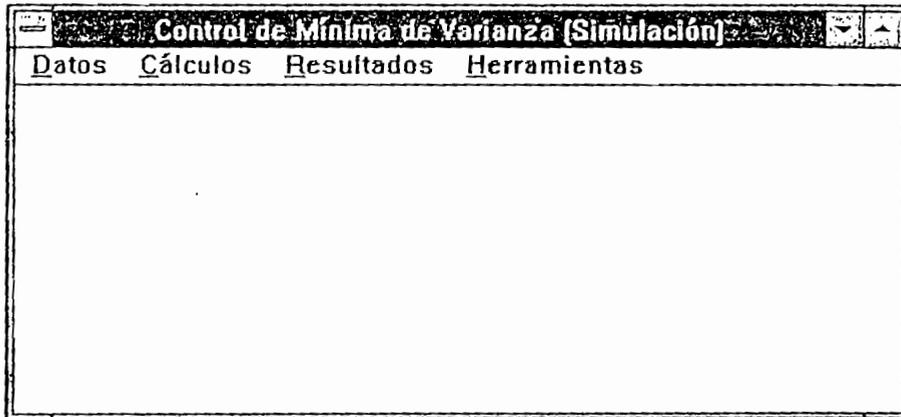


Figura A.5. Pantalla principal del Modo de Simulación.

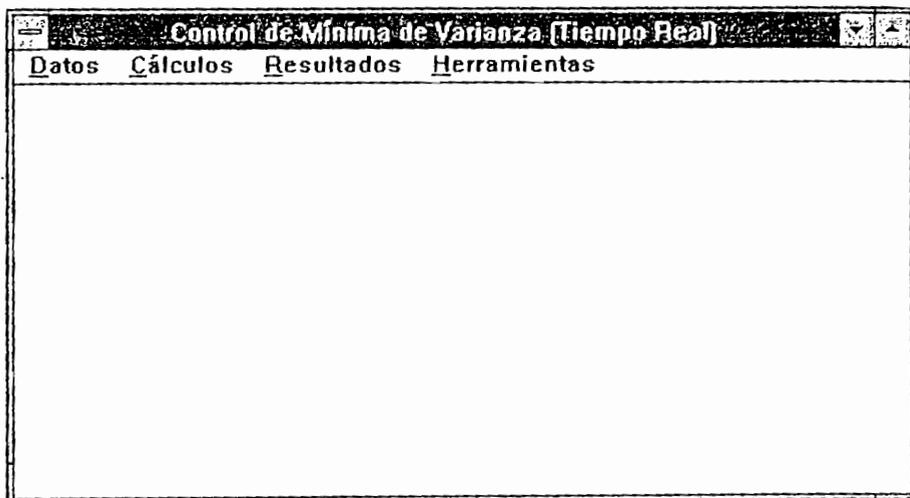


Figura A.6. Pantalla principal del Modo de Tiempo Real.

- Datos
- Cálculos
- Resultados, y
- Herramientas

El ítem Resultados despliega un submenú con las siguientes opciones (ver figura A.8) :

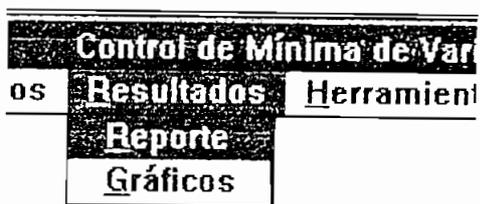


Figura A.8. Submenú del ítem Resultados

- **Reporte.** Invoca una pantalla en la cual se encuentra la información más importante del sistema.
- **Gráficos.** Invoca la pantalla de menú de gráficos que permite seleccionar los gráficos a ser visualizados.

El ítem Herramientas despliega un submenú con las siguientes opciones (ver figura A.9) :

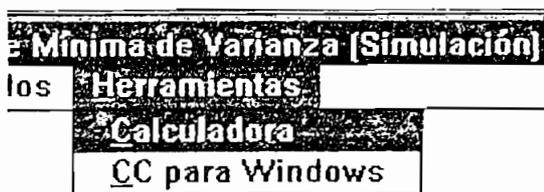


Figura A.9. Submenú del ítem Herramientas

La pantalla *Datos nuevos* (ver figura A.11) permite el ingreso del orden de los polinomios A, B, C y D, de los coeficientes de los mismos, del factor de ponderación r , del retardo (si es el caso), y de la varianza del ruido.

Ingreso de los datos

Orden de los polinomios : 2

Polinomio A
 Polinomio B
 Polinomio C
 Polinomio D

Valor del coeficiente de orden : 1

Anterior

Siguiete

Factor de ponderación r : 0

Retardo d : 0

Varianza del ruido = 1

Aceptar datos Cancelar

Figura A.11. Pantalla de *Ingreso de datos*

Para aumentar o disminuir el orden de los polinomios se cuenta con una barra de desplazamiento vertical. Para seleccionar el polinomio al que se desea ingresar los coeficientes se dispone de cuatro opciones. Para el ingreso del valor numérico de los coeficientes se cuenta con una caja de texto. Los botones Anterior y Siguiete permiten seleccionar el coeficiente del polinomio que se desea chequear o escribir. El factor de ponderación se lo ingresa a través de una caja de texto. El retardo se lo ingresa a través de una barra de desplazamiento vertical. Y por último la varianza del ruido se la ingresa a través de una caja de texto.

La pantalla Archivar datos (ver figura A.12) permite guardar en un archivo la información de el sistema con el que se está trabajando. El archivo creado tendrá la extensión .CMV. Para seleccionar el lugar donde será almacenado se dispone de una lista de directorios que permite moverse en los directorios de la unidad activa; de una lista de unidades que permite cambiar de unidad si es necesario y de una caja de texto donde se escribirá el nombre del archivo a ser creado o modificado. Si se desea sobrescribir un archivo existente se hace un click sobre el mismo en el recuadro lista de archivos y el nombre del archivo seleccionado se visualizará en el recuadro Nombre de archivo.

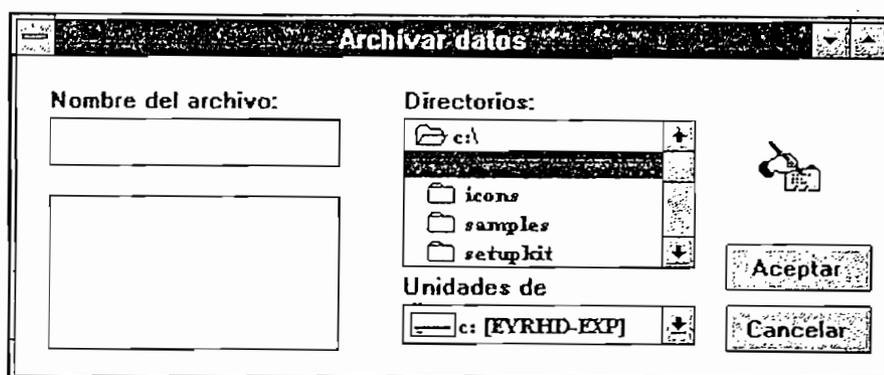


Figura A.12. Pantalla para Archivar datos

La pantalla Cargar datos (ver figura A.13) permite “navegar” a través de los directorios del disco duro y unidades de disco en búsqueda de los archivos con extensión .CMV. Para seleccionar el archivo se hace un click sobre su nombre en el recuadro Lista de archivos, el mismo que se visualizará en el recuadro Nombre del archivo. Para aceptar el archivo se hace un click en el botón Aceptar, caso contrario se presiona el botón Cancelar que oculta la pantalla y retorna a la pantalla anterior sin cargar ningún dato. Todos los ejemplos que se presentan en las secciones 4.1 y 4.2 se encuentran grabados en el directorio Ejemplos.

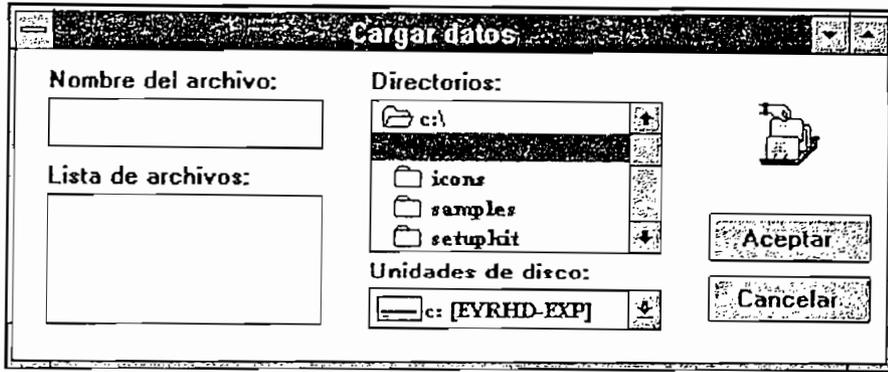


Figura A.13. Pantalla para Cargar datos

A.4 PRESENTACIÓN DE RESULTADOS

Reporte. Los resultados se presentan de dos maneras, la primera a través del *Reporte* y la segunda a través de los *Gráficos*.

La pantalla de *Reporte* (ver figura A.14) contiene un menú de Opciones, el mismo que tiene los siguientes ítems (ver figura A.15):

- **Imprimir.** Permite mandar a impresora los datos contenidos en el reporte.
- **Salir.** Oculta esta pantalla y retorna a la anterior.

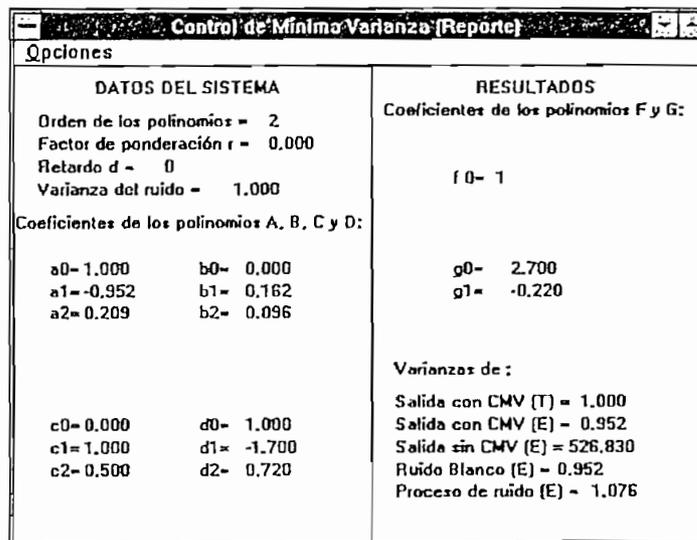


Figura A.14. Pantalla de Reporte de datos

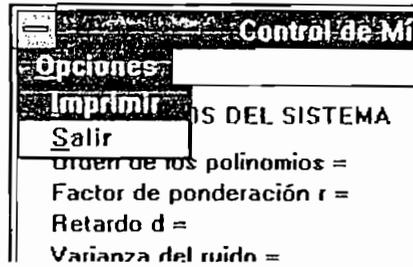


Figura A.15. Submenú de Opciones de la pantalla de Reporte.

Gráficos. La pantalla Menú de gráficos (ver figura A.16) permite escoger los gráficos a ser visualizados. Al hacer un clic en el botón *Visualizar* se invoca a la pantalla *Gráficos* (ver figura A.17), la misma que contiene un menú con las siguientes opciones (ver figura A.18) :

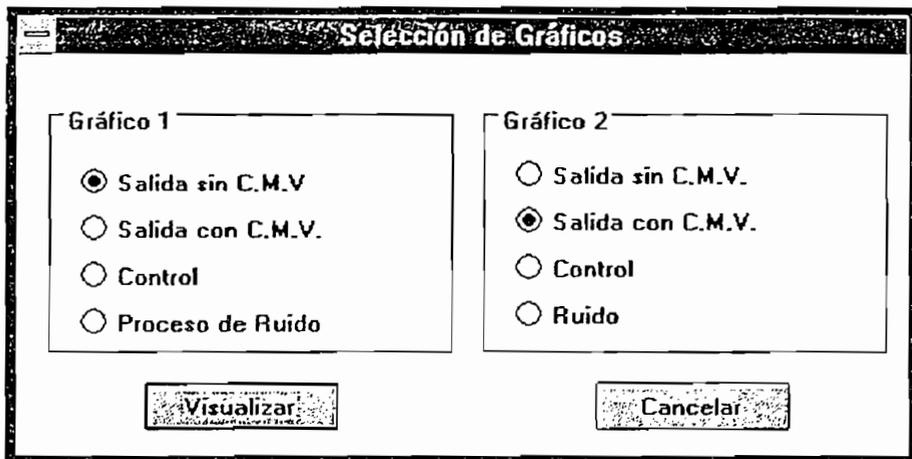


Figura A.16. Pantalla de Menú de gráficos

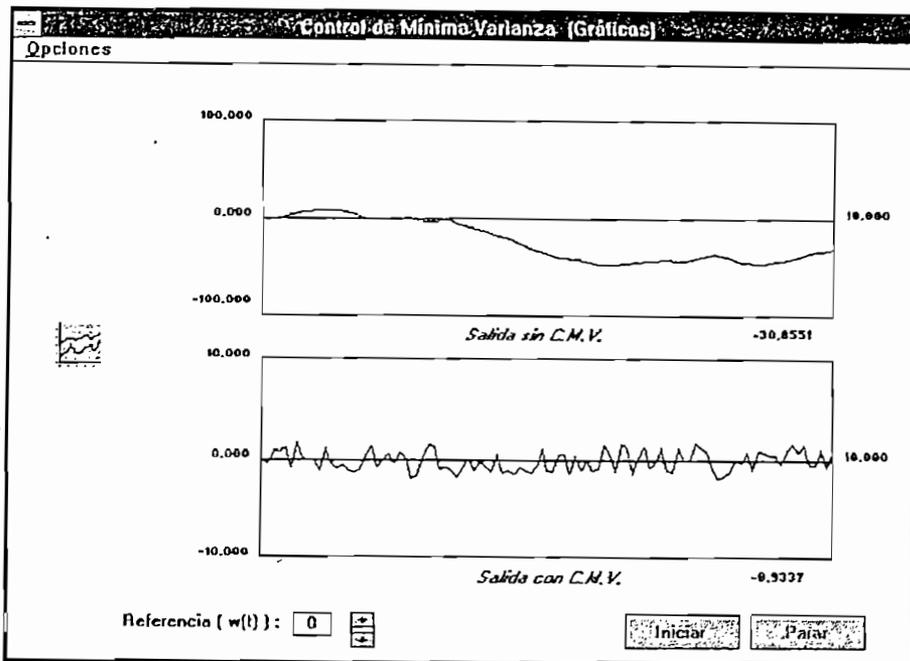


Figura A.17. Pantalla de Gráficas

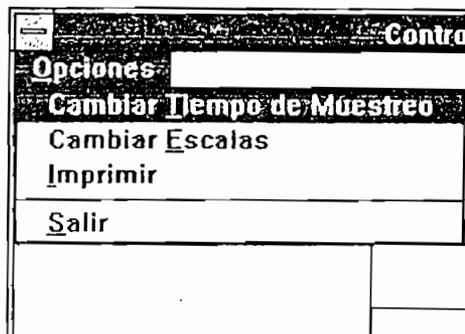
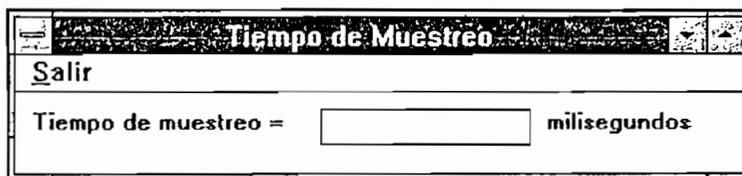


Figura A.18. Submenú de Opciones de la pantalla de Gráficas

- **Cambiar tiempo de muestreo.** Invoca a la pantalla del mismo nombre.
- **Cambiar escalas.** Invoca a la pantalla del mismo nombre.
- **Imprimir.** Envía a la impresora las curvas que se estén visualizando en ese momento.
- **Salir.** Oculta esta pantalla y retorna el control a la pantalla anterior.

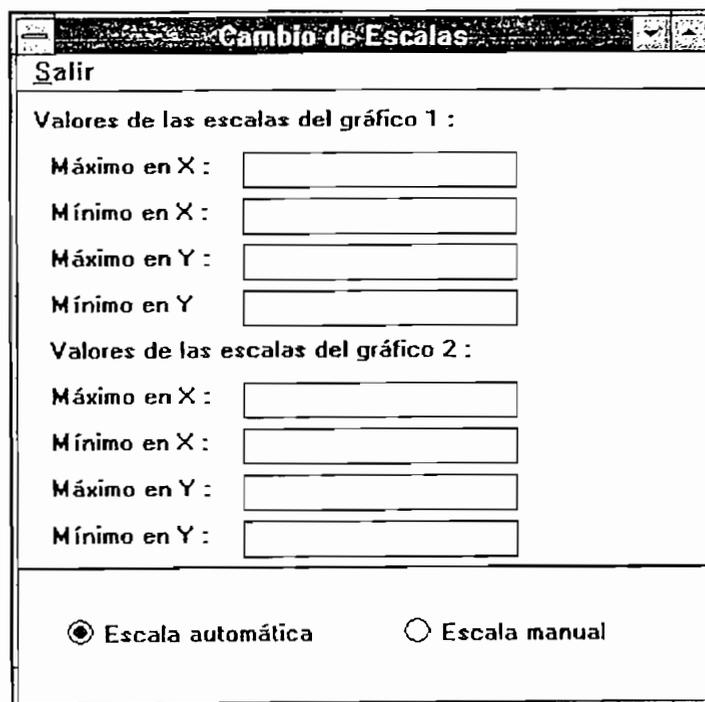
La pantalla Cambiar tiempo de muestreo (ver figura A.19) tiene un cuadro de texto donde se escribirá el nuevo tiempo de muestreo. Para salir de la misma se hace un clic en la opción *Salir*, quedando el nuevo dato cargado en memoria.



The screenshot shows a dialog box titled "Tiempo de Muestreo". At the top left is a "Salir" button. Below it, the text "Tiempo de muestreo =" is followed by a text input field, and then the text "milisegundos".

Figura A.19. Pantalla de Cambio de Tiempo de muestreo

La pantalla Cambiar escalas (ver figura A.20) tiene dos opciones : *Escala automática* y *Escala manual*.



The screenshot shows a dialog box titled "Cambio de Escalas". At the top left is a "Salir" button. Below it, the text "Valores de las escalas del gráfico 1 :" is followed by four input fields labeled "Máximo en X :", "Mínimo en X :", "Máximo en Y :", and "Mínimo en Y :". Below these is the text "Valores de las escalas del gráfico 2 :" followed by another four input fields labeled "Máximo en X :", "Mínimo en X :", "Máximo en Y :", and "Mínimo en Y :". At the bottom, there are two radio buttons: "Escala automática" (which is selected) and "Escala manual".

Figura A.20. Pantalla de Cambio de escalas

Cuando se trabaja en escala automática no se puede cambiar los valores mostrados en las cajas de texto. En cambio cuando se trabaja en escala manual es factible cambiar los valores de las cajas de texto. Para salir de esta pantalla se hace un clíck en la opción *Salir*, quedando los nuevos datos almacenados en memoria.

