

Análisis, Diseño y Construcción de un Control Digital Directo usando la Computadora IPN E-32

M. en C. Romeo Urbieta Parrazales
Profesor e investigador del CINTEC-IPN
M. en C. Teodoro Alvarez Sánchez
Jefe del Departamento de Programación del CINTEC-IPN

Martín S. Domínguez González
Claudia Lara Vivas
Manuel Sáenz Sánchez
Raymundo Téllez Cortéz
Alumnos de la Maestría del CINTEC-IPN

Este artículo tiene tres grandes bloques de trabajo, **análisis, diseño, y construcción** de un sistema de control digital directo de un motor de C.D. En el análisis y diseño del sistema se emplearon la transformaciones **s**, **z** y **w**, para representar los modelos del motor, controlador, convertidores y tacómetro; el algoritmo empleado para este proceso es uno de tipo Proporcional Integral y Derivativo (P.I.D. por sus iniciales). Los criterios de estabilidad usados fueron: Jury, Nyquist, Bode, eliminación de polos y respuesta en el tiempo, para obtener el rango de valores de las ganancias del controlador. En la construcción se empleó una computadora personal IPN-E32 para instalar el algoritmo de control y para monitorear evoluciones de variables de interés en tiempo real, dicho algoritmo fue estructurado en lenguaje Turbo C; además se manejó una interface electrónica entre la computadora y el motor de C.D., conteniendo un puerto paralelo, convertidores A/D, D/A, F/V, amplificador de corriente; y un motor de C.D. de 24V, 1A y 3000 rpm.

Introducción

Durante la última década, el empleo de una computadora digital como dispositivo compensador a aumentado, ya que su precio y confiabilidad han mejorado notablemente. Debido esto, actualmente los sistemas de control digital se emplean en múltiples aplicaciones: para máquinas herramienta, procesos metalúrgicos, procesos químicos, control de aviones, etc.

Las ventajas de usar control digital se reflejan en una mayor sensibilidad y una gran inmunidad a la distorsión de las señales, básicamente. La sensibilidad se consigue gracias al empleo de señales de baja energía, mientras que la inmunidad a la distorsión causada por el ruido y a las no linealidades, se obtiene gracias al buen acoplamiento de estas señales en los dispositivos digitales empleados.

La principal desventaja de usar control digital es que la precisión de un computador y de los convertidores asociados está limitada por una longitud de palabra finita; así mismo, los convertidores (A/D y D/A) introducen un error de cuantificación de amplitud. Cuando el error de cuantificación y el error debido al tamaño finito de palabra del computador son pequeños en relación con la amplitud de la señal, entonces el sistema es suficiente-

mente preciso y se pueden ignorar estas limitaciones. [1,2]

Análisis y Diseño de Control

El sistema cuenta con dos partes principales: el computador y la interface de potencia del motor. El computador ocupa una tarjeta convertidora A/D - D/A [6] controlada mediante un programa, el cual toma un valor analógico en la tarjeta de potencia (voltaje obtenido por el tacómetro), lo procesa y entrega un valor analógico, el cual excita a la etapa de potencia del motor [4,5]. La interface de potencia cuenta con un amplificador de potencia, y un convertidor de frecuencia a voltaje (F/V), éste último dará el voltaje que se introducirá al A/D, en relación a la velocidad del motor. El amplificador tiene una ganancia que producirá el voltaje necesario para el motor (0-24V), a partir del voltaje que proporciona el D/A (0-5V), [4,5]. En la **figura 1** se observa un diagrama general del lazo de control.

Función de Transferencia del Sistema

El sistema analizado se describe en la **figura 2**, la cual contempla todas y cada una de las partes y las funciones de control que involucra cada sección. Para efectuar el análisis es necesario encontrar la fun-

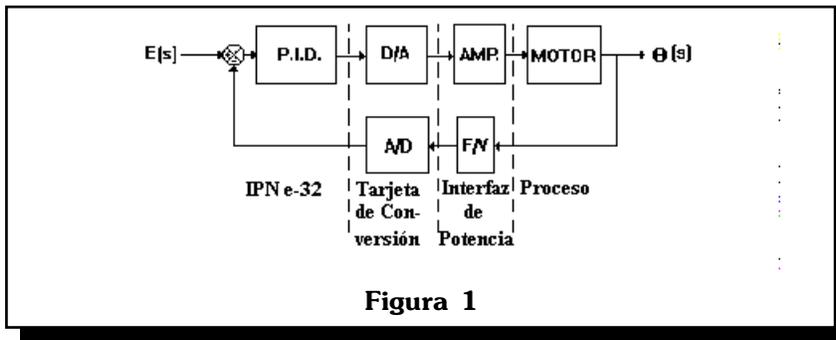


Figura 1

ción de transferencia del sistema, sin controlador P.I.D., y en lazo abierto. Empleando $K_1=4.8$, $K_m=2105$, $a=80$ y $T=1$ mseg, de acuerdo a la referencia [5], se obtienen las ecuaciones de la **figura 3**.

