

18
201



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA

“ESTUDIO Y DESARROLLO DE UN SISTEMA DE CONTROL AUTOSINTONIZABLE CON IMPLEMENTACION EN LA TARJETA TMS - 32010”

TESIS

Que para obtener el título de
INGENIERO EN COMPUTACION

Presentan:

ARTEAGA PEREZ MARCO ANTONIO

OSORIO GONZALEZ ROBERTO

Director

DR. TANG YU



México, D. F.

Junio de 1991.

TESIS CON FALLA DE ORIGEN



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

INDICE

I. Introducción	1
II. Controladores PID autosintonizables	
2.1. Introducción	7
2.2. Controlador PID	7
2.2.1. Controlador PID continuo	7
2.2.2. Sintonización del PID continuo	8
Método temporal	8
Método frecuencial	9
2.2.3. Discretización del controlador PID continuo	11
2.3. Controlador PID autosintonizable	12
2.3.1. Métodos temporales	13
Autosintonización en base a respuesta escalón en malla abierta	13
Autosintonización en base a respuesta escalón en malla cerrada	15
2.3.2. Método frecuencial	17
III. Implementación del controlador PID autosintonizable en la tarjeta TMS-32010	
3.1. Introducción	23
3.2. Breve descripción de la tarjeta TMS-32010	23
3.3. Descripción de la implementación	33
IV. Control de un motor de corriente directa	
4.1. Introducción	51
4.2. Motor de corriente directa (DC)	51
4.2.1. Modelo matemático	51
4.2.2. Respuesta en frecuencia	51
4.3. Experimentos del controlador PID autosintonizable con el motor de DC	53
4.3.1. Objetivos de los experimentos	53
4.3.2. Resultados de los experimentos	53

Pruebas para la planta sin freno	55
Señal escalón	56
Señal senoidal	58
Señal cuadrada	60
Pruebas para la planta con freno	63
Señal escalón	64
Señal senoidal	66
Señal cuadrada	68
V. Conclusiones	71
Apéndice	
A. Listado de programa ensamblador del controlador autosintonizable	75
Bibliografía	99

I. INTRODUCCION

La mayoría de los procesos industriales se controlan mediante controladores tipo PID (Proporcional-Integral-Derivativo). En el cual la señal de control es proporcional a la magnitud, la integral y la derivada del error, calculado como la diferencia entre una señal de referencia y la salida del proceso [Åström/Wittenmark 1990].

La popularidad de estos controladores se atribuye a la robustez de los mismos en un amplio rango de condiciones de operación y a la sencillez de su funcionamiento.

En el diseño de un controlador PID es necesario determinar tres parámetros: ganancia proporcional (K_p), constante de tiempo integrativa (T_i) y constante de tiempo derivativa (T_d). Al procedimiento que determina estos parámetros se le conoce como sintonización del controlador.

En muchas aplicaciones la sintonización se lleva a cabo por prueba y error, en donde una persona calificada ajusta los parámetros según su experiencia y compara el desempeño del sistema con un patrón preestablecido. Este es un proceso que implica un alto consumo de tiempo y es necesario repetirlo cada vez que hay un deterioro en el desempeño del sistema, causado por cambio de condiciones de trabajo (carga, temperatura, etc.).

Para resolver este problema se han desarrollado métodos de autosintonización. Estos métodos utilizan información sobre el proceso, obtenida mediante métodos de identificación o reconocimiento de patrones y ajustan el controlador automáticamente. Entre los métodos de autosintonización más populares se encuentran el método frecuencial [Åström/Hågglund 1984], en el cual se determina el punto crítico de la curva de Nyquist del proceso a través de una oscilación sostenida, generada por un control de relevador, usándose posteriormente la regla Ziegler-Nichols para la sintonización; y el método temporal [Nishikawa, et al 1984], [Kraus/Myron 1984], en el cual se usa la respuesta escalón para obtener la información del proceso, necesaria para determinar las constantes del controlador.

En esta tesis se estudia el controlador PID autosintonizable y la implementación del método frecuencial utilizando la tarjeta TMS-32010, la cual es una tarjeta de circuitos integrados destinada para el procesamiento digital de señales.

Se utilizó este controlador digital para controlar la velocidad de un motor de corriente directa (DC). Según los resultados obtenidos, este controlador funciona bien, aún cuando haya cambios muy bruscos en las condiciones de trabajo.

El resto de la tesis está organizada de la siguiente manera: en el capítulo II se explican brevemente los métodos de autosintonización, poniendo énfasis en el método frecuencial. El capítulo III trata la implementación del controlador autosintonizable en la tarjeta TMS-32010. En el capítulo IV se presentan los resultados del control del motor de DC. El programa ensamblador está listado en el apéndice A.

II. CONTROLADORES PID AUTOSINTONIZABLES

2.1. INTRODUCCION

Dado un proceso, el diseño del controlador PID consiste en encontrar los parámetros (K_p , T_i , T_d) del mismo. En este capítulo primero se describe brevemente el controlador PID y los métodos de sintonización manuales; posteriormente se muestra la discretización del PID continuo y, por último, se presentan tres algoritmos para automatizar estos métodos, es decir, métodos de autosintonización.

2.2 CONTROLADOR PID

2.2.1. Controlador PID continuo

En un controlador PID continuo, la señal de control está generada de acuerdo con la siguiente ecuación:

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{d e(t)}{dt} \right] \quad (2.1)$$

$$e(t) = r(t) - y(t)$$

donde

- $u(t)$: señal de control
- $y(t)$: salida de la planta
- $r(t)$: señal de referencia
- $e(t)$: señal de error
- K_p : sensibilidad proporcional
- T_d : tiempo derivativo
- T_i : tiempo integrativo

Este controlador está representado esquemáticamente en la fig. 2.1, donde $C(s)$ es la función de transferencia del controlador.

$$C(s) = \frac{U(s)}{E(s)} = K_p \left[1 + T_d s + \frac{1}{T_i s} \right] \quad (2.2)$$

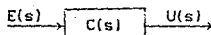


fig. 2.1

2.2.2. Sintonización del PID continuo

Para la sintonización del controlador PID se han desarrollado dos métodos básicos [Ziegler/Nichols 1942]: el método frecuencial (periodo fundamental) y el método temporal (respuesta escalón).

Método temporal

Este método se basa en el hecho de que muchos procesos industriales pueden ser modelados usando una aproximación a un sistema de primer orden con retardo [Åström/Wittenmark 1984].

$$P(s) = \frac{K}{\tau s + 1} e^{-Ls} \quad (2.3)$$

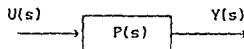
donde

K es la ganancia estática del proceso

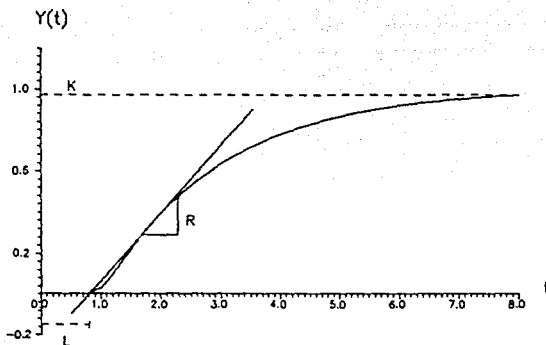
τ es la constante de tiempo

L es el retardo en la entrada

El método consiste en medir la respuesta escalón del sistema en lazo abierto, como se muestra en la figura 2.2.



(a) Sistema en lazo abierto



(b) Respuesta al escalón unitario
fig. 2.2

En la figura 2.2 (b), R es la pendiente de la tangente en el punto de inflexión de la curva de respuesta.

Una vez obtenidas las características que definen al sistema (R y L) se calculan los parámetros del controlador mediante las siguientes expresiones:

$$K_p = \frac{1.2}{RL}; \quad T_i = 2L \quad T_d = 0.5L \quad (2.1)$$

Método frecuencial

Este método usa la información del punto crítico de la curva de Nyquist, definido como el punto en donde la curva de Nyquist cruza por

primera vez el eje real negativo, para sintonizar el controlador PID.

Consideremos el sistema de control mostrado en la figura 2.3, donde K_p es la ganancia proporcional del controlador y $P(s)$ es la función de transferencia de la planta. De acuerdo con el criterio Nyquist, el sistema en malla cerrada es estable si la curva Nyquist no encierra el punto $-1/K_p + 0j$.

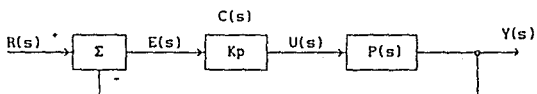


fig. 2.3

Un método experimental para obtener el punto crítico es ir aumentando paulatinamente la ganancia del controlador K_p , hasta que en la salida se obtenga una oscilación sostenida. En este caso la curva Nyquist cruza el punto $-1/K_p + 0j$, como se muestra en la figura 2.4.

A la ganancia del controlador que logra la oscilación sostenida se le llama ganancia crítica (K_c) y a la frecuencia de la oscilación sostenida se le llama frecuencia crítica (ω_c).

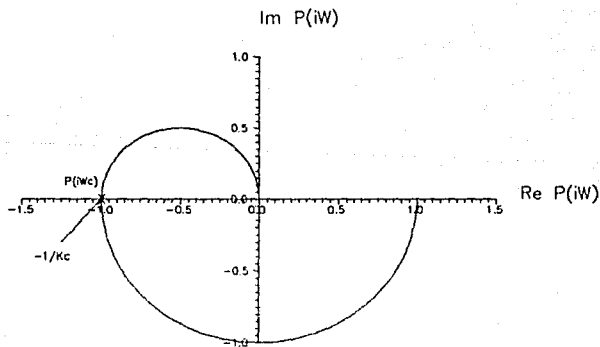


fig. 2.4

Una vez logradas las oscilaciones sostenidas, se mide el periodo de estas (T_c) y la ganancia del controlador (K_c), se ajustan los parámetros del PID en base a:

$$K_p = 0.6K_c; \quad T_i = 0.5T_c; \quad T_d = \frac{T_c}{8} \quad (2.5)$$

2.2.3. Discretización del controlador PID continuo

Teniendo en cuenta el auge actual de las computadoras, resulta conveniente implementar en estas algoritmos de control. Para ello se han desarrollado varios métodos de discretización de sistemas continuos. En el presente trabajo la discretización del controlador PID se realiza utilizando un método de integración numérica (aproximación de la regla rectangular por atraso).

El método seleccionado tiene la ventaja de ser de fácil aplicación, así como de asegurar estabilidad al realizarse el mapeo entre los planos s y z [Franklin/Powell 1980].

A continuación se presenta el desarrollo para obtener la expresión discreta del controlador PID.

Dada la función de transferencia $C(s)$ del PID (2.2)

$$C(s) = K_p \left[1 + \frac{1}{T_i s} + s T_d \right]$$

Se puede obtener el controlador PID discreto usando la regla por atraso

$$C(z) \approx C(s) \quad s = \frac{z - 1}{z T}$$

donde

T es el periodo de muestreo, de lo cual se obtiene

$$C(z) = K_p \left[1 + \frac{Tz}{T_i(z - 1)} + \frac{T_d(z - 1)}{Tz} \right] \quad (2.6)$$

por lo tanto, la acción de control consiste en

$$\text{Parte proporcional} \quad : \quad P(z) = K_p E(z)$$

$$\text{En el tiempo discreto} \quad : \quad p(KT) = K_p e(KT)$$

$$\text{Parte integrativa} \quad : \quad I(z) = \frac{K_p}{T_1} \frac{1}{[z-1/zT]} E(z)$$

$$\text{En el tiempo discreto} \quad : \quad i(KT) = i(KT-T) + \frac{K_p T}{T_1} e(KT)$$

$$\text{Parte derivativa} \quad : \quad D(z) = \frac{z-1}{zT} T_d K_p E(z)$$

$$\text{En el tiempo discreto} \quad : \quad d(KT) = \frac{T_d K_p}{T} [e(KT) - e(KT-T)]$$

Por tanto, la señal de control está dada por:

$$pid(KT) = p(KT) + i(KT) + d(KT) \quad (2.7)$$

2.3. CONTROLADOR PID AUTOSINTONIZABLE

El problema que presentan los métodos de sintonización antes expuestos, es que, si por alguna razón, las características de la planta cambian con el tiempo (lo que muy bien puede ocurrir, por ejemplo, por el deterioro de algún elemento de la planta) el controlador no cumplirá su misión con la eficiencia de un principio, lo que hace necesario volver a calcular los parámetros del PID (K_p , T_1 y T_d). Los métodos de autosintonización, como su nombre lo indica, sintonizan el controlador automáticamente.

2.3.1. Métodos temporales

Autosintonización en base a respuesta escalón en malla abierta

Un método de autosintonización es medir directamente los parámetros K , τ , L en la planta (2.3) a partir de la respuesta a escalón (figura 2.2 [b]), y usar las expresiones (2.4) para sintonizar el controlador PID. Sin embargo, este método sufre la desventaja de ser muy sensible a las mediciones, ya que el ruido y el error de medición que presentan a la salida causan desviaciones en los valores antes mencionados.

Otro método de autosintonización basado en el método temporal es el que usa las áreas S_0 y S_1 (figura 2.5) para la sintonización [Nishihawa, et al 1984]. Este método es menos sensible al ruido y al error de medición, ya que la integral representa un promedio de las mediciones.

Para una planta que puede ser aproximada con el modelo de primer orden con retardo, este método consiste en:

1) Aplicación de un pulso de ancho T_p y amplitud A al proceso en malla abierta;

2) Muestreo de la respuesta del proceso $\{ z(t) \}$ al pulso durante el tiempo T_p y obtención de la respuesta escalón de la planta $\{ y(t) \}$ en base a:

$$y(t) = \begin{cases} z(t) & ; 0 \leq t \leq T_p \\ z(t) + y(t-T_p) & ; t > T_p \end{cases}$$

3) Estimación de los valores característicos del proceso, haciendo uso de $y(t)$;

4) Cálculo de los parámetros del controlador empleando las reglas de sintonización de Ziegler-Nichols para el método temporal.

Considerando la aplicación del método para una planta representada por:

$$G(s) = \frac{K e^{-sL}}{\tau s + 1}$$

La regla de sintonización para el método temporal de Ziegler-Nichols es:

$$K_p = \frac{1.2}{a} \quad ; \quad T_i = 2 L \quad ; \quad T_d = \frac{L}{2} \quad (2.8)$$

donde

$$a = K \frac{L}{\tau}$$

L = tiempo de retardo del sistema

τ = constante de tiempo del sistema

por lo que podemos notar que el problema de sintonización radica básicamente en obtener los valores de K , L y τ . Para automatizar esto, en lugar de obtenerlos en forma gráfica, el método se basa en medir las áreas dadas por la curva de respuesta escalón del sistema, obteniéndose así L y τ . La ganancia del sistema está dada por la razón entre la entrada y la salida del proceso en estado estable

$$K = \frac{B}{A} \quad (2.9)$$

donde

B = valor final de la respuesta escalón

A = amplitud del pulso introducido al sistema.

