

SERVOAMPLIFICADOR PARA MOTORES DE DC.

Proyecto fin de carrera presentado por:

Juan Antonio Pizarro Martín

Julio Martín Rodríguez

Ángel Puerta Rubio

CENTRO DE ESTUDIOS SUPERIORES SALESIANOS LOYOLA

SISTEMA DE CONTROL DE MOTORES DE D.C BASADO EN EL
MICROCONTROLADOR LM629.

PROYECTO

PARA LA OBTENCION DEL TITULO DE BACHILLERATO MÁS 4

CNAM FRANCIA

- 1-. Juan Antonio Pizarro Martín.**
- 2-. Julio Martín Rodríguez.**
- 3-. Ángel Puerta Rubio.**

ARANJUEZ

CURSO 2001/2002

DOCUMENTOS

1 Justificación.

2 Memoria.

3 Presupuesto.

4 Pliego de condiciones.

5 Planos (Esquemas).

6 Anexos.

DOCUMENTOS N° 1

JUSTIFICACIÓN

INDICE

1 Justificación.

2 Objetivos.

3 Descripción.

4 Reglamentación.

1. Justificación.

El presente documento describe el trabajo realizado para el diseño de un variador - posicionador digital para el control de motores de corriente continua, con la suficiente robustez para ser capaz de controlar una amplia gama de motores de diferentes potencias.

2. Objetivos.

Su destino podría ser el control de un motor para un robot de cadena de montaje, de maniobras preprogramadas, o cualquier aplicación en que se requiera un determinado número de movimientos precisos en cuanto a posición, velocidad y tratamiento “inteligente” de su ruta.

3. Descripción.

En una primera aproximación, obtenemos la siguiente descripción:

- El equipo necesita atender las siguientes características:
 - √ Todos los parámetros del equipo deben ser programados mediante un terminal RS232 y un software desarrollado en exclusiva.
 - √ Parámetros de posicionado: Tiempo de aceleración, desaceleración, velocidad, ángulo de giro, posicionado absoluto, relativo, inicial.
 - √ Parámetros generales: Límite de corriente, de velocidad; regulación automática.
- Las condiciones de trabajo del equipo van a ser:
 - √ La alimentación de potencia puede variar entre 10 y 65 V_{DC} con una corriente máxima de 8 A y de 4 A. en funcionamiento normal.
 - √ Para alimentar el control se dispone de una línea industrial de corriente continua de 24 V_{DC}.
 - √ Es necesario que el equipo se proteja de sobrecalentamientos, cortocircuitos y bloqueos. Además interesa guardar datos de parámetros, porque el equipo va a ser autónomo, careciendo de vigilancia específica.
 - √ Va a trabajar en condiciones ambientales que no superarán los 60 °C, pero no dispone de ventilación especial.

Con los datos aportados se elabora una primera especificación:

SERVOAMPLIFICADOR para MOTORES DC		
ESPECIFICACIÓN PRELIMINAR		
Alimentación de potencia:	10 ÷ 65	V _{DC}
Corriente nominal:	4	A.
Corriente máxima:	8	A.
Interconexionado:		
Entradas lógicas de control:	24	V _{DC}
Aislamiento control / potencia:	OPTOACOPLADO	
Indicadores de disfunción:	Relé	
Control por "encoder":	TTL	
Diagnóstico de errores:		
Temperatura, corriente seguimiento:	LED's, RS232	
Tipo de control:		
Potencia: PWM	17 Khz	
Regulación automática:	Digital	PI
Almacenamiento de parámetros:	Windows software	
Almacenamiento de datos:	EEPROM	
Temperatura de funcionamiento:	0 ÷ 70 °C	
Enfriamiento:	Convección natural	
Dimensiones:	Formato 6U	
Conformidad con las normas CE: Certificaciones.		
EN 60204-1		
CEI 1000-4-2,3,4,5 de acuerdo con IEC 1800-3		
EN 55011 Grupo 1 de acuerdo con IEC 1800-3		

Hay que tener en cuenta que, en la actualidad, el diseño del tipo de control que se ha seleccionado, y que será descrito con más detalle, ha llegado a tal punto de integración que puede ser resuelto en su totalidad con circuitos integrados específicos, incluyendo: Un microcontrolador (μC), el control del motor (PID), la etapa de potencia (puente H) y muy pocos componentes lógicos.

En estas circunstancias, y por la índole del presente proyecto, se ha optado, junto con una descripción teórica del sistema en su conjunto, por el diseño de un puente de potencia de topología “Full Bridge” (Puente completo), habitualmente nombrado Puente H, con componentes de control y discretos con el objetivo de aumentar la capacidad de admitir motores de potencias superiores a las que se pueden lograr con la simple integración. Esto además viene a realzar la función didáctica del proyecto.

Sin embargo, el control del propio puente H lo llevará a cabo un circuito integrado específico que responde a la norma de control PID, que a su vez será comandado por un micro controlador de propósito general, dado que el diseño de este tipo de circuitos, además de ampliamente estudiado, resulta engorroso y actualmente inútil.

Una primera división nos permite distribuir el conjunto en tres apartados con misiones específicas y complementarias, cuya descripción se hará por separado, y cuya construcción puede ser perfectamente aislada, con objeto de hacer un reparto claro de tareas:

- a) El puente H, que ha de ser dimensionado para poder suministrar al motor la potencia precisa y en el grado adecuado para que, estando el motor protegido, tampoco el propio puente sufra ningún daño. Incluye la conexión (“drivers”) con el control PID, que va a enviar la modulación por ancho de pulso (PWM) que controlará la corriente por el motor de carga, y la señal de retorno que facilitará un “encoder” montado sobre el rotor del motor y que vendrá realimentada al PID, para que éste efectúe su trabajo.
- b) El “software” que va a decir al μC , y a través de éste al PID, qué debe hacer el motor, y en qué forma, además de gestionar una serie de situaciones irregulares a las que deberá aportar las soluciones adecuadas. Deberá almacenar en forma recuperable información del trabajo efectuado para el caso en que el sistema deba trabajar en forma autónoma largos periodos de tiempo, y sea preciso conocer o estudiar su comportamiento.
- c) La lógica que va a interconectar el μC con el PID, mediante el cálculo de los tiempos que requiera el control del bus (“timing”), con sus circuitos asociados, y los circuitos analógicos con salida digital que informarán al μC de las incidencias o mal funcionamiento en el circuito de potencia con relación a la corriente que circula por el motor, de pico o promedio, y la temperatura del circuito de potencia, así como la indicación codificada de este tipo de eventos con un “display” numérico “a pie de máquina” y con un elemento de desconexión manual rápida.

4. Reglamentación.

En la reglamentación de este proyecto se han tenido en cuenta todas y cada una de las especificaciones contenidas en :

- Conformidad con las normas CE: certificaciones.
 - EN 60204-1.
 - CEI 1000-4-2,3,4,5 de acuerdo con IEC 1800-3.
 - EN 55011 Grupo 1 de acuerdo con IEC 1800-3.
 - Normalización Nacional (normas UNE).

DOCUMENTOS N° 2

MEMORIA

INDICE

1 Controlador PID.

2 Bloques más importantes del MPID (LM629).

2.1 Control PID.

3 Motores de propósito especial.

4 Hardware del sistema.

4.1 Descripción del sistema de control de velocidad.

4.2 Filtro PID.

4.3 Descripción de la arquitectura del dispositivo MPID.

4.4 Operación de lectura y escritura del MPID.

4.5 Salida motor.

4.6 Descripción de los terminales del MPID.

4.7 Microcontrolador (μ c).

4.8 Descripción de la arquitectura del dispositivo Puente H.

4.9 Formas de señales del PWM.

4.10 Descripción del sistema de alarmas.

4.11 Circuito reloj.

4.12 Reset del sistema.

5 Descripción de dispositivos.

6 Calculo del sistema.

6.1 Calculo del disipador.

6.2 Calculo el AO diferencial.

6.3 Calculo del Filtro Paso Bajo.

7 Construcción de la tarjeta de control.

7.1 Diseño de la tarjeta.

7.2 Diseño del circuito de control.

8 Software del sistema.

8.1 Conexiones del LM629.

8.2 Modulo de programación para control de la tarjeta.

8.3 Modulo de programación del filtro.

9 Tipos de movimiento.

10 Ajuste del filtro PID.

11 Descripción de los programas.

12 Gestión de incidencias.

A continuación se facilita una descripción de los elementos del sistema y sus relaciones, para lograr una visión de conjunto.

1. Controlador PID

Opción seleccionada:

LM629M-8 (SMD) DS009219-21 con 24 contactos, de la casa "National semiconductors".

Justificación.

Para la aplicación que le vamos a dar resulta suficiente y cumple nuestras expectativas un modelo integrado que cumpla la norma del sistema PID para control de motores. En concreto, hemos seleccionado el modelo LM629 de la casa "National semiconductors", por las razones que se exponen a continuación.

Existen otros modelos, como el L6515 de la casa "SGS Thomson", con el sistema que denominan "Twin Loop control", pero no resulta tan completo como el seleccionado, y nos obligaría a utilizar su sistema de realimentación, que requiere mayor circuitería, siendo menos completo en su oferta. Su aportación es un costo más bajo en el caso de que pensáramos en una aplicación autónoma más simple, y con menores exigencias de robustez y versatilidad.

Otros modelos consultados nos remiten a aplicaciones completamente integradas, para motores de más baja potencia de la que requerimos.

El modelo seleccionado, LM629, aporta el algoritmo de filtrado PID a partir de los datos de posición-velocidad que le envía un μC y la realimentación que devuelve un "encoder" incremental situado sobre el rotor del motor. El LM629 recoge del encoder información de posición y velocidad, además de un impulso "índice", de uso opcional, del que carece por ejemplo el L6515, y que tiene su utilidad en la gestión de interrupciones por acumulación de datos "indexados" de trayectoria efectuada.

Este modelo en concreto (a diferencia del LM628 de la misma casa, que aporta una salida digital de 8 bits para su conversión a señal analógica de control) ofrece salida modulada en ancho de pulso (PWM) para el control de la potencia, adecuada para el control de una topología en puente completo (full bridge) de MOSFET, en H, que es la etapa de potencia seleccionada en nuestro trabajo. Se compone de una señal de magnitud de corriente proporcional al ancho del pulso y otra de "signo" que determina el sentido de giro.

Ofrece además un "set" de comandos para el envío de los coeficientes del PID y gestión de interrupciones de alto nivel y sencilla programación.

El control de operaciones (comandos) y transferencia de datos se efectúa a través de un puerto paralelo de 8 bits, que puede ser controlado por un μC situado sobre la misma placa del sistema. Este a su vez puede operar de forma autónoma, sobre secuencias dadas, previa programación, o ser operado a través de puerto serie UART (recepción-transmisión asíncrona universal), con protocolo RS232, desde una CPU remota.

(Por razones de brevedad, no vamos a considerar, en principio, esta última posibilidad, aunque queda abierta su implementación como opción factible y no excesivamente complicada, que podría aportar versatilidad al sistema).

Vamos a ver algunas de las características generales del LM629, aunque en el siguiente apartado veremos en profundidad todos los sistemas de que esta compuesto nuestro MPID.

Encapsulado.

Se presenta en encapsulado DIP (Dual in line) para montaje convencional o superficial (SMD), LM629N o LM629M, con 28 y 24 contactos respectivamente.

El SMD pudiera encarecer el proceso de montaje, pero para el prototipo vamos a emplear SMD por tener menos conexiones, que en el modelo clásico son sobrantes. Para la construcción de una sola unidad, optamos por el SMD puesto que el circuito, salvo sustitución por rotura, no precisará ser desmontado. Si se piensa en serie, sería necesario estudiar costes de fabricación antes de decidir.

Luego seleccionamos el LM629M (SMD) DS009219-21 con 24 contactos, dos de ellos sobrantes.

Frecuencia del reloj.

La frecuencia de reloj (CLK) ofrece dos posibilidades: Hasta 6 o hasta 8 Mhz ($1 < f_{clk} < 8$ Mhz).

En principio se selecciona la más alta al no esperarse dificultad del μC , que trabajará fácilmente con frecuencias de 20 MHz o superiores, aunque queda a expensas de la influencia del reloj sobre la frecuencia de trabajo del PWM para el puente H.

En cualquier caso, tratamos de una aplicación donde puede ser crítica la velocidad de respuesta en control y en atención a interrupciones. En el código se denota con el añadido de la cifra del reloj: LM629M-8.

Alimentación.

Su alimentación es de 5v nominales ($4.5 < V_{DD} < 5.5$), con salidas entradas (I/O) compatibles con tecnología CMOS y HCMOS, y consumo máximo de 605 mW, y no resulta crítica, fuera de las precauciones habituales. Se alimentará, al igual que el resto de la lógica, a través de un convertidor monolítico, con entrada entre 16 y 40 voltios y salida fija de 5 voltios, con potencia sobrada.

Otras consideraciones.

La tolerancia a cargas electrostáticas es de 2000 v, suficientemente alta para una manipulación sin especiales precauciones, y la soldadura en SMD admite 300 °C durante 10 segundos.

Necesidades de "hardware".

Precisará el diseño de la circuitería que haga de "interface" con el μC , la inherente a la línea diferencial de realimentación del "encoder" y la línea de control de potencia del puente H, aislada mediante opto-acopladores de la lógica.

Se va a considerar la posibilidad de aprovechar manualmente la opción de "RESET" por "hardware", si se llega a estimar útil. Se trataría de un pulsador conectado directamente a la patilla RST para activar un "reseteado" manual del LM629; pero está por definir si resulta práctico o conveniente activar manualmente la secuencia que supone el reset.

Sus principales características son:

- 3 registros de 32 bits para la posición, la velocidad y la aceleración.
- Un filtro digital PID programable.
- Una señal de salida PWM de 8 bits.
- Un generador interno de perfiles de velocidad trapezoidales.
- La velocidad, la posición y los parámetros del filtro pueden ser cambiados durante el movimiento.
- Distintos modos de operación en cuanto a posición y a la velocidad.
- Interrupciones programables en tiempo real.
- Capacidad de gestión de señales incrementales en cuadratura.

Sistemas de control del MPID.

El control de tipo proporcional – integral – derivativo (P.I.D.), de naturaleza digital, basado en el procesador de control de movimiento LM 629, es el que permite la programación y tratamiento del movimiento del motor.

Los datos que precisa el PID se le hacen llegar a través de línea de control paralelo, mediante el cálculo de los tiempos (“timing”) de la lógica adecuada para un perfecto sincronismo.

Ésta lo comunica con un MICRO CONTROLADOR (μ C) donde se tiene acceso a programar movimientos y velocidades y gestionar los eventuales problemas mediante interrupciones, originadas por “software” cuando proceden de las informaciones del PID, o por “hardware” si la procedencia es la etapa de potencia que alimenta al motor.

Así mismo, tendrá capacidad para almacenar información de los desplazamientos realizados por el motor, para uso posterior. La programación del μ C será específica para cada aplicación, aunque versátil y reprogramable.
El modelo escogido es un microcontrolador 8051.

La etapa de potencia la forman 4 MOSFET del tipo IRFP 250N montados en topología “H” (full bridge) excitados por “drivers” específicos, del tipo IR2110, que conducen la información para su disparo mediante modulación por ancho de pulso (PWM), facilitada por el PID.

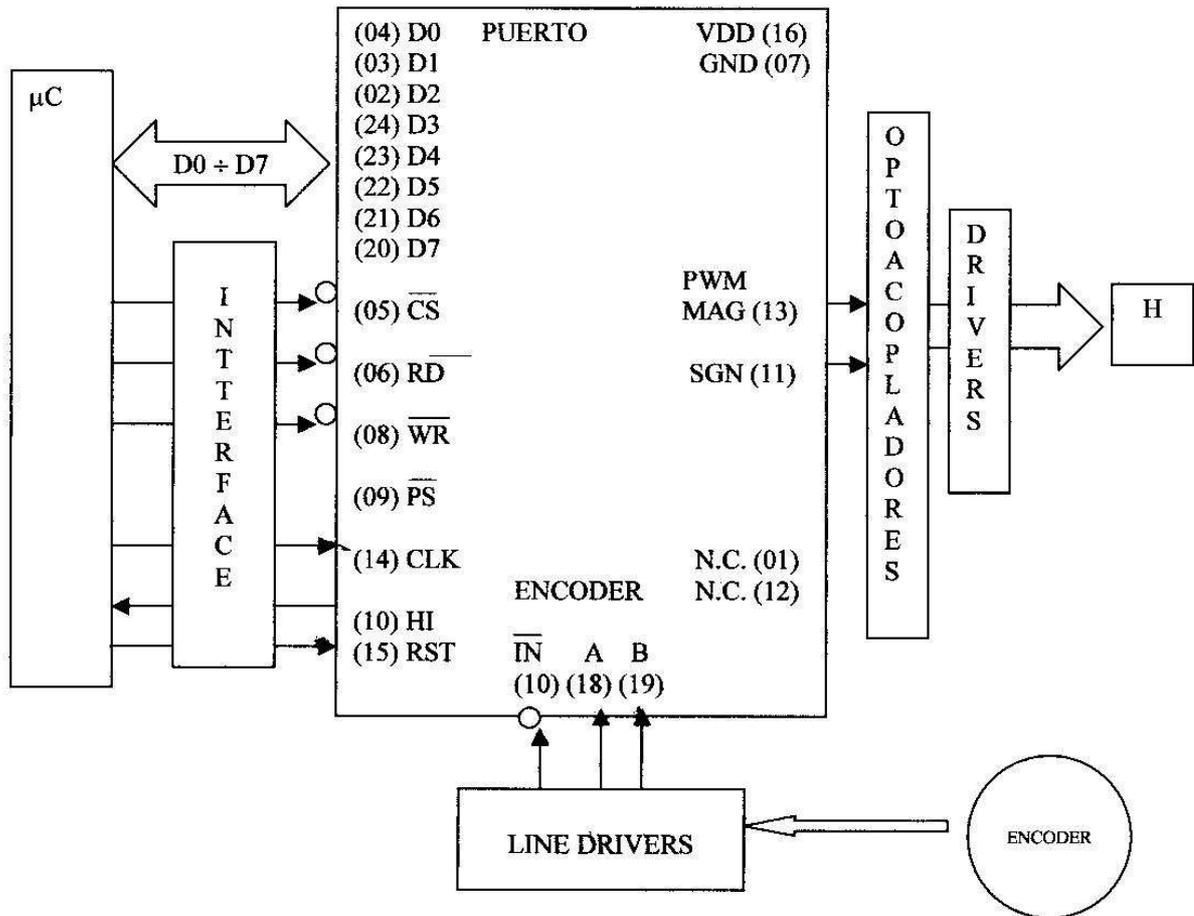
Esta arquitectura en “H” es la adecuada para este tipo de control, dado que permite la regulación de la potencia suministrada al motor y el sentido en que es aplicada a éste.

Esta característica del control de potencia permite que la alimentación para el motor admita una amplia gama de modelos y potencias de motores, dado que la modulación por ancho de pulso es capaz de conmutar los MOSFET en forma que el motor sea excitado por la corriente media adecuada en cada situación.

Las características técnicas se apuntarán en el ANEXO A (hoja de características del fabricante).

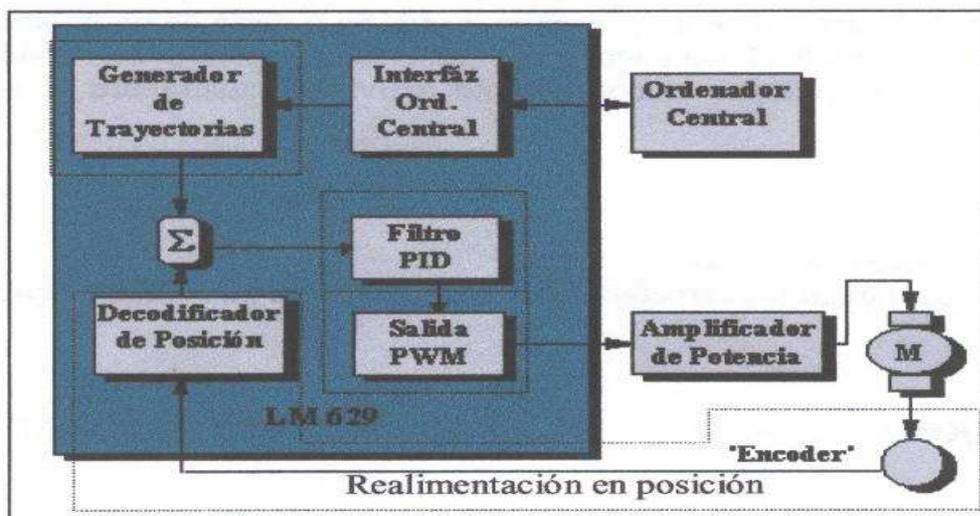
El ANEXO A, estará compuesto por un Cd-Rom, donde estarán incluidos en formato PDF todos los manuales de los distintos dispositivos a utilizar y una relación de carpetas donde se encontraran los Esquemas del circuito y sus correspondientes placas BOARD.

Esquema general del controlador LM 629M-8 y sus periféricos.



Modo de operación.

El diagrama de bloques típico para construir un servo sistema usando un LM629 es el siguiente:



El procesador se comunica con el LM629 a través de un puerto de entrada / salida para facilitar la programación del perfil trapezoidal de velocidad y del filtro PID. Para iniciar el movimiento, el procesador central carga los valores de aceleración, velocidad y posición deseada en el generador de trayectorias del LM629.

En cada período de muestreo estos valores son usados para calcular la nueva posición deseada, a la que se le resta la posición real del eje, obtenida por el codificador óptico incremental y el decodificador de posición. Así se origina la señal de error, que es la entrada al filtro compensador PID.

El convertidor digital-analógico convierte la señal producida a otra aplicable al motor. De nuevo se decodifica la posición y se genera otro perfil de velocidad.

Los cuatro bloques funcionales más importantes del LM629 son:

- a) El generador de trayectorias.
- b) La realimentación en posición.
- c) El filtro digital PID.
- d) La salida PWM.

Después de hacer una breve descripción del trabajo que tenemos que desarrollar, vamos a entrar dentro de alguno de los dispositivos que vamos a utilizar, apuntando una descripción más detallada y profunda de cada uno.

2. Bloques más importantes del MPID anfitrión el LM629.

2.1 Control P.I.D.

El control PID, siglas que denotan control Proporcional – Integral – Derivativo, viene despertando el interés del diseñador desde los primeros desarrollos del control automático, aunque se carecía entonces de la infraestructura digital actual. Desde un punto de vista teórico, la implementación de este tipo de control siempre resultó atractiva y en la actualidad disponemos ya de la tecnología digital avanzada que ha permitido la construcción de los microcontroladores PIDs.

A continuación se expone la norma de control PID.

Norma del control P.I.D.

En un controlador PID, la acción del control se genera como una suma de tres términos, que matemáticamente se expresa:

$$u(t) = u_p(t) + u_I(t) + u_D(t) \quad (1)$$

Donde u_p representa la parte proporcional, u_I es la parte integral y u_D la parte derivativa.

Control proporcional.

La parte proporcional indica una realimentación de ganancia.

$$u_p(t) = Ke(t) \quad (2)$$

Siendo e el error de control, y K la ganancia del controlador. El error se define como la diferencia entre el punto del valor establecido como ideal y_{sp} y la salida efectiva del proceso y , en forma:

$$e(t) = y_{sp}(t) - y(t) \quad (3)$$

Sustituyendo (3) en (2) nos queda:

$$u_p(t) = K(by_{sp}(t) - y(t)) \quad (4)$$

Donde al coeficiente b lo llamamos "ponderación del valor ideal", que es el que permite el ajuste independiente de la respuesta del "punto del valor ideal" y la respuesta de variación de carga.

Control integral.

El controlador proporcional generalmente añade al sistema un "error de estabilización". La acción integral se introduce para eliminar este error y toma la forma:

$$u_i(t) = K_i \int e(s) ds = \frac{K}{T_i} \int e(s) ds \quad (5)$$

Reinicio automático.

Como indicamos anteriormente, un control proporcional añade un "error de estabilización". Para eliminarlo se puede añadir un término de reinicio ajustable, aplicado manualmente a la señal de control. El control proporcional dado por la ecuación (2) se convierte entonces en:

$$u_p(t) = Ke(t) + u_b(t) \quad (6)$$

Donde u_b es el término de reinicio.

Históricamente, la acción integral fue el resultado de un intento por obtener el ajuste automático del término de reinicio.

En forma gráfica se puede entender consultando la siguiente figura 1.1:

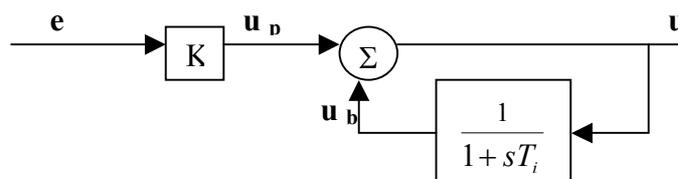


Figura 1.1 Controlador con acción integral implementada como reinicio automático.

La idea es filtrar la parte baja de la frecuencia de la señal de error y añadir ésta a la parte proporcional. Nótese que el lazo cerrado tiene realimentación positiva. Analíticamente, el sistema de la figura se puede expresar:

$$U(s) = K\left(1 + \frac{1}{sT_i}\right)E(s) \quad (7)$$

Lo que representa la respuesta de un controlador proporcional-integral (PI). De esta manera, tenemos:

$$u_b(t) = \frac{K}{T_i} \int e(s) ds = u_i(t) \quad (8)$$

El reinicio automático es entonces lo mismo que la acción integral. Nótese, sin embargo, que la "ponderación del valor ideal" no se genera al obtener la acción integral como reinicio automático.

Control derivativo.

El control derivativo se utiliza para disponer de una acción anticipativa, es decir, la derivada nos informa de una tendencia que afectará al futuro, y que se puede analizar y corregir si procede.

Su forma más sencilla es:

$$u_D(t) = k_d \frac{de(t)}{dt} = KT_d \frac{de(t)}{dt} \quad (9)$$

Combinando ambas acciones, integral y derivativa, resulta:

$$u_p(t) + u_D(t) = K\left[e(t) + T_d \frac{de(t)}{dt}\right] \quad (10)$$

Esto significa que la acción de control se basa en la extrapolación lineal del error T_d unidades hacia delante. El parámetro T_d , que denominamos tiempo derivativo, se puede interpretar en forma intuitiva, observando la gráfica de la figura 1.2.

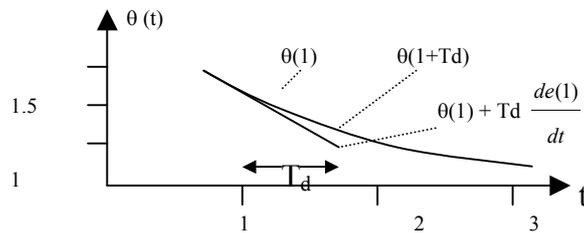


Figura 1.2 Interpretación de la acción derivativa como predicción del error.

La diferencia principal entre un controlador PID y otros más complejos radica en que el modelo dinámico permite mejores predicciones que una extrapolación estrictamente lineal. Veamos algunos casos:

Para algunas aplicaciones prácticas, el "punto del valor ideal" es, en general, constante, por ejemplo, el mantenimiento de una determinada velocidad angular. En este caso, la derivada del "punto del valor ideal" es cero, excepto para aquellos instantes de tiempo donde el "punto del valor ideal" es alterado. En estos instantes de tiempo la derivada puede llegar a ser muy grande y la extrapolación lineal no resulta útil cuando se utiliza para la predicción en tales señales.

Tampoco es adecuada, evidentemente, cuando la señal medida cambia rápidamente en comparación con la predicción T_d .

Una mejora en la representación de la acción derivativa es, entonces:

$$U_D(s) = \frac{KT_d s}{1 + sT_d / N} (cY_{sp}(s) - Y(s)) \quad (11)$$

La señal se pasa a través de un filtro paso-bajo cuya constante de tiempo es T_d / N . El parámetro c es una "ponderación del valor ideal", que es generalmente puesto a cero.

Ponderación del valor ideal.

El controlador PID introduce ceros extras durante la transmisión desde el "punto del valor ideal" a la salida.

Las ecuaciones y los ceros del controlador PID pueden ser determinados como las raíces de la ecuación.

$$cT_i T_d s^2 + bT_i s + 1 = 0 \quad (12)$$

Donde no se dan ceros extras si $b = 0$ y $c = 0$. Si solo $c = 0$, entonces hay un cero extra para:

$$s = -\frac{1}{bT_i} \quad (13)$$

Este cero puede tener una influencia significativa en la respuesta del "punto del valor ideal". La sobre oscilación es generalmente grande para $b = 1$, y puede ser reducida sustancialmente utilizando un valor pequeño de b . Esta resulta una solución mucho mejor que la forma tradicional de sincronizar al controlador.

En la figura 1.3, se muestra un control PI con función de transferencia.

$$G_p(s) = \frac{1}{s+a}$$

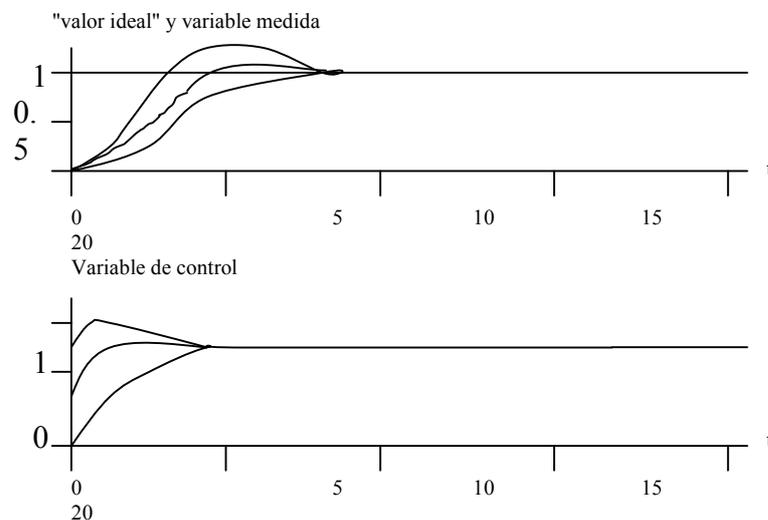


Figura 1.3 La utilidad de la "ponderación del valor ideal". Los valores del parámetro de la "ponderación del valor ideal" es 0, 0.5 y 1.

Diferentes representaciones del controlador PID.