Observando la ecuación se aprecia un polo en $z=1$; por lo tanto, el error en estado estacionario a lazo cerrado será igual a cero. El siguiente paso será calcular los valores de las constantes K_p , K_i y K_d , para lo cual se igualará el numerador de la función de transferencia del control P.I.D. con el denominador de la función a lazo abierto [3]. Esto puede apreciarse en la **figura 4**.

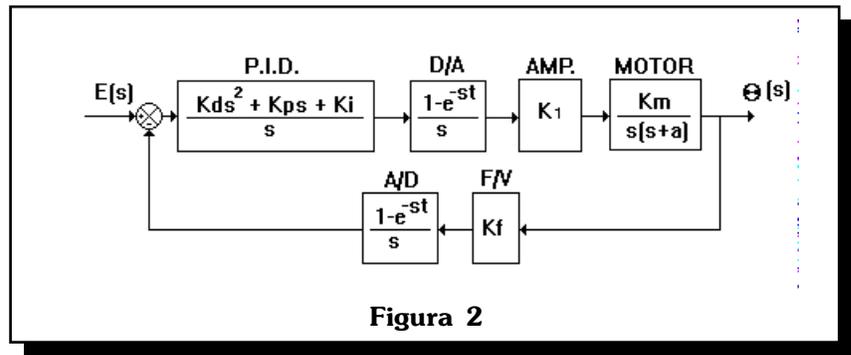


Figura 2

6. A esta ecuación se aplican los criterios de estabilidad. Variando los valores de K_p , K_i y K_d , se tienen ecuaciones iguales en estructura a la ecuación 14, pero con diferentes

gráficas de respuesta. Al final de las pruebas, se obtienen los valores siguientes:

$$\frac{\Theta[z]}{E[z]} = \frac{0.5(z+0.974)}{z - 0.9963z + 0.0036} \quad - (15)$$

$$K_p = 7.854$$

$$K_i = 0$$

$$K_d = 0.094$$

Prueba de estabilidad de Nyquist, Bode y Jury

El análisis de sistemas de control muestreados se realiza bajo el dominio del tiempo y la frecuencia. Los métodos más apropiados para el análisis de estos sistemas son:

1. **Dominio de la frecuencia.**- Método de transformada s (Transformada de Laplace), respuesta en frecuencia, transformada z y transformada w (transformación bilineal).

$$G(s) = (1 - e^{-st}) \frac{K_1 K_m}{s^2 (s+a)} \quad - (1)$$

$$G(s) = (1 - e^{-st}) \frac{10104}{s^2 (s+80)} \quad - (2)$$

$$G(z) = \frac{0.0049(z+0.974)}{(z-1)(z-0.923)} \quad - (3)$$

Figura 3

De la ecuación [7], se obtienen los valores para K_p , K_i y K_d , **figura 5**.

Una vez obtenidos los valores, la función de transferencia del sistema compensado a lazo cerrado se obtiene substituyendo dichos datos en las ecuaciones [11], [12], [13] y [14], como se observa en la **figura**

$$G(s) = \frac{K_d s^2 + K_p s + K_i}{s} \quad - (4)$$

$$G(z) = \frac{K_d [z-1]}{T[z]} + K_p + \frac{K_i T [z+1]}{2[z-1]} \quad - (5)$$

$$G(z) = \frac{[K_i T^2 + 2K_p T + 2K_d]z^2 + [K_i T^2 - 2K_p T - 4K_d]z + 2K_d}{2Tz[z-1]} \quad - (6)$$

$$[K_i T^2 + 2K_p T + 2K_d]z^2 + [K_i T^2 - 2K_p T - 4K_d]z + 2K_d = [z-1][z-0.923] \quad - (7)$$

Figura 4

$$\begin{aligned}
 K_i T^2 + 2K_p T + 2K_d &= 1 & - (8) \\
 K_i T^2 - 2K_p T - 4K_d &= -1.923 & - (9) \\
 2K_d &= 0.923 & - (10) \\
 K_d &= 0.462 \\
 K_p &= 38.5 \\
 K_i &= 0
 \end{aligned}$$

Figura 5

2. Dominio del tiempo.- Método de ecuación de diferencias y respuesta impulso.

Estos métodos se utilizan para analizar el comportamiento del sistema en el dominio de la frecuencia y determinar si éste cumple con las condiciones necesarias de estabilidad.

Un sistema de control con retroalimentación lineal continuo es estable si todos los polos de la función de transferencia de circuito cerrado están en la mitad izquierda del plano *s*; el plano *z* está relacionado con el plano *s* por la transformación:

$$z = e^{sT}$$

$$z = e^{(s + j\omega)T}$$

Esta relación se puede escribir también como:

$$|z| = e^{\sigma T}$$

$$z = wT$$

En la parte izquierda del plano *s* se tiene $s < 0$, por lo tanto, la magnitud correspondiente de *z* varía de 0 a 1. Por esta razón, el eje imaginario del plano *s* corresponde al círculo unitario en el plano *z* y la parte interna del mismo círculo corresponde a la mitad izquierda del plano *s*. En consecuencia, se puede establecer que un sistema mues-

trado es estable si todos los polos de la función de transferencia de circuito cerrado están dentro del círculo unitario (figura 7).