Dada la respuesta escalón del sistema mostrada en la figura 2.5

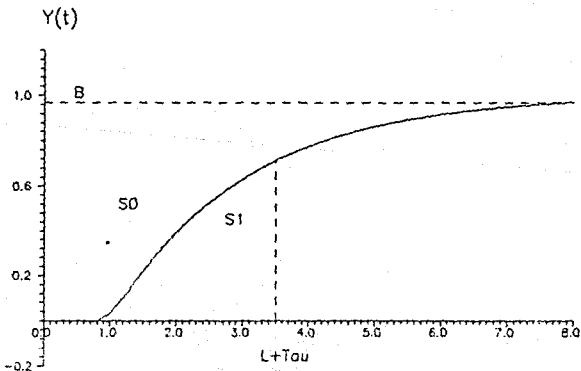


Fig. 2.5

Se calculan los parámetros en base a:

$$\begin{aligned}\tau + L &= \frac{S_0}{K} \\ \tau &= \frac{e S_1}{K}\end{aligned}\tag{2.10}$$

donde

e = base del logaritmo natural

La amplitud A y el ancho T_p del pulso de prueba se determinan de modo que se tenga una respuesta apropiada para obtener los parámetros del proceso.

Este método es extensible a procesos de la forma

$$G_p(s) = \frac{K_p e^{-Ls}}{n \prod_{j=1} (1 + \tau_j s)}$$

Autosintonización en base a respuesta escalón en malla cerrada

Controlador autosintonizable EXACT. El "EXACT Controller" (Expert Adaptive Controller Tuning) fué desarrollado por la compañía Foxboro [T.W. Kraus/T.J. Myron 1984].

Este es un método de autosintonización que consiste en observar el error en un sistema de lazo cerrado cuando se aplica a la planta una perturbación.

Este método comienza asignándole a K_p , T_i y T_d valores aleatorios o valores que se consideren adecuados para conseguir un buen control de la planta. Puesto que el método se basa en la convergencia de K_p , T_i y T_d a valores con los que se logre un buen desempeño del sistema, si los valores iniciales no son correctos, estos se modifican durante el proceso de autosintonización.

Para realizar la autosintonización, primero se aplica una perturbación al sistema, consistente en un pulso. Posteriormente se observa si el error sobrepasa un límite preestablecido (se debe

considerar el ruido que pueda existir); si es el caso, entonces se miden 3 picos (E_1 , E_2 y E_3 [figura 2.6]) calculándose el valor del periodo entre picos, así como del sobrepaso y del amortiguamiento. Estos últimos valores se calculan con las siguientes relaciones:

$$\text{Sobrepaso} = \frac{E_2}{E_1} \quad \text{Amortiguamiento} = - \frac{E_3 - E_2}{E_1 - E_2} \quad (2.11)$$

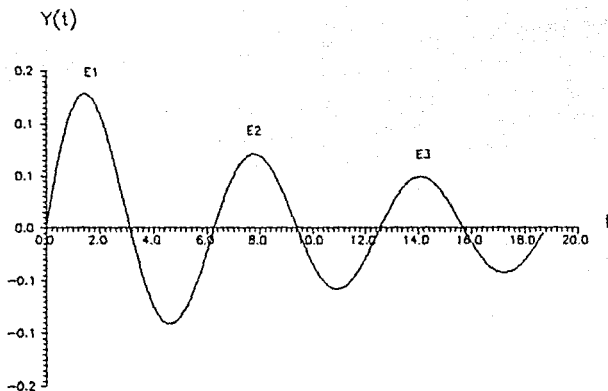


fig. 2.5

Ya que se tienen estos dos valores se comprueba que no sean mayores que otros definidos previamente para utilizarlos en el criterio de autosintonización. Si son más grandes de lo deseado se procede al cálculo de nuevos parámetros.

Ziegler y Nichols proponen la siguiente relación para ajustar los parámetros T_i y T_d

$$\frac{T_i}{P_e} \rightarrow 0.5 \quad \frac{T_d}{P_e} \rightarrow 0.12 \quad (2.12)$$

donde

P_e = período entre picos

Para ajustar la ganancia del controlador se puede considerar que un tiempo de respuesta grande merece una K_p menor.

Si al cabo de un tiempo determinado no se producen los picos E_2 y E_3 , entonces se asignan valores aleatorios a estos y se continúa el proceso normal.

2.3.2. Método frecuencial

Este es el método implementado en el presente trabajo. Se basa en la regla de Ziegler-Nichols para el ajuste de parámetros de un controlador tipo PID, derivado a su vez de la observación de que estos parámetros pueden ser determinados a partir del conocimiento de un punto en la curva de Nyquist de un sistema en lazo abierto. Este punto es la intersección de dicha curva con el eje real negativo, descrito normalmente en términos de la ganancia crítica K_c y del período crítico T_c [Áström/Wittenmark 1984].

La teoría necesaria para comprender el funcionamiento del controlador PID autosintonizable básicamente comprende el criterio de estabilidad de Nyquist, funciones descriptivas, ciclos límites, margen de ganancia (complementariamente margen de fase) y discretización de sistemas en tiempo continuo. El lector no familiarizado con estos criterios puede encontrar alguna información en la bibliografía que se proporciona al final del presente trabajo [Ogata 1986].

Funcionamiento del controlador autosintonizable. Experimentalmente se ha visto que un sistema con un retraso de fase de al menos π radianes puede oscilar con un período T_c bajo un control de relevador. Para determinar la ganancia crítica y el período crítico se implementa una realimentación del sistema con el control de relevador en cascada, como se muestra a continuación:

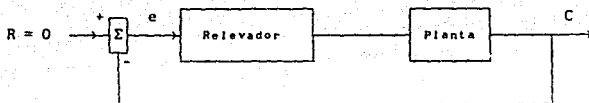


fig. 2.7

El error e es entonces una señal periódica con periodo T_c . Si d es la amplitud del relevador, se sigue de la expansión de una serie de Fourier que la primera armónica de la salida del relevador tiene la amplitud $4d/\pi$. Considerando la salida del proceso con un valor igual a a , entonces, la ganancia crítica se calcula como

$$k_c = \frac{4d}{\pi a} \quad (2.13)$$

resultado que se obtiene de un aproximación por medio de la función descriptiva $\{N(a)\}$. Esta función, para un relevador ideal es

$$N(a) = \frac{4d}{\pi a} \quad (2.14)$$

Puede presentar ventajas el utilizar otras no-linealidades además del relevador sencillo. Por ejemplo, un relevador con histéresis tiene la ventaja de ser menos sensible al ruido.

De este modo, un control por relevador da la información necesaria acerca del proceso para aplicar los criterios de diseño. Además, este método tiene la ventaja de que permite manipular la amplitud del ciclo límite con una amplitud apropiada del relevador. Asimismo, se hace notar que con esta implementación se generará automáticamente una señal de entrada al proceso, la cual tiene una frecuencia de $\omega_c = 2\pi/T_c$; lo que asegura que el punto crítico sobre la curva de Nyquist puede ser determinado adecuadamente.

Una vez conocido el punto crítico, es posible aplicar la regla de diseño de ajuste de parámetros de Ziegler-Nichols; esto no impide, por supuesto, utilizar cualquier otra regla basada en el conocimiento de este punto crítico.

Cálculo de parámetros del controlador PID. Una vez que se ha logrado averiguar el punto crítico, es relativamente fácil crear un controlador con el margen de ganancia deseado. Un criterio simple para calcular la ganancia proporcional es

$$K_p = \frac{k_c}{A_m} \quad (2.15)$$

donde A_m es el margen de ganancia deseado y k_c es la ganancia crítica.

Esto se entiende si recordamos que para la ganancia crítica (k_c) y la frecuencia crítica (ω_c) el valor de $|k_c G(j\omega_c)| = 1$, por lo mismo, si se tiene $k = k_c/A_m$, en la misma condición tendríamos $|kG(j\omega_c)| = 1/A_m$. Como el margen de ganancia se define como el inverso de este resultado, implica que su valor es, precisamente, A_m (figura 2.8).

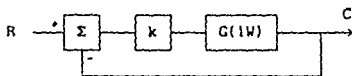


Fig. 2.8

Si por otro lado se toma el modelo en frecuencia de un controlador tipo PID como

$$C(s) = K_p \left[1 + \frac{1}{sT_i} (1 - sT_d) \right] \quad (2.16)$$

ya que K_p ha sido fijada como k_c/A_m , teniendo de esta manera el valor deseado del margen de ganancia, entonces es necesario que, cuando se tenga período crítico y ganancia crítica, el segundo término de la relación valga cero, lo que se puede lograr con la siguiente regla

$$T_d = \frac{1}{\omega_c^2 T_i} \quad (2.17)$$

donde

$$\omega_c = 2\pi/T_c$$

Esta relación permite mantener el margen de ganancia deseado. El tiempo de integración puede ser elegido arbitrariamente.

La regla de Ziegler-Nichols dice que

$$K_p = \frac{k_c}{2}; \quad T_i = \frac{T_c}{2}; \quad T_d = \frac{T_c}{8} \quad (2.18)$$

Es posible combinar ambos criterios utilizando $A_m = 2$ (lo que equivale a un poco más de 6 db) y $T_i = T_c/2$, obteniendo el valor de T_d por la relación antes mostrada (2.17).

III. IMPLEMENTACION DEL CONTROLADOR PID AUTOSINTONIZABLE

EN LA TARJETA TMS-32010

3.1. INTRODUCCION

En este capítulo se considera el problema de implementación del controlador PID autosintonizable, usando el método frecuencial, en una tarjeta TMS-32010. Para ello, primero se describe, en forma breve, la tarjeta TMS-32010; posteriormente se presenta una descripción detallada de la implementación.

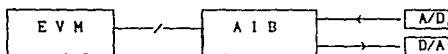
3.2. BREVE DESCRIPCION DE LA TARJETA TMS-32010

La tarjeta TMS-32010 es una tarjeta de procesamiento digital de señales diseñada para soportar una amplia gama de aplicaciones que requieren procesamiento numérico a alta velocidad. Dentro de estas aplicaciones podemos encontrar al control digital.

En esta sección se describirá brevemente la arquitectura básica de la tarjeta, así como algunas de sus características de programación.

3.2.1. ARQUITECTURA BASICA.

La tarjeta TMS-32010 consta de dos módulos de operación : módulo de evaluación (EVM), y módulo de interfaz analógica (AIB).



(Fig. 3.1

A. Módulo de Evaluación (EVM)

En este módulo se realiza el procesamiento de la información, para ello está dividido en dos modos de operación, el primero consiste en modo sistema operativo, el cual permite la comunicación entre el usuario y la tarjeta, mientras que el segundo es el modo de evaluación de programa.

Modo Sistema Operativo

En este apartado se expone solamente la comunicación usuario-tarjeta que se ha utilizado para el desarrollo de la tesis, la cual está dada por :

i) *Preparación de la tarjeta para recibir y ensamblar un programa fuente que se le transmita usando el puerto RS-232 de una PC.* Para éste fin se cuenta con una terminal para interactuar con el sistema operativo de la tarjeta, habilitando a esta con la instrucción

?ASM 2 1

Una vez lista la tarjeta para recepción se transmite el programa fuente desde la PC usando un programa de transmisión de información ASCII por el RS-232 (en este caso el programa de transmisión empleado es el TMSS desarrollado en la DEFFI-UNAM, en lenguaje C). El programa a transmitir debe iniciar con el símbolo ">" y terminar con "<", quedando entre estos símbolos el programa fuente. La edición de este programa puede realizarse con cualquier editor de texto ASCII. Ya transmitido el programa fuente, la PC queda "desconectada" de la tarjeta y puede ser empleada en forma normal.

ii) *Indicación a la tarjeta del modo de reloj que se va a emplear.* En el caso del presente trabajo se emplea el modo EXTERNAL CLOCK, con el objeto de sincronizar a la tarjeta EVM con la tarjeta AIB. Para dicha operación se emplea la instrucción

?INIT EXTERNAL.

iii) *Ejecución del programa.* Esto se puede realizar en una forma continua, ignorando puntos de ruptura (BREAK POINTS) usando la instrucción

?RUN

Igualmente se puede correr en forma continua pero considerando los puntos de ruptura; comenzando la ejecución del programa en la instrucción indicada por el PC (Program Counter) y usando la instrucción

?EX

O bien, el programa puede ser ejecutado paso a paso con la instrucción

?SS

iv) *Ruptura de la ejecución del programa.* Para esto, la tarjeta cuenta con un botón de RESET, el cual al ser activado detiene la ejecución del programa salvando totalmente el contexto de la corrida.

v) *Desplegado de la información.* Para poder desplegar información, ya sea de la memoria de programa, o bien de la memoria de datos, la tarjeta debe estar en modo de sistema operativo, es decir, sin ejecutar programa.

La información que se puede desplegar es :

a) Contenido de la memoria de programa

?PRASM D.I. D.F.

la cual será desplegada en código hexadecimal y su respectivo mnemónico, para cada instrucción; a partir de la dirección de inicio (D.I.) hasta la dirección final (D.F.).

Las instrucciones pueden ser alteradas usando

?FFPM D.I. D.F. C.H.

donde C.H. es el correspondiente código hexadecimal de la nueva instrucción.

Debido a que ya no hay interacción entre la PC y la tarjeta, las nuevas instrucciones (si se dió el caso) deben ser, posteriormente, cambiadas en el programa fuente.

b) Contenido de la memoria de datos.

Esta información puede ser desplegada usando la instrucción

?DDM D.I. D.F.

y también puede ser alterada, usando la instrucción

?FDM D.I. D.F. V.H.

en donde V.H. es el valor hexadecimal del nuevo dato.

La forma en que se conecta el equipo mencionado es :

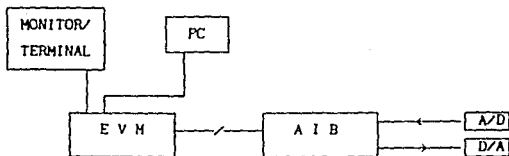


Fig. 3.2

Modo de Evaluación de Programa

En este modo el EVM funciona como una unidad de procesamiento de información digital que cuenta con las siguientes características fundamentales :

- Ciclo de instrucción de 200-ns
- Memoria RAM de datos (on-chip) de 288 bytes (144 x 16 word)
- Palabra de datos/instrucción de 16 bits
- ALU/acumulador de 32 bits
- Multiplicador de 16 x 16 bits en 200-ns
- Registro de corrimiento (barrel shifter) de 0 a 15 bits, para realizar corrimientos de la palabra de la memoria de datos en la ALU
- Registro de corrimiento para realizar corrimientos del acumulador en la memoria de datos
- Ocho canales de entrada y ocho canales de salida.
- Bus de datos bidireccional de 16 bits con velocidad de transferencia de 40-megabits por segundo
- Interrupción con salvado completo del contexto
- Aritmética de punto fijo, signada y en complemento a dos
- Stack de 4 x 12 bits
- PC (Program Counter) de 12 bits
- Registros de autoincremento/decremento para direccionamiento indirecto de datos y contador de ciclos. Permitiendo además realizar ciclos sencillos de manipulación de tablas de datos
- Fuente de alimentación de +5, +12 y -12 volts

Arquitectura

La tarjeta TMS-32010 utiliza una arquitectura HARDVARD modificada, la cual consiste en tener separadas las memorias de programa y de datos, pero permitiendo la transferencia entre los espacios de programa y datos. Esta característica permite almacenar constantes en la memoria de programa para ser leídas en la RAM, eliminando la necesidad de una ROM para las constantes. A continuación se describen los elementos de la tarjeta

Memoria de programa. - Consiste de 4K palabras de 16 bits.

