



FACULTAD DE INGENIERIA



UNIVERSIDAD NACIONAL
AUTÓNOMA DE MEXICO

**“CONTROL DE PROPORCIÓN DE CARGA Y
DETECCIÓN DE LIMITES DE OPERACIÓN EN
PRUEBAS CON GATOS SERVOHIDRAULICOS”**

TESIS QUE PRESENTA

GABRIEL JESÚS VELASCO GONZÁLEZ

PARA OBTENER EL GRADO DE

INGENIERO ELÉCTRICO Y ELECTRÓNICO

COMITÉ DE TESIS

ING. ENRIQUE GOMEZ ROSAS

ING. RODOLFO PETERS LAMMEL

ING. GABRIEL ALEJANDRO JARAMILLO MORALES

M. I. ANTONIO SALVÁ CALLEJA

ING. GLORIA MATA HERNÁNDEZ

*11 DE OCTUBRE
MEXICO, DISTRITO FEDERAL*



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

AGRADECIMIENTOS

La presente tesis refleja el arduo trabajo realizado durante mi carrera de licenciatura y el cual no hubiese sido posible sin aquellas personas que a su manera me brindaron su apoyo y las cuales merecen el agradecimiento.

A mis padres, Isabel y Mario, por toda la confianza que me brindaron hasta alcanzar este sueño, que también es suyo. Me han otorgado la mejor de las herramientas para la vida, muchas gracias.

A mis hermanos, Isabel, Abel, Joel y Mario, les agradezco, su ejemplo, ayuda y apoyo económico. De no haber sido por ustedes éste camino habría sido muy pesado y quizá interminable, este título es suyo por igual.

A mi esposa Abby por todo el cariño y apoyo brindado durante toda la carrera. Eres una de las maravillas que me ha otorgado la universidad.

Agradezco al departamento de Instrumentación del Instituto de Ingeniería. A mi director de tesis el Ing. Enrique Gómez quien me enseñó a aterrizar mis conocimientos y los amplió, al Ing. Rodolfo Petters profesor, mentor y amigo a quien le reconozco su valioso apoyo dentro y fuera del aula de clases.

A mis amigos Cesar, Iván, Tenshi, Alex, Virgilio, Schofi, Lucho y Efra, quienes trabajaron a mi lado, me apoyaron y amenizaron los estudios.

Y a todos mis profesores por haber compartido sus conocimientos.

La presente tesis la dedico a mi hijo Galeb quien es mi motivación para seguir adelante.

INDICE

	PAGINA
1. SERVOSISTEMAS	3
1.1 Tipos de sistemas de control	3
1.1.1 Lazo abierto	3
1.1.2 Lazo cerrado	3
1.2 Componentes de los sistemas de control	4
1.2.1 Transductores y sensores	4
1.2.1.1 De posición.	4
1.2.1.2 De desplazamiento	5
1.2.1.3 De velocidad.	5
1.2.1.4 De presión.	5
1.2.1.5 De temperatura.	6
1.2.2 Elementos actuadores	6
1.2.3 Comparadores	7
2. TIPOS DE SEÑAL	8
2.1 CLASIFICACIÓN DE LAS SEÑALES	8
2.1.1 Señales de tiempo continuo y de tiempo discreto.	8
2.1.2 Señal par e impar	10
2.1.3 Señales periódicas y no periódicas	11
2.1.4 Señales determinísticas y aleatorias	12
3. CONVERTIDORES DE SEÑAL	13
3.1 CONVERTIDOR ANALÓGICO-DIGITAL (CAD)	13
3.1.1 Principios operativos de los CADs.	13
3.1.2 Muestreo de señales analógicas. Teorema de Shannon	15
3.2 CONVERTIDOR DIGITAL-ANALÓGICO (CDA)	15
3.3 EL PWM	16
3.3.1 Parámetros importantes	17
3.3.2 Aplicaciones	17
3.3.2.1 En relación con los motores	18
3.3.2.2 Como parte de un convertidor CAD	18
3.3.2.3 Regulación de tensión	18
3.3.2.4 Potencia de entrega	19
4. MICROCONTROLADORES	20
4.1 QUE ES UN MICROCONTROLADOR	20
4.2 EL PIC	21
4.2.1 Juego de instrucciones y entorno de programación	22
4.2.1.1 Tamaño de palabra	22
4.3 Características	23
4.3.1 EL Procesador o CPU.	23
4.3.2 Frecuencia	23
4.3.3 EL PWM	25
4.3.4 Conversor A/D:	25

4.3.5	Puerto de comunicación serie síncrono	28
4.3.6	USART	28
4.3.7	Registro STATUS	29
4.3.8	Registro OPTION	30
4.4	Programación del PIC	30
4.4.1	Programadores	31
4.5	PICs modernos	31
4.5.1	Clones del PIC	31
4.5.2	PICs gíreles	31
4.5.3	PICs para procesamiento de señal (dsPICs)	31
4.5.4	PICs más comúnmente usados	32
4.6	Microcontrolador PIC16F873	32
4.6.1	Principales características	32
4.6.2	Descripción de los terminales	33
4.6.3	El PIC16F873 por dentro	34
4.6.4	Organización de la memoria.	34
4.6.5	Organización de la memoria	36
4.7	EL PIC18FF4436	37
4.7.1	Principales características	37
4.7.2	Descripción de los terminales	38
4.7.3	El PIC18F4439 por dentro	39
4.7.4	Organización de la memoria	40
4.7.5	Organización de la memoria. Memoria de datos	41
5.	AMPLIFICADOR OPERACIONAL (OPAM)	43
5.1	Comportamiento en continua (DC)	43
5.1.1	Lazo abierto	43
5.1.2	Lazo cerrado	44
5.2	Comportamiento en alterna (AC)]	44
5.2.1	Análisis	44
5.2.1	Configuraciones	44
5.2.1.1	Comparador	44
5.2.1.2	Seguidor	45
5.2.1.3	Inversor	45
5.2.1.4	No inversor	46
5.2.1.5	Sumador inversor	46
5.2.1.6	Diferenciador	46
5.2.1.7	Integrador ideal	47
5.2.1.8	Derivador ideal	47
5.2.1.8	Otros	47
5.3	APLICACIONES	48
5.4	ESTRUCTURA	48
5.5	PARÁMETROS	50
5.6	LIMITACIONES	50
5.6.1	SATURACIÓN	50
5.6.2	TENSIÓN DE OFFSET	50

5.6.3	CORRIENTES	51
5.6.4	CARACTERÍSTICAS TENSIÓN-FRECUENCIA	51
5.6.5	CAPACITANCIA	51
5.6.6	DERIVA TÉRMICA	51
5.8	EFECTO CARGA	52
6.	EL ATS	53
6.1	El sistema ATS	53
6.2	DIAGRAMA DE BLOQUES DEL ATS	53
6.3	EL ACTUADOR DEL ATS	54
6.4	EL MÓDULO DE CONTROL	56
6.5	A SERVO-VÁLVULA DEL ATS	56
7.	DISPOSITIVO DE PROPORCIONALIDAD ENTRE CARGA	57
7.1	Sistema analógico	60
7.1.1	Diseño	60
7.1.2	Diagrama del sistema analógico.	65
7.2	Sistema Digital	65
7.2.1	Diseño	65
7.2.1.2	Diagrama de flujo.	67
7.2.3	Diagrama del sistema digital.	79
7.2.4	Programa del microcontrolador.	79
8.	DISPOSITIVO DE DETECCIÓN DE LÍMITES EN OPERACIÓN.	86
8.1	Sistema analógico-digital.	87
8.1.1	Diseño.	87
8.1.2	Diagrama del sistema.	89
8.1.3	Programa del PIC16F873 para el detector de límites de carga y desplazamiento.	89
8.1.3.1	Diagrama de flujo.	89
8.1.3.2	Programa	92
8.2	Sistema digital.	94
8.2.1	Diseño.	94
8.2.2	Programa para el PIC18F4439 para el detector de límites de carga y desplazamiento.	94
8.2.2.1	Diagrama de flujo.	94
8.2.2.2	Programa	108
9.	RESULTADOS	118
9.1	Dispositivo de detección de límites.	118
9.2	Dispositivo de proporcionalidad de carga.	121
10.	CONCLUSIONES	123

Objetivo de la propuesta:

Desarrollar un dispositivo electrónico para el sistema servo-hidráulico ATS, del Instituto de Ingeniería, que mantenga proporcionalidad de carga, entre los cuatro gatos del sistema; además de limitar la carga y el desplazamiento a fin de evitar valores no deseados durante la ejecución o ajuste de un ensayo.

Definición del problema:

La prueba implementada, sobre la estructura de concreto armado, requiere que los esfuerzos aplicados por los gatos hidráulicos tengan una proporcionalidad entre los verticales y los horizontales para satisfacer las condiciones del ensayo. Así mismo al realizar la prueba, se pueden dar condiciones que pongan en peligro tanto la ejecución de la misma como la integridad del personal y del equipo.

El reproducir un evento en un laboratorio, tiene como fin el controlar el mayor número de variables del sistema para poder realizar un análisis posterior del mismo. Para este caso, la carga aplicada en forma vertical a la estructura, debe mantener una relación con la carga aplicada horizontalmente, esto con el fin de poder simular una estructura de concreto de mayor altura.

El ensayo realizado, pretende llevar a la estructura de concreto a punto de falla, por lo que el experimento puede considerarse de riesgo para el personal que realiza la prueba y para el equipo que hace fallar la estructura. Un detector de límites de operación es implementado como medida de seguridad, avisando al usuario cuando el sistema supera los valores de carga y desplazamiento establecidos para la prueba, y realizando un paro de la prueba si el operador así lo desea.

Método:

El dispositivo que mantendrá la relación entre las cargas aplicadas horizontalmente y verticalmente, tomará lectura de los valores en operación proporcionados por el servo-sistema de las dos cargas aplicadas horizontalmente y las procesará entregando el valor con el que deben operar los gatos hidráulicos que aplican las cargas verticales. El procesamiento consta de realizar el producto de la suma o resta de los valores de carga horizontal, debido a que las cargas verticales trabajan en antiparalelo, por un factor de atenuación (alfa) generando la señal que es aplicada al servo-sistema como el valor de operación para las cargas verticales.

En forma paralela, otro dispositivo se encargará del control de la detección de los límites en operación del sistema servo-hidráulico. Las señales entregadas por el servo-mecanismo de los niveles de carga y desplazamiento deben de ser analizadas por el dispositivo, y éste debe de generar una respuesta al sistema cuando los límites, establecidos por el usuario, hallan sido alcanzados durante la operación. Las respuestas generadas por el dispositivo son un aviso al usuario, o un aviso y paro de la prueba.

Las señales proporcionadas por el servo-sistema son analógicas, por lo que la captura para el procesamiento y el mismo procesamiento puede ser de forma digital o analógica. Se desarrollarán para ambos dispositivos los dos tipos de sistemas con el fin de comparar el funcionamiento teórico de ambos e implementar al sistema ATS el que se considere más conveniente.

Inventario de materias/ temas de la carrera que se utilizan para el desarrollo del seminario / tesis

Análisis de Circuitos Eléctricos
Análisis de Sistemas y Señales
Circuitos Integrados Analógicos
Diseño Digital
Medición e Instrumentación
Microprocesadores y Microcontroladores

Resultados esperados:

Obtener los sistemas de control, y de proporcionalidad de carga durante la operación de los gatos hidráulicos del servo-mecanismo y el de detección de los límites de carga y desplazamiento, a ser implementados a partir de la comparación de entre los dos dispositivos, analógico y digital, diseñados para cada caso.

1. SERVOSISTEMAS

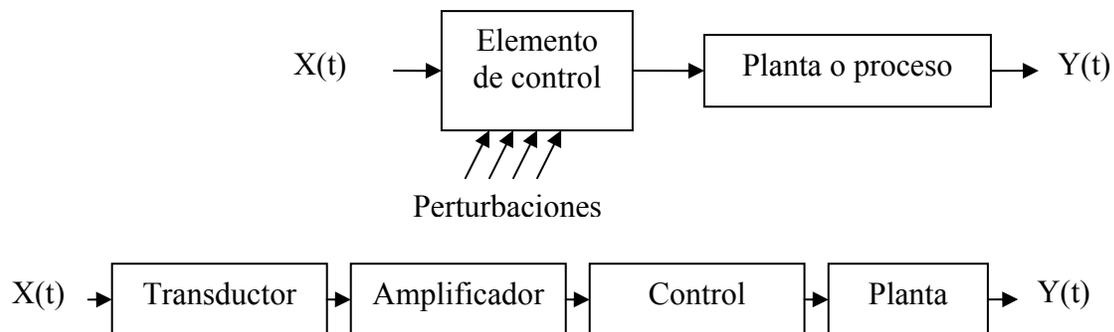
Los procesos de control han ido adquiriendo una gran importancia en los procesos industriales y automatización en general, siendo su finalidad proporcionar una respuesta adecuada a un estímulo determinado

Ya desde la segunda mitad del siglo XIX los ingenieros inventaron máquinas capaces de regular su actividad por ellas mismas; a estas máquinas las llamamos servomecanismos. Se trata de dispositivos que son capaces de captar información del medio y por tanto modificar sus estados en función de las circunstancias y regular su actividad de cara a la consecución de una meta.

1.1 Tipos de sistemas de control

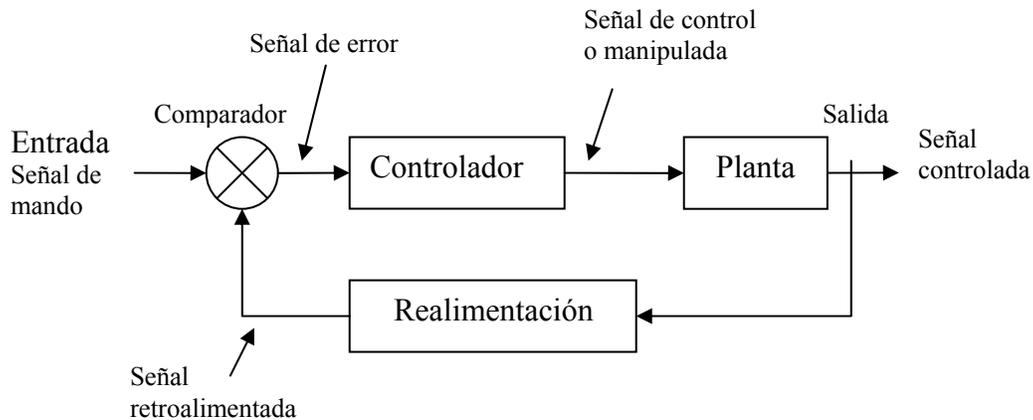
1.1.1 Lazo abierto

Son sistemas donde la salida no tiene efecto sobre la señal de control. En estos sistemas la variable de salida puede distanciarse considerablemente de la deseada debido a las perturbaciones.



1.1.2 Lazo cerrado

Son sistemas donde la salida tiene efecto sobre la señal de control, por lo que en estos sistemas las perturbaciones tienen menos incidencia sobre las variables de salida en función de la especificación de entrada.



1.2 Componentes de los sistemas de control

1.2.1 Transductores y sensores

Los transductores son elementos que transforman una magnitud física en otra que puede ser interpretada por el sistema de control.

El captador es la parte del transductor que recibe la magnitud física que deseamos transformar. No obstante, un mismo elemento puede recibir el nombre de transductor si se encuentra en la entrada del sistema de control, y captador si se encuentra en lazo realimentado.

Transductores y sensores más usuales.

1.2.1.1 De posición.

Nos proporciona información sobre la presencia de un objeto. Los podemos clasificar según su principio de funcionamiento en:

Finales de carrera. A su vez se clasifican en función del elemento sensor en: de palanca, de émbolo o de varilla. Son dispositivos del tipo todo o nada cuyo principio de funcionamiento es similar a un interruptor eléctrico.

Detectores de proximidad inductivos. Constan de una bobina eléctrica que puede crear un campo magnético estático o no. Al acercarse un objeto metálico a la bobina se modifica la inducción de ésta por efecto de histéresis.

Detectores de proximidad capacitivos. Se basan en la variación de la capacidad C que experimenta un capacitor cuando se modifica la separación d que existe entre las armaduras, el dieléctrico K o las áreas A enfrentadas de las armaduras A .

$$C = \frac{K\epsilon_0 A}{d}$$

Detectores de proximidad ópticos. Permiten detectar todo tipo de objetos, tanto sólidos como líquidos. Se basan en la reflexión y detección de un haz luminoso que normalmente es infrarrojo para evitar luces parásitas.

1.2.1.2 De desplazamiento

Nos proporciona información sobre la posición relativa de un objeto. Los podemos clasificar según su principio de funcionamiento en:

Radar. Se basa en la emisión modular de radiaciones electromagnéticas y la captación de los ecos que se producen. La distancia del objeto que refleja la radiación electromagnética queda determinado por el tiempo transcurrido entre la emisión y la recepción. Este sistema nos permite determinar grandes distancias.

$$d = c \cdot \frac{\Delta t}{2}$$

Siendo c la velocidad de propagación de las ondas.

Detectores lineales de distancias pequeñas. Están basados fundamentalmente en un potenciómetro lineal para medir distancias rectas, o un potenciómetro angular para medir variaciones angulares.

1.2.1.3 De velocidad.

Permite medir la velocidad lineal o angular de un objeto expresada en r.p.m. Su principio de funcionalidad es muy similar al de un dínamo.

1.2.1.4 De presión.

Nos proporciona información sobre la presión que está ejerciendo un fluido o un sólido sobre otro. Los podemos clasificar según su principio de funcionamiento en:

Mecánicos. Se basan en el desplazamiento o deformación de ciertos elementos del transductor.

Electromecánicos. Se basan en la deformación o desplazamiento de algunas partes del transductor que son usadas para actuar sobre elementos eléctricos (potenciómetros, capacitores y bobinas), modificando una tensión o una intensidad de corriente eléctrica.

Dentro de este grupo tenemos las galgas extensiométricas que son unos elementos metálicos con una forma determinada, están basadas en la variación de resistencia eléctrica que experimentan algunas aleaciones cuando se modifica su forma.

Piezoeléctricos. Están basados en las propiedades de ciertos materiales como el cuarzo que al ser sometidos a una presión generan una tensión eléctrica entre sus caras proporcional a la presión generada entre las mismas..

1.2.1.5 De temperatura.

Nos proporciona información sobre la temperatura ambiental o la de un objeto cercano. Los podemos clasificar según su principio de funcionamiento en:

Dilatación. Se basa en el efecto de cambio de dimensiones que sufren los materiales cuando se modifica su temperatura.

Termorresistencias. Se basa en la variación de la resistencia eléctrica que experimentan los metales con la temperatura.

$$R_T = R_0(1 + \alpha \cdot \Delta T)$$

Donde R_0 representa la resistencia eléctrica del metal a temperatura normal de 25° C y α el coeficiente térmico propio de cada metal.

Termistores. Se basan en la resistencia de variación eléctrica que sufren los semiconductores como consecuencia de la temperatura. Se denominan NTC si la resistencia disminuye al aumentar la temperatura y PTC si aumenta con la temperatura.

Termopares. Consisten en la unión de dos metales distintos y se basan en la generación de una fuerza electromotriz proporcional a la diferencia de temperaturas aplicadas a las uniones.

Pirómetros de radiación. Se utilizan para medir grandes temperaturas y se basan en que todo cuerpo caliente emite una radiación proporcional a la cuarta potencia de su temperatura.

1.2.2 Elementos actuadores.

Son los elementos de un sistema de control que nos proporcionan las variables de salida. Los más usuales son:

Electro-válvula. Es una válvula accionada eléctricamente que permite la regulación del caudal de un fluido.

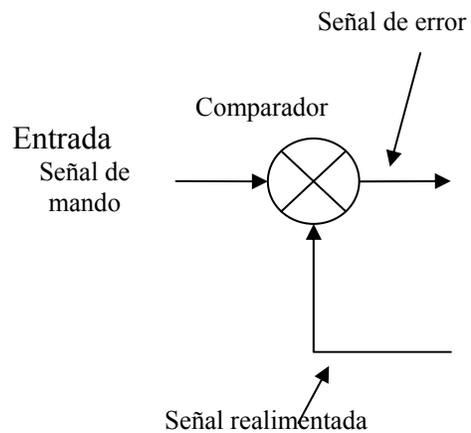
Motores eléctricos. Transforman la energía eléctrica en mecánica. Se destacan los motores paso a paso utilizados en los sistemas de gran precisión. Estos motores transforman un impulso eléctrico en un desplazamiento angular proporcional.

Cilindros con pistón (Pistón). Ampliamente utilizados en neumática e hidráulica, proporcionan un desplazamiento lineal en función de una presión y de un fluido determinado.

1.2.3 Comparadores

También denominados detectores de error. Son elementos del sistema de control encargados de proporcionar una señal en función de la diferencia existente entre señales de entrada y la señal realimentada.

Su símbolo es:



2. TIPOS DE SEÑAL

De manera directa o indirecta siempre estamos en contacto con las señales, permitiendo la comunicación entre las personas y las máquinas.

2.1 CLASIFICACIÓN DE LAS SEÑALES

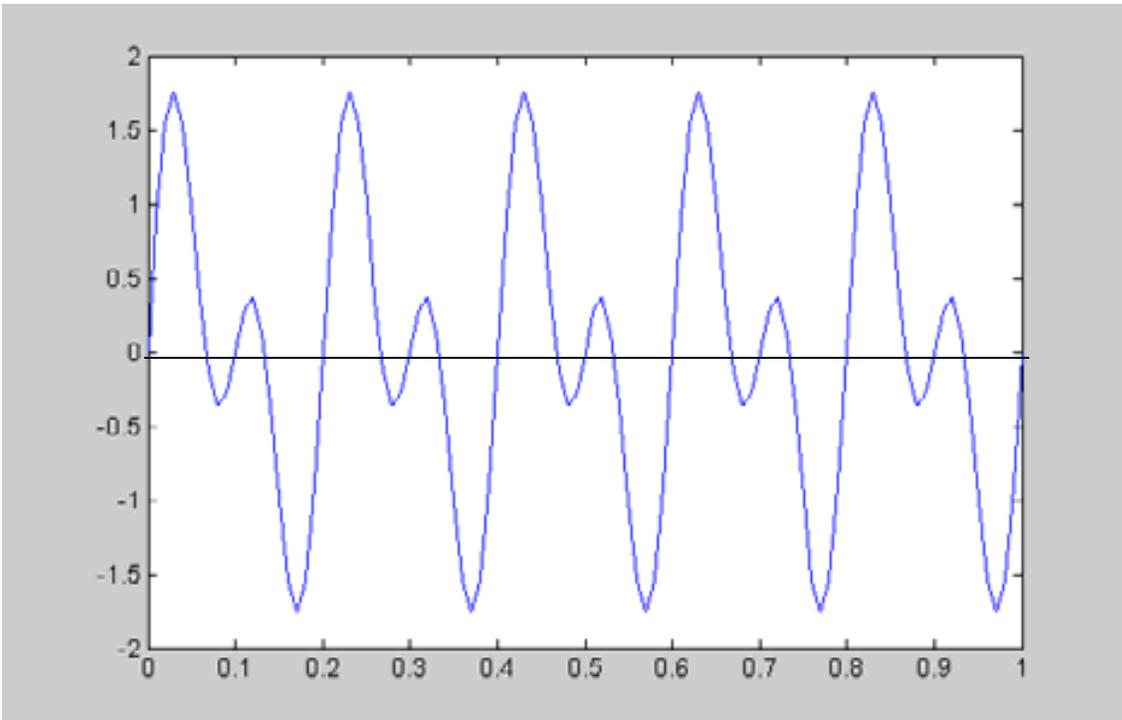
Las señales describen una gran cantidad de fenómenos físicos y su representación también es variada. Las podemos encontrar en forma de datos, en forma de gráficas o mediante una representación matemática en función de una o más variables independientes.

Una clasificación de señales, en términos generales es:

- De tiempo continuo y de tiempo discreto
- Par e impar
- Periódicas y no periódicas
- Determinísticas y aleatorias

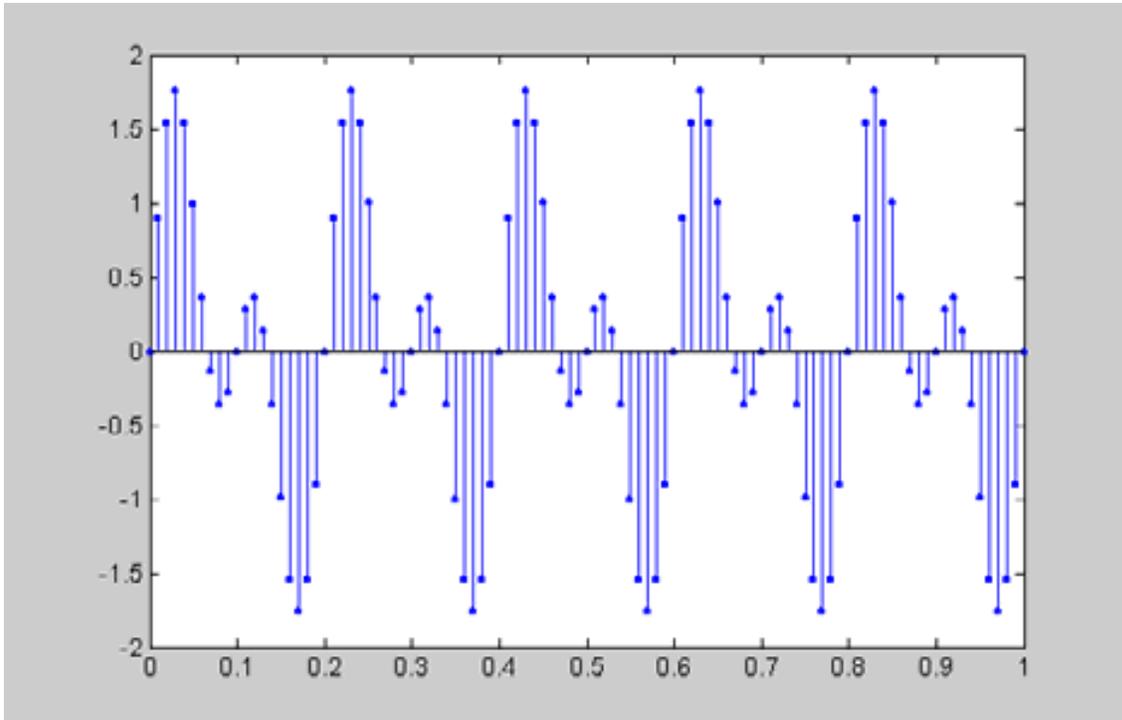
2.1.1 Señales de tiempo continuo y de tiempo discreto.