El diagrama de Nyquist se muestra en la figura 8; una manera de entenderlo mejor es por medio del diagrama de Bode, el cual nos muestra el margen de fase y el margen de ganancia del sistema. Para nuestro caso, el margen de fase es aproximadamente 39 grados, y el margen

$$\begin{aligned}
 G'(z) &= \frac{2.45(z+0.974)}{z(z-1)} & - (11) \\
 H(z) &= 0.0074 & - (12) \\
 \frac{\Theta(z)}{E(z)} &= \frac{G'(z)}{1+G'(z)H(z)} & - (13) \\
 \frac{\Theta(z)}{E(z)} &= \frac{2.45(z+0.974)}{z - 0.982z + 0.0175} & - (14)
 \end{aligned}$$

Figura 6

trado es estable si todos los polos de la función de transferencia de circuito cerrado están dentro del círculo unitario (figura 7).

El método de estabilidad de Nyquist establece que al graficar la función de transferencia en el plano *z*, el lugar geométrico resultante no debe encerrar al punto -1,0, ya que esto implica la existencia de polos o ceros en el lado derecho del

de ganancia es 6.4 dB; los parámetros anteriores nos dan una idea de la estabilidad relativa del sistema. El diagrama de Bode se muestra en la figura 9.

Por último se realiza la prueba de estabilidad de Jury, que por medios algebraicos muestra si existe o no estabilidad en el sistema.

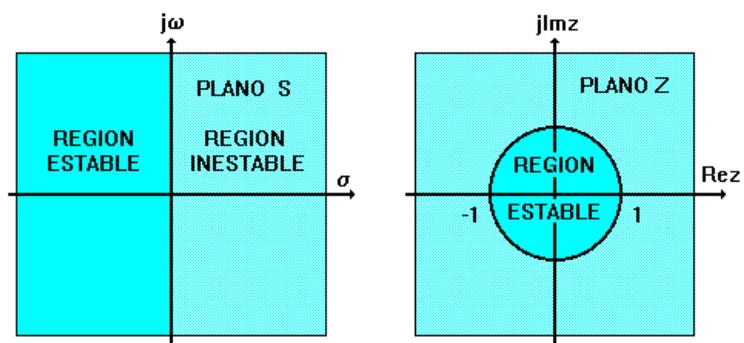
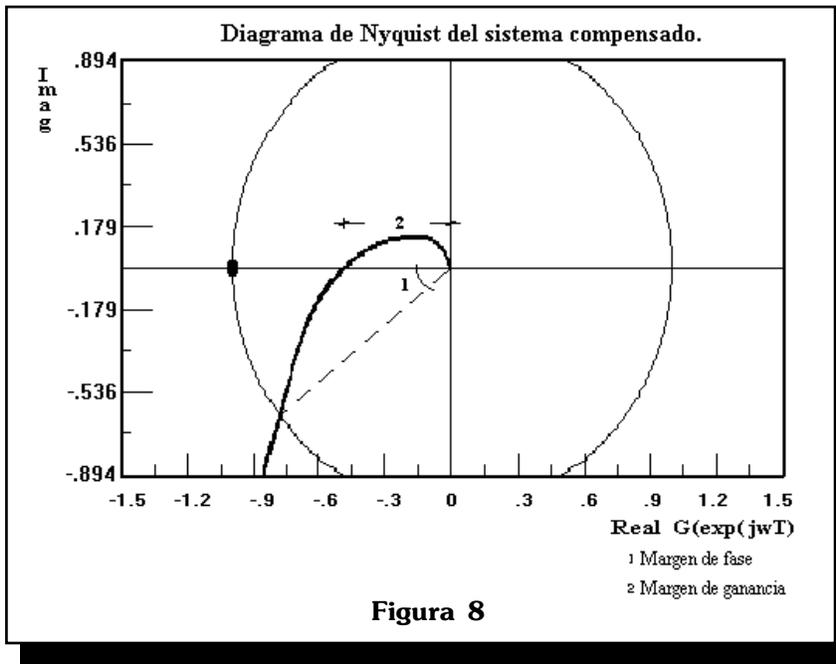


Figura 7



De la Ec. [15] se deduce la ecuación característica :

$$F(z) = z^2 - 0.9963z + 0.0036$$

1a. Prueba de Jury

$$F(1) > 0$$

$$(1)^2 - 0.9963(1) + 0.0036 > 0$$

$$0.0073 > 0$$

2a. Prueba de Jury

$$(-1)^n F(-1) > 0 \quad \{n = \text{grado de la ecuación}\}$$

$$(-1)^2 [(-1)^2 - 0.9963(-1) + 0.0036] > 0$$

$$1.9999 > 0$$

3a. Prueba de Jury

z^0	z^1	z^2
0.0036	-0.9963	1

$$|a_0| < a_2$$

$$|0.0036| < 1$$

Por lo tanto, según el criterio de Jury, el sistema controlado por el

P.I.D. es estable. La gráfica de respuesta en el tiempo del sistema incluyendo el controlador P.I.D. se muestra en la **figura 10**. Jugando con los valores de las constantes se puede obtener otro tipo de respuesta, ya sea con mayor o menor amortiguación