Memoria de datos. - Consiste de una RAM de 144 palabras de 16 bits, dividida en dos páginas, la página cero va de la dirección 0 a la 127 y la página 1 de la dirección 128 a la 143. Todos los operandos de datos no-inmediatos residen en esta RAM.

Para lograr que los datos residan en la RAM, se cuenta con las instrucciones TBLR (TaBLE Read) para transferir valores de la memoria de programa en la memoria de datos, TBLW (TaBLE Write) para transferir datos de la memoria de datos a la de programa.

Esta memoria puede ser direccionada en forma inmediata, directa e indirecta.

Registros. - Se cuenta con los siguientes registros:

Registros auxiliares. - Existen dos registros auxiliares (ARO y ARI) de 16 bits que no son parte de la RAM de datos. Estos registros pueden ser usados para tres funciones: almacenamiento temporal, direccionamiento indirecto de la RAM de datos y control de ciclos.

Apuntador de registro auxiliar. - El apuntador de registro auxiliar (ARP) es un solo bit, el cual es parte del registro de status. Este indica que registro auxiliar se está empleando actualmente.

Apuntador de página de memoria de datos. - Este es un bit que pertenece al registro de estatus e indica la página de la memoria de datos que se está empleando. Este bit es concatenado, en modo de direccionamiento directo, con siete bits de la palabra de instrucción para formar la dirección del dato a utilizar, de la memoria de datos.

Funcionamiento de Entrada/Salida. - La entrada o salida de datos de un y hacia un periférico está acompañada por las instrucciones IN u OUT, respectivamente.

Elementos Aritméticos.- La tarjeta cuenta con cuatro elementos aritméticos : la ALU, el acumulador, el multiplicador y los corrimientos. Todas las operaciones son realizadas usando aritmética de complemento a dos.

ALU.- Es una unidad aritmética-lógica de propósito general que opera con palabras de datos de 32 bits. La unidad suma, resta y realiza operaciones lógicas. El acumulador es siempre el destino, así como el operando primario.

Acumulador.- Almacena la salida de la ALU y frecuentemente es usado como entrada a la misma. Este opera con una palabra de 32 bits. El acumulador está dividido en dos palabras: palabra alta (bits 31 a 16) y palabra baja (bits 15 a 0). Para almacenar las palabras alta y baja del acumulador en la memoria de datos se cuenta con las instrucciones SACH y SACL.

Dentro de las condiciones del acumulador que pueden ser utilizadas para realizar brinco se tiene :

INSTRUCCION	CONDICION DESEADA EN EL ACUMULADOR
BLZ	< 0
BLEZ	<= 0
BGZ	> 0
BGEZ	>= 0
BNZ	<> 0
BZ	= 0

Multiplicador.- El multiplicador consiste de tres unidades : el registro T, el registro P y el arreglo de multiplicación. El registro T es un registro de 16 bits que almacena el multiplicando, mientras que el registro P es un registro de 32 bits que almacena el

producto. El multiplicador se almacena en una localidad de memoria de datos.

Después de realizar la multiplicación (MPY o MPYK) se debe utilizar una instrucción que combine al registro P con el acumulador (PAC, APAC, SPAC, LTA, o LTD).

Corrimientos.- Se cuenta con dos tipos de corrimientos para la manipulación de datos : un barrel shifter para correr los datos de la memoria de datos dentro de la ALU y parallel shifter para correr el acumulador dentro de la memoria de datos.

Pin Bio.- Este pin es un pin externo que soporta operaciones de brinco y bit-test. Cuando se presenta una señal baja en este pin, la ejecución de la instrucción BIOZ causará la realización de un brinco a una dirección indicada. Este pin es muestreado cada ciclo de reloj.

B. Módulo de Interfaz Analógica (AIB)

A través de este módulo se realiza la conversión analógica-digital (A/D) y digital-analógica (D/A), además de tener interfaz con el procesador TMS-32010 (EVM). Este módulo cuenta con las siguientes características básicas :

- Convertidor analógico-digital de 12 bits con muestreador y retén
- Convertidor digital-analógico de 12 bits
- Puerto de salida de 16 bits para un D/A adicional o bien aplicaciones definidas por el usuario
- Puerto de entrada de 16 bits para un A/D adicional o bien aplicaciones definidas por el usuario
- Dos filtros paso bajas
- Amplificador de audio
- Decodificador de TBLW

- Entrada/salida para extensión de memoria de datos
- Area de prototipos para aplicaciones del usuario
- Convertidor analógico-digital
 - Resolución : 12 bits
 - Entrada analógica : -10 V a +10 V
 - Salida digital : 16 bits en complemento a dos
 - Tiempo de conversión : 25 microsegundos
- Convertidor digital-analógico
 - Resolución : 12 bits
 - Salida analógica : -10 V a +10 V
 - Entrada digital : 16 bits en complemento a dos
 - Tiempo de conversión : 250 nanosegundos
- Muestreo y retén
 - Tiempo de adquisición al 0.1% : 4 microsegundos
 - Promedio de pérdida en la salida : 0.3 V/sec
 - Paso de retén : 10 mV
- Reloj para frecuencia de muestreo
 - Rango : 76.29 Hz a 5.00 MHz
- Memoria de datos extendida
 - Capacidad en la tarjeta : 8192 x 16 bits

* El convertidor de 12 bits coloca su información, a través de un latch, en los 12 bits más significativos de los 16 bits del bus de datos (compatible con el formato Q15).

Para poder hacer uso del AIB es preciso indicar el modo en que se va a operar, así como la frecuencia de muestreo (si es el caso) que se va a emplear. El modo de operación se controla mediante el registro de control. Para nuestro caso el modo de control para adquisición de datos es un modo automático de recepción de datos, con un modo transparente para la conversión D/A.

La frecuencia de muestreo (Fsr) está dada por:

$$F_{sr} = \frac{F_{clkout}}{N + 1} \quad \text{o} \quad N = \frac{F_{clkout}}{F_{sr}} - 1$$

en donde:

Fclkout = 5 MHz

N = constante cargada en el registro de reloj, a través del puerto 1 con la instrucción OUT

Fsr = frecuencia de muestreo deseada por el usuario

Se da como rango seguro para la constante N de 144 a 65535 (34.48 KHz a 76.29 Hz). La salida del reloj para la velocidad de muestreo es periódica para todos los valores de N.

3.2.2. ADMINISTRACION DE LOS RECURSOS DEL PROCESADOR

En esta sección se dará, en forma breve, una explicación de cómo usar algunas de las características de la TMS-32010, con objeto de hacer un uso eficiente de la misma.

Aritmética de punto fijo

Considerando que la realización de cálculos en la tarjeta está basado en una representación numérica en complemento a dos de punto fijo, usando 16 bits, podemos decir que cada número de 16 bits está dado por 1 bit de signo, 1 bits para la parte entera y 15-1 bits para la parte fraccionaria, así, el número

00000010 10100000
└──────────┘ punto decimal

tiene el valor de 2.625. Este número se dice que está representado en formato Q8 (8 bits para la parte fraccionaria). Su rango de valores está

entre -128 (1000000000000000) y 127.996 (011111111111111), en donde la precisión de un número en Q_8 es aproximadamente 0.004.

La multiplicación, en formato Q_{15} (formato usado en el presente trabajo) se maneja de la siguiente manera :

$$Q_{15} * Q_{15} = Q_{30}$$

Al realizarse esta operación, en el resultado vamos a tener 32 bits, en donde los 2 MSB son de signo, por lo que si deseamos un número en Q_{30} con la misma precisión que los factores (que quede en Q_{15}), se tiene que realizar un corrimiento a la izquierda. Tomando la parte alta de los 32 bits se tiene el número en Q_{15} .

En general, el producto de un número de i bits enteros y f bits fraccionales por un segundo número con j bits enteros y g bits fraccionales dará como resultado un número con $(i+j)$ bits enteros y $(f+g)$ bits fraccionales. La mayor precisión posible para representar a este número con 16 bits tendrá $(i+j)$ bits enteros y $(15-i-j)$ bits fraccionales.

Para el caso de la suma es necesario que los dos números a intervenir en la operación tengan el mismo formato.

Subrutinas

Dado que el procesador cuenta con un stack, es posible implementar subrutinas. Sin embargo, hay que recordar que este stack es de 4 niveles. Para ampliar estos niveles mediante software se puede hacer uso de las instrucciones PUSH y POP, así como de la memoria de datos.

Control de ciclos

Para este fin, se utiliza la instrucción BANZ, la cual pregunta por cero y posteriormente decrementa el registro auxiliar seleccionado por el ARP.

En ambos casos, el máximo número que se puede emplear en un registro auxiliar para usarlo como contador o apuntador de direcciones contiguas es 512, debido a que sólo los 9 LSB de cada registro están implementados como contadores.

SOFTWARE ADICIONAL CON QUE CUENTA LA TARJETA TMS-32010

- *Ensamblador XASM10*. Genera listado de código de instrucción y programa en código objeto
- *Ligador LINK10*
- *Simulador SIM10*. Usa el programa en código objeto generado por el ensamblador.

3.3. DESCRIPCION DE LA IMPLEMENTACION

El controlador PID autosintonizable se puede expresar esquemáticamente como :

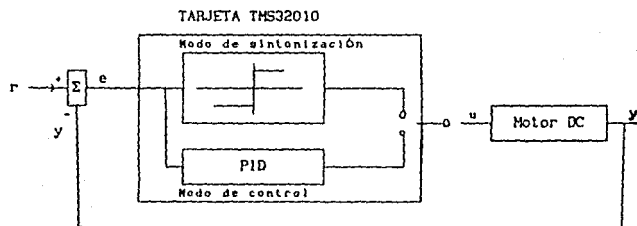


Fig. 3.3

En el modo de sintonización el programa genera oscilaciones en la planta debidas a la acción del relevador. Estas oscilaciones son medidas de la siguiente manera: para averiguar el periodo de la señal se cuentan tres cruces por cero, el primero como referencia para empezar a medir el tiempo que corresponde al periodo crítico y los otros dos precisamente para saber cuando es que se ha completado un periodo de la señal.

Cuando ha sido obtenido el periodo de la señal, se procede a calcular la amplitud de la onda de salida, lo que se lleva a efecto de la siguiente manera: se miden los valores extremos de la amplitud de salida (el más alto y el más pequeño), después se hace la diferencia entre estos dos valores, siendo el resultado el valor pico a pico de la señal; una

vez que se tiene este valor, es dividido entre dos para así obtener la amplitud del ciclo límite.

El período crítico se obtiene, como bien se ve, de manera directa; para obtener la ganancia crítica se utiliza la fórmula que se obtiene a partir de la función descriptiva:

$$N(a) = \frac{4d}{\pi a}$$

donde

a = amplitud de la señal de oscilación

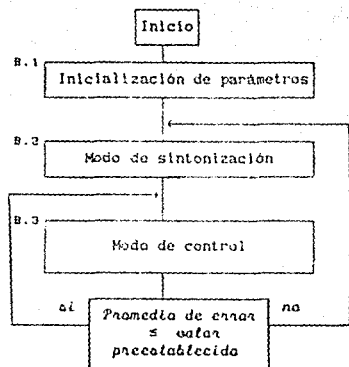
d = amplitud del relevador

En el modo de control el programa calcula la señal de control con las constantes K_p , T_i y T_d recién ajustadas.

El programa con que se implementa el controlador autosintonizable se muestra en el apéndice A del presente trabajo. A continuación se presenta, en forma de diagramas de bloques, la composición general del programa, explicando brevemente la función de cada una de las rutinas del mismo; se hace notar que los diagramas de flujo se crearon con la finalidad de facilitar al lector el entendimiento del programa, por lo mismo, no siempre corresponderá el flujo de estos al que tiene el programa en su implementación, aunque las variaciones son mínimas.

En los diagramas de flujo se utiliza la letra tipo *script* para indicar condiciones (en vez de rombos, que es la forma convencional).

El siguiente diagrama de flujo representa la manera en que funciona el programa; a cada bloque se le asigna un nombre que da idea de la función que realiza; en caso de ser necesario se da una explicación más detallada de las rutinas que componen un bloque específico y, además, se le asigna una clave para su mejor ubicación en el momento de ser explicado.



B.1 INICIALIZACION DE PARAMETROS

En este bloque se procede a inicializar todas las constantes que se utilizan en el programa.

La constante MODE es un parámetro que indica a la tarjeta su modo de funcionamiento.

RATE y RATE1 indican velocidad de muestreo. RATE1 se utiliza al momento de controlar y es menor a RATE, que se utiliza en el cálculo de parámetros (experimentalmente se observó la conveniencia de estos dos tiempos de muestreo).

RATE1 debe ser menor que RATE debido a que un valor grande de frecuencia produce pérdida de precisión en el cálculo de la salida de control, ya que implicaría un valor pequeño de tiempo, con el cual no se pueden realizar las operaciones necesarias para el cálculo del valor de salida de la señal de control.

Las constantes PI, PICINV, UNO, MUNO, CUATRO, CINCO, MAXERR, FORMAT, PUNTOS, D, MENOSD, SAMPLE, FRONTN, FRONTP, HIST, LIMP y LIMN son constantes que se utilizan en la lógica interna del programa. Si es

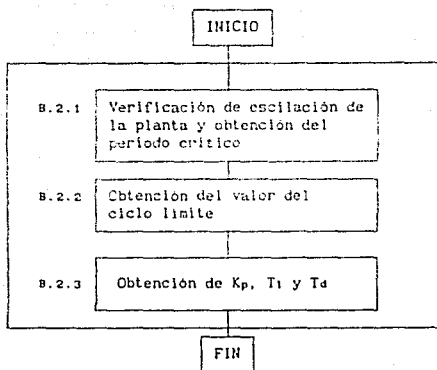
oportuno, se mencionará en las rutinas (en los diagramas de flujo) el uso que se les da.

A continuación se dará la relación de los valores asociados a estas constantes. Algunas tienen la especificación de que se encuentran en el formato Qis.