Una es de tiempo continuo si se define para cada instante de tiempo en un intervalo específico de tiempo. La amplitud de las señales de tiempo continuo varían constantemente con el tiempo y tienen normalmente una notación de la forma $x(t)$.

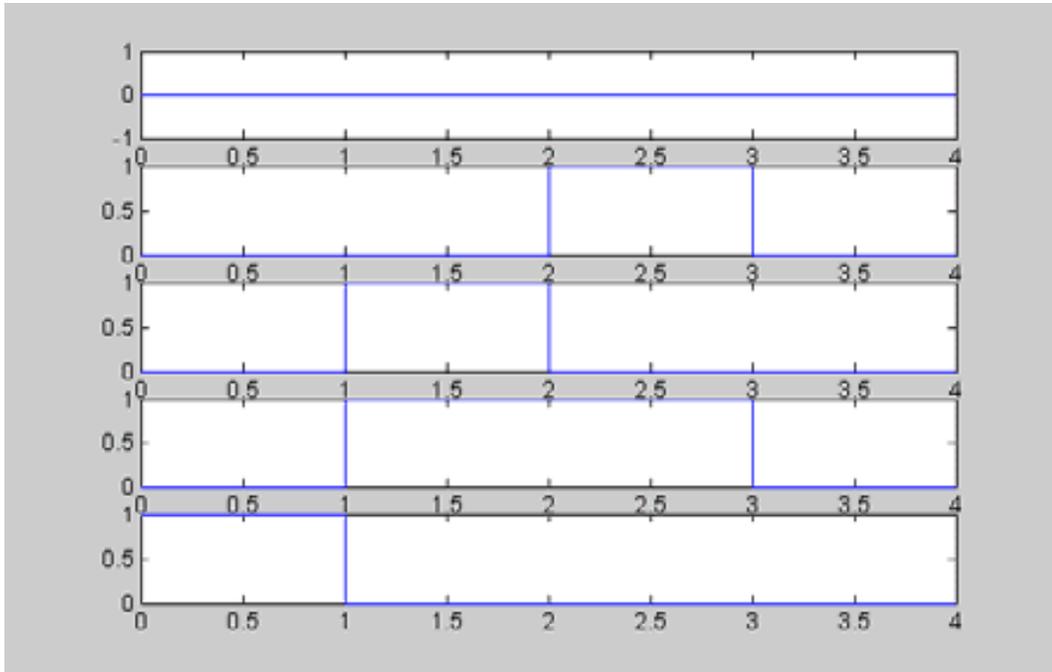


Por otra parte las señales de tiempo discreto son continuas en amplitud y discretas en tiempo, por lo que la variable independiente toma únicamente valores discretos. Es común

obtener una señal discreta mediante el muestreo de una señal continua. Las señales de tiempo discreto se denotan normalmente como $x(n)$ y de manera más específica como $x(nT)$ donde T corresponde a un intervalo o periodo de muestreo donde n denota un entero que toma valores positivos o negativos.

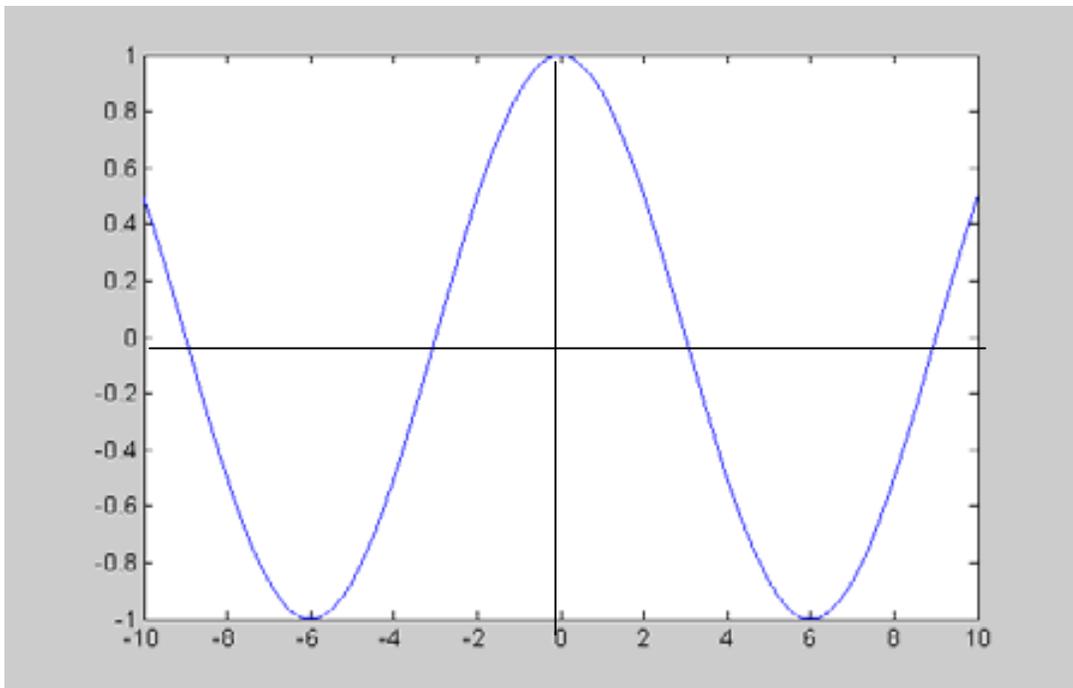


Las señales digitales también son discretas, la diferencia básica consiste en que su magnitud toma valores discretos, por lo que las señales se caracterizan por ser discretas en amplitud y en tiempo.

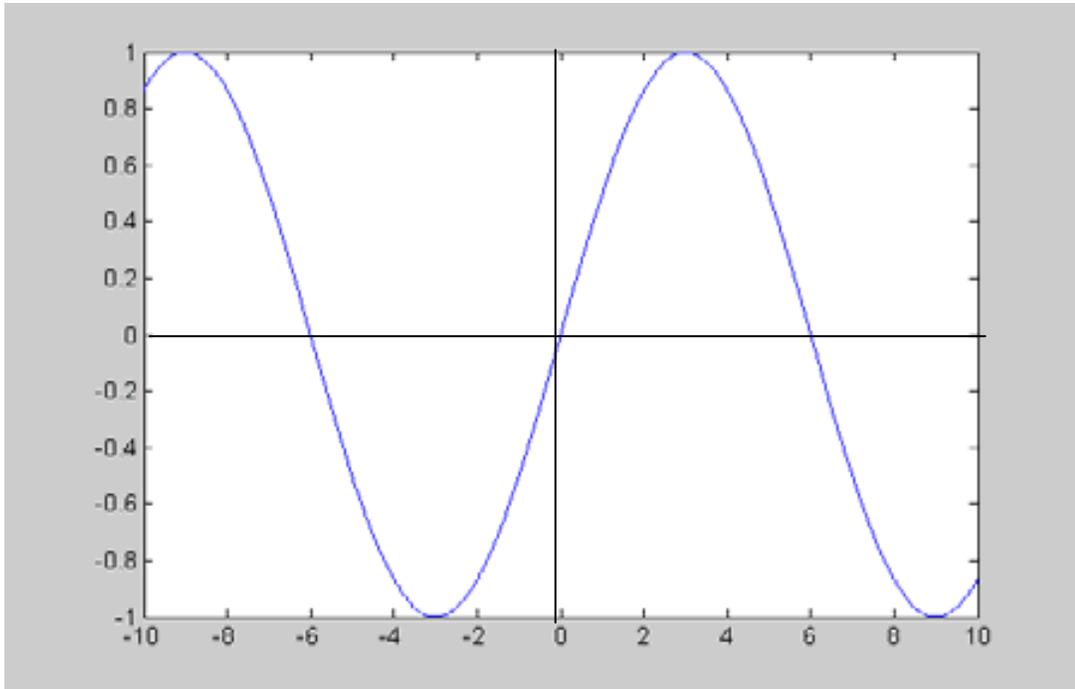


2.1.2 Señal par e impar

Una señal es impar si para toda t o para toda n es idéntica a su reflexión, inversión o transposición en el tiempo, con respecto al eje vertical en el origen.

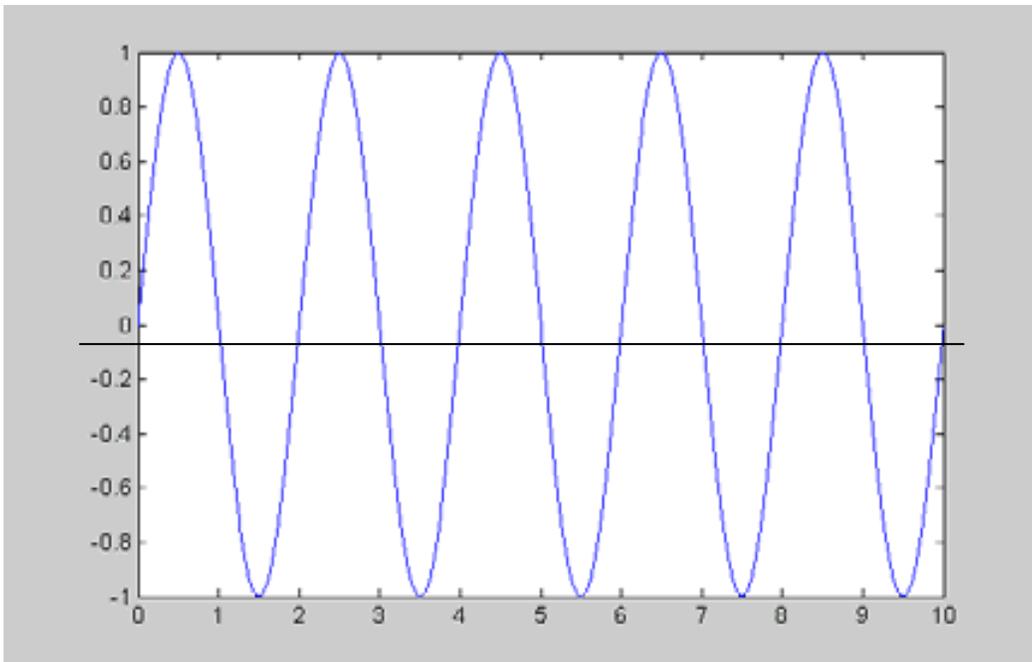


De lo contrario, la señal se considera impar.



2.1.3 Señales periódicas y no periódicas

Una señal es periódica si, para toda t o n , tiene un punto donde la señal comienza a repetirse. Esta repetición de la señal es cada cierto tiempo, por lo que se establece que la señal tienen un periodo normalmente denominado T , y cada periodo representa un ciclo completo de la señal. El recíproco del periodo es la frecuencia fundamental de la señal f_s y ésta define que tan frecuentemente se repite la señal en un intervalo de tiempo, su unidad es el hertz (Hz) o ciclos por segundo.



De no presentar un patrón similar en ningún punto del tiempo, la señal se considera aperiódica.

2.1.4 Señales determinísticas y aleatorias

Una señal determinística es aquella en la que se conoce su valor en cualquier tiempo, en este sentido, se hace referencia a señales que se modelan mediante una representación matemática en función de una o más variables independientes, o bien mediante una función conocida en términos del número de muestras.

Las señales aleatorias son aquellas en las que se desconoce el valor que representará a la señal en cualquier instante de tiempo.

3. CONVERTIDORES DE SEÑAL

3.1 CONVERTIDOR ANALÓGICO-DIGITAL (CAD)

La salida de los sensores, que permiten al equipo electrónico interactuar con el entorno, es normalmente una señal analógica, continua en el tiempo. En consecuencia, esta información debe convertirse en una binaria (cada dato analógico decimal codificado a una palabra formada por unos y ceros) con el fin de adaptarla a los circuitos procesadores y de presentación. Un convertidor analógico-digital (CAD) es un circuito electrónico integrado cuya salida es la palabra digital resultado de convertir la señal analógica de entrada.

La conversión a digital se realiza en dos fases: cuantificación y codificación. Durante la primera se muestrea la entrada y a cada valor analógico obtenido se asigna un valor o estado que depende del número de bits del CAD. El valor cuantificado se codifica en binario en una palabra digital, cuyo número de bits depende de la línea de salida del CAD. Estos dos procesos determinan el diseño del circuito integrado.

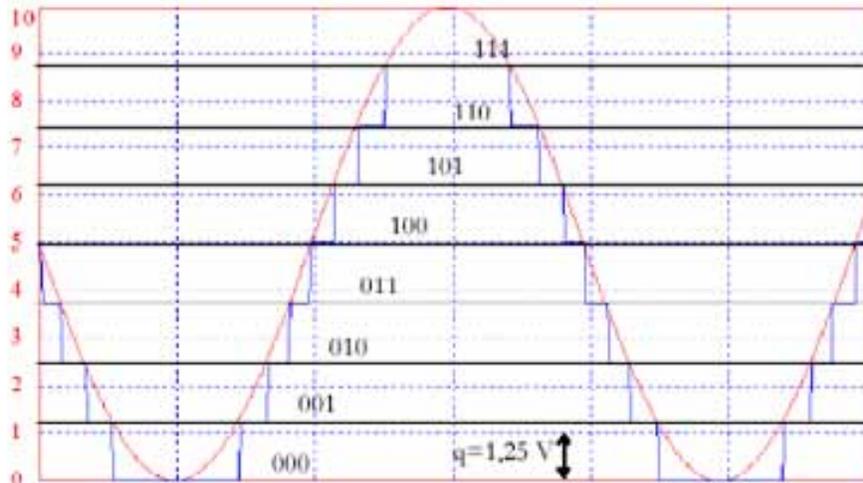
En la práctica, el proceso de conversión está sujeto a numerosas limitaciones resultado de los procesos de fabricación. Las más relevantes son el tiempo de conversión y la finitud del número de estados de salida. La conversión involucra un tiempo y, en consecuencia, supone una incertidumbre que limita la velocidad máxima de la entrada. Los valores discretos del proceso de cuantificación llevan consigo un error y una limitación de resolución del circuito. La elección del CAD en un diseño electrónico dependerá de la adaptación de sus rasgos a los requerimientos de la aplicación.

3.1.1 Principios operativos de los CADs.

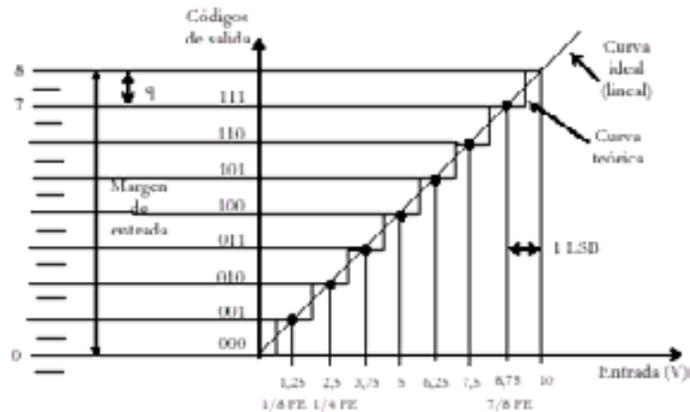
En un CAD de N bits hay 2^N estados de salida y una resolución (porción más pequeña de señal que produce un cambio apreciable en la salida) de $1/2^N$. Con frecuencia la resolución se expresa a partir del margen de entrada del convertidor para definir el intervalo de cuantización o espacio de 1 LSB (Least Significant Bit; bit menos significativo).

$$1LSB = q = \frac{\text{Margen}}{2^N}$$

Aquí la representación de A/D de 3 bits a una entrada analógica senoidal de valor medio 5.



El CAD es un dispositivo no lineal, por lo que no tiene sentido la consideración de función transferencia. Su relación entrada-salida viene dada por una característica escalonada.



Los puntos de decisión pueden considerarse en los extremos o en los centros de cada intervalo de cuantificación. Por ejemplo, en el caso de los 8 estados anteriores, si las transiciones se dan en los extremos, la última se daría en el estado 7, que correspondería a una tensión de entrada de $7 \times 1.25 = 8.75$. Esto es igual que hacer $10 - 1.25$ que resulta a su vez de la generalidad:

$$\text{Margen} - q = \text{margen} - \frac{\text{Margen}}{2^N} = \text{Margen} \left(1 - \frac{1}{2^N} \right)$$

Para este mismo caso de 8 estados, si las transiciones se dan en los puntos centrales de los intervalos, entonces, la última transición se da en el punto de decisión correspondiente al estado $7 + q/2 = 7 \times 1.25 + 0.125 = 8.875$. En general, se da en el punto:

$$\text{Margen} - q/2 = \text{margen} - \frac{\text{Margen}}{2^{N+1}} = \text{Margen} \left(1 - \frac{1}{2^{N+1}} \right)$$

3.1.2 Muestreo de señales analógicas. Teorema de Shannon.

Al muestrear una señal de entrada, el CAD almacena su valor analógico en instante de tiempo fijo y con la misma distancia (periodo de muestreo) determinado por el circuito de muestreo y retención (Sample and Hold, S&H circuit). Si la información que porta la señal no experimenta cambio brusco se puede muestrear a frecuencia baja sin temor a perder información crucial de la señal. Sin embargo, si la señal de interés fluctúa con velocidad, una velocidad de muestreo baja conlleva pérdida de información cuando se trata de reproducir la señal original a partir de las muestras (cuantificación). En estos casos es necesario muestrear con mayor velocidad para asegurar la reproducción fiel de la señal capturada.

3.2 CONVERTIDOR DIGITAL-ANALÓGICO (CDA)

En el mundo real las señales analógicas varían constantemente, pueden variar lentamente como la temperatura o muy rápidamente como una señal de audio. Lo que genera que las señales analógicas sean un tanto difíciles de guardar debido al gran número de valores contenidos en la señal.

Una solución es convertir las señal analógica en una señal digital y guardar esta información binaria que ocupa un menor espacio, y para la posterior recuperación de la señal analógica se utiliza un convertidor Digital-Analógico (DAC).

Un conversor digital analógico o DAC (Digital to Analogue Converter) es un dispositivo para convertir datos digitales en señales de corriente o de tensión analógica.

Se utilizan profusamente en los reproductores de discos compactos, en los reproductores y procesadores de sonido y de cintas de vídeo digitales.

Un DAC contiene normalmente una red resistiva divisora de tensión que tiene una tensión de referencia estable y fija como entrada. Las resistencias varían en proporciones definidas y el flujo de corriente de cada uno está directamente relacionado con el valor binario del bit recibido. Hay que definir que tan exacta será la conversión entre la señal analógica y la digital, para lo cual se define la resolución que tendrá.

Éste es un DAC de 4 bits, cada entrada digital puede ser sólo un "0" o un "1". Donde 0 es el bit menos significativo (LSB) y 3 es el más significativo (MSB). El voltaje de salida analógica tendrá uno de 16 posibles valores dados por una de las 16 combinaciones de la entrada digital.

La resolución del DAC se define de dos maneras. Primera, el número máximo de bits de salida (la salida digital). Este dato permite determinar el número máximo de combinaciones en la salida digital. Este número máximo está dado por:

$$\text{Combinaciones} = 2^n$$

Donde n es el número de bits.

También la resolución se entiende como el voltaje necesario (señal analógica) para lograr que en la salida (señal digital) haya un cambio del bit menos significativo.(LSB). Para hallar la resolución se utiliza la siguiente fórmula:

$$Resolución = \frac{VoFS}{2^n - 1}$$

Donde n es igual al número de bits del DAC y VoFS (full screen) es el voltaje que hay que poner a la entrada del convertidor para obtener una conversión máxima (todas las salidas son "1")

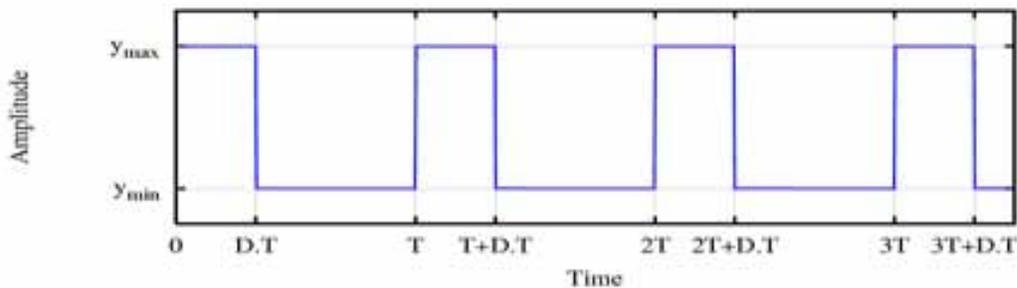
Se puede ver que mientras más bits tenga el convertidor más exacta será la conversión. Por ejemplo si se tienen diferentes tipos de DAC y todos ellos pueden tener una salida máxima de 15 volts, se puede ver que la resolución y exactitud de la salida analógica es mayor cuando más bits tenga.

No de bits del DAC	Resolución
4 bits	15 V / 15 = 1 V
8 bits	15 V / 255 = 58.8 mV
16 bits	15 V / 65536 = 0.23 mV
32 bits	15 V / 4294967296 = 0.0000035 mV

3.3 EL PWM

La modulación de ancho de pulso (PWM) de una señal o fuente de energía implica la modulación de su ciclo de trabajo, ya sea para transmitir información a través de un canal de comunicaciones o control de la cantidad de energía enviada a una carga.

El PWM utiliza una onda cuadrada cuyo ciclo de trabajo es modulado, resultando una variación en el valor medio de la forma de onda. Si consideramos que una forma de onda cuadrada $f(t)$, con un valor bajo $Y_{mín}$, un alto valor $Y_{máx}$ un ciclo de D, el valor medio de la forma de onda está dada por:



$$\bar{y} = \frac{1}{T} \int_0^T f(t) dt$$

Como $f(t)$ es una onda cuadrada, y su valor es $Y_{m\acute{a}x}$ $0 < t < D \cdot T$, y de $Y_{m\acute{i}n}$ para $D \cdot T < t < T$. La expresi3n anterior se convierte en:

$$\begin{aligned} \bar{y} &= \frac{1}{T} \left(\int_0^{DT} Y_{m\acute{a}x} dt + \int_{DT}^T Y_{m\acute{i}n} dt \right) \\ \bar{y} &= \frac{D \cdot T \cdot Y_{m\acute{a}x} + T(1 - D)Y_{m\acute{i}n}}{T} \\ \bar{y} &= D \cdot Y_{m\acute{a}x} + (1 - D)Y_{m\acute{i}n} \end{aligned}$$

Cuando $Y_{m\acute{i}n} = 0$:

$$\bar{y} = D \cdot Y_{m\acute{a}x}$$

A partir de esto, es evidente que el valor medio de la se\~{n}al depende del ciclo de trabajo D .

La construcci3n t\~{i}pica de un circuito PWM se lleva a cabo mediante un comparador con dos entradas y una salida. Una de las entradas se conecta a un oscilador de onda triangular, mientras que la otra queda disponible para la se\~{n}al moduladora. En la salida la frecuencia es generalmente igual a la de la se\~{n}al triangular y el ciclo de trabajo est\~{a} en funci3n de la portadora.

La principal desventaja que presentan los circuitos PWM es la posibilidad de que haya interferencias generadas por radiofrecuencia. Estas pueden minimizarse ubicando el controlador cerca de la carga y realizando un filtrado de la fuente de alimentaci3n.

3.3.1 Par\~{a}metros importantes

Algunos par\~{a}metros importantes de un PWM son:

- La relaci3n de amplitudes entre la se\~{n}al portadora y la moduladora, siendo recomendable que la \u00faltima no supere el valor pico de la portadora y est\~{e} centrada en el valor medio de esta.
- La relaci3n de frecuencias, donde en general se recomienda que la relaci3n entre la frecuencia de la portadora y la de se\~{n}al sea de 10 a 1.

3.3.2 Aplicaciones

Diagrama de ejemplo de la utilizaci3n de la *Modulaci3n de Ancho de Pulso* (PWM) en un variador de frecuencia

En la actualidad, existen muchos circuitos integrados que integran la funci3n PWM, adem\~{a}s de otros muy particulares para lograr circuitos funcionales que puedan controlar

fuentes conmutadas, controles de motores, controles de elementos termoelectrónicos, *choppers* para sensores en ambientes ruidosos y algunas otras aplicaciones. Se distinguen por fabricar este tipo de integrados compañías como Texas Instruments, National Semiconductor, Maxim y algunas otras más.