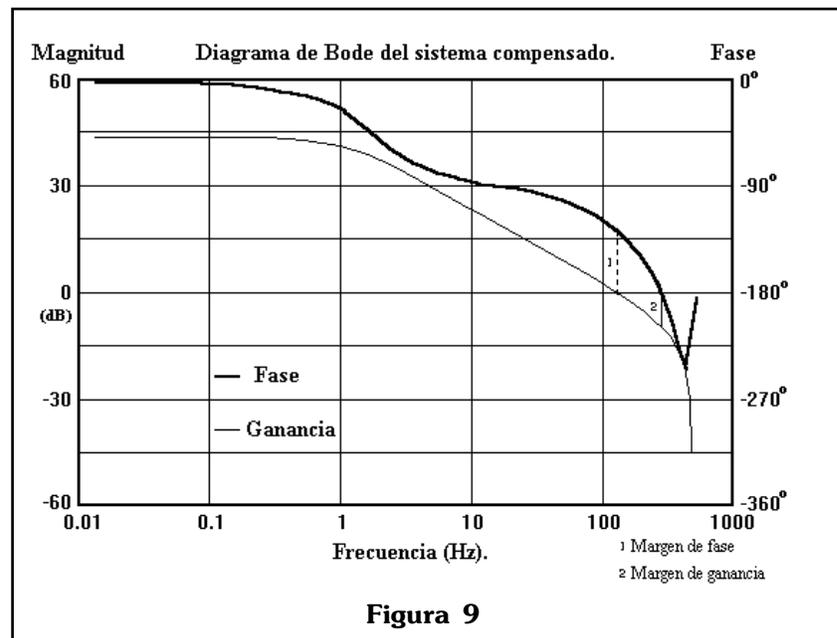
Diseño y Construcción

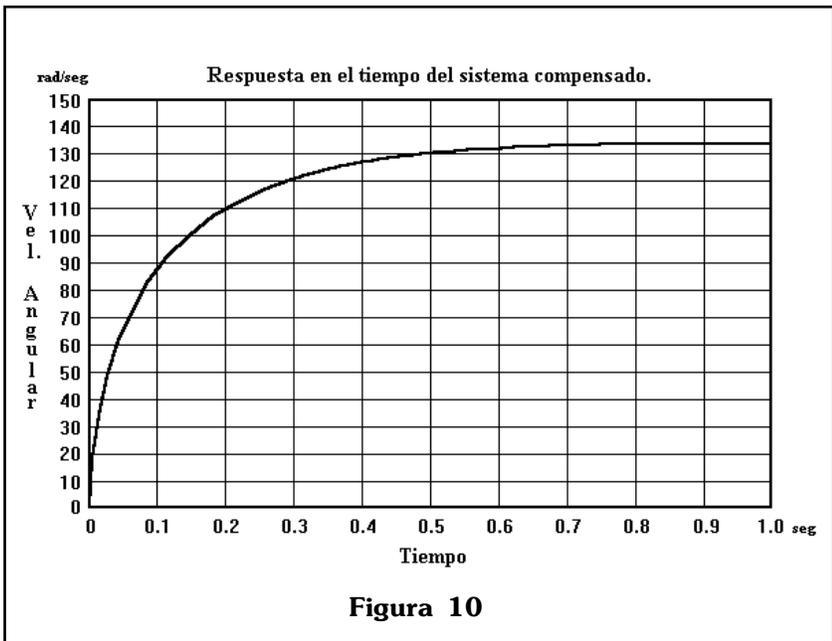
El sistema controlador se basa en el empleo de la computadora IPN-E32 con un algoritmo de control P.I.D., para conexión con el motor, recepción y envío de datos con convertidores A/D, D/A, F/V; también cuenta con un amplificador de corriente y un motor de C.D. de 24V, 1 A.

Los convertidores A/D, D/A se encuentran conectados en la tarjeta instalada en el bus de la computadora; mientras que el F/V (convertidor frecuencia-voltaje) con el amplificador están en la tarjeta de interface de potencia [4], ver **figura 11**. La descripción de la tarjeta "interface de potencia", diagrama esquemático y el programa realizado para realizar el control se detallan a continuación.

Descripción del Circuito

El circuito consta de una fuente de alimentación propia, que pro-





proporciona los siguientes voltajes: +24V, -24V, +15V, -15V, +10V y +5V; con el fin de hacer el sistema autónomo los dos primeros proporcionan hasta 3A, que alimentan a un amplificador operacional de potencia (LM675); los dos siguientes a un amplificador operacional (LM1458), mientras que con los 10V se alimenta al convertidor de frecuencia a voltaje (LM2907) y 5V al circuito que está integrado al servomotor. La fuente de alimentación está basada en los reguladores 78XX, 79XX, diodos zener y arreglo de transistores, como se aprecia en el diagrama esquemático.