Para iniciar la graficación de las señales seleccionadas se hace un clíck en el botón *Iniciar* . Si se está trabajando en seguimiento la barra de desplazamiento vertical estará habilitada permitiendo cambiar la referencia de la salida. Para suspender la graficación se hace un clíck en el botón *Parar*.

A.5 TRABAJANDO EN TIEMPO REAL

En tiempo real, la señal de control $u(t)$ se la envía hace a través de la salida análoga VO6 de la tarjeta de adquisición de datos DAS - 128 y la señal de salida de la planta $yp(t)$ se la mide a través de la entrada VI6 de la misma tarjeta.

Debe recordarse que el rango de trabajo de los voltajes de las salidas y entradas analógicas de la tarjeta DAS - 128 es de cero a diez voltios, y que cualquier sobrevoltaje puede dañar la misma.

Cuando se trabaja en tiempo real debe tomarse en cuenta que no es posible visualizar las señales salida con control y salida con control al mismo tiempo.

Las plantas que se pueden manejar con el presente programa son plantas hasta de cuarto orden. Aunque la planta tenga un retardo nulo ($d = 0$), por software es posible darle hasta dos retardos ($d = 2$), para esto simplemente se escoge el retardo que se desee en la pantalla de ingreso de datos (ver figura A .11).

APÉNDICE B : LISTADO DEL PROGRAMA

B.1 DECLARACION DE VARIABLES GLOBALES

Global TiempoReal, Varcontrol As Integer

Global ModoTrabajo, m, n, d, opciongrafo1, opciongrafo2 As Integer

Global tiempo1, tiempo2, tiempo1t, tiempo2t, tiempo1tm1, tiempo2tm1 As Double

Global Kp, r, VarRuido, VarSalida As Double

Global a0, a1, a2, a3, a4 As Double

Global b0, b1, b2, b3, b4 As Double

Global c0, c1, c2, c3, c4 As Double

Global d0, d1, d2, d3, d4 As Double

Global f0, f1, f2, g0, g1, g2, g3 As Double

Global grafo1x1, grafo2x1, maxy1, maxy2, maxy1tm1, maxy2tm1 As Double

Global yst, ystm1, ystm2, ystm3, ystm4, ystm5, ystm6, ystm7, ystm8, ystm9, ystm10 As Double

Global yct, yctm1, yctm2, yctm3, yctm4, yctm5, yctm6, yctm7, yctm8, yctm9, yctm10 As Double

Global vt, vtm1, et, etm1, etm2, etm3, etm4, etm5, etm6, etm7, etm8 As Double

Global ut, utm1, utm2, utm3, utm4, utm5, utm6, utm7, utm8, utm9, utm10 As Double

Global ust, ustm1, ustm2, ustm3, ustm4, ustm5, ustm6, ustm7, ustm8, ustm9, ustm10 As Double

Global wt, wtm1, wtm2, wtm3, wtm4, wtm5, wtm6, wtm7, wtm8 As Double

Global grafo1xmin, grafo1xmax, grafo1ymin, grafo1ymax, grafo2xmin, grafo2xmax, grafo2ymin, grafo2ymax As Double

Global Coef(7), Coef_A(7), Coef_B(7), Coef_C(7), Coef_D(7) As Double

Global M1(11), N1(11), Q0(11), Q1(12), Q2(13) As Double

Global BF1, BF2, BF3, BF4, BF5, BF6 As Double

Global TU1, TU2, TU3, TU4, TU5, TU6, TU7, TU8, TU9, TU10 As Double

Global AC1, AC2, AC3, AC4, AC5, AC6, AC7, AC8 As Double

Global GCD1, GCD2, GCD3, GCD4, GCD5, GCD6, GCD7 As Double

Global AG1, AG2, AG3, AG4, AG5, AG6, AG7 As Double

Global AD1, AD2, AD3, AD4, AD5, AD6, AD7, AD8 As Double

Global BD1, BD2, BD3, BD4, BD5, BD6, BD7, BD8 As Double

Declare Function Medir Lib "Ioport.dll" (ByVal PUERTO As Integer, ByVal MEDIDA As Integer) As Integer

Declare Function Enviar Lib "Ioport.dll" (ByVal PUERTOS As Integer, ByVal VCONTROL As Integer) As Integer

B.2 SUBRUTINAS DE LA PANTALLA DE PRESENTACIÓN

```
Sub Form_Load ()
```

```
    Timer1.Interval = 2000  
    Timer1.Enabled = True
```

```
End Sub
```

```
Sub Timer1_Timer ()
```

```
    Conminva.Show 1
```

```
End Sub
```

B.3 SUBRUTINAS DE LA PANTALLA PRINCIPAL

```
Sub AcercaDe_Click ()
```

```
    Acerca.Show 1
```

```
End Sub
```

```
Sub Form_Load ()
```

```
    Presentacion.Timer1.Enabled = False  
    Presentacion.Hide
```

```
End Sub
```

```
Sub Modotrabajo_Click ()
```

```
    PANTMODTRABAJO.Show 1
```

```
End Sub
```

```
Sub Salirprograma_Click ()
```

```
    End
```

```
End Sub
```

```
Sub Simulacion_Click ()
```

```
    TIEMPOREAL = 0  
    Simprin.Show 1
```

```
End Sub
```

```

Sub Tiemporeal_Click ()
    TIEMPOREAL = 1
End Sub

Sub Timer1_Timer ()
    Timer1.Enabled = False
End Sub

Sub UsoPrograma_Click ()
    Ayudaprin.Show 1
End Sub

```

B.4 SUBROUTINAS DE LA PANTALLA DE SELECCIÓN DE MODO DE TRABAJO

```

Sub Aceptarmodo_Click ()
    If Tiemporeal = 0 Then
        Simprin.Show 1
    ElseIf Tiemporeal = 1 Then
        Simprin.Caption = "Control de Mínima Varianza (Tiempo Real)"
        Simprin.Show 1
    End If
    PantModTrabajo.Hide
End Sub

Sub Command1_Click ()
End Sub

Sub Form_Load ()
    Tiemporeal = 0
End Sub

Sub Option1_Click ()
    Tiemporeal = 0
End Sub

Sub Option2_Click ()
    Tiemporeal = 1
End Sub

```

B.5 SUBROUTINAS DE LA PANTALLA ACERCA DEL PROGRAMA

```
Sub Cerraracerca_Click ()  
    Acerca.Hide  
End Sub
```

B.6 SUBROUTINAS DE LA PANTALLA SECUNDARIA

```
Sub Archivar_datos_Click ()
```

```
    Guardardat.Show 1
```

```
End Sub
```

```
Sub calcul_Click ()
```

```
wertx = Shell("calc.exe", 1)
```

```
End Sub
```

```
Sub Cardatos_Click ()
```

```
End Sub
```

```
Sub Cargardatos_Click ()
```

```
Cargardat.Show 1
```

```
End Sub
```

```
Sub Datosnuevos_Click ()
```

```
    ingredat.Show 1
```

```
End Sub
```

```
Sub Form_Load ()
```

```
    'Inicializo el valor de las variables
```

```
    n = 1
```

```
    d = 0
```

```
    R = 0
```

```
    VarRuido = 1
```

```
    a1 = 0
```

```
    a2 = 0
```

```
    a3 = 0
```

```
    a4 = 0
```

```
    a0 = 0
```

```
    b0 = 0
```

```
    b1 = 0
```

```
    b2 = 0
```

```
    b3 = 0
```

```
    b4 = 0
```

```
    b0 = 0
```

```
    c1 = 0
```

```
    c2 = 0
```

```
c3 = 0
c4 = 0
c0 = 0
d1 = 0
d2 = 0
d3 = 0
d4 = 0
d0 = 0
f1 = 0
f2 = 0
g0 = 0
g1 = 0
g2 = 0
g3 = 0
```

'Inicializo el estado de las opciones del menu

```
Datosnuevos.Enabled = False
Archivardatos.Enabled = False
Simulacion.Enabled = False
Resultados.Enabled = False
```

End Sub

```
Sub Graficos_Click ()
'graficos.Show 1
End Sub
```

```
Sub Graficosim_Click ()
```

```
opciongrafo1 = 1
opciongrafo2 = 2
```

```
MenuGraficos.Show 1
```

End Sub

```
Sub Guardardatos_Click ()
```

```
Guardardat.Show 1
```

End Sub

```
Sub pipe_Click ()
```

```
qsdX = Shell("c:\ccw2\ccw2.exe", 1)
```

End Sub

```
Sub Reporte_Click ()
```

```
Reporte.Show 1
End Sub
```

```
Sub Reportesim_Click ()
```

```
Reporte.Show 1
```

End Sub

```
Sub Salir_Click ()
```

```
Unload Simprin
conminva.Tiemporeal.enabled = True
End Sub
```

```
Sub Seleccion_Click ()

selecal.Show 1
Simprin.Datosnuevos.enabled = True

End Sub
```

```
Sub Simulacion_Click ()

Simulacion
Simprin.Resultados.enabled = True

End Sub
```

B.7 SUBRUTINAS DE LA PANTALLA DE SELECCIÓN DE ALGORITMOS

```
Sub Aceptar_Click ()

If (Regulacion = True) Then

    Graficos.DespVSetPoint.Enabled = False
    Graficos.SetPoint.Enabled = False

If (TiempoMuerto = 0) Then

    Ingredat.DesVRetardo.Enabled = False

    If (AigualD = 0) Then

        Ingredat.PolinomioD.Enabled = True
        Modotrabajo = 100

    ElseIf (AigualD = 1) Then

        Ingredat.PolinomioD.Enabled = False
        Modotrabajo = 101

    End If

ElseIf (TiempoMuerto = 1) Then

    Ingredat.DesVRetardo.Enabled = True

    If (AigualD = 0) Then

        Ingredat.PolinomioD.Enabled = True
        Modotrabajo = 110

    ElseIf (AigualD = 1) Then

        Ingredat.PolinomioD.Enabled = False
        Modotrabajo = 111
```

```

    End If

    End If

    End If

    If (Seguimiento = True) Then

        Graficos.DespVSetPoint.Enabled = True
        Graficos.SetPoint.Enabled = True

        If (TiempoMuerto = 0) Then

            Ingedat.DesVRetardo.Enabled = False

            If (AigualD = 0) Then

                Ingedat.PolinomioD.Enabled = True
                Modotrabajo = 200

            Elseif (AigualD = 1) Then

                Ingedat.PolinomioD.Enabled = False
                Modotrabajo = 201

            End If

        Elseif (TiempoMuerto = 1) Then

            Ingedat.DesVRetardo.Enabled = True

            If (AigualD = 0) Then

                Ingedat.PolinomioD.Enabled = True
                Modotrabajo = 210

            Elseif (AigualD = 1) Then

                Ingedat.PolinomioD.Enabled = False
                Modotrabajo = 211

            End If

        End If

    End If

    End Sub

    Selecal.Hide

    End Sub

    Sub Cancelar_Click ()

        Unload Selecal

    End Sub

```

```

Sub Command1_Click ()
    Seleccion.Hide
End Sub

Sub Form_Load ()
    If Modotrabajo = 100 Then
        Regulacion.Value = True
        TiempoMuerto.Value = 0
        AigualD.Value = 0

    ElseIf Modotrabajo = 101 Then
        Regulacion.Value = True
        TiempoMuerto.Value = 0
        AigualD.Value = 1

    ElseIf Modotrabajo = 110 Then
        Regulacion.Value = True
        TiempoMuerto.Value = 1
        AigualD.Value = 0

    ElseIf Modotrabajo = 111 Then
        Regulacion.Value = True
        TiempoMuerto.Value = 1
        AigualD.Value = 1

    End If

    If Modotrabajo = 200 Then
        Seguimiento.Value = True
        TiempoMuerto.Value = 0
        AigualD.Value = 0

    ElseIf Modotrabajo = 201 Then
        Seguimiento.Value = True
        TiempoMuerto.Value = 0
        AigualD.Value = 1

    ElseIf Modotrabajo = 210 Then
        Seguimiento.Value = True
        TiempoMuerto.Value = 1
        AigualD.Value = 0

    ElseIf Modotrabajo = 211 Then
        Seguimiento.Value = True
        TiempoMuerto.Value = 1
        AigualD.Value = 1

    End If
End Sub

```

```
Sub Regulacion_Click ()
```

```
    TiempoMuerto.Enabled = True  
    AigualD.Enabled = True
```

```
End Sub
```

B.8 SUBROUTINAS DE LA PANTALLA PARA INGRESO DE DATOS

```
Sub Aceptardatos_Click ()
```

```
    a0 = Coef_A(0)  
    a1 = Coef_A(1)  
    a2 = Coef_A(2)  
    a3 = Coef_A(3)  
    a4 = Coef_A(4)
```

```
    b0 = Coef_B(0)  
    b1 = Coef_B(1)  
    b2 = Coef_B(2)  
    b3 = Coef_B(3)  
    b4 = Coef_B(4)
```

```
    c0 = Coef_C(0)  
    c1 = Coef_C(1)  
    c2 = Coef_C(2)  
    c3 = Coef_C(3)  
    c4 = Coef_C(4)
```

```
    d0 = Coef_D(0)  
    d1 = Coef_D(1)  
    d2 = Coef_D(2)  
    d3 = Coef_D(3)  
    d4 = Coef_D(4)
```

```
    If (ModoTrabajo = 111 Or ModoTrabajo = 101 Or ModoTrabajo = 211 Or ModoTrabajo =  
201) Then
```

```
        d0 = a0  
        d1 = a1  
        d2 = a2  
        d3 = a3  
        d4 = a4
```

```
    End If
```

```
    f1 = 0  
    f2 = 0  
    g0 = 0  
    g1 = 0  
    g2 = 0  
    g3 = 0
```

```
    ingredat.PolinomioA.Value = True  
    Simprin.Archivardatos.Enabled = True  
    Simprin.Simulacion.Enabled = True  
    Simprin.Resultados.Enabled = False
```

Unload ingredat

End Sub

Sub Anterior_Click ()

Siguiente.Enabled = True

If m > 0 Then

m = m - 1

If m = 0 Then

Anterior.Enabled = False

Siguiente.Default = True

End If

End If

Numerocoef.Caption = m

Ingresocoef.SetFocus

If PolinomioA = True Then

Ingresocoef.Text = Format\$(Coef_A(m))

Elseif polinomioB = True Then

Ingresocoef.Text = Format\$(Coef_B(m))

Elseif polinomioC = True Then

Ingresocoef.Text = Format\$(Coef_C(m))

Elseif PolinomioD = True Then

Ingresocoef.Text = Format\$(Coef_D(m))

End If

End Sub

Sub Cancelar_Click ()

ingredat.PolinomioA.Value = True

Unload ingredat

End Sub

Sub Command2_Click ()

ingredat.Hide

End Sub

Sub DesVOrdenPol_Change ()

ordenpol.Caption = DesVOrdenPol.Value

n = Val(DesVOrdenPol.Value)

m = ordenpol.Caption

Numerocoef.Caption = m

Siguiente.Enabled = False

Anterior.Enabled = True

Anterior.Default = True

```

    Ingresocoef.Text = Format$(Coef_A(m))

End Sub

Sub DesVRetardo_Change ()

    Retardo.Caption = DesVRetardo.Value
    d = Val(DesVRetardo.Value)

End Sub

Sub FactorPonderacion_Change ()

    If FactorPonderacion.Text = "" Then

        r = 0
        Beep
        FactorPonderacion.Text = "0"

    Else

        r = Val(FactorPonderacion.Text)

    End If

End Sub

Sub FactorPonderacion_KeyPress (KeyAscii As Integer)

    c$ = Chr$(KeyAscii)

    If c$ < "0" Or c$ > "9" Then

        If Asc(c$) = 8 Then

            KeyAscii = 8

        ElseIf c$ <> "8" Then

            KeyAscii = 0

            Beep

        End If

    End If

End Sub

Sub Form_Load ()

    DesVOrdenPol.Value = n
    DesVRetardo.Value = d

    Coef_A(1) = a1
    Coef_A(2) = a2
    Coef_A(3) = a3
    Coef_A(4) = a4
    Coef_A(0) = a0

```

```
Coef_B(0) = b0
Coef_B(1) = b1
Coef_B(2) = b2
Coef_B(3) = b3
Coef_B(4) = b4
```

```
Coef_C(1) = c1
Coef_C(2) = c2
Coef_C(3) = c3
Coef_C(4) = c4
Coef_C(0) = c0
```

```
Coef_D(1) = d1
Coef_D(2) = d2
Coef_D(3) = d3
Coef_D(4) = d4
Coef_D(0) = d0
```

```
Ingresocoef.Text = Format$(Coef_A(n))
FactorPonderacion = Format$(r)
VarianzaRuido.Text = Format$(VarRuido)
```

```
'Inicializo los valores de los comandos y etiquetas
```

```
m = ordenpol.Caption
Numerocoef.Caption = m
Siguiente.Enabled = False
Anterior.Enabled = True
Anterior.Default = True
n = Val(DesVOrdenPol.Value)
Retardo.Caption = Val(DesVRetardo.Value)
```

```
If (ModoTrabajo = 111 Or ModoTrabajo = 101 Or ModoTrabajo = 211 Or ModoTrabajo =
201) Then
```

```
    PolinomioD.Enabled = False
```

```
End If
```

```
End Sub
```

```
Sub Form_Paint ()
    ingredat.Ingresocoef.SetFocus
End Sub
```

```
Sub Ingresocoef_Change ()
```

```
    If Ingresocoef.Text = "" Then
```

```
        Coef(m) = 0
        Beep
        Ingresocoef.Text = "0"
    Else
```

```
        Coef(m) = Val(Ingresocoef.Text)
```

```
    End If
```

```
If PolinomioA = True Then
```

```
Ingresocoef.SetFocus
m = ordenpol.Caption
Numerocoef.Caption = m
Siguiete.Enabled = False
Anterior.Enabled = True
Anterior.Default = True
Ingresocoef.Text = Format$(Coef_C(m))
```

End Sub

```
Sub PolinomioD_Click ()
```

```
Ingresocoef.SetFocus
m = ordenpol.Caption
Numerocoef.Caption = m
Siguiete.Enabled = False
Anterior.Enabled = True
Anterior.Default = True
Ingresocoef.Text = Format$(Coef_D(m))
```

End Sub

```
Sub Siguiete_Click ()
```

```
Anterior.Enabled = True
```

```
If m < n Then
```

```
    m = m + 1
```

```
    If m = n Then
```

```
        Siguiete.Enabled = False
```

```
        Anterior.Default = True
```

```
    End If
```

```
End If
```

```
Numerocoef.Caption = m
```

```
Ingresocoef.SetFocus
```

```
If PolinomioA = True Then
Ingresocoef.Text = Format$(Coef_A(m))
Elseif polinomioB = True Then
Ingresocoef.Text = Format$(Coef_B(m))
Elseif polinomioC = True Then
Ingresocoef.Text = Format$(Coef_C(m))
Elseif PolinomioD = True Then
Ingresocoef.Text = Format$(Coef_D(m))
End If
```