MODE	←	>FA	
RATE	←	499	(10khz)
RATE1	←	>7FFF	(152hz)
INVPI	←	1/π	Qis
PICINV	←	1/π ²	Qis
UNO	←	1	
MUNO	←	-1	
PVENTS	←	0.25	Qis
CINCO	←	0.05	Qis
MAXERR	←	3	
FORMAT	←	15	
PUNTOS	←	0.5	Qis
D	←	0.14	(1.4v)
MENOSD	←	-0.14	(-1.4v)
SAMPLE	←	0.0001	Qis
FRONTN	←	>A800	(7v)
FRONTP	←	>5500	(-7v)
HIST	←	0.07	(0.7v)
LIMP	←	>C000	
LIMN	←	>4000	

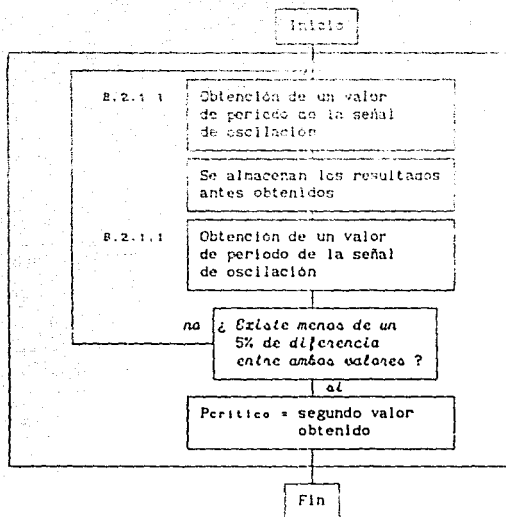
Los valores de la histéresis (HIST) y del valor de salida del relevador (D y MENOSD), así como las velocidades de muestreo RATE y RATE1 han sido probados con éxito con el motor de DC; no se excluye la posibilidad de ser modificados en alguna otra planta.

B.2 MODO DE SINTONIZACION



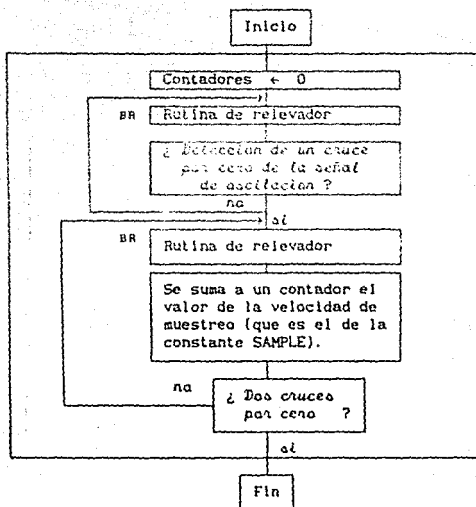
En esta parte del programa se carga el valor de la frecuencia de muestreo con RATE.

E.2.1 VERIFICACION DE OSCILACION DE PLANTA Y OBTENCION
DEL PERIODO CRITICO



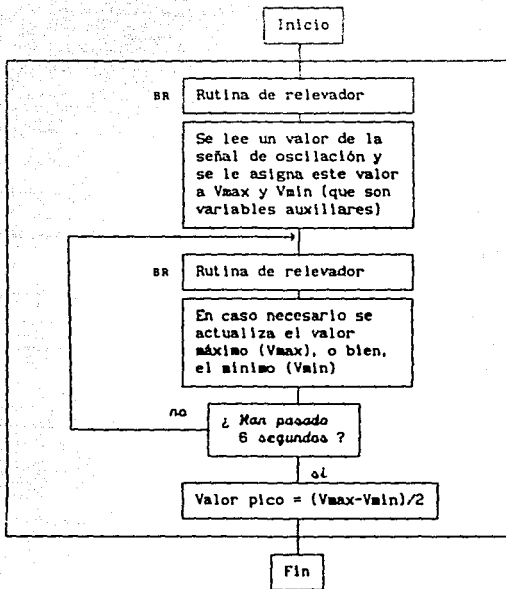
Periodo (periodo critico) se representa en el programa con la variable llamada PCRIT.

B.2.1.1 OBTENCIÓN DE UN VALOR DE PERIODO DE LA SEÑAL DE OSCILACIÓN



En esta rutina el contador que va a tener el valor del periodo de la señal muestreada permanece sin cambios hasta que se detecte un cruce por cero, momento en el cual se empieza a medir el tiempo que tarda en producirse un periodo completo de la señal.

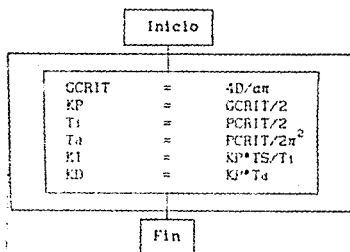
B.2.2 OBTENCION DEL VALOR DEL CICLO LIMITE



En este caso, la variable Vmax no se llama así en el programa, sino se llama ANPID, mientras que Vmin es TEMP y el valor pico es CCRIT (esto es por economía de memoria de datos). También se hace notar que, para ser congruentes con la lógica de la tarjeta, que invierte el signo del valor de entrada y de salida en la conversión A/D y D/A, en realidad se le resta a Vmin, Vmax.

El tiempo durante el cual se muestrean datos es de 6 segundos porque experimentalmente se observó que, con este lapso de tiempo, no se pierden datos para la obtención del valor máximo o del valor mínimo.

B.2.3 OBTENCION DE KP, Td Y Ti



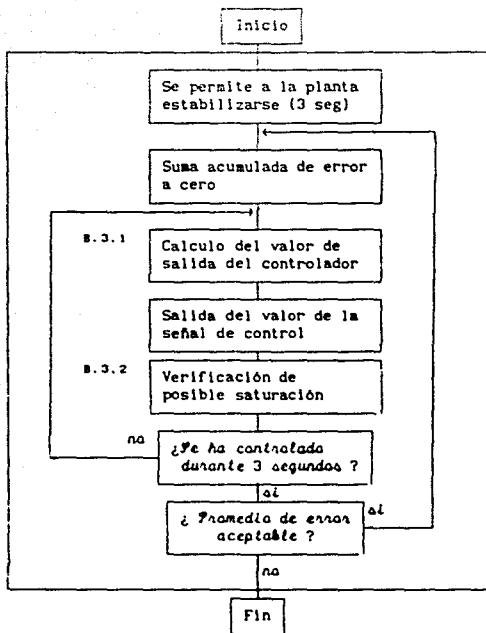
En este bloque se obtienen los valores de los parámetros del controlador. Se hace notar que la programación del cálculo de estos valores es bastante más complicada de lo que se muestra.

Por otro lado, por conveniencia de programación, en la fórmula del controlador no se utilizan T_i y T_d , sino las variables auxiliares K_I y K_D .

Los nombres de las variables que se mencionan en el diagrama de flujo significan lo siguiente:

GCRIT	Ganancia CRITica
PCRIT	Periodo CRITico
KP	Constante Proporcional
KI	Constante Integrativa
KD	Constante Derivativa
TI	Tiempo Integrativo (T_i)
TD	Tiempo Derivativo (T_d)

B.3 CONTROL DE LA PLANTA



Al iniciar esta rutina el periodo de muestreo se carga con RATE1.

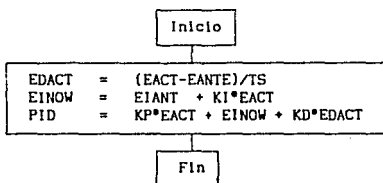
Para detectar un error más grande de lo deseable se verifica cada 3 segundos si el promedio de error de los errores muestreados durante ese periodo de tiempo excede los 100mV, si es así se calculan nuevos parámetros, ya que se supone que el controlador no funciona como es debido. La fórmula utilizada para obtener el promedio de error es la siguiente:

$$\bar{e} = \frac{\sum_{i=1}^n e_i}{n}$$

donde

n = número de muestras
 e_i = error en la muestra i
 \bar{e} = error promedio

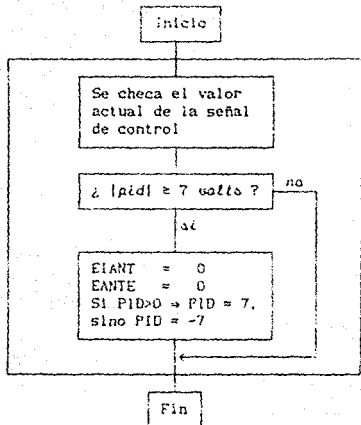
B.3.1 CALCULO DEL VALOR DE LA SEÑAL DE CONTROL



En esta rutina se calcula el valor de salida de la señal de control (PID). Se hace notar que el valor de PID se invierte a la salida debido a que el motor invierte el valor que se tenga a su entrada. Los nombres de las variables dentro de la fórmula (que es la que se menciona en el capítulo dos) corresponden a los utilizados en el programa y significan lo siguiente:

EACT	Error ACTUAL
EANT	Error ANTERIOR
EDACT	Error Derivativo ACTUAL
EIAANT	Error Integrativo ANTERIOR
EINOW	Error Integrativo actual (del Inglés NOW [ahora])
KP	Constante Proporcional
KI	Constante Integrativa
kD	Constante Derivativo

B.3.2 VERIFICACION DE POSIBLE SATURACION



Este bloque evita que el sistema alcance el valor límite de salida en la señal de control.

Como se trata de un sistema digital no existe saturación; cuando se alcanza un valor límite (+10volts = 8000H ó -10volts = 7FFFH), se puede pasar de +10 a -10 volts (o viceversa) en forma brusca restando 1 a 8000H (o sumando 1 a 7FFFH), de manera que se evita esto volviendo a condiciones iniciales. Al valor de salida de la señal (PID) se le asigna el valor límite de 7 (o en su caso -7) volts. En realidad, el diagrama de flujo se encuentra simplificado, ya que no se pregunta por el valor absoluto de PID, sino que se le compara, según su signo, con 7volts (LIMP) ó -7volts (LIMN).

RUTINAS GENERALES DEL PROGRAMA

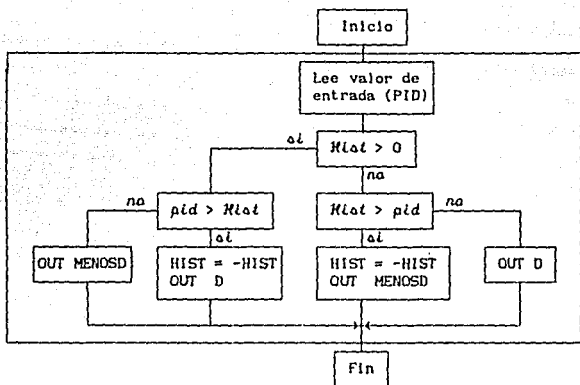
El programa cuenta con varias rutinas que pueden ser llamadas desde cualquier bloque del sistema. Sólo una de ellas, la rutina de relevador,

ha sido mencionada explícitamente en los diagramas de flujo. Existen, además, otras 2 rutinas de uso general: la rutina de división, de la cual se presentará su diagrama de flujo junto con el de la rutina de relevador, y una rutina de 5 líneas que sirve para convertir un número de Q30 a Q15.

ER RUTINA DEL RELEVADOR

Esta rutina implementa el relevador (la alinealidad con histéresis) que se necesita para hacer oscilar a la planta.

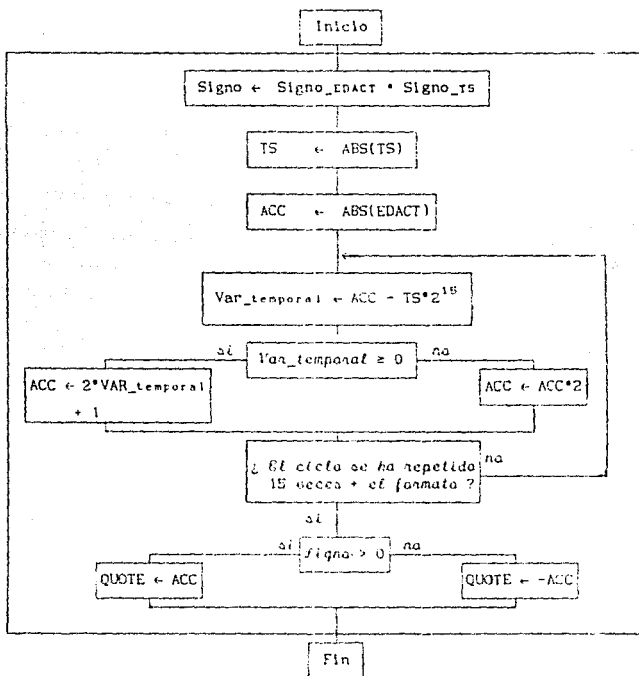
A continuación se muestra el diagrama de flujo correspondiente.



En este caso, en PID no se tiene la señal de salida del controlador, sino un dato de la señal de entrada. Esto se realiza así para economizar la memoria de datos.

RUTINA DE DIVISION

Esta rutina realiza la división entre las variables EDACT y TS, asignándole el resultado a la variable QUOTE. El hecho de utilizar las primeras dos variables se debe no sólo a la consabida economía de memoria, sino también a que se ahorra código al momento de programar ya que la división de estas dos variables se da dentro del programa.



Se hace notar que el formato que se menciona en una condición del

diagrama de flujo se refiere a la cantidad de bits que se le asignan a la parte fraccionaria del resultado. En la implementación este formato está guardado en la constante FORMAT y vale 15 (el resultado quedará en Q15).

Por otro lado se recuerda que en un sistema binario el multiplicar por 2 equivale a hacer corrimientos a la izquierda, que es lo que se utiliza para implementar la división.

DIAGRAMA DE IMPLEMENTACION FISICA

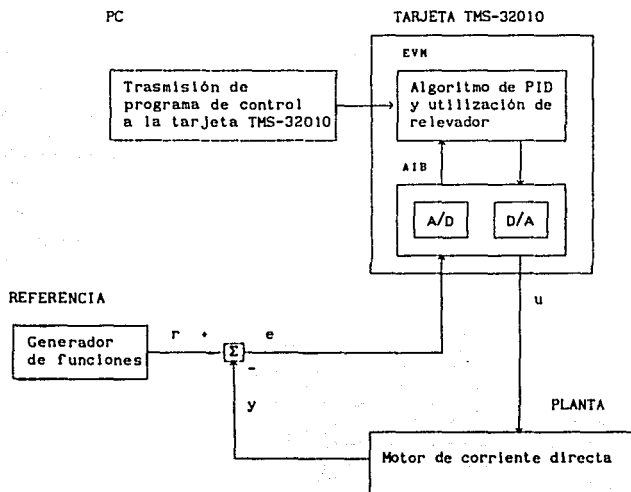


fig. 3.4

donde

- r = señal de referencia
- y = señal de la planta (velocidad)
- e = error
- u = señal de control

IV. CONTROL DE UN MOTOR DE CORRIENTE DIRECTA

4.1. INTRODUCCION

El control que se desea realizar en el presente trabajo, es el de la velocidad de un motor de corriente directa empleando un controlador tipo PID.

Para mostrar cómo se realiza el cálculo de parámetros del controlador, así como la calidad de su control, se presentan en este capítulo una serie de gráficas que muestran el funcionamiento del sistema completo (el programa implementado en la tarjeta TMS-32010 [relevador-controlador] y la respuesta de la planta).

4.2. MOTOR DE CORRIENTE DIRECTA (DC)

4.2.1. Modelo matemático

En esta tesis lo que se desea controlar es la velocidad de un motor de corriente directa, para lo cual se tiene el modelo matemático siguiente :

$$G(s) = \frac{K e^{-sL}}{\tau s + 1} \quad (4.1)$$

donde

- K es la ganancia de la planta
- τ es la constante de tiempo de la planta
- L es el retardo de fase

4.2.2 Respuesta en frecuencia

Como se mencionó en el capítulo dos, la no-linealidad por medio de un relevador con histéresis, se puede representar mediante la siguiente función descriptiva

$$H = \frac{4d}{\pi} \angle \sin^{-1}(h/a)$$

donde

- a es el valor de la amplitud de la entrada sinusoidal;
- d es la amplitud de salida del relevador;
- h es el ancho de histéresis.