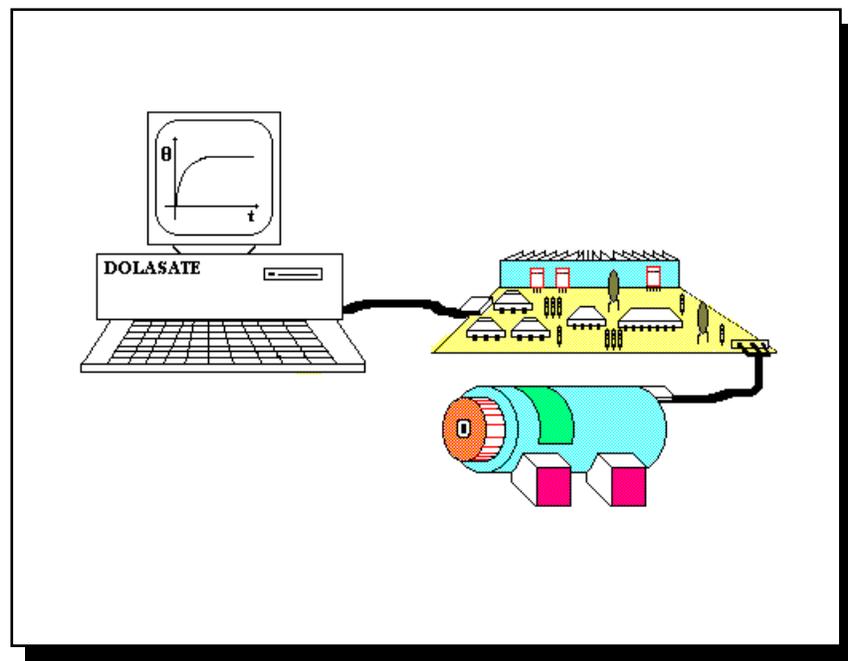
La tarjeta cuenta con un F/V (LM2907) el cual se ajustó (según formulas del manual de National Semiconductor) para permitir en la entrada frecuencias de 1600 Hz. a 8140 Hz., proporcionando a la salida un rango de 0.96V a 4.8 V. La salida del F/V está conectada a un amplificador operacional LM 1458 como seguidor de voltaje, con el fin de presentar una impedancia de salida más baja (75 ohms) al A/D, que es implementado por el LM2907. Lo anterior permite que

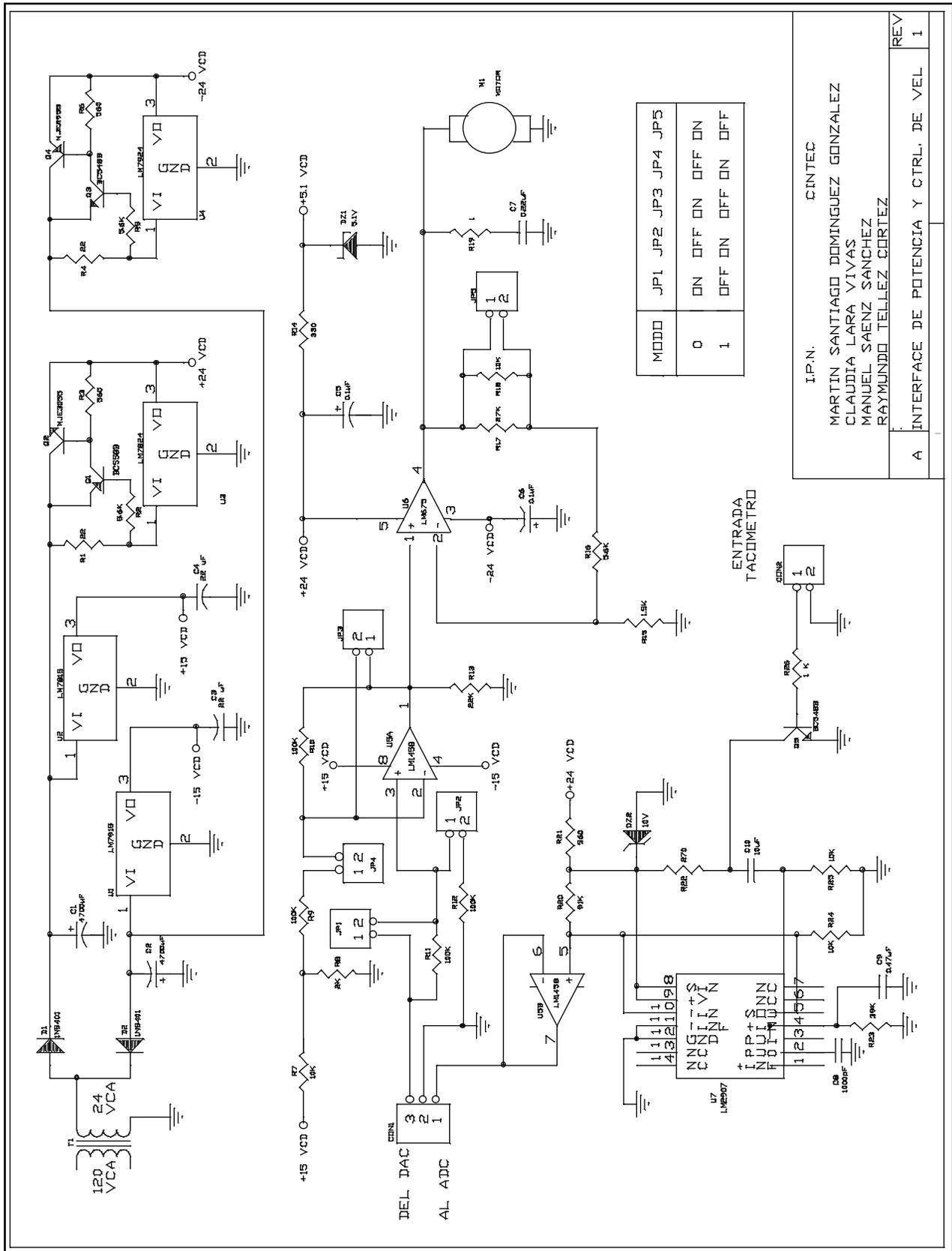
estos voltajes puedan introducirse al A/D; de esta manera se envía a la computadora un dato digital, el cual indica la velocidad a la que gira el motor.