End Sub

```
Sub VarianzaRuido_Change ()
```

```
If VarianzaRuido.Text = "" Then
```

```
    VarRuido = 1
```

```
    Beep
```

```
    VarianzaRuido.Text = "1"
```

```

Else
    VarRuido = Val(VarianzaRuido.Text)
End If
End Sub

Sub VarianzaRuido_KeyPress (KeyAscii As Integer)
    c$ = Chr$(KeyAscii)
    If c$ < "0" Or c$ > "9" Then

        If Asc(c$) = 8 Then
            KeyAscii = 8
        ElseIf c$ <> "8" Then
            KeyAscii = 0
        Beep
    End If
End If
End Sub

```

B.9 SUBROUTINAS DE LA PANTALLA DE ARCHIVAR DATOS

```

Sub Aceptar_Click ()
    On Error GoTo Error_en_guardado

    Open NomArchivo.Text For Output As #1
    Write #1, ModoTrabajo, n, d, r, VarRuido, a0, a1, a2, a3, a4, b0, b1, b2, b3, b4, c0, c1,
    c2, c3, c4, d0, d1, d2, d3, d4
    Close #1
    Guardardat.Hide

Salir:
    Exit Sub

Error_en_guardado:
    If (Err = 64) Then
        Beep
        MsgBox "Nombre del archivo es erróneo", 48, "Control de Mínima Varianza"
    ElseIf (Err = 70) Then
        Beep
    End If
End Sub

```

```
MsgBox "Unidad de disco con protección en contra de escritura", 16, "Control de Mínima Varianza"
```

```
End If
```

```
Resume Salir
```

```
End Sub
```

```
Sub Cancelar_Click ()
```

```
Unload Guardardat
```

```
End Sub
```

```
Sub Dir1_Change ()
```

```
File1.Path = Dir1.Path
```

```
If Right(File1.Path, 1) <> "\" Then
```

```
NomArchivo.Text = File1.Path & "\" & File1.FileName
```

```
Else
```

```
NomArchivo.Text = File1.Path & File1.FileName
```

```
End If
```

```
End Sub
```

```
Sub Drive1_Change ()
```

```
On Error GoTo Error_en_unidad_de_disco
```

```
Dir1.Path = Drive1.Drive
```

```
Salir1:
```

```
Exit Sub
```

```
Error_en_unidad_de_disco:
```

```
If Err = 68 Then
```

```
Beep
```

```
MsgBox "Problemas con la unidad de disco", 16, "Control de Mínima Varianza"
```

```
Drive1.Drive = "c:"
```

```
Resume Salir1
```

```
End If
```

```
End Sub
```

```
Sub File1_Click ()
```

```
Const MB_DEFBUTTON2 = 256, MB_ICONEXCLAMATION = 48, MB_YESNO = 4, IDYES = 6, IDNO = 7
```

```
Dim Response, DgDef
```

```
If (FileName = ejemplo1 Or FileName = ejemplo2 Or FileName = ejemplo3 Or FileName = ejemplo4 Or FileName = ejemplo5 Or FileName = ejemplo6) Then
```

```

DgDef = MB_DEFBUTTON + MB_ICONEXCLAMATION + MB_YESNO

Response = MsgBox("Este archivo ya existe. Desea Reescribirlo?", DgDef, "Control de
Mínima Varianza")

If Response = IDYES Then

    If Right(File1.Path, 1) <> "\" Then
        NomArchivo.Text = File1.Path & "\" & File1.FileName
    Else
        NomArchivo.Text = File1.Path & File1.FileName
    End If

Else

    Exit Sub

End If

End If

End Sub

Sub File1_PathChange ()

    If Right(File1.Path, 1) <> "\" Then
        NomArchivo.Text = File1.Path & "\" & File1.FileName
    Else
        NomArchivo.Text = File1.Path & File1.FileName
    End If

End Sub

Sub Form_Load ()

File1.Path = "c:\conminva\ejemplos"
Dir1.Path = File1.Path
Drive1.Drive = Dir1.Path

    If Right(File1.Path, 1) <> "\" Then
        NomArchivo.Text = File1.Path & "\" & "*.cmv"
    Else
        NomArchivo.Text = File1.Path & "*.cmv"
    End If

End Sub

```

B.10 SUBRUTINAS DE LA PANTALLA CARGAR DATOS

```

Sub Aceptar_Click ()

    On Error GoTo Error_en_apertura

    Open NomArchivo.Caption For Input As #1
    Input #1, ModoTrabajo, n, d, r, VarRuido, a0, a1, a2, a3, a4, b0, b1, b2, b3, b4, c0, c1,
c2, c3, c4, d0, d1, d2, d3, d4
    Close #1
    Cargardat.Hide

```

```

If (ModoTrabajo = 111 Or ModoTrabajo = 101 Or ModoTrabajo = 211 Or ModoTrabajo =
201) Then

    d0 = a0
    d1 = a1
    d2 = a2
    d3 = a3
    d4 = a4

End If

If (ModoTrabajo = 111 Or ModoTrabajo = 101 Or ModoTrabajo = 110 Or ModoTrabajo =
100) Then

    Graficos.DespVSetPoint.Enabled = False
    Graficos.SetPoint.Enabled = False

End If

If (ModoTrabajo = 211 Or ModoTrabajo = 201 Or ModoTrabajo = 210 Or ModoTrabajo =
200) Then

    Graficos.DespVSetPoint.Enabled = True
    Graficos.SetPoint.Enabled = True

End If

Simprin.Simulacion.Enabled = True
Simprin.Datosnuevos.Enabled = True
Simprin.Resultados.Enabled = False

Salir:

    Exit Sub

Error_en_apertura:

If (Err = 76) Then

    Beep
    MsgBox "Ningún archivo ha sido seleccionado", 48, "Control de Mínima Varianza"
    Resume Salir
End If
Resume Salir

End Sub

Sub Cancelar_Click ()

    Unload Cargardat

End Sub

Sub Dir1_Change ()
    File1.Path = Dir1.Path

    If Right(File1.Path, 1) <> "\" Then
        NomArchivo.Caption = File1.Path & "\" & File1.FileName
    Else
        NomArchivo.Caption = File1.Path & File1.FileName
    End If

```

```

End Sub

Sub Drive1_Change ()

On Error GoTo Error_en_unidad_de_disco

    Dir1.Path = Drive1.Drive

Salir1:

    Exit Sub

Error_en_unidad_de_disco:

    If Err = 68 Then
        Beep
        MsgBox "Problemas con la unidad de disco seleccionada", 16, "Control de Mínima
Varianza"
        Drive1.Drive = "c:"
        Resume Salir1

    End If

End Sub

Sub File1_Click ()

    If Right(File1.Path, 1) <> "\" Then
        NomArchivo.Caption = File1.Path & "\" & File1.FileName
    Else
        NomArchivo.Caption = File1.Path & File1.FileName
    End If

End Sub

Sub Form_Load ()

File1.Path = "c:\conminva\ejemplos"
Dir1.Path = File1.Path
Drive1.Drive = Dir1.Path

    If Right(File1.Path, 1) <> "\" Then
        NomArchivo.Caption = File1.Path & "\" & File1.FileName
    Else
        NomArchivo.Caption = File1.Path & File1.FileName
    End If

End Sub

```

B.11 SUBRUTINA SIMULACIÓN

```

Sub Simulacion ()

'Inicialización de variables

```

```
f1 = 0
f2 = 0
g0 = 0
g1 = 0
g2 = 0
g3 = 0
```

'Chequeo si A = D

```
If (ModoTrabajo = 111 Or ModoTrabajo = 101 Or ModoTrabajo = 211 Or ModoTrabajo = 201) Then
```

```
    d0 = a0
    d1 = a1
    d2 = a2
    d3 = a3
    d4 = a4
    d5 = a5
```

```
End If
```

'Cálculo de los coeficientes de los polinomios F y G

```
If d = 0 Then
```

```
    f0 = 1
    g0 = c1 - d1
    g1 = c2 - d2
    g2 = c3 - d3
    g3 = c4 - d4
```

```
Elsif d = 1 Then
```

```
    f0 = 1
    f1 = c1 - d1
    g0 = c2 - d2 - f1 * d1
    g1 = c3 - d3 - f1 * d2
    g2 = c4 - d4 - f1 * d3
    g3 = -f1 * d4
```

```
Elsif d = 2 Then
```

```
    f0 = 1
    f1 = c1 - d1
    f2 = c2 - d2 - f1 * d1
    g0 = c3 - d3 - f1 * d2 - f2 * d1
    g1 = c4 - d4 - f1 * d3 - f2 * d2
    g2 = -f1 * d4 - f2 * d3
    g3 = -f2 * d4
```

```
End If
```

'Cálculo de la varianza de la salida

```
VarSalida = VarRuido * (1 + f1 ^ 2 + f2 ^ 2)
```

'Cálculo de los coeficientes auxiliares para graficación

```
TU5 = (BF5 + factorrb * c4)
TU6 = (BF6)
```

```
Elseif (ModoTrabajo = 110 Or ModoTrabajo = 100 Or ModoTrabajo > 199 Or ModoTrabajo < 220) Then
```

```
TU1 = (BDF1 + factorrb)
TU2 = (BDF2 + factorrb * AC1)
TU3 = (BDF3 + factorrb * AC2)
TU4 = (BDF4 + factorrb * AC3)
TU5 = (BDF5 + factorrb * AC4)
TU6 = (BDF6 + factorrb * AC5)
TU7 = (BDF7 + factorrb * AC6)
TU8 = (BDF8 + factorrb * AC7)
TU9 = (BDF9 + factorrb * AC8)
TU10 = (BDF10)
```

```
End If
```

```
GCD1 = (c1 - d1) * g0
GCD2 = (c2 - d2) * g0 + (c1 - d1) * g1
GCD3 = (c3 - d3) * g0 + (c2 - d2) * g1 + (c1 - d1) * g2
GCD4 = (c4 - d4) * g0 + (c3 - d3) * g1 + (c2 - d2) * g2 + (c1 - d1) * g3
GCD5 = (c4 - d4) * g1 + (c3 - d3) * g2 + (c2 - d2) * g3
GCD6 = (c4 - d4) * g2 + (c3 - d3) * g3
GCD7 = (c4 - d4) * g3
```

```
If (ModoTrabajo >= 200) Then
```

```
'Cálculo de la constante Kp
```

```
  If ((1 + a1 + a2 + a3 + a4) = 0) Then
```

```
    Kp = 1
```

```
  Else
```

```
    Kp = (b0 + b1 + b2 + b3 + b4) / (1 + a1 + a2 + a3 + a4)
```

```
  End If
```

```
End If
```

```
CalcVarianzas
```

```
Simprin.Resultados.Enabled = True
```

```
End Sub
```

B.12 SUBROUTINA PARA EL CÁLCULO DE LA VARIANZAS

```
Sub CalcVarianzas ()
```

```
  Randomize
```

```
  wt = 5
```

```
  wtm1 = 5
```

```
suma_cuad_e(0) = 0
suma_cuad_yc(0) = 0
suma_cuad_ys(0) = 0
suma_cuad_pr(0) = 0
```

```
suma_e(0) = 0
suma_yc(0) = 0
suma_ys(0) = 0
suma_pr(0) = 0
```

```
For l = 1 To 1000
```

```
e(l) = Sqr(VarRuido) * (Rnd - .5)
cuad_e(l) = e(l) * e(l)
et = e(l) / Sqr(VarRuido)
```

```
'REGULACION
```

```
'A = D
```

```
If (ModoTrabajo = 111 Or ModoTrabajo = 101) Then
```

```
    If (b0 < 0 Or b0 > 0) Then
```

```
        ut = (1 / b0) * (-(TU1 * utm1 + TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 +
TU6 * utm6) - (g0 * yctm1 + g1 * yctm2 + g2 * yctm3 + g3 * yctm4))
```

```
        If (d = 0) Then
```

```
            TerminoU = (b0 * ut + b1 * utm1 + b2 * utm2 + b3 * utm3 + b4 * utm4)
```

```
        Elseif (d = 1) Then
```

```
            TerminoU = (b0 * utm1 + b1 * utm2 + b2 * utm3 + b3 * utm4 + b4 * utm5)
```

```
        Elseif (d = 2) Then
```

```
            TerminoU = (b0 * utm2 + b1 * utm3 + b2 * utm4 + b3 * utm5 + b4 * utm6)
```

```
        End If
```

```
        yct = TerminoU + Sqr(VarRuido) * (et + c1 * etm1 + c2 * etm2 + c3 * etm3 + c4 *
etm4) - (a1 * yctm1 + a2 * yctm2 + a3 * yctm3 + a4 * yctm4)
```

```
        yc(l) = yct
```

```
        cuad_yc(l) = yc(l) * yc(l)
```

```
        vt = Sqr(VarRuido) * (et + c1 * etm1 + c2 * etm2 + c3 * etm3 + c4 * etm4)
```

```
        pr(l) = vt
```

```
        cuad_pr(l) = pr(l) * pr(l)
```

```
    Elseif (b0 = 0) Then
```

```
        utm1 = (1 / TU1) * (-(TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 + TU6 *
utm6) - (g0 * yctm1 + g1 * yctm2 + g2 * yctm3 + g3 * yctm4))
```

If (d = 0) Then

TerminoU = (b1 * utm1 + b2 * utm2 + b3 * utm3 + b4 * utm4)

Elsel (d = 1) Then

TerminoU = (b1 * utm2 + b2 * utm3 + b3 * utm4 + b4 * utm5)

Elsel (d = 2) Then

TerminoU = (b1 * utm3 + b2 * utm4 + b3 * utm5 + b4 * utm6)

End If

yct = TerminoU + Sqr(VarRuido) * (et + c1 * etm1 + c2 * etm2 + c3 * etm3 + c4 * etm4) - (a1 * yctm1 + a2 * yctm2 + a3 * yctm3 + a4 * yctm4)

yc(l) = yct

cuad_yc(l) = yc(l) * yc(l)

vt = Sqr(VarRuido) * (et + c1 * etm1 + c2 * etm2 + c3 * etm3 + c4 * etm4)

pr(l) = vt

cuad_pr(l) = pr(l) * pr(l)

End If

'A <> D

Elsel (ModoTrabajo = 110 Or ModoTrabajo = 100) Then

If (b0 < 0 Or b0 > 0) Then

ut = (1 / b0) * (-(TU1 * utm1 + TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 + TU6 * utm6 + TU7 * utm7 + TU8 * utm8 + TU9 * utm9 + TU10 * utm10) - (g0 * yctm1 + AG1 * yctm2 + AG2 * yctm3 + AG3 * yctm4 + AG4 * yctm5 + AG5 * yctm6 + AG6 * yctm7 + AG7 * yctm8))

If (d = 0) Then

TerminoU = (b0 * ut + BD1 * utm1 + BD2 * utm2 + BD3 * utm3 + BD4 * utm4 + BD5 * utm5 + BD6 * utm6 + BD7 * utm7 + BD8 * utm8)

Elsel (d = 1) Then

TerminoU = (b0 * utm1 + BD1 * utm2 + BD2 * utm3 + BD3 * utm4 + BD4 * utm5 + BD5 * utm6 + BD6 * utm7 + BD7 * utm8 + BD8 * utm9)

Elsel (d = 2) Then

TerminoU = (b0 * utm2 + BD1 * utm3 + BD2 * utm4 + BD3 * utm5 + BD4 * utm6 + BD5 * utm7 + BD6 * utm8 + BD7 * utm9 + BD8 * utm10)

End If

```

    yct = TerminoU + Sqr(VarRuido) * (et + AC1 * etm1 + AC2 * etm2 + AC3 * etm3 +
    AC4 * etm4 + AC5 * etm5 + AC6 * etm6 + AC7 * etm7 + AC8 * etm8) - (AD1 * yctm1 + AD2
    * yctm2 + AD3 * yctm3 + AD4 * yctm4 + AD5 * yctm5 + AD6 * yctm6 + AD7 * yctm7 + AD8
    * yctm8)

```

```

    yc(l) = yct
    cuad_yc(l) = yc(l) * yc(l)

```

```

    vt = Sqr(VarRuido) * (et + AC1 * etm1 + AC2 * etm2 + AC3 * etm3 + AC4 * etm4 +
    AC5 * etm5 + AC6 * etm6 + AC7 * etm7 + AC8 * etm8)

```

```

    pr(l) = vt
    cuad_pr(l) = pr(l) * pr(l)

```

```

Elseif (b0 = 0) Then

```

```

    utm1 = (1 / TU1) * (-(TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 + TU6 *
    utm6 + TU7 * utm7 + TU8 * utm8 + TU9 * utm9 + TU10 * utm10) - (g0 * yctm1 + AG1 *
    yctm2 + AG2 * yctm3 + AG3 * yctm4 + AG4 * yctm5 + AG5 * yctm6 + AG6 * yctm7 + AG7 *
    yctm8))

```

```

If (d = 0) Then

```

```

    TerminoU = (BD1 * utm1 + BD2 * utm2 + BD3 * utm3 + BD4 * utm4 + BD5 * utm5 +
    BD6 * utm6 + BD7 * utm7 + BD8 * utm8)

```

```

Elseif (d = 1) Then

```

```

    TerminoU = (BD1 * utm2 + BD2 * utm3 + BD3 * utm4 + BD4 * utm5 + BD5 * utm6 +
    BD6 * utm7 + BD7 * utm8 + BD8 * utm9)

```

```

Elseif (d = 2) Then

```

```

    TerminoU = (BD1 * utm3 + BD2 * utm4 + BD3 * utm5 + BD4 * utm6 + BD5 * utm7 +
    BD6 * utm8 + BD7 * utm9 + BD8 * utm10)

```

```

End If

```

```

    yct = TerminoU + Sqr(VarRuido) * (et + AC1 * etm1 + AC2 * etm2 + AC3 * etm3 +
    AC4 * etm4 + AC5 * etm5 + AC6 * etm6 + AC7 * etm7 + AC8 * etm8) - (AD1 * yctm1 + AD2
    * yctm2 + AD3 * yctm3 + AD4 * yctm4 + AD5 * yctm5 + AD6 * yctm6 + AD7 * yctm7 + AD8
    * yctm8)

```

```

    yc(l) = yct
    cuad_yc(l) = yc(l) * yc(l)

```

```

    vt = Sqr(VarRuido) * (et + AC1 * etm1 + AC2 * etm2 + AC3 * etm3 + AC4 * etm4 +
    AC5 * etm5 + AC6 * etm6 + AC7 * etm7 + AC8 * etm8)

```

```

    pr(l) = vt
    cuad_pr(l) = pr(l) * pr(l)

```

```

End If

```

```

'SEGUIMIENTO

```

```

'A = D

```

```

Elseif (ModoTrabajo = 211 Or ModoTrabajo = 201) Then

```

```

    If (b0 < 0 Or b0 > 0) Then

```

$$ut = (1 / b0) * (-(TU1 * utm1 + TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 + TU6 * utm6 + TU7 * utm7 + TU8 * utm8 + TU9 * utm9 + TU10 * utm10) - (g0 * yctm1 + AG1 * yctm2 + AG2 * yctm3 + AG3 * yctm4 + AG4 * yctm5 + AG5 * yctm6 + AG7 * yctm8) + (1 + (FACTORRB / KP)) * (wtm1 + AC1 * wtm2 + AC2 * wtm3 + AC3 * wtm4 + AC4 * wtm5 + AC5 * wtm6 + AC6 * wtm7 + AC7 * wtm8 + AC8 * wtm9))$$