El controlador PID discutido en la sección anterior puede ser descrito como.

$$U(s) = G_{sp}(s)Y_{sp}(s) - G_c(s)Y(s) \quad (14)$$

Donde:

$$G_{sp}(s) = K \left(b + \frac{1}{sT_i} + c \frac{sT_d}{1 + sT_d / N} \right) \quad (15)$$

$$G_c(s) = K \left(1 + \frac{1}{sT_i} + \frac{sT_d}{1 + sT_d / N} \right) \quad (16)$$

El comportamiento lineal del controlador se caracteriza entonces por dos funciones de transferencia: $G_{sp}(s)$, que representa la transmisión de la señal desde el "punto del valor ideal" a la variable de control, y $G_c(s)$, que describe la transmisión de la señal desde la salida del proceso a la variable de control.

Nótese que la transmisión de señal desde la salida del proceso hasta la señal de control es diferente del camino de la transmisión de señal desde el "punto del valor ideal" a la señal de control si para el parámetro "ponderación del valor ideal" tenemos que $b \neq 1$ o $c \neq 1$. El controlador PID dispone entonces de dos grados de libertad.

El controlador PID se comporta como un algoritmo de control simple que tiene siete parámetros:

Ganancia del controlador K , tiempo integral T_i , tiempo derivativo T_d , ganancia derivativa máxima N , "ponderación del valor ideal" b y c y constante de tiempo del filtro T_f .

Los parámetros K , T_i y T_d , son los parámetros primarios que normalmente son discutidos.

El parámetro N es una constante cuyo valor oscila entre 5 y 20.

El parámetro de la "ponderación del valor ideal" b tiene normalmente los valores 0 o 1, aunque puede tomar otros valores.

El parámetro c generalmente tiene valor cero para controladores comerciales.

La forma estándar.

El controlador dado por las ecuaciones anteriores está extraído de la denominada *forma estándar* o la forma ISATM (Instrument Society America, por sus siglas en inglés).

La *forma estándar* admite ceros complejos, lo cual es útil al trabajar con sistemas de control con polos oscilatorios.

La parametrización dada para la ecuación (14) es la forma habitual, aunque existen otros tipos.

La forma paralela.

Es una ligera variación de la forma estándar, y se describe.

$$U(s) = k[bY_{sp}(s) - Y(s)] + \frac{k_i}{s}[Y_{sp}(s) - Y(s)] + \frac{k_d s}{1 + sT_{df}}[cY_{sp}(s) - Y(s)] \quad (17)$$

Esta forma tiene la ventaja de que es fácil obtener un control puramente proporcional, integral o derivativo simplemente escogiendo los parámetros adecuados con valor a cero.

La interpretación de T_i y T_d como tiempo de integración y tiempo de predicción se pierden en esta representación.

Los parámetros de los controladores dados por las ecuaciones anteriores están relacionados por:

$$\begin{aligned} k &= K \\ k_i &= \frac{K}{T_i} \\ k_d &= KT_d \end{aligned} \quad (18), (19) \text{ y } (20)$$

La forma dada por la ecuación (17) se usa generalmente en cálculos analíticos, debido a la linealidad de los parámetros. Sin embargo, los parámetros no tienen una interpretación física sencilla.

La forma en serie.

Si $T_i > 4T_d$ la función de transferencia $G_c(s)$ puede ser escrita como.

$$G'_c(s) = K' \left(1 + \frac{1}{sT'_i}\right) (1 + sT'_d)$$

Esta forma es llamada forma en serie.

Si $N = 0$, los parámetros se relacionan con la forma paralela en el siguiente modo:

$$K = K' \frac{T'_i + T'_d}{T'_i}$$

$$T_i = T'_i + T'_d$$

$$T_d = \frac{T'_i T'_d}{T'_i + T'_d}$$

La relación inversa es:

$$K' = \frac{K}{2} (1 + \sqrt{1 - 4T_d/T_i})$$

$$T'_i = \frac{T_i}{2} (1 + \sqrt{1 - 4T_d/T_i})$$

$$T'_d = \frac{T_i}{2} (1 - \sqrt{1 - 4T_d/T_i})$$

Para $N \neq 0$ se obtienen formulas similares, pero más complejas.

Nótese que la forma paralela admite ceros complejos, mientras que la forma en serie tiene ceros reales.

La forma en serie es también llamada la *forma clásica* debido a que ésta es la obtenida naturalmente cuando se implementa un controlador con reinicio automático.

Algoritmos de velocidad.

Los controladores PID dados por las ecuaciones vistas son llamados algoritmos posicionales, debido a que la salida del algoritmo es la variable de control.

En algunos casos es más natural que el algoritmo de control genere la velocidad de cambio de la señal de control. Tal ley de control es denominada algoritmo de velocidad. En diseños digitales, los algoritmos de velocidad se llaman también algoritmos incrementales.

Algunos controladores primitivos diseñados para el control de motores utilizaban algoritmos de velocidad. Estos algoritmos y estructuras se mantuvieron por los fabricantes en espera de que la tecnología alcanzara el grado de desarrollo necesario para contar con productos compatibles con la antigua tecnología. Sin embargo, existen razones de protección contra efectos indeseables inherentes al control por los que es importante el conocimiento y uso de algoritmos de velocidad.

Por ello, no hemos querido pasar por alto este apéndice.

3. Motores de propósito especial.

Motores de pasos.

Los motores de pasos son dispositivos utilizados en una variedad de aplicaciones donde movimientos con una pequeña cantidad de torque son necesarios. Aplicaciones típicas de este tipo de motores es el movimiento de mecanismos en impresoras, posicionamiento de láser en dispositivos ópticos, control de tablas rotatorias entre otras.

Los motores de pasos funcionan al aplicarles secuencias de pulsos de entrada (típicamente sobre cuatro cables de conexión unidos a cuatro interruptores de control), en cada pulso el motor gira un ángulo dado, por ejemplo 1.8° . A este ángulo de giro se le denomina *ángulo de paso*. Los ángulos de paso estándar típicos en grados son: 0.72, 1.8, 3.75, 7.5, 15 y 18.

De esta manera un motor de pasos con un ángulo de paso de 3.75° completará una revolución en exactamente 96 pasos. La tolerancia radial de posicionamiento típica que se maneja para este tipo de motores es de 3%-5% para un paso; de esta manera para el ejemplo de 3.75° la tolerancia sería, a 4%, $0.04 \cdot 3.75$, es decir 0.15° .

Los motores de pasos están contruidos basándose en estatores y rotores de múltiples polos. Así, por ejemplo, un motor típico contiene un estator de cuatro polos y un rotor de seis polos, y utiliza cuatro interruptores de control. Las secuencias de pulsos aplicadas a los interruptores son generadas típicamente por un circuito integrado especializado, lo cual permite un control mas adecuado.

Motores sincros y resolvers.

Sincros es un nombre general para una serie de motores que indica posición de movimiento rotatorio duplicado. *El resolver*, en algunas ocasiones considerado como un subtipo del *synchro*. Tanto el motor *Sincro* como el motor *resolver* son dispositivos para pequeños torques.

Los *sincros* son pequeños motores cilíndricos que varían en diámetro desde $\frac{1}{2}$ " hasta 4", dependiendo de su potencia de salida. Este tipo de motor funciona convirtiendo una señal de entrada de fase sencilla en tres voltajes no balanceados que están 120° fuera de fase eléctricamente.

Los motores *sincros* están contruidos a base de un estator con tres devanados y un rotor con un devanado que es alimentado con un voltaje a través de dos escobillas por medio de dos anillos deslizantes (colectores). Existen siete tipos principales de motores *sincros*: transmisor de torque (TX), transmisor de control (CX), transmisor diferencial de torque (TDX), transmisor diferencial de control (CDX), receptor de torque (TR), receptor diferencial de torque (TDR) y transformador de control (CT). Una aplicación típica de estos motores es utilizar un transmisor de torque (TX) para manejar a un receptor de torque (TR). Así, si el TX rota a un ángulo dado, el TR rotará el mismo ángulo. Otro uso es generar la suma o la sustracción de ángulos de rotación; en este caso se colocan en serie un TX, un transmisor diferencial de torque (TDX) y un TR. Así, la posición de TR será el resultado de la suma o resta del ángulo de rotación de TX y el ángulo de rotación de TDX.

Los *resolvers* son motores parecidos a los *sincros*, pero varían en los siguientes aspectos: El desplazamiento eléctrico en este caso es de 90° , y no de 120° ; tanto el estator como el rotor contienen dos devanados; un voltaje de entrada puede ser aplicado tanto al estator como al rotor. Una aplicación típica de un *resolver* es en la navegación o en la determinación de alturas.

Servomotores – D.C. y A.C.

Los *servomotores de D.C. y A.C.* son dispositivos ideales para manejar altas capacidades de torque, control preciso de velocidad y control preciso de posición. De esta manera, podemos listar las principales características de los servomotores como sigue: pueden producir grandes torques a altas velocidades; son capaces de mantener una posición estática; son capaces de mantener bajas velocidades o nulo movimiento sin sobrecalentarse; son capaces de invertir su dirección de movimiento rápidamente para alcanzar una posición o una velocidad rápidamente, los servomotores son capaces de acelerar o desacelerar rápidamente; los servomotores son capaces de retornar a una posición específica después de un intervalo de tiempo sin generar movimientos erráticos.

Los servomotores se construyen con cuatro, seis o más polos y pueden ser adecuados para poder utilizar suministro de potencia trifásica. Tanto la velocidad rotacional como el torque en este tipo de motores, son controladas por la diferencia de fase entre el devanado principal y el devanado de control. Invertir la diferencia de fase nos permite controlar retardos en el movimiento o cambiar la dirección de rotación del motor.

De esta manera, si llamamos V_s al voltaje de alimentación del devanado principal y V_c al voltaje de alimentación del devanado de control, y si ambos voltajes tienen la misma amplitud, tenemos lo siguiente: para un torque máximo las formas de onda de V_s y V_c están desfasadas 90° ; para velocidades menores, el desfase entre V_s y V_c es menor a los 90° ; para un movimiento nulo las formas de onda están perfectamente en fase.

Otro tipo de motores.

Se discuten aquí brevemente siete tipos de motores: *motor de rotor de disco*, *motor de efecto-hall*, *motor de histéresis*, *motor de reluctancia*, *motor lineal*, *motor de torque* y *motor inverso*.

El *motor de rotor de disco*, es también llamado motor de movimiento espiral. Consiste de un estator con magnetos cilíndricos permanentes. Los magnetos son alternadamente de polaridad norte y sur. El rotor de disco es eléctricamente alimentado con corriente a través de un conmutador. El rotor tiene una gran cantidad de trayectorias de corriente y funciona de manera similar que en un motor de DC o un motor universal de AC. El motor de rotor de disco difiere de otros en que su rotor tiene una muy baja inercia; de esta manera, la relación del torque de motor a la inercia es muy alta. Por esta razón este tipo de motor es utilizado en aplicaciones donde las aceleraciones, desaceleraciones y cambios de sentido de rotación rápidas son esenciales.

El *motor de efecto-hall*, es un motor de DC eficiente, confiable y de baja potencia. Su operación está basada en el efecto hall. El efecto hall toma lugar cuando un campo magnético y una corriente son aplicados a un material semiconductor especial o a un cristal. El campo magnético y la corriente aplicada producen un voltaje a través del material semiconductor. Este voltaje de DC resultante es proporcional al producto del campo y de la corriente.

El *motor de histéresis*, es esencialmente un motor de inducción que funciona a velocidad síncrona y tiene un rotor especial. Su estator puede ser de cualquier tipo que produzca un campo magnético giratorio. Estos motores son de baja potencia y de baja eficiencia. Los motores de hysteresis son utilizados para lentos arranques, aplicaciones síncronas, tales como los temporizadores. El rotor consiste de un núcleo de material no magnético y una capa de material magnético adherido al núcleo. El principio de operación de este motor es el retardo de acumulación del campo magnético del rotor causado por efecto de histéresis. El material seleccionado para la superficie del rotor debe de tener características de alta hysteresis. Aleaciones que contienen cobalto o vanadium son materiales que cumplen con el requisito y son generalmente utilizadas.

El *motor de reluctancia*, es otro motor que funciona a velocidad síncrona. Su ventaja sobre un motor síncrono convencional es que este no contiene devanados en el rotor. La ausencia de devanado en el rotor elimina la necesidad de colectores, escobillas y cualquier alimentación hacia el rotor. Al eliminar estos elementos se reduce los costos por mantenimiento. Sin embargo, el motor de reluctancia es un motor de torque pequeño, bajo factor de potencia y un motor de baja eficiencia.

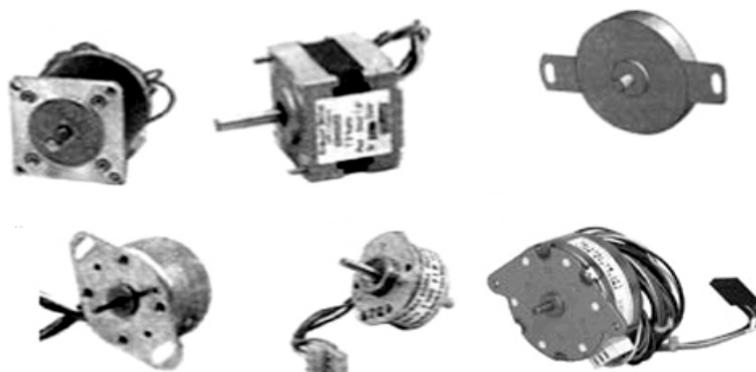
El rotor de este tipo de motores tiene una indentación entre polos y contiene barras de cobre colocadas axialmente sobre los polos. El rotor tiene diferentes reluctancias magnéticas a diferentes ángulos del rotor. El estator puede ser de cualquier tipo que produzca un campo magnético giratorio.

Los *motores lineales*, son esencialmente solenoides de posición variable. Están contruidos principalmente a partir de bobinas y magnetos permanentes. Cuando una corriente fluye por las bobinas, la fuerza magnética atrae o repele a los magnetos permanentes y se genera el movimiento.

El *motor de torque*, es un tipo especial de motor lineal que tiene típicamente el tamaño de un puño y son generalmente costosos. El movimiento de estos motores es normalmente de milésimas de pulgada. La ventaja de estos motores es la gran linealidad que presentan entre la señal de entrada y su mecanismo de desplazamiento. Una aplicación típica de estos motores es en servo válvulas hidráulicas. El torque por el movimiento del motor es usado para variar el flujo de una válvula hidráulica piloto. También, el torque del rotor tiene una relativamente alta frecuencia de respuesta y puede seguir rápidas variaciones de la señal de entrada.

En el motor “*inside-out*”, la parte normalmente definida como rotor es un elemento estacionario. Los solenoides son embobinados alrededor del elemento estacionario y los conectores de la potencia son colocados fuera del rotor. La parte normalmente definida como estator, en este caso es el elemento que rota. El elemento que esta rotando está hecho de magnetos permanentes y el elemento estacionario es alimentado con pulsos de D.C. o A.C. para crear el campo magnético giratorio. Los motores “*inside-out*” se encuentran en algunas máquinas de oficina en aplicaciones especiales.

A continuación se puede observar una figura con algunos de los MOTORES mencionados.



4. Hardware del sistema.

4.1. Descripción del sistema de control de velocidad.

Realmente el sistema está definido por el procesador anfitrión que manda todos los comandos al microcontrolador PID programable LM629 (MPID), en donde llega primeramente el valor de la señal inicial, cuando el motor inicia su movimiento y posteriormente retroalimenta el valor de la señal del encoder.

Esta es decodificada dentro del MPID realizándose la suma de estos valores, se genera una diferencia de error que actúa directamente en el filtro PID quien se encarga de corregir los valores que llegan al dispositivo de puente H.

Los MPIDs, son dispositivos dedicados de control de motores de CD y servomecanismos, que ocupan señales de retroalimentación de cuadratura incremental, en la figura 2.1 y 2.2 se muestra el sistema de control de los motores mediante los dispositivos LM628 y el LM629 respectivamente.

Ambos el LM628 y el LM629 tienen un bus periférico, y pueden ser programados por una computadora (procesador). Esta aplicación no es recomendable por presentar un punto de iniciación concreto para la programación de estos controladores de movimiento de precisión. Si el enfoque es sobre el desarrollo de programas cortos, donde esta prueba es funcionalmente en sistemas Globales, y ponen el fundamento para más programas complejos. Este representa también un método para armonizar el filtro PID de compensación de lazo.

En la figura 2.1 y 2.2 se ve que el sistema de control lo realiza en interfaz directa con un procesador dentro del mismo LM628 o LM629 donde se tienen 6 líneas de control y 8 líneas a un puerto de E/S que se conecta a una computadora, en nuestro diseño los datos pasan directos de la computadora.

Durante los inicios de hardware y software cuando se utiliza un DAC a la salida por default se utiliza el modo de 8 bits. Si un LM628 utiliza un DAC de 12 bits en el lazo de control, el comando PORT12 puede ser emitido inmediatamente seguido del bloque de inicio del hardware, accionándose todos los reset subsecuentes. La falla para que el comando PORT12 sea emitido, es el comportamiento errático e impredecible del motor.

Si el lazo de control utiliza un DAC de 8 bits, el comando de 8 bit no debe ser ejecutado, esto depende del resultado errático e impredecible del motor.

Un LM629 ignorará el comando PORT8 (ya que da una salida PWM en signo / magnitud de 8 bits). Y el comando PORT12 puede ser no emitido en el sistema basado en el LM629.

De hecho el LM628 y el LM629 funcionan igual excepto en el modo de utilizar la salida de puerto en el comando PORT12.

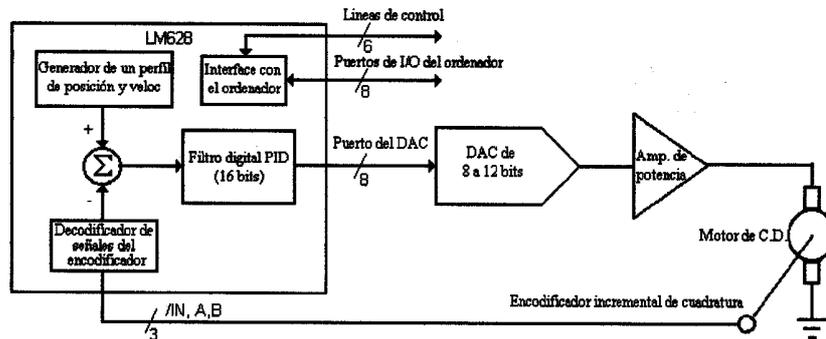


Figura 2.1 Sistema de Control para un Motor Basado en el LM628.

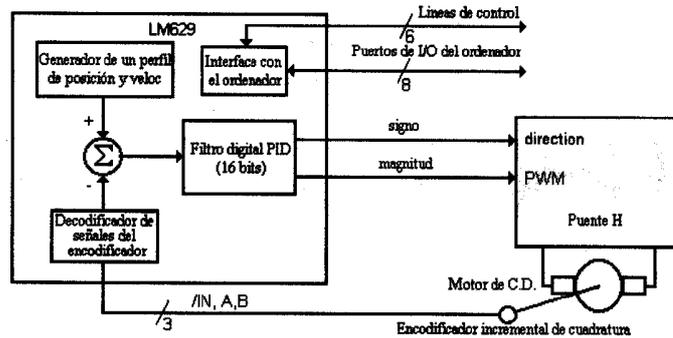


Figura 2.2 Sistema de Control para un Motor Basado en el LM629.

4.2. Filtro PID.

El MPID, usa un filtro digital PID (*proporcional, integral y derivativo*) para compensar el lazo de control. El motor se sostiene en la posición deseada al aplicar una fuerza restauradora para el motor que es: proporcional al error de posición, más el error de integración, y más el error derivativo. La siguiente ecuación discreta en el tiempo ilustra el control ejecutado por el MPID.

$$u(n) = K_p * e(n) + K_i \sum_{n=0}^n e(n) + K_d [e(n) - e(n-1)] \quad (1)$$

Donde $u(n)$ es la señal de salida de control en el tiempo “n” de muestreo, $e(n)$ es el error de posición en el tiempo “n” de muestreo, “n” indica la muestra de la derivada de muestreo, y K_p , K_i , y K_d son los parámetros cargados (ajustados por el usuario o preajustados de fábrica).

El primer término, el *proporcional*, provee una fuerza restauradora proporcional al error de posición, de la misma forma como un resorte obedece a la ley de Hooke.

El segundo término, el de *integración*, provee una fuerza restauradora que crece con el tiempo, asegurando que el error de posición sea cero. Aunque exista una carga constante del torque, el motor será capaz de llevarlo al error de posición cero.

El tercer término, el *derivativo*, provee una fuerza proporcional a la razón de cambio del error de posición, este actúa de la misma forma que un amortiguador viscoso en un sistema de resorte de masa. El intervalo de muestreo asociado con el término derivativo es seleccionado por el usuario, todo esto hace que el MPID sea capaz de controlar un amplio rango de cargas inerciales (constante mecánica de tiempo), ya que la derivada continua provee una mejor aproximación. En general, los intervalos de muestreo son muy útiles en operaciones de baja velocidad.

En la operación, el algoritmo del filtro recibe una señal de error de 16 bits desde el lazo de control del sumador, donde la señal de error se satura en 16 bits para asegurar un comportamiento. En resumen, al multiplicarse por los coeficientes del filtro K_p , la señal de error se agrega a una acumulación de errores previos para formar la señal integral, y en una razón determinada al escoger el intervalo de muestreo derivativo, el error previo se sustrae de este (para formar la señal derivada). Todas las operaciones de multiplicación del filtro son de 16 bits, donde únicamente se utiliza la parte baja del producto de los 16 bits.

La señal integral se mantiene en 24 bits, pero solamente se utiliza la parte alta de los 16 bits. Esta es la técnica de escalar un rango útil de los valores coeficientes de K_i . Los 16 bits se desplazan 8 posiciones a la derecha y multiplicados por el coeficiente K_i del filtro para formar el término desde el cual, contribuye a la salida del motor. La magnitud absoluta de este producto es comparada con los coeficientes K_i , y el valor más pequeño en magnitud se apropia y se asigna entonces a la señal del motor.

La señal de derivación se multiplica por el coeficiente K_d a cada intervalo de muestreo derivativo. Este producto contribuye a la salida del motor en cada intervalo de muestreo, independientemente del intervalo de muestreo derivativo escogido por el usuario.

El K_p , K_i limitado y K_d en productos son sumados para formar una cantidad de 16 bits. Dependiendo del modo de salida (tamaño de palabra) ya sea de 8 o de 12 bits para la salida de control del motor.

4.3. Descripción de la arquitectura del dispositivo MPID.

Diagrama a bloques del sistema MPID.

En la figura 2.3 se observa el MPID con sus terminales de entrada-salida, el microprocesador, el puente H, el motor, y el encoder, que forman el lazo de control PID de estudio.

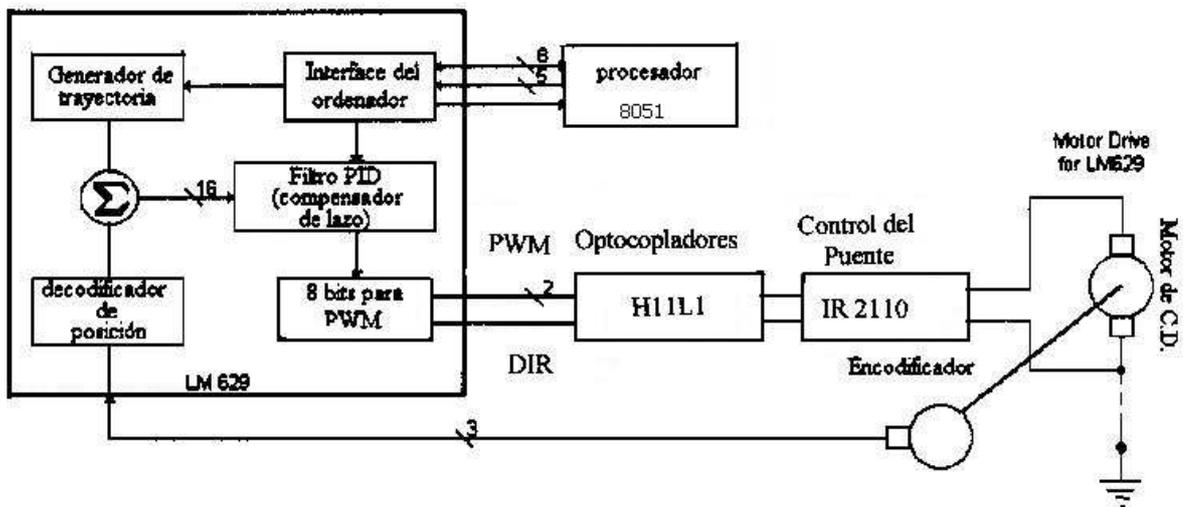


Figura 2.3 Diagrama de bloques del sistema del LM629.

Arquitectura del Microcontrolador PID (MPID).

En esta sección se detallan cada uno de los componentes y su uso dentro de la arquitectura del microcontrolador PID usado, ver figura 2.4. Se tiene una *ROM* de 1k x 16 bits que usa instrucciones de 16 bits, donde el algoritmo de control es almacenado. Se cuenta con un secuenciador *PLA* que decodifica esas instrucciones y da transferencia de señales de datos cronometradas para un bus de datos de 16 bits. Tiene una *RAM* donde son almacenados los parámetros de trayectoria de doble palabra (32 bits). Cuenta con un registro de posición de 32 bits de suficiente rango dinámico, el cual también es usado para valores de velocidad y aceleración. Y por último, hay un *ALU* de 32 bits que se usa para soportar multiplicaciones de 16 x 16 bits para valores del error y coeficiente del filtro PID.

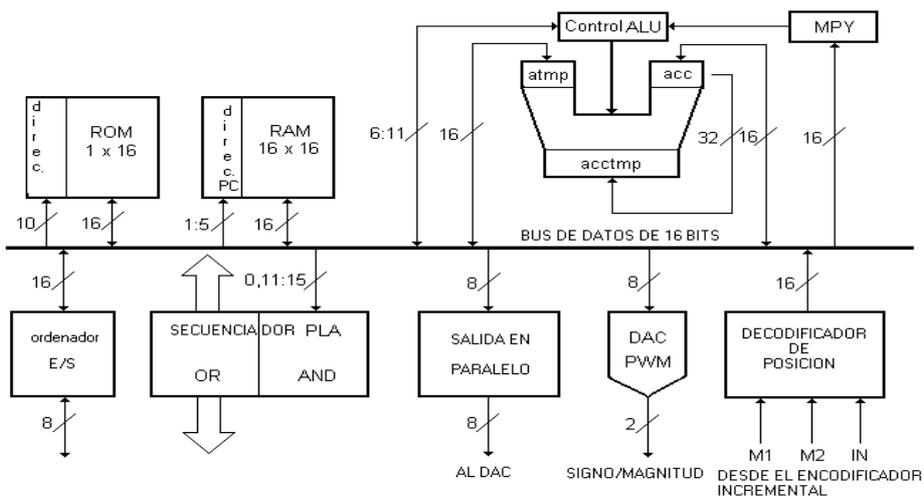


Figura 2.4 Arquitectura del Microcontrolador PID (MPID)

Decodificación de la posición del eje del motor.

El microcontrolador PID tiene una interfaz para capturar tres señales de la posición óptica en el eje del encoder. Dos señales cuadradas de entrada M_1 y M_2 son decodificadas para obtener la información de la posición y la dirección, una tercera señal de entrada opcional IN usada para capturar la posición una vez por revolución, ver figura 2.4. Los cuatro estados de la señal cuadrada de posición son decodificados por el microcontrolador PID, ver figura 2.9. Las señales de entrada tienen un incremento de cuatro tiempos para la resolución de la posición en el número de líneas del encoder. Las “N” líneas del encoder pueden ser decodificadas como una posición de “4N” cuentas, en dicho dispositivo. El secuenciador *PLA* decodifica la base de las señales M_1 , M_2 e IN y determina si el eje del motor se mueve hacia adelante o hacia atrás, o permanece inmóvil.

Dentro del secuenciador se tiene un contador ascendente de 16 bits que mantiene la posición actual del eje del motor una vez por revolución, cuando las tres señales M_1 , M_2 e IN están en bajo.

4.4. Operaciones de lectura y escritura del MPID.

El procesador escribe comandos al MPID vía el puerto de entrada / salida, cuando la terminal de entrada 16 (Port Select /PS) está en bajo lógico. El código del comando deseado se aplica a la línea del puerto paralelo, cuando la terminal de entrada 15 (/WR) es activada. El byte de comando pasa los valores al MPID, al subir la señal de entrada /WR. Cuando el byte del comando de escritura es necesario, primero se lee el byte de estado y enseguida se prueba el estado de una bandera, llamada “Bit-busy” (bit 0).

Si el Bit-busy está en uno lógico, el comando de escritura no se realiza. El “Bit-busy” nunca es más grande que 100µsegundos, típicamente cae entre los 15 a 25 µseg.

El procesador lee el byte de estado del MPID de una manera similar, es decir, al activar la señal de la terminal 13 (/RD) mientras que /PS está en bajo, la información del estado permanece válida mientras /RD esté en bajo.

Los datos de lectura y de escritura desde el MPID se aplican con /PS en alto. Estas escrituras y lecturas se dan siempre en números enteros (de 1 a 7) con palabras de 2 bytes, siendo el primero de cada palabra el más significativo. Cada byte requiere ser probado en lectura o escritura. Cuando se transfiere la palabra de dato, primero es necesario leer el byte de estado y luego probar el estado del “Bit-busy”. Cuando el “Bit-busy” esta en bajo lógico, el usuario puede entonces transferir secuencialmente ambos bytes al comprimirse en una palabra de dato, pero el Bit-busy puede volverse a probar y determinar que está en bajo, antes de atender la transferencia del par de bytes siguientes. La transferencia de datos se envía vía las interrupciones internas del MPID, el “Bit-busy” informa a la computadora cuando el MPID interrumpe para transferir el dato, si el comando se escribe cuando el Bit-busy esta en alto, este será ignorado.

El “Bit-busy” irá a alto después de escribir un byte comando, o leer o escribir un segundo byte de datos.

4.5. Salidas al motor.

Las salidas al motor del MPID al DAC pueden configurarse para proveer ya sea una salida paralela, pasando valores de 8 bits, o una salida de 12 bits multiplexados. La salida de 8 bits puede ser directamente conectada al DAC. La salida de 12 bits puede fácilmente ser demultiplexada pasando a un valor externo de 6 bits y un DAC de entrada de 12 bits. El dato de salida del DAC y los 12 bits son código binario con desplazamiento, el código de 8 bits es 80 hex y el código de 12 bits para el cero es 800H.

Los valores menores de estos causan un torque negativo para aplicarse al motor, y en forma contraria, los valores grandes causan un torque positivo. Cuando el MPID se configura para una salida de 12 bits, provee señales que controlan la demultiplexación del proceso.