3.3.2.1 En relación con los motores

La **Modulación por ancho de pulsos** (PWM o *Pulse Width Modulation*) es una técnica utilizada para regular la velocidad de giro de los motores eléctricos. Mantiene el par motor constante y no supone un desaprovechamiento de la energía eléctrica. Se utiliza tanto en corriente continua como en alterna, como su nombre lo indica, al controlar: un momento alto (encendido o alimentado) y un momento bajo (apagado o desconectado), controlado normalmente por relevadores (baja frecuencia) o MOSFET o tiristores (alta frecuencia).

Otros sistemas para regular la velocidad modifican la tensión eléctrica, con lo que disminuye el par motor; o interponen una resistencia eléctrica, con lo que se pierde energía en forma de calor en esta resistencia.

Otra forma de regular el giro del motor es variando el tiempo entre pulsos de modulación por frecuencia de pulsos de duración constante.

En los motores de corriente alterna también se puede utilizar la variación de frecuencia. La modulación por ancho de pulsos también se usa para controlar servo motores, los cuales modifican su posición de acuerdo con el ancho del pulso enviado cada un cierto período que depende de cada servo motor. Esta información puede ser enviada utilizando un microprocesador como el Z80, o un microcontrolador (por ejemplo, un PIC 16F877A de la empresa Microchip).

3.3.2.2 Como parte de un convertidor CAD

Otra aplicación es enviar información de manera analógica. Es útil para comunicarse de forma analógica con sistemas digitales.

Para un sistema digital es relativamente fácil medir cuanto dura una onda cuadrada. Sin embargo, si no se tiene un conversor analógico digital no se puede obtener información de un valor analógico, ya que sólo se puede detectar si hay una determinada tensión, 0 ó 5 voltios por ejemplo (valores digitales de 0 y 1), con una cierta tolerancia, pero no puede medirse un valor analógico. Sin embargo, el **PWM** en conjunción con un oscilador digital, un contador y una puerta **AND** como puerta de paso, podrían fácilmente implementar un CAD

3.3.2.3 Regulación de tensión

PWM se utiliza también en el eficiente regulador de voltaje. Por el cambio de voltaje a la carga con el ciclo de trabajo adecuado, la salida será una tensión aproximada al nivel deseado. El ruido de conmutación suele ser filtrado con un inductor y un capacitor.

Un método de medida de la tensión de salida. Cuando es inferior a la tensión deseada, se enciende el interruptor. Cuando el voltaje de salida está por encima de la tensión deseada, se apaga el interruptor.

3.3.2.4 Potencia de entrega

PWM se puede utilizar para reducir la cantidad total de la energía suministrada a una carga normalmente sin pérdidas sufridas cuando una fuente de alimentación está limitada por los medios de resistencia. Esto se debe a que la potencia media entregada es proporcional a la modulación del ciclo. Con una tasa suficientemente alta de modulación y filtros electrónicos pasivos se puede suavizar el tren de pulsos y recuperar una forma de onda media analógica.

PWM de alta frecuencia para los sistemas de control de potencia son fácilmente realizables con interruptores semiconductores. El discreto encendido / apagado de los estados de la modulación se utilizan para controlar el estado del interruptor (es) que a su vez son el control de voltaje o corriente a través de toda la carga. La principal ventaja de este sistema son los conmutadores que no representan caída de tensión alguna a través de ellos. El producto de la corriente y la tensión en un momento dado define la potencia disipada por el cambio, por lo tanto no hay potencia disipada por el cambio. Siendo realistas, interruptores semiconductores como MOSFETs o BJTs no son interruptores ideales, pero los controladores de alta eficiencia sí pueden ser construidos.

PWM también es usado a menudo para controlar el suministro de energía eléctrica a otro dispositivo, como en el control de velocidad de motores eléctricos, control de volumen de los amplificadores de clase D de audio o el control de brillo de las fuentes de luz y muchas otras aplicaciones de la electrónica de potencia. Por ejemplo, los reguladores de luz para uso doméstico emplean un tipo específico de control PWM

4. MICROCONTROLADORES

4.1 QUE ES UN MICROCONTROLADOR

Circuito integrado o chip que incluye en su interior las tres unidades funcionales de un ordenador: CPU, Memoria y Unidades de E/S; es decir, se trata de un computador completo en un solo circuito integrado. Aunque sus prestaciones son limitadas, además de dicha integración, su característica principal es su alto nivel de especialización. Aunque los hay del tamaño de un sello de correos, lo normal es que sean incluso más pequeños, ya que, lógicamente, forman parte del dispositivo que controlan.

Un **microcontrolador** es un microprocesador optimo para ser utilizado para controlar equipos electrónicos. Los microcontroladores representan la inmensa mayoría de los chips de ordenadores vendidos, sobre un 50% son controladores "simples" y el restante corresponde a DSPs más especializados.

Un microcontrolador difiere de una CPU normal, debido a que es más fácil convertirla en un ordenador en funcionamiento, con un mínimo de chips externos de apoyo. La idea es que el chip se coloque en el dispositivo, enganchado a la fuente de energía y de información que necesite, y eso es todo. Un microprocesador tradicional no le permitirá hacer esto, ya que espera que todas estas tareas sean manejadas por otros chips.

Por ejemplo, un microcontrolador típico tendrá un generador de reloj integrado y una pequeña cantidad de memoria RAM y ROM/ EPROM/ EEPROM, significando que para hacerlo funcionar, todo lo que se necesita son unos pocos programas de control y un cristal de sincronización. Los microcontroladores disponen generalmente también de una gran variedad de dispositivos de entrada/salida, como convertidores de analógico a digital, temporizadores, UARTs y buses de interfaz serie especializados, como I²C y CAN. Frecuentemente, estos dispositivos integrados pueden ser controlados por instrucciones de procesadores especializados. Los modernos microcontroladores frecuentemente incluyen un lenguaje de programación integrado, como el BASIC que se utiliza bastante con este propósito.

Los microcontroladores negocian la velocidad y la flexibilidad para facilitar su uso. Debido a que se utiliza bastante sitio en el chip para incluir funcionalidad, como los dispositivos de entrada/salida o la memoria que incluye el microcontrolador, se ha de prescindir de cualquier otra circuitería.

Los microcontroladores más comunes en uso son:

- Atmel
 - AVR
- Hitachi
 - H8
- Holtek
 - HT8
- Intel
 - 8 bits

- 8XC42
 - MCS51
 - 8xC251
 - 16 bits
 - MCS96
 - MXS296
- National Semiconductor
 - COP8
- Microchip
 - Gama baja (Familia 12Cxx de 12 bits) (p.e. PIC12C508)
 - Gama media (Familia 12Fxx, 16Cxx y 16Fxx de 14 bits)(p.e. PIC16F84)
 - Gama alta (18Cxx y 18Fxx de 16 bits) (p.e. PIC18F452)
 - dsPIC (DSPs)
- Motorola
 - 8 bits
 - 68HC05
 - 68HC08
 - 68HC11
 - 16 bits
 - 68HC12
 - 68HC16
 - 32 bits
 - 683xx
- NEC
 - 78K
 - ST
 - ST 62
 - ST 7
- Texas Instruments
 - TMS370
- Zilog
 - Z8
 - Z86E02
- Genérico
 - Algunas arquitecturas de microcontrolador están disponibles por tal cantidad de vendedores y en tantas variedades que podrían tener, con total corrección, su propia categoría. Entre ellos encontramos, principalmente, las variantes de 8051 y Z80.

4.2 EL PIC

Los **PIC** son una familia de microcontroladores tipo RISC fabricados por Microchip Technology Inc. y derivados del PIC1650, originalmente desarrollado por la división de microelectrónica de General Instruments.

El nombre actual no es un acrónimo. En realidad, el nombre completo es **PICmicro**, aunque generalmente se utiliza como *Peripheral Interface Controller* (Controlador de Interfaz Periférico).

El PIC original se diseñó para ser usado con la nueva UCP de 16 bits CP16000. Siendo en general una buena UCP, ésta tenía malas prestaciones de E/S, y el PIC de 8 bits se desarrolló en 1975 para mejorar el rendimiento del sistema quitando peso de E/S a la UCP. El PIC utilizaba microcódigo simple almacenado en ROM para realizar estas tareas; y aunque el término no se usaba por aquel entonces, se trata de un diseño RISC que ejecuta una instrucción cada 4 ciclos del oscilador.

En 1985, dicha división de microelectrónica de General Instruments se convirtió en una filial y el nuevo propietario canceló casi todos los desarrollos que para esas fechas la mayoría estaban obsoletos. El PIC, sin embargo, se mejoró con EPROM para conseguir un controlador de canal programable. Hoy en día multitud de PICs vienen con varios periféricos incluidos (módulos de comunicación serie, UARTs, núcleos de control de motores, etc.) y con memoria de programa desde 512 a 32.000 palabras (una *palabra* corresponde a una instrucción en ensamblador, y puede ser 12, 14 ó 16 bits, dependiendo de la familia específica de PICmicro).

4.2.1 Juego de instrucciones y entorno de programación

El PIC usa un juego de instrucciones tipo RISC, cuyo número puede variar desde 35 para PICs de gama baja a 70 para los de gama alta. Las instrucciones se clasifican entre las que realizan operaciones entre el acumulador y una constante, entre el acumulador y una posición de memoria, instrucciones de condicionamiento y de salto/retorno, implementación de interrupciones y una para pasar a modo de bajo consumo llamada *sleep*.

Microchip proporciona un entorno de desarrollo freeware llamado MPLAB que incluye un simulador software y un ensamblador. Otras empresas desarrollan compiladores C y BASIC. Microchip también vende compiladores para los PICs de gama alta ("C18" para la serie F18 y "C30" para los dsPICs) y se puede descargar una edición para estudiantes del C18 que inhabilita algunas opciones después de un tiempo de evaluación.

Para Pascal existe un compilador de código abierto, JAL, lo mismo que PicForth para el lenguaje Forth. GPUTILS es una colección de herramientas distribuidas bajo licencia GNU que incluye ensamblador y enlazador, y funciona en Linux, MacOS y Microsoft Windows. GPSIM es otra herramienta libre que permite simular diversos dispositivos hardware conectados al PIC.

4.2.1.1 Tamaño de palabra

El tamaño de palabra de los microcontroladores PIC es fuente de muchas confusiones. Todos los PICs (excepto los dsPIC) manejan datos en trozos de 8 bits, con lo que se deberían llamar microcontroladores de 8 bits. Pero a diferencia de la mayoría de UCPs, el PIC usa arquitectura Harvard, por lo que el tamaño de las instrucciones puede ser distinto del de la palabra de datos. De hecho, las diferentes familias de PICs usan tamaños de

instrucción distintos, lo que hace difícil comparar el tamaño del código del PIC con el de otros microcontroladores. Por ejemplo, pongamos que un microcontrolador tiene 6144 bytes de memoria de programa: para un PIC de 12 bits esto significa 4096 palabras y para uno de 16 bits, 3072 palabras.

4.3 Características

Los PICs actuales vienen con una amplia gama de mejoras hardware incorporadas:

- Núcleos de UCP de 8/16 bits con Arquitectura Harvard modificada
- Memoria Flash y ROM disponible desde 256 bytes a 256 kilobytes
- Puertos de E/S (típicamente 0 a 5,5 volts)
- Temporizadores de 8/16 bits
- Tecnología Nanowatt para modos de control de energía
- Periféricos serie síncronos y asíncronos: USART, AUSART, EUSART
- Conversores analógico/digital de 10-12 bits
- Comparadores de tensión
- Módulos de captura y comparación PWM
- Controladores LCD
- Periférico MSSP para comunicaciones I2C, SPI, y I2S
- Memoria EEPROM interna con duración de hasta un millón de ciclos de lectura/escritura
- Periféricos de control de motores
- Soporte de interfaz USB
- Soporte de controlador Ethernet
- Soporte de controlador CAN
- Soporte de controlador LIN

Soporte de controlador Irda

4.3.1 EL Procesador o CPU.

El procesador responde a la arquitectura RISC donde la mayoría de instrucciones se ejecutan en un ciclo de reloj, excepto las instrucciones de salto que necesitan dos ciclos. La ALU (Unidad Aritmético Lógica) ubicada dentro del procesador realiza las operaciones lógicas y aritméticas con dos operandos, uno que recibe desde el registro W (registro de trabajo) y otro que puede provenir de cualquier registro interno.

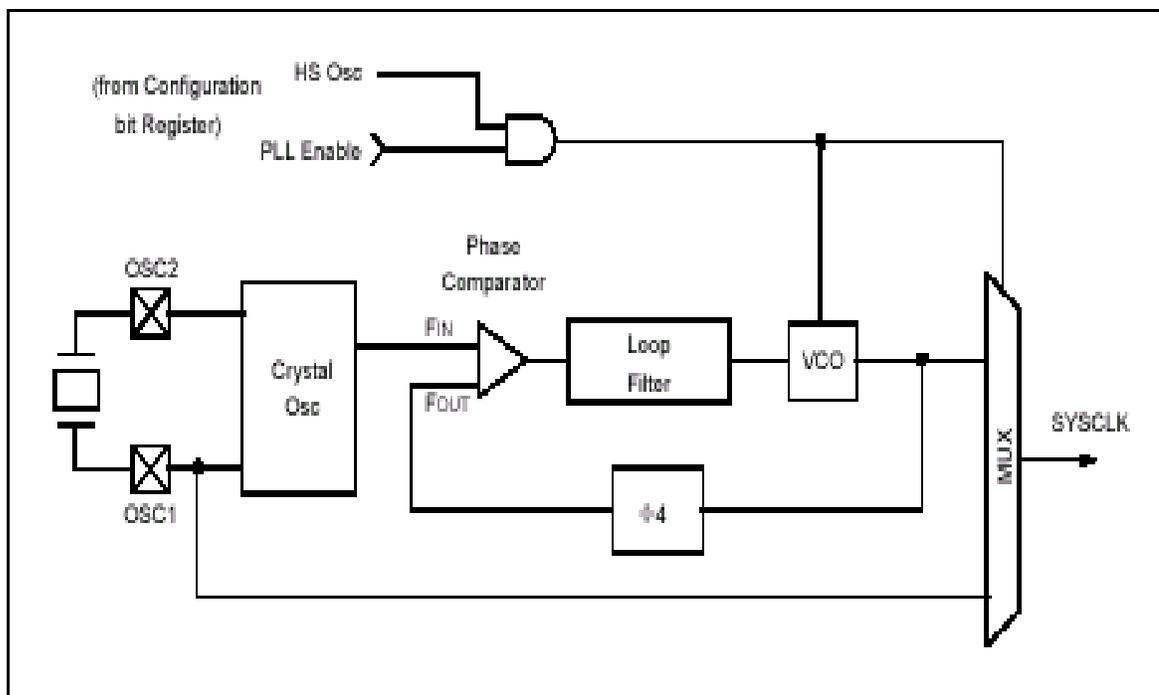
4.3.2 Frecuencia

La frecuencia de reloj principal determina el ciclo de instrucción (tiempo que tarda una instrucción en ejecutarse). En los PICs, un ciclo de instrucción emplea cuatro periodos de oscilación del reloj principal. Por ejemplo si la frecuencia del reloj principal es de 4 MHz, un ciclo de instrucción tardara en realizarse:

$$T_{\text{oscilación del reloj principal}} = 1/F_{\text{del reloj principal}} = 1/4\text{MHz} = 250\text{ns}$$
$$\text{Ciclo de instrucción} = T_{\text{oscilación}} \times 4 = 250\text{ns} \times 4 = 1\mu\text{s}$$

Según el PIC y los componentes externos que se coloquen se configuran los osciladores:

- Tipo RC. Es un oscilador de bajo costo y poca estabilidad. Solo precisa una resistencia y un capacitor externos.
- Tipo HS. Es un oscilador de alta velocidad y muy estable funciona en frecuencias comprendidas entre 4 y 20MHz. Utiliza un cristal de cuarzo o un resonador cerámico.
- Tipo XT. También emplea el cristal de cuarzo o el resonador cerámico. Trabaja en frecuencias medias, comprendidas entre 100KHz y 4MHz.
- Tipo LP. Empleado en aplicaciones de bajo consumo, lo que conlleva una frecuencia pequeña. A más velocidad, más consumo. Usa cristal o resonador, y las frecuencias de trabajo oscilan entre 35 y 200KHz.
- Tipo HS + PLL. El cristal de 5Mhz es conectado en los pines OSC1 y OSC2 RC, habilitando un PLL . Es un oscilador de bajo costo y poca estabilidad. Solo precisa una resistencia y un capacitor externos.
- Tipo EC. Utiliza un reloj externo que es conectado al pin OSC1
- Tipo ECIO. Reloj externo con pin de entrada-salida habilitado. El pin OSC2 es habilitado como entrada_salida de propósito general.



Estado de reposo: cuando el microcontrolador funciona en modo reposo (sleep) la corriente necesaria es menor de 3mA.

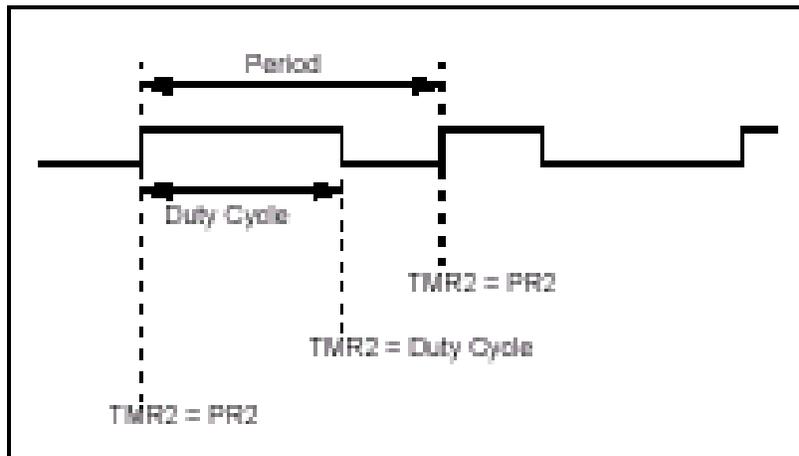
Perro guardián (Watchdog). El temporizador perro guardián (watchdog) es independiente del reloj principal (posee su propio oscilador), por lo tanto en el modo en bajo consumo puede seguir funcionando. Cuando llegue al valor máximo FFh, se produce el desbordamiento del perro_guardián, se iniciará tomando el valor 00h y provocará un reset.

El tiempo típico es de 18 ms, pero utilizando un divisor de frecuencia (programable por software) se pueden alcanzar 2.3 segundos.

La utilización del perro guardián permite controlar los posibles pérdidas de continuidad del programa, esto es si durante el programa hemos previsto poner a cero el perro guardián para evitar que se genere un reset, en el momento que por un fallo no previsto el programa se quede en un bucle sin fin, al no poder poner a cero el perro guardián, éste generará un reset sacando al programa del bucle.

4.3.3 EL PWM

El PIC cuenta con módulos PWM de 10 bits. En este modo de trabajo se consiguen pulsos cuyo ancho en el nivel alto es de duración variable y que son enormemente útiles en el control de los motores. El pin RC2/CCP1 está configurado como salida y la oscilación entre los niveles lógicos 0 y 1 a intervalos variables de tiempo. Se intenta obtener un pulso cuyo nivel alto tenga un ancho variable (Duty Cycle), dentro del intervalo fijo del período del pulso.



Para conseguir la oscilación de la pata se usa un comparador que pone a 1 (Set) un flip-flop cuando el valor del registro PR2 coincide con la parte alta del TMR2, momento en que TMR2 toma el valor 00H. Luego el flip-flop se resetea y se pone a 0 cuando otro comparador detecta la coincidencia del valor existente en CCPR1H con el de la parte alta de TMR2. Variando los valores que se cargan en PR2 y en CCPR1L se varía el intervalo de tiempo en que la pata de salida está a 1 y a 0.

El periodo del PWM puede ser calculado usando la fórmula.

$$\text{PeriodoPWM} = [(PR2) + 1] \cdot 4 \cdot T_{osc} \cdot (\text{valor_preescalarTMR2})$$

4.3.4 Conversor A/D:

Los PIC16F87X y PIC18F4439 disponen de un conversor A/D de 10 bits de resolución 5 y 8 canales de entrada respectivamente. A través de una entrada seleccionada se aplica la señal analógica a un capacitor de captura y mantenimiento luego dicho valor se introduce al

convertor que usando la técnica de aproximaciones sucesivas proporciona un resultado digital equivalente de 10 bits. La tensión de referencia puede implementarse con la tensión interna de alimentación del PIC o bien con una externa introducida por las patas RA3/AN3/VREF+ y RA2/AN2/VREF-.

El convertidor A/D tiene cuatro registros:

- Registro de resultado alto (ARDES)
- Registro de resultado bajo (ADRESL)
- Registro de control 0 (ADCON0)
- Registro de control 1 (ADCON1)

El registro ADCON0 controla la operación del módulo A/D. El ADCON1 configura la función de los pins del puerto.

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/DONE	—	ADON
bit 7						bit 0	

bit 7-6 **ADCS1:ADCS0**: A/D Conversion Clock Select bits (ADCON0 bits in **bold**)

ADCON1 <ADCS2>	ADCON0 <ADCS1:ADCS0>	Clock Conversion
0	00	Fosc/2
0	01	Fosc/8
0	10	Fosc/32
0	11	FRC (clock derived from the internal A/D RC oscillator)
1	00	Fosc/4
1	01	Fosc/16
1	10	Fosc/64
1	11	FRC (clock derived from the internal A/D RC oscillator)

bit 5-3 **CHS2:CHS0**: Analog Channel Select bits

- 000 = Channel 0 (AN0)
- 001 = Channel 1 (AN1)
- 010 = Channel 2 (AN2)
- 011 = Channel 3 (AN3)
- 100 = Channel 4 (AN4)
- 101 = Channel 5 (AN5)⁽¹⁾
- 110 = Channel 6 (AN6)⁽¹⁾
- 111 = Channel 7 (AN7)⁽¹⁾

Note 1: These channels are unimplemented on PIC18F2X39 devices. Do not select any unimplemented channel.

- bit 2 **GO/DONE:** A/D Conversion Status bit
When ADON = 1:
 1 = A/D conversion in progress (setting this bit starts the A/D conversion, which is automatically cleared by hardware when the A/D conversion is complete)
 0 = A/D conversion not in progress
- bit 1 **Unimplemented:** Read as '0'
- bit 0 **ADON:** A/D On bit
 1 = A/D converter module is powered up
 0 = A/D converter module is shut-off and consumes no operating current

ADCON0

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/DONE	—	ADON
bit 7						bit 0	

- bit 7-6 **ADCS1:ADCS0:** A/D Conversion Clock Select bits (ADCON0 bits in **bold**)

ADCON1 <ADCS2>	ADCON0 <ADCS1:ADCS0>	Clock Conversion
0	00	F _{osc} /2
0	01	F _{osc} /8
0	10	F _{osc} /32
0	11	FRC (clock derived from the internal A/D RC oscillator)
1	00	F _{osc} /4
1	01	F _{osc} /16
1	10	F _{osc} /64
1	11	FRC (clock derived from the internal A/D RC oscillator)

- bit 5-3 **CHS2:CHS0:** Analog Channel Select bits

000 = Channel 0 (AN0)
 001 = Channel 1 (AN1)
 010 = Channel 2 (AN2)
 011 = Channel 3 (AN3)
 100 = Channel 4 (AN4)
 101 = Channel 5 (AN5)⁽¹⁾
 110 = Channel 6 (AN6)⁽¹⁾
 111 = Channel 7 (AN7)⁽¹⁾

Note 1: These channels are unimplemented on PIC18F2X39 devices. Do not select any unimplemented channel.

- bit 2 **GO/DONE:** A/D Conversion Status bit
When ADON = 1:
 1 = A/D conversion in progress (setting this bit starts the A/D conversion, which is automatically cleared by hardware when the A/D conversion is complete)
 0 = A/D conversion not in progress
- bit 1 **Unimplemented:** Read as '0'
- bit 0 **ADON:** A/D On bit
 1 = A/D converter module is powered up
 0 = A/D converter module is shut-off and consumes no operating current

ADCO1

4.3.5 Puerto de comunicación serie síncrono:

Este módulo, llamado MSSP, integrado en los PIC's proporciona un excelente medio de comunicación con otros microcontroladores o periféricos que trabajan en serie.

Tiene dos alternativas de trabajo:

- SPI (Serial Peripheral Interface).
- I²C (Inter-Integrated Circuit).

La comunicación en modo SPI la utilizan principalmente las memorias RAM y EEPROM y utiliza tres líneas. En el modo I²C sólo se emplean dos líneas y se usa en la comunicación de circuitos integrados diversos. Básicamente el módulo MSSP está basado en dos registros: el SSPSR que es un registro de desplazamiento que transforma la información serie en paralelo y viceversa, y el registro SSPBUF que actúa como buffer de la información que se recibe o se transmite en serie. En transmisión, el byte que se desea enviar se carga en el SSPBUF y automáticamente se traspasa a SSPSR, donde se va desplazando bit a bit, sacándolo al exterior al ritmo de los pulsos de reloj. En recepción, los bits van entrando al ritmo del reloj por una pata del PIC y se van desplazando en el SSPSR hasta que lo llenan, en cuyo momento se traspasa la información al SSPEUF.