If (d = 0) Then

TerminoU = (b0 * ut + b1 * utm1 + b2 * utm2 + b3 * utm3 + b4 * utm4)

Elsif (d = 1) Then

TerminoU = (b0 * utm1 + b1 * utm2 + b2 * utm3 + b3 * utm4 + b4 * utm5)

Elsif (d = 2) Then

TerminoU = (b0 * utm2 + b1 * utm3 + b2 * utm4 + b3 * utm5 + b4 * utm6)

End If

$$yct = \text{TerminoU} + \text{Sqr}(\text{VarRuido}) * (et + c1 * etm1 + c2 * etm2 + c3 * etm3 + c4 * etm4) - (a1 * yctm1 + a2 * yctm2 + a3 * yctm3 + a4 * yctm4)$$

yc(l) = yct

cuad_yc(l) = yc(l) * yc(l)

$$vt = \text{Sqr}(\text{VarRuido}) * (et + c1 * etm1 + c2 * etm2 + c3 * etm3 + c4 * etm4)$$

pr(l) = vt

cuad_pr(l) = pr(l) * pr(l)

Elsif (b0 = 0) Then

$$utm1 = (1 / TU1) * (-(TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 + TU6 * utm6 + TU7 * utm7 + TU8 * utm8 + TU9 * utm9 + TU10 * utm10) - (g0 * yctm1 + AG1 * yctm2 + AG2 * yctm3 + AG3 * yctm4 + AG4 * yctm5 + AG5 * yctm6 + AG7 * yctm8) + (1 + (FACTORRB / KP)) * (wtm1 + AC1 * wtm2 + AC2 * wtm3 + AC3 * wtm4 + AC4 * wtm5 + AC5 * wtm6 + AC6 * wtm7 + AC7 * wtm8 + AC8 * wtm9))$$

If (d = 0) Then

TerminoU = (b1 * utm1 + b2 * utm2 + b3 * utm3 + b4 * utm4)

Elsif (d = 1) Then

TerminoU = (b1 * utm2 + b2 * utm3 + b3 * utm4 + b4 * utm5)

Elsif (d = 2) Then

TerminoU = (b1 * utm3 + b2 * utm4 + b3 * utm5 + b4 * utm6)

End If

$$yct = \text{TerminoU} + \text{Sqr}(\text{VarRuido}) * (et + c1 * etm1 + c2 * etm2 + c3 * etm3 + c4 * etm4) - (a1 * yctm1 + a2 * yctm2 + a3 * yctm3 + a4 * yctm4)$$

yc(l) = yct

cuad_yc(l) = yc(l) * yc(l)

$$vt = \text{Sqr}(\text{VarRuido}) * (et + c1 * etm1 + c2 * etm2 + c3 * etm3 + c4 * etm4)$$

pr(l) = vt

cuad_pr(l) = pr(l) * pr(l)

End If

'A <> D

Elseif (ModoTrabajo = 210 Or ModoTrabajo = 200) Then

If (b0 < 0 Or b0 > 0) Then

ut = (1 / b0) * (-(TU1.* utm1 + TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 + TU6 * utm6 + TU7 * utm7 + TU8 * utm8 + TU9 * utm9 + TU10 * utm10) - (g0 * yctm1 + AG1 * yctm2 + AG2 * yctm3 + AG3 * yctm4 + AG4 * yctm5 + AG5 * yctm6 + AG7 * yctm8) + (1 + (FACTORRB / KP)) * (wtm1 + AC1 * wtm2 + AC2 * wtm3 + AC3 * wtm4 + AC4 * wtm5 + AC5 * wtm6 + AC6 * wtm7 + AC7 * wtm8 + AC8 * wtm9))

If (d = 0) Then

TerminoU = (b0 * ut + BD1 * utm1 + BD2 * utm2 + BD3 * utm3 + BD4 * utm4 + BD5 * utm5 + BD6 * utm6 + BD7 * utm7 + BD8 * utm8)

Elseif (d = 1) Then

TerminoU = (b0 * utm1 + BD1 * utm2 + BD2 * utm3 + BD3 * utm4 + BD4 * utm5 + BD5 * utm6 + BD6 * utm7 + BD7 * utm8 + BD8 * utm9)

Elseif (d = 2) Then

TerminoU = (b0 * utm2 + BD1 * utm3 + BD2 * utm4 + BD3 * utm5 + BD4 * utm6 + BD5 * utm7 + BD6 * utm8 + BD7 * utm9 + BD8 * utm10)

End If

yct = TerminoU + Sqr(VarRuido) * (et + AC1 * etm1 + AC2 * etm2 + AC3 * etm3 + AC4 * etm4 + AC5 * etm5 + AC6 * etm6 + AC7 * etm7 + AC8 * etm8) - (AD1 * yctm1 + AD2 * yctm2 + AD3 * yctm3 + AD4 * yctm4 + AD5 * yctm5 + AD6 * yctm6 + AD7 * yctm7 + AD8 * yctm8)

yc(l) = yct

cuad_yc(l) = yc(l) * yc(l)

vt = Sqr(VarRuido) * (et + AC1 * etm1 + AC2 * etm2 + AC3 * etm3 + AC4 * etm4 + AC5 * etm5 + AC6 * etm6 + AC7 * etm7 + AC8 * etm8)

pr(l) = vt

cuad_pr(l) = pr(l) * pr(l)

Elseif (b0 = 0) Then

utm1 = (1 / TU1) * (-(TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 + TU6 * utm6 + TU7 * utm7 + TU8 * utm8 + TU9 * utm9 + TU10 * utm10) - (g0 * yctm1 + AG1 * yctm2 + AG2 * yctm3 + AG3 * yctm4 + AG4 * yctm5 + AG5 * yctm6 + AG7 * yctm8) + (1 + (FACTORRB / KP)) * (wtm1 + AC1 * wtm2 + AC2 * wtm3 + AC3 * wtm4 + AC4 * wtm5 + AC5 * wtm6 + AC6 * wtm7 + AC7 * wtm8 + AC8 * wtm9))

If (d = 0) Then

TerminoU = (BD1 * utm1 + BD2 * utm2 + BD3 * utm3 + BD4 * utm4 + BD5 * utm5 + BD6 * utm6 + BD7 * utm7 + BD8 * utm8)

```

Elseif (d = 1) Then

    TerminoU = (BD1 * utm2 + BD2 * utm3 + BD3 * utm4 + BD4 * utm5 + BD5 * utm6 +
    BD6 * utm7 + BD7 * utm8 + BD8 * utm9)

    Elseif (d = 2) Then

        TerminoU = (BD1 * utm3 + BD2 * utm4 + BD3 * utm5 + BD4 * utm6 + BD5 * utm7 +
        BD6 * utm8 + BD7 * utm9 + BD8 * utm10)

    End If

    yct = TerminoU + Sqr(VarRuido) * (et + AC1 * etm1 + AC2 * etm2 + AC3 * etm3 +
    AC4 * etm4 + AC5 * etm5 + AC6 * etm6 + AC7 * etm7 + AC8 * etm8) - (AD1 * yctm1 + AD2
    * yctm2 + AD3 * yctm3 + AD4 * yctm4 + AD5 * yctm5 + AD6 * yctm6 + AD7 * yctm7 + AD8
    * yctm8)
    yc(l) = yct
    cuad_yc(l) = yc(l) * yc(l)

    vt = Sqr(VarRuido) * (et + AC1 * etm1 + AC2 * etm2 + AC3 * etm3 + AC4 * etm4 +
    AC5 * etm5 + AC6 * etm6 + AC7 * etm7 + AC8 * etm8)
    pr(l) = vt
    cuad_pr(l) = pr(l) * pr(l)

End If

End If
*****
'REGULACION

If (ModoTrabajo > 100 And ModoTrabajo < 112) Then

    yst = (Sqr(VarRuido) * (et + c1 * etm1 + c2 * etm2 + c3 * etm3 + c4 * etm4) - (d1 * ystm1
    + d2 * ystm2 + d3 * ystm3 + d4 * ystm4))
    ys(l) = yst
    cuad_ys(l) = ys(l) * ys(l)

'SEGUIMIENTO

'A = D

Elseif (ModoTrabajo = 211 Or ModoTrabajo = 201) Then

    ust = wtm1

    If (d = 0) Then

        TerminoUs = (b0 * ust + b1 * utm1 + b2 * ustm2 + b3 * ustm3 + b4 * ustm4)

    Elseif (d = 1) Then

        TerminoUs = (b0 * ustm1 + b1 * ustm2 + b2 * ustm3 + b3 * ustm4 + b4 * ustm5)

    Elseif (d = 2) Then

```

```

TerminoUs = (b0 * ustrn2 + b1 * ustm3 + b2 * ustm4 + b3 * ustm5 + b4 * ustm6)

End If

    yst = TerminoUs + Sqr(VarRuido) * (et + c1 * etm1 + c2 * etm2 + c3 * etm3 + c4 * etm4)
- (a1 * ystm1 + a2 * ystm2 + a3 * ystm3 + a4 * ystm4)
    ys(l) = yst
    cuad_ys(l) = ys(l) * ys(l)

'A <> D

Elseif (ModoTrabajo = 210 Or ModoTrabajo = 200) Then

    ust = wtm1

    If (d = 0) Then

        TerminoUs = (b0 * ust + BD1 * ustm1 + BD2 * ustm2 + BD3 * ustm3 + BD4 * ustm4 +
BD5 * ustm5 + BD6 * ustm6 + BD7 * ustm7 + BD8 * ustm8)

        Elseif (d = 1) Then

            TerminoUs = (b0 * ustm1 + BD1 * ustm2 + BD2 * ustm3 + BD3 * ustm4 + BD4 * ustm5
+ BD5 * ustm6 + BD6 * ustm7 + BD7 * ustm8 + BD8 * ustm9)

            Elseif (d = 2) Then

                TerminoUs = (b0 * ustm2 + BD1 * ustm3 + BD2 * ustm4 + BD3 * ustm5 + BD4 * ustm6
+ BD5 * ustm7 + BD6 * ustm8 + BD7 * ustm9 + BD8 * ustm10)

            End If

            yst = TerminoUs + Sqr(VarRuido) * (et + AC1 * etm1 + AC2 * etm2 + AC3 * etm3 + AC4
* etm4 + AC5 * etm5 + AC6 * etm6 + AC7 * etm7 + AC8 * etm8) - (AD1 * ystm1 + AD2 *
ystm2 + AD3 * ystm3 + AD4 * ystm4 + AD5 * ystm5 + AD6 * ystm6 + AD7 * ystm7 + AD8 *
ystm8)
            ys(l) = yst
            cuad_ys(l) = ys(l) * ys(l)

        End If

        *****

suma_cuad_e(l) = suma_cuad_e(l - 1) + cuad_e(l)
suma_cuad_yc(l) = suma_cuad_yc(l - 1) + cuad_yc(l)
suma_cuad_ys(l) = suma_cuad_ys(l - 1) + cuad_ys(l)
suma_cuad_pr(l) = suma_cuad_pr(l - 1) + cuad_pr(l)

suma_e(l) = suma_e(l - 1) + e(l)
suma_yc(l) = suma_yc(l - 1) + yc(l)
suma_ys(l) = suma_ys(l - 1) + ys(l)
suma_pr(l) = suma_pr(l - 1) + pr(l)

Corrido_de_Valores

Next l

Valesp_cuad_e = suma_cuad_e(1000) / 1000

```

```
Valesp_cuad_yc = suma_cuad_yc(1000) / 1000  
Valesp_cuad_ys = suma_cuad_ys(1000) / 1000  
Valesp_cuad_pr = suma_cuad_pr(1000) / 1000
```

```
Valesp_e = suma_e(1000) / 1000  
Valesp_yc = suma_yc(1000) / 1000  
Valesp_ys = suma_ys(1000) / 1000  
Valesp_pr = suma_pr(1000) / 1000
```

```
Var_e_exp = Valesp_cuad_e - (Valesp_e * Valesp_e)  
Var_yc_exp = Valesp_cuad_yc - (Valesp_yc * Valesp_yc)  
Var_ys_exp = Valesp_cuad_ys - (Valesp_ys * Valesp_ys)  
Var_pr_exp = Valesp_cuad_pr - (Valesp_pr * Valesp_pr)
```

```
End Sub
```

B.13 SUBROUTINAS DE LA PANTALLA DE REPORTE DE DATOS

```
Sub Form_Load ()
```

```
If n = 1 Then
```

```
label3.Visible = False  
label4.Visible = False  
label5.Visible = False  
ca2.Visible = False  
ca3.Visible = False  
ca4.Visible = False
```

```
label15.Visible = False  
label16.Visible = False  
label17.Visible = False  
cb2.Visible = False  
cb3.Visible = False  
cb4.Visible = False
```

```
label27.Visible = False  
label28.Visible = False  
label29.Visible = False  
cc2.Visible = False  
cc3.Visible = False  
cc4.Visible = False
```

```
label38.Visible = False  
label39.Visible = False  
label40.Visible = False  
cd2.Visible = False  
cd3.Visible = False  
cd4.Visible = False
```

```
label32.Visible = False  
label33.Visible = False  
cg2.Visible = False  
cg3.Visible = False
```

Elseif n = 2 Then

label4.Visible = False
label5.Visible = False
ca3.Visible = False
ca4.Visible = False

label16.Visible = False
label17.Visible = False
cb3.Visible = False
cb4.Visible = False

label28.Visible = False
label29.Visible = False
cc3.Visible = False
cc4.Visible = False

label39.Visible = False
label40.Visible = False
cd3.Visible = False
cd4.Visible = False

label32.Visible = False
label33.Visible = False
cg2.Visible = False
cg3.Visible = False

Elseif n = 3 Then

label5.Visible = False
ca4.Visible = False

label17.Visible = False
cb4.Visible = False

label29.Visible = False
cc4.Visible = False

label40.Visible = False
cd4.Visible = False

label33.Visible = False
cg3.Visible = False

Elseif n = 4 Then

End If

If d = 0 Then

label22.Visible = False
cf1.Visible = False

label23.Visible = False
cf2.Visible = False

Elseif d = 1 Then

label23.Visible = False
cf2.Visible = False

End If

'Asignacion de valores a las etiquetas

ca0.Caption = Format\$(a0, "0.000")
ca1.Caption = Format\$(a1, "0.000")
ca2.Caption = Format\$(a2, "0.000")
ca3.Caption = Format\$(a3, "0.000")
ca4.Caption = Format\$(a4, "0.000")

cb0.Caption = Format\$(b0, "0.000")
cb1.Caption = Format\$(b1, "0.000")
cb2.Caption = Format\$(b2, "0.000")
cb3.Caption = Format\$(b3, "0.000")
cb4.Caption = Format\$(b4, "0.000")

cc0.Caption = Format\$(c0, "0.000")
cc1.Caption = Format\$(c1, "0.000")
cc2.Caption = Format\$(c2, "0.000")
cc3.Caption = Format\$(c3, "0.000")
cc4.Caption = Format\$(c4, "0.000")

cd0.Caption = Format\$(d0, "0.000")
cd1.Caption = Format\$(d1, "0.000")
cd2.Caption = Format\$(d2, "0.000")
cd3.Caption = Format\$(d3, "0.000")
cd4.Caption = Format\$(d4, "0.000")

cf1.Caption = Format\$(f1, "0.000")
cf2.Caption = Format\$(f2, "0.000")

cg0.Caption = Format\$(g0, "0.000")
cg1.Caption = Format\$(g1, "0.000")
cg2.Caption = Format\$(g2, "0.000")
cg3.Caption = Format\$(g3, "0.000")

OrdenPol.Caption = Format\$(n)
FactorPonderacion.Caption = Format\$(r, "0.000")
Retardo.Caption = Format\$(d)
VarianzaRuido.Caption = Format\$(VarRuido, "0.000")
Varianza.Caption = Format\$(VarSalida, "0.000")
Var_e.Caption = Format\$(12 * var_e_exp, "0.000")
Var_pr.Caption = Format\$(12 * var_pr_exp, "0.000")
Var_con.Caption = Format\$(12 * var_yc_exp, "0.000")
Var_sin.Caption = Format\$(12 * var_ys_exp, "0.000")

End Sub

Sub Imprimir_Click ()