El MPID provee 8 bits de señales de salida PWM de signo y magnitud, para excitar al amplificador de interrupciones.

Son estas dos señales independientes, la señal de amplitud y la señal de signo, las que determinan la tensión y la polaridad, respectivamente.

La señal de amplitud consiste en un tren de impulsos, a una frecuencia constante ($512/f_{clk}$), cuya duración es función de la señal de control obtenida del PID. Cuanto mayor sea esta, mayor será el llamado factor de servicio o “*duty-cycle*” y , por lo tanto, mayor será la tensión que reciba el motor.

El rango de a (a = duración del pulso dividido entre el periodo de la señal) esta entre $0/128$ (motor parado) y $127/128$ (mayor tensión con polaridad positiva) ó $128/127$ (tensión continua, polarización negativa).

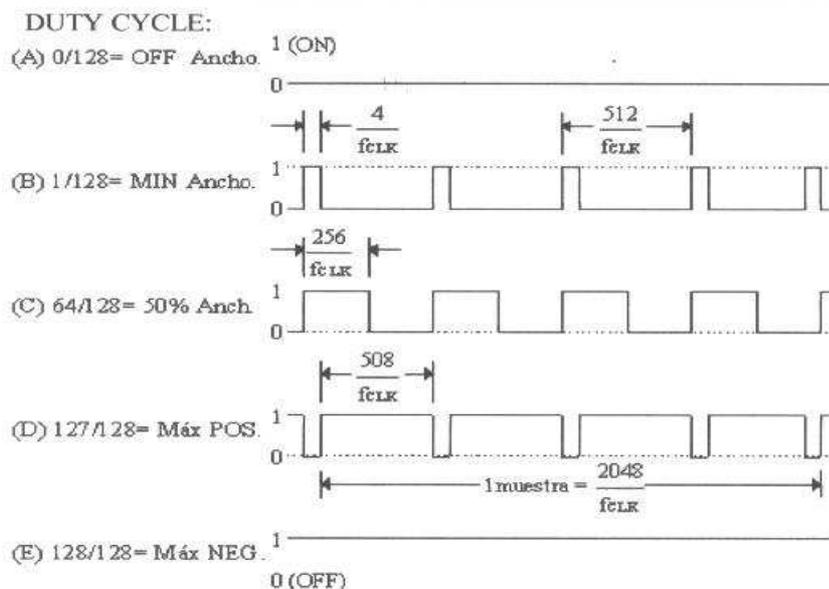


Figura 2.5 Señales de salida de PWM

Hay cuatro periodos PWM en un intervalo de muestreo del LM629. De este modo, para un LM629 con reloj de 8 Mhz se incrementa el rango de salida 3,9 Khz a una frecuencia de conmutación del puente en H de 1,5Khz.

La tabla 3-6, del apartado 10, muestra un formato de señales de salida en magnitud PWM.

4.6. Descripción de los terminales del MPID.

El dispositivo MPID esta montado en un empaque de 28 terminales en doble línea.

El terminal 1 de entrada /IN, recibe un pulso opcional desde el encoder y puede ser sostenido en alto. Si no se usa, da la posición de la flecha cuando este pulso es leído y cuando las terminales 2 y 3 están en bajo.

Los terminales 2 y 3, reciben las señales del encoder en las entradas *A* y *B*. Estas señales de cuadratura vienen del encoder incremental y están desfasadas 90° una de otra, así la dirección de rotación del motor.

Los terminales 4 al 11 son puertos de E/S (del D0 al D7). Es un puerto bidireccional de datos el cual conecta al procesador de la computadora, usado para escribir comandos y datos al MPID, y para leer el byte de estado y datos desde el MPID que es controlado por las siguientes terminales: /CS(terminal 12), /PS(terminal 16), /RD(terminal 13) y /WR(terminal 15).

El terminal 12, es entrada al Chip Select (/CS), usada para seleccionar la operación de lectura y escritura del MPID.

El terminal 13, es entrada para lectura (/RD), usada para leer el estado y datos.

El terminal 14, es tierra (GND).

El terminal 15, es entrada de escritura (/WR) , usada para escribir comandos y datos.

El terminal 16, es entrada para selección de puerto (/PS), usado para seleccionar puertos para comandos si está en bajo, o datos si está en alto. Los siguientes modos son controlados en esta terminal:

- Comandos que son escritos al puerto de comandos (con la terminal 16 en bajo).
- El byte de estado es leído desde el puerto de comandos (con la terminal 16 en bajo).
- Los datos son escritos y leídos por la vía del puerto de datos.

El terminal 17, es salida de interrupción del ordenador (HI), esta señal alerta al ordenador (por medio de un servicio de una rutina de interrupción) cuando una condición de interrupción ha ocurrido.

Los terminales 18 al 25, son puerto del DAC (DAC0 al DAC7), es salida de puerto la cual es usada en tres modos diferentes:

- Modo para el MPID (LM628, salida de 8 bits), siendo el MSB el terminal 18, y el LSB el terminal 25.
- Modo para el MPID (LM628, salida de 12 bits), este tiene doble salida de palabras de 6 bits multiplexadas, el MSB esta en el terminal 18 y el LSB en el terminal 23, el terminal 24 es usado para el multiplexado de las palabras.
- Modo para el MPID (LM629 con salidas de signo / magnitud), las salidas en esta señal PWM son, con signo en el terminal 18 (o 11 para montaje de superficie) y una señal PWM en magnitud sobre el terminal 19 (o 13 para montaje de superficie), los terminales del 20 al 25 no son usados en el LM629.

El terminal 26, es entrada de reloj (Clock), donde recibe el sistema de reloj.

El terminal 27, es entrada para reset (/RST), y es activo en bajo.

El terminal 28, es el suministro de voltaje (V_{DD}), con un voltaje de 5 volts.

MICROCONTROLADOR (μ C)

Justificación

A estas alturas no es necesario hacer la presentación de éste tipo de componentes, que llevan ya muchos años, desde la década de los ochenta, introducidos en el mundo electrónico y en la vida cotidiana. No vamos pues a entrar en su utilidad, sino que nos ceñiremos a nuestra aplicación y su actividad en ella.

Las necesidades de control de nuestro sistema son las siguientes:

- Un puerto paralelo de ocho bits, de comunicación con el MPID.
- Señales de control para el MPID, en concreto:
 - √ Selección del periférico, “Chip select”, CS.
 - √ Señal de aviso de lectura, RD.
 - √ Señal de aviso de escritura, WR.
 - √ Selección de uso del puerto de comunicaciones para datos o instrucciones del MPID, PS.
 - √ Reloj que se enviará al MPID, CLKp.
 - √ Línea de recepción de interrupciones generadas por el MPID.

√ Línea de RESET “hardware” del MPID.

- Línea de recepción de interrupciones generadas por el puente H.
- Líneas de información de datos aportados por el puente H.

Las capacidades que precisaremos del μ C van a ser:

- Velocidad, reloj, superior a 8 Mhz, dado que ésta va a ser la velocidad del MPID.
- Dos puertos paralelo de 8 bits mínimo, uno para la comunicación con el PID y el otro para señales de control.
- Memorias RAM y ROM. No se considera necesario ampliar la memoria.
- Puerto serie para comunicación con el exterior del sistema.
- Memoria EPROM para su uso en la comunicación con el exterior del sistema.

Son muchos los microcontroladores que actualmente cumplen estos requisitos mínimos y diversas las casas que los fabrican. Existen, además de los microcontroladores clásicos, los DSP (Digital Signal Processing), cuya arquitectura interna difiere del μ C con objeto de agilizar la comunicación, para el trabajo en tiempo real, pero en nuestro caso esta actividad la llevará a cabo el MPID, por tanto el μ C es adecuado y suficiente.

Fabricantes de referencia en este campo son “Motorola” e “Intel”. Se ha buscado información de ambas casas, teniendo ofertas similares. En concreto, y aunque se ha optado por la familia 8051 de “Intel”, se estudió la familia 68HC11 de “Motorola”, que aporta en su modelo MC68HC11F1 un convertidor analógico digital (ADC) interno de 8 bits que pudiera resultar útil en el manejo de datos analógicos procedentes del puente H.

Pero esto se va a resolver en forma diferente, y por otro lado en la comparativa de velocidades del reloj, la oferta de la familia 8051 es más ventajosa.

No podemos obviar tampoco que la información disponible de la familia 8051 ha sido mayor, siendo muy utilizados y muy estudiados. Por tanto hemos optado por un sistema ya conocido y ampliamente experimentado.

La elección del modelo, con arreglo a las necesidades dio como resultado el uso del 87C51RC, encapsulado DIP, con 40 contactos.

Sin embargo, y como era de esperar, al recabar información en la página “web” de Intel, encontramos la última actualización de la familia, los 8XC251XX, que prometen compatibilidad “hardware” y “software” con la primitiva MCS51, mejorando aspectos externos e internos. El encapsulado es de 44 pines, PLCC, puesto que no es preciso mantener compatibilidad. Aunque no es objeto del trabajo, en paralelo con la descripción del nuevo modelo seleccionado incluimos datos del antiguo en plan comparativo.

El modelo seleccionado, equivalente al 87C51RS va a ser el 87C251SB de la misma casa Intel. Entre corchetes [x], se incluyen las diferencias, cuando las hay, con el antiguo modelo.

Sus características generales son:

- 1 K Byte de memoria RAM. [512 bytes]
- 16 K ROM / OTPROM / EPROM. [32K]
- Reloj (CLK) de 16 Mhz. [12 Mhz]
- 4 puertos paralelo bidireccionales de 8 bits.
- 3 contadores de 16 bits con:
 - √ 1 contador arriba / abajo
 - √ Salida de reloj
- Puerto serie programable “Full Duplex”
- Set de instrucciones ampliado
- 3 contadores de 16 bits
- Estructura de Interrupciones
 - √ Siete fuentes de interrupción
 - √ Cuatro niveles de prioridad
- Vigilante Watchdog hardware
- Reducción de consumo tipos “idle” y “power down”.
- Más detalles en las características del fabricante.

µC 87C251SB. DESCRIPCIÓN DE PATILLAS.

PIN	SIGNO	DESCRIPCIÓN	ALTERNATIVA
01	VSS1	ALIMENTACIÓN	
02	P1.0	PUERTO 1.0 / T2	-
03	P1.1	PUERTO 1.1 / T2EX	
04	P1.2	PUERTO 1.2 / ECI	
05	P1.3	PUERTO 1.3 / CEXO	
06	P1.4	PUERTO 1.4 / CEX1	
07	P1.5	PUERTO 1.5 / CEX2	
08	P1.6	PUERTO 1.6 / CEX3 / WAIT #	
09	P1.7	PUERTO 1.7 / CEX4 / A17 / WCLK	
10	RST	RESET.	-
11	P3.0	PUERTO 3.0. / RECEPCIÓN SERIE	RXD
12	VCC2	ALIMENTACIÓN	
13	P3.1	PUERTO 3.1. / TRANSMISIÓN SERIE	TXD
14	P3.2	PUERTO 3.2 / INTERRUPCION 0.	/INT0
15	P3.3	PUERTO 3.3 / INTERRUPCION 1.	/INT1

16	P3.4	PUERTO 3.4 / TIMER 0.	T0
17	P3.5	PUERTO 3.5 / TIMER 1.	T1
18	P3.6	PUERTO 3.6 / ESCRITURA.	/WR
19	P3.7	PUERTO 3.7 / LECTURA / A16	/RD
20	XTAL2	CRISTAL DE RELOJ EXTERNO / ENTRADA RELOJ EXTERNO.	-
21	XTAL1	CRISTAL DE RELOJ EXTERNO / COMUN RELOJ EXTERNO.	-
22	V _{ss}	COMÚN ALIMENTACIÓN.	-
23	VSS2		
24	P2.0	PUERTO 2.0 / DIRECCIÓN ALTA 8.	A8
25	P2.1	PUERTO 2.1 / DIRECCIÓN ALTA 9.	A9
26	P2.2	PUERTO 2.2 / DIRECCIÓN ALTA 10	A10
27	P2.3	PUERTO 2.3 / DIRECCIÓN ALTA 11.	A11
28	P2.4	PUERTO 2.4 / DIRECCIÓN ALTA 12.	A12
29	P2.5	PUERTO 2.5 / DIRECCIÓN ALTA 13.	A13
30	P2.6	PUERTO 2.6 / DIRECCIÓN ALTA 14..	A14
31	P2.7	PUERTO 2.7 / DIRECCIÓN ALTA 15.	A15
32	/PSEN	STROBE MEMORIA EXTERNA.	-
33	ALE	HABILITA LATCH DIRECCIONES / ENTRADA PROGRAMACIÓN EPROM.	/PROG
34	VSS2		
35	/EA	HABILITA ACCESO EXTERNO / ALIMENTACIÓN PROGRAMACIÓN EPROM.	V _{pp}
36	P0.7	PUERTO 0.7 / DIRECCIÓN BAJA 7.	AD7
37	P0.6	PUERTO 0.6 / DIRECCIÓN BAJA 6.	AD6
38	P0.5	PUERTO 0.5 / DIRECCION BAJA 5.	AD5
39	P0.4	PUERTO 0.4 / DIRECCION BAJA 4.	AD4
40	P0.3	PUERTO 0.3 / DIRECCION BAJA 3.	AD3
41	P0.2	PUERTO 0.2 / DIRECCION BAJA 2..	AD2
42	P0.1	PUERTO 0.1 / DIRECCION BAJA 1.	AD1
43	P0.0	PUERTO 0.0 / DIRECCION BAJA 0.	AD0
44	V _{cc}	ALIMENTACIÓN POSITIVA.	-

2.1 Descripción de la arquitectura del dispositivo puente H.

En la figura 2.6 se muestra un arreglo de cuatro interruptores conocido como puente “H”, este puente recibe una alimentación de DC y permite un control bidireccional de la corriente que circula por el motor.

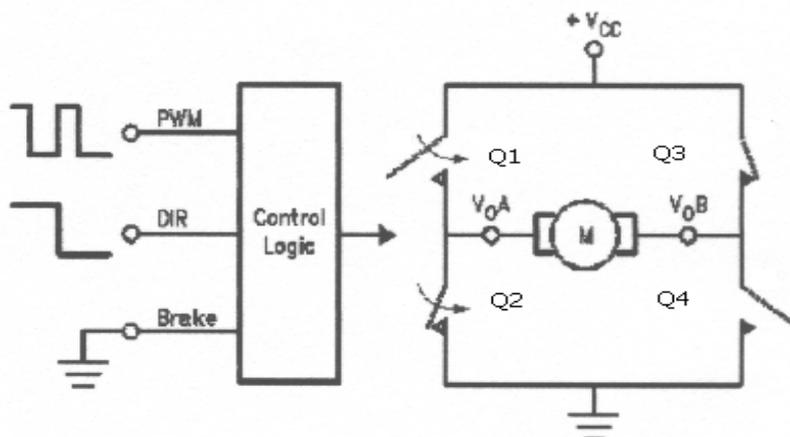


Figura 2.6 Puente H

Etapas de potencia.

Justificación.

Los requerimientos de potencia de nuestro trabajo están por encima de las soluciones completamente integradas para el control de motores de corriente continua, al superarse los 10 W que soporta esta solución.

En consecuencia, la etapa de potencia ha de ser construida con componentes activos y discretos y dimensionados éstos para atender los requerimientos previstos. La topología no ofrece alternativa, una vez decidido el controlador que la va a dirigir, el LM629, con salida PWM, por lo que se diseñará un puente en H.

Igualmente sucede con los elementos de conmutación, que entran en la gama de posibilidades de los MOSFET, no existiendo necesidad de una potencia mayor que requiriera IGBT o conmutadores controlados de mayor potencia.

El control del puente, proveniente del controlador, va a llevar el disparo de los MOSFET mediante "drivers" integrados del tipo IR2110, de "International Rectifiers", específico para este tipo de aplicaciones, que soporta entrada PWM y salidas complementarias referenciada a masa una y flotante la otra, para solventar los problemas inherentes al referenciado del disparo en las líneas de puerta no referenciadas a la masa de potencia, Q3 y Q1. También aporta una desactivación del puente por exceso de corriente que proviene del "sensado" de la corriente de la línea de potencia.

El proceso de selección de componentes del puente y su dimensionado se detalla posteriormente.

A continuación se resume el funcionamiento básico de esta topología.

Convertidor "Full-Bridge", puente completo (H).

Partiendo de una alimentación continua (potencia) aplicada sobre Q1 (+) y Q2 (-) y aplicando señales de disparo a los MOS FET, se pueden dar las siguientes situaciones:

1 Conducción de Q1 y Q4:

La carga inductiva se ve sometida a una tensión de magnitud igual a la de la fuente, invertida en signo. Por ella circula una corriente de sentido negativo respecto de la referencia.

2 Conducción de Q1 y Q3 o Q2 y Q4:

La carga resulta cortocircuitada, con tensión cero y sin circulación de corriente.

4 Conducción de Q3 y Q4 o Q1 y Q2:

Estas situaciones resultan prohibidas, porque tienen como consecuencia cortocircuitar la potencia.

Cuando aplicamos un control de puerta a los MOSFET en forma que ésta sea la misma para Q2 y Q3 e igual pero complementaria para Q1 y Q4, el circuito alternará el signo de tensión y corriente siguiendo el ciclo de disparo (ciclo de trabajo) sobre la carga. Al tratarse de una carga inductiva, y si el ciclo de trabajo fue calculado para permitir la carga y descarga del bobinado, a través de la carga circula una corriente media que excitará el motor de forma proporcional al tiempo en que resulten disparadas las líneas de MOSFET.

Con las combinaciones descritas anteriormente, fácilmente se ve que, actuando sobre el disparo, podemos regular el nivel de la corriente media de excitación del motor (velocidad) y su sentido (sentido de giro). Igualmente se puede para el motor, impidiendo que circule corriente por su bobinado. Este es el tipo de control que precisa nuestra aplicación.

Concretando, al disponer de un control de disparo en forma de magnitud y signo, lo que éste pretende es regular la velocidad mediante el ancho del pulso considerado respecto de una frecuencia fija (ciclo de trabajo) y su sentido de giro con la señal de signo.

Consideraciones.

Se fijará una magnitud máxima de la alimentación (potencia), que será la máxima potencia del motor que podamos situar como carga. Esta potencia máxima viene limitada por las prestaciones del tipo de MOSFET seleccionado. Se determinarán las dimensiones de diodos rápidos (Schotky) entre drenador y fuente de los MOS FET, en función de los picos de tensión por carga inductiva esperados.

Se fijará un ciclo de trabajo que también tendrá un máximo, del orden de kilo hercios, determinado por la capacidad de los MOSFET. Esta frecuencia limitará también el tipo de carga que se pueda situar, porque debe ser coherente con las características inductivas del motor, en el sentido de que sus bobinados sean capaces de seguir el ciclo de carga y descarga requerido. El ciclo de trabajo se fijará mediante el reloj del controlador LM629. Esta señal de control se conducirá al driver IR2110 a través de opto acopladores, para evitar interferencias entre el circuito lógico y el de potencia, aislándolos entre sí.

Se fijarán unos "tiempos muertos" en los cuales no exista excitación de puerta ninguna, durante las transiciones, para dar tiempo a los MOSFET a alcanzar el estado requerido de conducción o bloqueo, ya que su capacidad de entrada en puerta retrasa su respuesta.

De lo contrario se producirán combinaciones indeseadas. Se determinarán por las características de los MOSFET y se fijarán en la entrada de los "drivers" IR2110 mediante una red R-C. Al tratarse de nano segundos, estos tiempos no resultan significativos con respecto de la frecuencia de trabajo.

Se determinará el tipo y dimensionado de los disipadores de los MOS FET para garantizar la potencia máxima fijada, calculados sobre la impedancia térmica que suponga la conmutación. La realimentación del sistema tomará muestras de un "encoder" fijado sobre el rotor del motor, conducida al controlador LM629 mediante "line drivers" de línea diferencial.

Se tomará muestra del consumo de corriente del puente, mediante sensado sobre la línea de retorno de potencia, para controlar su magnitud máxima y media que será tratada y digitalizada por un interfase diseñado al efecto, para su lectura por el μC , con objeto de que éste determine los límites de funcionamiento esperado y active una protección si se ven superados.

Se establecerá un control de exceso de temperatura que se tomará sobre el disipador de los MOSFET y activará una interrupción del μC . Se diseñará el interfase que conduzca esta señal de interrupción.

2.2 Formas de señales del PWM.

El puente H, realiza la interfaz fácilmente con diferentes formas de señal PWM. Enseguida se describen las dos formas más populares de PWM.

1. *Simple señal de fase del PWM.* Esta consiste de una simple señal de una variable de ciclo obligado en la cual son en codificadas ambas *la dirección* y la información de *la amplitud* como se ve en la figura 2.6a.

Donde una señal PWM con un ciclo obligado del 50% representa el manejo del cero, ya que el valor de la red de voltaje (integrada sobre un período) entregado a la carga es cero.

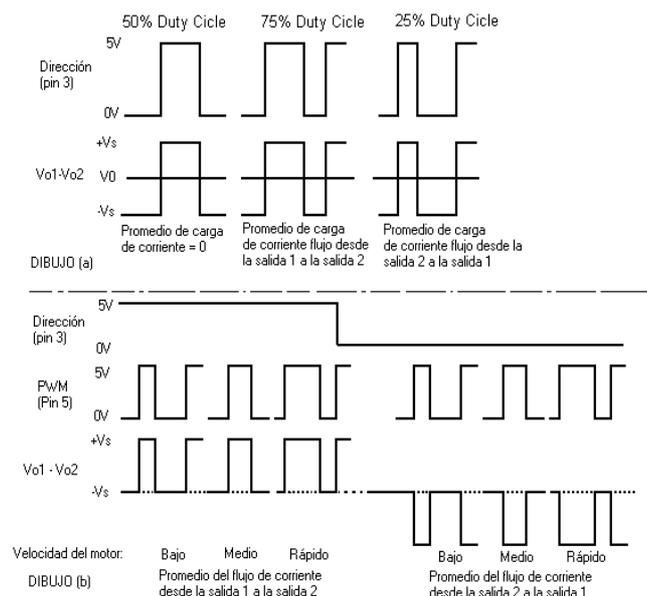


Figura 2.6. Dos tipos de formas de onda.

2. *Signo y magnitud de PWM*, consiste de una dirección separada y una señal de amplitud ver figura 2.6b. La señal de magnitud es modulada de ciclo obligado, y la ausencia de una señal de pulso (una lógica continua de nivel bajo), representa un manejo de cero. La corriente entregada a la carga es proporcional al ancho de pulso.

Limitación de corriente.

El circuito de protección para la limitación de corriente será incorporado dentro del diseño del puente H. Pero como cualquier dispositivo de potencia, es importante que se considere el efecto de la corriente que surge sustancialmente a través del dispositivo. El sensado de corriente se tomara a través de una resistencia de 0.1ohmio, este estará formado por un “AO” del tipo LM 324 y este conectara a un comparador del tipo LM 311.

4.10. Descripción del sistema de alarmas.

Opciones seleccionadas:

ADC0808, Conversor Analógico-Digital de 8 Bit , de la casa “National semiconductor”.

Justificación.

Este es un circuito conversor A/D de 8 entradas, de la cuales sólo utilizaremos una para controlar la intensidad media y la intensidad eficaz. Pudiendo haber elegido otro aprovechando más sus características, la elección de éste se basa en:

- Su fácil interface con todos los microprocesadores.
- El no requerimiento de ajuste de cero o escala completa.
- Su rapidez y gran precisión en la conversión.

En la configuración elegida para el esquema general recibe una señal analógica proveniente del filtro paso-bajo (intensidad media) y otra del conversor a valor eficaz AD536 (intensidad eficaz). Para elegir que señal queremos controlar en cada momento una de las entradas de selección las conectamos con el Puerto 1 del micro, para que con un valor 0 ó 1 seleccione la señal analógica de la que queremos obtener su valor en digital por la salida de conversor A/D. Cuando esta señal supere los valores establecidos como máximos, activará el microprocesador.

AD536A, Integrated Circuit True RMS-to-DC Converter, de la casa “Analog Devices”.

Justification.

Se eligió la configuración del AD536A de la *figura 7* (ver ANEXO A de este circuito). La principal causa es:

- Reduce con el filtro exterior de dos polos el rizado de salida alterna (AC), consiguiendo una gran mejoría en la precisión de la medida que nos compensa el aumento del tiempo que se produce en la realización de la conversión (RMS a DC).

Una vez realizada la elección, explicaremos más detalladamente el funcionamiento:

El convertidor de valor eficaz a continuo, recibe en su entrada una señal con componente AC, mandada por el amplificador operacional en montaje diferencial, en su interior en el divisor cuadrático medio el AD536 toma continuamente la muestra para valorar su tensión eficaz y trasladar ese valor a la salida en continua (DC), a través del filtro externo de dos polos conseguiremos que ese valor en la salida tenga el mínimo error, eso es, que lleve una mínima componente de AC.

OP07, Ultralow FOCET Voltage Operational Amplifier, de la casa “Analog Devices”.

Justificación.

Este dispositivo se utilizara en dos configuraciones, como amplificador diferencial y como filtro paso bajo.

a) amplificador diferencial.

El amplificador operacional diferencial está conectado a una resistencia de sensado, la cual pretende, a través de su bajo valor, no influir en el valor de corriente que esté entregando el puente H en todo momento (recordar $\text{Intensidad} = \text{Tensión} / \text{Resistencia}$), apreciando también por esta fórmula que en ella habrá un pequeño valor de caída de tensión (0.2 voltios a 1.5 voltios * ver cálculos del amplificador operacional diferencial*). La elección del amplificador operacional en montaje diferencial es para amplificar esta señal, intentándola llevar a unos valores de tensión mayores con los que se pueda trabajar mejor, influyendo menos el ruido y las capacidades parásitas.

b) Filtro paso bajo.

Hemos elegido un filtro paso-bajo activo de primer orden porque es muy estable y no tiene peligro de entrar en oscilaciones.

Este circuito pretende eliminar la componente de alterna que lleva la señal entregada por el amplificador operacional diferencial, esto se realiza calculando el filtro paso-bajo para una frecuencia de corte de 1 hertzio, así obtendremos en su salida un nivel de tensión continua correspondiente a la corriente media que demanda el motor, por la relación de ganancia ofrecida por el amplificador operacional diferencial, pues a este filtro le configuramos en montaje seguidor de emisor para que su ganancia sea 1, no modificando el nivel de tensión de la señal entregada por el amplificador operacional.

Alarma por temperatura.

Este circuito está basado en la utilización de un interruptor térmico, su funcionamiento sería el siguiente:

a) Si la temperatura es inferior a $(70 \pm 5) ^\circ\text{C}$, el interruptor térmico permanecerá abierto, llegándole a la patilla de interrupción del microprocesador INT1 activa nivel bajo, un nivel alto (5 voltios), lo que significa que el microprocesador no indicará un fallo de temperatura.

b) Si la temperatura del disipador es superior a $(70 \pm 5) ^\circ\text{C}$, el interruptor térmico cerrará su contacto introduciendo un nivel bajo por la patilla de interrupción INT1 del microprocesador, dejándola activada, indicando que existe temperatura elevada en los mosfets.

Esta señal llegará a la patilla de interrupción INT1 a través de un Octoacoplador para separar la señal del circuito de potencia.

Hasta que no caiga la temperatura $50 ^\circ\text{C}$ (ver características del interruptor térmico en ANEXO A) no volverá a su posición natural de normalmente abierto.

Funcionamiento del esquema completo de alarmas y errores.

El fin perseguido por este esquema es el de conseguir que a un microprocesador le llegue en todo momento el valor de corriente medio y eficaz, que demande el puente H para que cuando estos valores superen a los establecidos como máximos, el microprocesador a través de su bit de interrupciones atienda al error y mande las correspondientes órdenes al LM629 para que regule el motor a un funcionamiento establecido como correcto.

Para obtener una más clara explicación vamos a dividir las señales controladas:

- Alarma de temperatura: colocando un interruptor térmico normalmente abierto en el disipador del puente H, activará por la configuración de circuito únicamente el bit de interrupciones, cuando la temperatura sea superior a la especificada.

- Alarma por la superación de la intensidad media y alarma por superación de la intensidad eficaz: ambas parten de un amplificador operacional en montaje diferencial, que pretende aumentar la tensión que recae en la resistencia de sensado del puente H y por una relación tensión-corriente obtener el valor de esta última.
- Respecto a la superación de la intensidad media: la señal proveniente del amplificador operacional diferencial es conducida a la entrada de un filtro paso-bajo con frecuencia de corte de 1 Hertzio, con el propósito de eliminar el valor de alterna de la señal de entrada y obtener en la salida del filtro el valor de tensión de continua medio que está entregando actualmente el puente H. Esta señal la recibe un conversor analógico-digital para tratarla y obtener un valor digital con el que poder atacar al microprocesador.
- Respecto a la superación de la intensidad eficaz: la señal recibida del amplificador operacional diferencial se conecta a la entrada de un conversor a valor eficaz, que a su vez se conecta a la entrada analógica de un conversor analógico digital, para que éste la convierta en un valor digital ejecutable por el microprocesador. A través de una línea de dirección, escogeremos mediante el microprocesador si queremos leer intensidad eficaz o intensidad media.

Representación de las alarmas.

Esta se llevara a cabo por medio de un visualizador de 7 segmentos, el cual se describirá a continuación.

LM7446A (DECODIFICADOR BCD A 7 SEGMENTOS).

DESCRIPCIÓN GENERAL.

El 7446A y 7447A se caracterizan por salidas activas a nivel bajo diseñadas para manejar leds de ánodo común o indicadores incandescentes directamente; y el 7448 se caracteriza por salidas activas a nivel alto para manejar buffers de lámparas o leds de cátodo común. Todos los circuitos tienen controles de entrada/salida de puesta en blanco y una entrada de pruebas de lámparas.

DESCRIPCIÓN DE FUNCIONAMIENTO.

El decodificador/excitador BDC a siete segmentos consta de puertas NAND, buffers de entrada y siete puertas AND-OR INVERT. Esto ofrece salidas de corriente de absorción (SINK CURRENT) elevada con un bajo activo para excitar indicadores directamente. Siete puertas NAND y un excitador se hallan conectadas por parejas para poder ofrecer los datos BCD y su complemento a las siete puertas decodificadoras AND-OR-INVERT. La puerta restante NAND y tres buffers de entrada proporcionan la entrada de lámpara de prueba, entrada de BLANKING salida de RIPPLE-BLANKING y entrada RIPPLE BLANKING.