La ecuación característica para un sistema de lazo cerrado es

$$MG(s) + 1 = 0$$

por lo tanto, la ganancia y el periodo crítico de la planta se obtienen cuando esta ecuación es satisfecha, lo que implica que

$$G(s) = \frac{-1}{N}$$

En la figura 4.1 se muestra la respuesta en frecuencia de $G(s)$, junto con la curva en frecuencia del inverso negativo de la función descriptiva $(-1/N)$, con el objeto de indicar por qué existen oscilaciones en el sistema al ponerlo en modo de sintonización.

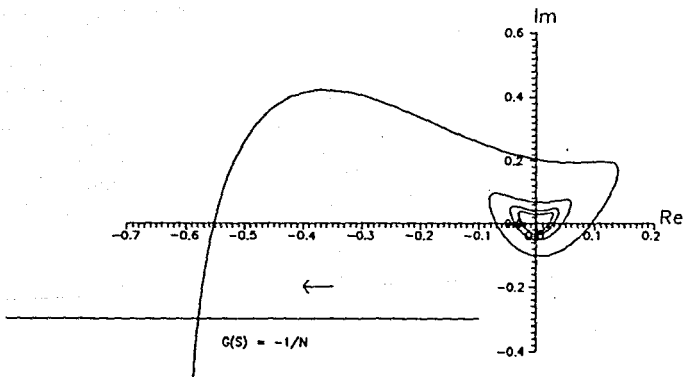


fig. 4.1

Como se puede observar, si se satisface la condición de $G(s) = -1/N$ (relación que se obtiene de la ecuación característica); justo cuando esta condición se cumple, se obtiene un valor de periodo muy cercano al periodo crítico de la planta ($t_c = 2\pi/\omega_c$). Para fines prácticos, este valor puede ser considerado como el periodo crítico, siendo la ganancia crítica (k_c) igual al valor del módulo de N en ese momento ($|14D/\omega|$).

4.3. EXPERIMENTOS DEL CONTROLADOR PID AUTOSINTONIZABLE CON EL MOTOR DE DC

4.3.1. Objetivos de los experimentos

Los objetivos que se desean alcanzar son :

- Mostrar la eficiencia del algoritmo de autosintonización implementado en el presente trabajo;
- Mostrar la ventaja que tiene un PID autosintonizable sobre un PID convencional; y
- Mostrar gráficamente las etapas del controlador autosintonizable.

4.3.2. Resultados de los experimentos

Las pruebas realizadas se llevan a cabo utilizando tres señales de referencia distintas : señal senoidal, señal cuadrada y señal escalón.

En las pruebas realizadas con el programa, y que se presentan un poco más adelante en este capítulo, lo primero que se hace es inicializar los parámetros del PID con valores no necesariamente adecuados a las características de la planta (intervalo A de las gráficas), lo que provoca un mal desempeño del sistema (véanse las gráficas de control y error, en la segunda se observa la existencia de un error considerable, en tanto que en la primera se puede ver cómo el controlador no genera ninguna señal de corrección), lo que a su vez deriva en la conexión del relevador al sistema. Lo que se desea ilustrar en esta parte del experimento es cómo un PID mal sintonizado realiza un mal control; si este sistema fuese un controlador convencional, controlaría mal hasta que un ingeniero de control lo sintonizara nuevamente; en cambio, el

controlador autosintonizable detecta el mal funcionamiento del sistema y automáticamente se pone en modo de sintonización, lo que se puede observar en las gráficas durante el intervalo B (en la señal de control se puede ver la onda cuadrada que genera el relevador, mientras que en la señal de error se puede observar la señal oscilatoria que se produce en el motor). A partir de ese momento se puede constatar la eficacia de la sintonización realizada por el sistema.

Para mostrar la robustez del controlador, en el punto C se le aplica una perturbación a la planta, de la cual se recupera tan pronto como se le deja de aplicar. Si la perturbación resulta ser muy prolongada (intervalo D), se realizará una nueva sintonización (intervalo E).

Cabe hacer notar que en el caso de la señal cuadrada el punto C indica la perturbación prolongada y el intervalo D la nueva sintonización.

Se obtuvieron 12 gráficas para la planta en condiciones normales y 12 para la planta con perturbación (en este caso la perturbación la constituye un freno magnético), la cual implica un cambio considerable en la constante de tiempo del motor, así como en su ganancia, situación que en términos estrictos significa el trabajar con una nueva planta. En ambos casos las 12 gráficas están constituidas por 3 grupos (uno para cada tipo de señal) de 4 gráficas cada uno (referencia, salida, control y error). El objetivo de este experimento es mostrar que la sintonización se realiza adecuadamente sin tener en consideración un previo conocimiento de las características de la planta. Los valores aproximados de la ganancia (k) y la constante de tiempo (τ) para la planta sin freno son $k=68.6$ y $\tau=1.07\text{seg.}$ mientras que para la planta con freno son $k=7.33$ y $\tau=0.056\text{seg.}$

Las gráficas representan la señal de entrada, la señal de salida, la señal de control y la señal de error. Se muestran en ellas los valores de K_p , T_i y T_d que se obtuvieron por medio del autosintonizador.

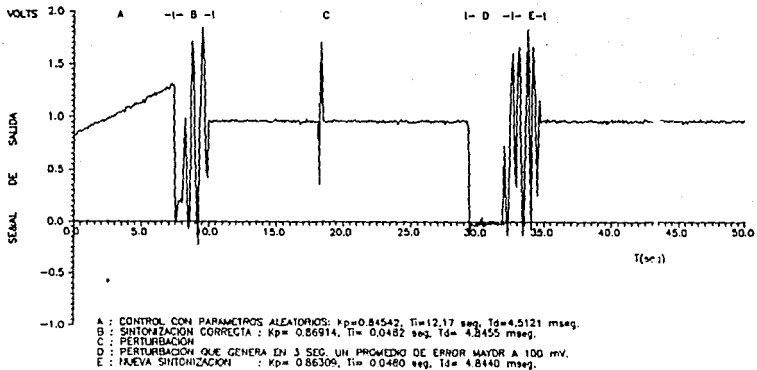
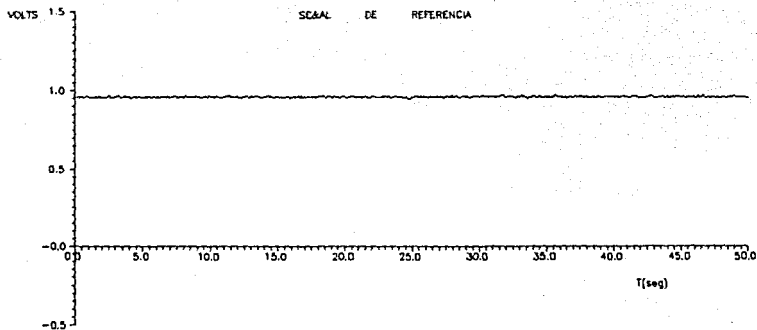
Pruebas para la señal escalón. Se realizaron pruebas para una señal escalón con el objeto de comprobar qué tan estable resulta la velocidad de la planta bajo la acción del controlador, a la vez que observar la capacidad del sistema de alcanzar un valor preestablecido.

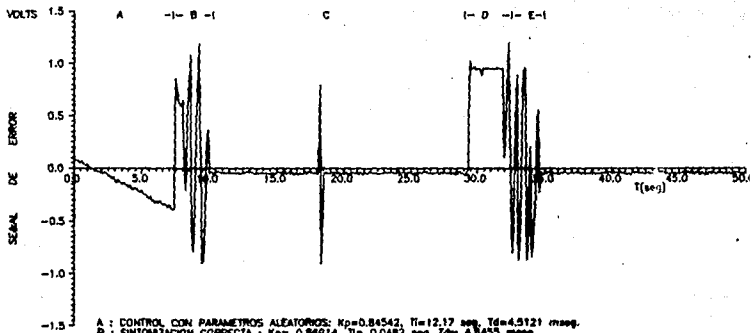
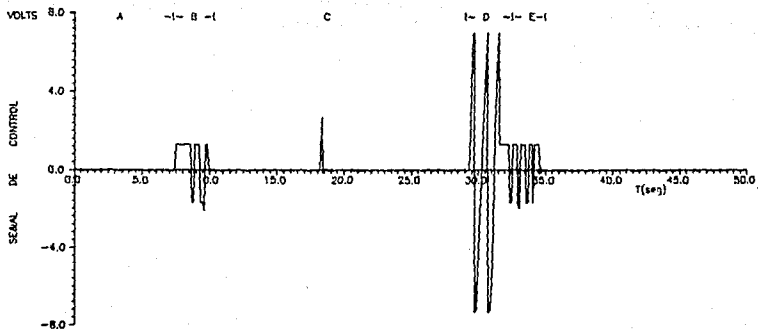
Pruebas para la señal senoidal. El objetivo de estas pruebas es comprobar la capacidad del sistema para seguir una señal que presenta cambios suaves en el tiempo.

Pruebas para la onda cuadrada. Con estos experimentos se pretende ver qué tan bien puede responder el sistema a cambios bruscos en la señal de referencia. A esta señal se le pasa por un filtro paso bajas, antes de ser aplicada a la planta, con el objeto de suavizar un poco su pendiente, ya que, por características propias del motor, no es posible seguir en forma adecuada a la señal original.

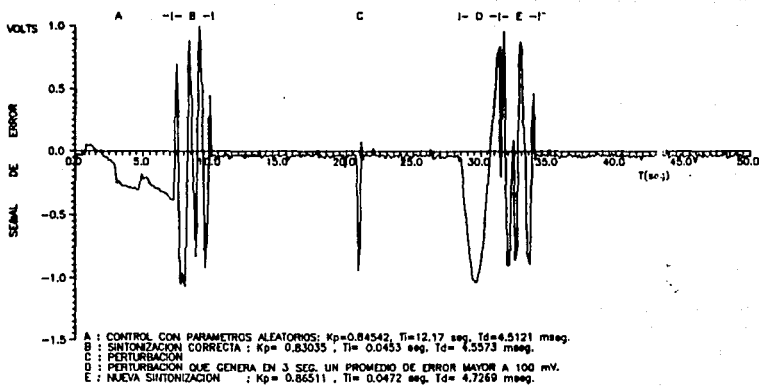
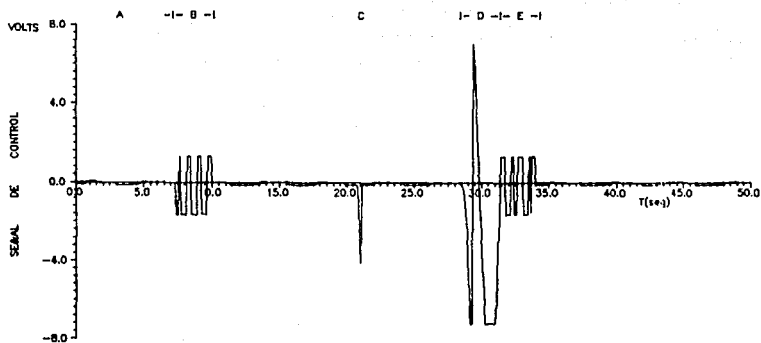
A. PRUEBAS PARA LA PLANTA SIN FRENO

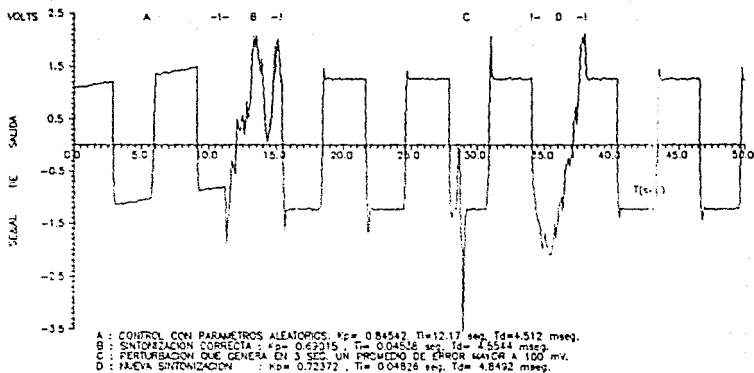
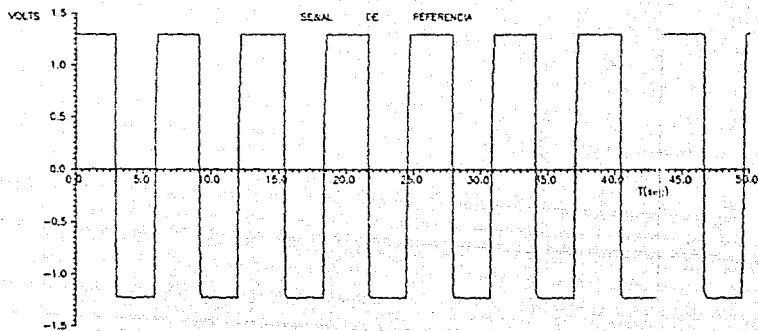
Las gráficas que se muestran en las siguientes páginas corresponden a las pruebas que se hicieron a la planta sin la aplicación del freno magnético.

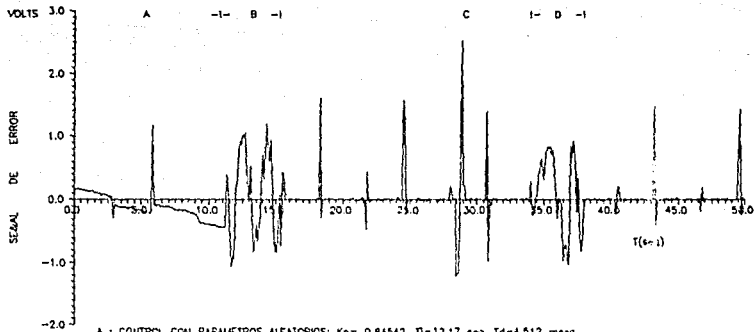
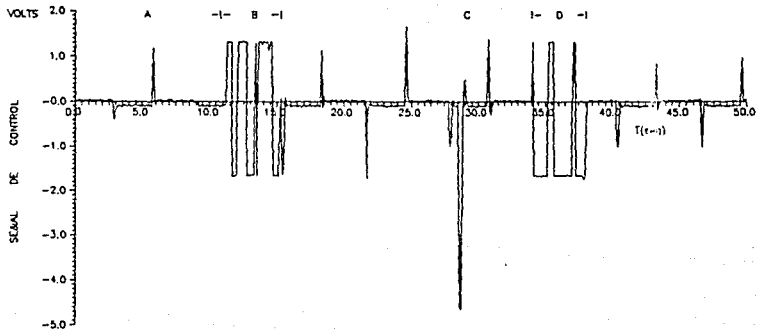




A : CONTROL CON PARAMETROS ALEATORIOS: $K_p=0.84542$, $T_i=12.17$ seg, $T_d=4.5121$ msec.
 B : SINTONIZACION CORRECTA : $K_p= 0.84914$, $T_i= 0.0482$ seg, $T_d= 4.8453$ msec.
 C : PERTURBACION
 D : PERTURBACION QUE GENERA EN 3 SEG. UN PROMEDIO DE ERROR MAYOR A 100 mV.
 E : NUEVA SINTONIZACION : $K_p= 0.86309$, $T_i= 0.0480$ seg, $T_d= 4.8460$ msec.