Impresiondatos

End Sub

```
Sub Salir_Click ()
```

```
    Unload reporte
```

```
End Sub
```

B.14 SUBROUTINA PARA IMPRIMIR EL REPORTE DE DATOS

```
Sub Impresiondatos ()
```

```
Coordey = 3
```

```
Delta = .5
```

```
'INICIALIZACION DE LA ESCALA DE LA HOJA
```

```
    Printer.ScaleWidth = 21
```

```
    Printer.ScaleHeight = 29.7
```

```
    Printer.ScaleTop = 0
```

```
    Printer.ScaleLeft = 0
```

```
'TITULO DEL REPORTE DEL DATOS
```

```
    Printer.CurrentX = 2.5
```

```
    Printer.CurrentY = 1
```

```
    Printer.FontSize = 16
```

```
    Printer.Print "CONTROL DE MINIMA VARIANZA - REPORTE DE DATOS"
```

```
'DATOS DEL SISTEMA
```

```
    Printer.FontSize = 12
```

```
    Printer.FontItalic = True
```

```
    Printer.CurrentX = 3.5
```

```
    Printer.CurrentY = Coordey - Delta
```

```
    Printer.Print "DATOS DEL SISTEMA"
```

```
    Printer.CurrentX = 4
```

```
    Printer.CurrentY = Coordey + Delta
```

```
    Printer.Print "Orden de los polinomios = " & n
```

```
    Printer.CurrentX = 4
```

```
    Printer.CurrentY = Coordey + 2 * Delta
```

```
    Printer.Print "Factor de ponderación = " & Format$(r, "0.00000")
```

```
    Printer.CurrentX = 4
```

```
    Printer.CurrentY = Coordey + 3 * Delta
```

```
    Printer.Print "Retardo d = " & d
```

```
    Printer.CurrentX = 4
```

```
    Printer.CurrentY = Coordey + 4 * Delta
```

```
    Printer.Print "Varianza del ruido = " & Format$(VarRuido, "0.00000")
```

```
    Printer.CurrentX = 4
```

```
    Printer.CurrentY = Coordey + 6 * Delta
```

```
    Printer.Print "Coeficientes de los polinomios : "
```

```
Printer.CurrentX = 4  
Printer.CurrentY = Coordy + 8 * Delta  
Printer.Print "a0 = " & Format$(a0, "0.00000")
```

```
Printer.CurrentX = 4  
Printer.CurrentY = Coordy + 9 * Delta  
Printer.Print "a1 = " & Format$(a1, "0.00000")
```

```
Printer.CurrentX = 10  
Printer.CurrentY = Coordy + 8 * Delta  
Printer.Print "b0 = " & Format$(b0, "0.00000")
```

```
Printer.CurrentX = 10  
Printer.CurrentY = Coordy + 9 * Delta  
Printer.Print "b1 = " & Format$(b1, "0.00000")
```

```
If n > 1 Then
```

```
Printer.CurrentX = 4  
Printer.CurrentY = Coordy + 10 * Delta  
Printer.Print "a2 = " & Format$(a2, "0.00000")
```

```
Printer.CurrentX = 10  
Printer.CurrentY = Coordy + 10 * Delta  
Printer.Print "b2 = " & Format$(b2, "0.00000")
```

```
End If
```

```
'$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
```

```
If n = 2 Then
```

```
Printer.CurrentX = 4  
Printer.CurrentY = Coordy + 13 * Delta  
Printer.Print "c0 = " & Format$(c0, "0.00000")
```

```
Printer.CurrentX = 4  
Printer.CurrentY = Coordy + 14 * Delta  
Printer.Print "c1 = " & Format$(c1, "0.00000")
```

```
Printer.CurrentX = 4  
Printer.CurrentY = Coordy + 15 * Delta  
Printer.Print "c2 = " & Format$(c2, "0.00000")
```

```
Printer.CurrentX = 10  
Printer.CurrentY = Coordy + 13 * Delta  
Printer.Print "d0 = " & Format$(d0, "0.00000")
```

```
Printer.CurrentX = 10  
Printer.CurrentY = Coordy + 14 * Delta  
Printer.Print "d1 = " & Format$(d1, "0.00000")
```

```
Printer.CurrentX = 10  
Printer.CurrentY = Coordy + 15 * Delta  
Printer.Print "d2 = " & Format$(d2, "0.00000")
```

```
Printer.CurrentX = 3.5  
Printer.CurrentY = Coordy + 17 * Delta  
Printer.Print "RESULTADOS"
```

```

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 19 * Delta
Printer.Print "Coeficientes de los polinomios F y G :."

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 21 * Delta
Printer.Print "f0 = 1.00000"

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 22 * Delta
Printer.Print "f1 = " & Format$(f1, "0.00000")

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 23 * Delta
Printer.Print "f2 = " & Format$(f2, "0.00000")

Printer.CurrentX = 10
Printer.CurrentY = Coordey + 21 * Delta
Printer.Print "g0 = " & Format$(g0, "0.00000")

Printer.CurrentX = 10
Printer.CurrentY = Coordey + 22 * Delta
Printer.Print "g1 = " & Format$(g1, "0.00000")

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 25 * Delta
Printer.Print "Varianza de la salida controlada (teórica) = " & Format$(Varsalida,
"0.00000")

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 26 * Delta
Printer.Print "Varianza de la salida controlada (experimental) = " & Format$(12 *
Var_yc_exp, "0.00000")

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 27 * Delta
Printer.Print "Varianza de la salida sin control (experimental) = " & Format$(12 *
Var_ys_exp, "0.00000")

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 28 * Delta
Printer.Print "Varianza del ruido blanco (experimental) = " & Format$(12 * Var_e_exp,
"0.00000")

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 29 * Delta
Printer.Print "Varianza del proceso de ruido (experimental) = " & Format$(12 *
Var_pr_exp, "0.00000")

'%%%%%%%%%%%%%%
%%%%%%%%%%%%%%

Elseif n = 3 Then

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 11 * Delta
Printer.Print "a3 = " & Format$(a3, "0.00000")

```

```
Printer.CurrentX = 10
Printer.CurrentY = Coordey + 11 * Delta
Printer.Print "b3 = " & Format$(b3, "0.00000")
```

```
Printer.CurrentX = 4
Printer.CurrentY = Coordey + 14 * Delta
Printer.Print "c0 = " & Format$(c0, "0.00000")
```

```
Printer.CurrentX = 4
Printer.CurrentY = Coordey + 15 * Delta
Printer.Print "c1 = " & Format$(c1, "0.00000")
```

```
Printer.CurrentX = 4
Printer.CurrentY = Coordey + 16 * Delta
Printer.Print "c2 = " & Format$(c2, "0.00000")
```

```
Printer.CurrentX = 4
Printer.CurrentY = Coordey + 17 * Delta
Printer.Print "c3 = " & Format$(c3, "0.00000")
```

```
Printer.CurrentX = 10
Printer.CurrentY = Coordey + 14 * Delta
Printer.Print "d0 = " & Format$(d0, "0.00000")
```

```
Printer.CurrentX = 10
Printer.CurrentY = Coordey + 15 * Delta
Printer.Print "d1 = " & Format$(d1, "0.00000")
```

```
Printer.CurrentX = 10
Printer.CurrentY = Coordey + 16 * Delta
Printer.Print "d2 = " & Format$(d2, "0.00000")
```

```
Printer.CurrentX = 10
Printer.CurrentY = Coordey + 17 * Delta
Printer.Print "d3 = " & Format$(d3, "0.00000")
```

```
Printer.CurrentX = 3.5
Printer.CurrentY = Coordey + 19 * Delta
Printer.Print "RESULTADOS"
```

```
Printer.CurrentX = 4
Printer.CurrentY = Coordey + 21 * Delta
Printer.Print "Coeficientes de los polinomios F y G :"
```

```
Printer.CurrentX = 4
Printer.CurrentY = Coordey + 23 * Delta
Printer.Print "f0 = 1.00000"
```

```
Printer.CurrentX = 4
Printer.CurrentY = Coordey + 24 * Delta
Printer.Print "f1 = " & Format$(f1, "0.00000")
```

```
Printer.CurrentX = 4
Printer.CurrentY = Coordey + 25 * Delta
```

```

Printer.Print "f2 = " & Format$(f2, "0.00000")

Printer.CurrentX = 10
Printer.CurrentY = Coordey + 23 * Delta
Printer.Print "g0 = " & Format$(g0, "0.00000")

Printer.CurrentX = 10
Printer.CurrentY = Coordey + 24 * Delta
Printer.Print "g1 = " & Format$(g1, "0.00000")

Printer.CurrentX = 10
Printer.CurrentY = Coordey + 25 * Delta
Printer.Print "g2 = " & Format$(g2, "0.00000")

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 28 * Delta
Printer.Print "Varianza de la salida controlada (teórica) = " & Format$(Varsalida,
"0.00000")

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 29 * Delta
Printer.Print "Varianza de la salida controlada (experimental) = " & Format$(12 *
Var_yc_exp, "0.00000")

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 30 * Delta
Printer.Print "Varianza de la salida sin control (experimental) = " & Format$(12 *
Var_ys_exp, "0.00000")

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 31 * Delta
Printer.Print "Varianza del ruido blanco (experimental) = " & Format$(12 * Var_e_exp,
"0.00000")

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 32 * Delta
Printer.Print "Varianza del proceso de ruido (experimental) = " & Format$(12 *
Var_pr_exp, "0.00000")

'++++++
+++++

Elseif n = 4 Then

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 11 * Delta
Printer.Print "a3 = " & Format$(a3, "0.00000")

Printer.CurrentX = 10
Printer.CurrentY = Coordey + 11 * Delta
Printer.Print "b3 = " & Format$(b3, "0.00000")

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 12 * Delta
Printer.Print "a4 = " & Format$(a4, "0.00000")

Printer.CurrentX = 10
Printer.CurrentY = Coordey + 12 * Delta
Printer.Print "b4 = " & Format$(b4, "0.00000")

```

```
Printer.CurrentX = 4  
Printer.CurrentY = Coordey + 15 * Delta  
Printer.Print "c0 = " & Format$(c0, "0.00000")
```

```
Printer.CurrentX = 4  
Printer.CurrentY = Coordey + 16 * Delta  
Printer.Print "c1 = " & Format$(c1, "0.00000")
```

```
Printer.CurrentX = 4  
Printer.CurrentY = Coordey + 17 * Delta  
Printer.Print "c2 = " & Format$(c2, "0.00000")
```

```
Printer.CurrentX = 4  
Printer.CurrentY = Coordey + 18 * Delta  
Printer.Print "c3 = " & Format$(c3, "0.00000")
```

```
Printer.CurrentX = 4  
Printer.CurrentY = Coordey + 19 * Delta  
Printer.Print "c4 = " & Format$(c4, "0.00000")
```

```
Printer.CurrentX = 10  
Printer.CurrentY = Coordey + 15 * Delta  
Printer.Print "d0 = " & Format$(d0, "0.00000")
```

```
Printer.CurrentX = 10  
Printer.CurrentY = Coordey + 16 * Delta  
Printer.Print "d1 = " & Format$(d1, "0.00000")
```

```
Printer.CurrentX = 10  
Printer.CurrentY = Coordey + 17 * Delta  
Printer.Print "d2 = " & Format$(d2, "0.00000")
```

```
Printer.CurrentX = 10  
Printer.CurrentY = Coordey + 18 * Delta  
Printer.Print "d3 = " & Format$(d3, "0.00000")
```

```
Printer.CurrentX = 10  
Printer.CurrentY = Coordey + 19 * Delta  
Printer.Print "d4 = " & Format$(d4, "0.00000")
```

```
Printer.CurrentX = 3.5  
Printer.CurrentY = Coordey + 21 * Delta  
Printer.Print "RESULTADOS"
```

```
Printer.CurrentX = 4  
Printer.CurrentY = Coordey + 23 * Delta  
Printer.Print "Coeficientes de los polinomios F y G :"
```

```
Printer.CurrentX = 4  
Printer.CurrentY = Coordey + 25 * Delta  
Printer.Print "f0 = 1.00000"
```

```
Printer.CurrentX = 4  
Printer.CurrentY = Coordey + 26 * Delta  
Printer.Print "f1 = " & Format$(f1, "0.00000")
```

```

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 27 * Delta
Printer.Print "f2 = " & Format$(f2, "0.00000")

Printer.CurrentX = 10
Printer.CurrentY = Coordey + 25 * Delta
Printer.Print "g0 = " & Format$(g0, "0.00000")

Printer.CurrentX = 10.
Printer.CurrentY = Coordey + 26 * Delta
Printer.Print "g1 = " & Format$(g1, "0.00000")

Printer.CurrentX = 10
Printer.CurrentY = Coordey + 27 * Delta
Printer.Print "g2 = " & Format$(g2, "0.00000")

Printer.CurrentX = 10
Printer.CurrentY = Coordey + 28 * Delta
Printer.Print "g3 = " & Format$(g3, "0.00000")

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 31 * Delta
Printer.Print "Varianza de la salida controlada (teórica) = " & Format$(Varsalida,
"0.00000")

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 32 * Delta
Printer.Print "Varianza de la salida controlada (experimental) = " & Format$(12 *
Var_yc_exp, "0.00000")

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 33 * Delta
Printer.Print "Varianza de la salida sin control (experimental) = " & Format$(12 *
Var_ys_exp, "0.00000")

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 34 * Delta
Printer.Print "Varianza del ruido blanco (experimental) = " & Format$(12 * Var_e_exp,
"0.00000")

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 35 * Delta
Printer.Print "Varianza del proceso de ruido (experimental) = " & Format$(12 *
Var_pr_exp, "0.00000")

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Elseif n = 1 Then

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 12 * Delta
Printer.Print "c0 = " & Format$(c0, "0.00000")

Printer.CurrentX = 4
Printer.CurrentY = Coordey + 13 * Delta
Printer.Print "c1 = " & Format$(c1, "0.00000")

```

```
Printer.CurrentX = 10
Printer.CurrentY = Coordey + 12 * Delta
Printer.Print "d0 = " & Format$(d0, "0.00000")
```

```
Printer.CurrentX = 10
Printer.CurrentY = Coordey + 13 * Delta
Printer.Print "d1 = " & Format$(d1, "0.00000")
```

```
Printer.CurrentX = 3.5
Printer.CurrentY = Coordey + 15 * Delta
Printer.Print "RESULTADOS"
```

```
Printer.CurrentX = 4
Printer.CurrentY = Coordey + 17 * Delta
Printer.Print "Coeficientes de los polinomios F y G :"
```

```
Printer.CurrentX = 4
Printer.CurrentY = Coordey + 19 * Delta
Printer.Print "f0 = 1.00000"
```

```
Printer.CurrentX = 4
Printer.CurrentY = Coordey + 20 * Delta
Printer.Print "f1 = " & Format$(f1, "0.00000")
```

```
Printer.CurrentX = 4
Printer.CurrentY = Coordey + 21 * Delta
Printer.Print "f2 = " & Format$(f2, "0.00000")
```

```
Printer.CurrentX = 10
Printer.CurrentY = Coordey + 19 * Delta
Printer.Print "g0 = " & Format$(g0, "0.00000")
```

```
Printer.CurrentX = 4
Printer.CurrentY = Coordey + 23 * Delta
Printer.Print "Varianza de la salida controlada (teórica) = " & Format$(Varsalida,
"0.00000")
```

```
Printer.CurrentX = 4
Printer.CurrentY = Coordey + 24 * Delta
Printer.Print "Varianza de la salida controlada (experimental) = " & Format$(12 *
Var_yc_exp, "0.00000")
```

```
Printer.CurrentX = 4
Printer.CurrentY = Coordey + 25 * Delta
Printer.Print "Varianza de la salida sin control (experimental) = " & Format$(12 *
Var_ys_exp, "0.00000")
```

```
Printer.CurrentX = 4
Printer.CurrentY = Coordey + 26 * Delta
Printer.Print "Varianza del ruido blanco (experimental) = " & Format$(12 * Var_e_exp,
"0.00000")
```

```
Printer.CurrentX = 4
Printer.CurrentY = Coordey + 27 * Delta
```



```

opciongrafo2 = 4

End Sub

Sub Grafo2salccmv_Click ()

opciongrafo2 = 2

End Sub

Sub Grafo2salscmv_Click ()

opciongrafo2 = 1

End Sub

Sub Visualizar_Click ()

If opciongrafo1 = 1 Then

graficos.Titulografo1.Caption = "Salida sin C.M.V."

Elseif opciongrafo1 = 2 Then

graficos.Titulografo1.Caption = "Salida con C.M.V."

Elseif opciongrafo1 = 3 Then

graficos.Titulografo1.Caption = "Control"

Elseif opciongrafo1 = 4 Then

graficos.Titulografo1.Caption = "Proceso de Ruido"

End If

If opciongrafo2 = 1 Then

graficos.Titulografo2.Caption = "Salida sin C.M.V."

Elseif opciongrafo2 = 2 Then

graficos.Titulografo2.Caption = "Salida con C.M.V."

Elseif opciongrafo2 = 3 Then

graficos.Titulografo2.Caption = "Control"

Elseif opciongrafo2 = 4 Then

graficos.Titulografo2.Caption = "Ruido"

End If

graficos.Show 1

grafo1X1 = 0

```

```
grafo2x1 = 0
```

```
End Sub
```

B.16 SUBROUTINAS DE LA PANTALLA DE GRÁFICOS

```
Sub Cambescalas_Click ()
```

```
  CambioEscalas.Show 1
```

```
  parar.Enabled = False
```

```
  iniciar.Enabled = True
```

```
  encerrar
```

```
  iniciar.SetFocus
```

```
End Sub
```

```
Sub CambTiempoMuestreo_Click ()
```

```
  TiempoMuestreo.Text1.Text = graficos.Muestreo.Interval
```

```
  TiempoMuestreo.Show 1
```

```
  parar.Enabled = False
```

```
  iniciar.Enabled = True
```

```
  encerrar
```

```
  iniciar.SetFocus
```

```
End Sub
```

```
Sub DespVSetPoint_Change ()
```

```
If TiempoReal = 1 Then
```

```
  refer = despvsetpoint.Value
```

```
  setpoint.Caption = (refer / 3) + 5
```

```
Else
```

```
  setpoint.Caption = despvsetpoint.Value
```

```
End If
```

```
wtm1 = despvsetpoint.Value
```

```
' setpoint.Caption = despvsetpoint.Value
```

```
' wtm1 = despvsetpoint.Value
```

```
End Sub
```

```
Sub Form_Load ()
```

```
If TiempoReal = 1 Then
```

```
  despvsetpoint.SmallChange = 3
```

```
  despvsetpoint.LargeChange = 3
```

```
  despvsetpoint.Min = 6
```

```
  Cambescalas.Enabled = False
```

```
  grafo1ymax = 10
```

```
  grafo1ymin = 0
```

```
End If
```

```
tiempo1 = 100
```

```
tiempo2 = 100
```

```
grafo1.Cls  
grafo2.Cls
```

```
parar.Enabled = False
```

```
despvsetpoint.Value = 0  
setpoint.Caption = despvsetpoint.Value  
wtrm1 = despvsetpoint.Value
```

```
If (ModoTrabajo > 99 And ModoTrabajo < 120) Then
```

```
despvsetpoint.Enabled = False
```

```
End If
```

```
End Sub
```

```
Sub Imprimir_Click ()
```

```
label1.Visible = True  
picture1.Visible = False  
setpoint.Visible = False  
despvsetpoint.Visible = False  
label2.Visible = False  
iniciar.Visible = False  
parar.Visible = False  
PrintForm  
iniciar.Visible = True  
parar.Visible = True  
setpoint.Visible = True  
despvsetpoint.Visible = True  
label2.Visible = True  
picture1.Visible = True  
label1.Visible = False  
parar.Enabled = False  
iniciar.Enabled = True  
encerar  
iniciar.SetFocus
```

```
End Sub
```

```
Sub Iniciar_Click ()
```

```
On Error GoTo Error_en_escalas
```

```
grafo1x1 = 0  
grafo2x1 = 0
```

```
If TiempoReal = 0 Then
```

```
If (CambioEscalas.Option1 = True) Then
```

```
If opciongrafo1 = 1 Then
```

```
grafo1ymin = -50  
grafo1ymax = 50
```

```

Elseif opciongrafo1 = 2 Then

    grafo1ymin = -10
    grafo1ymax = 10

Elseif opciongrafo1 = 3 Then

    grafo1ymin = -10
    grafo1ymax = 10

Elseif opciongrafo1 = 4 Then

    grafo1ymin = -5
    grafo1ymax = 5

End If

grafo1xmin = 0
grafo1xmax = 10

tiempo1 = grafo1xmax / .1

graficos.g1ymin.Caption = Format(grafo1ymin, "0.000")
graficos.g1ymax.Caption = Format(grafo1ymax, "0.000")
graficos.g1xmin.Caption = Format(grafo1xmin, "0")
graficos.g1xmax.Caption = Format(tiempo1, "0")

If opciongrafo2 = 1 Then

    grafo2ymin = -50
    grafo2ymax = 50

Elseif opciongrafo2 = 2 Then

    grafo2ymin = -10
    grafo2ymax = 10

Elseif opciongrafo2 = 3 Then

    grafo2ymin = -10
    grafo2ymax = 10

Elseif opciongrafo2 = 4 Then

    grafo2ymin = -5
    grafo2ymax = 5

End If

grafo2xmin = 0
grafo2xmax = 10

tiempo2 = grafo2xmax / .1

graficos.g2ymin.Caption = Format(grafo2ymin, "0.000")
graficos.g2ymax.Caption = Format(grafo2ymax, "0.000")
graficos.g2xmin.Caption = Format(grafo2xmin, "0")
graficos.g2xmax.Caption = Format$(tiempo2, "0")