El circuito acepta información BCD (decimal codificado binario) de 4 bits y dependiendo del estado de las entradas auxiliares, decodifica estos datos para atacar un indicador de siete segmentos. Los niveles de salida lógica-positiva, así como las condiciones requeridas en las entradas auxiliares, se indican en la tabla de verdad. Las configuraciones de salida del decodificador/excitador están diseñadas para aguantar las tensiones relativamente elevadas requeridas por los indicadores de siete segmentos. Las salidas aguantan 15 voltios con una corriente inversa máxima de $250\mu\text{A}$. La presentación visual de entradas BCD por encima de 9 son símbolos únicos para denotar las condiciones de entrada.

El decodificador/excitador lleva incorporados control automático de CERO-BLANKING en el flanco posterior y/o anterior (RBI y RBO). La prueba de lámpara (LAMP TEST LT) de estos tipos puede realizarse en cualquier momento en que el nudo BI/RBO esté en nivel alto. Ambos contienen una entrada para contrarrestar el BLANKING (BI), que puede utilizarse para controlar la “intensidad de la lámpara” o para inhibir las salidas.

NOTAS:

BI/RBO está cableada AND sirviendo como entrada de BLANKING (BI) y/o salida RIPPLE-BLANKING (RBO). La entrada de BLANKING (BI) debe estar abierta o mantenida a un nivel alto cuando se deseen funciones de salida 0 a 15, y la entrada de RIPPLE-BLANKING (RBI) debe estar abierta o a nivel alto si no se desea el BLANKING de un 0 decimal. X=la entrada puede ser alta o baja.

Cuando se aplica un nivel bajo a la entrada de BLANKING (condición forzada) todas las salidas de segmento se van a nivel bajo independientemente del estado de cualquier otra condición de entrada.

Cuando la entrada de RIPPLE-BLANKING (RBI) y las entradas A, B, C, y D esté a nivel bajo, con la entrada de prueba de lámpara a nivel alto, todas las salidas de segmento se van a nivel alto y la salida de RIPPLE-BLANKING (RBO) se pone a nivel bajo (condición de respuesta).

Cuando la entrada de BLANKING/salida de RIPPLE-BLANKING (BI/RBO) esté abierta o mantenida a nivel alto y se aplique un nivel bajo a la entrada de prueba de lámpara (LAMP TEST), todas las salidas de segmento se ponen a nivel bajo.

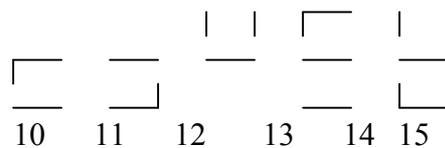
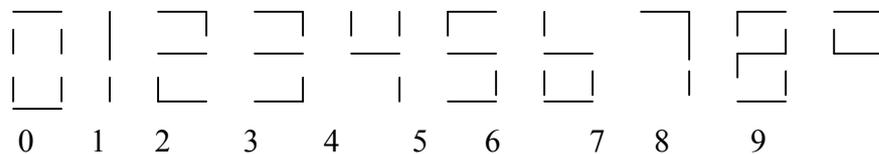
Todas las demás combinaciones que no están recogidas en las notas, se refieren al funcionamiento normal del circuito para un valor de entrada en BCD tenemos en la salida representado decimalmente mediante un display ese valor.

DECIMAL O FUNCIÓN	ENTRADAS						E/S	SALIDAS							NOTA
	LT	RBI	D	C	B	A	BI/RBO	a	b	c	d	e	f	g	
0	1	1	0	0	0	0	1	0	0	0	0	0	0	1	A
1	1	X	0	0	0	1	1	1	0	0	1	1	1	1	A
2	1	X	0	0	1	0	1	0	0	1	0	0	1	0	A

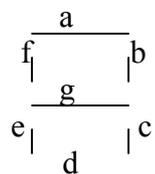
3	1	X	0	0	1	1	1	0	0	0	0	1	1	0	A
4	1	X	0	1	0	0	1	1	0	0	1	1	0	0	A
5	1	X	0	1	0	1	1	0	1	0	0	1	0	0	A
6	1	X	0	1	1	0	1	1	1	0	0	0	0	0	A
7	1	X	0	1	1	1	1	0	0	0	1	1	1	1	A
8	1	X	1	0	0	0	1	0	0	0	0	0	0	0	A
9	1	X	1	0	0	1	1	0	0	0	1	1	0	0	A
10	1	X	1	0	1	0	1	1	1	1	0	0	1	0	A
11	1	X	1	0	1	1	1	1	1	0	0	1	1	0	A
12	1	X	1	1	0	0	1	1	0	1	1	1	0	0	A
13	1	X	1	1	0	1	1	0	1	1	0	1	0	0	A
14	1	X	1	1	1	0	1	1	1	1	0	0	0	0	A
15	1	X	1	1	1	1	1	1	1	1	1	1	1	1	A
BI	X	X	X	X	X	X	0	1	1	1	1	1	1	1	B
RBI	1	0	0	0	0	0	0	1	1	1	1	1	1	1	C
LT	0	X	X	X	X	X	1	0	0	0	0	0	0	0	D

DESIGNACIONES NUMÉRICA.

VISUALIZACIÓN RESULTANTE



Donde:

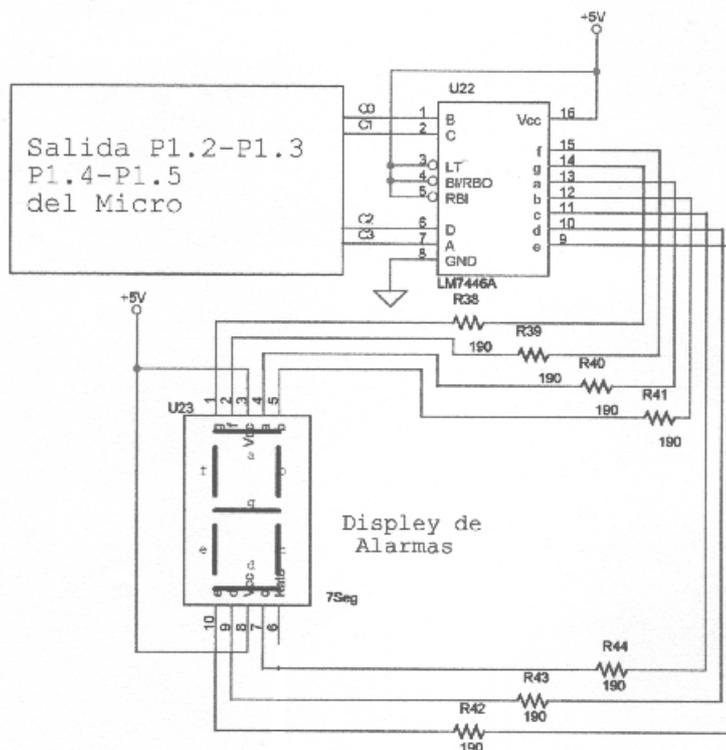


CIRCUITO DE VISUALIZACIÓN DE ALARMAS.

En primer lugar conectamos LT, RBI y BI/RBO a VCC (+5 Voltios), para tenerlas a un nivel alto, necesario para el requerido funcionamiento en BCD del 7446. Las entradas A, B, C y D reciben las salidas del puerto 1 del microprocesador, el cual, escribirá en ellas ciertas combinaciones binarias según el error o el buen funcionamiento del esquema general, quedando representadas en un visualizador óptico (display), en un código más inteligible para el hombre. Seguidamente presentamos la tabla de códigos:

VISUALIZACIÓN	SITUACIÓN
0	No hay problemas.
1	Trayectoria completa con éxito.
2	Error de desbordamiento de datos.
3	Error de posición.

En la siguiente figura se puede ver el diseño utilizado.



Las diferentes patillas de un Display de 7 segmentos son las siguientes:

Patilla 1-	Corresponde al segmento g.	Patillas 3 y 8-	van conectadas a Vcc.
Patilla 2-	“ “	f.	Patilla 6-es donde conectan todos los anodos.
Patilla 4-	“ “	a.	
Patilla 5-	“ “	b.	
Patilla 9-	“ “	d.	
Patilla 10-	“ “	e.	

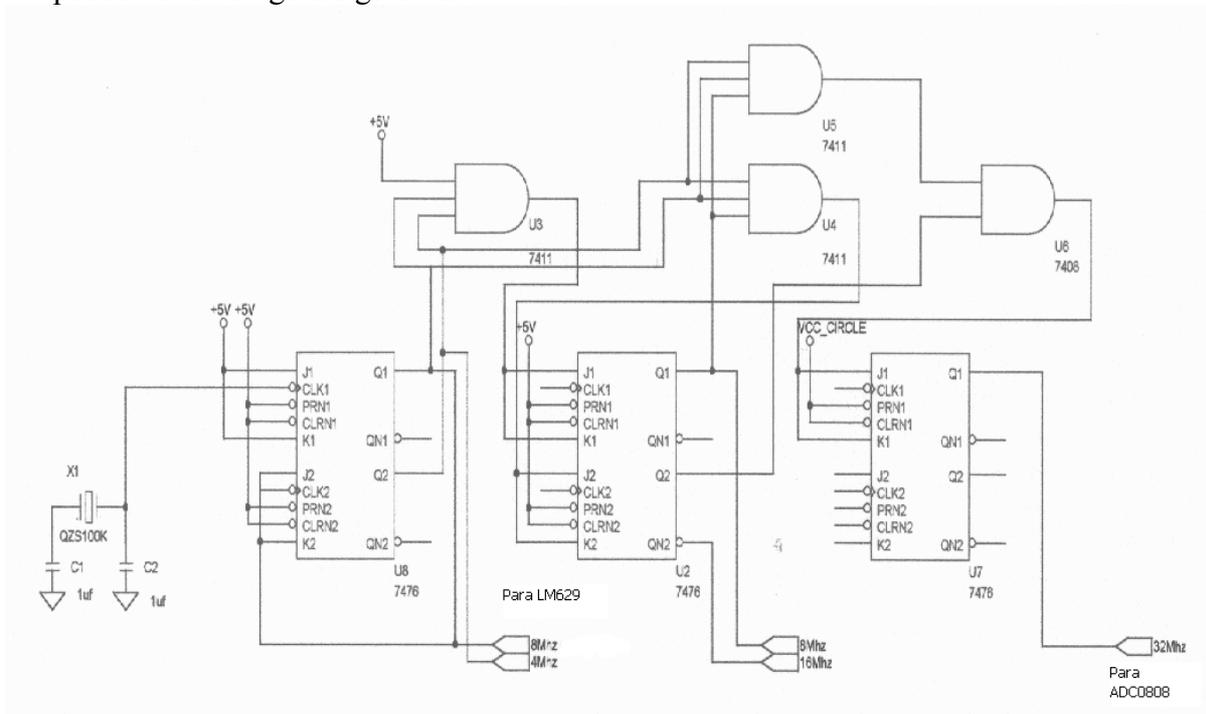
4.11. Circuitos de reloj.

Partiendo del preámbulo de la obtención de un cristal de cuarzo de 16 MHz para atacar al microprocesador, debemos obtener una señal de reloj de 8 MHz para el LM628 y de 500KHz para el convertor A/D ADC0808 para hacer lo propio.

Para cumplir estos objetivos y con nuestros conocimientos, podemos apreciar que todas las frecuencias a conseguir son submúltiplos en potencia de dos de la principal.

Por ello, cabe pensar en la utilización de contadores para realizar una división en frecuencia, o una segunda forma, basada en el manejo de básculas para conseguir el mismo objetivo.

Nosotros nos hemos decantado por ésta última, pensando en la mejor utilización de un solo tipo de integrado como elemento principal la báscula (flip-flop 7476). Como se puede ver en la figura siguiente.



Su funcionamiento será:

Salida del contaje codificado				
16	8	4	2	1
0	0	0	0	0
0	0	0	0	1
0	0	0	1	0
0	0	0	1	1
0	0	1	0	0
0	0	1	0	1
0	0	1	1	0
0	0	1	1	1
0	1	0	0	0
0	1	0	0	1
0	1	0	1	0
0	1	0	1	1
0	1	1	0	0
0	1	1	0	1
0	1	1	1	0
0	1	1	1	1
1	0	0	0	0
1	0	0	0	1
1	0	0	1	0
1	0	0	1	1
1	0	1	0	0
1	0	1	0	1
1	0	1	1	0
1	0	1	1	1
1	1	0	0	0
1	1	0	0	1
1	1	0	1	0
1	1	0	1	1
1	1	1	0	0
1	1	1	0	1
1	1	1	1	0
1	1	1	1	1

Tabla 1

El circuito consta de cinco flip-flop en serie, los cuales están interconectados a través de puertas AND, de forma parecida a un registro de desplazamiento. La señal de reloj se aplica a todos los flip-flop del contador en paralelo, con lo cual los estados de sus salidas cambian simultáneamente. Debido a esta disposición de la señal de reloj, este tipo de circuito se denomina “contador síncrono”.

La regla por la cual en la columna uno de la tabla 1 existe una alternancia continua de 1 y 0, se puede interpretar como la salida de un flip-flop JK que se dispara continuamente. El flip-flop JK, FF1, del esquema general se dispara cuando existe nivel 1 en sus entradas J y K. La columna de peso 2 de la tabla 1, también tiene una alternancia entre 1 y 0, sólo que ahora el cambio se produce cada dos impulsos de reloj. Comparando las secuencias de la columna del 2 con la de la 1 se establece que cuando la columna 1 está en lógica 1, a la llegada del próximo impulso de cómputo, la columna del 2 cambia su estado lógico. Esto se puede implementar conectando las dos entradas de FF2 a la salida Q de FF1, cuya señal producirá el disparo de FF2.

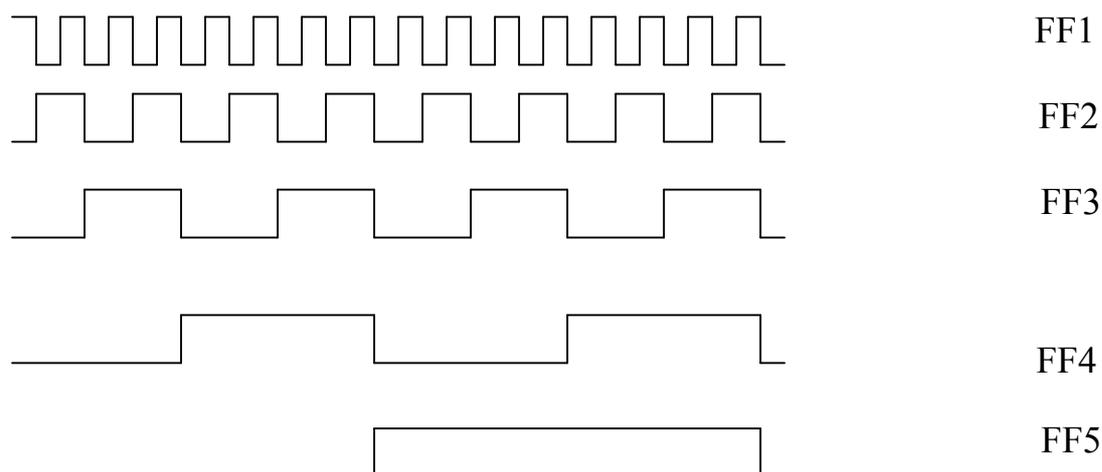


Figura 1

Se pudo comprobar en el diagrama de tiempo de la figura 1 que FF1 está a nivel 1, justo antes del impulso en que cambia de estado FF2. La alternancia del nivel 1 en FF1 produce el disparo y la apropiada alternancia de niveles en FF2.

Si analizamos el estado de FF3 con respecto a los de la columna de peso 4 de la tabla 1, se demuestra que para disparar FF3 es preciso que FF1 y FF2 estén a nivel 1, puesto que las columnas del 1 y del 2 se encuentran en estado 1 justo antes de que cambie el estado de la columna 4. La implementación del disparo de FF3 se logra, teniendo en cuenta lo anterior, mediante una puerta AND que tiene como entradas las salidas de FF1 y FF2 y como salida una línea que ataca las entradas J y K de FF3.

Un análisis similar demostraría lo mismo para FF4 y FF5, o cualquier otro flip-flop que se desee añadir: la entrada J y K de cualquier flip-flop está controlada por la salida de una puerta AND que tiene como entradas las salidas Q de los flip-flop anteriores.

Una vez realizado este diseño, conseguimos mejorarlo teóricamente respecto a la utilización de un menor número de circuitos integrados, lo que lleva consigo una mejora también respecto:

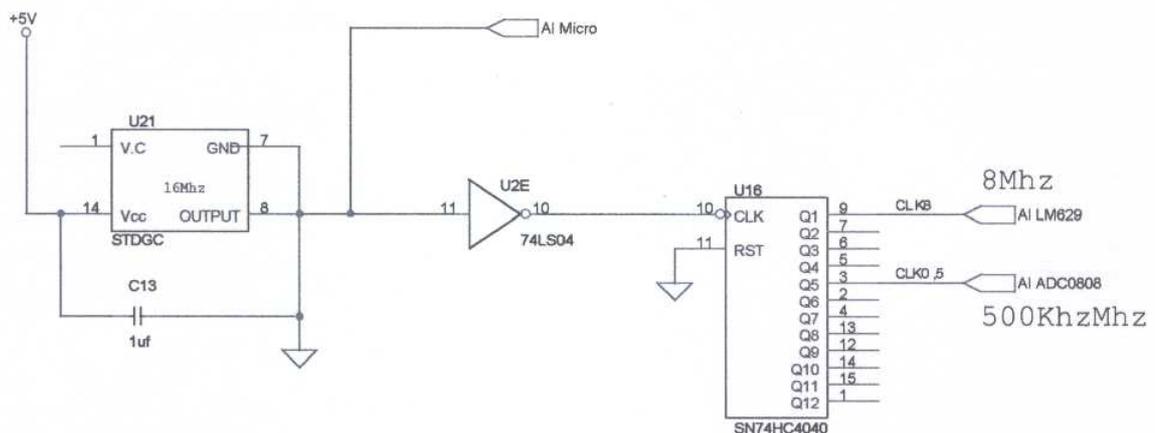
- Reducción de espacio en el diseño práctico.
- Reducción de cableado y por consiguiente menor posibilidad de falsos contactos

El circuito integrado que consigue realizar el mismo trabajo es el MM74HC4040.

Al ser este proyecto teórico, creemos más conveniente la utilización de esta última idea, pero damos al lector la posibilidad de las dos configuraciones por los distintos fines y/o predilecciones que pueda tener.

Ahora nuestro objetivo es explicar el circuito de reloj realizado con el MM74HC4040. Su funcionamiento es sencillo, conectando RESET a 0 Voltios el dispositivo está preparado para trabajar.

Conectaremos la salida del cristal de cuarzo de 16 MHz (reloj del microprocesador) a la entrada de reloj del MM74HC4040 para a partir de esta frecuencia, obtener las frecuencias de 8 MHz en la salida Q1 que realiza una división por 2 y 500KHz en la salida Q5 que realiza una división por 32, con las que se excitarán el LM629 y el convertidor analógico-digital ADC0808 respectivamente.



4.12. Reset del sistema.

Hemos optado por un circuito específico para el caso, que en principio era el TL7705, pero ha sido sustituido por un equivalente más sencillo.

En aplicaciones con microprocesadores con entrada de /RESET activa en bajo, el MAX7705 es compatible en funcionalidad y encapsulado con el TL7705, salvo que no necesita componentes externos. Aporta inmunidad ante "glitches" de alimentación y garantiza un "delay" en el reset adecuado en el arranque (power-up).

Monitoriza la alimentación del microprocesador o sistema digital implicado. El /RESET resulta valido para caídas de 1 V en la alimentación.

Provee un reset al sistema en arranque (power-up), por caída de la alimentación (power-down) y situaciones de disturbio. Cuando la alimentación baja del umbral establecido, la salida /RESET es nivel bajo y mantiene al microprocesador en reset. Esta salida /RESET volverá a nivel alto 280 mseg después de que la alimentación supere el umbral de reset. Afecta tanto al microprocesador como al MPID.

Para conseguir un reset manual, se implementa un pulsador que une con masa la línea de reset para ambos circuitos, útil en emergencias o situaciones recomendadas por el indicador de errores.

5. Descripción de dispositivos.

Diodo SCHOTKY.

Como se ha comentado anteriormente es el encargado de amortiguar los picos de tensión, se usara el BAT49, que soporta una tensión V_{RRM} de 80V.

HIGH AND LOW SIDE DRIVER (IR2110).

Son los encargados de proporcionar el disparo inferior y superior de los MOSFET, la circuitería que acompaña al driver es típica para este tipo de montajes, teniendo en cuenta que los condensadores, resistencias y diodos utilizados son los adecuados para el disparo del MOSFET elegido.

- Diodo se utilizara el D1N4148.
- Resistencia utilizada de 22K.
- Condensador utilizados de 0.1 y 1uF.

LOW POWER QUAD OPERATIONAL AMPLIFIERS (LM324).

Utilizado como amplificador diferencial, amplificara la señal de sesado del puente H y la comunicara a un comparador que se detallara seguidamente.

VOLTAJE COMPARADOR (LM311).

La salida de este se aplicara a la función Shutdown del IR2110, esta función apagara los transistores hasta la llegada del siguiente pulso, lo que permite un control de corriente ciclo por ciclo.

QUAD 2-INPUT NAND GATE (74LS00).

Esta compuesta por puertas Nand, Son las encargadas de realizar las conexiones entre el LM629 salidas PWM y los Octo acopladores, están llevaran una resistencia a su salida para limetar la señal de los Octo acopladores.

HEX SHMITT TRIGGER (CD40106C).

Es un circuito complementario monolítico de Mos, construido con transistores de realce de tipo N y de P, todas sus entradas están protegidas contra el daño debido a las descargas estáticas, y es el encargado de transmitir la señas de salida del los Octo acopladores al circuito de potencia a través del los draiver IR2110.

OPTOISOLATORS LOGIC OUTPUT (H11L1).

Octo acopladores, son los encargados de aislar la parte de control con la de potencia. Un octo acoplador combina un dispositivo semiconductor formado por un foto emisor, un fotorreceptor y el camino por donde se transmite la luz. Todos estos elementos se encuentran dentro de un encapsulado del tipo DIP.

La señal de entrada es aplicada al foto emisor y la salida es tomada del foto rector. Los octo acopladores convierten una señal eléctrica en una señal luminosa modulada y la vuelven a convertir en una señal eléctrica, en ellos reside el aislamiento eléctrico que pueden establecer entre circuitos.

La serie de H11L1 tiene un detector de nivel alto con una velocidad optimizable junto a un diodo que emite infrarrojos de arseniuro de galio, la salida incorpora un disparador de Schmitt, que proporciona la histéresis para la inmunidad de ruido y formar el pulso, utiliza una salida de colector abierto para la flexibilidad máxima de su uso.

Algunas de sus características:

- Alta tarifa de datos.
- Impulsión compatible con microprocesadores.
- La salida compatible con la lógica hunder 16mA en el máximo 0.4V
- Histéresis con./desc. Garantizada en el umbral.
- Capacidad amplia del voltaje de fuente, compatible con todos los sistemas populares lógica.

QUAD DIFFERENTIAL LINE RECEIVERS (DS26LS32C).

QUAD HIGH SPEED DIFFERENTIAL LINE DRIVE (DS26LS31C).

Estos dos componentes forman una parte muy importante del circuito, aunque no están representados en el esquema, solamente hay una pequeña anotación, ahora se intentara realizar una pequeña aclaración ya que de ella puede partir el buen funcionamiento entre el circuito y el motor a través del Encoder.

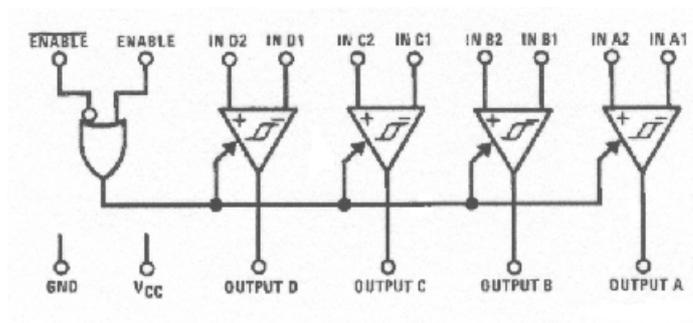
El DS26LS32C son receptores diferenciales de línea del cuadrángulo diseñados para motores, el Rs-422 y el Rs-423 (Son conductores de línea) se usan para la transmisión de datos digitales equilibrada y desequilibrada.

Tiene una sensibilidad de entrada de 200 mV sobre una gama de voltajes de entrada de $\pm 7V$.

Tiene incorporados los resistores pull-down que previenen la oscilación de la salida en los canales insusitados.

Las salidas de triple estado del TM, con la opción de la salida complementaria permiten para recibir directamente sobre un ómnibus de datos.

En la figura siguiente podemos ver el diagrama lógico.



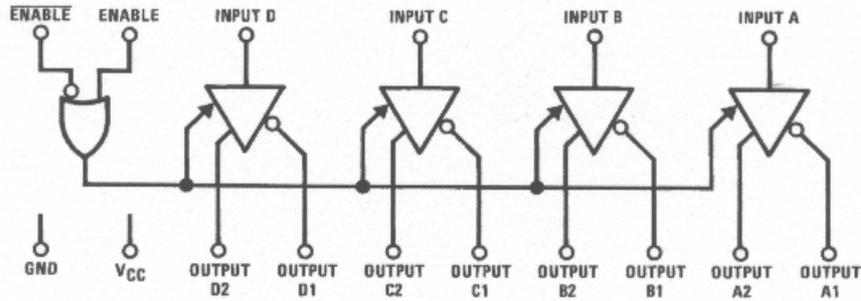
QUAD DIFFERENTIAL LINE RECEIVERS (DS26LS32C)

El DS26LS31C es un conductor de línea diferencial del cuadrángulo para líneas equilibradas de datos del excedente digital de la transmisión, se diseña para proporcionar la impulsión diferenciada unipolar a las líneas de conductor de doble retorcido o de alambres paralelos de la transmisión.

El circuito proporciona un permitir e inhabilita la función a los cuatro conductores de transmisión.

Ofrece salidas de triple estado y lógicamente las salidas complementarias de ANDel, las entradas son todas de LS compatibles.

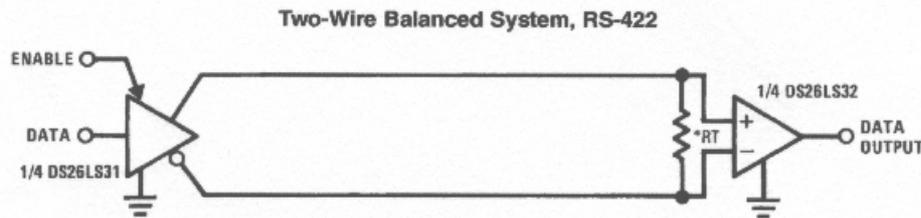
En la figura siguiente podemos ver el diagrama lógico.



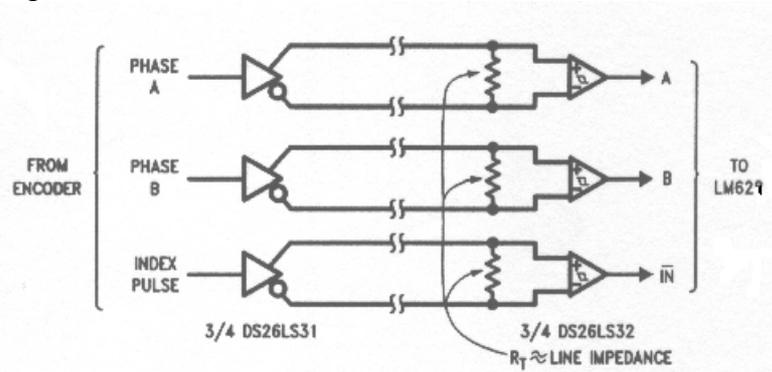
QUAD HIGH SPEED DIFFERENTIAL LINE DRIVE (DS26LS31C)

Entre estos dos dispositivos estará conectada una línea de impedancia, esta distancia va a depender de la situación que tengamos entre el circuito y el Encoder.

En la siguiente figura se puede ver la conexión de la línea RS-422 entre los dispositivos.

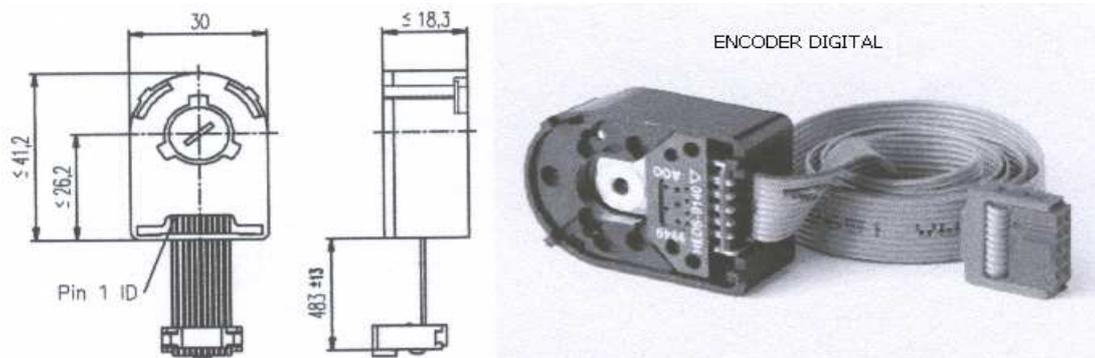


En la figura solo se muestra una línea, en nuestro proyecto a desarrollar estará compuesta por tres líneas que va a conectar el dispositivo LM629, entradas A, B, In del Encoder con las salidas del propio Encoder situado en el eje del motor, como podemos comprobar en la figura siguiente:



RT es la resistencia Pull-down, su valor va a depender de la distancia de la línea.

Existen líneas de impedancia ya constituidas formadas por el Encoder digital y el propio ds26ls31c, como se muestra en la figura siguiente.



IRPF250N.

Transistor MOSFET, son los que actúan como interruptores y dan paso al funcionamiento del motor DC.

AD536A, Convertidor de valor eficaz a continuo.

El AD536 computa directamente el valor eficaz de una señal compleja de entrada conteniendo componentes de alterna (AC) y continua (DC).

Las pérdidas de alimentación con las entradas conectadas no causan fallo en la unidad. Las entradas y salidas están completamente protegidas.

Tiene un ancho de banda de 300 khz con 3dB de error para señales de un nivel de 100 mV.

Descripción de funcionamiento.

El AD536 computa el verdadero valor medio cuadrático de una señal AC ó (AC + DC), dando en su salida un nivel equivalente en DC.