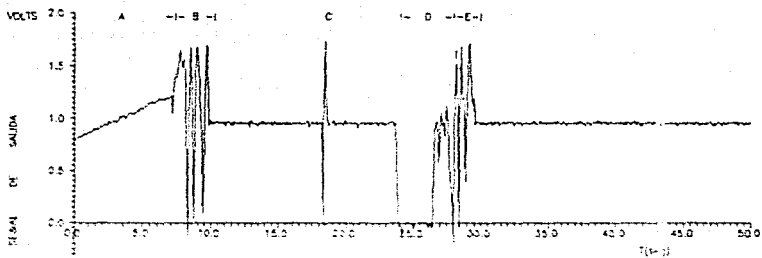
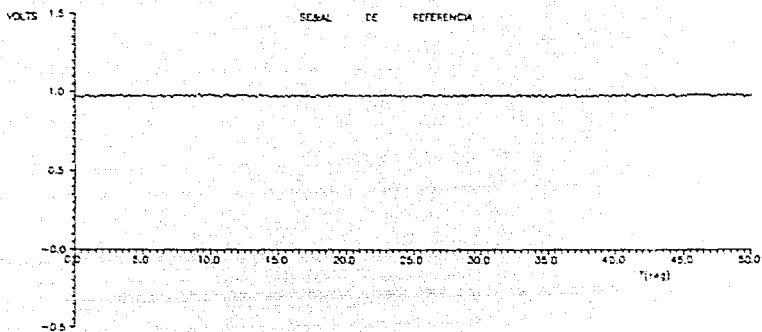




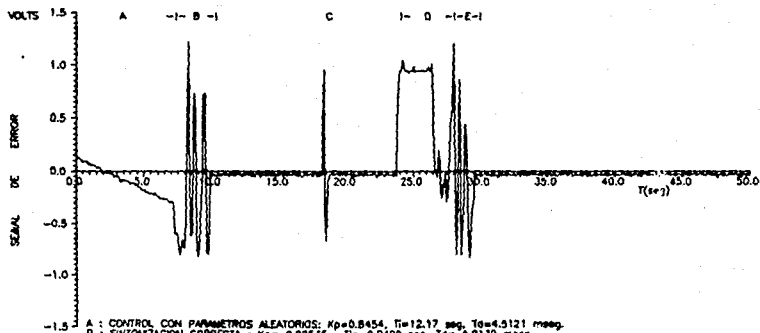
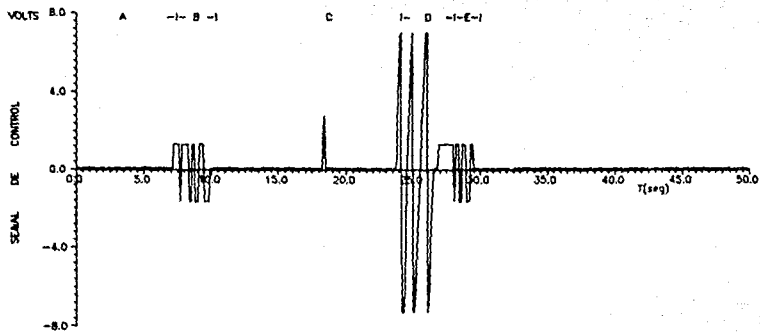
A : CONTROL CON PARAMETROS ALEATORIOS : $K_p = 0.84542$, $T_i = 12.17$ seg. , $T_d = 4.512$ msec.
 B : SINTONIZACION CORRECTA : $K_p = 0.69015$, $T_i = 0.04538$ seg. , $T_d = 4.5544$ msec.
 C : PERTURBACION QUE GENERA EN 3 SEC. UN PROMEDIO DE ERROR MAYOR A 100 mV.
 D : NUEVA SINTONIZACION : $K_p = 0.72372$, $T_i = 0.04826$ seg. , $T_d = 4.8492$ msec.

B. PRUEBAS PARA LA PLANTA CON FRENO

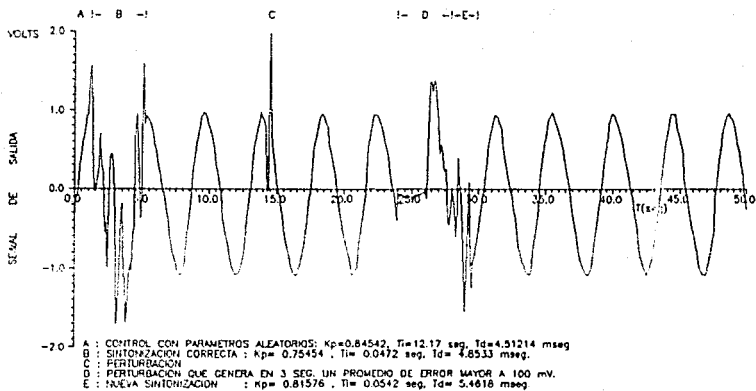
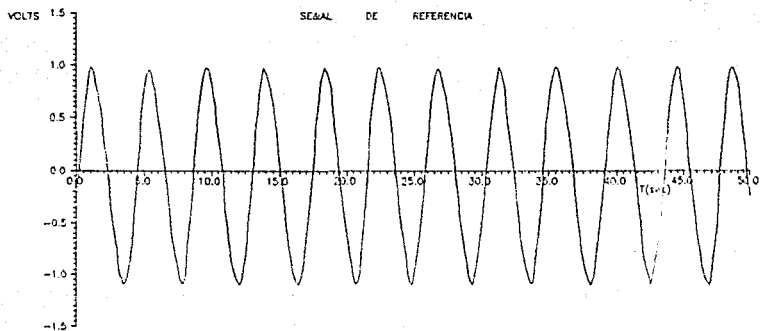
Las gráficas que se muestran en las siguientes páginas corresponden a las pruebas realizadas para la planta al aplicársele una perturbación usando el freno magnético. El objetivo de estas pruebas es demostrar el buen desempeño del controlador autosintonizable, ya que al aplicársele el freno a la planta, se cambia la dinámica de la misma (es como si se trabajara con otra planta).

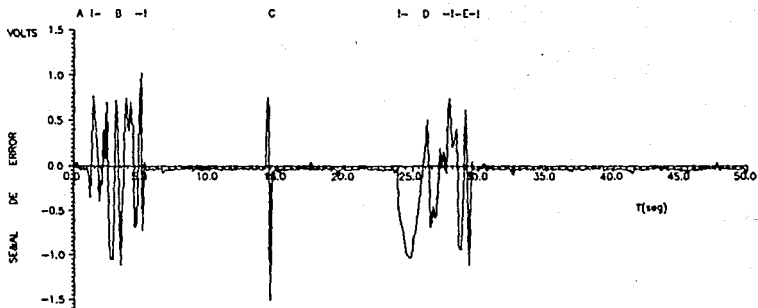
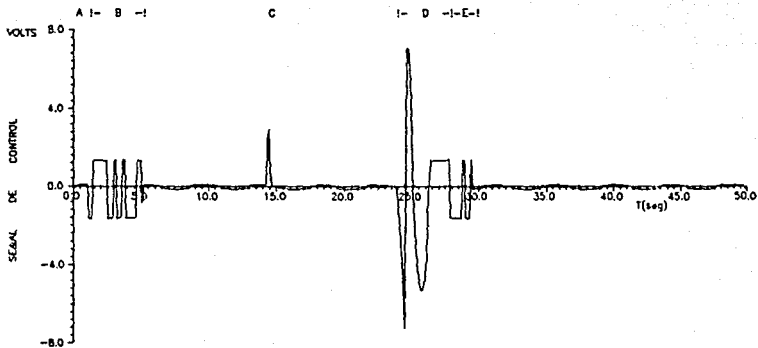


A : CONTROL CON PARAMETROS ALEATORIOS. $K_p=0.8454$. $T_i=12.17$ seg. $T_d=4.5121$ msec.
 B : SINTONIZACION CORRECTA : $K_p=0.92535$. $T_i=0.0490$ seg. $T_d=4.9139$ msec.
 C : PERTURBACION
 D : PERTURBACION QUE GENERA EN 3 SEG. UN PROMEDIO DE ERROR MAYOR A 100 mV.
 E : NUEVA SINTONIZACION : $K_p=0.93701$. $T_i=0.0485$ seg. $T_d=4.8527$ msec.

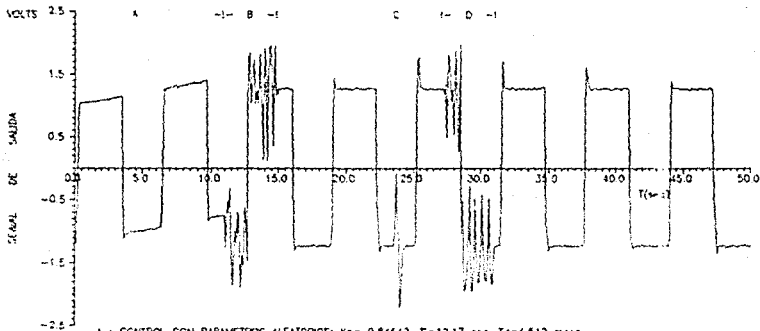
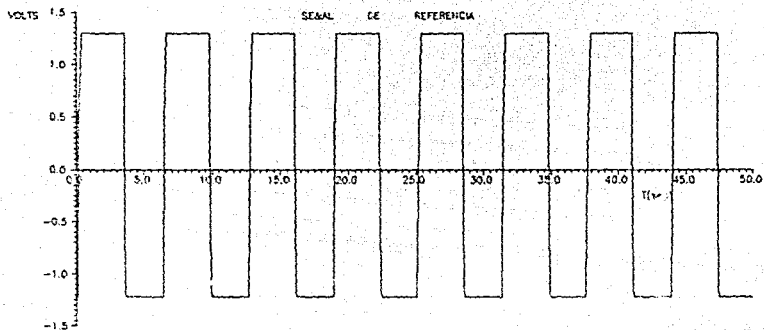


A : CONTROL CON PARAMETROS ALEATORIOS: $K_p=0.8454$, $T_i=12.17$ seg, $T_d=4.3121$ msec.
 B : SINTONIZACION CORRECTA : $K_p= 0.92555$, $T_i= 0.0490$ seg, $T_d= 9.9138$ msec.
 C : PERTURBACION
 D : PERTURBACION QUE GENERA EN 3 SEG. UN PROMEDIO DE ERROR MAYOR A 100 mV.
 E : NUEVA SINTONIZACION : $K_p= 0.83701$, $T_i= 0.0485$ seg, $T_d= 4.8527$ msec.

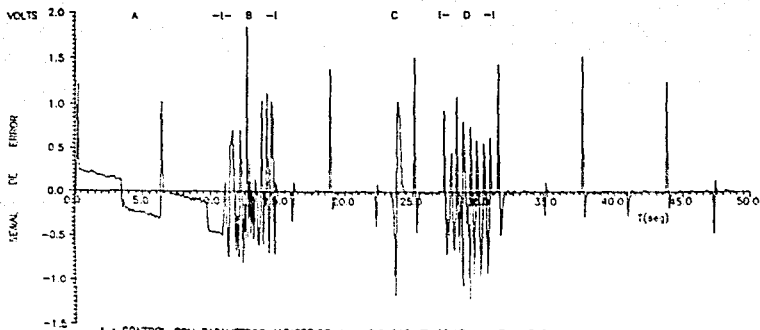
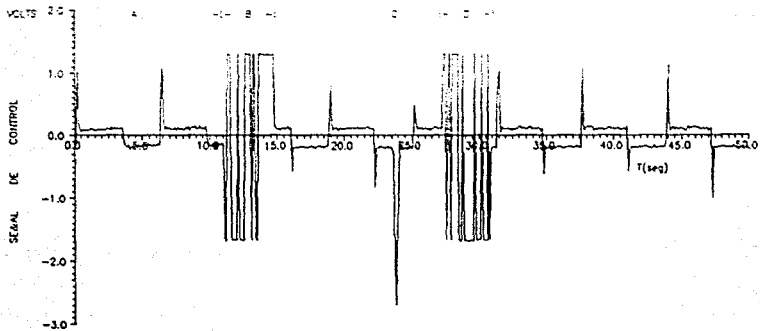




-2.0 A : CONTROL CON PARAMETROS ALEATORIOS : $K_p=0.84542$, $T_i=12.17$ seg, $T_d=4.51214$ mseg
 B : SINTONIZACION CORRECTA : $K_p=0.75454$, $T_i=0.0472$ seg, $T_d=4.8533$ mseg.
 C : PERTURBACION
 D : PERTURBACION QUE GENERA EN 3 SEG. UN PROMEDIO DE ERROR MAYOR A 100 mV.
 E : NUEVA SINTONIZACION : $K_p=0.81576$, $T_i=0.0542$ seg, $T_d=5.4618$ mseg.



- A : CONTROL CON PARAMETROS ALEATORIOS: $K_p = 0.56542$, $T_i = 12.17$ seg, $T_d = 4.512$ msec.
 B : SINTONIZACION CORRECTA : $K_p = 0.72911$, $T_i = 0.05549$ seg, $T_d = 4.8402$ msec.
 C : PERTURBACION QUE GENERA EN 3 SEG. UN PROMEDIO DE ERROR MAYOR A 100 mV.
 D : NUEVA SINTONIZACION : $K_p = 0.73083$, $T_i = 0.06217$ seg, $T_d = 5.2218$ msec.



A : CONTROL CON PARAMETROS ALEATORIOS: $K_p = 0.84542$, $T_i = 12.17$ seg, $T_d = 4.512$ msec.
 B : SINTONIZACION CORRECTA : $K_p = 0.78111$, $T_i = 0.06549$ seg, $T_d = 1.842$ msec.
 C : PERTURBACION QUE GENERA EN 10 SEG. UN PROMEDIO DE ERROR MAYOR A 100 mV.
 D : NUEVA SINTONIZACION : $K_p = 0.73083$, $T_i = 0.06217$ seg, $T_d = 6.2218$ msec.