En el modo SPI se utilizan tres líneas del PIC, mientras que en el modo I²C sólo se emplean dos líneas para la comunicación del PIC maestro con los circuitos integrados que funcionan como esclavos.

4.3.6 USART

El USART soporta la comunicación serie síncrona y asíncrona. Puede funcionar como un sistema de comunicación bidireccional asíncrono o full duplex, adaptándose a multitud de periféricos que transfieren información de esta manera. También puede trabajar en modo unidireccional o halfduplex.

En resumen puede trabajar de tres maneras:

- SERIE ASÍNCRONA (Full duplex, bidireccional).
- SERIE SÍNCRONA-MAESTRO (Halfduplex, unidireccional).
- SERIE SÍNCRONA-ESCLAVO (Halfduplex, unidireccional).

En el primero, las transferencias de información se realizan sobre dos líneas TX y RX, saliendo y entrando los bits por dichas líneas al ritmo de la frecuencia controlada internamente por el USART.

En el modo síncrono la comunicación se realiza sobre dos líneas, la DT que traslada los bits en ambos sentidos a la frecuencia de los pulsos de reloj que salen por la línea CK desde el maestro.

4.3.7 Registro STATUS

R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	TO	PD	Z	DC	C
bit/						bit0	

Este registro contiene algunos bits que representan el estado de la ALU (operaciones aritméticas y lógicas), el estado de Reset y el bit de selección del banco de la memoria de datos.

No utilizar la instrucción CLRF STATUS. Cuando se ejecuta una instrucción que afecta los bits Z, C o DC, la escritura sobre los mismos queda inhabilitada. Si desea forzar a 0 alguno de estos bits utilice la instrucción BCF (BCF junto a BSF, SWAPF y MOVWF son las únicas instrucciones que no afectan el estado del registro STATUS). Descripción de cada bit:

- IRP. Se utiliza para el direccionamiento indirecto (permite seleccionar el banco).
 - IRP=0: Banco 0 y 1 seleccionado.
 - IRP=1: Banco 2 y 3 seleccionado.
 - RP1: RP0: Bits de selección del banco de memoria de datos.
 - RP0=0 y RP1=0: Banco 0 seleccionado.
 - RP0=0 y RP1=1: Banco 1 seleccionado.
 - RP0=1 y RP1=0: Banco 2 seleccionado.
 - RP0=1 y RP1=1: Banco 3 seleccionado.

- TO. Bit de “Time-out”: Este bit se coloca automáticamente en 1 luego de ejecutarse la instrucción CLRWDT, SLEEP o de finalizado el ciclo de Power-up (inicialización). El bit vuelve a 0 automáticamente cuando ocurre un evento de time-out de WDT.

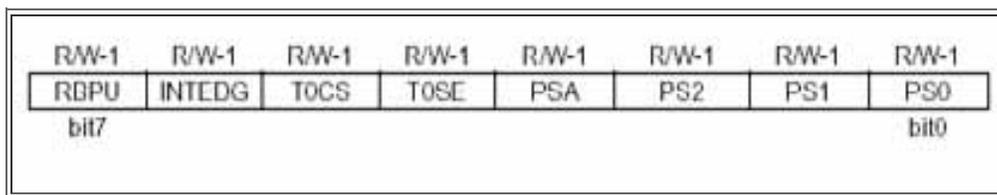
- PD. Bit de “Power-down”: Este bit se coloca automáticamente en 1 luego de ejecutarse la instrucción CLRWDT o de finalizado el ciclo de Power-up (inicialización). El bit vuelve a 0 automáticamente cuando se ejecuta la instrucción SLEEP.

- Z. Bit Cero
 - Z=1. Si el resultado de una operación aritmética o lógica es cero.
 - Z=0. Si el resultado de una operación aritmética o lógica no es cero.

- DC. Bit de acarreo del 4to bit (útil para trabajar con dígitos BCD):
 - DC=1. En una operación de suma, el resultado de la operación entre los 4 bits menos significativos de los sumandos es mayor a 15. El resultado en una resta es igual o mayor a 0.

- DC=0. En una operación de suma, el resultado de la operación entre los 4 bits menos significativos de los sumandos es mayor a 15. El resultado en una resta es menor a 0.
- C. Bit de acarreo
 - C=1. El resultado de una suma es mayor a 255. El resultado de una resta es igual o mayor a 0.
 - C=0. El resultado de una suma es menor o igual a 255. El resultado de una resta es menor a 0.

4.3.8 Registro OPTION



A este registro lo etiquetaremos OPTION_REG. Este cambio se debe a que OPTION es una etiqueta reservada (la instrucción OPTION se utiliza en algunos microcontroladores de la línea). El registro contiene una serie de bits de control relacionados con la configuración de TMR0, la interrupción externa y la habilitación de las resistencias de pull-up del PORTB. Todos los bits del registro se pueden leer o escribir desde el software.

Descripción de cada Bit:

- RBPU. Habilitación de las resistencias de “pull-up” del PORTB. Cuando un terminal se utiliza como entrada, el mismo debe tener un nivel de tensión de VDD (1 lógico) o 0 Vcc (0 lógico). Para fijar la tensión a uno de estos niveles debe conectarse, en cada entrada, una resistencia. En el caso del PORTB, los PIC16F87X tienen incorporadas resistencias de pull-up (fijan la tensión en VDD).
 - RBPU=1. Resistencias de “pull-up” habilitadas.
 - RBPU=0. Resistencias de “pull-up” inhabilitadas.
- Los demás bits se relacionan con las interrupciones.

4.4 Programación del PIC

Para transferir el código de un ordenador al PIC normalmente se usa un dispositivo llamado programador. La mayoría de PICs que Microchip distribuye hoy en día incorporan ICSP (*In Circuit Serial Programming*, programación serie incorporada) o LVP (*Low Voltage Programming*, programación a bajo voltaje) lo que permite programar el PIC directamente en el circuito destino. Para la ICSP se usan los pines RB6 y RB7 como reloj y datos y el MCLR para activar el modo programación aplicando un voltaje de unos 11 volt. Existen muchos programadores de PICs, desde los más simples que dejan al software los detalles de comunicaciones, a los más complejos que pueden verificar el dispositivo a diversas tensiones de alimentación e implementan en hardware casi todas las funcionalidades. Muchos de estos programadores complejos incluyen ellos mismos PICs preprogramados

como interfaz para enviar las órdenes al PIC que se desea programar. Uno de los programadores más simples es el TE20, que utiliza la línea TX del puerto RS232 como alimentación y las líneas DTR y CTR para mandar o recibir datos cuando el microcontrolador está en modo programación. El software de programación puede ser el ICprog, muy común entre la gente que utiliza este tipo de microcontroladores.

Se pueden obtener directamente de Microchip muchos programadores / depuradores (octubre 2005):

4.4.1 Programadores

- PICStart Plus (puerto serie)
- Promate II (puerto serie)
- MPLAB PM3 (puerto serie y USB)
- ICD2 (puerto serie y USB)
- PICKit 1 (USB)

4.5 PICs modernos

Los viejos PICs con memoria PROM o EPROM se están renovando gradualmente por chips con memoria Flash. Así mismo, el juego de instrucciones original de 12 bits del PIC1650 y sus descendientes directos han sido suplantados por juegos de instrucciones de 14 y 16 bits. Microchip todavía vende versiones PROM y EPROM de la mayoría de los PICs para soporte de aplicaciones antiguas o grandes pedidos.

4.5.1 Clones del PIC

Por todos lados surgen compañías que ofrecen versiones del PIC más baratas o mejoradas. La mayoría suelen desaparecer rápidamente. Una de ellas que va perdurando es Unicorn (antiguamente Scenix) que vende clones del PIC que funcionan mucho más rápido que el original. OpenCores tiene un núcleo del PIC16F84 escrito en Verilog.

4.5.2 PICs wireless

El microcontrolador rfPIC integra todas las prestaciones del PICmicro de Microchip con la capacidad de comunicación wireless UHF para aplicaciones RF de baja potencia. Estos dispositivos ofrecen un diseño muy comprimido para ajustarse a los cada vez más demandados requerimientos de miniaturización en aparatos electrónicos.

4.5.3 PICs para procesamiento de señal (dsPICs)

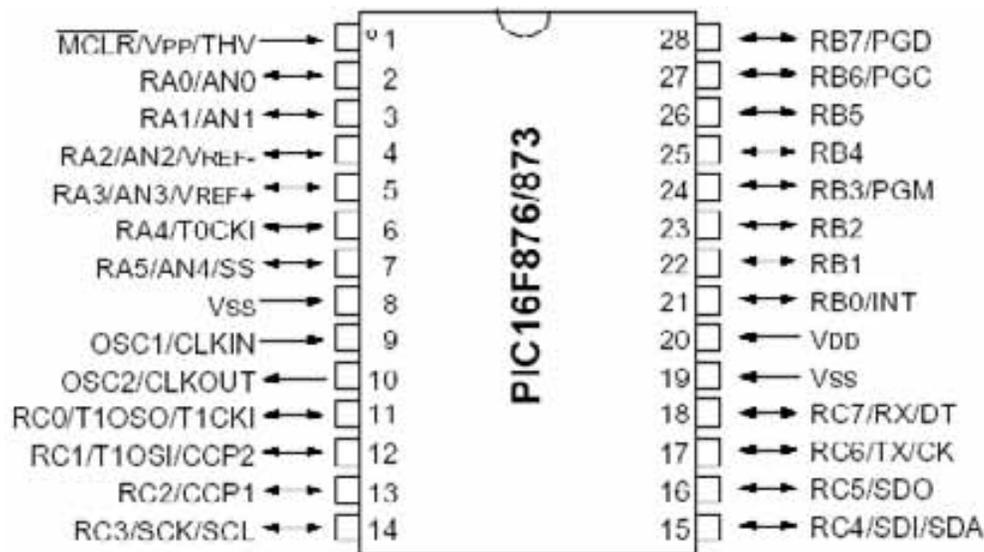
Los dsPICs son el último lanzamiento de Microchip, comenzando a producirlos a gran escala a finales de 2004. Son los primeros PICs con bus de datos inherente de 16 bits. Incorporan todas las posibilidades de los anteriores PICs y añaden varias operaciones de DSP implementadas en hardware, como multiplicación con suma de acumulador (*multiply-accumulate*, o *MAC*), *barrel shifting*, *bit reversion* o multiplicación 16x16 bits.

4.5.4 PICs más comúnmente usados

- PIC12C508/509 (encapsulamiento reducido de 8 pines, oscilador interno, popular en pequeños diseños como el iPod remote)
- PIC16F84 (Considerado obsoleto, pero imposible de descartar y muy popular)
- PIC16F84A (Buena actualización del anterior, algunas versiones funcionan a 20 MHz, compatible 1:1)
- PIC12F629/675
- PIC16F628
- La familia PIC16F87X (los hermanos mayores del PIC16F84, con cantidad de mejoras incluidas en hardware. Bastante común en proyectos de aficionados)
- PIC18F452

4.6 Microcontrolador PIC16F873.

Este microcontrolador forma parte de la familia de los PIC16F87X.



4.6.1 Principales características

- Set de instrucciones reducido (RISC). Sólo 35 instrucciones para aprender.
- Las instrucciones se ejecutan en un sólo ciclo de máquina excepto los saltos que requieren 2 ciclos.
- Opera con una frecuencia de clock reloj de hasta 20 MHz (ciclo de máquina de 200 ns)
- Memoria de programa: 4096 posiciones de 14 bits.
- Memoria de datos: Memoria RAM de 192 bytes (8 bits por registro).
- Memoria EEPROM: 128 bytes (8 bits por registro).
- Stack de 8 niveles.
- 22 Terminales de I/O que soportan corrientes de hasta 25 mA.

- 5 Entradas analógicas de 10 bits de resolución.
- 3 Timers.
- Módulos de comunicación serie, comparadores, PWM.
- Características especiales:
- La memoria de programa se puede reescribir hasta 1000 veces.
- La memoria EEPROM se puede reescribir hasta 1000000 de veces.
- Los datos almacenados en la memoria EEPROM se retienen por 40 años y no se borran al quitar la alimentación al circuito.
- 13 fuentes de interrupción:
 - Señal externa (RB0).
 - Desborde de TMR0
 - Cambio en el estado de los terminales RB4, RB5, RB6 o RB7.
 - Ciclo de escritura en la memoria EEPROM completado.
 - Ciclo de conversión A/D finalizado
 - etc.

4.6.2 Descripción de los terminales

1-Terminales de entrada-salida (22 en total)

PORTA: RA0-RA5:

- Los terminales RA0-RA3 y RA5 son bidireccionales y manejan señales TTL. Pueden configurarse como entradas analógicas
- El terminal RA4 como entrada es Schmitt Trigger y como salida es colector abierto. Este terminal puede configurarse como clock de entrada para el contador TMR0.

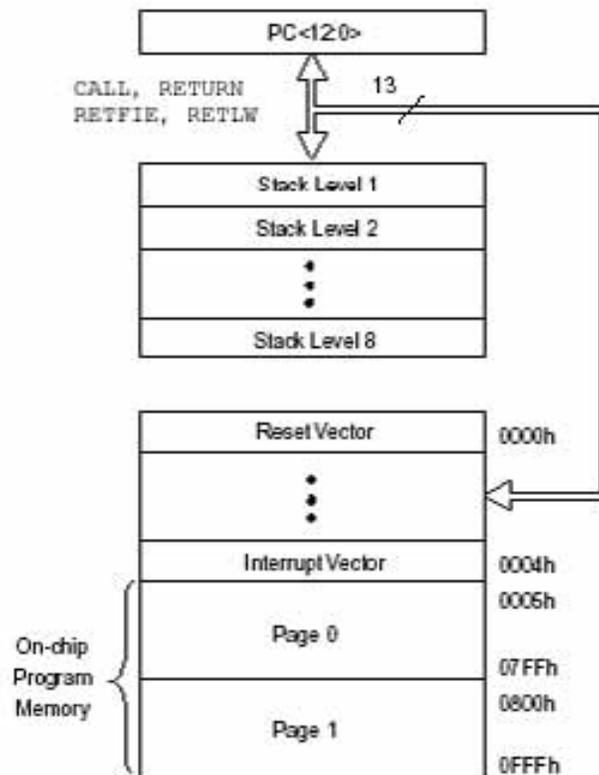
PORTB: RB0-RB7:

- Los terminales RB0-RB7 son bidireccionales y manejan señales TTL.
- Por software se pueden activar las resistencias de “pull-up” internas que evitan el uso de resistencias externas en caso de que los terminales se utilicen como entrada (esto permite, en algunos casos, reducir el número de componentes externos necesarios).
- RB0 se puede utilizar como entrada de pulsos para la interrupción externa.
- RB4-RB7 se pueden utilizar para la interrupción por cambio de estado.
- PORTC. Los terminales RC0-RC7 son bidireccionales y manejan señales TTL.
- Se utilizan en los módulos de PWM, comparadores y transmisión serial.

2-Otros terminales

- VDD. Positivo de alimentación. 2-6 Vcc.
- VSS. Negativo de alimentación.
- MCLR. Master Clear (Reset). Mientras en este terminal haya un nivel bajo (0 Vcc), el microcontrolador permanece inactivo.
- OSC1/CLKIN. Entrada del oscilador (cristal). Entrada de oscilador externo.
- OSC2/CLKOUT. Salida del oscilador (cristal).

- Memoria de uso general
- Registros especiales
- Memoria EEPROM
- El PIC16F873 dispone de una memoria de programa de 4096 posiciones (0000H a 03FFH) de 14 bits cada una. La memoria está dividida en 2 páginas.
- El contador de programa (PC) es el registro que contiene, en todo momento, la dirección de la próxima instrucción a ejecutarse. Este registro, de 13 bits, es capaz de direccionar hasta 8192 posiciones de memoria (más que suficiente para las 4096 del PIC16F873).
- Stack de 8 niveles. Son 8 registros de 13 bits en los que el microcontrolador va almacenando las direcciones de retorno de las diferentes subrutinas.
- La posición 0000H de la memoria es el denominado Vector de Reset. Ésta es la dirección de memoria que se carga en PC cuando el microcontrolador, estando alimentado, sale del estado de reset (MCLR=5Vcc). En esta dirección se debe almacenar un GOTO al inicio del programa.
- La dirección 0004H es el denominado Vector de interrupción. Cuando el sistema de interrupciones se encuentra habilitado y se produce un evento, en PC se carga automáticamente esta dirección. En ella debe comenzar la rutina de atención a las interrupciones.
- La memoria de programa propiamente dicha se divide en dos páginas (0005-07FF y 0800-0FFF). La página se debe seleccionar por software. Para ello se utiliza el bit 3 del registro PCLATH.



4.6.5 Organización de la memoria.

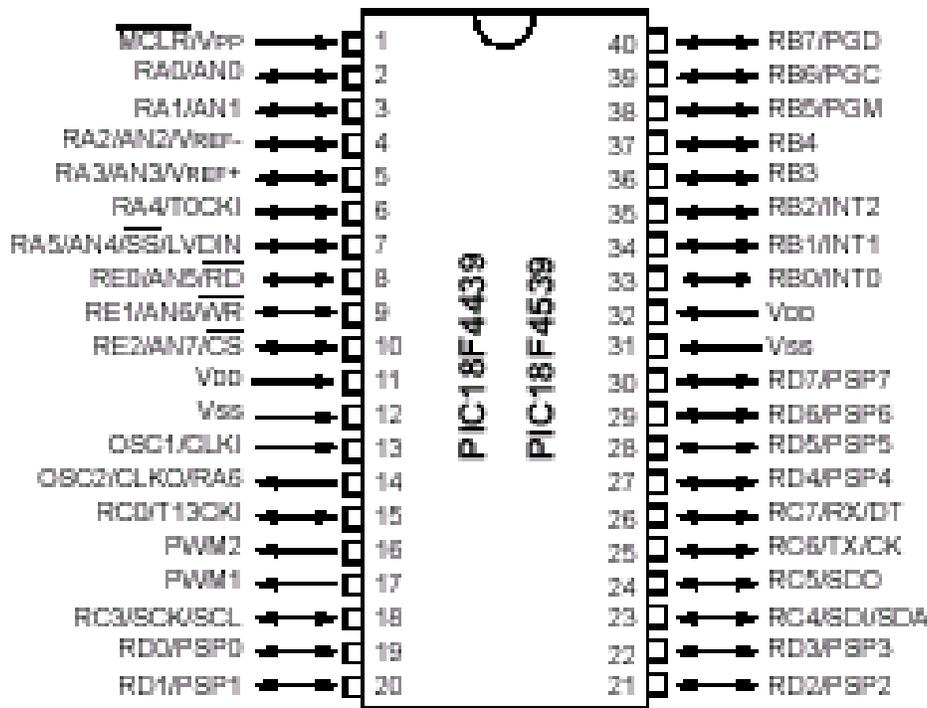
Indirect addr. ⁽¹⁾	00h	Indirect addr. ⁽¹⁾	80h	Indirect addr. ⁽¹⁾	100h	Indirect addr. ⁽¹⁾	180h
TMR0	01h	OPTION_REG	81h	TMR0	101h	OPTION_REG	181h
PCL	02h	PCL	82h	PCL	102h	PCL	182h
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h
FSR	04h	FSR	84h	FSR	104h	FSR	184h
PORTA	05h	TRISA	85h		105h		105h
PORTD	06h	TRISD	86h	PORTD	106h	TRISB	106h
PORTC	07h	TRISC	87h		107h		107h
	08h		88h		108h		108h
	09h		89h		109h		109h
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh
PIR1	0Ch	PIE1	8Ch	EEDATA	10Ch	EECON1	18Ch
PIR2	0Dh	PIE2	8Dh	EEADR	10Dh	EECON2	18Dh
TMR1L	0Eh	PCON	8Eh	EEDATH	10Eh	Reserved ⁽²⁾	18Eh
TMR1H	0Fh		8Fh	EEADRH	10Fh	Reserved ⁽²⁾	18Fh
T1CON	10h		90h		110h		100h
TMR2	11h	SSPCON2	91h				
T2CON	12h	PR2	92h				
SSPBUF	13h	SSPAD0	93h				
SSPCON	14h	SSPSTAT	94h				
CCPR1L	15h		95h				
CCPR1H	16h		96h				
CCP1CON	17h		97h				
RCSTA	18h	TXSTA	98h				
TXREG	19h	SPBRG	99h				
RCREG	1Ah		9Ah				
CCPR2L	1Bh		9Bh				
CCPR2H	1Ch		9Ch				
CCP2CON	1Dh		9Dh				
ADRESH	1Eh	ADRESL	9Eh				
ADCON0	1Fh	ADCON1	9Fh				
	20h		A0h				
General Purpose Register 96 Bytes		General Purpose Register 96 Bytes		accesses 20h-7Fh		accesses A0h - FFh	
Bank 0	7Fh	Bank 1	FFh	Bank 2	17Fh	Bank 3	1FFh
					16Fh 170h		1EFh 1F0h

- La memoria de datos es del tipo RAM. Está compuesta por registros de 8 bits y dividida en 2 áreas:
 - Registros especiales. Cada uno tiene una función específica que se analizará más adelante.
 - Registros de uso general. Son los registros que le permiten al usuario almacenar valores temporalmente (variables) durante la ejecución del programa.
- La memoria está dividida en 4 bancos.
 - Banco 0: 000H-07FH
 - Banco 1: 080H-0FFH
 - Banco 2: 100H-17FH
 - Banco 3: 180H-1FFH

- Cada banco se compone de 128 registros de 8 bits cada uno. Los primeros 20 registros se reservan para los registros especiales, los restantes se utilizan para uso general. No todos los registros están implementados físicamente. El banco se debe seleccionar por software. Para ello se utilizan los bits RP0 y RP1 del registro STATUS.
- Registros de uso general: sólo están implementados físicamente los registros del banco 0 y el 1. Son 192 registros. Las direcciones en el banco 2 y 3 están mapeadas a las direcciones correspondientes del banco 0 y 1 respectivamente. Por ejemplo, la dirección 02CH (banco 0) y 12CH (banco 2) acceden al mismo registro (02CH).
- Los registros especiales están repartidos en los 4 bancos.

4.7 EL PIC18FF4436

Este microcontrolador forma parte de la familia de los PIC18FXX36.



4.7.1 Principales características

- Set (RISC) de 75 instrucciones para aprender.
- Opera con una frecuencia de reloj de hasta 40 MHz
- Memoria de programa: 6144 posiciones de 14 bits. (14 kbytes)
- Memoria de datos. Memoria SRAM de 640 bytes (8 bits por registro).
- Memoria EEPROM. 256 bytes (8 bits por registro).
- Stack de 31 niveles.
- 32 Terminales de I/O que soportan corrientes de hasta 25 mA.
- 8 Entradas analógicas de 10 bits de resolución.
- 3 Timers.
- 2 módulos de comunicación serie, comparadores, PWM.

- Características especiales:
- La memoria de programa se puede reescribir hasta 1000 veces.
- La memoria EEPROM se puede reescribir hasta 1000000 de veces.
- Los datos almacenados en la memoria EEPROM se retienen por 40 años y no se borran al quitar la alimentación al circuito.
- 16 fuentes de interrupción:
 - Señal externa (RB0).
 - Desborde de TMR0
 - Cambio en el estado de los terminales RB4, RB5, RB6 o RB7.
 - Ciclo de escritura en la memoria EEPROM completado.
 - Ciclo de conversión A/D finalizado
 - etc.

4.7.2 Descripción de los terminales

1-Terminales de entrada-salida (32 en total)

PORTA: RA0-RA6:

- Las terminales RA0-RA1 son bidireccionales y manejan señales TTL. Pueden configurarse como entradas analógicas.
- RA2 es bidireccional y maneja señales TTL. Puede ser configurada como entrada analógica o como entrada de referencia baja de tensión del A/D.
- RA3 es bidireccional y maneja señales TTL. Puede ser configurada como entrada analógica o como entrada de referencia alta de tensión del A/D.
- RA4 es bidireccional y maneja señales TTL. Puede ser configurada como entrada de reloj de un timer externo.
- RA5 es bidireccional y maneja señales TTL. Puede ser configurado como entrada analógica, como entrada es Schmitt Trigger o como entrada analógica detectora de baja tensión.
- RA6 es bidireccional y maneja señales TTL. Puede ser configurado como la salida del reloj oscilador

PORTB: RB0-RB7:

- Las terminales RB0-RB7 son bidireccionales y manejan señales TTL.
- Por software se pueden activar las resistencias de pull-up internas, que evitan el uso de resistencias externas en caso de que los terminales se utilicen como entrada (esto permite, en algunos casos, reducir el número de componentes externos necesarios).
- RB0-RB2 se puede utilizar como entrada de pulsos para la interrupción externa.
- RB4-RB7 se pueden utilizar para la interrupción por cambio de estado.

PORTC: RC0-RC7:

- Las terminales RC0-RC7 son bidireccionales y manejan señales TTL.
- Se utilizan en los módulos de PWM, comparadores y transmisión serial.

PORTD: RD0-RD7

- Las terminales RD0-RD7 son bidireccionales o puerto paralelo data SLAVE.