El único componente externo requerido para desarrollar las medidas con la completa precisión especificada es el condensador, el cual completa el periodo promedio. El valor de este condensador determina la precisión en baja frecuencia AC, nivel de rizado y el tiempo de establecimiento.

Diferentes configuraciones.

*nota:

Las figuras sobre las que se hará alusión se encuentra en el apartado de anexo A.

• **Figura 1** (Configuración con alimentación simétrica +VS y -VS)

Mide el valor eficaz de una señal AC y DC presente en la entrada. Mostrará un error para bajas frecuencias de entrada como una función del filtro capacitivo Cav.

• **Figura 3** (Configuración con alimentación referenciada a masa. Tensión de 5 a 36 Voltios)

Si se desea rechazar la entrada DC, un condensador es añadido en serie con la entrada. El condensador debe ser sin polaridad. Es aconsejable colocar en alimentación un condensador cerámico de 0.1 μF a masa tan cerca del dispositivo como sea posible.

La mayor limitación de esta conexión es que únicamente las señales AC pueden ser medidas, ya que el estado de la entrada diferencial se debe separar de masa para una operación correcta.

En la *figura 5*, se puede apreciar en la gráfica la elección de diferentes condensadores Cav según la frecuencia de trabajo.

• **Figura 7** (Configuración elegida para el proyecto)

El mejor método para reducir el rizado de salida es el uso de un POST-FILTER.

Si un único filtro polo es usado (C3 quitado, RX corto), y C2 es aproximadamente 2 X Cav, el rizado es reducido y el tiempo de establecimiento es incrementado (Ver *figura 8*, por ejemplo Cav = 1 μF , C2 = 2.2 μF y F = 60 Hz, el rizado se reduce del 10% al 0.3%, aunque el tiempo es incrementado por un factor de 3. Cav y C2 pueden ser reducidos para realizar la compensación deseada. Con dos polos PST-FILTER, el error de rizado se reduce aún bastante más y el tiempo apenas sube respecto al de un polo-filtro, por ello hemos elegido dicha configuración.

*Para todas las configuraciones se utiliza:

$$VR_{\text{rms}} = A_{\text{ug}} [V_{\text{in}}^2 / V_{\text{rms}}]$$

$$V_{\text{out}} = 2 R_2 * I_{\text{rms}} = V_{\text{in rms}}$$

OP07, Amplificador Operacional.

Amplificador operacional de alta gama porque nos ofrece:

- Bajísima dependencia de sus características frente a la temperatura.
- Bajísima tensión de offset.
- Bajísimo ruido.
- Bajísima deriva de la tensión de offset con la temperatura.

Se utilizara como se ha dicho anteriormente en dos configuraciones, como comparador y como filtro paso bajo.

MM74HC76, Dual JK Flip-flop With Set And Clear.

Estas básculas utilizan tecnología microCMOS, que permite:

- Alta velocidad (30 MHz mínimo).
- Bajo consumo.
- Alta inmunidad al ruido.

Todas las entradas están protegidas del peligro por descargas electrostáticas.

Descripción de funcionamiento.

Cada báscula (flip-flop) tiene entradas independientes J, K, PRESET, CLEAR y DE RELOJ, salidas Q y \bar{Q} . Estos dispositivos son atacados con flancos en la entrada de reloj y cambian de estado en la transición negativa del pulso de reloj. Clear y Preset son independientes del reloj y son ejecutados por un nivel lógico bajo en la entrada correspondiente.

Tabla de verdad.

ENTRADAS					SALIDAS	
PR	CLR	CLK	J	K	Q	\bar{Q}
L	H	X	X	X	H	L
H	L	X	X	X	L	H
L	L	X	X	X	L*	L*
H	H	↓	L	L	Q _o	Q _o
H	H	↓	H	L	H	L
H	H	↓	L	H	L	H
H	H	↓	H	H	TOGGLE	
H	H	H	X	X	Q _o	Q _o

*Notas:

- L → Nivel bajo.
- H → Nivel alto.
- X → No importa nivel.
- ↓ → Flanco de bajada.
- L* → Condición inestable, no se garantiza.

Q_0 y $\overline{Q_0}$ → Se refiere a que mantienen el nivel anterior con esa configuración.

TOGGLE → Báscula. Se produce un continuo cambio de nivel en cada salida, a la llegada del flanco del reloj.

Este dispositivo se utilizara en el circuito de señales de reloj para obtener diferentes frecuencias de reloj. Con el estará acompañado varias puertas AND de dos y tres entrada como son :

MM74HC11, AND de tres entradas.

MM74HC08, AND de dos entradas.

ADC0808, Converters with 8-channel Multiplexer.

El componente de adquisición de datos ADC0808, es un dispositivo CMOS monolítico con un convertidor analógico-digital de 8 bits, multiplexor de 8 canales y lógica de control compatible con microprocesador.

El convertidor analógico-digital de 8 bits utiliza aproximaciones sucesivas como técnica de conversión. Éste tiene como característica un comparador estabilizado con un chokeado de alta impedancia, un divisor de tensión con $256R$ con un árbol de conmutación analógico y un registro de aproximaciones sucesivas. El multiplexor de 8 canales puede acceder directamente a cualquiera de las 8 señales analógicas por separado.

El dispositivo elimina la necesidad de ajuste de 0 y escala completa externos. Fácil conexión con microprocesadores, se da por los multiplexores latcheados y sus entradas de direcciones decodificadas y salidas latcheados TTL-Triestado.

El diseño del ADC0808 ha sido optimizado incorporando los aspectos más deseables de varias técnicas de conversión analógico-digital.

El ADC0808 ofrece: alta velocidad, influencia mínima de la temperatura sobre su funcionamiento y consumo de potencia mínimo.

Descripción funcional.

El dispositivo contiene un multiplexor de 8 señales analógicas por separado. Una particular entrada del canal es seleccionada usando el decodificador de direcciones. (Ver tabla 1, que muestra el estado de las líneas para seleccionar cualquiera de las entradas).

Seleccionando canal analógico	Líneas de dirección		
	C	B	A
IN0	L	L	L
IN1	L	L	H
IN2	L	H	L
IN3	L	H	H
IN4	H	L	L
IN5	H	L	H
IN6	H	H	L
IN7	H	H	H

*** Tabla 1**

La monotonidad es particularmente importante en el control de sistemas con realimentación en lazo cerrado. Una relación no monótona puede producir oscilaciones, que serán catastróficas para el sistema.

Un proceso de conversión se verá interrumpido al recibir un nuevo pulso de comienzo de conversión. La conversión continua se puede realizar llevando la salida final de conversión (EOC) a la entrada (SC). Si se usa de este modo, un pulso externo de comienzo de conversión se deberá aplicar después del encendido.

El comparador de chopeado estabilizado, convierte la señal de entrada DC (continua) en una señal AC (alterna), esta señal se lleva luego a un amplificador AC de alta ganancia y se le restaura el nivel DC. Esta técnica limita la deriva de los componentes del amplificador, ya que lo que se deriva es un componente DC, que no se pasa por el amplificador de AC. Esto hace que el convertidor entero analógico-digital sea extremadamente insensible a la temperatura, a fallos con el tiempo y errores por el offset de entrada.

SN74HC4040, Divisor de frecuencia.

Este dispositivo nos proporcionara las frecuencias requerida en nuestro sistema

STDGC 16Mhz.

Este dispositivo nos proporciona la señal de 16Mhz para nuestro sistema.

LM7476A, Decodificador BCD de 7 segmentos.

Este dispositivo, es el encargado de proporcionar las señales al Display de 7 segmentos, se caracteriza por tener las salida a nivel bajo.

DISPLAY 7 Segmentos, de anodo comun.

En el se representaran los errores producidos en nuestro sistema, cada numero significara un error determinado.

TL7705A, μ P Power-Supply Monitor with reset.

Este sistema en aplicaciones con microprocesadores con entrada de reset, se utilizara para efectuar el reset a nuestro circuito.

TERMICO To220 (212-829)Normally closed.

Este dispositivo esta basado en un interruptor térmico, actúa por el aumento de temperatura, se utiliza la serie 212-829 en el cual el contacto térmico esta normalmente cerrado.

MGDI-18 : 18W POWER (+5V) Y MGDI-19 : 20W POWER (-+15V).

Dadas las especificaciones de nuestro trabajo, se tendrá una alimentaron de 25v y esta se aprovechara a través de los convertidores utilizados para la alimentación de todo el sistema.

Las características de operación de los diferentes elementos, se pueden consultar en el Anexo A. (Cd-rom adjunto).

6. CALCULO DEL SISTEMA.

Especificaciones del equipo.

En este apartado se van ha exponer algunas especificaciones de los distintos dispositivos y teoría para el diseño y calculo del puente H y disipación, como son:

- Propagación del calor.
- Formas de transmitir calor.
- Resistencias térmicas.
- Temperaturas.
- Potencia disipada.

Alimentación del puente H. (de 10 a 65v).

Corriente max (8A).

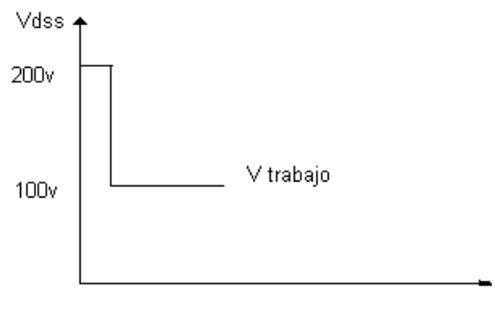
Corriente min (4A).

Utilizaremos el **Mosfet IRFP 250N**.

Características (Ver anexo B).

Vds 200V	Tj -55 / +175 °C	tr 43ns
Rds(on) 0.075 Ω	Rjc 0.7 °C/W	tf 33ns
Id 30A	Rcs 0.24 °C/W	
Cápsula TO-247AC	Rja 40°C/W	

Consideramos una tensión de trabajo de 50% menos de la tensión del Mosfet de Vdss de 200V.



$$V_{trabajo} = V_{dss} \times 50\% = 200 \times 50\% = 100v$$

El montaje estará especificado para diversos motores, hasta una potencia max de 1500W. Donde la corriente max que circulara por el puente H, será aproximadamente de.

$$P_{max} = V_t \times I$$

$$I = \frac{P_{max}}{V_t} = \frac{1500}{100} = 15A$$

Si la alimentación especificada es de 10 a 65V, la Potencia máxima estará comprendida.

- Para 10V $P_{max} = V_{mi} \times I = 10 \times 15 = 150W$

- Para 65V $P_{max} = V_{ma} \times I = 65 \times 15 = 975W$

Realizado estos cálculos podemos asegurar que el Mosfet, estará trabajando por debajo de la Pmax que admite, con un margen de seguridad “DERATING” del 50%.

Otra consideración que hay que tener en cuenta, es cuando el Mosfet trabaja con cargas inductivas (Motores).

Las pérdidas de conmutación de los Mosfet, son muy difíciles de calcular por las capacidades que posee y en algunos casos estos cálculos son algo genéricos.

Las bobinas se oponen al cambio brusco de corriente y producen picos de tensión $V=L \frac{di}{dt}$ (escalón infinito de tensión).

Vds tendrá que ser mayor que Vcc para que cuando se produzca el pico este no afecte a la destrucción del Mosfet, Vds es facilitada por el fabricante en las hojas de características.

Rds(on) cuanto mas pequeña sea menor pérdidas de conducción tendremos.

Para amortiguar los picos de tensión se colocaran un diodo Schotky como red Snubber, este ira colocado en paralelo entre los terminales Drain (Drenador) y Source (Fuente) del Mosfet.

Temperatura de funcionamiento (0 a 70°C).

Al trabajar con Mosfet, estos provocan unas pérdidas de potencia y estas serán durante el tiempo de conmutación.

- De conducción a Bloqueo (tr o trr).....Grandes.
- De Bloqueo a Conducción (trf).....Pequeñas.

Las pérdidas de potencia se traducen en calor (Se eleva la energía cinética de las moléculas y estas producen un aumento de temperatura sobre el dispositivo) si esta aumenta en exceso puede producir daños en el dispositivo hasta la destrucción.

Esta consideración hace necesario que tengamos presente la refrigeración, ya que un incremento excesivo puede provocar la reducción de la vida útil del dispositivo o su destrucción, por ello es necesario refrigerar (evacuar calor).

En principio la evacuación del calor depende del encapsulado del componente, pero a menudo es insuficiente, hay que elegir un encapsulado adecuado o suplementar este dotándolo de mayor superficie y volumen.

La capacidad térmica de un dispositivo es la medida de ritmo de cambio de la energía térmica en temperatura. Da una idea del ritmo de cambio de temperatura que puede soportar el componente.

$c = \text{calor específico}$

$$C_{tn} = c \times m$$

$m = \text{masa}$

Los disipadores mejoran la capacidad térmica de los dispositivos, dado que el calor se genera en el interior del componente y es preciso evacuarlo al exterior.

Introducción a los disipadores de calor.

El estudio térmico de los dispositivos de potencia es fundamental para un rendimiento óptimo de los mismos. Esto es debido a que en todo semiconductor, el flujo de la corriente eléctrica produce una pérdida de energía que se transforma en calor.

El calor produce un incremento de la temperatura del dispositivo. Si este incremento es excesivo e incontrolado, inicialmente provocará una reducción de la vida útil del elemento y en el peor de los casos lo destruirá.

En Electrónica de Potencia la REFRIGERACIÓN juega un papel muy importante en la optimización del funcionamiento y vida útil del semiconductor de potencia.

Propagación del calor.

En todo semiconductor el flujo de la corriente eléctrica produce una pérdida de energía que se transforma en calor. Esto es debido al movimiento desordenado en la estructura interna de la unión.

El calor elevará la energía cinética de las moléculas dando lugar a un aumento de temperatura en el dispositivo; si este aumento es excesivo e incontrolado provocará una reducción de la vida útil del dispositivo y en el peor de los casos su destrucción.

Es por ello que la evacuación del calor generado en el semiconductor es una cuestión de gran importancia para asegurar el correcto funcionamiento y duración del dispositivo.

La capacidad de evacuación del calor al medio ambiente podrá variar según el tipo de cápsula pero en cualquier caso será demasiado pequeña, por lo que necesita una ayuda adicional para transferir el calor disipado mediante un dispositivo de mayor volumen y superficie conocido como disipador de calor, el cual hace de puente para evacuar el calor de la cápsula al medio ambiente.

Para este proceso existen varias formas de transmitir el calor al exterior (Al ambiente).

La experiencia demuestra que el calor producido por un foco calorífico se propaga por todo el espacio que lo rodea. Esta transmisión del calor puede producirse de tres formas:

1.- CONDUCCIÓN:

Es el principal medio de transferencia de calor. Se realiza por la transferencia de energía cinética entre moléculas, es decir, se transmite por el interior del cuerpo estableciéndose una circulación de calor. La máxima cantidad de calor que atravesará dicho cuerpo será aquella para la cual se consigue una temperatura estable en todos los puntos del cuerpo.

En este tipo de transmisión se debe tener en cuenta la conductividad térmica de las sustancias (cantidad de calor transmitido por unidad de tiempo, superficie, gradiente de temperatura).

2.- CONVECCIÓN:

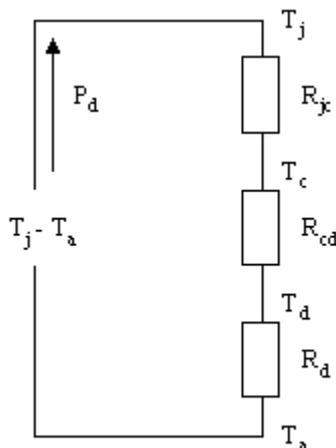
El calor de un sólido se transmite mediante la circulación de un fluido que le rodea y este lo transporta a otro lugar, a este proceso se le llama convección natural. Si la circulación del fluido está provocada por un medio externo se denomina convección forzada.

3.- RADIACIÓN:

El calor se transfiere mediante emisiones electromagnéticas que son irradiadas por cualquier cuerpo cuya temperatura sea mayor a cero grados Kelvin. El estado de la superficie influye en gran medida en la cantidad de calor radiado. Las superficies mates son más favorables que las pulidas y los cuerpos negros son los de mayor poder de radiación, por este motivo se efectúa un ennegrecimiento de la superficie radiante. La transferencia de calor por radiación no se tiene en cuenta puesto que a las temperaturas a que se trabaja ésta es despreciable.

Parámetros que intervienen en el cálculo.

Para que un semiconductor disipe la potencia adecuada, hay que mantener la temperatura de la unión por debajo del máximo indicado por el fabricante.



El paso de la corriente eléctrica produce un aumento de la temperatura de la unión (T_j). Si ésta se quiere mantener a un nivel seguro, debemos evacuar al exterior la energía calorífica generada por la unión. Para que se produzca un flujo de energía calorífica de un punto a otro, debe existir una diferencia de temperatura. El calor pasará del punto más caliente al más frío, pero aparecen factores que dificultan este paso. A estos factores se les denomina resistencias térmicas.

Por lo tanto, aprovechando la ley de ohm realizamos la siguiente comparación eléctrica mostrada en la figura adjunta. Asemajaremos las temperaturas a tensiones, las resistencias térmicas a las resistencias óhmicas y el flujo de calor a una corriente eléctrica.

Por lo tanto, aprovechando la ley de ohm realizamos la siguiente comparación eléctrica mostrada en la figura adjunta. Asemajaremos las temperaturas a tensiones, las resistencias térmicas a las resistencias óhmicas y el flujo de calor a una corriente eléctrica.

Al igual que en un circuito eléctrico, se puede decir que:

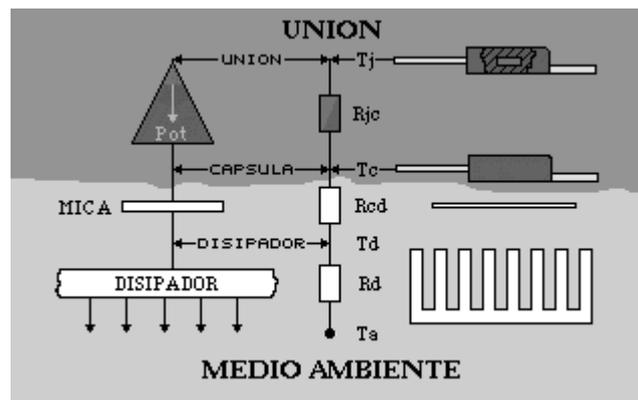
$$T_j - T_a = P \times R_{ja}$$

De la figura se obtiene la expresión:

$$P_d = \frac{T_j - T_a}{R_{jc} + R_{cd} + R_d}$$

Resistencias térmicas.

En la siguiente figura se muestra la igualdad entre el circuito equivalente de resistencias térmicas y los elementos en un montaje real:



R_{jc} = Resistencia unión – contenedor.

R_{cd} = Resistencia contenedor – disipador.

R_d = Resistencia del disipador.

T_a = Temperatura ambiente.

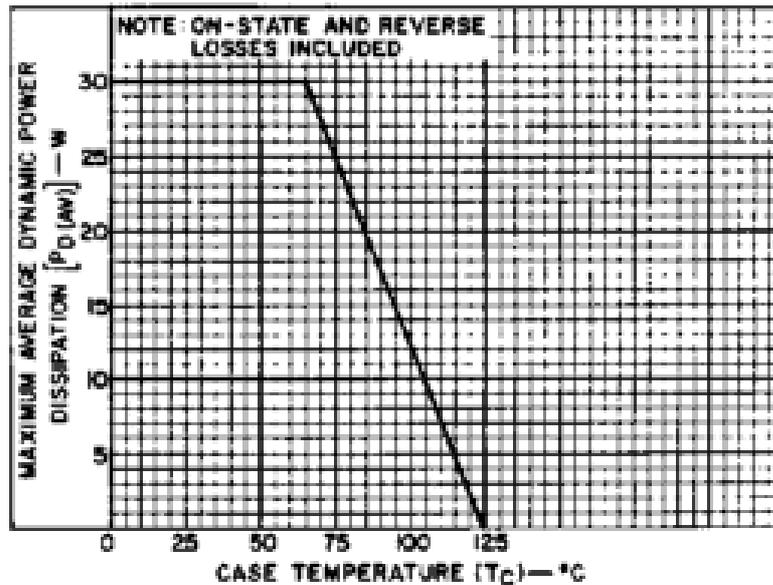
T_d = Temperatura del disipador.

T_c = Temperatura del contenedor.

T_j = Temperatura de la unión

Resistencia Unión - Cápsula (R_{jc}).

En este caso el foco calorífico se genera en la unión del propio cristal semiconductor, de tal forma que el calor debe pasar desde este punto al exterior del encapsulado. Generalmente este dato lo suministra el fabricante, y dependerá del tipo de cápsula del dispositivo. Aparecerá bien directamente o indirectamente en forma de curva de reducción de potencia. En la figura siguiente se muestra este tipo de curva.



Esta muestra la potencia en función de la temperatura de la cápsula. En ella la pendiente de la recta dada es la resistencia unión cápsula. La fórmula que se utiliza para el cálculo de esta resistencia es:

$$R_{jc} = \frac{T_{jmax} - T_c}{P_d}$$

Donde estos datos se obtienen de la curva de reducción de potencia, que será propia de cada dispositivo. Debemos de tener en cuenta que P_d es la dada por el fabricante y no la que disipará el dispositivo en el circuito. Normalmente T_c vale 25 °C.

Si tomamos de un manual los datos correspondientes del IRFP250N serán:

P_dmáx=214W.

T_jmáx =175 °C.

Sustituyendo estos valores en la siguiente ecuación, se obtiene el valor de la R_{jc}:

$$R_{jc} = \frac{T_{jmax} - T_c}{P_d} = \frac{175 - 25}{214} = 0.7^\circ C / W$$

y ésta es, precisamente, la R_{jc} indicada en los manuales para el IRFP250N.

Resistencia cápsula - Disipador (Rcd).

Es la resistencia térmica entre el semiconductor y el disipador.

Este valor depende del sistema de fijación del disipador y el componente, y del estado de planitud y paralelismo de las superficies de contacto, puesto que a nivel microscópico, solo contactan por unos puntos, quedando huecos de aire que entorpecen la transmisión del calor.

También depende del tipo de material que se interponga entre ambas superficies de contacto. Los elementos que se sitúan entre la cápsula y el disipador pueden ser de dos tipos:

- a. Pastas conductoras de calor, que pueden ser o no ser conductoras de la electricidad.
- b. Láminas aislantes eléctricas que se pueden emplear conjuntamente con siliconas conductoras de calor como mica, kelafilm, etc. También las hay conductoras de calor que no precisan pasta de silicona.

El tipo de contacto entre cápsula y disipador podrá ser:

- Directo.
- Directo más pasta de silicona.
- Directo más mica aislante.
- Directo más mica aislante más pasta de silicona.

El valor de esta resistencia térmica influye notablemente en el cálculo de la superficie y longitud que debe disponer la aleta que aplicaremos al dispositivo a refrigerar. Cuanto más baja es Rcd menor será la longitud y superficie de la aleta requerida.

Por ejemplo, para una cápsula TO.3 se tiene que con contacto directo más pasta de silicona la $R_{cd} = 0,12 \text{ } ^\circ\text{C/W}$, que con contacto directo $R_{cd} = 0,25 \text{ } ^\circ\text{C/W}$, que con contacto directo más mica y más pasta de silicona $R_{cd} = 0,4 \text{ } ^\circ\text{C/W}$, y que con contacto directo más mica $R_{cd} = 0,8 \text{ } ^\circ\text{C/W}$.

Por lo tanto podemos decir que cuando no sea necesario aislar el dispositivo, el tipo de contacto que más interesa es el directo más pasta de silicona, ya que da el menor valor de Rcd y si hubiese que aislar con mica interesa montar mica más pasta de silicona ya que la Rcd es menor que si se monta solo con mica. Por ello podemos obtener la siguiente conclusión: La mica aumenta la Rcd mientras que la pasta de silicona la disminuye y como se ha dicho cuanto más pequeña sea la Rcd menor superficie de aleta refrigeradora.

Resistencia del disipador (Rd).

Representa el paso por convección al aire del flujo calorífico a través del elemento disipador. Este dato será, en la práctica, la incógnita principal de nuestro problema, puesto que según el valor que nos de el cálculo, así será el tipo de aleta a emplear. Depende de muchos factores: potencia a disipar, condiciones de la superficie, posición de montaje y en el caso de disipadores planos factores como el grosor del material y el tipo de encapsulado.

Para el cálculo de la resistencia se pueden utilizar las siguientes fórmulas:

$$R_d = \frac{T_j - T_a}{P_d} - (R_{jc} + R_{cd})$$

$$R_d = R_{ja} - (R_{jc} + R_{cd})$$

$$R_d = \frac{T_d - T_a}{P_d}$$

Este valor de R_{ja} no es el que da el fabricante ya que éste la suministra sin disipador, y la que hay que utilizar es con disipador. El fabricante la facilita como suma de R_{jc} y R_{ca} puesto que ignora el tipo de disipador que utilizaremos.

Una vez calculada la R_d se pasa a elegir la aleta refrigeradora. Para la elección de la aleta, habrá que tener en cuenta que el tipo de encapsulado del dispositivo a refrigerar sea el adecuado para su montaje en la aleta disipadora que se haya elegido.

Después de cumplir la condición anterior hay que calcular la longitud o la superficie del disipador elegido. Para ello es necesario disponer de uno de los dos tipos de gráficas que ofrecen los fabricantes de disipadores, la R_d - longitud y la R_d - superficie.

Según la gráfica de que se disponga se obtendrá un valor de longitud o un valor de superficie de disipador que hay que montar para refrigerar adecuadamente el dispositivo semiconductor.

Resistencia Unión - Ambiente (R_{ja}).

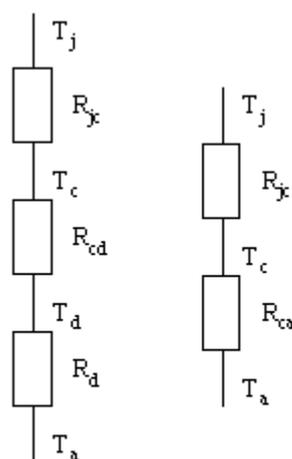


Fig. a

Fig. b

Como su nombre indica es la resistencia que existe entre la unión del semiconductor y el ambiente. Con esta resistencia deberemos de distinguir dos casos, el de resistencia unión ambiente con disipador y sin disipador. Cuando se habla de resistencia unión ambiente sin disipador, nos referimos a la resistencia unión contenedor junto con la contenedor ambiente:

$$R_{ja} = R_{jc} + R_{ca} \quad (\text{fig. b})$$

Este valor lo suministra el fabricante en función del tipo de cápsula.

Cuando se habla de la resistencia unión ambiente con disipador nos referimos a la suma de la resistencia unión contenedor (R_{jc}), la resistencia contenedor disipador (R_{cd}) y la resistencia disipador ambiente (R_d):

$$R_{ja} = R_{jc} + R_{cd} + R_d \quad (\text{fig.a})$$

Este valor no es conocido ya que varía según el tipo de disipador que se utilice. El valor de R_{ja} dependerá de los valores de R_d y de R_{cd} . Como no es un valor fijo, no existe una tabla de valores típicos.

Temperaturas.

Temperatura de la unión (T_j).

La temperatura máxima de la unión es el límite superior de temperatura a la que no se debe llegar y menos sobrepasar si queremos evitar la destrucción de la unión. Este dato es un valor que se suele suministrar, normalmente, en los manuales de los fabricantes de semiconductores.

Si este valor no se refleja en dichos manuales o, simplemente, no se encuentra, podremos adoptar unos valores típicos en función del dispositivo a refrigerar como los mostrados en la tabla que se expone a continuación:

DISPOSITIVO

-de unión de Germanio	Entre 100 y 125°C
-de unión de Silicio	Entre 150 y 200°C
-JFET	Entre 150 y 175°C
-MOSFET	Entre 175 y 200°C
-Tiristores	Entre 100 y 125°C
-Transistores de Unión	Entre 100 y 125°C
-Diodos de Silicio	Entre 150 y 200°C
-Diodos Zener	Entre 150 y 175°C

Se debe distinguir entre la temperatura máxima de la unión permitida para un dispositivo y la temperatura real de la unión a la que se pretende que trabaje dicho elemento o dispositivo que, lógicamente, siempre será menor que la máxima permitida. El objetivo del que diseña será mantener la temperatura de la unión por debajo de la máxima. Para ello se utiliza un coeficiente (K) de seguridad cuyo valor dará una temperatura de la unión comprendida entre el 50% y el 70% de la máxima. Por lo tanto k estará comprendido entre 0,5 y 0,7. Le asignamos el valor según el margen de seguridad que queremos que tenga el dispositivo. La temperatura de la unión que se utilizará en los cálculos será:

$$T_j = T_{j\text{máx}} \times k$$

Las condiciones de funcionamiento en función de k serán:

Para valores de $k=0,5$. Dispositivo poco caliente. Máximo margen de seguridad, pero el tamaño de la aleta refrigeradora será mayor.

Para valores de $k=0,6$. Dimensión menor de la aleta refrigeradora sin que el dispositivo se caliente demasiado.

Para valores de $k=0,7$. Máximo riesgo para el dispositivo, máxima economía en el tamaño de la aleta refrigeradora. Este coeficiente de seguridad exige que la aleta se sitúe en el exterior.

Temperatura de la Cápsula (Tc).

Este dato no se suministra en los manuales ya que depende del valor de la potencia que disipa el dispositivo, de la resistencia del disipador y de la temperatura ambiente. Por lo tanto solo podemos calcularla cuando conozcamos todos los datos reflejados en alguna de las siguientes expresiones:

$$T_c = P_d \times (R_{cd} + R_d) + T_a$$

$$T_c = T_j - (P_d \times R_{jc})$$

Temperatura del disipador (Td).

Este valor se obtiene a partir de la potencia disipada P_d , de la resistencia térmica de la aleta R_d y finalmente de la temperatura ambiente T_a . Se calculará con cualquiera de estas expresiones:

$$T_d = (P_d \times R_d) + T_a$$

$$T_d = T_c - (P_d \times R_{cd})$$

La temperatura obtenida será siempre inferior a la temperatura de la cápsula aunque será lo suficientemente alta en la mayoría de los casos como para no poder tocar el disipador con las manos.

Esto no es motivo de preocupación ya que se han tomado las medidas necesarias como para que la temperatura de la unión disponga de un margen de seguridad dentro de los márgenes ya explicados.

Puede suceder que la temperatura de la aleta es bastante elevada, tanto que si se toca con un dedo notaríamos que quema. Pero en todo momento la temperatura de la unión entrará con amplio margen dentro de los límites permitidos. No obstante, si se quiere disminuir esta temperatura, solo hay que calcular de nuevo la resistencia térmica R_d de la aleta, poniendo esta vez 0,5 como factor (k) necesario para determinar T_j . Ello llevará a adoptar una aleta más grande, pero tanto la T_c , como la T_d disminuirán como se desea.