Una vez procesado el dato por el algoritmo P.I.D. (como puede verse en el programa); en el D/A se tiene una señal analógica que entra a un amplificador operacional

LM1458 como seguidor de voltaje, con el fin de presentar ante el D/A una impedancia de entrada alta. La salida del seguidor de voltaje se introduce al amplificador operacional de potencia LM675, el cual tiene un factor de amplificación de 4.74 (práctico), por lo que es posible alimentar al motor hasta con 21V aproximadamente (teóricamente sería hasta 24V, pero el amplificador operacional no puede proporcionar este voltaje a su salida, debido a que está siendo polarizado con 24V). El circuito está provisto para ser configurado de otra forma por medio de "puentes" y así ofrecer el giro en ambos sentidos. El LM1458 está configurado como sumador de dos voltajes: V_{in} y $-2.5V$; como V_{in} sólo puede variar de 0V a 5V, a la salida del sumador tendremos variaciones de $-2.5V$ a $2.5V$, los cuales permiten que el amplificador de potencia LM675 (con otro factor de amplificación: 9.6) pueda entregar voltajes positivos y negativos (± 24 volts) al motor.

Posteriormente, teniéndose una referencia (que será el valor espera-





do en la entrada del A/D) que indique la velocidad con que gira el motor (proporcionada por el F/V), se ejecuta el algoritmo P.I.D., y mediante el D/A se generará un voltaje que llegará al motor por medio de la interface de potencia, el cual alcanza y estabiliza la velocidad deseada, indicada en la referencia.

A continuación se tiene el listado del programa empleado en el sistema :

Prueba y Ajustes

Para lograr la implantación del sistema se probaron las partes que lo conforman de manera individual. Cabe resaltar que en la tarjeta de "interface de potencia" se realizaron algunas pruebas y ajustes, tomándose en cuenta las características de la tarjeta de los convertidores A/D y D/A, y las necesidades de alimentación del motor de C.D.;

se sabe que a la salida del D/A se tienen voltajes de 0V a 5V, por lo que fue necesario aplicar un factor de ganancia de 4.8 al amplificador, para lograr alimentar al motor en un rango de 0V a 24V. Dependiendo de esta alimentación, el motor produce una frecuencia de giro, la cual, mediante el FVC, deberá producir voltajes de 0 a 5 volts que alimentan al ADC; esto último se logra configurando el circuito integrado LM2907 según fórmulas del manual de National Semiconductor.

Conclusiones

En el análisis y diseño del sistema de control se pudo representar a éste en una función de transferencia en transformadas *s*, *z*, *w*; las pruebas (simulación) de estabilidad del sistema de control se aplicaron en conjunto para obtener un panorama más claro y confiable de funcionamiento (rangos de valores Kp, Ki, Kd). Aunque estas pruebas dieron un amplio rango de estabilidad, al llevarlos a la práctica (tiempo real) se tuvieron que realizar algunos ajustes (como por ejemplo, eliminar los sobretiros, y obtener un seguimiento del controlador igual a la del proceso, escogiendo un solo valor óptimo para cada Kp, Ki, y Kd) con lo cual se consiguió que el motor corriera perfectamente. También se pudo comprobar que algoritmo P.I.D. es ligeramente mejor que el P.I.