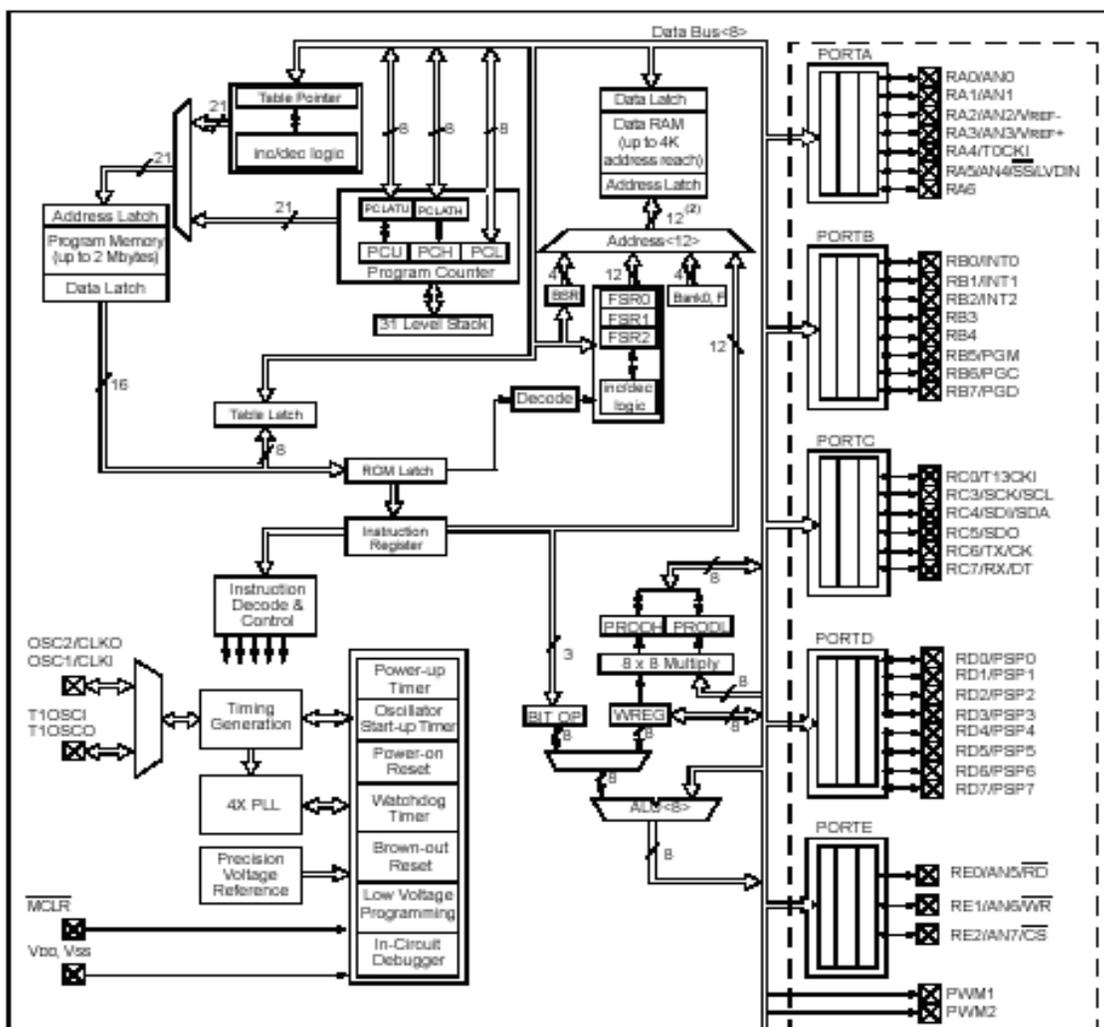
PORTE: RE0-RE2

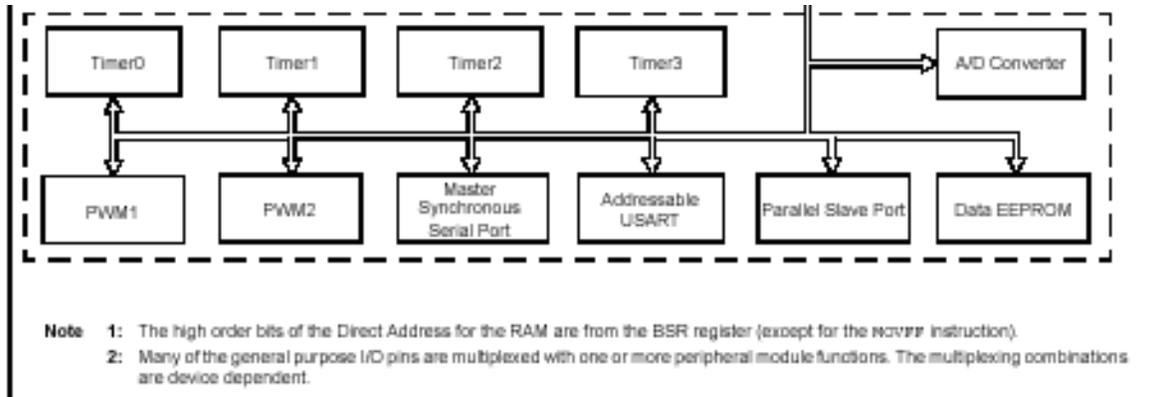
- Las terminales RE0-RE2 son bidireccionales. Pueden ser configuradas como entradas analógicas

2-Otros terminales

- PWM1: Salida 1 (control de motor)
- PWM2: Salida 2 (control de motor)
- VDD: Positivo de alimentación. 2-6 Vcc.
- VSS: Negativo de alimentación.
- MCLR: Master Clear (Reset). Mientras en este terminal haya un nivel bajo (0 Vcc), el microcontrolador permanece inactivo.
- OSC1/CLKIN: Entrada del oscilador (cristal). Entrada de oscilador externo.
- OSC2/CLKOUT: Salida del oscilador (cristal).

4.7.3 El PIC18F4439 por dentro



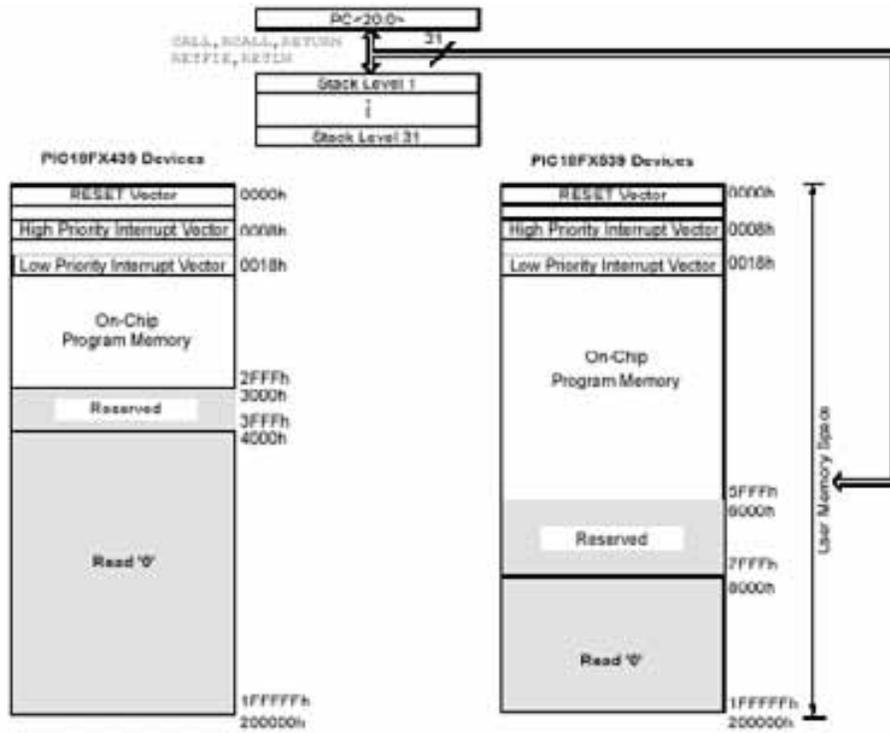


Como se observa en el esquema del diagrama de bloques del PIC, éste consta de un procesador con una ALU y un Decodificador de Instrucciones, una memoria de programas tipo FLASH de 4K palabras de 14 bits, una memoria de datos SRAM con 640 posiciones de 8 bits. También existe una zona de 256 bytes de EEPROM para datos no volátiles. Finalmente dispone de interrupciones, un temporizador, WDT (perro guardián), los terminales de E/S (PORTA, PORTB, PORTC, PORTE y PORTD) configurables por software y los módulos especiales (timers, comparadores, puerto serie).

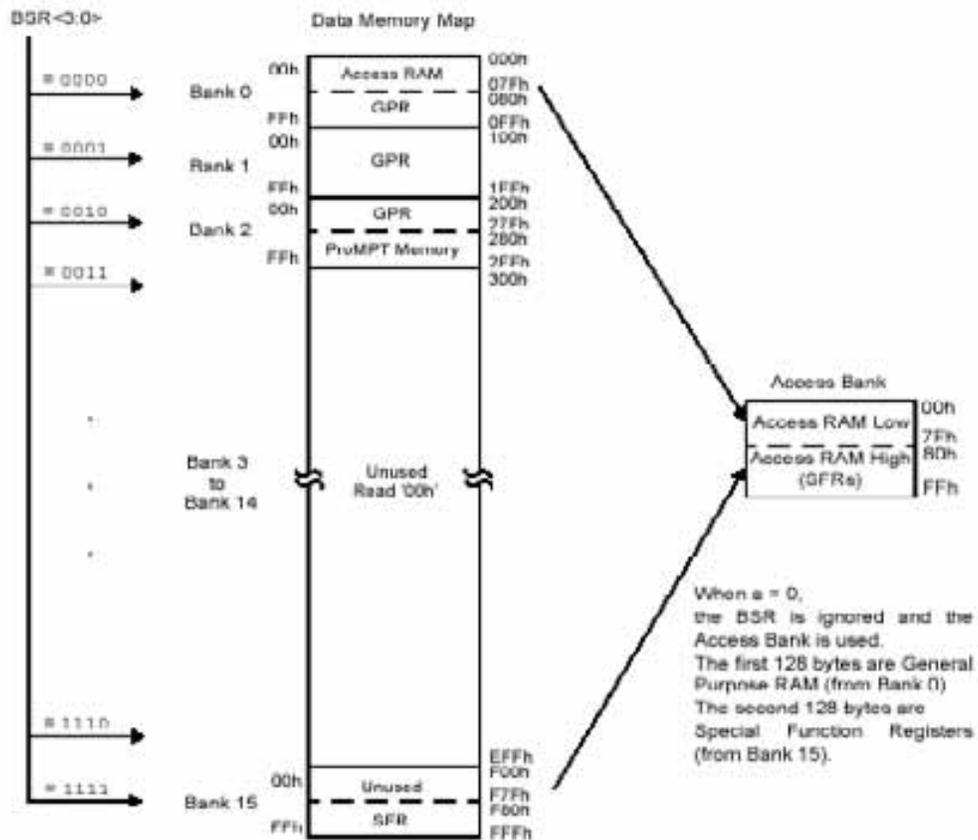
4.7.4 Organización de la memoria.

Hay 3 bloques de memoria en el PIC18F4439. La memoria de programa y la memoria de datos. Cada bloque tiene su propio bus (arquitectura Harvard). La memoria de datos está dividida en:

- Memoria de programa
 - Memoria RAM
 - Memoria EEPROM
- El PIC18F4439 dispone de una memoria de programa de 12Kbytes, o 6K de palabras simples de instrucción de memoria FLASH de la dirección 0000h a 02FFh.
 - Los siguientes 4Kbytes de espacio (de 3000h a 3FFFh) son reservados para el control de motor kernel.
 - El vector de reset está en la dirección 0000h y los vectores de interrupción de 0008h a 0018h.
 - El contador de programa (PC) es el registro que contiene, en todo momento, la dirección de la próxima instrucción a ejecutarse. Este registro, de 13 bits, es capaz de direccionar hasta 8192 posiciones de memoria.
 - Stack de 31 niveles. Son 31 registros de 13 bits en los que el microcontrolador va almacenando las direcciones de retorno de las diferentes subrutinas.



4.7.5 Organización de la memoria. Memoria de datos

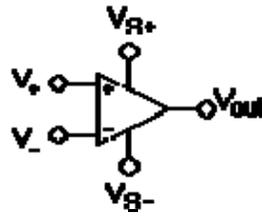


Address	Name	Address	Name	Address	Name	Address	Name
FFFh	TOSLJ	FD9h	INDF2 ⁽²⁾	FBFh	CCPR1H	F0Fh	IPR1
FFEh	TOSH	FDEh	POSTINC2 ⁽²⁾	FBEh	CCPR1L	F0Eh	PIR1
FFDh	TOSL	FDDh	POSTDEC2 ⁽²⁾	FBDh	CCP1CON ²	F9Dh	PIE1
FFCh	STKP1R	FDCh	PREINC2 ⁽²⁾	FBCCh	CCPR2H	F9Ch	---
FFBh	PCLATU	FDBh	PLUSWZ ⁽²⁾	FBBh	CCPR2L	F9Bh	---
FFAh	PCLATH	FDAh	FSR2H	FBAh	CCP2CON ²	F9Ah	---
FF9h	PCL	FD9h	FSR2L	F89h	---	F99h	---
FF8h	TBLPTRLJ	FD8h	STATUS	F88h	---	F98h	---
FF7h	TBLPTRH	FD7h	TMR0H	F87h	---	F97h	---
FF6h	TBLPTRL	FD6h	TMR0L	F86h	---	F96h	TRIGE ⁽²⁾
FF5h	TABLAT	FD5h	T0CON	F85h	---	F95h	TRISD ⁽²⁾
FF4h	PRODH	FD4h	---	F84h	---	F94h	TRISC ⁽²⁾
FF3h	PRODL	FD3h	O5CCON ²	F83h	TMR3H	F93h	TRISB
FF2h	INTCON	FD2h	LVDCON	F82h	TMR3L	F92h	TRISA
FF1h	INTCON2	FD1h	WDTCON	F81h	T3CON	F91h	---
FF0h	INTCON3	FD0h	RCON	F80h	---	F90h	---
FEFh	INDFO ⁽²⁾	FCFh	TMR1H	FAFh	SPDRG	F8Fh	---
FEeh	POSTINC0 ⁽²⁾	FCEh	TMR1L	FAEh	RCREG	F8Eh	---
FECh	POSTDEC0 ⁽²⁾	FCDh	T1CON	FADh	TXREG	F8Dh	LATE ⁽²⁾
FEbCh	PREINC0 ⁽²⁾	FCCh	TMR2 ²	FACCh	TXSTA	F8Ch	LATD ⁽²⁾
FEbH	PLUSW0 ⁽²⁾	FC0h	PR2 ²	FADh	RCSTA	F80h	LATC ⁽²⁾
FEACh	FSR0H	FCAh	T2CON	FAAh	---	F8Ah	LATB
FE9h	FSR0L	FC9h	SSPBUF	FA9h	EEADR	F89h	LATA
FE8h	WREG	FC8h	SSPAD	FABh	EEDATA	F88h	---
FE7h	INDF1 ⁽²⁾	FC7h	SSPSTAT	FA7h	EECON2	F87h	---
FEFh	POSTINC1 ⁽²⁾	FC6h	SSPCON1	FABh	EECON1	F86h	---
FE6h	POSTDEC1 ⁽²⁾	FC6h	SSPCON2	FA5h	---	F86h	---
FE4h	PREINC1 ⁽²⁾	FC4h	ADRESH	FA4h	---	F04h	PORTE ⁽²⁾
FE3h	PLUSW1 ⁽²⁾	FC3h	ADRESL	FA3h	---	F83h	PORTD ⁽²⁾
FE2h	FSR1H	FC2h	ADCON0	FA2h	IPR2	F82h	PORTC ⁽²⁾
FE1h	FSR1L	FC1h	ADCON1	FA1h	PIR2	F81h	PORTB
FE0h	BSR	FC0h	---	FA0h	PIE2	F80h	PORTA

- La memoria de datos es del tipo RAM. Está compuesta por registros de 12 bits y dividida en 2 áreas:
 - Registros especiales: Cada uno tiene una función específica que se analizará más adelante.
 - Registros de uso general: Son los registros que le permiten al usuario almacenar valores temporalmente (variables) durante la ejecución del programa.
- La memoria está dividida en 16 bancos de 256 bytes.
 - 640 bytes de data RAM extendidos desde el banco 0 al banco 2. (000h a 27Fh)
 - El bloque de 128 (280h a 2FFh) es usado como memoria de datos del control motor kernel, y no están habilitados para usarse.
 - Los registros especiales están repartidos en los 4 bancos.

5. AMPLIFICADOR OPERACIONAL (OPAM)

El símbolo de un **A.O.** es el mostrado en la siguiente figura:



Los terminales son:

V_+ : entrada no inversora

V_- : entrada inversora

V_{OUT} : salida

V_{S+} : alimentación positiva

V_{S-} : alimentación negativa

Las terminales de alimentación pueden recibir diferentes nombres, por ejemplo en los **A.O.** basados en FET V_{DD} y V_{SS} respectivamente. Para los basados en BJT son V_{CC} y V_{EE} .

Normalmente los pines de alimentación son omitidos en los diagramas eléctricos por claridad.

El Amplificador Operacional ideal se caracteriza por:

1. Resistencia de entrada, (R_{en}), tiende a infinito.
2. Resistencia de salida, (R_o), tiende a cero.
3. Ganancia de tensión de lazo abierto, (A), tiende a infinito
4. Ancho de banda (BW) tiende a infinito.
5. $v_o = 0$ cuando $v_+ = v_-$.

5.1 Comportamiento en continua (DC)

5.1.1 Lazo abierto

Si no existe realimentación la salida del **A.O.** será la resta de sus dos entradas multiplicada por un factor. Este factor suele ser del orden de 100.000 (que se considerará infinito en calculos con el componente ideal). Por lo tanto si la diferencia entre las dos tensiones es de 1mV la salida debería ser 100V. Debido a la limitación que supone no poder entregar más tensión de la que hay en la alimentación, el **A.O.** estará saturado si se da este caso. Si la tensión más alta es la aplicada en el pin + la salida será la que corresponde a la alimentación V_{S+} , mientras que si la tensión más alta es la del pin - la salida será la alimentación V_{S-} .

5.1.2 Lazo cerrado

Se conoce como lazo a la realimentación en un circuito. Aquí se supondrá realimentación negativa. Para conocer el funcionamiento de esta configuración se parte de las tensiones en las dos entradas exactamente iguales, se supone que la tensión en el pin + sube y, por tanto, la tensión en la salida también se eleva. Como existe la realimentación entre la salida y la entrada -, la tensión en esta entrada también se eleva, por tanto la diferencia entre las dos entradas se reduce, disminuyéndose también la salida. Este proceso pronto se estabiliza y se tiene que la salida es la necesaria para mantener las dos entradas, idealmente, con el mismo valor.

Siempre que hay realimentación negativa se aplican estas dos aproximaciones para analizar el circuito:

$$\begin{aligned}V_+ &= V_- \\I_+ &= I_- = 0\end{aligned}$$

5.2 Comportamiento en alterna (AC)]

En principio la ganancia calculada para continua puede ser aplicada para alterna, pero a partir de ciertas frecuencias aparecen limitaciones. Para tener un amplificador operacional se debe contar con el 741op

5.2.1 Análisis

Para analizar un circuito en el que haya **A.O.** puede usarse cualquier método, pero uno habitual es:

Comprobar si tiene realimentación negativa

Si tiene realimentación negativa se pueden aplicar las reglas del apartado anterior

Definir la corrientes en cada una de las ramas del circuito

Aplicar el método de los nudos en todos los nodos del circuito excepto en los de salida de los amplificadores (porque en principio no se puede saber la corriente que sale de ellos)

Aplicando las reglas del apartado 2 resolver las ecuaciones para despejar la tensión en los nodos donde no se conozca.

5.2.1 Configuraciones

5.2.1.1 Comparador

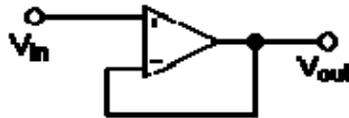


Ésta es una aplicación sin la realimentación. Compara entre las dos entradas y saca una salida en función de qué entrada sea mayor. Se puede usar para adaptar niveles lógicos.

$$V_{out} = \begin{cases} V_{S+} & V_1 > V_2 \\ V_{S-} & V_1 < V_2 \end{cases}$$

5.2.1.2 Seguidor

Es aquel circuito que proporciona a la salida la misma tensión que a la entrada

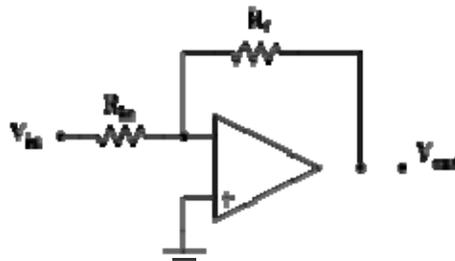


Se usa como un buffer, para eliminar efectos de carga o para adaptar impedancias (conectar un dispositivo con gran impedancia a otro con baja impedancia y viceversa)

Como la tensión en las dos las entradas es igual: $V_{out} = V_{in}$

$$Z_{in} = \infty$$

5.2.1.3 Inversor



El análisis de este circuito es el siguiente:

$$V_+ = V_- = 0$$

Definiendo corrientes: $\frac{V_{in} - 0}{R_{in}} = -\frac{V_{out} - 0}{R_f}$ y de aquí se despeja

$$V_{out} = -V_{in} \frac{R_f}{R_{in}}$$

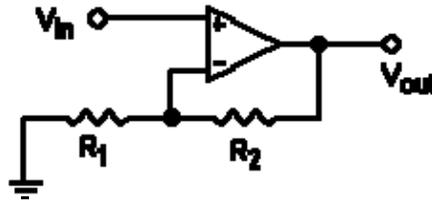
Para el resto de circuitos el análisis es similar.

$$Z_{in} = R_{in}$$

Por lo cual podemos controlar la impedancia de entrada mediante la elección de R_1

$$V_o = -\frac{R_2}{R_1} \cdot V_{in}$$

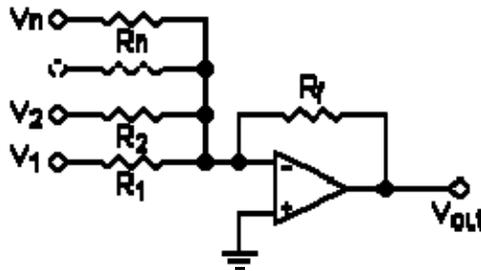
5.2.1.4 No inversor



$$V_{out} = V_{in} \left(1 + \frac{R_2}{R_1} \right)$$

$$Z_{in} = R_{in}$$

5.2.1.5 Sumador inversor



La salida está invertida

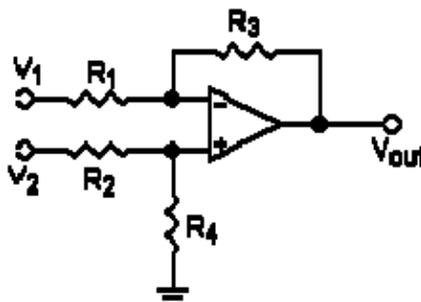
Para resistencias independientes R_1, R_2, \dots, R_n

$$V_{out} = -R_f \left(\frac{V_1}{R_1} + \frac{V_2}{R_2} + \dots + \frac{V_n}{R_n} \right)$$

La expresión se simplifica bastante si se usan resistencias del mismo valor

Impedancias de entrada: $Z_n = R_n$

5.2.1.6 Diferenciador

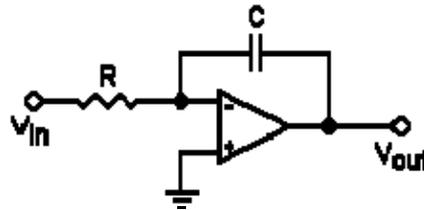


Para resistencias independientes R_1, R_2, R_3, R_4 :

$$V_{out} = V_2 \left(\frac{(R_3 + R_1)R_4}{(R_4 + R_2)R_1} \right) - V_1 \left(\frac{R_3}{R_1} \right)$$

Igual que antes esta expresión puede simplificarse con resistencias iguales
La impedancia diferencial entre dos entradas es $Z_{in} = R_1 + R_2$

5.2.1.7 Integrador ideal



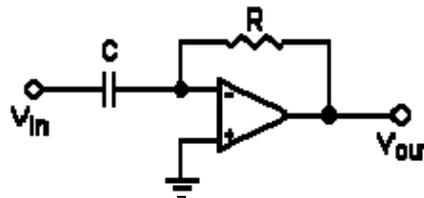
Integra e invierte la señal (V_{in} y V_{out} son funciones dependientes del tiempo)

$$V_{out} = \int_0^t -\frac{V_{in}}{RC} dt + V_{inicial}$$

$V_{inicial}$ es la tensión de salida en el origen de tiempos ($t = 0$). Este circuito también se usa como filtro

NOTA: En la práctica se realizan modificaciones a este circuito porque no es estable.

5.2.1.8 Derivador ideal



Deriva e invierte la señal respecto al tiempo

$$V_{out} = -RC \frac{dV_{in}}{dt}$$

Este circuito también se usa como filtro

NOTA: Es un circuito que no se utiliza en la práctica porque no es estable.