Temperatura ambiente (T_a).

En la interpretación de este dato puede haber alguna confusión ya que se puede tomar su valor como la temperatura del medio ambiente cuando en realidad es la temperatura existente en el entorno donde está ubicado el disipador.

Potencia disipada.

La potencia máxima es un dato que nos dará el fabricante. Este dato es para las mejores condiciones de funcionamiento del dispositivo, es decir, para una temperatura de la cápsula de 25°C y un disipador adecuado. Por ejemplo, si de un determinado transistor nos dice el fabricante que puede disipar un máximo de 214 Watios, a primera vista se puede pensar que disipando 195 Watios no se corre ningún riesgo puesto que hay un margen con respecto al máximo y no se necesita disipador. Si conocemos la temperatura de la unión es de 175 °C y R_{ja} de 40 °C/W se tiene:

$$P_d = \frac{T_j - T_a}{R_{ja}} = \frac{175 - 25}{40} = 3,75w$$

Esta es la máxima potencia disipable sin disipador. Se puede ver que este valor se queda muy por debajo del indicado por el fabricante. Si consideramos una aleta con una buena resistencia térmica como puede ser una de 0,6 °C/W y unas resistencias térmicas cápsula - disipador R_{cd} y unión - cápsula R_{jc} de 0,7 °C/W y de 0,24 °C/W respectivamente, ambos valores también bastante adecuados, se tendrá:

$$P_d = \frac{T_j - T_c}{R_{jc} + R_{cd} + R_g} = \frac{175 - 25}{0,7 + 0,24 + 0,6} = 105,6w$$

Si hiciéramos disipar 195W como pretendíamos se destruiría la unión. Como se puede observar la potencia obtenida es superior a la disipable sin disipador e inferior a la que nos suministra el fabricante. Ello es debido a que el fabricante ha calculado la P_dmáx manteniendo la temperatura de la cápsula a 25 °C, cosa que en la práctica es imposible:

$$P_d = \frac{T_j - T_c}{R_{jc}} = \frac{175 - 25}{0,7} = 214w$$

Como se ha dicho este dato de 214 W es para las mejores condiciones de funcionamiento y el fabricante debe indicar en cuales se realizó esa medida.

Resumiendo, es importante saber interpretar adecuadamente los datos suministrados por el fabricante, de lo contrario pueden aparecer sorpresas desagradables.

Sabemos que la máxima potencia que se puede hacer disipar a un semiconductor sin disipador viene dada por el cociente entre el incremento de la temperatura y la resistencia térmica unión ambiente:

$$P_{dmax} = \frac{T_{jmax} - T_a}{R_{ja}}$$

Donde R_{ja} es la que nos suministra el fabricante que no incluye R_d. Cuando se utiliza un disipador, la resistencia térmica se divide en tres parámetros: la resistencia entre la unión y el contenedor (R_{jc}), entre la cápsula y el disipador (R_{cd}) y entre el disipador y el ambiente (R_d):

$$P_{dmax} = \frac{T_{jmax} - T_a}{R_{jc} + R_{cd} + R_d}$$

Después de las especificaciones de nuestro equipo y de las notas aclaratorias sobre la disipación, que es uno de los factores mas importantes de calculo de etapas de potencia, **vamos a pasar al calculo de nuestro circuito.**

La energía que los MOSFET tendrán que afirmar es uno de los factores mas principales a decidir.

La energía disipada en un MOSFET es el voltaje que pasa a través de el en un determinado tiempo, aunque el MOSFET esta cambiando gran cantidad de energía.

En el caso de que el MOSFET este en el tiempo (ton) la potencia disipada será:

$$Pd = I^2 \times R_{dson} = 15A \times 0,045\Omega = 16,8w$$

Rdson según datasheet.

Esta es la potencia disipada que tendrá que disipar sin disipador.

Otro problema para el calculo del disipador es cuando el MOSFET esta cambiando entre estados, por un periodo de tiempo corto, e MOSFET estará trabajando medio encendido y medio apagado, si ahora realizamos el calculo considerando los valores a la mitad la potencia disipada será:

$$Pd = \frac{V_{cc}}{2} \times \frac{I}{2} = \frac{65 \times 15}{4} = 243,7w$$

Podemos ver que es muy superior, sin embargo el MOSFET estará disipando solamente este valor para un periodo de tiempo muy corto que el MOSFET estará cambiando entre estados.

Por lo que podemos ver que potencia disipación depende de los tiempos de conmutación del MOSFET.

Existe otro calculo par la disipación media de los MOSFET y para ella es empleada esta ecuación:

$$480 \times \frac{\text{time_to_switch_over}}{\text{time_getween_switchings}}$$

6.1. PASOS A SEGUIR PARA EL CALCULO DEL DISIPADOR.

La mayoría de fabricantes de semiconductores proporcionan los datos suficientes para poder calcular el disipador que necesitamos. Necesitamos como punto de partida, la temperatura máxima que puede alcanzar la unión del transistor. Esta temperatura no se deberá alcanzar en ningún caso, para no destruir el componente. Normalmente el fabricante proporciona el "operating temperature range" por ejemplo, para nuestro caso según datasheet -55 to 175 °C indica que la temperatura máxima es de 175°C. Nosotros podemos tomar unos coeficientes de seguridad k ya comentados como sigue:

$k = 0.5$ para un diseño normal con temperatura moderada.

$k = 0.6$ para economizar en tamaño de disipador.

$k = 0.7$ cuando el disipador permanezca en posición vertical y en el exterior (mejora de convección).

Con el coeficiente k , y tomando la temperatura máxima de funcionamiento como T_j , tenemos la expresión:

$$T = k T_j - T_a = P_d (R_{jc} + R_{cd} + R_{da})$$

Si no disponemos de estos datos, podemos tomar como $T_j = 200\text{ °C}$ para transistores de silicio, y $T_j = 125\text{ °C}$ para transistores de germanio.

El flujo de calor, desde la unión PN hasta el ambiente tiene que atravesar varios medios, cada uno con diferente resistencia térmica.

- Resistencia unión-cápsula (R_{jc}). Viene dado en manuales y tablas, y depende de la construcción de la cápsula. El tipo TO-247AC.
- Resistencia cápsula-disipador (R_{cd}). Depende del encapsulado y del aislamiento, si lo hay, entre el componente y el disipador. El aislante puede ser mica, pasta de silicona y otros medios. Cada uno presenta diferente resistencia térmica.
- Resistencia disipador-ambiente (R_{da}). Este es el que tratamos de calcular.

Para condiciones de funcionamiento normal la Potencia disipada será:

$$P_d = \frac{T_j - T_a}{R_{jc} + R_{cd} + R_{da}}$$

En la mayoría de los casos lo desconocido será R_{da} que despejando de la formula anterior será:

$$R_{da} = \frac{T_j - T_a}{P_d} - (R_{jc} + R_{cd})$$

Para nuestro caso y aplicando los datos del datasheet facilitados por el fabricante:

$T_j = 175\text{ °C}$.

$R_{jc} = 0.7\text{ °C/w}$.

De nuestro montaje y tablas, deducimos:

$R_{cd} = 0.5\text{ °C/w}$ (para TO.247AC y contacto directo y silicona).

TABLA DE RESISTENCIAS TERMICAS DE DISIPADORES

RESISTENCIA TERMICA CAPSULA-DISIPADOR				
Tipo de contenedor	Contacto directo sin mica	Contacto directo y silicona	Contacto con mica	Contacto con mica silicona
TO.5	1	0,7	--	--
TO.39	1	0,7	2	1,5
TO.126	1,4	1	1,4	1,3
TO.220	0,8	0,5	1,4	1,2
TO.247AC	0,8	0,5	1,4	1,2
TO.152	0,8	0,5	1,2	0,9
TO.90	0,5	0,3	1	0,7
TO.3P.	0,4	0,2	2,1	1,5
TO.59	1,2	0,7	--	--
TO.117	2	1,7	--	--
SOT.48	1,8	1,5	--	--
DIA.4L	1,1	0,7	--	--
TO.66	1,1	0,65	--	--

De nuestros cálculos en conmutación:

$P_d = 16,8w$.

$T_{amx} = 70^{\circ}C$ (especificada en la especificación de nuestro proyecto).

El valor de la R_{da} será:

Nota: utilizaremos $K = 0.6$ para economizar el tamaño del disipador.

$$R_{da} = \frac{K(T_j - T_a)}{P_d} - (R_{jc} + R_{cd}) = \frac{0.6(175 - 70)}{16,8} - (0,7 + 0,5) = 2,55^{\circ}C / W$$

Buscamos en catalogo y encontramos el disipador siguiente:



Radiador para TO-247AC

Tiene una $R = 5 \text{ } ^\circ\text{C}/\text{w}$, es suficiente, teniendo en cuenta que ya hemos tomado un coeficiente k de seguridad igual a 0.6 y nos aseguramos de sobra. Con este radiador,

podemos calcular la temperatura que alcanzará el mismo cuando el MOSFET disipa 16,8w de una forma muy sencilla:

$$T_d - T_a = R_{da} \cdot P_d \text{ ---> } T_d = R_{da} \cdot P_d + T_a = 5 \cdot 16,8 + 70 = 154 \text{ } ^\circ\text{C}$$

La elección del coeficiente k es arbitraria por nuestra parte. Podemos perfectamente elegir $k=1$ pero nos arriesgamos mucho. Es preferible en este caso subir la temperatura ambiente de diseño a más grados, para evitar que se destruya. Tener en cuenta que si el dispositivo está en una caja, la temperatura fácilmente sube a 40 grados y más, por eso nuestros cálculos están realizados con la temperatura máxima.

Ahora para asegurarnos más vamos a calcular la temperatura de unión (T_j) que alcanzaría para una R_d de $2,55^\circ\text{C}/\text{w}$.

$$T_j = (R_{jc} + R_{cd} + R_{da})P_d + T_a = (0,7 + 0,5 + 2,55)16,8 + 70 = 133^\circ\text{C}$$

Esta sería la máxima que vamos a tener en funcionamiento, luego no llegamos al dado por el fabricante.

Como se ha podido observar los pasos para el cálculo del disipador, han sido para la máxima potencia para la que ha sido diseñado este trabajo.

Si para desarrollar un montaje con un determinado motor el cual no va a ser modificada su potencia por el cambio de otro, estos cálculos pueden ajustarse a un determinado disipador para el ahorro bien de este o de espacio.

Por ejemplo utilizando el mismo Mosfet y un motor con las características siguientes:

- Potencia nominal de 400W. (P).
- Máx corriente en continua a 5000 rpm. (I_o).
- Tensión nominal. (V_i).

Y utilizando la fórmula siguiente:

$$P_d = I_o^2 \times (R_{dson} \times K) \times D + 0,5 \times V_i \times I_o \times (t_r + t_f) f_s$$

Pd: Potencia disipada por el transistor MOSFET.

Io: Corriente de salida nominal.

Vi: Tensión de entrada de alimentación.

Rds(on): Resistencia de drenaje-fuente en estado de conducción del MOSFET.

tr + tf: tiempo de conmutación total del MOSFET (tiempo de subida y tiempo de bajada).
Asumiendo que el circuito de excitación es adecuado para efectuar tiempos de conmutación de: $tr+tf = 100ns$.

K: factor de ajuste de Rds del MOSFET por efectos de la temperatura.

D: valor estimativo de Duty-Cycle, Se adoptamos el valor 1 suponiendo el peor de los casos.

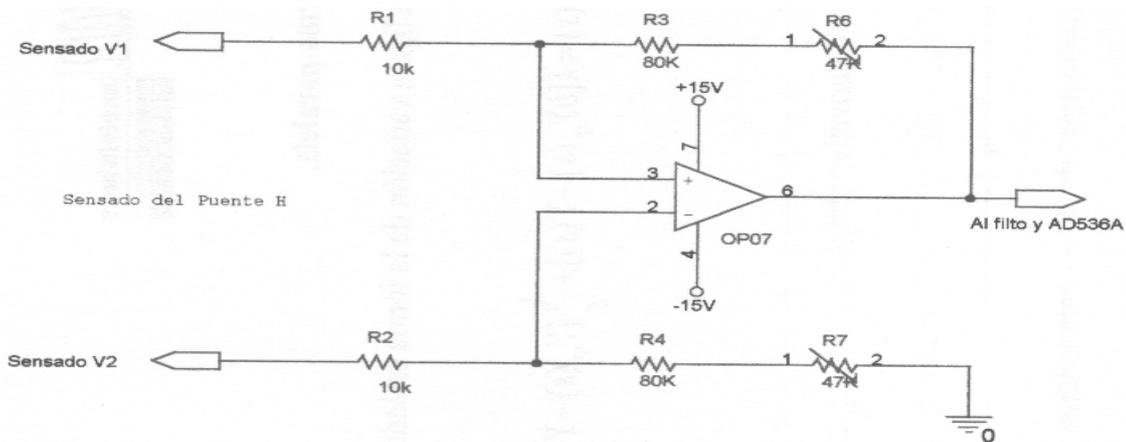
fs: es la frecuencia de PWM.

Para mantener un margen de seguridad se sumara a la I nominal un 20%.

Con este calculo se obtendrá una Pd, que será para este determinado motor y siguiendo los mismos pasos anteriores se obtendrá un nuevo valor de Rda.

El valor de este nuevo disipador solo será efectivo para potencia elegida de este motor.

6.2. Calculo del AO diferencial.



V. alimentación ± 15 Voltios

Si $R1 \neq R3$ y $R2 \neq R4$

$$S = V1 \left[\frac{R4}{R3 + R4} + \frac{R2 \times R4}{(R2 + R4)R1} \right] - V2 \times \frac{R2}{R1} \quad (\text{Fórmula A})$$

Pero haciendo:

$R_1 = R_3$ y $R_2 = R_4$ nos quedaría:

$$S = \frac{R_2}{R_1} (V_1 - V_2) \quad (\text{Fórmula B})$$

Siendo:

$\frac{R_2}{R_1}$ → ganancia del A.O (G).

$(V_1 - V_2)$ → caída de tensión en la resistencia de sensado.

S → salida del A.O.

La caída de tensión en la resistencia de sensado puede oscilar entre las siguientes tensiones:

$$V_{\min} = (V_1 - V_2)_{\min} = R_1 \times I_{\min} = 0.1\Omega \times 2A = 0.2 \text{ Voltios}$$

$$V_{\max} = (V_1 - V_2)_{\max} = R_1 \times I_{\max} = 0.1\Omega \times 15A = 1.5 \text{ Voltios}$$

Haciendo $R_2 = 100K$ y $R_1 = 10K$ daremos al A.O una ganancia de 10, según la fórmula B, obteniendo así una tensión en la salida "S" del A.O 10 veces mayor que el diferencial de tensión existente en la resistencia de sensado, por lo tanto, tendremos unas tensiones de salida de:

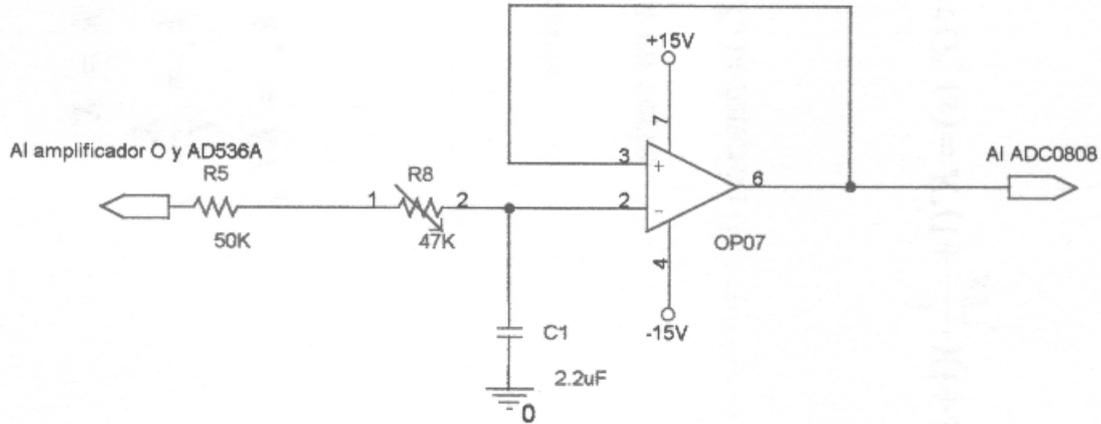
$$S_{\text{outmin}} = (V_1 - V_2)_{\min} \times G = 0.2 \times 10 = 2 \text{ Voltios}$$

$$S_{\text{outmax}} = (V_1 - V_2)_{\max} \times G = 1.5 \times 10 = 15 \text{ Voltios}$$

*Nota:

En la posición R_2 colocaremos una resistencia de 80K más una resistencia ajustable en serie de 47K, en lugar de una resistencia de 100K, con el fin de poder ajustar la ganancia del amplificador a 10.

6.2. Calculo del Filtro Paso Bajo.



$$\frac{V_o}{V_{in}} = \frac{CS}{R + (1/CS)} = \frac{1}{1 + RCS}$$

$$f_c = \frac{1}{2\pi RC}$$

* $f_c = 1\text{Hz}$ (Frecuencia supuesta para eliminar la componente alterna)

* $C = 2.2 \mu\text{F}$ Tántalo sin polaridad (supuesto el condensador para calcular resistencia)

$$R = \frac{1}{2\pi f_c C} = \frac{1}{2\pi \times 1\text{Hz} \times 2.2 \times 10^{-6} \text{F}}$$

Para realizar un perfecto ajuste de la frecuencia de corte, dividiremos el valor de la resistencia del cálculo en una resistencia de 50 K y un potenciómetro de 47 K, con este último regularemos la frecuencia deseada.

7. Construcción de la tarjeta de control.

Consideraciones de operación.

El sistema de control de la tarjeta fue diseñado para controlar 1 motor, con su en codificador.

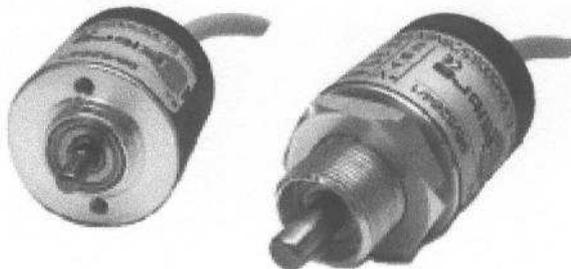
Las variables a controlar son la velocidad, aceleración y Posición, por medio de los dispositivos MPID y el puente H.

El control en la variable de la velocidad, consiste en mantener fija una velocidad sin que importe si se tiene o no una presencia de carga en el motor, hasta que este sea frenado. El control de la aceleración con la que el motor logra su movimiento, y permite llegar a determinada velocidad se puede variar a voluntad, siempre y cuando no sea mayor que la velocidad deseada.

En el control de posición el motor parte de un estado de reposo en cierta posición, llegando a una determinada posición deseada, y terminando en reposo.

El sistema censa la velocidad a través de un en codificador incremental, formado por un disco ranurado colocado en el eje del motor, donde un dispositivo opto electrónico (emisor-receptor) permite cuadrar la señal cuando el disco gira, mandando cuatro diferentes estados, que permiten saber en que dirección gira el motor.

En la siguiente figura se pueden observar algún dispositivo de los llamados Encoder incremental.



Elementos del sistema.

La tarjeta cuenta con los siguientes elementos:

- Microprocesador (8051).
- LM629.
- 74LS00.
- H11L1.
- CD40106C.
- 2-IR2110.
- IRFP250N.
- D1N4118.
- BTA49.
- LM324.
- LM311.
- MGDI-18 : 18W POWER y MGDI-19 : 20W POWER.
- 2- OP07.
- AD356A.
- ADC0808.
- TO220 (178-805). TERMICO.
- LM7446A.
- MM74HC11.
- MM74HC08.
- DS26LS31 y 32.
- TL7705A.
- STDGC 16Mhz.
- 74LS04.
- SN74HC4040.
- Conectores de 14 y 10 Pines.
- Componentes pasivos, RESISTENCIAS ,CONDENSADORES Y DIODOS.

7.1. Diseño de la tarjeta.

El diagrama del circuito se realizó con el paquete de ORCAD versión 9.0, como se observa en el apartado de esquemas, se contaban ya con algunas librerías y otras se han confeccionado. El diseño de las caras o lados de componentes y soldaduras, así como el “silkscreen” (colocación de componentes) se confecciono con el mismo programa ORCAD en el apartado de LAYOT, este también se pueden ver en apartado de esquemas.

7.2. Diseño del circuito de control.

El control de velocidad consiste en mantener fija una velocidad, aunque se tenga la ausencia o presencia de carga en el motor. Inicialmente el motor está en reposo, se debe acelerar hasta llegar a la velocidad deseada, y mantenerla sin importar las variaciones en la carga, hasta que llegue un frenado. La aceleración con la que el motor alcanza su movimiento, se puede variar a voluntad, siempre y cuando no sea mayor que la velocidad deseada. El control de velocidad es en ambos sentidos de giro.

El control de posición, parte de un estado de reposo en cierta posición, y va a otra posición determinada, terminando en reposo. Para lo anterior se presenta una aceleración, una velocidad constante por cierto tiempo, y una desaceleración para llegar exactamente en reposo a la posición deseada. La aceleración-velocidad- desaceleración deseadas, deben permanecer invariables ante las variaciones de la carga. El control de posición es para ambos sentidos de giro.

El sistema sensa la velocidad a través de un codificador incremental (encoder), que está formado por un disco ranurado puesto sobre el eje del motor, y dos sensores opto electrónicos (emisor y receptor), de un circuito electrónico que permite cuadrar la señal, la señal cuadrada que manda cada sensor al girar el disco está desfasada $\pi/2$, dando 4 diferentes estados que permiten saber si el motor gira en un sentido u otro.

El **MPID** es un dispositivo procesador de control de movimientos dedicados, y diseñado para usarlo con una variedad de servomotores a escobillas de C.D., y otros servomecanismos, los cuales proveen una señal cuadrada de retroalimentación de posición incremental.

El **MPID**, se encarga de comparar el valor medido con el valor deseado, generando la señal de error que pasa por el filtro PID, en función del error que genera una señal **PWM**.

El **punto H**, recibe la señal **PWM** a través de los opto acopladores y del driver y actúa sobre el motor, proporcionando la polarización, voltaje y corrientes requeridos para lograr el control deseado sobre el motor.

8. Software del sistema.

Una parte importante es contar con un software que permita la comunicación del microcontrolador (μC) con el dispositivo programable (MPID). A continuación se describen las necesidades y los condicionantes de este software.

8.1. Conexión del LM629.

En la Figura 2.7, se muestra la conexión del LM629 con el Bus y el Encoder, el Bus requieren 8 líneas de datos, lectura y escritura, solicitud de interrupción, descodificación de direcciones y con el Encoder señales de cuadratura A y B e índice IN.

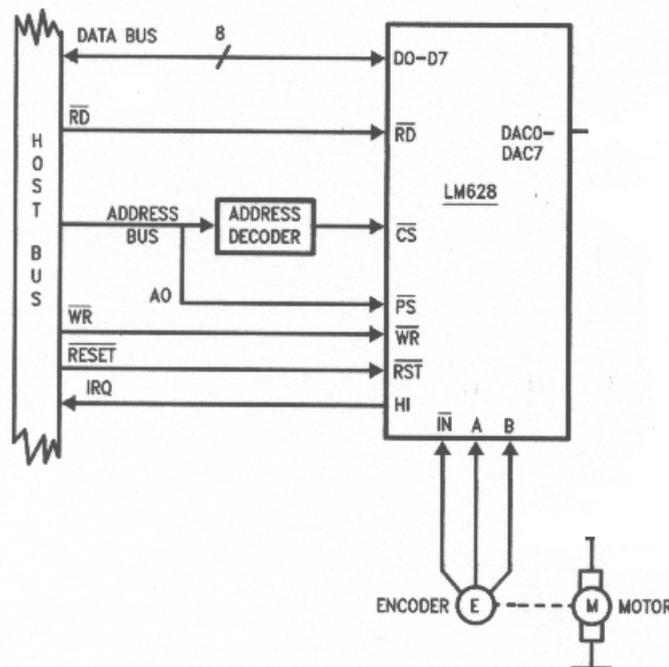


Figura 2.7 Conexión del LM629 con el μ C.

8.2. Módulos de programación para control de la tarjeta.

Módulo de chequeo del Bit-busy.

Es el primer módulo requerido. Comprueba el *Bit-busy*, que forma parte de la palabra de estado del MPID e indica cuando es posible la transferencia de datos. La ejecución se repite hasta que el bit indica que el MPID no está ocupado. Se ejecuta cuando el procesador escribe un byte comando, o lee o escribe el segundo byte de datos, ver tabla 3.1. Cuando el *Bit-busy* está "ocupado" el MPID ignora cualquier comandos o intento de transferencia de datos.

El módulo verifica el *Byte de estado*, y espera hasta que el *Bit-busy* esté cero, lo que garantiza la comunicación. Distribución en tabla 3.2.

En la figura 2.8, se observa el diagrama de flujo que representa el módulo de chequeo del bit del *Bit-busy*, Este módulo se empleará a lo largo de toda la programación.

La lectura del *Byte de estado* se ejecuta enviando al MPID un comando *RDSTAT*, que es directamente soportado por el hardware del MPID llevando a nivel lógico bajo $\overline{/CS}$, $\overline{/PS}$ y $\overline{/RD}$.

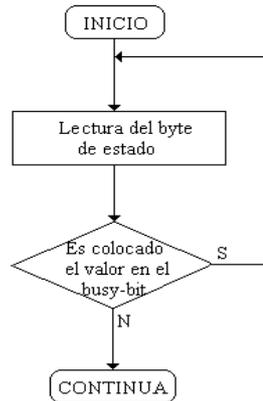


Figura 2.8 Módulo de prueba del bit-busy.

Módulo de iniciación.

Un módulo de iniciación contiene: un comando de puesta a cero, otro de iniciación, un control de interrupción, y comandos de transferencia de datos. Puede llevar el bloque de inicio para hardware y un comando PORT12. Al entrar la alimentación se lleva a cabo la puesta a cero hardware que consiste en llevar la conexión /RST a un nivel lógico bajo. El reset necesita al menos 1.5 ms de aplicación. El MPID ignora cualquier comando o transferencia de datos durante la ejecución del reset.

En la figura 3.2, se muestra un Bloque de inicio de hardware, que incluye una prueba funcional del MPID, esta prueba puede ser completada inmediatamente después del inicio de hardware.

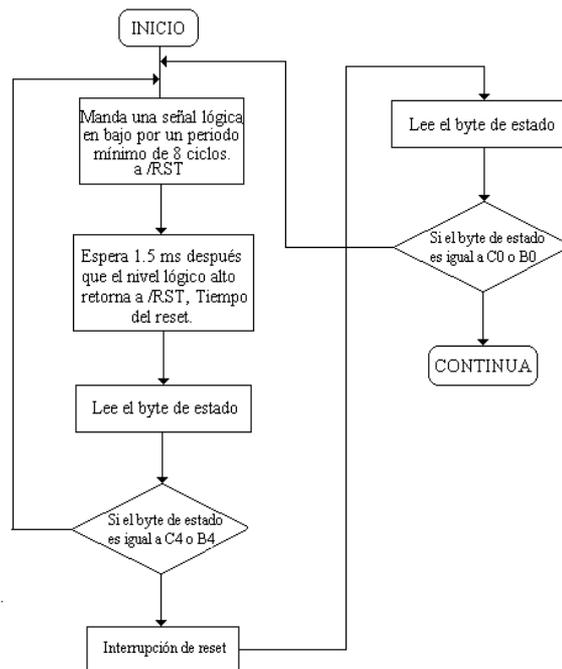


Figura 3.2 Diagrama de flujo del Reset.

Iniciación de interrupciones.

El comando *RSTI*, permite al usuario poner a cero los bits de la interrupción en la palabra del byte de estado, como se puede ver en la tabla 3-1.

El comando *RSTI* también pone a cero el terminal de salida de interrupciones /HI. Cualquier combinación de bits será puesta a cero en una sola secuencia del comando.

El reset se puede enviar también por software ejecutando los mismos trabajos y atendiendo a los mismos requerimientos de ejecución de 1.5 ms máximo.

8.3. Módulo de programación del filtro.

Parámetros de carga del filtro (Coeficientes).

Una secuencia de carga de filtro incluye: un comando LFIL, una palabra de control y un número variable de palabras de datos.

El comando LFIL carga los coeficientes del filtro sobre *buffers* primarios antes de enviarlos al filtro para su actualización.

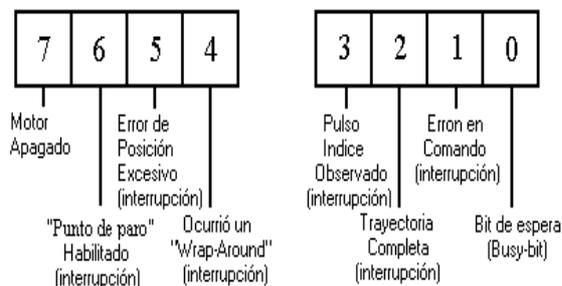


Tabla 3-1 Distribución de bits del byte de estado.

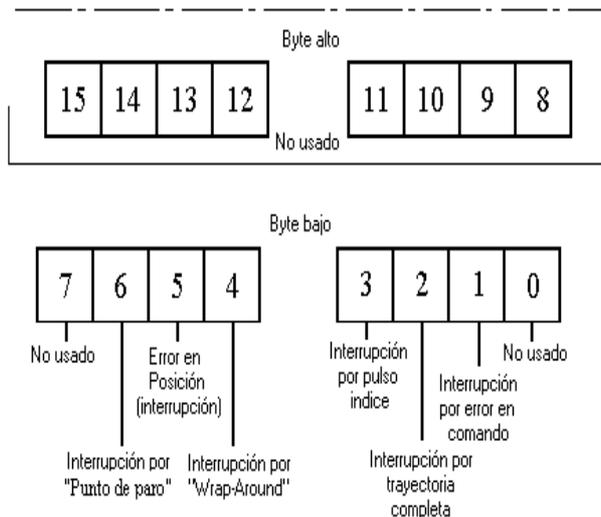


Tabla 3-2 Distribución de bits para interrupción del reset.

Los dos bytes de datos se escriben tras el comando LFIL, en coherencia con la palabra de control. El primer byte de la palabra de control da el coeficiente de la muestra derivativa d_s (selección del intervalo de la muestra derivativa), y el segundo byte indica, con lógica de *unos* en las respectivas posiciones cuales son los coeficientes del filtro que se van a enviar, ver los parámetros en la tablas 3-3 y 3-4, para cualquier combinación.