V. CONCLUSIONES

En este trabajo se han estudiado los controladores tipo PID autosintonizables desarrollados en el dominio del tiempo y de la frecuencia; a la vez que se ha realizado un programa en lenguaje ensamblador para la implementación del controlador PID, desarrollado en el dominio de la frecuencia, en una tarjeta TMS-32010. De los resultados experimentales se ha llegado a las siguientes conclusiones:

1) Como se puede apreciar en la primera parte de los experimentos realizados, esto es, cuando se hace trabajar un controlador PID con malos parámetros, se observa que un PID normal hubiera controlado de manera inadecuada siempre, mientras que un PID autosintonizable presenta la ventaja de corregir este error calculando nuevos parámetros;

2) Se puede constatar cómo los parámetros del controlador (K_p , T_i y T_d) varían poco en las distintas pruebas, esto es debido a que dependen directamente de K_c y T_c , siendo estos valores inherentes a la planta y, por tanto, independientes de la entrada a la misma;

3) Se observa claramente cómo la práctica corresponde a la teoría: se crea una señal cuadrada con el relevador, se tiene un ciclo límite a partir del cual se puede obtener tanto K_c como T_c , valores estos últimos que permitan calcular los parámetros del controlador adecuados a las características de la planta. También se constata cómo el controlador con los parámetros de esta forma obtenidos funciona adecuadamente;

4) Comparando este método de sintonización con los métodos de sintonización del controlador PID continuo, podemos notar que, si existen variaciones en las condiciones de la planta, el algoritmo de autosintonización es capaz de corregir automáticamente el error por medio del cálculo de nuevos parámetros, en tanto que los otros métodos de sintonización necesitarían de un ingeniero de control que analizara las condiciones actuales de la planta para obtener nuevamente un control adecuado, con lo que podemos ver que para la autosintonización no se requiere un conocimiento previo de las características de la planta. Esta capacidad de autorregulación es lo que hace superior a un controlador autosintonizable con respecto a un controlador normal;

5) Este controlador autosintonizable resulta ser de fácil implementación, a la vez que muestra robustez contra cambio de parámetros en la planta;

6) Buen desempeño logrado para un mayor rango de operación; y

7) El programa se puede adaptar para su implementación en circuitos impresos.

APENDICE A.

LISTADO DEL PROGRAMA ENSAMBLADOR DEL CONTROLADOR AUTOSINTONIZABLE

0001 •
 0002 • PROGRAMA DE IMPLEMENTACION DEL CONTROLADOR
 0003 • AUTOSINTONIZABLE TIPO PID, EN LA TARJETA
 0004 • DE PROCESAMIENTO DIGITAL DE SEÑALES
 0005 • TMS-32010
 0006 •
 0007 0000 AORG 0
 0008 0000 F900 B START
 0001 0019
 0009 •
 0010 0000 ERRORH EQU 0
 0011 0001 ERRORL EQU 1
 0012 0002 ANPID EQU 2
 0013 0003 TEMPC EQU 3
 0014 0004 TI EQU 4
 0015 0005 TS EQU 5
 0016 0006 KI EQU 6
 0017 0007 KP EQU 7
 0018 0008 KD EQU 8
 0019 0009 EACT EQU 9
 0020 000A EIA NT EQU >A
 0021 000B EINOW EQU >B
 0022 000C EANTE EQU >C
 0023 000D EDACT EQU >D
 0024 000E TEMP EQU >E
 0025 000F PID EQU >F
 0026 0010 FORMAT EQU >10
 0027 0011 TEMSCN EQU >11
 0028 0012 QUOTE EQU >12
 0029 0013 PUNTOS EQU >13
 0030 0014 HIST EQU >14
 0031 0015 D EQU >15
 0032 0016 MENOSD EQU >16
 0033 0017 GCRIT EQU >17
 0034 0018 DEC EQU >18

0035	0019	UNO	EQU	>19
0036	001A	CINCO	EQU	>1A
0037	001B	PCRIT	EQU	>1B
0038	001C	SAMPLE	EQU	>1C
0039	001D	MAXERR	EQU	>1D
0040	001E	RESUL	EQU	>1E
0041	001F	AYUDA	EQU	>1F
0042	0020	PI	EQU	>20
0043	0021	CUATRO	EQU	>21
0044	0022	REPEAT	EQU	>22
0045	0023	PICINV	EQU	>23
0046	0024	FRONTN	EQU	>24
0047	0025	FRONTP	EQU	>25
0048	0026	MUNO	EQU	>26
0049	0027	LIMP	EQU	>27
0050	0028	LIMN	EQU	>28
0051	0029	TD	EQU	>29
0052	002A	NPID	EQU	>2A
0053		*		
0054		* CONSTANTES		
0055		*		
0056	0002	FFFF	MONTE	DATA -1
0057	0003	0000	CERO	DATA 0
0058	0004	0001	ONE	DATA 1
0059	0005	0002	CRUCES	DATA 2
0060	0006	000F	FRAK	DATA 15
0061	0007	02FB	NVECES	DATA 763
0062	0008	FFFF	IVECES	DATA >FFFF
0063	0009	0003	ERRMAX	DATA 3
0064		*		
0065		* NUMEROS EN Q15		
0066		*		
0067	000A	2000	FOUR	DATA 8192
0068	000B	0667	FIVE	DATA 1639
0069	000C	0003	PH	DATA 3

ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA

0070	000D	00D8	PM2	DATA	216
0071	000E	F6E5	HISTER	DATA	>F6E5
0072	000F	EDC0	DE	DATA	>EDC0
0073	0010	4000	HALF	DATA	16384
0074	0011	28BE	PI15	DATA	10430
0075	0012	067C	PIC15	DATA	1660
0076	0013	A800	FRONTS	DATA	>A800
0077	0014	C000	COMPA2	DATA	>C000
0078	0015	4000	COMPA3	DATA	>4000
0079			•		
0080	0016	00FA	MODE	DATA	>FA
0081	0017	01F3	RATE	DATA	499
0082	0018	7FFF	RATE1	DATA	>7FFF
0083			•		
0084			•	RUTINA DE INICIO DE PROGRAMA	
0085			•	SE CARGAN LAS CONSTANTES	
0086			•		
0087	0019	6E00	START	LDPK	0
0088	001A	7E16		LACK	MODE
0089	001B	6700		TBLR	0
0090	001C	4800		OUT	0,0
0091	001D	7E17		LACK	RATE
0092	001E	6700		TBLR	0
0093	001F	4900		OUT	0,1
0094	0020	7E11		LACK	PI15
0095	0021	6720		TBLR	PI
0096	0022	7E12		LACK	PIC15
0097	0023	6723		TBLR	PICINV
0098	0024	7E04		LACK	ONE
0099	0025	6719		TBLR	UNO
0100	0026	7E02		LACK	MONE
0101	0027	6726		TBLR	MUNO
0102	0028	7E0A		LACK	FOUR
0103	0029	6721		TBLR	CUATRO

0104 002A 7E0B	LACK FIVE
0105 002B 671A	TBLR CINCO
0106 002C 7E09	LACK ERRMAX
0107 002D 671D	TBLR MAXERR
0108 002E 7E06	LACK FRAK
0109 002F 6710	TBLR FORMAT
0110 0030 7E10	LACK HALF
0111 0031 6713	TBLR PUNTOS
0112 0032 6880	LARP O
0113 0033 7E0F	LACK DE
0114 0034 6715	TBLR D
0115 0035 7F89	ZAC
0116 0036 1015	SUB D
0117 0037 5016	SACL MENOSD
0118 0038 7E0C	LACK PM
0119 0039 671C	TBLR SAMPLE
0120 003A 7E13	LACK FRONTS
0121 003B 6724	TBLR FRONTN
0122 003C 7F89	ZAC
0123 003D 1024	SUB FRONTN
0124 003E 5025	SACL FRONTP
0125 003F 7E0E	LACK HISTER
0126 0040 6714	TBLR HIST
0127 0041 7E14	LACK COMPA2
0128 0042 6727	TBLR LIMP
0129 0043 7E15	LACK COMPA3
0130 0044 6728	TBLR LIMN
0131 0045 F900	B VERPER
0046 00CD	

0132				
0133			RUTINA DE RELEVADOR	
0134				
0135	0047	F800	RELAY	BIOZ SIGUE
	0048	004B		
0136	0049	F900	B	RELAY
	004A	0047		
0137	004B	420F	SIGUE	IN PID,2
0138	004C	2014	LAC	HIST
0139	004D	FC00	BGZ	ETIQ1
	004E	005B		
0140	004F	200F	LAC	PID
0141	0050	1014	SUB	HIST
0142	0051	FA00	BLZ	ETIQ2
	0052	0056		
0143	0053	4A15	OUT	D,2
0144	0054	F900	B	VOLVER
	0055	0065		
0145	0056	F800	ETIQ2	CALL NIEGA
	0057	0066		
0146	0058	4A16	OUT	MENOSD,2
0147	0059	F900	B	VOLVER
	005A	0065		
0148	005B	200F	ETIQ1	LAC PID
0149	005C	1014	SUB	HIST
0150	005D	FC00	BGZ	ETIQ3
	005E	0062		
0151	005F	4A16	OUT	MENOSD,2
0152	0060	F900	B	VOLVER
	0061	0065		
0153	0062	F800	ETIQ3	CALL NIEGA
	0063	0066		
0154	0064	4A15	OUT	D,2
0155	0065	7F8D	VOLVER	RET

0156	0066	7F89	NIEGA	ZAC	
0157	0067	1014		SUB	HIST
0158	0068	5014		SACL	HIST
0159	0069	7F8D		RET	
0160				*	
0161				*	RUTINA DE DIVISION
0162				*	
0163				*	DIVIDE EDACT/TS=QUOTE
0164				*	
0165	006A	6A0D	DIV	LT	EDACT
0166	006B	6D05		MPY	TS
0167	006C	7F8E		PAC	
0168	006D	5811		SACH	TEMSCN
0169	006E	2005		LAC	TS
0170	006F	7F88		ABS	
0171	0070	5005		SACL	TS
0172	0071	7E0F		LACK	15
0173	0072	0010		ADD	FORMAT
0174	0073	500E		SACL	TEMP
0175	0074	200D		LAC	EDACT
0176	0075	7F88		ABS	
0177	0076	380E		LAR	O, TEMP
0178	0077	6405	KPDVNC	SUBC	TS
0179	0078	F400		BANZ	KPDVNC
	0079	0077			
0180	007A	5012		SACL	QUOTE
0181	007B	2011		LAC	TEMSCN
0182	007C	FD00		BGEZ	DONE
	007D	0081			
0183	007E	7F89		ZAC	
0184	007F	1012		SUB	QUOTE
0185	0080	5012		SACL	QUOTE
0186	0081	7F8D	DONE	RET	

0187		•			
0188		•	RUTINA DE INICIALIZACION DE ERRORES		
0189		•	ANTERIORES DEL PID		
0190		•			
0191	0082 7E03	INIT	LACK	CERO	
0192	0083 670A		TBLR	EIANT	
0193	0084 670C		TBLR	EANTE	
0194	0085 200F		LAC	PID	
0195	0086 FA00		BLZ	CARGAN	
	0087 008C				
0196	0088 2025		LAC	FRONTP	
0197	0089 500F		SACL	PID	
0198	008A F900		B	BACK	
	008B 008E				
0199	008C 2024	CARGAN	LAC	FRONTN	
0200	008D 500F		SACL	PID	
0201	008E 7F8D	BACK	RET		
0202		•			
0203		•	RUTINA DE CONVERSION DE Q30 A Q15		
0204		•			
0205	008F 7F8E	CONV	PAC		
0206	0090 5903		SACH	TEMPC, 1	
0207	0091 7F89		ZAC		
0208	0092 6003		ADDH	TEMPC	
0209	0093 7F8D		RET		
0210		•			
0211		•	RUTINA DE OBTENCION DE PERIODO		
0212		•			
0213	0094 7E05	PERIOD	LACK	CRUCES	
0214	0095 6718		TBLR	DEC	
0215	0096 7E03		LACK	CERO	
0216	0097 671E		TBLR	RESUL	
0217	0098 F800		CALL	RELAY	
	0099 0047				

0218	009A	200F	LAC	PID
0219	009B	5002	SACL	ANPID
0220	009C	F800	ALOOP	CALL RELAY
	009D	0047		
0221	009E	200F	LAC	PID
0222	009F	FA00	BLZ	ET110
	00A0	00A5		
0223	00A1	2002	LAC	ANPID
0224	00A2	FA00	BLZ	LOOPO
	00A3	00AD		
0225	00A4	F900	B	ET120
	00A5	00A9		
0226	00A5	2002	ET110 LAC	ANPID
0227	00A7	FD00	BGEZ	LOOPO
	00A8	00AD		
0228	00A9	200F	ET120 LAC	PID
0229	00AA	5002	SACL	ANPID
0230	00AB	F900	B	ALOOP
	00AC	009C		
0231			*	
0232			* ESPERA A DETECTAR UN CRUCE POR CERO	
0233			* POSTERIORMENTE DETECTA OTROS DOS	
0234			* PARA CALCULAR EL PERIODO DE	
0235			* OSCILACION	
0236			*	
0237	00AD	200F	LOOPO LAC	PID
0238	00AE	5002	SACL	ANPID
0239	00AF	F800	ALOOP	CALL RELAY
	00B0	0047		
0240	00B1	200F	LAC	PID
0241	00B2	FA00	BLZ	ET11
	00B3	00BC		
0242	00B4	2002	LAC	ANPID
0243	00B5	FD00	BGEZ	ET12
	00B6	00C2		

0244	00B7	2018		LAC	DEC
0245	00B8	1019		SUB	UNO
0246	00B9	5018		SACL	DEC
0247	00BA	F900		B	ET12
	00BB	00C2			
0248	00BC	2002	ET11	LAC	ANPID
0249	00BD	FA00		BLZ	ET12
	00BE	00C2			
0250	00BF	2018		LAC	DEC
0251	00C0	1019		SUB	UNO
0252	00C1	5018		SACL	DEC
0253	00C2	200F	ET12	LAC	PID
0254	00C3	5002		SACL	ANPID
0255	00C4	661E		ZALS	RESUL
0256	00C5	001C		ADD	SAMPLE
0257	00C6	501E		SACL	RESUL
0258	00C7	2018		LAC	DEC
0259	00C8	FED0		BNZ	ALOOP
	00C9	00AF			
0260					
0261					• EL VALOR DEL PERIODO QUEDA EN RESUL
0262					•
0263	00CA	201E		LAC	RESUL
0264	00CB	5012		SACL	QUOTE
0265	00CC	7F8D		RET	
0266					•
0267					• RUTINA DE VERIFICACION DE SE&AL PERIODICA
0268					•
0269	00CD	F800	VERPER	CALL	PERIOD
	00CE	0094			
0270	00CF	6612		ZALS	QUOTE

0271 *
 0272 * OBTUVO PERIODO1. LO PONE EN EACT Y TS
 0273 *
 0274 00D0 5009 SACL EACT
 0275 00D1 5005 SACL TS
 0276 00D2 F800 CALL PERIOD
 00D3 0094
 0277 00D4 6612 ZALS QUOTE
 0278 *
 0279 * OBTUVO PERIODO2. LO PONE EN PCRIT
 0280 *
 0281 00D5 501B SACL PCRIT
 0282 00D6 2009 LAC EACT
 0283 00D7 1012 SUB QUOTE
 0284 00D8 7F88 ABS
 0285 *
 0286 * OBTIENE | PERIODO1 - PERIODO2 |
 0287 *
 0288 00D9 500D SACL EDACT
 0289 00DA F800 CALL DIV
 00DB 006A
 0290 *
 0291 * OBTUVO |PERIODO1 - PERIODO2 | / 100
 0292 *
 0293 00DC 2012 LAC QUOTE
 0294 00DD 101A SUB CINCO
 0295 *
 0296 * SE SALE SI |PERIODO1-PERIODO2|/PERIODO1 < 0.05
 0297 * ES DECIR SE TIENE UNA SESAL DE OSCILACION
 0298 * PERIODICA. EN CASO CONTRARIO SIGUE CALCULANDO
 0299 * UN PERIODO QUE NO VARIE EN MAS DE UN 5%
 0300 *
 0301 00DE F000 BGZ VERPER
 00DF 00CD