5.2.1.8 Otros

- Osciladores, como el *punte de Wien*

- Convertidores carga-tensión
- Convertidores corriente-tensión
- Filtros activos
- Girador (simula un inductor)

5.3 APLICACIONES

- Calculadoras analógicas
- Filtros
- Preamplificadores y buffers de audio y vídeo
- Reguladores
- Conversores
- Evitar el efecto de carga
- Adaptadores de niveles (por ejemplo CMOS y TTL)

5.4 ESTRUCTURA

Aunque es usual presentar al **A.O.** como una *caja negra* con características ideales es importante entender la forma en que funciona, de esta forma se podrán entender mejor las limitaciones que presenta.

Los diseños varían entre cada fabricante y cada producto, pero todos los **A.O.** tienen básicamente la misma estructura interna que consiste en tres etapas:

Amplificador diferencial: es la etapa de entrada que proporciona una baja amplificación del ruido y gran impedancia de entrada. Suelen tener una salida diferencial.

Amplificador de tensión: proporciona una ganancia de tensión.

Amplificador de salida: proporciona la capacidad de suministrar la corriente necesaria, tiene una baja impedancia de salida y, usualmente, protección frente a cortocircuitos

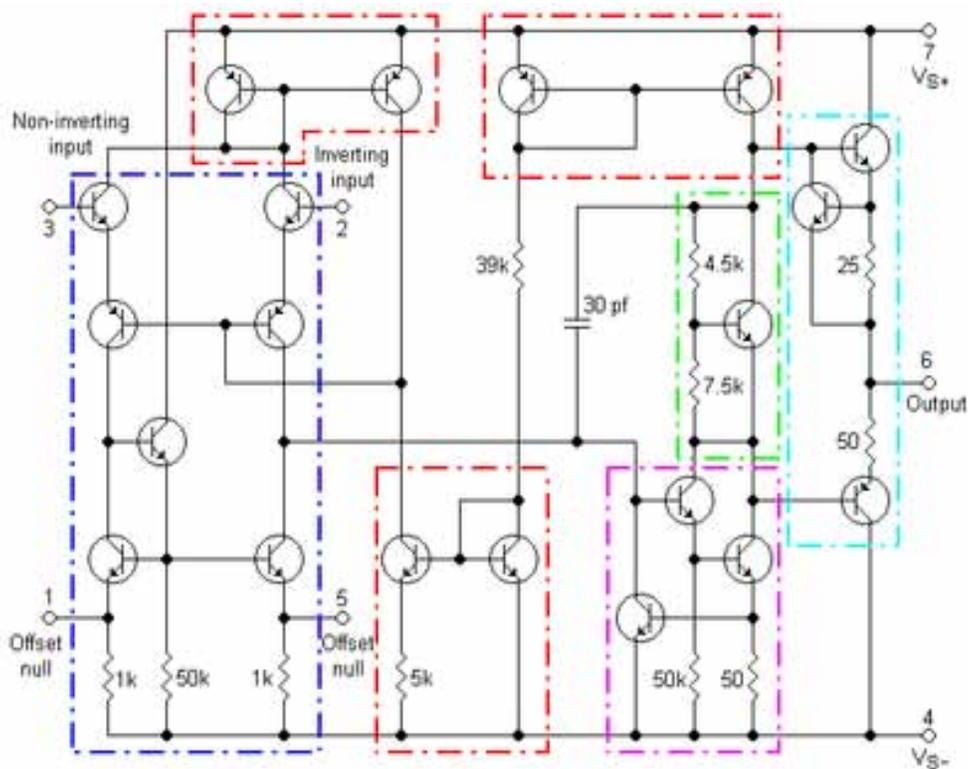


Diagrama electrónico del operacional 741.

En el diagrama se destaca en azul el amplificador diferencial. Éste es el responsable de que las corrientes de entrada no sean cero, pero sí respecto a las de los colectores. (Nótese como a pesar de aproximar las corrientes de entrada a 0, si éstas realmente fueran 0 el circuito no funcionaría). La impedancia de entrada es de unos $2M\Omega$.

Las etapas en rojo son espejos de corriente. El superior de la izquierda sirve para poder soportar grandes tensiones en modo común en la entrada. El superior de la derecha proporciona una corriente a la circuitería de salida para mantener la tensión. El inferior tiene una baja corriente de colector debido a las resistencias de $5k\Omega$. Se usa como conexión de gran impedancia a la alimentación negativa para poder tener una tensión de referencia sin que haya efecto de carga en el circuito de entrada.

Los pines llamados *Offset null* son usados para eliminar las tensiones de offset que pueda haber en el circuito. La etapa de ganancia en tensión es NPN.

La sección verde es un desplazador de tensión. Esto proporciona una caída de tensión constante sin importar la alimentación. En el ejemplo 1V. Esto sirve para prevenir la distorsión.

El capacitor se usa como parte de un filtro paso bajo para reducir la frecuencia y prevenir que el **A.O** oscile.

La salida en cebleste es un amplificador PNP seguidor con emisor “push-pull”. El rango de la tensión de salida es de un volt menos a la alimentación, la tensión colector-emisor de los transistores de salida nunca puede ser totalmente cero. Las resistencias de salida hacen que la corriente de salida esté limitada a unos 25mA. La resistencia de salida no es cero, pero con realimentación negativa se aproxima.

5.5 PARAMETROS

Ganancia en lazo abierto. Indica la ganancia de tensión en ausencia de realimentación. Se puede expresar en unidades naturales (V/V, V/mV) o logarítmicas (dB). Son valores habituales 100.000 a 1.000.000 V/V.

Tensión en modo común. Es el valor medio de tensión aplicada a ambas entradas del operacional.

Tensión de offset. Es la diferencia de tensión, aplicada a través de resistencias iguales, entre las entradas de un operacional que hace que su salida tome el valor cero.

Corriente de offset. Es la diferencia de corriente entre las dos entradas del operacional que hace que su salida tome el valor cero.

Margen de entrada diferencial. Es la mayor diferencia de tensión entre las entradas del operacional que mantienen el dispositivo dentro de las especificaciones.

Corrientes de polarización (*bias*) de entrada. Corriente media que circula por las entradas del operacional en ausencia de señal
“Slew rate”. Velocidad de cambio de la tensión de salida ante un escalón a la entrada. Se mide en V/ μ s, kV/ μ s o similares.

Rechazo de rizado en modo común. Relación entre la variación de la tensión de *offset* y el valor de la tensión en modo común. Se mide en dB.

5.6 LIMITACIONES

5.6.1 SATURACION

Un **A.O** típico no puede suministrar más de la tensión a la que se alimenta, normalmente algunos volt menos. Cuando se da este valor se dice que se satura, pues ya no está amplificando. La saturación puede ser aprovechada por ejemplo en circuitos comparadores. Un concepto asociado a éste es el “Slew rate”.

5.6.2 TENSIÓN DE OFFSET

Es la diferencia de tensión que se da entre sus dos patillas de entrada cuando la tensión de salida es nula. Se puede expresar su dependencia de la temperatura (T) como:

$$V_{OFFSET} = V_{OFFSET}(T_0) + \frac{\Delta V_{OFFSET}}{\Delta T} (T - T_0)$$

Donde T_0 es una temperatura de referencia.

Una característica derivada de ésta es el PSRR que se expresa como:

$$\frac{1}{PSRR} = \frac{\Delta V_{OFFSET}}{\Delta V_{CC}}$$

5.6.3 CORRIENTES

Aquí hay dos tipos de corrientes que considerar y que los fabricantes suelen proporcionar:

$$I_{OFFSET} = |I_+ - I_-|$$

$$I_{BIAS} = \frac{I_+ + I_-}{2}$$

Idealmente ambas deberían ser cero.

5.6.4 CARACTERISTICAS TENSION-FRECUENCIA

Al **A.O.** típico también se le conoce como amplificador realimentado en tensión (VFA). En él hay una importante limitación respecto a la frecuencia: El producto de la ganancia en tensión por el ancho de banda es constante.

Como la ganancia en lazo abierto es del orden de 100.000 un amplificador con esta configuración sólo tendría un ancho de banda de unos pocos hertz. Al realimentar negativamente se baja la ganancia a valores del orden de 10 a cambio de tener un ancho de banda aceptable. Existen modelos de diferentes A.O. para trabajar en frecuencias superiores, en estos amplificadores es primordial mantener las características a frecuencias más altas que el resto, sacrificando a cambio un menor valor de ganancia u otro aspecto técnico.

5.6.5 CAPACITANCIA

El **A.O.** presenta capacitancias (capacidades) parásitas, las cuales producen una disminución de la ganancia conforme se aumenta la frecuencia.

5.6.6 DERIVA TERMICA

Debido a que una unión semiconductor varía su comportamiento con la temperatura, los **A.O.** también cambian sus características, en este caso hay que diferenciar el tipo de transistor en el que está basado, así las corrientes anteriores variarán de forma diferente con la temperatura si son bipolares o JFET.

5.8 EFECTO CARGA

El efecto de carga, también conocido como regulación es la pérdida de voltaje a medida que aumenta la carga, viene dado por la relación:

$$VL = Vo * [1 / (1 + (Ro / RL))]$$

Donde:

Ro. resistencia inicial.

RL. resistencia inducida

6. EL ATS

6.1 El sistema ATS

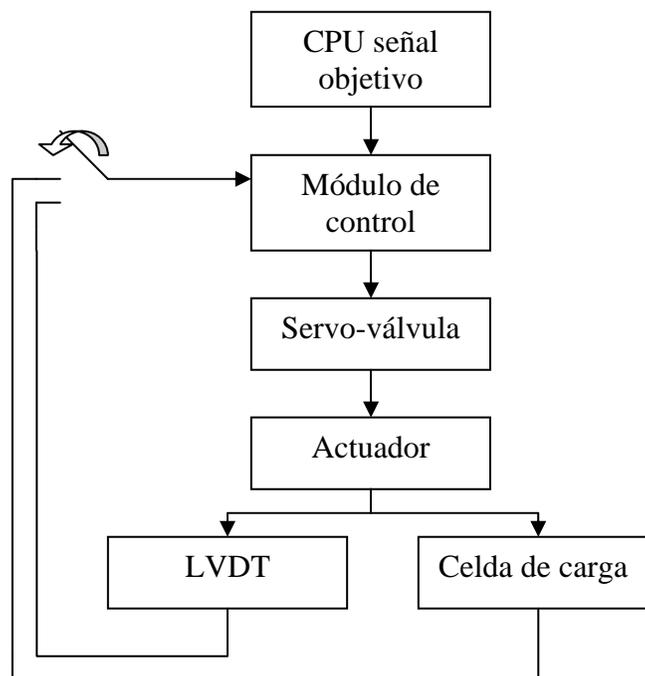
El sistema ATS cuenta con cuatro actuadores, los cuales mediante una servo válvula, marca MTS modelo 256 que controla el flujo de aceite que los alimenta, son controlados en carga o desplazamiento.

Los actuadores cuentan con una celda de carga cada uno con el propósito de conocer la carga aplicada por el cabezal del pistón, sobre alguna estructura. Además cuenta con un LVDT (transductor de posición) cada gato que permite conocer la posición del actuador.

El sistema cuenta con un lazo de control cerrado el cual puede operar con la señal de carga o desplazamiento logrando un control de carga o desplazamiento.

Así pues el módulo de control es capaz de seleccionar la señal de control y generar la señal objetivo aplicada al circuito de lazo cerrado que genera la señal de error.

6.2 DIAGRAMA DE BLOQUES DEL ATS



6.3 EL ACTUADOR DEL ATS

El actuador el ATS consta de un pistón el cual cuenta con dos cámaras en su armadura. Cada cámara es alimentada por un flujo de aceite, el cual es regulado por la servo-válvula.

El fin de tener dos cámaras en la armadura, es para realizar un desplazamiento de mayor precisión del pistón. Si se desea que el pistón tenga un cierto desplazamiento en un cierto sentido, una de las cámaras se llenará, mientras la otra libera aceite, en el momento de que la señal de error generada por el módulo de control indique que el punto objetivo a sido sobrepasado, la segunda cámara ahora será llenada para compensar el sobrepaso. Así hasta que la señal de error sea cero.

Hay que hacer constar que al ser un sistema hidráulico de una gran dimensión, la velocidad de desplazamiento del pistón es mucho menor a la velocidad de respuesta del sistema, por lo que la estabilización en el punto objetivo es muy eficaz.

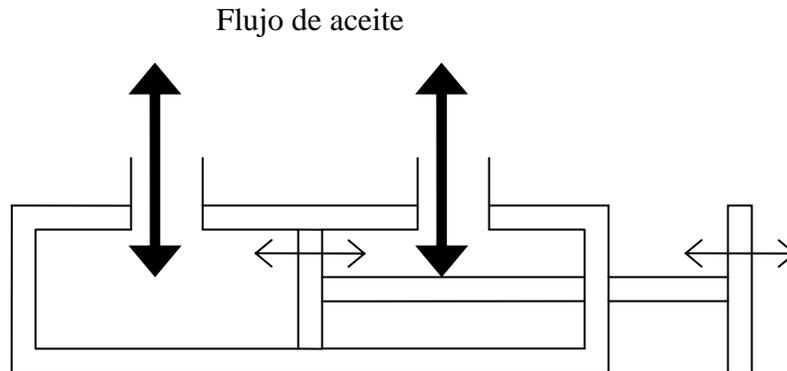
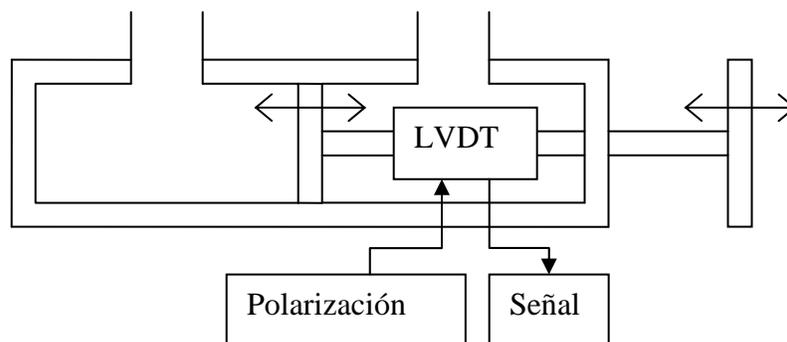


DIAGRAMA DE PISTON CON CAMARAS

En el interior de la segunda cámara existe un LVDT utilizando al émbolo del pistón como núcleo, generando una variación de tensión cada que el émbolo se desplace, proporcionando así la posición de éste, referido a una señal de tensión.

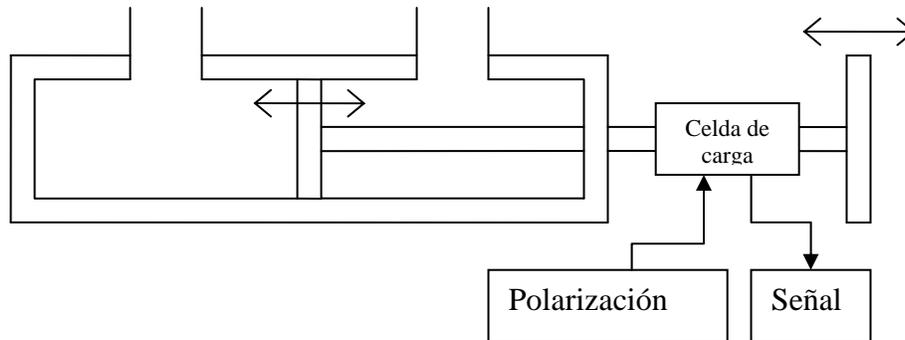


El LVDT del sistema ATS es polarizado con una fuente de AC a 3 Vpp y entrega una señal de 100mV al máximo desplazamiento. Por lo que la posición está dada mediante la siguiente fórmula.

$$Posición = \frac{Voltaje\ de\ la\ señal}{100mV} \times máximo\ desplazamiento$$

Esta señal es alimentada a un acondicionador que da una señal de $\pm 10V$ de corriente directa.

En el cabezal del pistón existe una celda de carga, la cual tiene como fin el generar una señal de tensión proporcional a la **carga/fuerza** ejercida por el actuador sobre alguna estructura.



Al generar una carga por parte del actuador contra la estructura, la celda de carga sufre una deformación y con base en la variación de resistencia eléctrica que experimentan algunas aleaciones cuando se modifica su forma, el nivel de tensión generado en la celda varía en función de la carga/fuerza ejercida.

La celda de carga está polarizada con una fuente de DC a 10V y genera una señal de 20mV a carga completa. Por lo que la carga aplicada por el gato está dada por la siguiente ecuación:

$$Carga = \frac{Voltaje\ de\ la\ señal}{20mV} \times Carga\ máxima$$

La carga que pueden aplicar los actuadores horizontales es de 100 toneladas, mientras que la carga que de los actuadores verticales es solo de 50 toneladas.

De igual forma que la señal del LVDT, antes de entrar al módulo de control, la señal generada por la celda de carga es acondicionada y se entrega una señal de $\pm 10V$ al módulo de control.

6.4 EL MÓDULO DE CONTROL

El módulo de control proporciona el servo control. El módulo de control compara la señal objetivo con alguna de las dos señales provenientes de los transductores, de no ser iguales, el módulo de control realiza un proceso de control y genera una señal de error la cual es alimentada a la servo-válvula como señal de control. El módulo de control cuenta con un control de válvula, el cual contiene el circuito necesario para el control de los tres estados del carrete esclavo de posición (slave spool). El control de la válvula proporciona 10 Hz de excitación para los tres estados del LVDT.

La entrada realimentación del módulo de control debe ser de +/- 10 V a escala completa, y su frecuencia de respuesta es de 3 kHz. El amplificador de la válvula entrega de 40mA a 20mA.

6.5 LA SERVO-VÁLVULA DEL ATS

La servo-válvula modelo 256 tiene tres aplicaciones típicas:

- Control de actuadores hidráulicos en posición
- Control de actuadores hidráulicos en carga
- Control de actuadores hidráulicos en velocidad

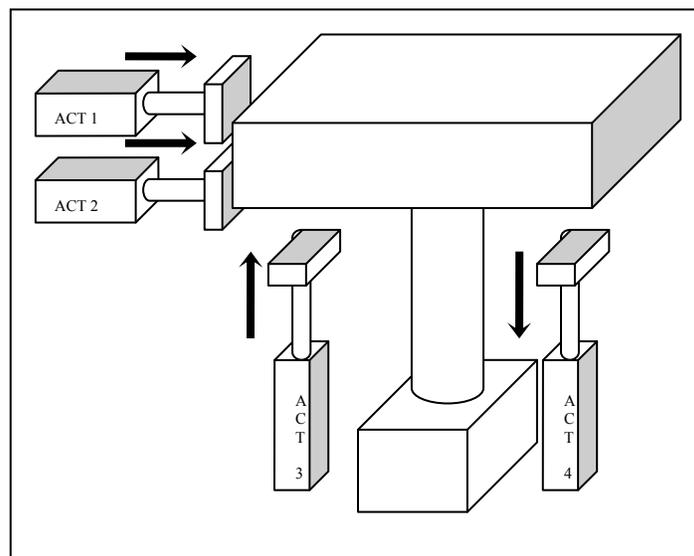
En el sistema ATS del Instituto de Ingeniería solo se trabaja a carga controlada y desplazamiento controlado (posición).

Opera hasta con 21 MPa de presión y un flujo de 40 gpm (galones por minuto).

7. DISPOSITIVO DE PROPORCIONALIDAD ENTRE CARGA

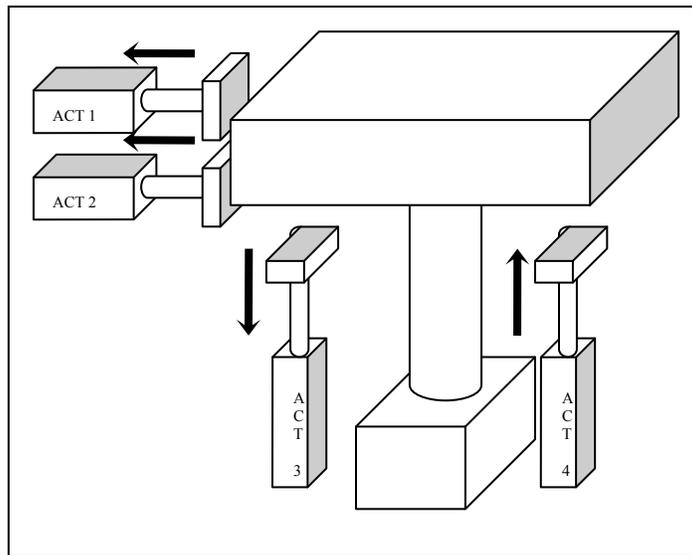
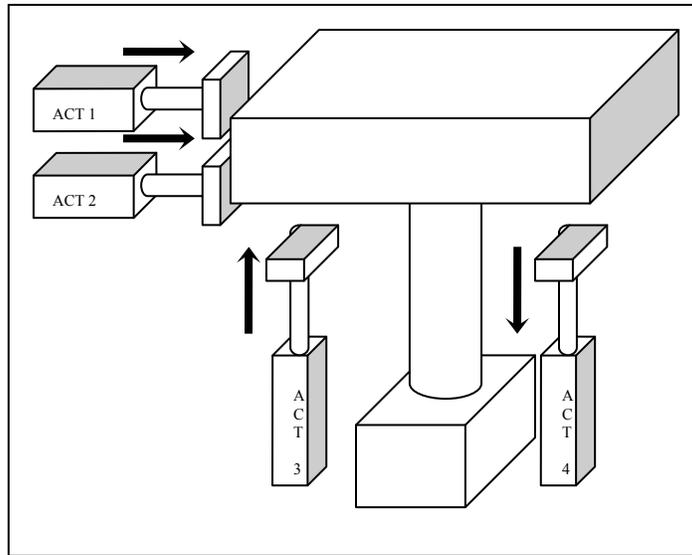
El sistema ATS cuenta con cuatro servo-actuadores hidráulicos que pueden trabajar bajo el concepto de carga o desplazamiento controlado, cada uno de ellos cuenta con una servo-válvula que regula el flujo de aceite, en función de la señal obtenida del transductor de carga o desplazamiento que se encuentra acoplado mecánicamente al actuador y de la señal objetivo, las cuales son procesadas por el sistema de control ATS.

El dispositivo a construir debe mantener una relación entre las dos cargas aplicadas horizontalmente y las dos verticales, por lo que la magnitud de carga de los actuadores verticales (Actuadores 3 y 4) estará en dependencia de la señal de carga entregada por los actuadores horizontales (Actuadores 1 y 2).



Mientras los actuadores horizontales trabajan en forma paralela, los verticales lo hacen en antiparalelo; es decir, mientras uno empuja el otro jala; sin perder la proporcionalidad de la carga horizontal con respecto de la vertical.





La carga horizontal total es la suma de las cargas individuales aplicadas horizontalmente. Esta magnitud total es con el que se deberá mantener la proporcionalidad de cada gato vertical, por lo que el primer paso del procesamiento de las señales de carga, es la adición entre si de los niveles de carga horizontal.

$$C_3 = C_1 + C_2$$

$$C_4 = -(C_1 + C_2)$$

Siendo C_n nivel de carga dado por el actuador n.

Nótese que el nivel con los actuadores 3 y 4 son de signo contrario, esto con el fin de que operen en forma antiparalela. Por lo que también se tiene que hacer un inversor para procesar la señal que será implementada sobre uno de los gatos verticales, en este caso sobre el actuador 4.

El nivel de carga total horizontal sobrepasa la capacidad con la que pueden trabajar los gatos verticales, pues la capacidad de cada actuador horizontal es de cien toneladas que es el doble de la capacidad de un actuador vertical, dando como resultado, que si se aplicara el valor máximo de la carga total horizontal (doscientas toneladas) en los gatos verticales, no respetaría la proporcionalidad ya que solo se tiene una cuarta parte de capacidad, cincuenta toneladas.

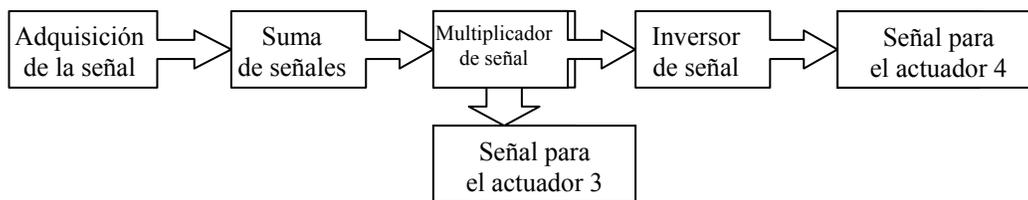
La carga total tiene que ser modificada antes de ser utilizada para el control de los actuadores verticales. Éste cambio será dado al multiplicar la señal por un factor alfa, esto con el fin de que los actuadores verticales mantengan la relación de proporcionalidad requerida con la relación de la estructura.

$$C_3 = \alpha(C_1 + C_2)$$

$$C_4 = -\alpha(C_1 + C_2)$$

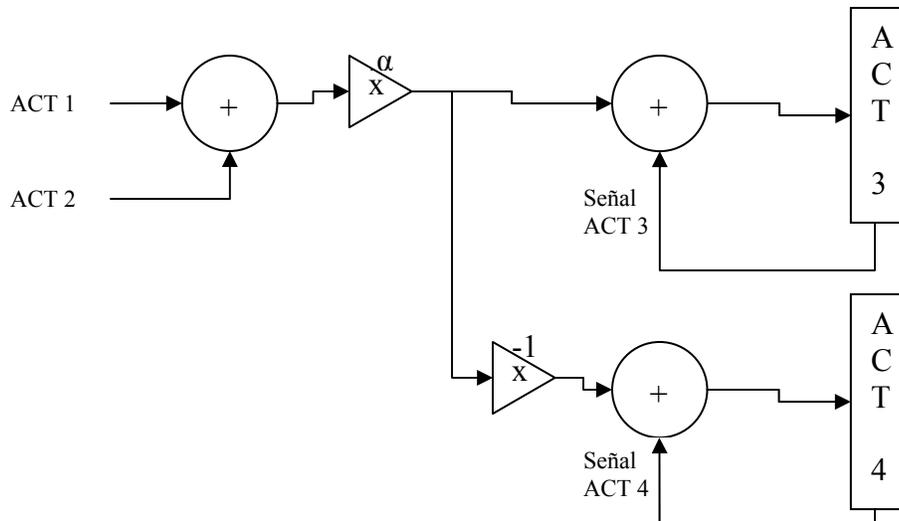
Al hacer variable el factor alfa, se puede cambiar la proporcionalidad existente entre las cargas, adecuándose a diferentes estructuras.