Detrás de la palabra de control se escriben los coeficientes del filtro. Cada coeficiente forma un par de bytes de datos en una palabra, así cualquier combinación de los cuatro coeficientes puede ser cargada dentro de una sola secuencia del comando LFIL.

El número de palabras de datos de coeficientes puede oscilar entre cero y cuatro.

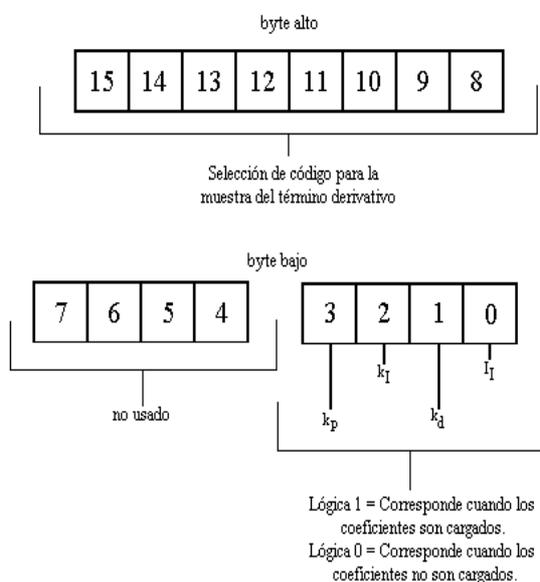


Tabla 3-3 Palabra de control del filtro.

Posición de bits en la palabra de control								d_s	Intervalo T_d
15	14	13	12	11	10	9	8		
0	0	0	0	0	0	0	0	1	T_s
0	0	0	0	0	0	0	1	2	$2T_s$
0	0	0	0	0	0	1	0	3	$3T_s$
0	0	0	0	0	0	1	1	4	$4T_s$
				•				•	•
				•				•	•
				•				•	•
1	1	1	1	1	1	1	1	256	$256T_s$

TABLA 3-4. Selección de código del término derivativo en un intervalo de una muestra.

Donde:

$$T_s = (2048) \times \left(\frac{1}{f_{clk}} \right) \text{ período de la muestra del sistema} \quad (1)$$

$$T_d = ds \times T_s \longrightarrow \text{Intervalo de la muestra del término derivativo} \quad (2)$$

Módulo de trayectoria (programación).

Carga de parámetros de la trayectoria.

Una secuencia de carga de trayectoria incluye: el comando LTRJ, una palabra de control de trayectoria y una variable con un número de palabras de datos.

El comando LTRJ carga los parámetros sobre buffers primarios. Los dos bytes de datos se escriben inmediatamente después del LTRJ e incluyen: la palabra de control de trayectoria, el primer byte del programa con lógica uno en las respectivas posiciones del bit, el modo de la trayectoria (velocidad o posición), el modo de la dirección de la velocidad, y el modo de paro (ver módulo de paro). El segundo byte indica, con lógica de unos, las respectivas posiciones de cada bit. También indica si los parámetros son absolutos o relativos, vea la tabla 3-5, cualquier combinación de los tres parámetros puede ser cargada en una sola secuencia del comando LTRJ.

Detrás de la palabra de control de la trayectoria va el parámetro de la trayectoria. Cada parámetro ocupa un par de palabras de datos (para bytes de datos). Para cualquier combinación, los tres parámetros se pueden cargar en una sola secuencia del comando LTRJ.

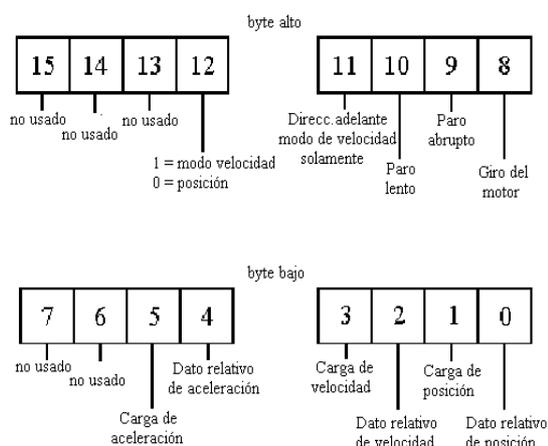


Tabla 3-5 Distribución de bits de la trayectoria.

Módulo de paro.

Representa la forma de pasar del movimiento al paro cuando se trabaja en modo velocidad, porque cuando el MPID opera en el modo de posición normalmente el paro es siempre uniforme y ocurre automáticamente al final de una trayectoria especificada (aquí el módulo de paro no es necesario). Pero en situaciones especiales puede ser preciso, como en un paro prematuro, o anticipado.

Cuando el MPID opera en modo velocidad, el paro se realiza siempre por medio de un módulo de paro.

Carga de parámetros de trayectoria.

Ver la tabla 3-5. El primer byte de la palabra de control de la trayectoria selecciona el modo de paro del motor, el bit 9 de la palabra de control de la trayectoria indica paro abrupto: este modo cambia el movimiento (a máxima desaceleración) para pasar la posición a una posición inicial.

Si se desea un modo de paro uniforme, se selecciona el bit 10 de la palabra de control de la trayectoria: este modo cambia el movimiento por medio de una desaceleración uniforme en coherencia con el rango de la aceleración.

Tras la detención por un módulo de paro el motor maneja una señal remanente de cero y el sistema de control puede no afectar la posición del eje, que puede estacionado en giro libre. Esto indica una amplificación del offset: el eje puede rotar lentamente pero con una capacidad de torque mínimo, a menos que se carguen parámetros para una nueva trayectoria.

Programa de fase de lazo.

Al arrancar es preciso determinar la polaridad de la señal de control del motor. Si es incorrecta (o sea que hay una inversión en el lazo), se puede presionar al eje a una posición indeseada, resultando que la velocidad del motor se descontrola, condición de giro uniforme a alta velocidad.

La inversión del lazo para el MPID (LM628), puede ser corregida de tres formas: intercambiando la posición del eje por medio de intercambiar la señal del encoder (canal A y canal B), intercambiando el control de la potencia del motor, o invirtiendo la señal del comando del motor después de la aplicación del manejo de amplificación del motor.

La inversión del lazo para el MPID (LM629) puede ser corregida de tres formas: intercambiando la posición del eje por medio de intercambiar la señal del encoder (canal A y canal B), intercambiando el control de la potencia del motor, o invirtiendo la señal del PWM SIGN.

9. Tipos de movimientos.

Movimiento de posición absoluta.

El programa de movimiento de *posición absoluta* utiliza: Los programas de los módulos de inicialización, el del filtro, la secuencia del comando LTRJ y el comando STT.

Los factores que influyen en el desarrollo de este programa incluyen: El programa puede enviar el cálculo de parámetros de trayectoria, el programa puede enviar el flujo requerido para cargar y ejecutar un movimiento de posición absoluta. La correcta realización del movimiento puede verificarse a través de una simple observación visual.

Para los cálculos de trayectoria es necesario saber que es un decodificador de líneas y un decodificador de cuentas, que es lo que se expone a continuación:

El encodificador incremental de cuadratura EIC (quadrature incremental encoder).

Como un suplemento al cálculo de parámetros de trayectoria, se da una información respecto de la diferencia entre *decodificación de líneas* y *decodificación de cuentas*.

Un encodificador incremental de cuadratura, codifica el eje de rotación en forma de pulsos eléctricos, la figura 2.9, detalla las señales generadas por un encodificador incremental de cuadratura (quadrature incremental encoder) de 3 canales M1, M2 e IN. El contador del MPID, decodifica la señal incremental de cuadratura para determinar la posición absoluta del eje.

La resolución de un encodificador integral de cuadratura, es usualmente especificado como el número de *líneas*, este número indica el número de ciclos de la señal de salida, para cada revolución completa del eje. Por ejemplo, un encodificador de *N líneas*, genera N ciclos de señales de salida, durante cada revolución completa del eje.

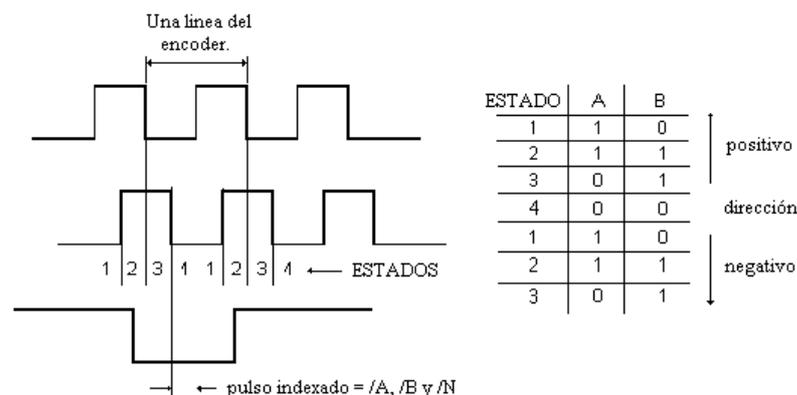


Figura 2.9 Señales cuadradas de salida del encoder, y tabla de decodificación de dirección.

Por definición, dos señales de dos canales A y B respectivamente, que están en cuadratura a 90°, o sea desfasadas 90° entre sí, si las señales se colocan juntas una de otra como se ve en la fig. 2.9, se atraviesa por cuatro estados digitales distintos durante cada ciclo completo de cada canal, cada transición de estado representa una *cuenta* de movimiento del eje, y el canal principal indica la dirección de rotación del eje.

Cada línea, sin embargo, representa un ciclo de las señales de salida, y cada ciclo representa *cuatro cuentas*, que se representan en la siguiente ecuación:

$$\text{Ecuación. } \left(N \frac{\text{CICLOS}}{\text{REVOLUCION}} \right) \times \left(4 \frac{\text{CUENTAS}}{\text{CICLO}} \right) = 4N \frac{\text{CUENTAS}}{\text{REVOLUCION}} \quad (3)$$

Si el sistema de referencia del encodificador utiliza 1000 *líneas*, entonces:

$$\text{Ecuación } \left(1000 \frac{\text{CICLOS}}{\text{REVOLUCION}} \right) \times \left(4 \frac{\text{CUENTAS}}{\text{CICLO}} \right) = 4000 \frac{\text{CUENTAS}}{\text{REVOLUCION}} \quad (4)$$

Período de la muestra.

El muestreo que sucede de la posición actual del eje a una frecuencia fija, es el recíproco del período de la muestra del sistema. Y el período de la muestra del sistema es la unidad de tiempo, en el cual la aceleración y la velocidad del eje se basan en la siguiente ecuación:

$$\text{Ecuación } T_s = (2048) \times \left(\frac{1}{T_{\text{CLOCK}}} \right) \text{ Período de la muestra del sistema.} \quad (5)$$

Si el sistema de referencia usa un reloj de 8 MHz, entonces el período de la muestra del sistema de referencia que se sigue directamente desde la definición será.

$$\text{Ecuación. } T_s = (2048) \times \left(\frac{1}{8 \times 10^6 \text{ Hz}} \right) = 256 \times 10^{-6} \frac{\text{SEGUNDOS}}{\text{MUESTRA}} \quad (6)$$

Cálculo de parámetros de la trayectoria.

Ejemplo: Sea un eje acelerado a 0.1 rev/seg^2 , hasta alcanzar una velocidad máxima de 0.2 rev/seg , entonces, se desacelera hasta parar exactamente a dos revoluciones de la posición inicial. Los cálculos de parámetros de la trayectoria para este movimiento son detallados a continuación.

NOTA: Una muestra = 1 ciclo = tiempo transcurrido de un pulso a otro como se ve en la figura 2.10.

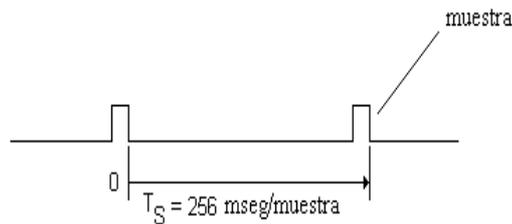


Figura 2.10, Una muestra = 1 ciclo = tiempo transcurrido de un pulso a otro.

Cálculo de velocidad, aceleración y posición.

Aceleración.

$$A = \left(4000 \frac{\text{CUENTAS}}{\text{REVOLUCION}} \right) \times \left(256 \times 10^{-6} \frac{\text{SEGUNDOS}}{\text{MUESTRA}} \right)^2 \times \left(0.1 \frac{\text{REVOLUCIONES}}{\text{SEGUNDO}^2} \right) =$$

$$A = 2.62 \times 10^{-5} \frac{\text{CUENTAS}}{\text{MUESTRA}^2}$$

$$A = \left(2.62 \times 10^{-5} \frac{\text{CUENTAS}}{\text{MUESTRA}^2} \right) \times (65536) = 1.718 \frac{\text{CUENTAS}}{\text{MUESTRA}^2}$$

$$A = 2 \frac{\text{CUENTAS}}{\text{MUESTRA}^2} \quad \text{Aceleración Redondeada}$$

$$A = 00\ 00\ 00\ 02 \text{ hex} \quad \frac{\text{CUENTAS}}{\text{MUESTRA}^2}$$

Posición.

$$P = \left(4000 \frac{CUENTAS}{REVOLUCION} \right) \times (2.0 REVOLUCIONES) = 8000 CUENTAS$$
$$P = 00\ 00\ 1F\ 40 \text{ hex CUENTAS}$$

Comentarios.

Después de completar el movimiento, el sistema de control intentará mantener constante la flecha en una posición absoluta con un consumo de corriente, si la flecha se toca ligeramente imprimiéndole una carga, forzándola a que vaya a la posición de reposo, y si se suelta la flecha, se moverá a la inversa de la fuerza a la posición deseada.

Movimiento de posición relativa.

Se explica la secuencia de la programación requerida para cargar y ejecutar un movimiento de posición relativa.

Un movimiento de posición relativo, es un movimiento que finaliza el número especificado (relativo) de cuentas, lejos de la presente posición del destino de la flecha, por ejemplo, si la presente posición destino de la flecha es de 10 cuentas, y se especifica una posición relativa de 30,000 cuentas como complemento del movimiento de la posición absoluta de la flecha, entonces el total será 30,010 cuentas (30,000 cuentas son relativas a 10 cuentas).

Carga de trayectoria de parámetros.

El primer byte de la palabra de control de la trayectoria, programa la posición del modo de operación, el segundo byte indica los tres parámetros de trayectoria que pueden ser cargados, también indica que la aceleración y la velocidad pueden ser valores absolutos, porque la posición tendrá un valor relativo.

Cálculo de parámetros de la trayectoria.

Con independencia de la posición actual del eje, este puede completar 30 revoluciones en dirección invertida o contraria, el tiempo total para completar el movimiento es de 15 segundos, y el tiempo total para la aceleración y desaceleración es de 5 segundos. Si el sistema de referencia utiliza un encoder de 1000 líneas, el número de *cuentas* para cada revolución completa del eje, y el número de *cuentas* totales para este movimiento de posición, se determinan como sigue:

$$\left(1000 \frac{\text{CICLOS}}{\text{REVOLUCIÓN}}\right) \times \left(4 \frac{\text{CUENTAS}}{\text{CICLO}}\right) = 4000 \frac{\text{CUENTAS}}{\text{REVOLUCIÓN}}$$

$$\left(4000 \frac{\text{CUENTAS}}{\text{REVOLUCIÓN}}\right) \times (30 \text{ REVOLUCIONES}) = 120,000 \text{ CUENTAS}$$

Con respecto al tiempo, dos terceras partes del movimiento son hechas a máxima velocidad, y una tercera parte es hecha a la mitad de la máxima velocidad. Por lo tanto el total de *cuentas* del movimiento durante el período de aceleración y desaceleración, es de una quinta parte del total de *cuentas* del movimiento.

Total de Cuentas del movimiento durante la aceleración y la desaceleración:

$$\frac{120,000 \text{ CUENTAS}}{5} = 24,000 \text{ CUENTAS}$$

Cuentas del movimiento durante la aceleración:

$$\frac{24,000 \text{ CUENTAS}}{2} = 12,000 \text{ CUENTAS}$$

Si el sistema de referencia usa un reloj de 8 MHz. El período de la muestra del sistema de referencia Se determina así:

$$T_s = (2048) \times \left(\frac{1}{8 \times 10^6 \text{ Hz}}\right) = 256 \times 10^{-6} \frac{\text{SEGUNDOS}}{\text{MUESTRA}}$$

El número de muestras durante la aceleración y desaceleración se obtiene así:

$$\frac{2.5 \text{ SEGUNDOS}}{256 \times 10^{-6} \frac{\text{SEGUNDOS}}{\text{MUESTRA}}} = 9768 \text{ MUESTRAS}$$

Usando el número de *cuentas* del movimiento durante la aceleración y el número de muestras durante la aceleración y desaceleración tenemos que:

$$s = \frac{at^2}{2} \quad \text{Distancia del movimiento durante el tiempo } t, \text{ a una aceleración } a.$$

Donde:

$$a = \frac{2s}{t^2} = \frac{2 \times 12,000 \text{ CUENTAS}}{(9768 \text{ MUESTRAS})^2} = 0.000252 \frac{\text{CUENTAS}}{\text{MUESTRA}^2}$$

El número de muestras durante la aceleración y desaceleración se obtiene así:

$$\frac{10 \text{ SEGUNDOS}}{256 \times 10^{-6} \frac{\text{SEGUNDOS}}{\text{MUESTRA}}} = 39062 \text{ MUESTRAS}$$

Usando el total de *cuentas* del movimiento para la máxima velocidad, y el número de muestras también para la máxima velocidad, se determina la velocidad como sigue:

$$\frac{96,000 \text{ CUENTAS}}{39,062 \text{ MUESTRA}} = 2,458 \frac{\text{CUENTAS}}{\text{MUESTRA}}$$

La aceleración y la velocidad son escalables:

$$\left(0.000252 \frac{\text{CUENTAS}}{\text{MUESTRA}^2} \right) \times (65,536) = 16,515 \frac{\text{CUENTAS}}{\text{MUESTRA}^2}$$

$$\left(2,458 \frac{\text{CUENTAS}}{\text{MUESTRA}} \right) \times (65,538) = 161,087,400 \frac{\text{CUENTAS}}{\text{MUESTRA}}$$

La aceleración y la velocidad son redondeadas al entero más cercano, y los tres parámetros de trayectoria son convertidos a hexadecimal como se ve:

$$A = 17 = 00\ 00\ 00\ 11\ hex \frac{CUENTAS}{MUESTRA^2}$$

$$V = 161,087 = 00\ 02\ 75\ 3F\ hex \frac{CUENTAS}{MUESTRA}$$

$$P = -120,000 = FF\ FE\ 2B\ 40\ hex\ CUENTAS$$

En un perfil de velocidad pueden existir ambos movimientos, el absoluto y el relativo, y al punto de unión de ambos se le llama punto de paro como se observa en la figura 4.1.

10. Ajuste del filtro PID.

Aspectos de fondo.

La respuesta transitoria de un sistema de control revela importante información sobre la calidad del control, un escalón en la entrada es fácil de generar, y es suficientemente drástico. La respuesta transitoria de un sistema de control es a menudo caracterizada por la respuesta para un escalón de entrada.

El paso de respuesta de un sistema de control puede ser caracterizada por tres atributos: máximo sobretiro, elevación de tiempo y establecimiento del tiempo. Estos atributos del paso de respuesta son definidos como sigue en una gráfica detallada en la figura 2.11.

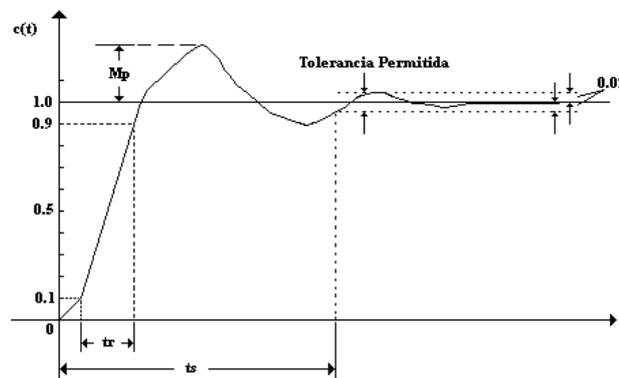


Figura 2.11, Paso de respuesta para una muestra del sistema.

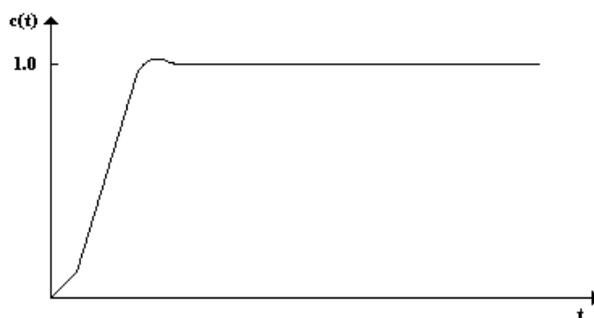


Figura 2.12, Respuesta de paso para un muestreo constante.

Un sistema de control que es críticamente muestreado, da una óptima ejecución. El paso de respuesta de un sistema de control que es críticamente muestreado, muestra la elevación mínima posible de tiempo, esto mantiene un rebase de cero y también cero resonancia como se ve en la figura 2.11.

El último objetivo de ajuste del filtro PID es para un muestreo del sistema de control del motor, dando un rastreo y estableciendo o fijando el tiempo como se muestra en la figura 2.12, la respuesta del filtro PID es la suma de los tres términos: un término proporcional, un término integral y un término derivativo. Cinco variables forman esta respuesta, en estas cinco variables se incluyen los tres coeficientes de ganancia K_p , K_i y K_d , el coeficiente del límite de integración il , y el coeficiente derivativo de la muestra ds . El filtro ajusta los coeficientes a valores iguales determinados para estas variables y son los valores que abarca el sistema de control.

Los coeficientes del filtro son determinados con una aproximación experimental en dos pasos. *En el 1er paso*, el valor de K_p , K_i y K_d (junto con il y ds) son variados y obtenidos sistemáticamente dentro de buenas características de respuestas razonables y mediante métodos *manuales y visuales*. Son usados para evaluar el efecto de cada coeficiente sobre el comportamiento del sistema. *En el 2do paso*, una señal de osciloscopio del sistema de paso de respuesta da información detallada sobre sistemas de muestreos, y los coeficientes del filtro determinados en el paso 1, son modificados para muestrear críticamente el sistema.

Paso 1 - Método visual y manual.

En el 1er paso, el valor de K_p , K_i y K_d , junto a il (límite de integración) y ds (coeficiente derivativo) son obtenidos sistemáticamente dentro de características buenas, mientras que las respuestas con métodos manuales y visuales y son usados para evaluar el efecto de cada coeficiente sobre el comportamiento del sistema.

Las siguientes 3 secciones son pasos ordenados para el ajuste del filtro PID.

Preparación del sistema.

La sección de iniciación, el programa de ajuste del filtro se ejecuta para preparar el sistema de ajuste del filtro. Esta sección inicia el sistema, predetermina los parámetros del filtro (K_p , K_i , $i_l = 0$, $K_d = 2$, $d_s = 1$), y los comandos del lazo de control para sostener el eje de la posición.

Después de ejecutar la sección de iniciación del programa del ajuste del filtro, ambos métodos desean que la actual posición del eje sea cero, por lo que la flecha puede estar estacionada. Cualquier desplazamiento del eje constituye un error de posición, pero con ambos K_p y K_i colocados a cero, el control de lazo puede no corregir este error.

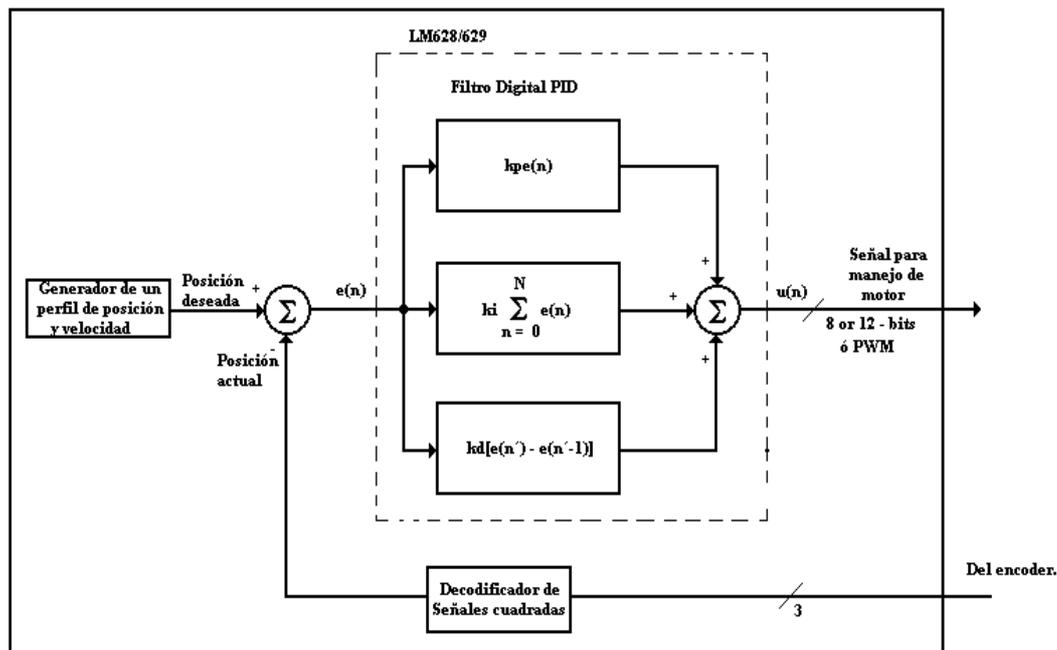


Figura 3.7 Diagrama a bloques del LM628/629.

A - Determinando el coeficiente de ganancia derivativo.

El término del filtro derivativo da un muestreo para eliminar la oscilación y minimizar el sobretiro, estabilizando el sistema. El muestreo está dado como una fuerza proporcional a la frecuencia de cambio de la posición de error, y la constante de proporcionalidad es $K_d \times d_s$.

El coeficiente Kd y ds son determinados con un algoritmo iterativo, el coeficiente Kd es incrementado sistemáticamente hasta que la flecha comience a oscilar a una frecuencia alta, el coeficiente es entonces incrementado por uno, el proceso de entrada se repite hasta que ds alcanza un valor apropiado para el sistema.

El período del sistema de muestreo coloca el intervalo de tiempo entre la actualización de la posición de error. El intervalo de la muestra derivativa es un entero múltiplo del período del sistema de la muestra. Colocando el intervalo de tiempo entre la posición sucesiva del error de las muestras usadas en el término derivativo afecta directamente al sistema de muestreo. El intervalo de la muestra derivativa será un pequeño tiempo de 5 a 10, tal que el sistema mecánico tendrá un tiempo uniforme, esto significa que muchos sistemas requieren un bajo ds en general, sin embargo, Kd y ds son colocados para dar el producto más grande de $Kd \times ds$, esto mantiene aceptable una vibración baja del motor.

B - Determinando un coeficiente de ganancia proporcional.

El error por las causas de carga inercial, la posición de error asociado al movimiento del eje, los disturbios externos y la carga de torque causan un error de desplazamiento, o sea un error de posición asociado con un eje estacionario o eje fijo.

El término proporcional del filtro da una fuerza restauradora para minimizar estos errores de posición, la fuerza restauradora es proporcional al error de posición, y se incrementa linealmente tal y como el error de posición es incrementado, entonces el coeficiente Kp de ganancia proporcional es la constante de proporcionalidad.

El coeficiente Kp , es determinado con un proceso iterativo, el valor de Kp es incrementada, y el sistema de muestreo constante es evaluado, esto es repetido hasta que el sistema sea críticamente muestreado. El sistema muestreado es evaluado manualmente. Retornando manualmente el eje, revelará cada incremento de Kp , incrementando el eje endurecido, el eje sentirá la carga de un resorte, y será forzada lejos de una posición deseada y liberará el resorte de la flecha retrocediendo. Si Kp es baja, el sistema está sobre muestreado, y la flecha se recupera lentamente. Si Kp es grande, entonces el sistema esta dentro del muestreo, y la flecha se recupera rápidamente, esto causa un sobretiro, o resonancia y posiblemente oscilación. El coeficiente de ganancia proporcional Kp es incrementado a un valor grande, esto no causa un sobretiro excesivo o resonancia, en este punto el sistema es críticamente muestreado, y por lo tanto da un óptimo rastreo estableciendo un tiempo.

C - Determinando el coeficiente de ganancia integral.

El término proporcional del filtro minimiza el doble error para la carga inercial y de torque. El término integral, sin embargo, da una fuerza correctiva que puede eliminar el siguiente error donde la flecha es girada, y el efecto de desviación de un torque con carga estática donde la flecha está estacionaria. Esta fuerza correctiva es proporcional al error de posición, y se incrementa linealmente con el tiempo (Figura 2.13). El coeficiente de ganancia integral Ki es la constante de proporcionalidad.

Valores altos de K_i dan una compensación rápida del torque, pero incrementa el sobretiro y la resonancia. En general, K_i puede ser colocado para valores pequeños que den un compromiso apropiado entre las características de los tres sistemas: Sobretiro, establecimiento de tiempo, y tiempo para cancelar los efectos de un torque con carga estática, o en sistemas con torques con carga estática insignificantes el valor de K_i será de cero.

La fuerza correctiva dada para el término integral incrementa linealmente con el tiempo. El coeficiente del límite de integración i_l actúa como un valor sujeto sobre esta fuerza para prevenir un efecto negativo en la integral, como se nota en la figura 2.13, donde i_l es el límite de la sumatoria de error (sobre el tiempo), no el producto de K_i y esta sumatoria. En muchos sistemas i_l puede ser colocado a un máximo valor sin tener cualquier efecto adverso. El término integral no afecta si se coloca como cero.