0302	00E0	F900	B	AMPLIT
	00E1	00F5		
0303			•	
0304			•	RUTINA DE OBTENCION DE VALORES
0305			•	PICO DE LA OSCILACION
0306			•	
0307			•	TEMP TIENE EL MAYOR VALOR POSITIVO Y
0308			•	ANPID TIENE EL MENOR VALOR NEGATIVO
0309			•	
0310	00E2	F800	PICOS	CALL RELAY
	00E3	0047		
0311	00E4	200F	LAC	PID
0312	00E5	100E	SUB	TEMP
0313	00E6	FA00	BLZ	CMPAN
	00E7	00EC		
0314	00E8	200F	LAC	PID
0315	00E9	500E	SACL	TEMP
0316	00EA	F900	B	FIN
	00EB	00F2		
0317	00EC	200F	CMPAN	LAC PID
0318	00ED	1002	SUB	ANPID
0319	00EE	FC00	BGZ	FIN
	00EF	00F2		
0320	00F0	200F	LAC	PID
0321	00F1	5002	SACL	ANPID
0322	00F2	F400	FIN	BANZ PICOS
	00F3	00E2		
0323	00F4	7F8D		RET

0324	•			
0325	•		RUTINA DE OBTENCION DE AMPLITUD	
0326	•		DE LA OSCILACION	
0327	•			
0328		00F5 F800	AMPLIT CALL	RELAY
		00F6 0047		
0329		00F7 200F	LAC	PID
0330		00F8 5002	SACL	ANPID
0331		00F9 500E	SACL	TEMP
0332		00FA 7E08	LACK	IVECES
0333		00FB 671F	TBLR	AYUDA
0334		00FC 7132	LARK	1,50
0335		00FD 6880	WIEDER LARP	0
0336		00FE 381F	LAR	0, AYUDA
0337		00FF F800	CALL	PICOS
		0100 00E2		
0338		0101 6881	LARP	1
0339		0102 F400	BANZ	WIEDER
		0103 00FD		
0340		0104 6880	LARP	0
0341		0105 2002	LAC	ANPID
0342		0106 100E	SUB	TEMP
0343		0107 5017	SACL	GCRIT
0344		0108 6A13	LT	PUNTOS
0345		0109 6D17	MPY	GCRIT
0346		010A 7F8E	PAC	
0347	•			
0348	•		GCRIT	CONTENDRA EL VALOR
0349	•		DE LA AMPLITUD DE OSCILACION	
0350	•			
0351		010B 5917	SACH	GCRIT, 1

0352 •

0353 • RUTINA DE CALCULO DE PARAMETROS

0354 •

0355 010C 6A13 CALPAR LT PUNTOS

0356 010D 6D17 MPY GCRIT

0357 010E 7F8E PAC

0358 010F 5905 SACH TS,1

0359 •

0360 • LA GANANCIA CRITICA ESTA DADA POR $GCRIT = 4 \cdot D / (A \cdot \pi)$

0361 • D=AMPLITUD DEL RELEVADOR Y A= AMPLITUD DE OSCILACION

0362 • LA CONSTANTE PROPORCIONAL ES $KP = GCRIT / 2$, POR LO TANTO

0363 • $KP = 2 \cdot D / (GCRIT \cdot \pi) = [D \cdot (1/\pi)] / (0.5 \cdot GCRIT)$

0364 •

0365 0110 6A15 LT D

0366 0111 6D20 MPY PI

0367 0112 7F8E PAC

0368 0113 590D SACH EDACT,1

0369 0114 F800 CALL DIV

 0115 006A

0370 0116 2012 LAC QUOTE

0371 0117 5007 SACL KP

0372 •

0373 • $TI = PCRIT / 2 = 0.5 \cdot PCRIT$

0374 •

0375 0118 6A13 LT PUNTOS

0376 0119 6D1B MPY PCRIT

0377 011A 7F8E PAC

0378 011B 5904 SACH TI,1

0379 •

0380 • $TD = PCRIT / (2 \cdot \pi \cdot PI) = KP \cdot PCRIT \cdot \{ 1 / (2 \cdot \pi \cdot PI) \}$

0381 •

0382 011C 6A1B LT PCRIT

0383 011D 6D23 MPY PICINV

0384 011E 7F8E PAC

0385 011F 5929 SACH TD,1

0386			
0387		• KD = KP • TD	
0388			
0389	0120 6A29	LT	TD
0390	0121 6D07	MPY	KP
0391	0122 7F8E	PAC	
0392	0123 5908	SACH	KD, 1
0393			
0394		• KI=KP*TS/TI	
0395			
0396	0124 7E0D	LACK	PM2
0397	0125 6705	TBLR	TS
0398	0126 6A07	LT	KP
0399	0127 6D05	MPY	TS
0400	0128 7F8E	PAC	
0401	0129 590D	SACH	EDACT, 1
0402	012A 2004	LAC	TI
0403	012B 5005	SACL	TS
0404	012C F800	CALL	DIV
	012D 006A		
0405	012E 2012	LAC	QUOTE
0406	012F 5006	SACL	KI
0407			
0408	0130 F900	B	CONTRO
	0131 0195		

```

0409      *
0410      *      RUTINA DE CALCULO DEL VALOR DE SALIDA
0411      *      DE LA SE&AL DE CONTROL (PID)
0412      *
0413      * EACT=ERROR ACTUAL
0414      * EANTE=ERROR ANTERIOR
0415      * EDACT=ERROR DERIVATIVO ACTUAL
0416      * EINOW=ERROR INTEGRATIVO ACTUAL
0417      * EIAANT=ERROR INTEGRATIVO ANTERIOR
0418      * TS=TIEMPO DE MUESTREO (1 UDT)
0419      * TI=CONSTANTE INTEGRATIVA
0420      * KD=CONSTANTE DERIVATIVA
0421      * KP=CONSTANTE PROPORCIONAL
0422      * EDACT=(EACT-EANT)/TS
0423      * EINOW=EIAANT+TI*TS*EACT
0424      * PID=KP*EACT+EINOW+KD*EDACT
0425      * ERRORH Y ERRORL CONSTITUYEN
0426      * LA REFERENCIA PARA SABER CUANDO
0427      * SE DEBEN MODIFICAR PARAMETROS
0428      *
0429 0132 F600 CALPID BIOZ INICIO
           0133 0136
0430 0134 F900      B      CALPID
           0135 0132
0431 0136 4209 INICIO IN      EACT,2
0432      *
0433      * SUMA ACUMULADA DE LA SE&AL DE ERROR
0434      *
0435 0137 2009      LAC      EACT
0436 0138 7F88      ABS
0437 0139 500E      SACL      TEMP
0438 013A 6601      ZALS      ERRORL
0439 013B 6000      ADDH      ERRORH
0440 013C 000E      ADD      TEMP

```

0441	013D	5001	SACL	ERRORL
0442	013E	5800	SACH	ERRORH
0443				
0444			• EDACT=(EACT-EANTE)/TS	
0445				
0446	013F	2009	LAC	EACT
0447	0140	100C	SUB	EANTE
0448	0141	500D	SACL	EDACT
0449	0142	F800	CALL	DIV
	0143	006A		
0450	0144	2012	LAC	QUOTE
0451	0145	500D	SACL	EDACT
0452				
0453			• EINOW=EIAWT + KI*EACT	
0454				
0455	0146	6A09	LT	EACT
0456	0147	6D06	MPY	KI
0457	0148	F800	CALL	CONV
	0149	008F		
0458				
0459			• AUMENTO EN LA PRECISION DEL ERROR INTEGRATIVO	
0460				
0461	014A	FE00	BNZ	NORMAL
	014B	0157		
0462	014C	2009	LAC	EACT
0463	014D	FF00	BZ	NORMAL
	014E	0157		
0464	014F	FA00	BLZ	CARUNO
	0150	0155		
0465	0151	7F89	ZAC	
0466	0152	6026	ADDDH	MUNO
0467	0153	F900	B	NORMAL
	0154	0157		

0468 0155 7F89 CARUNO ZAC
 0469 0156 6019 ADDH UNO
 0470 *
 0471 0157 600A NORMAL ADDH EIAINT
 0472 0158 580B SACH EINOW
 0473 *
 0474 * PID=KP*EACT + EINOW + KD*EDACT
 0475 *
 0476 0159 6A07 LT KP
 0477 015A 6D09 MPY EACT
 0478 015B F800 CALL CONV
 015C 008F
 0479 015D 600B ADDH EINOW
 0480 015E 580E SACH TEMP
 0481 015F 8A08 LT KD
 0482 0160 6D0D MPY EDACT
 0483 0161 F800 CALL CONV
 0162 008F
 0484 0163 600E ADDH TEMP
 0485 0164 580F SACH PID
 0486 0165 7F8D RET
 0487 *
 0488 * RUTINA DE SALIDA DE LA SE&AL DE CONTROL
 0489 * Y VERIFICACION DE CAMBIOS BRUSCOS
 0490 *
 0491 0166 F800 BECIN CALL INIT
 0167 0082
 0492 0168 F900 B FINAL
 0169 0194

0493 •
 0494 • SE NIEGA EL VALOR DE SALIDA PID PARA COMPENSAR
 0495 • LA GANANCIA NEGATIVA DEL MOTOR
 0496 •
 0497 016A 7F89 PRODAT ZAC
 0498 016B 100F SUB PID
 0499 016C 502A SACL NPID
 0500 016D 4A2A OUT NPID,2
 0501 •
 0502 • SE SALVAN VALORES DE ERRORES ANTERIORES
 0503 •
 0504 016E 660F ZALS PID
 0505 016F 5002 SACL ANPID
 0506 0170 6609 ZALS EACT
 0507 0171 500C SACL EANTE
 0508 0172 660B ZALS EINOW
 0509 0173 500A SACL EIA NT
 0510 0174 F800 CALL CALPID
 0175 0132
 0511 •
 0512 • SE VERIFICAN CAMBIOS BRUSCOS DE VOLTAJE
 0513 • Y SE LIMITA LA SE&AL DE SALIDA A (+/-) 7 VOLTS
 0514 •
 0515 0176 200F LAC PID
 0516 0177 FA00 BLZ TEST
 0178 017E
 0517 0179 1025 SUB FRONTP
 0518 017A FB00 BLEZ ZONA1
 017B 0181
 0519 017C F900 B BEGIN
 017D 0166
 0520 017E 1024 TEST SUB FRONTN
 0521 017F FA00 BLZ BEGIN
 0180 0166
 0522 •

0523	0181	200F	ZONA1	LAC	PID
0524	0182	FA00		BLZ	ZONA2
	0183	018C			
0525	0184	2002		LAC	ANPID
0526	0185	FD00		BGEZ	FINAL
	0186	0194			
0527	0187	1027		SUB	LIMP
0528	0188	FD00		BGEZ	FINAL
	0189	0194			
0529	018A	F900		B	BEGIN
	018B	0166			
0530	018C	2002	ZONA2	LAC	ANPID
0531	018D	FA00		BLZ	FINAL
	018E	0194			
0532	018F	1028		SUB	LIMP
0533	0190	FA00		BLZ	FINAL
	0191	0194			
0534	0192	F900		B	BEGIN
	0193	0166			
0535	0194	7F8D	FINAL	RET	

0535 *
0537 * BONTA PRINCIPAL DE CONTROL
0538 *

0539 0135 7538 CONGRUO LACK BICEI
0540 0136 6789 TELR 0
0541 0137 4889 ONE 0,1

0542 *
0543 0138 7533 LACK CERO
0544 0139 678C TELR EABRE
0545 013A 678A TELR ETAMB
0546 013B 7689 LACK FRO
0547 013C 6785 TELR TS
0548 013D 7689 CALL CALPIB

0549 013E 7687 LACK BICEI
0550 0140 6722 TELR REPEAT
0551 0141 6889 LAMP 0

0552 *
0553 * SE ESTABILIZA A LA PLANTA ANTES DE:
0554 * ENLACE EL ERROR PROBLEMA:
0555 *

0556 0142 7689 ESTAMB CALL FRONT
0143 016A
0557 014A 6822 ZALS REPEAT
0558 0145 8889 SIB 00
0559 0146 5822 SACL REPEAT
0560 0147 7689 ONE ESTAMB
0148 0142

0561 •
 0562 • INICIA ACCION DE CONTROL CONSIDERANDO
 0563 • UN ERROR PROMEDIO MAXIMO EN UN
 0564 • TIEMPO DETERMINADO
 0565 •
 0566 01A9 7E03 CARGAR LACK CERO
 0567 01AA 8700 TBLR ERRORH
 0568 01AB 8701 TBLR ERRORL
 0569 01AC 7E07 LACK NVECES
 0570 01AD 8722 TBLR REPEAT
 0571 01AE 8880 ANFAG LARP 0
 0572 01AF F800 CALL PRODAT
 01B0 018A
 0573 01B1 8622 ZALS REPEAT
 0574 01B2 1019 SUB UNO
 0575 01B3 5022 SACL REPEAT
 0576 01B4 FE00 BNZ ANFAG
 01B5 01AE
 0577 01B6 861D ZALS MAXERR
 0578 01B7 1000 SUB ERRORH
 0579 •
 0580 • SE SALE DE ESTA RUTINA SI EL ERROR PROMEDIO
 0581 • EN EL TIEMPO PREESTABLECIDO ES MAYOR AL ERROR
 0582 • PROMEDIO PERMITIDO. EN ESTE CASO SE VA A
 0583 • CALCULAR NUEVOS PARAMETROS.
 0584 •
 0585 01B8 FA00 BLZ START
 01B9 0019
 0586 01BA F900 B CARGAR
 01BB 01A9
 0587 •
 0588 END

BIBLIOGRAFIA

- Åström, K. J., Adaptation and Expert Control, ACC 1986, Boston.
- Åström, K. J., T. Hägglund. Automatic Tuning of Simple Regulators with Specification on Phase and Amplitud Margins, abril 1984, Proceedings IFAC.
- Åström, Karl Johan. Computer Controlled System: Theory and Desing, Prentice Hall, Inc. 1984.
- Franklin, Gene F., J. David Powell. Digital Control of Dynamic Systems, Addison-Wesley Publishing Company, 1980.
- Kraus, T.W., Myron, T.J. Self-Tuning PID Controller Uses Pattern Recognition Approaches. Control Engineering, Junio 1984.
- Nishikawa, Yoshikazu, et al. A Method for Auto-tuning of PID Control Parameters. International Federation of Automatic Control, 1984.
- Ogata, Katsuhiko. Ingeniería de control moderna, Prentice Hall. México, D.F., Mayo 1986.
- Texas Instrument. TMS32010 Analog Interface Board, User's Guide.
- Texas Instrument. TMS32010 Evaluation Module User's Guide.
- Texas Instrument. TMS-32010 User's Guide.