```

```
grafo1.Scale (grafo1xmin, grafo1ymax)-(grafo1xmax, grafo1ymin)
grafo2.Scale (grafo2xmin, grafo2ymax)-(grafo2xmax, grafo2ymin)
```

```
Elseif (CambioEscalas.Option1 = False) Then
```

```
graficos.g1ymin.Caption = Format(grafo1ymin, "0.000")
graficos.g1ymax.Caption = Format(grafo1ymax, "0.000")
graficos.g1xmin.Caption = Format(grafo1xmin, "0")
graficos.g1xmax.Caption = Format(grafo1xmax, "0")
```

```
graficos.g2ymin.Caption = Format(grafo2ymin, "0.000")
graficos.g2ymax.Caption = Format(grafo2ymax, "0.000")
graficos.g2xmin.Caption = Format(grafo2xmin, "0")
graficos.g2xmax.Caption = Format(grafo2xmax, "0")
```

```
grafo1.Scale (grafo1xmin, grafo1ymax)-(grafo1xmax, grafo1ymin)
grafo2.Scale (grafo2xmin, grafo2ymax)-(grafo2xmax, grafo2ymin)
```

```
End If
```

```
Elseif TiempoReal = 1 Then
```

```
graficos.ceroygrafo1.Visible = True
graficos.ceroygrafo2.Visible = True
```

```
grafo1xmin = 0
grafo1xmax = 10
grafo1ymax = 10
grafo1ymin = 0
```

```
grafo2xmin = 0
grafo2xmax = 10
grafo2ymax = 10
grafo2ymin = 0
```

```
tiempo1 = grafo1xmax / .1
tiempo2 = grafo2xmax / .1
```

```
graficos.g1ymax.Caption = Format$("10 V")
graficos.g1ymin.Caption = Format$("0 V")
graficos.g1xmin.Caption = Format(grafo1xmin, "0")
graficos.g1xmax.Caption = Format(tiempo1, "0")
```

```
graficos.g2ymax.Caption = Format$("10 V")
graficos.g2ymin.Caption = Format$("0 V")
graficos.g2xmin.Caption = Format(grafo2xmin, "0")
graficos.g2xmax.Caption = Format(tiempo2, "0")
```

```
grafo1.Scale (grafo1xmin, grafo1ymax)-(grafo1xmax, grafo1ymin)
grafo2.Scale (grafo2xmin, grafo2ymax)-(grafo2xmax, grafo2ymin)
```

```
End If
```

```
Muestreo.Enabled = True
```

```
iniciar.Enabled = False
parar.Enabled = True
encerar
grafo1.Cls
grafo2.Cls
```

```
Salir:
```

```
Exit Sub
```

```
Error_en_escalas:
```

```
    If (Err = 11) Then
```

```
        MsgBox "División por cero. Revise los valores ingresados de las escalas", 16, "Control de Mínima Varianza"
```

```
    ElseIf (Err = 6) Then
```

```
        MsgBox "Overflow", 16, "Control de Mínima Varianza"
```

```
    End If
```

```
Resume Salir
```

```
End Sub
```

```
Sub Muestreo_Timer ()
```

```
On Error GoTo Error_en_escalas1
```

```
Randomize
```

```
et = (Rnd - .5)
```

```
'If TiempoReal = 1 Then
```

```
    'A = D
```

```
    If (ModoTrabajo = 101 Or ModoTrabajo = 111 Or ModoTrabajo = 201 Or ModoTrabajo = 211) Then
```

```
        vt = Sqr(VarRuido) * (et + c1 * etm1 + c2 * etm2 + c3 * etm3 + c4 * etm4)
```

```
    'A <> D
```

```
    ElseIf (ModoTrabajo = 100 Or ModoTrabajo = 110 Or ModoTrabajo = 200 Or ModoTrabajo = 210) Then
```

```
        vt = Sqr(VarRuido) * (et + AC1 * etm1 + AC2 * etm2 + AC3 * etm3 + AC4 * etm4 + AC5 * etm5 + AC6 * etm6 + AC7 * etm7 + AC8 * etm8)
```

```
    End If
```

```
'End If
```

```
'GRAFICACION
```

```
If (opciongrafo1 = 1 Or opciongrafo2 = 1) Then
```

```

Salida_sin_CMV

If TiempoReal = 1 Then

    yst = (yst / 3) + 5
    ystm1 = (ystm1 / 3) + 5

End If

If (opciongrafo1 = 1) Then

    grafo1.Line (grafo1x1 - .1, ystm1)-(grafo1x1, yst), QBColor(13)

'CALCULO DEL MAXIMO DE LA CURVA

    maxy1 = Abs(ystm1)

    If (maxy1 < Abs(yst)) Then
        maxy1 = Abs(yst)
    End If

End If

If (opciongrafo2 = 1) Then

    grafo2.Line (grafo2x1 - .1, ystm1)-(grafo2x1, yst), QBColor(13)

'CALCULO DEL MAXIMO DE LA CURVA

    maxy2 = Abs(ystm1)

    If maxy2 < Abs(yst) Then
        maxy2 = Abs(yst)
    End If

End If

If TiempoReal = 1 Then

    yst = (yst - 5) * 3
    ystm1 = (ystm1 - 5) * 3

End If

End If

If (opciongrafo1 = 2 Or opciongrafo1 = 3 Or opciongrafo2 = 2 Or opciongrafo2 = 3) Then

    Control

    If TiempoReal = 1 Then

        yct = (yct / 3) + 5
        yctm1 = (yctm1 / 3) + 5
        ut = (ut / 3) + 5
        utm1 = (utm1 / 3) + 5
        utm2 = (utm2 / 3) + 5
    
```

```

End If

If (opciongrafo1 = 2) Then
    grafo1.Line (grafo1x1 - .1, yctm1)-(grafo1x1, yct), QBColor(9)
'CALCULO DEL MAXIMO DE LA CURVA
    maxy1 = Abs(yctm1)

    If maxy1 < Abs(yct) Then
        maxy1 = Abs(yct)
    End If

End If

If (opciongrafo1 = 3) Then
    If (b0 < 0 Or b0 > 0) Then
        grafo1.Line (grafo1x1 - .1, utm1)-(grafo1x1, ut), QBColor(10)
    ElseIf (b0 = 0) Then
        grafo1.Line (grafo1x1 - .1, utm2)-(grafo1x1, utm1), QBColor(10)
    End If

'CALCULO DEL MAXIMO DE LA CURVA
    maxy1 = Abs(utm2)

    If maxy1 < Abs(utm1) Then
        maxy1 = Abs(utm1)
    End If

End If

If (opciongrafo2 = 2) Then
    grafo2.Line (grafo2x1 - .1, yctm1)-(grafo2x1, yct), QBColor(9)
'CALCULO DEL MAXIMO DE LA CURVA
    maxy2 = Abs(yctm1)

    If maxy2 < Abs(yct) Then
        maxy2 = Abs(yct)
    End If

End If

If (opciongrafo2 = 3) Then
    If (b0 < 0 Or b0 > 0) Then
        grafo2.Line (grafo2x1 - .1, utm1)-(grafo2x1, ut), QBColor(10)
    ElseIf (b0 = 0) Then

```

```

    grafo2.Line (grafo2x1 - .1, utm2)-(grafo2x1, utm1), QBColor(10)

    End If

'CALCULO DEL MAXIMO DE LA CURVA
    maxy2 = Abs(utm2)

    If maxy2 < Abs(utm1) Then
        maxy2 = Abs(utm1)
    End If

End If

    If TiempoReal = 1 Then

        yct = (yct - 5) * 3
        yctm1 = (yctm1 - 5) * 3
        ut = (ut - 5) * 3
        utm1 = (utm1 - 5) * 3
        utm2 = (utm2 - 5) * 3

    End If

End If

If (opciongrafo1 = 1 Or opciongrafo1 = 2) Then

    Referencia1

End If

If (opciongrafo2 = 1 Or opciongrafo2 = 2) Then

    Referencia2

End If

If (opciongrafo1 = 4 Or opciongrafo2 = 4) Then

    If (opciongrafo1 = 4) Then

        If TiempoReal = 1 Then

            vtm1 = (vtm1 / 3) + 5
            vt = (vt / 3) + 5

        End If

        grafo1.Line (grafo1x1 - .1, vtm1)-(grafo1x1, vt), QBColor(4)

        If TiempoReal = 1 Then

            vtm1 = (vtm1 - 5) * 3
            vt = (vt - 5) * 3

        End If
    
```

'CALCULO DEL MAXIMO DE LA CURVA

maxy1 = Abs(vtm1)

If maxy1 < Abs(vt) Then

maxy1 = Abs(vt)

End If

End If

If (opciongrafo2 = 4) Then

eetm1 = Sqr(VarRuido) * etm1

eet = Sqr(VarRuido) * et

If TiempoReal = 1 Then

eetm1 = (eetm1 / 3) + 5

eet = (eet / 3) + 5

End If

grafo2.Line (grafo2x1 - .1, eetm1)-(grafo2x1, eet)

If TiempoReal = 1 Then

eetm1 = (eetm1 - 5) * 3

eet = (eet - 5) * 3

End If

'CALCULO DEL MAXIMO DE LA CURVA

maxy2 = Abs(eetm1)

If maxy2 < Abs(eet) Then

maxy2 = Abs(eet)

End If

End If

End If

If TiempoReal = 1 Then

refer = despvsetpoint.Value

setpoint.Caption = (refer / 3) + 5

Else

setpoint.Caption = despvsetpoint.Value

End If

wtm1 = despvsetpoint.Value

'REVALORACION DE LAS VARIABLES

Corrido_de_Valores

```

    grafo2x1 = 0
If (CambioEscalas.Option1 = True) Then

    If (maxy2 < maxy2tm1) Then

        maxy2 = maxy2tm1

    End If

    grafo2ymax = 1.5 * maxy2
    grafo2ymin = -1.5 * maxy2
    grafo2xmin = 0
    grafo2xmax = 10

    g2ymin.Caption = Format(grafo2ymin, "0.000")
    g2ymax.Caption = Format(grafo2ymax, "0.000")
    g2xmin.Caption = Format(grafo2xmin, "0")

    g2xmax.Caption = Format(tiempo2, "0")

    grafo2.Scale (grafo2xmin, grafo2ymax)-(grafo2xmax, grafo2ymin)

Elseif (CambioEscalas.Option1 = False) Then

    g2ymin.Caption = Format(grafo2ymin, "0.000")
    g2ymax.Caption = Format(grafo2ymax, "0.000")
    g2xmin.Caption = Format(grafo2xmin, "0")

    g2xmax.Caption = Format(tiempo2, "0")

    grafo2.Scale (grafo2xmin, grafo2ymax)-(grafo2xmax, grafo2ymin)

End If
End If

End If
*****

If TiempoReal = 1 Then

If grafo1x1 > (1.1 * grafo1xmax) Then

    grafo1.Cls
    grafo1x1 = 0

Elseif grafo2x1 > (1.1 * grafo2xmax) Then

    grafo2.Cls
    grafo2x1 = 0

End If

End If

maxy1tm1 = maxy1
maxy2tm1 = maxy2

```

```
parar.Enabled = True
```

```
'INCREMENTO DE LA VARIABLE X
```

```
grafo1x1 = grafo1x1 + .1
```

```
grafo2x1 = grafo2x1 + .1
```

```
Salir1:
```

```
Exit Sub
```

```
Error_en_escalas1:
```

```
    If (Err = 11) Then
```

```
        MsgBox "División por cero. Revise los valores ingresados de las escalas o reinicie la  
        graficación", 16, "Control de Mínima Varianza"
```

```
    Elseif (Err = 6) Then
```

```
        MsgBox "Overflow", 16, "Control de Mínima Varianza"
```

```
    End If
```

```
Resume Salir1
```

```
End Sub
```

```
Sub opciones_Click ()
```

```
Muestreo.Enabled = False
```

```
End Sub
```

```
Sub Parar_Click ()
```

```
Muestreo.Enabled = False
```

```
parar.Enabled = False
```

```
iniciar.Enabled = True
```

```
End Sub
```

```
Sub Salir_Click ()
```

```
    Unload graficos
```

```
End Sub
```

B.17 SUBRUTINAS DE LA PANTALLA CAMBIAR TIEMPO DE MUESTREO

```
Sub Form_Load ()
```

```
    Text1.Text = graficos.Muestreo.Interval
```

```
End Sub
```

```
Sub Text1_Change ()
```

```
    If (Text1.Text = "") Then
```

```

graficos.Muestreo.Interval = 500
Beep
Text1.Text = "500"

Elseif (Val(Text1.Text) = 50000) Then

graficos.Muestreo.Interval = 500
Beep
Text1.Text = ""
Text1.Text = "500"

Else

graficos.Muestreo.Interval = Text1.Text

End If

End Sub

Sub Text1_KeyPress (KeyAscii As Integer)

c$ = Chr$(KeyAscii)

If c$ < "0" Or c$ > "9" Then

    If Asc(c$) = 8 Then
        KeyAscii = 8
    Elseif c$ <> "8" Then
        KeyAscii = 0
    End If
    Beep
End If

End If

End Sub

Sub TMuestreoSalir_Click ()

If graficos.Muestreo.Interval <= 0 Or graficos.Muestreo.Interval > 50000 Then

graficos.Muestreo.Interval = 500

End If

tiempomuestreo.Hide
graficos.Grafo1.Cls
graficos.Grafo2.Cls

End Sub

```

B.18 SUBROUTINAS DE LA PANTALLA DE CAMBIO DE ESCALAS

```
Sub Form_Load ()
```

```
grafo1x1 = 0  
grafo2x1 = 0
```

```
If (option1 = True) Then
```

```
If opciongrafo1 = 1 Then
```

```
    grafo1ymin = -50  
    grafo1ymax = 50
```

```
Elseif opciongrafo1 = 2 Then
```

```
    grafo1ymin = -10  
    grafo1ymax = 10
```

```
Elseif opciongrafo1 = 3 Then
```

```
    grafo1ymin = -10  
    grafo1ymax = 10
```

```
Elseif opciongrafo1 = 4 Then
```

```
    grafo1ymin = -.5  
    grafo1ymax = .5
```

```
End If
```

```
    grafo1xmin = 0  
    grafo1xmax = 10  
    graficos.g1ymin.Caption = Format(grafo1ymin, "0.000")  
    graficos.g1ymax.Caption = Format(grafo1ymax, "0.000")  
    graficos.g1xmin.Caption = Format(grafo1xmin, "0")  
    graficos.g1xmax.Caption = Format(grafo1xmax, "0")
```

```
If opciongrafo2 = 1 Then
```

```
    grafo2ymin = -50  
    grafo2ymax = 50
```

```
Elseif opciongrafo2 = 2 Then
```

```
    grafo2ymin = -10  
    grafo2ymax = 10
```

```
Elseif opciongrafo2 = 3 Then
```

```
    grafo2ymin = -10  
    grafo2ymax = 10
```

```
Elseif opciongrafo2 = 4 Then
```

```

    grafo2ymin = -.5
    grafo2ymax = .5

End If

    grafo2xmin = 0
    grafo2xmax = 10
    graficos.g2ymin.Caption = Format(grafo2ymin, "0.000")
    graficos.g2ymax.Caption = Format(grafo2ymax, "0.000")
    graficos.g2xmin.Caption = Format(grafo2xmin, "0")
    graficos.g2xmax.Caption = Format(grafo2xmax, "0")

Elseif (option1 = False) Then

    graficos.g1ymin.Caption = Format(grafo1ymin, "0.000")
    graficos.g1ymax.Caption = Format(grafo1ymax, "0.000")
    graficos.g1xmin.Caption = Format(grafo1xmin, "0")
    graficos.g1xmax.Caption = Format(grafo1xmax, "0")

    graficos.g2ymin.Caption = Format(grafo2ymin, "0.000")
    graficos.g2ymax.Caption = Format(grafo2ymax, "0.000")
    graficos.g2xmin.Caption = Format(grafo2xmin, "0")
    graficos.g2xmax.Caption = Format(grafo2xmax, "0")

End If

Text1.Text = Format(grafo1xmax / .1)
Text2.Text = Format(grafo1xmin)
Text3.Text = Format(grafo1ymax)
Text4.Text = Format(grafo1ymin)

Text5.Text = Format(grafo2xmax / .1)
text6.Text = Format(grafo2xmin)
Text7.Text = Format(grafo2ymax)
Text8.Text = Format(grafo2ymin)

Text1.Enabled = False
Text2.Enabled = False
Text3.Enabled = False
Text4.Enabled = False
Text5.Enabled = False
text6.Enabled = False
Text7.Enabled = False
Text8.Enabled = False

End Sub

Sub Option1_Click ()

    grafo1x1 = 0
    grafo2x1 = 0

    If opciongrafo1 = 1 Then

        grafo1ymin = -50
        grafo1ymax = 50

```

```
Elseif opciongrafo1 = 2 Then
```

```
    grafo1ymin = -10  
    grafo1ymax = 10
```

```
Elseif opciongrafo1 = 3 Then
```

```
    grafo1ymin = -10  
    grafo1ymax = 10
```

```
Elseif opciongrafo1 = 4 Then
```

```
    grafo1ymin = -.5  
    grafo1ymax = .5
```

```
End If
```

```
    grafo1xmin = 0  
    grafo1xmax = 10  
    graficos.g1ymin.Caption = Format(grafo1ymin, "0.000")  
    graficos.g1ymax.Caption = Format(grafo1ymax, "0.000")  
    graficos.g1xmin.Caption = Format(grafo1xmin, "0")  
    graficos.g1xmax.Caption = Format(tiempo1, "0")
```

```
If opciongrafo2 = 1 Then
```

```
    grafo2ymin = -50  
    grafo2ymax = 50
```

```
Elseif opciongrafo2 = 2 Then
```

```
    grafo2ymin = -10  
    grafo2ymax = 10
```

```
Elseif opciongrafo2 = 3 Then
```

```
    grafo2ymin = -10  
    grafo2ymax = 10
```

```
Elseif opciongrafo2 = 4 Then
```

```
    grafo2ymin = -.5  
    grafo2ymax = .5
```

```
End If
```

```
    grafo2xmin = 0  
    grafo2xmax = 10  
    graficos.g2ymin.Caption = Format(grafo2ymin, "0.000")  
    graficos.g2ymax.Caption = Format(grafo2ymax, "0.000")  
    graficos.g2xmin.Caption = Format(grafo2xmin, "0")  
    graficos.g2xmax.Caption = Format(tiempo2, "0")
```

```
Text1.Text = Format(tiempo1)  
Text2.Text = Format(grafo1xmin)  
Text3.Text = Format(grafo1ymax)  
Text4.Text = Format(grafo1ymin)
```

```
Text5.Text = Format(tiempo2)  
text6.Text = Format(grafo2xmin)
```

```
graficos.Grafo1.Cls
graficos.Grafo2.Cls
```

```
End If
```

```
Salir:
```

```
Exit Sub
```

```
Error_en_escalas:
```

```
    If (Err = 11) Then
```

```
        MsgBox "División por cero. Revise los valores ingresados de las escalas", 16, "Control de Mínima Varianza"
```

```
    ElseIf (Err = 6) Then
```

```
        MsgBox "Overflow", 16, "Control de Mínima Varianza"
```

```
    End If
```

```
Resume Salir
```

```
End Sub
```

```
Sub Text1_Change ()
```

```
    If (Text1.Text = "") Then
```

```
        Beep
```

```
        Text1.Text = "100"
```

```
    ElseIf (Val(Text1.Text) = 0) Then
```

```
        Beep
```

```
        Text1.Text = ""
```

```
        Text1.Text = "100"
```

```
    End If
```

```
    grafo1xmax = Val(Text1.Text) / 10
```

```
    tiempo1 = Val(Text1.Text)
```

```
End Sub
```

```
Sub Text1_KeyPress (KeyAscii As Integer)
```

```
    c$ = Chr$(KeyAscii)
```

```
    If c$ < "0" Or c$ > "9" Then
```

```
        If Asc(c$) = 8 Then
```

```
            KeyAscii = 8
```

```
        ElseIf c$ <> "8" Then
```

```
            KeyAscii = 0
```

```

        Beep
    End If
End If
End Sub

Sub Text2_Change ()
    grafo1xmin = Val(Text2.Text)
End Sub

Sub Text2_KeyPress (KeyAscii As Integer)
    c$ = Chr$(KeyAscii)

    If c$ < "0" Or c$ > "9" Then

        If Asc(c$) = 8 Then
            KeyAscii = 8
        ElseIf c$ <> "8" Then
            KeyAscii = 0
        End If
        Beep
    End If
End If
End Sub

Sub Text3_Change ()
    If (Text3.Text = "") Then
        Beep

        If opciongrafo1 = 1 Then
            Text3.Text = "50"
        ElseIf opciongrafo1 = 2 Then
            Text3.Text = "10"
        ElseIf opciongrafo1 = 3 Then
            Text3.Text = 10
        ElseIf opciongrafo1 = 4 Then
            Text3.Text = "1"
        End If
    ElseIf (Val(Text3.Text) = 0) Then