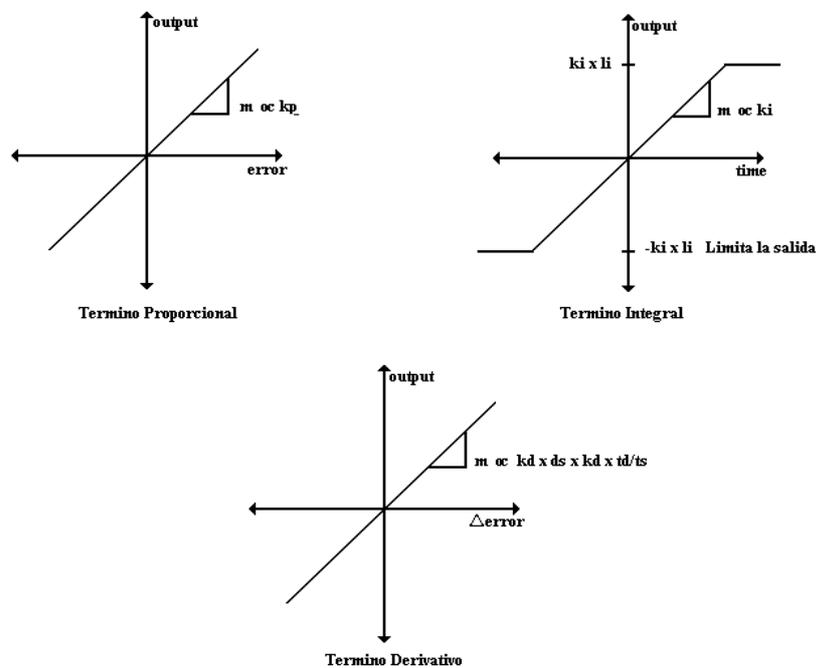


Figura 2.13 Componentes de fuerzas proporcional, integral y derivativo.

Paso 2 - Método de paso de respuesta.

En el segundo paso, para ajustar el filtro PID, una señal de osciloscopio del control del sistema de paso de respuesta se usa para evaluar exactamente sistemas de muestreo constante, y los coeficientes del filtro son determinados en un solo paso, son finamente ajustados para ajustar el sistema

Consideraciones.

Software.

En la sección de la generación de paso del programa de “ajuste de filtro”, el lazo de control da una señal pequeña respectivamente para el paso de entrada. Esto es, para ejecutar repetidamente un pequeño movimiento de la posición con una velocidad máxima y una aceleración grande, ver figura 3.7

Hardware.

Para el sistema de control de un motor, una señal de osciloscopio del sistema de paso de respuesta es un gráfico de la posición real del eje contra el tiempo después de un cambio pequeño e instantáneo, en la posición deseada.

Para un sistema basado en el MPID (LM628) no es necesario contar con un hardware extra, porque durante un paso, el voltaje que cruza el motor representa el sistema de paso de respuesta, y un osciloscopio es usado para generar un gráfico de esta respuesta.

Pero para un sistema basado en el MPID (LM629), si es necesario agregar un hardware extra para ver el sistema de paso de respuesta, porque durante un paso, el voltaje de salida de este circuito extra, representa el sistema de paso de respuesta, usando también un osciloscopio que es usado para generar un gráfico de esta respuesta.

Comandos de usuario.

Los comandos se categorizan por funciones:

1. - Iniciación.
2. - Interrupciones de Control.
3. - Filtros de Control.
4. - Trayectorias de Control.

COMANDO	TIPO	DESCRIPCIÓN	HEX	Nº Bytes
RESET	(1)	RESET LM629	00	0
PORT8	“	SALIDA 8 BITS	05	0
PORT12	“	SALIDA 12 BITS	06	0
DFH	“	DEFINE INICIO	02	0
SIP	(2)	FIJA POSICION INDICE	03	0
LPEI	“	INTERRUPCION EN ERROR	1B	2
LPES	“	PARADA EN ERROR	1A	2
SBPA	“	FIJAR PUERTO DE RUPTURA ABSOLUTO	20	4
SBPR	“	FIJAR PUERTO DE RUPTURA RELATIVO	21	4
MSKI	“	INTERRUPCION ENMASCARABLE	1C	2
RSTI	“	INTERRUPCION DE RESET	1D	2
LFIL	(3)	CARGA DE PARAMETROS DEL FILTRO	1E	2 A 10

UDF	(3)	ACTUALIZACION DATOS DEL FILTRO	04	0
LTRJ	(4)	CARGAR TRAYECTORIA	1F	2 A 14
STT	“	COMENZAR MOVIMIENTO	01	0
RDSTAT	(5)	LEER BYTE DE ESTADO	--	1
RDSIGS	“	LEER REGISTROS DE SEÑALES	0C	2
RDIP	“	LEER INDICE DE POSICIÓN	09	4
RDDP	“	LEER POSICION DESEADA	08	4
RDRP	“	LEER POSICIÓN REAL	0A	4
RDDV	“	LEER VELOCIDAD DESEADA	07	2
RDRV	“	LEER VELOCIDAD REAL	0B	2
RDSUM	“	LEER SUMA DE INT.	0D	2

TABLA 3-6 Lista de comandos de usuario.

11. Descripción de los programas.

INICIO: Inicialización

```

Inicio:  MOVX    @RST,LOW  Envía reset
         CALL   Delay_15  Espera 1.5 mseg
         CALL   Bit_b     Lectura del byte de estado
         SUBB  A,#C4h     ACC-C4h  C4h=1100 010 0
         JNC   Cnt       Si la diferencia es 0, ve a Continua
CALL     Bit_b     Lectura del byte de estado
         SUBB  A,#84h     ACC-84h  84h=1000 010 0
         JNC   Cnt       Si la diferencia es 0, ve a Continua
         JMP   Inicio    si no, vuelve a inicio
Cnt:     CALL   Bit_b
MOVX    @P0,RSTI  Envía clear interrupciones al MPID
         CALL   Bit_b     Lectura del byte de estado
         SUBB  A,#C0h     ACC-C0h  C0h=1100 000 0
         JNC   Fin       Si la diferencia es 0, ve a Fin
CALL     Bit_b     Lectura del byte de estado
         SUBB  A,#80h     ACC-80h  80h=1000 000 0
         JNC   Fin       Si la diferencia es 0, ve a Fin
         JMP   Inicio    si no, vuelve a inicio
Fin:     RET

```

Comentarios:

Nótese que en todas las comparaciones el Bit_busy es 0.

ESPERA: Mantener reset al menos 1.5 milisegundos

Delay_15

```

MOV TL1,#TL1   Precarga inferior contador
MOV TH1,#TL1   Precarga superior contador
SETB ET1       Habilita interrupción de contador
SETB PT1       Da prioridad 1
SETB TR1       Comienza la cuenta
ORL PCON,1     Modo bajo consumo
CLR TR1        Para contador
CLR PT1        Desactiva prioridad
CLR ET1        Deshabilita interrupción
RET

```

Comentarios:

Durante el delay, se pasa a modo IDLE, bajo consumo, porque la única operación es contar.

Obtención de la precarga.

Precarga = osc (Khz) * Tiempo espera (msg) / 12

LECTURA_SB: Lectura del Byte de estado

```

Bit_b:  MOVX @CS,LOW
MOVX @PS,LOW   Pon a 0 TTL /CS Y /PS
MOVX A,@ P0    Lee PUERTO 0
JB @B0,Bit_b  Si B0 ≠ 0, repite
RET           MPID no ocupado

```

Comentarios:

La instrucción “hardware” RDSTAT se ejecuta poniendo a 0 TTL /CS, /PS y /RD, pero en MOVX está implícito /RD a 0 TTL. Por otro lado, se pasa a Acumulador el byte entero, aunque se usa solo el Bit_busy, para poder utilizar la información acumulada posteriormente. El programa se asegura de que el MPID no está ocupado.

CARGA de COEFICIENTES del FILTRO PID

```

Flt_pid:  CALL Bit_b          Lectura byte de estado
MOVX     @P0,LFIL           Envía comando LFIL al MPID
CALL     Bit_b              Lectura byte de estado
MOVX     @P0,P_c            Envía palabra de control al MPID
CALL     C_pid              Envía coeficientes al buffer primario
CALL     Bit_b              Lectura byte de estado      (OPCIONAL)
MOVX     @P0,UDF            Actualiza datos del filtro  (OPCIONAL)
RET

```

Comentarios:

La descripción de comandos, coeficientes y palabras de control se explican en otro apartado. Las líneas opcionales se deben a que se puede proceder a la carga inmediatamente, o mantenerlas guardadas en buffer primario del MPID para actualizar con UDF con posterioridad, mientras el μ C realiza otra tarea.

COEFICIENTES del FILTRO PID: Decodificación de palabra de control y envío de datos.

```

C_pid:    JB    B3,Bit_2      Si bit 3 = 0 salta a Bit_2
          CALL Bit_b          Lectura byte de estado
MOVX     @P0,Kp              Envía coeficiente proporcional
Bit_2:   JB    B2,Bit_1      Si bit 2 = 0 salta a Bit_1
          CALL Bit_b          Lectura byte de estado
MOVX     @P0,Ki              Envía coeficiente integral
Bit_1:   JB    B1,Bit_0      Si bit 1 = 0 salta a Bit_0
          CALL Bit_b          Lectura byte de estado
MOVX     @P0,Kd              Envía coeficiente derivativo
Bit_0:   JB    B0,Bit_f      Si bit 0 = 0 salta a Bit_f
          CALL Bit_b          Lectura byte de estado
MOVX     @P0,Ii              Envía límites de integración
Bit_f:   RET

```

Comentarios:

Puede no ser enviado ningún coeficiente.

CARGA de TRAYECTORIA

```

Try:      CALL Bit_b      Lectura byte de estado
MOVX     @P0,LTRJ      Envía comando LTRJ al MPID
CALL     Bit_b          Lectura byte de estado
MOVX     @P0,P_c       Envía palabra de control al MPID
CALL     T_try          Envía datos de trayectoria al buffer primario
CALL     Bit_b          Lectura byte de estado      (OPCIONAL)
MOVX     @P0,STT       Inicia trayectoria          (OPCIONAL)
RET
  
```

Comentarios:

Las líneas opcionales, como en el filtro PID, se deben a que se puede proceder al arranque inmediatamente, o mantener los datos guardados en buffer primario del MPID para arrancar con STT con posterioridad, mientras el μC realiza otra tarea.

DATOS de TRAYECTORIA: Decodificación de palabra de control y envío de datos.

```

T_try:   JB    B5,Bit_3    Si bit 5 = 0 salta a Bit_3
         CALL Bit_b      Lectura byte de estado
MOVX     @P0,Ac          Envía dato de aceleración
Bit_3:   JB    B3,Bit_1    Si bit 3 = 0 salta a Bit_1
         CALL Bit_b      Lectura byte de estado
MOVX     @P0,Vel         Envía dato velocidad
Bit_1:   JB    B1,Bit_0    Si bit 1 = 0 salta a Bit_0
CALL     Bit_b          Lectura byte de estado
MOVX     @P0,Pos         Envía dato posición
Bit_0:   RET
  
```

Comentarios:

Puede no ser enviado ningún dato.

12. Gestión de interrupciones.

Se van a utilizar las entradas específicas /INT0 e /INT1 del 80251.

/INT0 viene, invertida, de la salida de interrupciones HI del MPID.

/INT1 viene del "switch" que corta la potencia por exceso de temperatura, a través de un opto acoplador para aislar las masas de potencia y digital.

Las lecturas de corrientes media y máxima del puente H se realizan de forma periódica y alternativa. Proceden de lecturas analógicas convertidas a digital por un convertor A/D que aporta su salida al puerto 2 del 80251. En él se leerán corrientes máxima y media, controlando el ADDRESS SELECT del convertor A/D, lecturas que se compararán con datos grabados en programa, en posiciones #Imax, #Imed.

/INT0 se sitúa en las direcciones 0003h a 000Ah (8X8 bits). Aquí se colocará una llamada a subrutina que descodifique el motivo de la interrupción del MPID y efectúe la operación pertinente en cada caso.

```
ORG    0003h      Inicio en 0003h
CALL   I_MPID     Llama al decodificador de errores
RETI                          Retorno de interrupción
```

La descodificación de errores se efectúa solicitando al MPID la información de su Byte de Estado, mediante una llamada RDSTAT, que es la misma utilizada para verificar el Bit Busy. Por tanto se empleará la misma subrutina Bit_b para solicitar la información, y se procederá a la descodificación.

/INT1 se sitúa en las direcciones 0013h a 001Ah (8X8 bits). Aquí se colocará una llamada a subrutina que prepare el sistema para reinicio, dado que el "switch" de temperatura mantendrá cortada la potencia hasta que la temperatura disminuya.

```
ORG    0013h      Inicio en 0013h
CALL   INT_T      Llama a tratamiento de temperatura excesiva
RETI                          Retorno de interrupción
```

```
INT_T:  MOVX      @RST,LOW  Envía reset al MPID
MOVX    @P1,Er_Temp      Envía al puerto 1 (display) mensaje de error de
                          Temperatura.
HALT                                Detén programa; espera RESET general manual. Reinicio.
JMP     Inicio
```

En el puerto 1 se ofrece, a través de un display BCD de 7 segmentos un código numérico, de 0 a 9, que identifica el tipo de interrupción detectado.

Posibles interrupciones del MPID

Al solicitar un RDSTAT, con motivo del envío por parte del MPID de una interrupción, obtenemos la información en el byte de estado compuesta por 8 bits.

De todos ellos, algunos son útiles tan solo en fase de depuración; otros son informativos y el bit 0 (Busy_bit), se utiliza como consulta.

A continuación se hace una breve descripción de cada uno, y se evalúa su uso o su enmascaramiento, con MSKI. Hay que considerar que el enmascaramiento afecta a la generación o no de la interrupción HI, pero la información se genera en el BYTE en cualquier caso. Salvo el bit 7 y el 0, todos generan interrupción (HI) y se pueden limpiar, bit a bit o en grupo, con RSTI. Todos son activos en lógica 1.

Bit 7: Motor apagado.

Se pone a 1 lógico cuando el motor se apaga. Esto sucede al enviar un reset, por un excesivo error de posición si se envió LPES, o si se usa LTRJ en una parada manual. El comando de arranque STT limpia el bit (lo pone a cero). No genera interrupción, luego será solo consultivo. Esta información se encuentra también en bit 2.

Bit 6: Breakpoint alcanzado.

Se pone a 1 lógico al alcanzar una posición prefijada por SBPA o SBPR. Se usa en depuración; no lo vamos a considerar y será enmascarado.

Bit 5: Error de posición excesivo.

Viene determinado en sus límites por los comandos LPEI o LPES.

Bit 4: Desbordamiento numérico.

Sucede al excederse la capacidad de posición de datos en los cálculos en modo velocidad. La información almacenada será errónea, y se debe corregir. La solución será volver a una situación preestablecida y reiniciar o, si se repitiera, modificar los datos.

Bit 3: Llegada de pulso índice.

Se activa si se envió SIP. Se utiliza en depuración. Será enmascarado.

Bit 2: Trayectoria completada.

Indica el fin con éxito de los comandos de trayectoria LTRJ, STT. Se usará para verificar el fin correcto. Es la OR de los bits 7 y 10.

Bit 1: Error de comando.

Cuando se intenta leer datos y procede una escritura o viceversa. Sólo se usará en depuración.

Bit 0: Ya descrito en otros apartados, indica cuándo es o no posible la comunicación con el MPID, y se usa continuamente. No genera interrupción.

Además, mediante el comando RDSIGS, obtenemos información adicional, que no genera interrupción HI:

Los bits 1 a 7 replican el byte de estado. El bit 0 se activa al enviar un comando SIP.

Bit 8: Modo salida de 8 o 12 bits. No es aplicable en nuestro modelo.

Bit 9: Apagado en caso de error de posición excesivo.

Se activa al ejecutar LPES y lo limpia LPEI. Solo útil en depuración.

Bit 10: Indica objetivo cumplido tras enviar STT. Esta información se encuentra en bit 2.

Bit 11: Modo velocidad. Indica esta selección. No es aplicable en nuestro modelo, porque el control va a ser automático, resultando un dato inútil.

Bit 12: Forward Direction. Solo en modo velocidad. Informativo.

Bit 13: Ejecutado UDF. Indica actualización de coeficientes del filtro, sin ejecución. Aplicable sólo en depuración.

Bit 14: Aceleración cargada, pero no ejecutada. Caso similar al bit 13.

Bit 15: "Host Interrupt". (HI). Es redundante, en "software" de la señal HI.

En definitiva, para una programación normal y ya depurada, solo nos interesa detectar la información de los bits 2, 4, 5 y periódicamente el bit 0. Podemos enmascarar los bits 1, 3, 6 y 7 y prescindir del comando RDSIGS.

Esto nos deja como significativos en el byte de estado: Trayectoria completada (bit 2), Desbordamiento (bit 4), Error de posición (bit 5), que pueden ser leídos bit a bit. Las combinaciones posibles serán:

bit	7	6	5	4	3	2	1	0	HEX
	0	0	0	0	0	0	0	0	0h
	0	0	0	0	0	1	0	0	4h
	0	0	0	1	0	0	0	0	10h
	0	0	0	1	0	1	0	0	14h
	0	0	1	0	0	0	0	0	20h
	0	0	1	0	0	1	0	0	24h
	0	0	1	1	0	0	0	0	30h
	0	0	1	1	0	1	0	0	34h
	1	0	0	0	0	0	0	0	80h
	1	0	0	0	0	1	0	0	84h
	1	0	0	1	0	0	0	0	90h
	1	0	0	1	0	1	0	0	94h
	1	0	1	0	0	0	0	0	A0h
	1	0	1	0	0	1	0	0	A4h
	1	0	1	1	0	0	0	0	B0h
	1	0	1	1	0	1	0	0	B4h

Para nuestro objetivo es suficiente considerar el byte de estado, que leemos y pasamos a acumulador con la subrutina Bit_b.

Descodificación de interrupciones MPID (HI)

(34)

I_MPID: CALL Bit_b Ejecuta Busy_bit; acumulador cargado con puerto 0.
 MOVX @P1,Er_4 Envía al puerto 1 (display) mensaje de error desconocido.

 MOV B,A Pasa la información del acumulador al registro B.

 CALL Bit_b

 MOVX @P0,RSTI Envía clear interrupciones al MPID

 MOV A,B Carga acumulador con registro B (byte de estado)

JB @B2,E0 Si bit 2 es cero salta a E1.

 MOVX @P1,Er_1 Envía al puerto 1 (display) mensaje de trayectoria completada.

 RET Vuelve a programa principal. No hay opción de 4 o 5.

E1: JB @B4,E2 Si bit 4 es cero, salta a E2.

 MOVX @P1,Er_2 Envía al puerto 1 (display) mensaje de error de desbordamiento.

 JB @B5,E2 Verifica si esta activado error de posición.

 JMP Flt_pid Si no, salta a cargar datos de nuevo.

E2: JB @B5,E3 Si bit 5 es cero salta a E3.

 MOVX @P1,Er_3 Envía al puerto 1 (display) mensaje de error de posición

E3: MOVX @RST, LOW Envía reset al MPID

 HALT Detén programa; espera RESET general manual.

Reinicio.

 JMP Inicio

Comentarios:

Primero se pasa a registro B la información del byte de estado, porque el acumulador va a ser usado de nuevo, y queremos antes que nada limpiar las interrupciones. Luego se devuelve la información al acumulador para operar y se envía al display mensaje de error desconocido, que se modificara solo si aparece otro identificado.

Si la interrupción se produjo por trayectoria completada, se excluyen las otras y no queda sino retornar a programa principal, indicando en display la incidencia.

Si se activo por desbordamiento, primero se verifica si se ha activado también error de posición, lo que resetea el sistema. En otro caso se repite la operación de carga de datos y arranque, reflejando la incidencia en display. Puede suceder que se repita el error: Si no se activa otra interrupción al repetir, se trata de la introducción de datos erróneos en la grabación del chip o en los cálculos.

Si se activo error de posición o no sabemos cuál es la causa de la interrupción, se envía el mensaje al display y en ambos caso se resetea el MPID. Si permanece activado el error desconocido, se debe investigar la causa antes de continuar.

Lectura de corrientes máxima y media

(21)

Lee_I: MOVX A,@P2 Lee PUERTO 2, I_{max} y pasa a acumulador (ACC)

SUBB A,#I_{max} ACC-I_{max}; diferencia lectura y valor prefijado

JNB AC,I_{med} Si bit AC en PSW es 1 (negativo) I máxima correcta, lee I media

CALL EI_{max} si es 0 (positivo), llama a error I_{max}.

I_{med}: MOVX @ADS,LOW escribe 0 en ADS (P1.0) para direccionar el A/D en I media

MOVX A,@P2 Lee PUERTO 2, I_{med} y pasa a acumulador (ACC)

SUBB A,#I_{med} ACC-I_{media}; diferencia lectura y valor prefijado

JNB AC,Fin Si bit AC en PSW es 1 (negativo) I media correcta, acaba.

CALL EI_{med} si es 0 (positivo), llama a error I_{media}.

Fin: MOVX @ADS,HIGH Escribe 1 en ADS (P1.0) para direccionar el A/D en I_{max}.

RET Retorno a programa principal.

Comentarios:

Preseleccionado por "hardware" el convertidor A/D en lectura de I máxima, se toma la lectura y se compara con la grabada en la posición I_{max}. Se consulta el bit AC de la palabra de estado (PWS) para ver si la bandera de acarreo indica positivo, lo que significaría que la lectura es mayor que la prefijada y por tanto esta fuera de tolerancia. En este caso, se envía el programa a secuencia de error de I_{max}. Si el acarreo es negativo, I máxima correcta, se selecciona en A/D la salida de I_{media} y se procede igual que en el caso anterior. Si ambas son correctas, se vuelve a seleccionar A/D en I máxima para siguiente lectura, y se vuelve a programa principal.

Error de corriente máxima:

EI_{max}: MOVX @RST,LOW Envía reset al MPID

MOVX @P1,Er_{Imx} Envía al puerto 1 (display) mensaje de error de I máxima.

HALT Detén programa; espera RESET general manual. Reinicio.

JMP Inicio

Error de corriente media:

```
El_med: MOVX      @RST,LOW  Envía reset al MPID
        MOVX      @P1,Er_Imd Envía al puerto 1 (display) mensaje de error de I media.
        HALT      Detén programa; espera RESET general manual. Reinicio.
        JMP      Inicio
```

Comentarios:

En ambos casos, se resetea el MPID, se envía el mensaje de error correspondiente al display y se detiene el programa, para reiniciar con RESET general manual cuando el operador lo estime. Con reset el programa se reiniciará en la posición Inicio; la última línea es una precaución.

El programa ocupa unos 220 bytes, 250 aportando espacios de seguridad, e incluyendo los códigos de error del display, es decir 2 kbs, y la memoria dispone de 16 kbs, lo que supone 14 Kbs libres.

El MPID necesita como máximo 9 bytes en cada trayectoria mas 2 bytes para las corrientes estimadas por trayectoria, espaciando, 15 bytes, se pueden programar datos para unas 100 trayectorias, añadiendo un programa que las lea secuencialmente, y tras la depuración de datos y coeficientes de cada una.

/EA a nivel alto, el procesador ejecuta instrucciones desde la EPROM interna hasta 4096 (4Kbytes), y las superiores a externa.

Tratamiento de tablas de datos:

La instrucción MOV DPTR#,datos, de 16 bits permite transferir datos de 16 bits desde memoria EPROM interna o externa. DPTR es el puntero de datos que se controla con un bucle contador para obtener la cantidad de datos y en la posición que queramos.

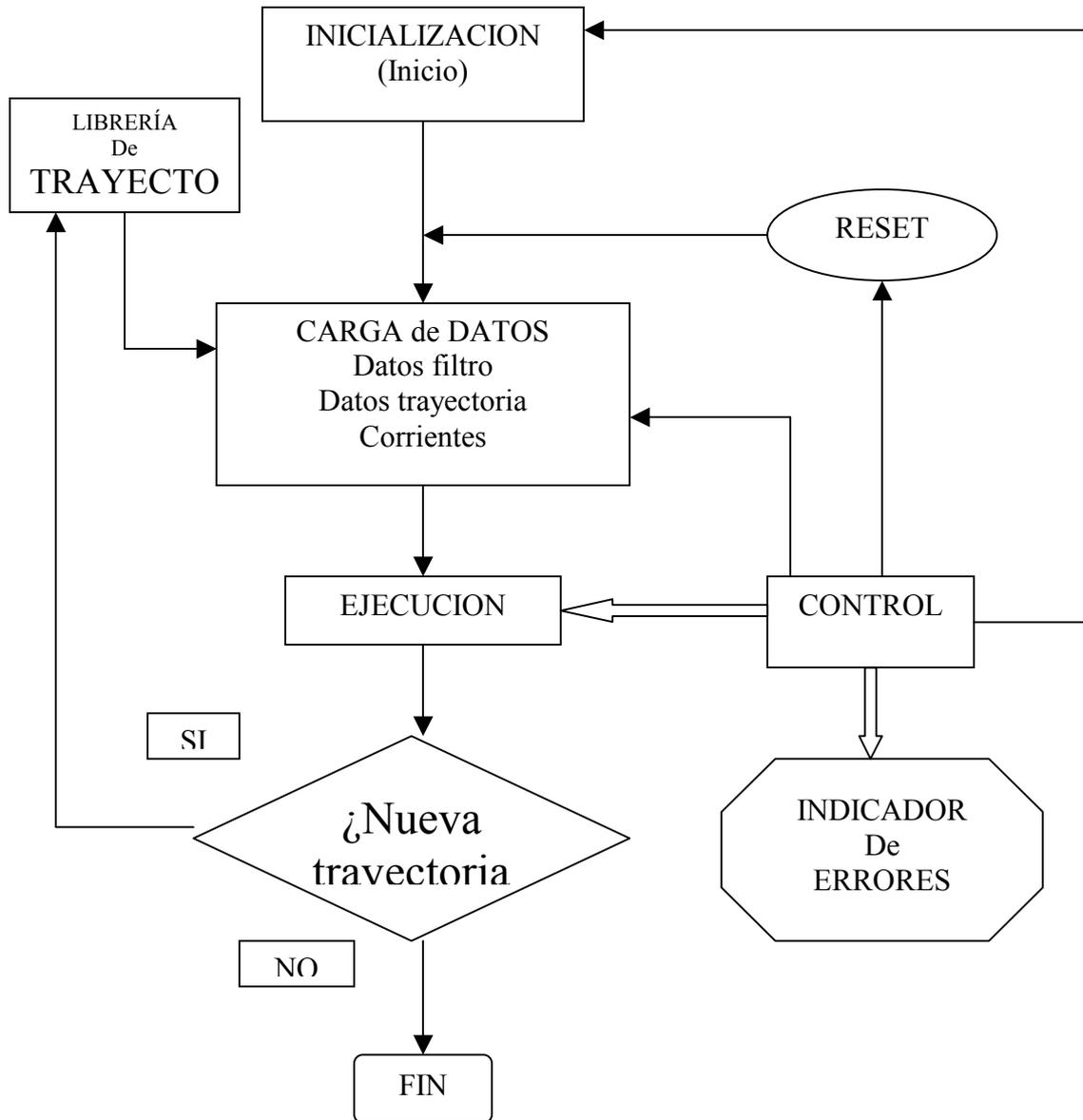
El bucle se establece mediante una constante de inicio de tabla y una instrucción del tipo

```
MOV    A,@A+ DPTR    Lee memoria de programa en (A+DPTR). Carga en
Acumulador.
```

O bien usando el PC (Program Counter) para indicar la dirección base de la tabla:

```
MOV    A,@A+PC    Lee memoria de programa en (A+PC). Carga en Acumulador.
```

DIAGRAMA de FLUJO TIPICO



13. ERRORES

La tabla de indicadores de error / informacion y su tratamiento, si procede, queda como sigue:

DISPLAY	SENAL	SIGNIFICADO	Tratamiento
0	Er_0	No hay problemas	Ninguno
1	Er_1	Trayectoria completada con exito.	Ninguno. (informativo).
2	Er_2	Error de desbordamiento de datos.	Nueva carga / Inicializacion.
3	Er_3	Error de posicion.	Inicializacion.
4	Er_Imx	Error corriente maxima.	Reset MPID. Reset manual.
5	Er_Imd	Error corriente media.	Reset MPID. Reset manual.
6	Er_Tmp	Temperatura	Reset.MPID. Reset manual.
9	Er-4	Error desconocido.	Reset MPID. Investigar / Reset manual.

DOCUMENTOS N° 3

PRESUPUESTO

1. Presupuesto.

El Presupuesto de Ejecución de Material se obtendrá, especificando la cantidad de cada una de las distintas Unidades Constructivas y sus correspondientes precios unitarios.

Para obtener el Presupuesto General, será preciso incrementar, el Presupuesto de Ejecución de Material en los porcentajes de Gastos Generales; Beneficio Industrial, Dirección de Obra y cualquier otro que proceda.

DOCUMENTOS N° 4

PLIEGO DE CONDICIONES

INDICE

1	Objetivos.	_____
.		
2	Ejecución del trabajo.	_____
3	Recepción de obra.	_____
4	En sayo electrónico.	_____
5	Instalación.	_____

1. Objetivo.

Este Pliego de Condiciones determina las condiciones mínimas aceptables para la ejecución del montaje, hardware del sistema, calculo del sistema, construcción de la tarjeta de control y diseño, software del sistema.

Los Pliegos de Condiciones Particulares podrán modificar las presentes prescripciones.

2. Ejecución del trabajo.

La ejecución de los Trabajos se realizara paso a paso y deberán realizarse conforme a las reglas del arte.

3. Recepción de obra.

Durante los trabajos o una vez finalizados, el Director de Obra podrá verificar que el trabajo y montaje están de acuerdo con las especificaciones de este Pliego de Condiciones.

Una vez finalizado el Proyecto, se deberá solicitar la oportuna recepción global de la obra.

4. Ensayo electrónico.

Los ensayos se realizaran progresivamente durante la ejecución del Proyecto, bien en conjunto o por separado para no dañar las partes del montaje ya comprobadas.

5. Instalación.

Una vez comprobado el apartado anterior se procederá a la ejecución y instalación del mismo, atendiendo a los requisitos y normas espuestas en el apartado 4 de los Documentos de la Justificación.

DOCUMENTOS N° 5

PLAMOS (ESQUEMAS)

INDICE

- 1 Esquema del convertidor de 24V DC a +5,+15 y -15V DC.

- 2 Esquema del convertidor preparado para el diseño de placa BOARD.

- 3 Esquema completo del dispositivo.

- 4 Esquema completo del dispositivo preparado para el diseño de placa BOARD.

- 5 Esquema orientativo para la programación de la Eprom del Micro.

- 6 Placa BOARD de los componentes.

- 7 Placa BOARD de pistas 4 capas.

- 8 Placa BOARD pista 1.

- 9 Placa BOARD pista 2.

- 10 Placa BOARD pista 3.

- 11 Placa BOARD pista 4.

- 12 Placa BOARD Talados.

DOCUMENTOS N° 6

ANEXO

1. ANEXO.

El anexo estará compuesto por un Cd-Rom, donde estarán incluidos en formato PDF todos los manuales de los distintos dispositivos utilizados en el Proyecto y también estará incluidos en el Cd-Rom en ficheros de ORCAD, que están compuestos por los esquemas de las de las placas de pistas para la posible utilización en el desarrollo de la ejecución del Proyecto.