La construcción del sistema de control digital se llevó a cabo con tecnología elaborada y adquirida en el mercado nacional. Se consiguió capacitar recursos humanos así como utilizar materiales nacionales propios, propiciando con esto último ahorros económicos e independencia tecnológica en esta área. Finalmente, se pudo observar que

```

/*
                INSTITUTO POLITECNICO NACIONAL
                CENTRO DE INVESTIGACION TECNOLOGICA EN COMPUTACION

TEORIA DE CONTROL

PROFESORES:
    M. EN C. ROMEO URBIETA PARRAZALEZ
    M. EN C. TEODORO ALVAREZ SANCHEZ
ALUMNOS:
    MARTIN S. DOMINGUEZ GONZALEZ
    CLAUDIA LARA VIVAS
    MANUEL SAENZ SANCHEZ
    RAYMUNDO TELLEZ CORTEZ

                DOLASATE INC.

CONTROL P.I.D:

Programa que completa el diagrama de flujo de un controlador empleando
el algoritmo P.I.D.

El programa acepta un nivel de referencia (velocidad del motor). El
sistema muestra al usuario el nivel alcanzado realmente.

El algoritmo P.I.D. trabaja en base a los valores predefinidos de las
constantes Kp, Ki, y Kd.

*/

#include <STDIO.H>
#include <DOS.H>
#include <CONIO.H>
#include <GRAPHICS.H>

#define PUERTO_A 0X100 /* DIRECCION DEL PUERTO A DEL 8255 */
#define PUERTO_B 0X101 /* DIRECCION DEL PUERTO B DEL 8255 */
#define PUERTO_C 0X102 /* DIRECCION DEL PUERTO C DEL 8255 */
#define CONTROL 0X103 /* DIRECCION DEL PUERTO DE CONTROL DEL 8255 */

#define KP 0 /* CONSTANTE PROPORCIONAL */
#define KI 0.3462 /* CONSTANTE INTEGRAL */
#define KD 1.3849 /* CONSTANTE DERIVATIVA */

void portada(); /* SUBROUTINA DE PRESENTACION */
void inicio(); /* SUBROUTINA DE INICIALIZACION DEL PROCESO */
void pid(); /* SUBROUTINA DE CONTROL P.I.D. */

char DATO; /* VARIABLE PARA LEER UN DATO DEL ADC */

int TECLA; /* VARIABLE QUE REGRESA EL CODIGO DE LA TECLA PRESIONADA
POR EL USUARIO */
    
```

```

float IN0 = 0, /* VALOR DE LA ENTRADA EN UN INSTANTE t
*/
IN1 = 0, /* VALOR DE LA ENTRADA EN UN INSTANTE t-1
*/
IN2 = 0, /* VALOR DE LA ENTRADA EN UN INSTANTE t-2
*/

OUT0 = 0, /* VALOR DE LA SALIDA EN UN INSTANTE t */
OUT1 = 0, /* VALOR DE LA SALIDA EN UN INSTANTE t-1 */

REF = 3, /* VALOR DE LA REFERENCIA (EN VOLTS) */
PASO = 3, /* VARIABLE DE PASO PARA LA REFERENCIA */
RETRO ; /* VALOR LEIDO DEL ADC */

/* ----- PROGRAMA PRINCIPAL ----- */
void main(void)
{
portada();
inicio();
pid();
}

/* ----- PRESENTACION ----- */
void portada()
{
int gd,
gm,
tam_x; /* Tamaños de letras a utilizar */

/* Inicializamos gráficos */
gd=DETECT;
initgraph(&gd,&gm,"c:\\tc\\bgi");

/* Encuentra el tamaño óptimo para las letras */
tam_x=(getmaxx()/4)/25;

/* Escribe el título */
settextstyle(TRIPLEX_FONT,HORIZ_DIR,tam_x);
outtextxy(getmaxx()/3,getmaxy()/4,"CONTROL");
settextstyle(TRIPLEX_FONT,HORIZ_DIR,tam_x);
outtextxy((15*getmaxx())/36,getmaxy()/2,"P.I.D.");
settextstyle(TRIPLEX_FONT,HORIZ_DIR,2);
outtextxy(getmaxx()-190,getmaxy()-30,"by DOLASATE INC.");

/* Cerramos gráficos */
getch();
}

/* ----- INICIALIZACION ----- */
void inicio()
{
/*
CONFIGURAMOS EL 8255:
PUERTO A SALIDA
PUERTO B ENTRADA
PUERTO C SALIDA
*/
outportb(CONTROL,0X82);

/* MANDAMOS 0 VOLTS AL DAC0 (FRENAMOS EL MOTOR) */
outportb(PUERTO_A,0x00);

/*
CAPTURAMOS EL VALOR:
PC5 1 LOGICO
PC6 1 LOGICO
*/
outportb(CONTROL,0X0B);
outportb(CONTROL,0X0D);

/*
DESACTIVAMOS EL DAC0

```

el circuito realizado “interface de potencia”, puede tomarse como base para implantar otros algoritmos de control y comprobar los desarrollos teóricos, definiendo así si los sistemas lineales son estables, con perturbaciones suaves y bruscas en la carga del motor

La prueba del sistema de control digital se llevó a cabo mediante el programa estructurado en lenguaje Turbo C, dicho programa, de creación propia, fue instalado en la memoria de una PC (IPN-E32). Aquí, la computadora fue utilizada como elemento fundamental de control y como una poderosa herramienta de monitoreo.