```

```

    Text4.Text = "-10"
Elseif opciongrafo1 = 3 Then
    Text4.Text = "-10"
Elseif opciongrafo1 = 4 Then
    Text4.Text = "-1"
End If
Elseif (Val(Text4.Text) = 0) Then
Beep
Text4.Text = ""
    If opciongrafo1 = 1 Then
        Text4.Text = "-50"
    Elseif opciongrafo1 = 2 Then
        Text4.Text = "-10"
    Elseif opciongrafo1 = 3 Then
        Text4.Text = "-10"
    Elseif opciongrafo1 = 4 Then
        Text4.Text = "-1"
    End If
End If

grafo1ymin = Val(Text4.Text)
End Sub

Sub Text4_KeyPress (KeyAscii As Integer)
    c$ = Chr$(KeyAscii)
    If c$ < "0" Or c$ > "9" Then

        If Asc(c$) = 8 Then
            KeyAscii = 8
        Elseif c$ <> "8" Then
            KeyAscii = 0
        Beep
    End If

```

```

    End If

End Sub

Sub Text5_Change ()

    If (Text5.Text = "") Then

        Beep
        Text5.Text = "100"

    ElseIf (Val(Text5.Text) = 0) Then

        Beep
        Text5.Text = ""
        Text5.Text = "100"

    End If

    grafo2xmax = Val(Text5.Text) / 10
    tiempo2 = Val(Text5.Text)
End Sub

Sub Text5_KeyPress (KeyAscii As Integer)

    c$ = Chr$(KeyAscii)

    If c$ < "0" Or c$ > "9" Then

        If Asc(c$) = 8 Then

            KeyAscii = 8

        ElseIf c$ <> "8" Then

            KeyAscii = 0

        Beep

    End If

End If

End Sub

Sub Text6_Change ()
grafo2xmin = Val(text6.Text)
End Sub

Sub Text6_KeyPress (KeyAscii As Integer)

    c$ = Chr$(KeyAscii)

    If c$ < "0" Or c$ > "9" Then

        If Asc(c$) = 8 Then

```

```

    KeyAscii = 8
    Elself c$ <> "8" Then
        KeyAscii = 0
        Beep
    End If
End If
End Sub
Sub Text7_Change ()
If (Text7.Text = "") Then
Beep
    If opciongrafo2 = 1 Then
        Text7.Text = "50"
    Elself opciongrafo2 = 2 Then
        Text7.Text = "10"
    Elself opciongrafo2 = 3 Then
        Text7.Text = 10
    Elself opciongrafo2 = 4 Then
        Text7.Text = "1"
    End If
Elself (Val(Text7.Text) = 0) Then
Beep
Text7.Text = ""
    If opciongrafo2 = 1 Then
        Text7.Text = "50"
    Elself opciongrafo2 = 2 Then
        Text7.Text = "10"
    Elself opciongrafo2 = 3 Then
        Text7.Text = 10
    Elself opciongrafo2 = 4 Then
        Text7.Text = "1"
    End If

```

```

End If

grafo2ymax = Val(Text7.Text)
End Sub

Sub Text7_KeyPress (KeyAscii As Integer)
    c$ = Chr$(KeyAscii)
    If c$ < "0" Or c$ > "9" Then

        If Asc(c$) = 8 Then
            KeyAscii = 8
        ElseIf c$ <> "8" Then
            KeyAscii = 0
        Beep
    End If
End If

End Sub

Sub Text8_Change ()
    If (Text8.Text = "") Then
        Beep
        If opciongrafo2 = 1 Then
            Text8.Text = "-50"
        ElseIf opciongrafo2 = 2 Then
            Text8.Text = "-10"
        ElseIf opciongrafo2 = 3 Then
            Text8.Text = "-10"
        ElseIf opciongrafo2 = 4 Then
            Text8.Text = "-1"
        End If
    ElseIf (Val(Text8.Text) = 0) Then
        Beep
        Text8.Text = ""
        If opciongrafo2 = 1 Then

```

```

    Text8.Text = "-50"
Elseif opciongrafo2 = 2 Then
    Text8.Text = "-10"
Elseif opciongrafo2 = 3 Then
    Text8.Text = "-10"
Elseif opciongrafo2 = 4 Then
    Text8.Text = "-1"
End If
End If
grafo2ymin = Val(Text8.Text)
End Sub
Sub Text8_KeyPress (KeyAscii As Integer)
    c$ = Chr$(KeyAscii)
    If c$ < "0" Or c$ > "9" Then
        If Asc(c$) = 8 Then
            KeyAscii = 8
        Elseif c$ <> "8" Then
            KeyAscii = 0
        Beep
    End If
End If
End Sub

```

B.19 SUBROUTINA PARA EL CÁLCULO DE LA SEÑAL DE SALIDA SIN CONTROL DE MÍNIMA VARIANZA

```

Sub Salida_sin_CMV ()
On Error GoTo Error_en_calculosalida
If TiempoReal = 0 Then
'REGULACION
If (ModoTrabajo > 100 And ModoTrabajo < 112) Then

```

$yst = (\text{Sqr}(\text{VarRuido}) * (et + c1 * etm1 + c2 * etm2 + c3 * etm3 + c4 * etm4) - (d1 * ystm1 + d2 * ystm2 + d3 * ystm3 + d4 * ystm4))$

'SEGUIMIENTO

'A = D

Elseif (ModoTrabajo = 211 Or ModoTrabajo = 201) Then

ust = wtm1

If (d = 0) Then

TerminoUs = (b0 * ust + b1 * utm1 + b2 * ustm2 + b3 * ustm3 + b4 * ustm4)

Elseif (d = 1) Then

TerminoUs = (b0 * ustm1 + b1 * ustm2 + b2 * ustm3 + b3 * ustm4 + b4 * ustm5)

Elseif (d = 2) Then

TerminoUs = (b0 * ustm2 + b1 * ustm3 + b2 * ustm4 + b3 * ustm5 + b4 * ustm6)

End If

$yst = \text{TerminoUs} + \text{Sqr}(\text{VarRuido}) * (et + c1 * etm1 + c2 * etm2 + c3 * etm3 + c4 * etm4) - (a1 * ystm1 + a2 * ystm2 + a3 * ystm3 + a4 * ystm4)$

'A <> D

Elseif (ModoTrabajo = 210 Or ModoTrabajo = 200) Then

ust = wtm1

If (d = 0) Then

TerminoUs = (b0 * ust + BD1 * ustm1 + BD2 * ustm2 + BD3 * ustm3 + BD4 * ustm4 + BD5 * ustm5 + BD6 * ustm6 + BD7 * ustm7 + BD8 * ustm8)

Elseif (d = 1) Then

TerminoUs = (b0 * ustm1 + BD1 * ustm2 + BD2 * ustm3 + BD3 * ustm4 + BD4 * ustm5 + BD5 * ustm6 + BD6 * ustm7 + BD7 * ustm8 + BD8 * ustm9)

Elseif (d = 2) Then

TerminoUs = (b0 * ustm2 + BD1 * ustm3 + BD2 * ustm4 + BD3 * ustm5 + BD4 * ustm6 + BD5 * ustm7 + BD6 * ustm8 + BD7 * ustm9 + BD8 * ustm10)

End If

$yst = \text{TerminoUs} + \text{Sqr}(\text{VarRuido}) * (et + AC1 * etm1 + AC2 * etm2 + AC3 * etm3 + AC4 * etm4 + AC5 * etm5 + AC6 * etm6 + AC7 * etm7 + AC8 * etm8) - (AD1 * ystm1 + AD2 * ystm2 + AD3 * ystm3 + AD4 * ystm4 + AD5 * ystm5 + AD6 * ystm6 + AD7 * ystm7 + AD8 * ystm8)$

End If

Elseif TiempoReal = 1 Then

'REGULACION

If (ModoTrabajo > 100 And ModoTrabajo < 112) Then

yst = (Sqr(VarRuido) * (et + c1 * etm1 + c2 * etm2 + c3 * etm3 + c4 * etm4) - (d1 * ystm1 + d2 * ystm2 + d3 * ystm3 + d4 * ystm4))

'SEGUIMIENTO

'A = D

Elseif (ModoTrabajo = 211 Or ModoTrabajo = 201) Then

ust = wtm1

referenciasincmv = (128 * ust / 15) + 127

zz = enviar(&H336, referenciasincmv)

medidasincmv = medir(&H336, medida)

ypst = 15 * (medidasincmv - 127) / 128

'A <> D

Elseif (ModoTrabajo = 210 Or ModoTrabajo = 200) Then

ust = wtm1

referenciasincmv = (126 * ust / 15) + 15

zz = enviar(&H336, referenciasincmv)

medidasincmv = medir(&H336, medida)

ypst = 15 * (medidasincmv - 127) / 128

End If

yst = ypst + vt

End If

Salir:

Exit Sub

Error_en_calculosalida:

If (Err = 11) Then

MsgBox "División por cero. Revise los valores ingresados de las escalas", 48, "Control de Mínima Varianza"

Elseif (Err = 6) Then

MsgBox "Overflow", 16, "Control de Mínima Varianza"

End If

Resume Salir

End Sub

B.20 SUBROUTINAS PARA GRAFICACIÓN DE LAS CURVAS DE REFERENCIA

Sub Referencia1 ()

If (ModoTrabajo > 200 Or ModoTrabajo = 200) Then

Graficos.Grafo1.Line (grafo1x1 - .1, wtm2)-(grafo1x1, wtm1), QBColor(12)

End If

End Sub

Sub Referencia2 ()

If (ModoTrabajo > 200 Or ModoTrabajo = 200) Then

Graficos.Grafo2.Line (grafo2x1 - .1, wtm2)-(grafo2x1, wtm1), QBColor(12)

End If

End Sub

B.21 SUBROUTINA PARA MOSTRAR LOS VALORES DE LAS VARIABLES QUE ESTÁN SIENDO GRAFICADAS

Sub valoresgrafos ()

If opciongrafo1 = 1 Then

graficos.ValorGrafo1.Caption = Format(ystm1, "0.0000")

Elseif opciongrafo1 = 2 Then

graficos.ValorGrafo1.Caption = Format(yctm1, "0.0000")

```

Elseif opciongrafo1 = 3 Then
    graficos.ValorGrafo1.Caption = Format(utm1, "0.0000")
Elseif opciongrafo1 = 4 Then
    graficos.ValorGrafo1.Caption = Format(etm1, "0.0000")
End If

If opciongrafo2 = 1 Then
    graficos.ValorGrafo2.Caption = Format(ystm1, "0.0000")
Elseif opciongrafo2 = 2 Then
    graficos.ValorGrafo2.Caption = Format(yctm1, "0.0000")
Elseif opciongrafo2 = 3 Then
    graficos.ValorGrafo2.Caption = Format(utm1, "0.0000")
Elseif opciongrafo2 = 4 Then
    graficos.ValorGrafo2.Caption = Format(etm1, "0.0000")
End If
End Sub

```

B.22 SUBROUTINA PARA REALIZAR EL DESPLAZAMIENTO EN EL TIEMPO DE LA VARIABLES

```
Sub Corrido_de_Valores ()
```

```

ypt = ypt1
ypt1 = ypt2

```

```

wtm9 = wtm8
wtm8 = wtm7
wtm7 = wtm6
wtm6 = wtm5
wtm5 = wtm4
wtm4 = wtm3
wtm3 = wtm2
wtm2 = wtm1
wtm1 = wt

```

```

etm8 = etm7
etm7 = etm6
etm6 = etm5
etm5 = etm4
etm4 = etm3
etm3 = etm2
etm2 = etm1
etm1 = et

```

```
ystm8 = ystm7  
ystm7 = ystm6  
ystm6 = ystm5  
ystm5 = ystm4  
ystm4 = ystm3  
ystm3 = ystm2  
ystm2 = ystm1  
ystm1 = yst
```

```
ustm10 = ustm9  
ustm9 = ustm8  
ustm8 = ustm7  
ustm7 = ustm6  
ustm6 = ustm5  
ustm5 = ustm4  
ustm4 = ustm3  
ustm3 = ustm2  
ustm2 = ustm1  
ustm1 = ust
```

```
yctm8 = yctm7  
yctm7 = yctm6  
yctm6 = yctm5  
yctm5 = yctm4  
yctm4 = yctm3  
yctm3 = yctm2  
yctm2 = yctm1  
yctm1 = yct
```

If (b0 < 0 Or b0 > 0) Then

```
utm10 = utm9  
utm9 = utm8  
utm8 = utm7  
utm7 = utm6  
utm6 = utm5  
utm5 = utm4  
utm4 = utm3  
utm3 = utm2  
utm2 = utm1  
utm1 = ut
```

End If

If (b0 = 0) Then

```
utm10 = utm9  
utm9 = utm8  
utm8 = utm7  
utm7 = utm6  
utm6 = utm5  
utm5 = utm4  
utm4 = utm3  
utm3 = utm2  
utm2 = utm1
```

End If

End Sub

B.23 SUBROUTINA PARA ENCERAR LAS VARIABLES

Sub encerar ()

ypt2 = 0
ypt1 = 0
ypt = 0
salidaplanta = 0
vt = 0

Graficos.DespVSetPoint.Value = 0
Graficos.SetPoint.caption = 0

wtm8 = 0
wtm7 = 0
wtm6 = 0
wtm5 = 0
wtm4 = 0
wtm3 = 0
wtm2 = 0
wtm1 = 0
wt = 0

etm8 = 0
etm7 = 0
etm6 = 0
etm5 = 0
etm4 = 0
etm3 = 0
etm2 = 0
etm1 = 0
et = 0

ystm8 = 0
ystm7 = 0
ystm6 = 0
ystm5 = 0
ystm4 = 0
ystm3 = 0
ystm2 = 0
ystm1 = 0
yst = 0

ustm10 = 0
ustm9 = 0
ustm8 = 0
ustm7 = 0
ustm6 = 0
ustm5 = 0
ustm4 = 0
ustm3 = 0
ustm2 = 0
ustm1 = 0
ust = 0

yctm8 = 0
yctm7 = 0

```
yctm6 = 0
yctm5 = 0
yctm4 = 0
yctm3 = 0
yctm2 = 0
yctm1 = 0
yct = 0
```

```
utm10 = 0
utm9 = 0
utm8 = 0
utm7 = 0
utm6 = 0
utm5 = 0
utm4 = 0
utm3 = 0
utm2 = 0
utm1 = 0
ut = 0
```

```
End Sub
```

B.24 RUTINA PARA CALCULAR LA SEÑAL DE CONTROL Y LA SALIDA CONTROLADA

```
Sub Control ()
```

```
On Error GoTo Error_en_calculocontrol
```

```
'RUTINAS PARA SIMULACION
```

```
If (TiempoReal = 0) Then
```

```
'REGULACION
```

```
'A = D
```

```
If (ModoTrabajo = 111 Or ModoTrabajo = 101) Then
```

```
    If (b0 < 0 Or b0 > 0) Then
```

```
        ut = (1 / b0) * -(TU1 * utm1 + TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 +
TU6 * utm6) - (g0 * yctm1 + g1 * yctm2 + g2 * yctm3 + g3 * yctm4))
```

```
    If (d = 0) Then
```

```
        TerminoU = (b0 * ut + b1 * utm1 + b2 * utm2 + b3 * utm3 + b4 * utm4)
```

```
    Elseif (d = 1) Then
```

```
        TerminoU = (b0 * utm1 + b1 * utm2 + b2 * utm3 + b3 * utm4 + b4 * utm5)
```

Elseif (d = 2) Then

TerminoU = (b0 * utm2 + b1 * utm3 + b2 * utm4 + b3 * utm5 + b4 * utm6)

End If

yct = TerminoU + Sqr(VarRuido) * (et + c1 * etm1 + c2 * etm2 + c3 * etm3 + c4 * etm4) - (a1 * yctm1 + a2 * yctm2 + a3 * yctm3 + a4 * yctm4)

Elseif (b0 = 0) Then

utm1 = (1 / TU1) * (-(TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 + TU6 * utm6) - (g0 * yctm1 + g1 * yctm2 + g2 * yctm3 + g3 * yctm4))

If (d = 0) Then

TerminoU = (b1 * utm1 + b2 * utm2 + b3 * utm3 + b4 * utm4)

Elseif (d = 1) Then

TerminoU = (b1 * utm2 + b2 * utm3 + b3 * utm4 + b4 * utm5)

Elseif (d = 2) Then

TerminoU = (b1 * utm3 + b2 * utm4 + b3 * utm5 + b4 * utm6)

End If

yct = TerminoU + Sqr(VarRuido) * (et + c1 * etm1 + c2 * etm2 + c3 * etm3 + c4 * etm4) - (a1 * yctm1 + a2 * yctm2 + a3 * yctm3 + a4 * yctm4)

End If

'A <> D

Elseif (ModoTrabajo = 110 Or ModoTrabajo = 100) Then

If (b0 < 0 Or b0 > 0) Then

ut = (1 / b0) * (-(TU1 * utm1 + TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 + TU6 * utm6 + TU7 * utm7 + TU8 * utm8 + TU9 * utm9 + TU10 * utm10) - (g0 * yctm1 + AG1 * yctm2 + AG2 * yctm3 + AG3 * yctm4 + AG4 * yctm5 + AG5 * yctm6 + AG6 * yctm7 + AG7 * yctm8))