Por lo tanto el sistema de control de proporcionalidad de carga queda conformado de la siguiente forma.



Una vez obtenidas estas señales, son alimentadas a los actuadores en lazo cerrado, donde la realimentación son las señales de los transductores en sus respectivos casos de operación. Esto con la finalidad de alimentar al actuador con la señal de error generada entre la señal objetivo y la señal real del transductor, lo que hace que la señal de alimentación al actuador sea cero al alcanzarse el punto objetivo.

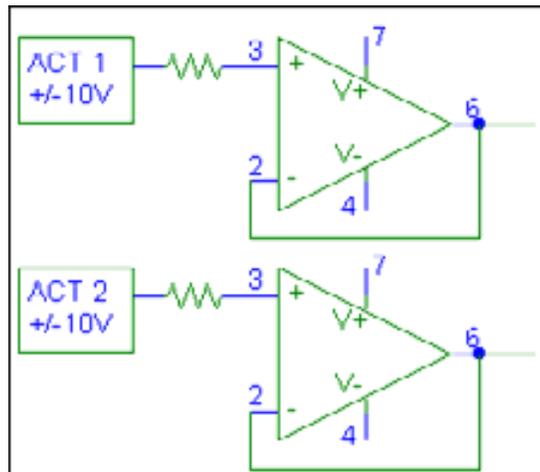
El sistema total de control queda de la siguiente forma.



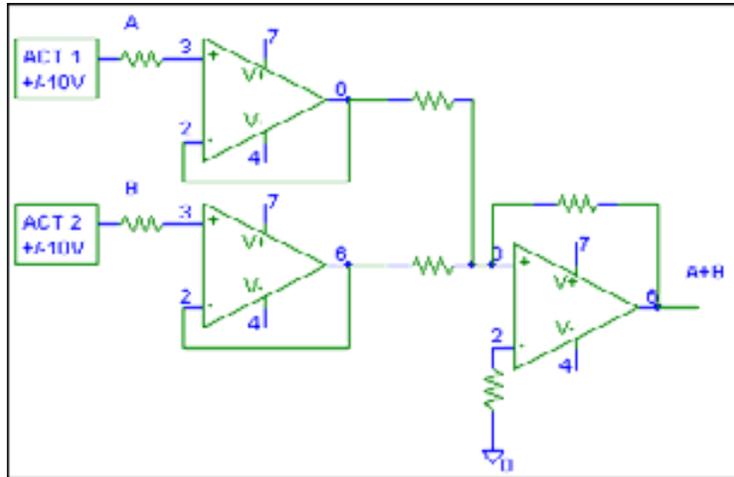
7.1 Sistema analógico

7.1.1 Diseño

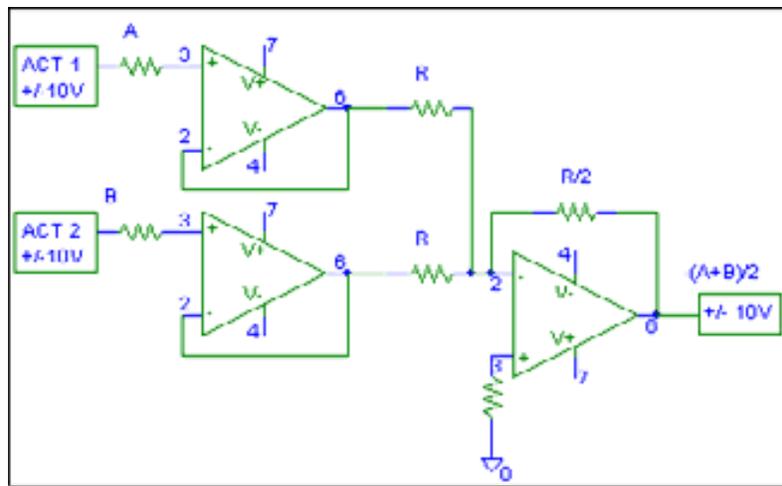
La adición entre señales de forma analógica puede ser llevada a cabo mediante el uso de un amplificador operacional (OPAM). Las señales de carga entregadas por las celdas de carga de los servo-actuadores hidráulicos horizontales, después de ser acondicionadas son introducidas a sistema que mantendrá la proporcionalidad entre las cargas, que denominaremos módulo alfa, mediante seguidores de tensión, esto con el fin de no tener pérdida alguna en las señales al acoplar las impedancias.



Después de esto las señales serán sumadas cada una con otro OPAM en una de sus configuraciones básicas. Obteniendo como resultado una señal representativa de la carga horizontal total.



La señal obtenida por el sumador tiene un valor de $\pm 20V$ cuando ambos actuadores trabajan a su máxima carga. La señal de comando para los actuadores debe ser de $\pm 10V$, por lo que utilizando la ganancia del sumador, se hace la atenuación de la señal a la mitad. Cabe notar que la señal resultante del sumador está afectada por un inversor.



Esta señal será atenuada o amplificada según sea requerido. La idea es variar el rango de operación de los actuadores verticales de cuatro veces la operación de los horizontales hasta una quinta parte; es decir:

$$4 < \frac{\text{Carga vertical}}{\text{Carga horizontal}} < 0.2$$

Los actuadores verticales deben trabajar con respecto de la máxima carga horizontal:

$$C = \alpha \frac{(A+B)}{2}$$

Y su respectivo inverso para el segundo actuador vertical.

En el primer caso:

$$4 = \frac{C_v}{C_h}; \quad C_h = \frac{C_v}{4};$$
$$C = \alpha \frac{(A+B)}{2}$$
$$50T = \alpha \frac{(6.25T + 6.25T)}{2}$$
$$10V = \alpha \frac{(0.625V + 0.625V)}{2}$$
$$\therefore \alpha = 16$$

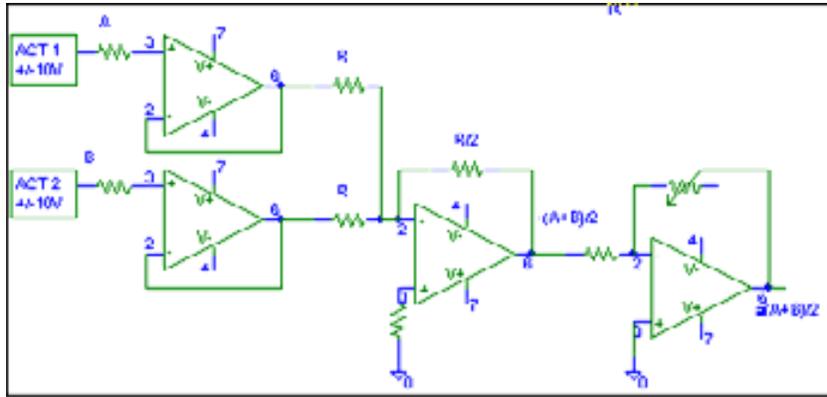
En el caso en el que se desea que la carga sea plena en ambas direcciones:

$$C = \alpha \frac{(A+B)}{2}$$
$$50T = \alpha \frac{(100T + 100T)}{2}$$
$$10V = \alpha \frac{(10V + 10V)}{2}$$
$$\therefore \alpha = 1$$

Y para el último caso:

$$0.2 = \frac{C_v}{C_h}; \quad C_h = \frac{C_v}{0.2}$$
$$C = \alpha \frac{(A+B)}{2}$$
$$40T = \alpha \frac{(100T + 100T)}{2}$$
$$8V = \alpha \frac{(10V + 10V)}{2}$$
$$\therefore \alpha = 0.8$$

Utilizando un amplificador inversor de ganancia variable, se logra obtener el factor alfa deseado.



El valor de la resistencia variable define el alfa utilizado. En un amplificador inversor la ganancia esta dada por la relación de las resistencias, así que para el caso de la mayor amplificación, $\alpha=16$, la relación de resistencias es:

$$A = -\frac{R_2}{R_1}; \text{ si } R_1 = R$$

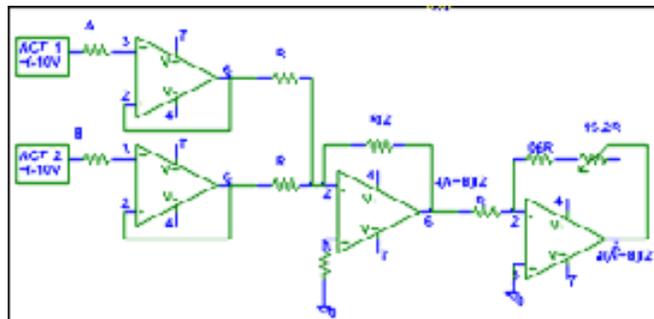
$$16 = \frac{16R}{R}$$

Y en la máxima atenuación.

$$A = -\frac{R_2}{R_1}; \text{ si } R_1 = R$$

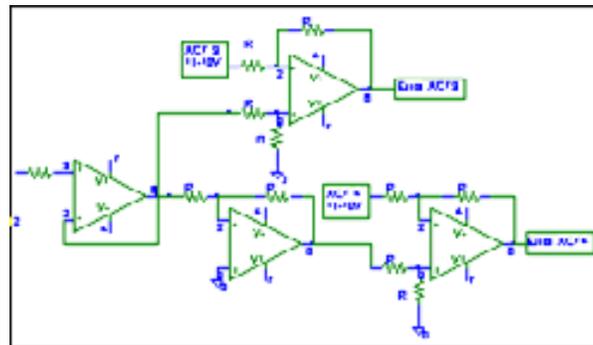
$$0.8 = \frac{0.8R}{R}$$

Por lo que se deja una resistencia fija que satisfaga la última condición y se colocará un potenciómetro el cual incremente el valor de la resistencia hasta alcanzar dieciséis veces el valor de R_1 para la amplificación.

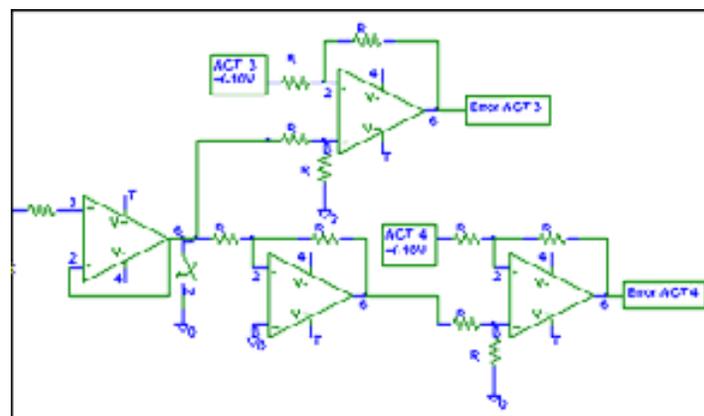


Ésta es la señal objetivo que se desea alcanzar en los gatos verticales. Esta señal es comparada con la señal proveniente de los acondicionadores de los transductores de los

gatos verticales, generando una señal de error. Para poder realizar la operación en direcciones opuestas de los actuadores verticales, a un módulo se le proporciona la señal inversa al generador de error.

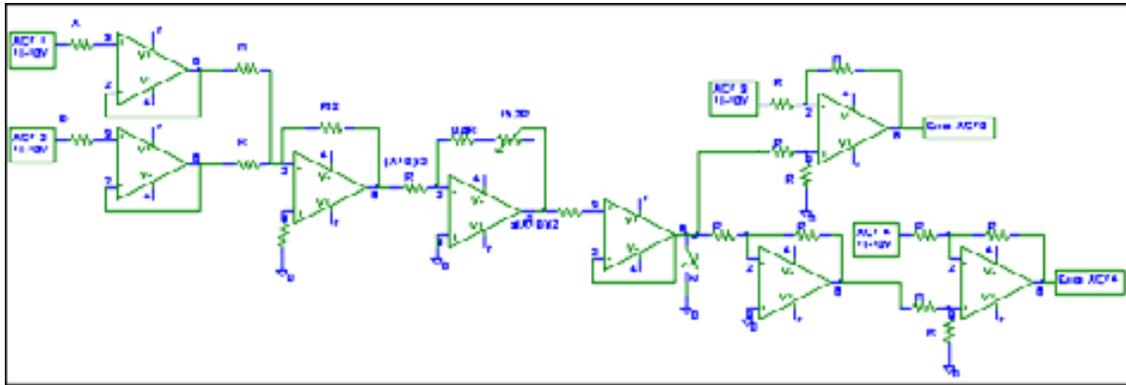


Al ser las cuatro resistencias del restador iguales, la relación de entradas salida tienen una ganancia de uno. De esta forma el circuito alfa operaría todo el tiempo, así que se agrega un interruptor conectado a tierra a la salida del amplificador operacional con ganancia α .



Cuando el interruptor esté cerrado, la señal de error generada será puramente la señal proveniente de los acondicionadores de los transductores.

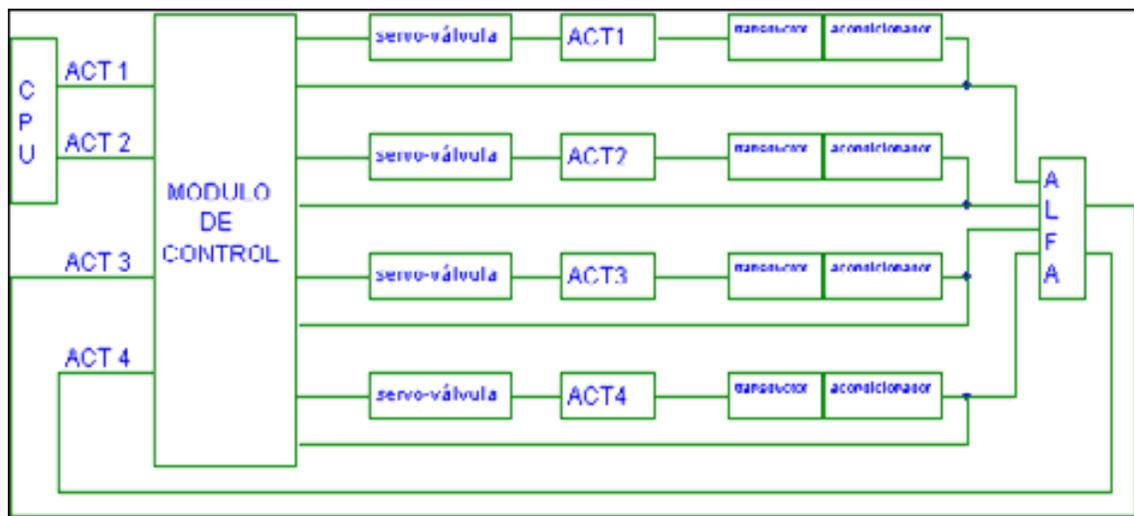
La señal proveniente de este procesamiento, ya sea la señal de error entre la señal objetivo y la señal de los transductores (interruptor abierto), o la señal de los transductores (interruptor cerrado), es alimentada al módulo de control que adiciona ésta a la señal de comando generada por el CPU.



En el caso de control de proporcionalidad, la señal de comando del CPU es cero, ya que la operación de los actuadores verticales está en total dependencia de las cargas horizontales por lo que no requiere estímulo externo alguno.

7.1.2 Diagrama del sistema analógico.

El circuito electrónico completo del acondicionador alfa, queda de la siguiente manera.



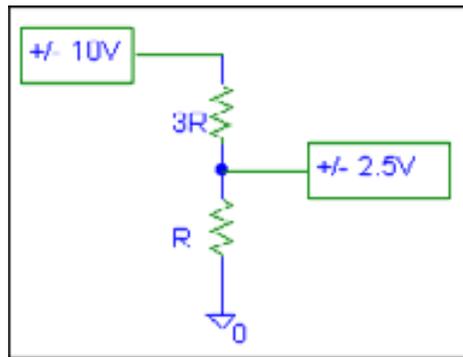
7.2 Sistema Digital

7.2.1 Diseño

Las señales entregadas por las celadas de carga son analógicas por lo que para lograr hacer un procesamiento digital de éstas, lo primero que se requiere es hacer un acondicionamiento de las señales; es decir, pasarlas a su forma digital. El PIC18F4439 cuenta con ocho canales A/D (Puertos 2-5 y 7-10) de 10 bits, las cuales pueden ser utilizadas para la adaptación de las señales de las celdas de carga.

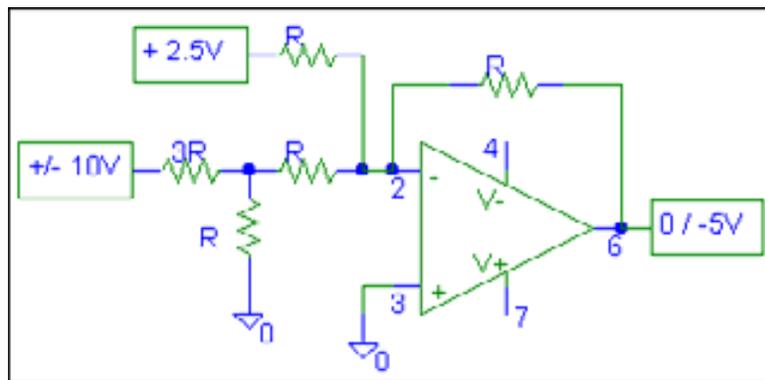
Los niveles de tensión proporcionados por los acondicionadores de las celdas de carga son de ± 10 V, mientras que el DAC del PIC trabaja de 0 a 5 V, siendo necesario acondicionar estas señales a las entradas del PIC.

La tensión se tiene que reducir a una cuarta parte, esto se hace mediante un simple divisor de tensión.

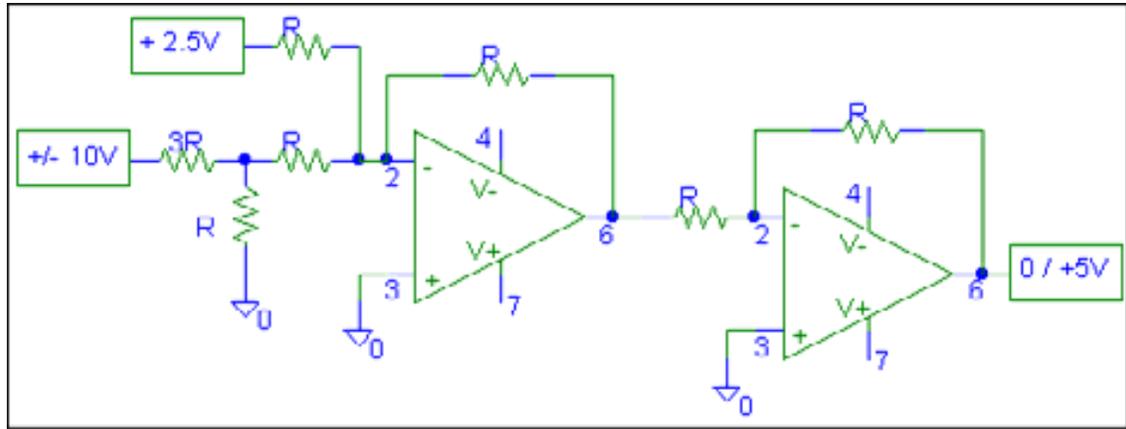


Sin embargo, el divisor entregará una señal de ± 2.5 V, así que se le tienen que sumar un nivel de directa (off set) de 2.5 V dejando una señal que opera de 0 a 5 V, lista para ser procesada por el DAC del PIC.

Se utiliza un amplificador sumador.



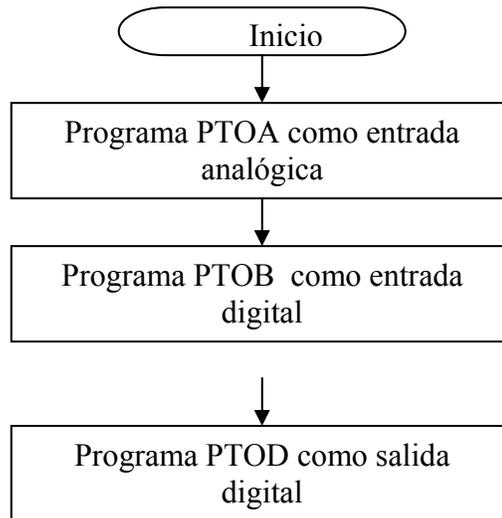
Esta señal queda invertida por la forma de operación del amplificador, para poder alimentar la señal al PIC se requiere de un amplificador inversor con ganancia unitaria.



Finalmente se obtienen una señal que puede ser procesada por el CAD del PIC. Este circuito se utilizará para cada señal analógica proveniente de los actuadores, en total cuatro, una por cada celda de carga.

7.2.1.2 Diagrama de flujo.

El primer paso de la programación del microcontrolador será la configuración de los puertos de entrada y salida. Se requieren de cuatro entradas con convertidor AD, una por cada actuador, tres entradas digitales y ocho salidas digitales.



El puerto B es configurado para obtener el valor de alfa deseado por el usuario. La relación de los actuadores está dada por la siguiente ecuación.

$$4 < \frac{\text{Carga vertical}}{\text{Carga horizontal}} < 0.2$$

Los actuadores verticales deben trabajar con respecto de la máxima carga horizontal; es decir:

$$C = \alpha \frac{(A+B)}{2}$$

El valor máximo de carga está dado por $\pm 2.5V$ después de la atenuación. Para el primer caso:

$$4 = \frac{C_v}{C_h}; \quad C_h = \frac{C_v}{4};$$

$$C = \alpha \frac{(A+B)}{2}$$

$$50T = \alpha \frac{(6.25T + 6.25T)}{2}$$

$$2.5V = \alpha \frac{(0.15625V + 0.15625V)}{2}$$

$$\therefore \alpha = 16$$

Y para el último caso:

$$0.2 = \frac{C_v}{C_h}; \quad C_h = \frac{C_v}{0.2}$$

$$C = \alpha \frac{(A+B)}{2}$$

$$40T = \alpha \frac{(100T + 100T)}{2}$$

$$2V = \alpha \frac{(2.5V + 2.5V)}{2}$$

$$\therefore \alpha = 0.8$$

Cabe señalar que las señales de entrada no solo tienen una atenuación, también un offset de $+2.5V$.

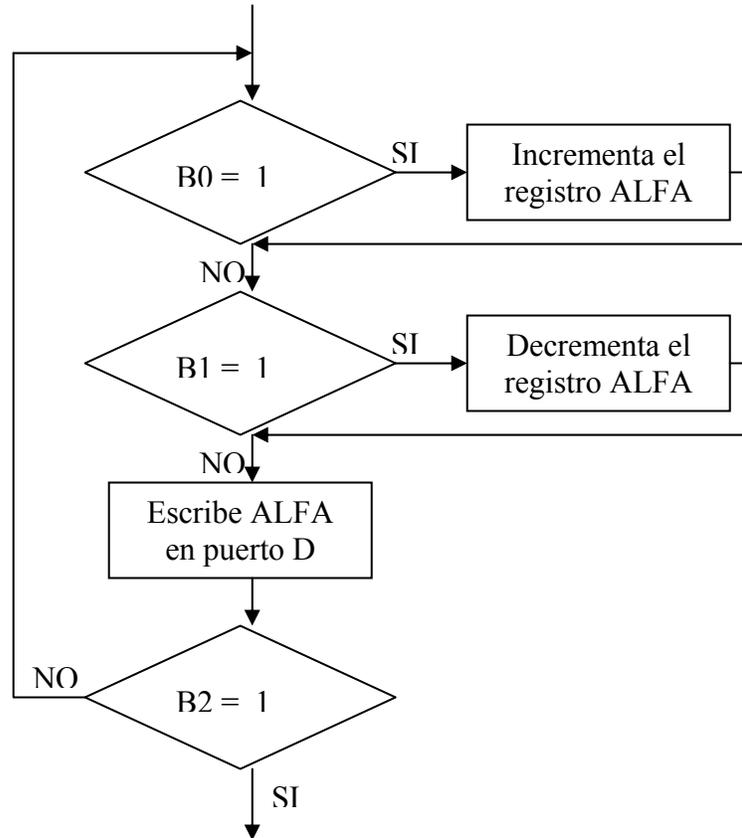
Una vez definidos los valores de alfa, se procede a determinar el número de bits que tendrá la palabra que lo represente. Para un valor de 16 se requiere de cinco bits, y para un valor de 0.8 es requerida una cadena infinita de bits, pues su representación binaria presenta una secuencia periódica. Por conveniencia de utilizar una palabra con el número de bits igual a los registros de memoria, se utilizará un formato de $Q_i = 4$, donde 4 bits representarán a la parte entera y 4 bits a la parte fraccional.