Bibliografía

- [1] DORF, Richard. *Sistemas Modernos de Control*. Addison-Wesley Iberoamericana, 2a. edición. México, 1989.
- [2] HOUPIS, Constantine. *Digital Control Systems*. McGraw-Hill, 2a. edición. Singapore, 1992.
- [3] KUO, Benjamin C. *Digital Control System*. Halt, Rinehart and Winston, Inc. 1980.
- [4] ALVAREZ, Teodoro. “Diseño y construcción de un sistema dedicado al control de motores de C.D. basado en el microcontrolador 8031 de INTEL.”. POLIBITS. Año IV, Vol.1, Num.10, Octubre-Diciembre 1992.
- [5] PARRAZALES, Romeo. “Simulación de la posición de un motor de C.D. de 1/15 de H.P controlado por un

```

PC5  0 LOGICO
PC6  0 LOGICO
*/
outportb(CONTROL,0X0A);
outportb(CONTROL,0X0C);
getche();
closegraph();
clrscr();

gotoxy(8,5);
printf("VEL. REQUERIDA");
gotoxy(28,5);
printf("VEL. ALCANZADA");
gotoxy(13,10);
printf("CONTROL DE VELOCIDAD");
}

/* ----- ALGORITMO ----- */
void pid()
{
for (;;){
/*
CONFIGURAMOS EL 8255:
PUERTO A     SALIDA
PUERTO B     ENTRADA
PUERTO C     SALIDA
*/
outportb(CONTROL,0X82);

/*
SE SELECCIONA LA INT0 DEL ADC COMO ENTRADA ANALOGICA:
PC10 LOGICO
PC20 LOGICO
PC30 LOGICO
*/
outportb(CONTROL,0X02);
outportb(CONTROL,0X04);
outportb(CONTROL,0X06);

/*
SE INICIA LA CONVERSION:
PC41 LOGICO
PC40 LOGICO
*/
outportb(CONTROL,0X09);
outportb(CONTROL,0X08);

/*
SE CONFIGURA 8255:
PUERTO A
PUERTO B
PUERTO C
*/
outportb(CONTROL,0X83);

/* ESPERAMOS A QUE TERMINE LA CONVERSION */
do{
DATO=inportb(PUERTO_C);
}while(DATO & 1);

/* LEEMOS UN DATO DEL ADC-IN0 (RETROALIMENTACION) */
DATO=inportb(PUERTO_B);

RETRO=(5.0*DATO)/255;

gotoxy(30,7);
printf("%.3f", (RETRO*657)/5);

/* REALIZAMOS EL CONTROL PID */
IN0=REF-RETRO;

OUT0=((KP+KI+KD)*IN0) - ((KP+(2*KD))*IN1) + (KD*IN2) + OUT1;

```

algoritmo PI usando la microcomputadora IPNe16-m." POLIBITS . Año IV, Vol.1,Num.9, Julio-Septiembre 1992.

- [6] PARRAZALES, Romeo. "Diseño y construcción de una tarjeta convertidora de 8 canales de A/D y 4 canales de D/A". POLIBITS, Año VI Vol. Num. 12. Enero-Marzo 1994.

```
IN2=IN1;
IN1=IN0;
OUT1=OUT0;

gotoxy(10,7);
printf("%3.3f", (OUT0*657)/5);

/* ESCRIBIMOS LA SALIDA DEL CONTROL AL DAC */
outportb(PUERTO_A, (unsigned char) OUT0);

outportb(CONTROL, 0X0B);
outportb(CONTROL, 0X0D);

outportb(CONTROL, 0X0A);
outportb(CONTROL, 0X0C);

/*
  ACEPTAMOS LAS TECLAS DE CURSOR
  ARRIBA AUMENTAMOS LA VARIABLE DE PASO
  ABAJO DISMINUIMOS LA VARIABLE DE PASO
  ENTER ACTUALIZAMOS EL VALOR DE LA REFERENCIA
  ESC SALIMOS DEL PROGRAMA
*/
if(kbhit())
{
  switch(getch()){
    case 72:
      if(PASO<=4.78) PASO+=0.02;
      gotoxy(20,12);
      printf("%1.3f", (PASO*675)/5);
      break;
    case 80:
      if(PASO>=1.02) PASO-=0.02;
      gotoxy(20,12);
      printf("%1.3f", (PASO*675)/5);
      break;
    case 13:
      REF=PASO;
      break;
    case 27:
      /* PARAMOS EL MOTOR */
      outportb(PUERTO_A, 0x00);

      outportb(CONTROL, 0X0B);
      outportb(CONTROL, 0X0D);

      outportb(CONTROL, 0X0A);
      outportb(CONTROL, 0X0C);
      exit();
    default:
      break;
  }
}
```