If (d = 0) Then

TerminoU = (b0 * ut + BD1 * utm1 + BD2 * utm2 + BD3 * utm3 + BD4 * utm4 + BD5 * utm5 + BD6 * utm6 + BD7 * utm7 + BD8 * utm8)

Elseif (d = 1) Then

TerminoU = (b0 * utm1 + BD1 * utm2 + BD2 * utm3 + BD3 * utm4 + BD4 * utm5 + BD5 * utm6 + BD6 * utm7 + BD7 * utm8 + BD8 * utm9)

Elseif (d = 2) Then

TerminoU = (b0 * utm2 + BD1 * utm3 + BD2 * utm4 + BD3 * utm5 + BD4 * utm6 + BD5 * utm7 + BD6 * utm8 + BD7 * utm9 + BD8 * utm10)

End If

yct = TerminoU + Sqr(VarRuido) * (et + AC1 * etm1 + AC2 * etm2 + AC3 * etm3 + AC4 * etm4 + AC5 * etm5 + AC6 * etm6 + AC7 * etm7 + AC8 * etm8) - (AD1 * yctm1 + AD2 * yctm2 + AD3 * yctm3 + AD4 * yctm4 + AD5 * yctm5 + AD6 * yctm6 + AD7 * yctm7 + AD8 * yctm8)

Elseif (b0 = 0) Then

utm1 = (1 / TU1) * (-(TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 + TU6 * utm6 + TU7 * utm7 + TU8 * utm8 + TU9 * utm9 + TU10 * utm10) - (g0 * yctm1 + AG1 * yctm2 + AG2 * yctm3 + AG3 * yctm4 + AG4 * yctm5 + AG5 * yctm6 + AG6 * yctm7 + AG7 * yctm8))

If (d = 0) Then

TerminoU = (BD1 * utm1 + BD2 * utm2 + BD3 * utm3 + BD4 * utm4 + BD5 * utm5 + BD6 * utm6 + BD7 * utm7 + BD8 * utm8)

Elseif (d = 1) Then

TerminoU = (BD1 * utm2 + BD2 * utm3 + BD3 * utm4 + BD4 * utm5 + BD5 * utm6 + BD6 * utm7 + BD7 * utm8 + BD8 * utm9)

Elseif (d = 2) Then

TerminoU = (BD1 * utm3 + BD2 * utm4 + BD3 * utm5 + BD4 * utm6 + BD5 * utm7 + BD6 * utm8 + BD7 * utm9 + BD8 * utm10)

End If

yct = TerminoU + Sqr(VarRuido) * (et + AC1 * etm1 + AC2 * etm2 + AC3 * etm3 + AC4 * etm4 + AC5 * etm5 + AC6 * etm6 + AC7 * etm7 + AC8 * etm8) - (AD1 * yctm1 + AD2 * yctm2 + AD3 * yctm3 + AD4 * yctm4 + AD5 * yctm5 + AD6 * yctm6 + AD7 * yctm7 + AD8 * yctm8)

End If

'SEGUIMIENTO

'A = D

Elseif (ModoTrabajo = 211 Or ModoTrabajo = 201) Then

If (b0 < 0 Or b0 > 0) Then

$$ut = (1 / b0) * (-(TU1 * utm1 + TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 + TU6 * utm6 + TU7 * utm7 + TU8 * utm8 + TU9 * utm9 + TU10 * utm10) - (g0 * yctm1 + AG1 * yctm2 + AG2 * yctm3 + AG3 * yctm4 + AG4 * yctm5 + AG5 * yctm6 + AG7 * yctm8) + (1 + (FACTORRB / KP)) * (wtm1 + AC1 * wtm2 + AC2 * wtm3 + AC3 * wtm4 + AC4 * wtm5 + AC5 * wtm6 + AC6 * wtm7 + AC7 * wtm8 + AC8 * wtm9))$$

If (d = 0) Then

$$\text{TerminoU} = (b0 * ut + b1 * utm1 + b2 * utm2 + b3 * utm3 + b4 * utm4)$$

Elseif (d = 1) Then

$$\text{TerminoU} = (b0 * utm1 + b1 * utm2 + b2 * utm3 + b3 * utm4 + b4 * utm5)$$

Elseif (d = 2) Then

$$\text{TerminoU} = (b0 * utm2 + b1 * utm3 + b2 * utm4 + b3 * utm5 + b4 * utm6)$$

End If

$$yct = \text{TerminoU} + \text{Sqr}(\text{VarRuido}) * (et + c1 * etm1 + c2 * etm2 + c3 * etm3 + c4 * etm4) - (a1 * yctm1 + a2 * yctm2 + a3 * yctm3 + a4 * yctm4)$$

Elseif (b0 = 0) Then

$$utm1 = (1 / TU1) * (-(TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 + TU6 * utm6 + TU7 * utm7 + TU8 * utm8 + TU9 * utm9 + TU10 * utm10) - (g0 * yctm1 + AG1 * yctm2 + AG2 * yctm3 + AG3 * yctm4 + AG4 * yctm5 + AG5 * yctm6 + AG7 * yctm8) + (1 + (FACTORRB / KP)) * (wtm1 + AC1 * wtm2 + AC2 * wtm3 + AC3 * wtm4 + AC4 * wtm5 + AC5 * wtm6 + AC6 * wtm7 + AC7 * wtm8 + AC8 * wtm9))$$

If (d = 0) Then

$$\text{TerminoU} = (b1 * utm1 + b2 * utm2 + b3 * utm3 + b4 * utm4)$$

Elseif (d = 1) Then

$$\text{TerminoU} = (b1 * utm2 + b2 * utm3 + b3 * utm4 + b4 * utm5)$$

Elseif (d = 2) Then

$$\text{TerminoU} = (b1 * utm3 + b2 * utm4 + b3 * utm5 + b4 * utm6)$$

End If

$$yct = \text{TerminoU} + \text{Sqr}(\text{VarRuido}) * (et + c1 * etm1 + c2 * etm2 + c3 * etm3 + c4 * etm4) - (a1 * yctm1 + a2 * yctm2 + a3 * yctm3 + a4 * yctm4)$$

End If

'A <> D

Elseif (ModoTrabajo = 210 Or ModoTrabajo = 200) Then

If (b0 < 0 Or b0 > 0) Then

$$ut = (1 / b0) * (-(TU1 * utm1 + TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 + TU6 * utm6 + TU7 * utm7 + TU8 * utm8 + TU9 * utm9 + TU10 * utm10) - (g0 * yctm1 + AG1 * yctm2 + AG2 * yctm3 + AG3 * yctm4 + AG4 * yctm5 + AG5 * yctm6 + AG7 * yctm8) + (1 + (FACTORRB / KP)) * (wtm1 + AC1 * wtm2 + AC2 * wtm3 + AC3 * wtm4 + AC4 * wtm5 + AC5 * wtm6 + AC6 * wtm7 + AC7 * wtm8 + AC8 * wtm9))$$

If (d = 0) Then

$$TerminoU = (b0 * ut + BD1 * utm1 + BD2 * utm2 + BD3 * utm3 + BD4 * utm4 + BD5 * utm5 + BD6 * utm6 + BD7 * utm7 + BD8 * utm8)$$

Elseif (d = 1) Then

$$TerminoU = (b0 * utm1 + BD1 * utm2 + BD2 * utm3 + BD3 * utm4 + BD4 * utm5 + BD5 * utm6 + BD6 * utm7 + BD7 * utm8 + BD8 * utm9)$$

Elseif (d = 2) Then

$$TerminoU = (b0 * utm2 + BD1 * utm3 + BD2 * utm4 + BD3 * utm5 + BD4 * utm6 + BD5 * utm7 + BD6 * utm8 + BD7 * utm9 + BD8 * utm10)$$

End If

$$yct = TerminoU + Sqr(VarRuido) * (et + AC1 * etm1 + AC2 * etm2 + AC3 * etm3 + AC4 * etm4 + AC5 * etm5 + AC6 * etm6 + AC7 * etm7 + AC8 * etm8) - (AD1 * yctm1 + AD2 * yctm2 + AD3 * yctm3 + AD4 * yctm4 + AD5 * yctm5 + AD6 * yctm6 + AD7 * yctm7 + AD8 * yctm8)$$

Elseif (b0 = 0) Then

$$utm1 = (1 / TU1) * (-(TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 + TU6 * utm6 + TU7 * utm7 + TU8 * utm8 + TU9 * utm9 + TU10 * utm10) - (g0 * yctm1 + AG1 * yctm2 + AG2 * yctm3 + AG3 * yctm4 + AG4 * yctm5 + AG5 * yctm6 + AG7 * yctm8) + (1 + (FACTORRB / KP)) * (wtm1 + AC1 * wtm2 + AC2 * wtm3 + AC3 * wtm4 + AC4 * wtm5 + AC5 * wtm6 + AC6 * wtm7 + AC7 * wtm8 + AC8 * wtm9))$$

If (d = 0) Then

$$TerminoU = (BD1 * utm1 + BD2 * utm2 + BD3 * utm3 + BD4 * utm4 + BD5 * utm5 + BD6 * utm6 + BD7 * utm7 + BD8 * utm8)$$

Elseif (d = 1) Then

$$TerminoU = (BD1 * utm2 + BD2 * utm3 + BD3 * utm4 + BD4 * utm5 + BD5 * utm6 + BD6 * utm7 + BD7 * utm8 + BD8 * utm9)$$

Elseif (d = 2) Then

$$TerminoU = (BD1 * utm3 + BD2 * utm4 + BD3 * utm5 + BD4 * utm6 + BD5 * utm7 + BD6 * utm8 + BD7 * utm9 + BD8 * utm10)$$

End If

yct = TerminoU + Sqr(VarRuido) * (et + AC1 * etm1 + AC2 * etm2 + AC3 * etm3 + AC4 * etm4 + AC5 * etm5 + AC6 * etm6 + AC7 * etm7 + AC8 * etm8) - (AD1 * yctm1 + AD2 * yctm2 + AD3 * yctm3 + AD4 * yctm4 + AD5 * yctm5 + AD6 * yctm6 + AD7 * yctm7 + AD8 * yctm8)

End If

End If

End If

'RUTINAS PARA TIEMPO REAL

If (TiempoReal = 1) Then

'REGULACION

'A = D

If (ModoTrabajo = 111 Or ModoTrabajo = 101) Then

If (b0 < 0 Or b0 > 0) Then

ut = (1 / b0) * (-(TU1 * utm1 + TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 + TU6 * utm6) - (g0 * yctm1 + g1 * yctm2 + g2 * yctm3 + g3 * yctm4))

VarControl = (128 * ut / 15) + 127

cortarcontrol

zz = Enviar(&H336, VarControl)

salidaplanta = Medir(&H336, medida)

If d = 0 Then

ypt = 15 * (salidaplanta - 127) / 128

Elsel d = 1 Then

ypt1 = 15 * (salidaplanta - 127) / 128

Elsel d = 2 Then

ypt2 = 15 * (salidaplanta - 127) / 128

End If

Elsel (b0 = 0) Then

utm1 = (1 / TU1) * (-(TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 + TU6 * utm6) - (g0 * yctm1 + g1 * yctm2 + g2 * yctm3 + g3 * yctm4))

VarControl = (128 * utm1 / 15) + 127

```

cortarcontrol
zz = Enviar(&H336, VarControl)

salidaplanta = Medir(&H336, medida)

If d = 0 Then

ypt = 15 * (salidaplanta - 127) / 128

Elseif d = 1 Then

ypt1 = 15 * (salidaplanta - 127) / 128

Elseif d = 2 Then

ypt2 = 15 * (salidaplanta - 127) / 128

End If

End If

'A <> D

Elseif (ModoTrabajo = 110 Or ModoTrabajo = 100) Then

If (b0 < 0 Or b0 > 0) Then

    ut = (1 / b0) * (-(TU1 * utm1 + TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5
+ TU6 * utm6 + TU7 * utm7 + TU8 * utm8 + TU9 * utm9 + TU10 * utm10) - (g0 * yctm1 +
AG1 * yctm2 + AG2 * yctm3 + AG3 * yctm4 + AG4 * yctm5 + AG5 * yctm6 + AG6 * yctm7 +
AG7 * yctm8))

    VarControl = (128 * ut / 15) + 127
    cortarcontrol
    zz = Enviar(&H336, VarControl)

    salidaplanta = Medir(&H336, medida)

    If d = 0 Then

ypt = 15 * (salidaplanta - 127) / 128

    Elseif d = 1 Then

.ypt1 = 15 * (salidaplanta - 127) / 128

    Elseif d = 2 Then

ypt2 = 15 * (salidaplanta - 127) / 128

    End If

Elseif (b0 = 0) Then

    utm1 = (1 / TU1) * (-(TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 + TU6 *
utm6 + TU7 * utm7 + TU8 * utm8 + TU9 * utm9 + TU10 * utm10) - (g0 * yctm1 + AG1 *

```

yctm2 + AG2 * yctm3 + AG3 * yctm4 + AG4 * yctm5 + AG5 * yctm6 + AG6 * yctm7 + AG7 * yctm8))

VarControl = (128 * utm1 / 15) + 127
cortarcontrol
zz = Enviar(&H336, VarControl)

salidaplanta = Medir(&H336, medida)

If d = 0 Then

ypt = 15 * (salidaplanta - 127) / 128

Elseif d = 1 Then

ypt1 = 15 * (salidaplanta - 127) / 128

Elseif d = 2 Then

ypt2 = 15 * (salidaplanta - 127) / 128

End If

End If

'SEGUIMIENTO

'A = D

Elseif (ModoTrabajo = 211 Or ModoTrabajo = 201) Then

If (b0 < 0 Or b0 > 0) Then

ut = (1 / b0) * (-(TU1 * utm1 + TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 + TU6 * utm6 + TU7 * utm7 + TU8 * utm8 + TU9 * utm9 + TU10 * utm10) - (g0 * yctm1 + AG1 * yctm2 + AG2 * yctm3 + AG3 * yctm4 + AG4 * yctm5 + AG5 * yctm6 + AG7 * yctm8) + (1 + (FACTORRB / KP)) * (wtm1 + AC1 * wtm2 + AC2 * wtm3 + AC3 * wtm4 + AC4 * wtm5 + AC5 * wtm6 + AC6 * wtm7 + AC7 * wtm8 + AC8 * wtm9))

VarControl = (128 * ut / 15) + 127
cortarcontrol
zz = Enviar(&H336, VarControl)

salidaplanta = Medir(&H336, medida)

If d = 0 Then

ypt = 15 * (salidaplanta - 127) / 128

Elseif d = 1 Then

ypt1 = 15 * (salidaplanta - 127) / 128

Elseif d = 2 Then

ypt2 = 15 * (salidaplanta - 127) / 128

End If

Elseif (b0 = 0) Then

utm1 = (1 / TU1) * -(TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 + TU6 * utm6 + TU7 * utm7 + TU8 * utm8 + TU9 * utm9 + TU10 * utm10) - (g0 * yctm1 + AG1 * yctm2 + AG2 * yctm3 + AG3 * yctm4 + AG4 * yctm5 + AG5 * yctm6 + AG7 * yctm8) + (1 + (FACTORRB / KP)) * (wtm1 + AC1 * wtm2 + AC2 * wtm3 + AC3 * wtm4 + AC4 * wtm5 + AC5 * wtm6 + AC6 * wtm7 + AC7 * wtm8 + AC8 * wtm9)

VarControl = (128 * ut / 15) + 127
cortarcontrol
zz = Enviar(&H336, VarControl)

salidaplanta = Medir(&H336, medida)

If d = 0 Then

ypt = 15 * (salidaplanta - 127) / 128

Elseif d = 1 Then

ypt1 = 15 * (salidaplanta - 127) / 128

Elseif d = 2 Then

ypt2 = 15 * (salidaplanta - 127) / 128

End If

End If

'A <> D

Elseif (ModoTrabajo = 210 Or ModoTrabajo = 200) Then

If (b0 < 0 Or b0 > 0) Then

ut = (1 / b0) * -(TU1 * utm1 + TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 + TU6 * utm6 + TU7 * utm7 + TU8 * utm8 + TU9 * utm9 + TU10 * utm10) - (g0 * yctm1 + AG1 * yctm2 + AG2 * yctm3 + AG3 * yctm4 + AG4 * yctm5 + AG5 * yctm6 + AG7 * yctm8) + (1 + (FACTORRB / KP)) * (wtm1 + AC1 * wtm2 + AC2 * wtm3 + AC3 * wtm4 + AC4 * wtm5 + AC5 * wtm6 + AC6 * wtm7 + AC7 * wtm8 + AC8 * wtm9)

VarControl = (128 * ut / 15) + 127
cortarcontrol
zz = Enviar(&H336, VarControl)

salidaplanta = Medir(&H336, medida)

If d = 0 Then

ypt = 15 * (salidaplanta - 127) / 128

Elseif d = 1 Then

ypt1 = 15 * (salidaplanta - 127) / 128

Elseif d = 2 Then

```
ypt2 = 15 * (salidaplanta - 127) / 128
```

```
End If
```

```
Elseif (b0 = 0) Then
```

```
    utm1 = (1 / TU1) * (-(TU2 * utm2 + TU3 * utm3 + TU4 * utm4 + TU5 * utm5 + TU6 *  
    utm6 + TU7 * utm7 + TU8 * utm8 + TU9 * utm9 + TU10 * utm10) - (g0 * yctm1 + AG1 *  
    yctm2 + AG2 * yctm3 + AG3 * yctm4 + AG4 * yctm5 + AG5 * yctm6 + AG7 * yctm8) + (1 +  
    (FACTORRB / KP)) * (wtm1 + AC1 * wtm2 + AC2 * wtm3 + AC3 * wtm4 + AC4 * wtm5 +  
    AC5 * wtm6 + AC6 * wtm7 + AC7 * wtm8 + AC8 * wtm9))
```

```
    VarControl = (128 * utm1 / 15) + 127
```

```
    cortarcontrol
```

```
    zz = Enviar(&H336, VarControl)
```

```
    salidaplanta = Medir(&H336, medida)
```

```
    If d = 0 Then
```

```
        ypt = 15 * (salidaplanta - 127) / 128
```

```
    Elseif d = 1 Then
```

```
        ypt1 = 15 * (salidaplanta - 127) / 128
```

```
    Elseif d = 2 Then
```

```
        ypt2 = 15 * (salidaplanta - 127) / 128
```

```
    End If
```

```
End If
```

```
End If
```

```
    yct = ypt + vt
```

```
End If
```

```
Salir:
```

```
Exit Sub
```

```
Error_en_calculocontrol:
```

```
    If (Err = 11) Then
```

```
        MsgBox "División por cero. Revise los datos de los sistemas", 48, "Control de Mínima  
        Varianza"
```

```
    Elseif (Err = 6) Then
```

```
        MsgBox "Overflow", 16, "Control de Mínima Varianza"
```

```
    End If
```

```
Resume Salir
```

```
End Sub
```