Por lo tanto la resolución es de 0.0625, y con el número de bits utilizados, los valores de alfa para el sistema digital serán:

$$0.8125 < \alpha < 15.9375$$

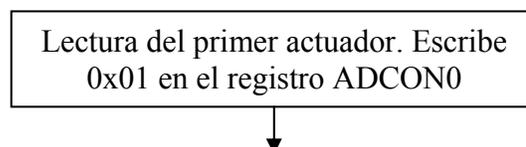
En valores de alfa múltiplos de la resolución.

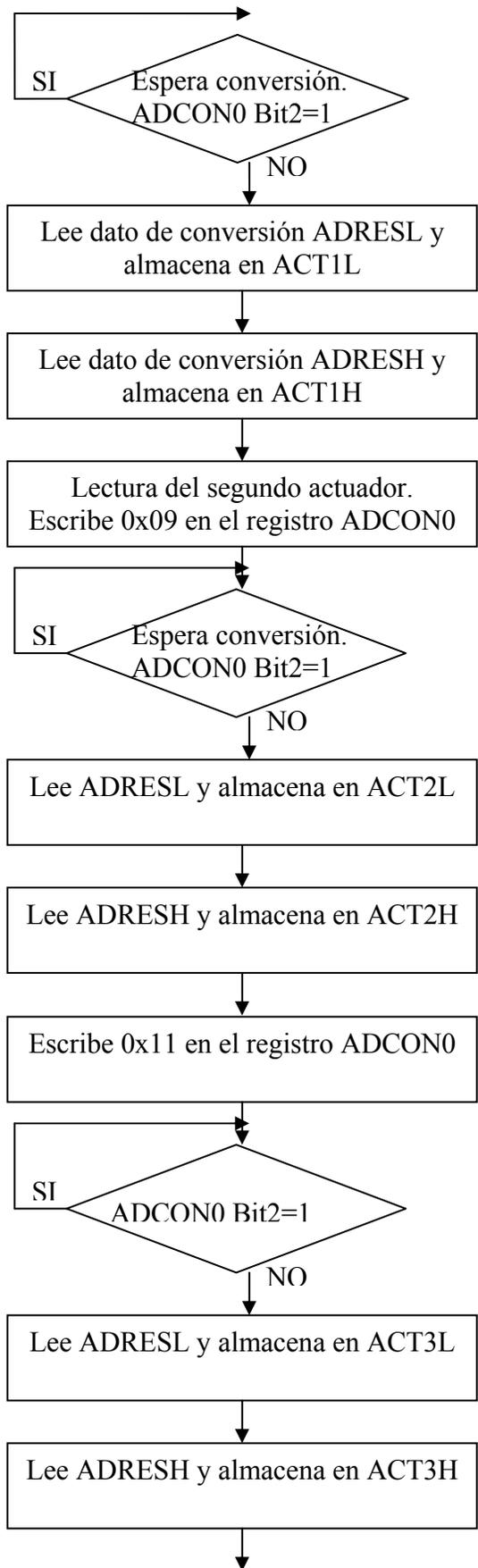
Por lo tanto alfa será almacenado en un solo espacio de memoria del microcontrolador, el cual es mostrado por el puerto C.

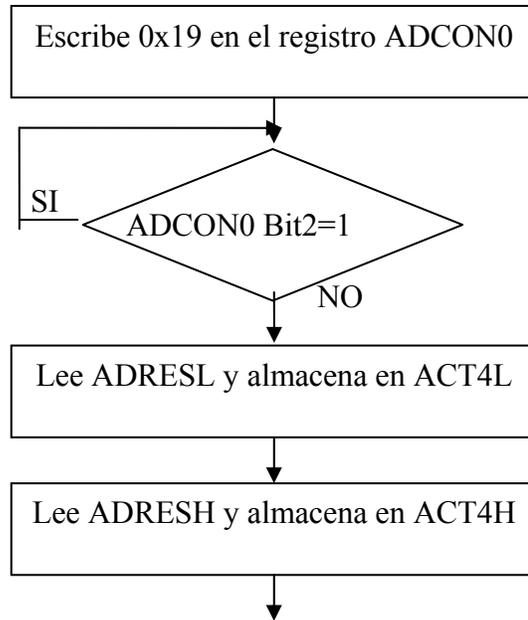


El microcontrolador cuenta con un convertidor analógico-digital de 10 bits guardando la conversión de la señal de entrada en dos espacios de memoria de 8 bits, ADRESH y ADRESL y la selección del canal en función es mediante programación (bits 3, 4 y 5 del registro ADCON0). Una vez terminada la conversión, se activa una bandera de aviso (bit 2 registro ADCON0)

Son cuatro señales analógicas las que se reciben, requiriendo asignar ocho espacios de memoria para almacenar el valor de la conversión. Estos registros serán identificados por el número de actuador y si es la parte alta o baja de la palabra, ejemplo: ACT1L es ADRESL (parte baja) de la señal digital del gato uno, la parte alta será ACT1H.

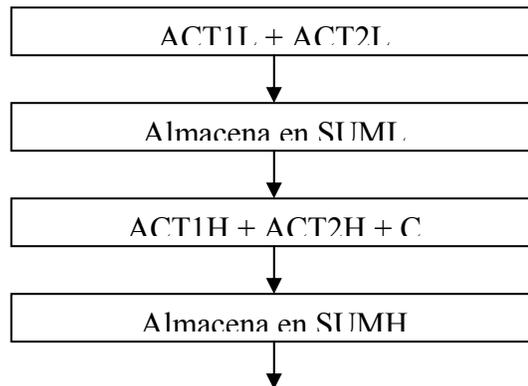






Una vez obtenidas las representaciones digitales del actuador 1 y 2, se procede a hacer una suma entre ambos. Cada señal está formada por dos palabras de 8 bits que implica la realización de la suma en partes.

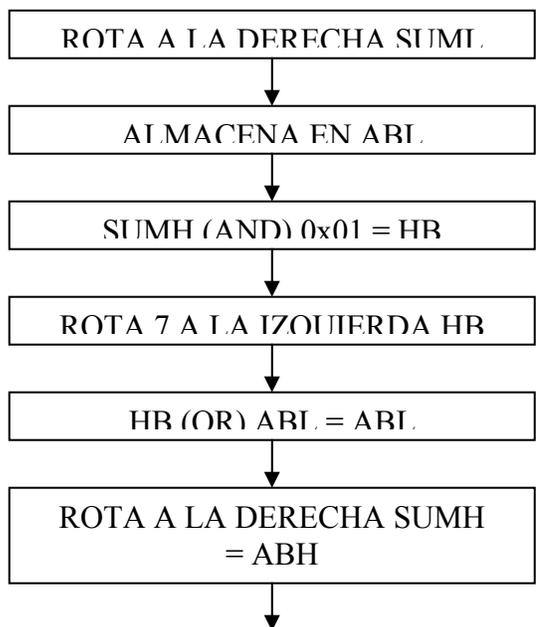
Primero se suma ACT1L con ACT2L, partes bajas, y se coloca el resultado en SUML, lo siguiente es realizar la suma entre ACT1H, ACT2H y el acarreo generado de la adición anterior, el resultante es colocado en SUMH



La suma tiene como valor máximo una palabra de 11 bits y por conveniencia de operación se desea seguir manteniendo una lógica de 10 bits, pues éste es el tamaño de las palabras para el PWM. Por lo que se hace una división entre dos del producto de la adición. La división entre dos en binario se realiza haciendo un corrimiento a la derecha de los bits de la palabra.

SUML es recorrido en un bit a la derecha y guardado en ABL, pero el bit más alto tiene que ser el bit más bajo de SUMH, por lo que se hace un AND de SUMH con 0x01 para obtener este bit y es almacenado en HB, este registro sufre un corrimiento de 7 bits a la izquierda o

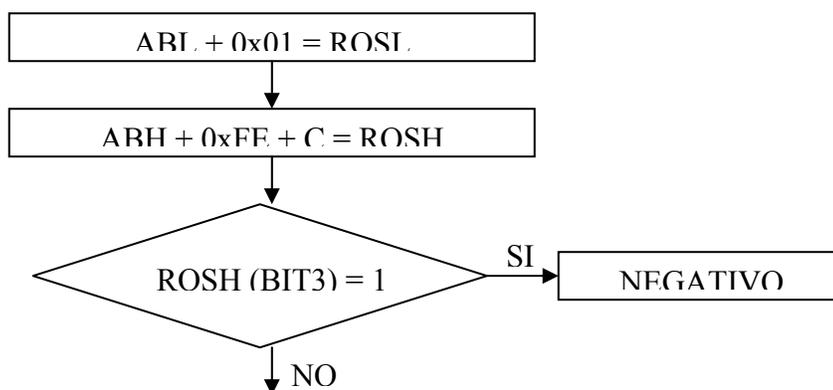
un corrimiento de 1 bit a la derecha que mande el bit más bajo a la parte alta y es almacenado en el mismo registro, HB. HB es sumado con ABL y guardado en ABL. Finalmente SUMH sufre un corrimiento a la derecha y colocado en ABH.

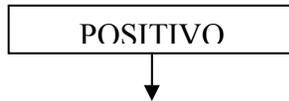


El valor total de la suma trae consigo un valor de offset de 2.5V que en su forma binaria de 10 bits es 01 1111 1111, el cual debe ser reemplazado antes de la multiplicación por alfa para poder mantener la relación entre cargas.

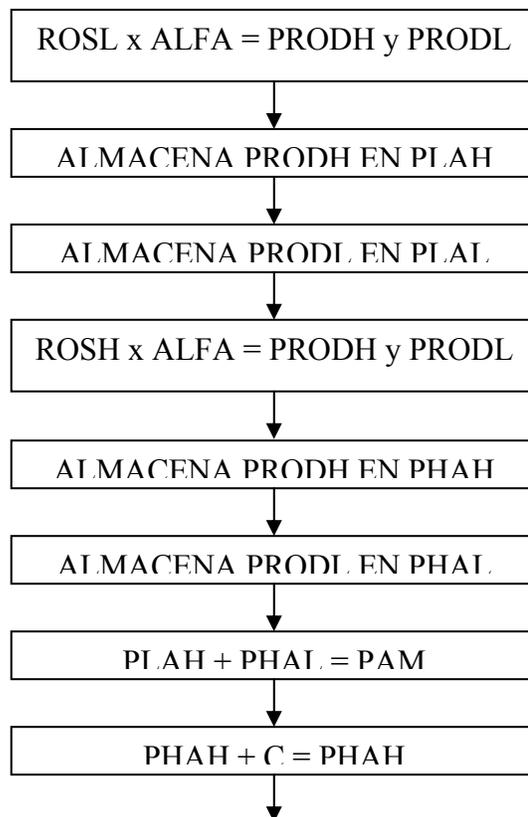
La resta con lógica de 10 bits se realiza de la siguiente forma. Al número a sustraer se le cambia por su complemento a uno y se suma. El complemento de nuestro valor es 10 0000 0001. El bit 11 (bit 3 de la palabra alta) es el bit de signo, si éste es un cero, el resultado es positivo, y en caso que éste sea un uno el resultado es negativo.

Lo primero es sumar ABL con 0x01 y guardar el resultado en ROSL, después a ABH se adiciona 0xFE y el acarreo, el resultante es puesto en ROSH. Y se obtiene el tercer bit para determinar si el resultado es positivo o negativo.

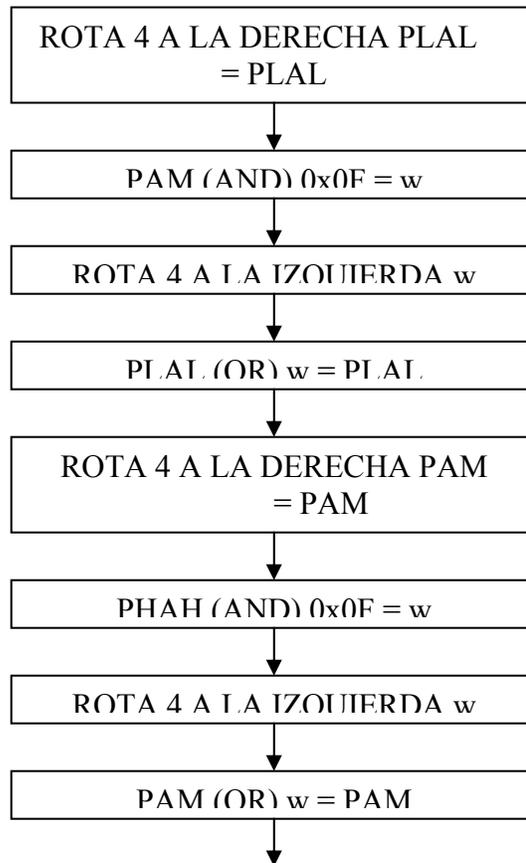




Para cuando el resultado es positivo, se procede directamente a la multiplicación por alfa. La multiplicación se hace por partes. La primera es el producto entre ROSL y alfa, el resultado se genera en los registros PRODH y PRODL, los cuales son almacenados en PLAH y PLAL respectivamente. El producto de ROSH y alfa es colocado en los espacios de memoria PHAH y PHAL. Posteriormente se suman PLAH y PHAL, poniendo el resultado en PAM, el acarreo es adicionado a PHAH y colocado en el mismo registro. El producto de un número de 10 bits y uno de 8, es una palabra de 17 bits. Ésta tendrá un formato de punto flotante igual a la suma de los formatos, Q_i 's, de los operandos. En nuestro caso es un formato de $Q_i=4$, lo que deja a los últimos 13 bits como la parte entera de la operación.



PLAL sufre un corrimiento de 4 bits a la derecha y colocado en el mismo registro, se realiza un AND con PAM y 0x0F para obtener los primeros cuatro bits, los cuales son recorridos a la izquierda cuatro veces y este nuevo número es adicionado a PLAL y guardado en el mismo registro. PAM es corrido en cuatro a la derecha y almacenado, se obtienen los primeros cuatro bits de PHAH y son sumados en la parte alta de PAM de la misma forma que se hizo con la parte baja. Esta nueva palabra es almacenada en PAM. Ahora los 13 bits de interés estarán almacenados en PAM y PLAL.

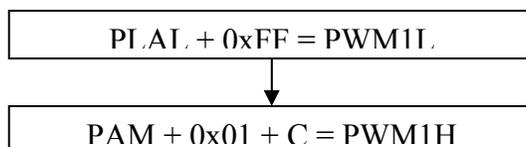


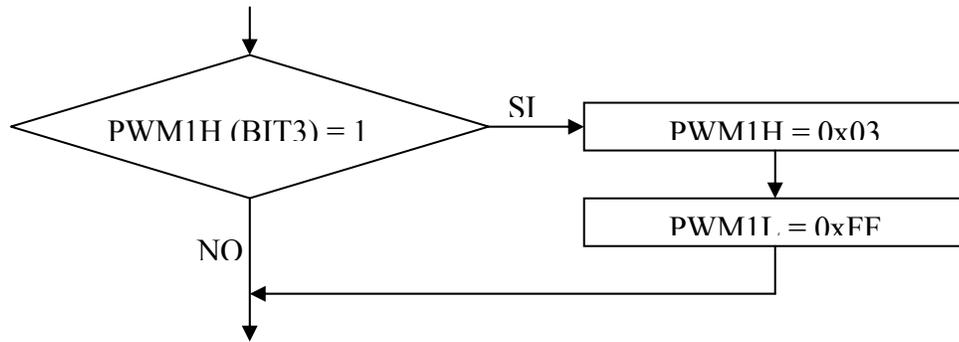
Este valor es la proporción con la que deben operar los actuadores verticales.

La salida será a través del PWM, el cual es de 10 bits. La señal de salida es una señal de amplitud 0 a 5V, convirtiendo a 2.5V como el eje de referencia de operación. La palabra de escritura en el PWM para obtener este nivel es 01 1111 1111.

Mientras para el actuador 3 la proporción será sumada a la referencia, para el actuador 4 ésta será restada. Pero cuando la suma supere los 10 bits, implica una saturación en la operación hacia el lado positivo, y si la resta es menor a cero hay una saturación hacia la parte negativa.

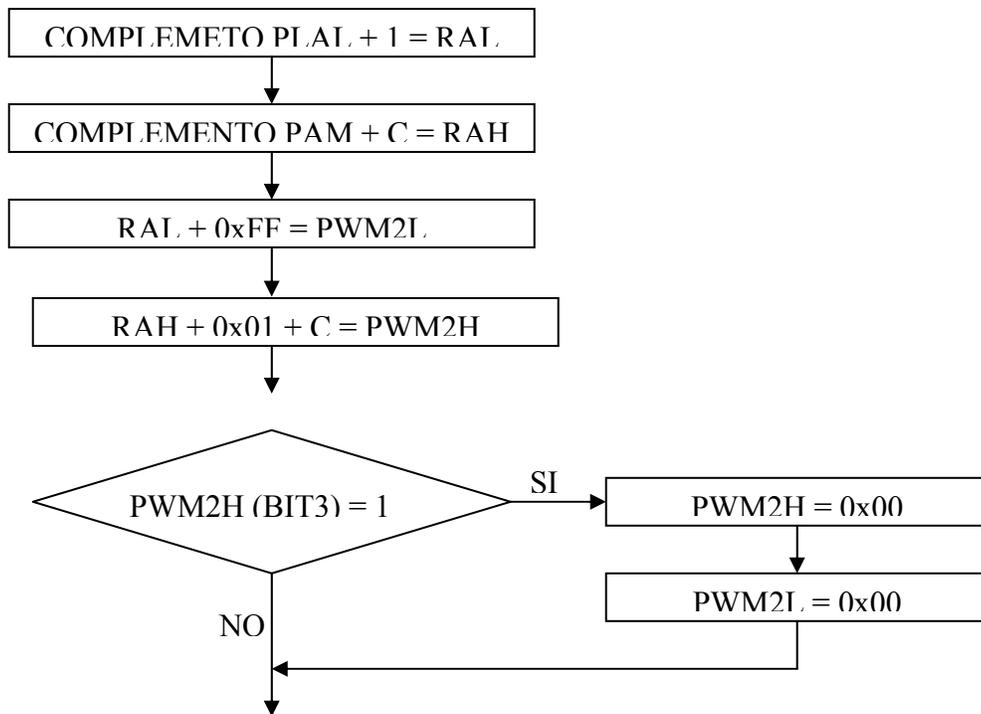
Se realiza la adición entre PLAL y 0xFF, el resultado es colocado en PWM1L. PAM es sumado con 0x01 y el acarreo de la operación anterior, y se almacena en PWM1H. En dado caso de que el tercer bit de PWM1H sea uno, significa que la palabra es mayor a 10 bits, lo que significa una saturación, de ser así, se escribirá 0xFF en PWM1L y 0x03 en PWM1H, de lo contrario se dejaron los valores ya escritos.



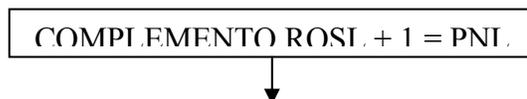


Para el actuador 4, se restará la proporción a la referencia. Primero se hace complemento a PLAL y se suma 1, guardando el resultado en RAL, igualmente se realiza complemento a PAM y se la adiciona el acarreo, colocando el producto en RAH.

Se suma 0xFF con RAL almacenando el resultante en PWM2L. El acarreo, RAH y 0x01 son sumados y guardados en PWM2H. Si el tercer bit de PWM2H es 1, significa un valor menor a cero que es una saturación negativa, por lo que se escribirá 0x00 en PWM2L y 0x00 en PWM2H. Si la resta resultara en un valor positivo se conservan los valores.



Si el tercer bit de la operación AND entre ROSH y 0x04 resultara negativo. A ROSL se le obtiene su complemento y se le suma uno, guardando el resultado en PNL, se hace complemento de ROSH y se le adiciona el acarreo almacenando lo que sale en PNH.



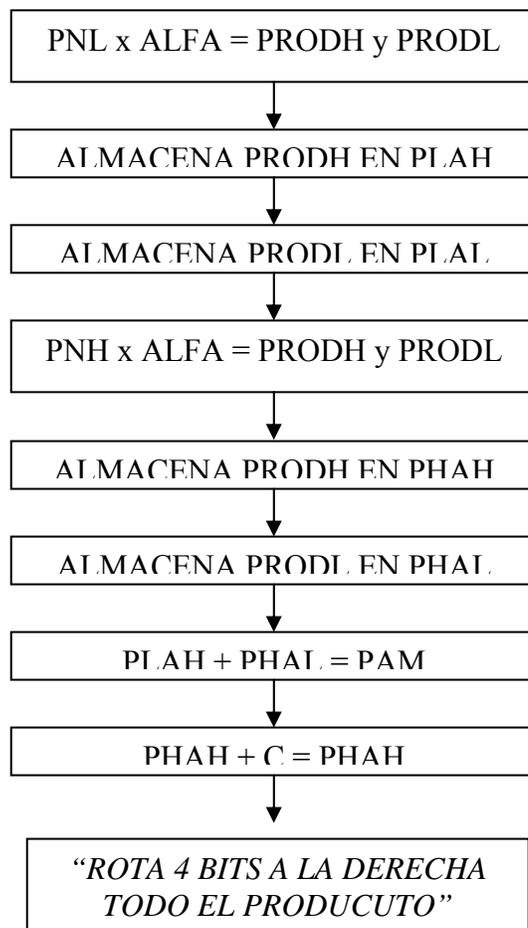
$$\text{COMPLEMENTO ROSH} + C = \text{PNH}$$



Se realiza producto entre PNL y alfa, el resultado se genera en los registros PRODH y PRODL, los cuales son almacenados en PLAH y PLAL respectivamente. El producto de PNH y alfa es colocado en los espacios de memoria PHAH y PHAL. Posteriormente se suman PLAH y PHAL, poniendo el resultado en PAM, el acarreo es adicionado a PHAH y colocado en el mismo registro

Nuevamente se hace el corrimiento de los 4 bits a la derecha dejando los 13 bits de interés estarán almacenados en PAM y PLAL.

Ahora la señal para el PWM1H y PWM1L será generada a través de la resta de la proporción a la palabra 01 1111 1111, mientras que para el PWM2H y PWM2L se realiza una suma.

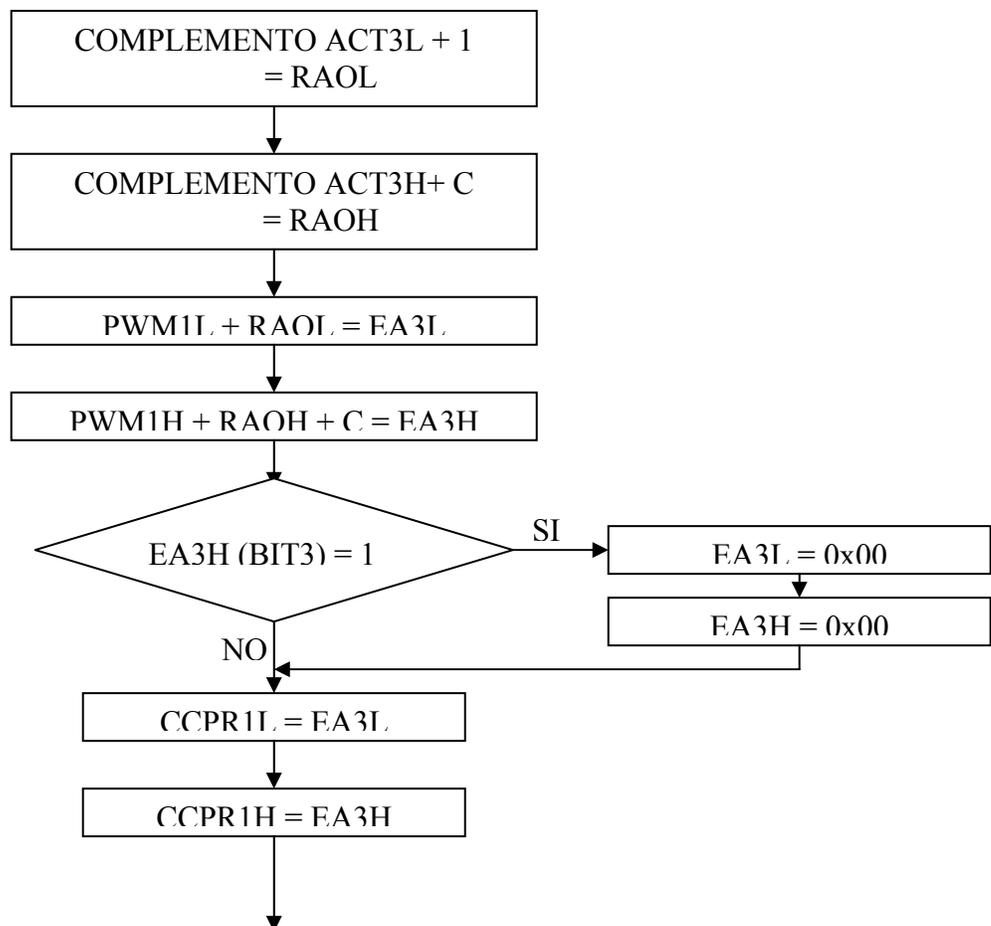


De igual forma que en el caso anterior, cuando se sobrepasen los 10 bits se escribirá la palabra 11 1111 1111, y si el valor de la resta fuera negativo un 00 0000 0000, para garantizar la saturación de los actuadores.

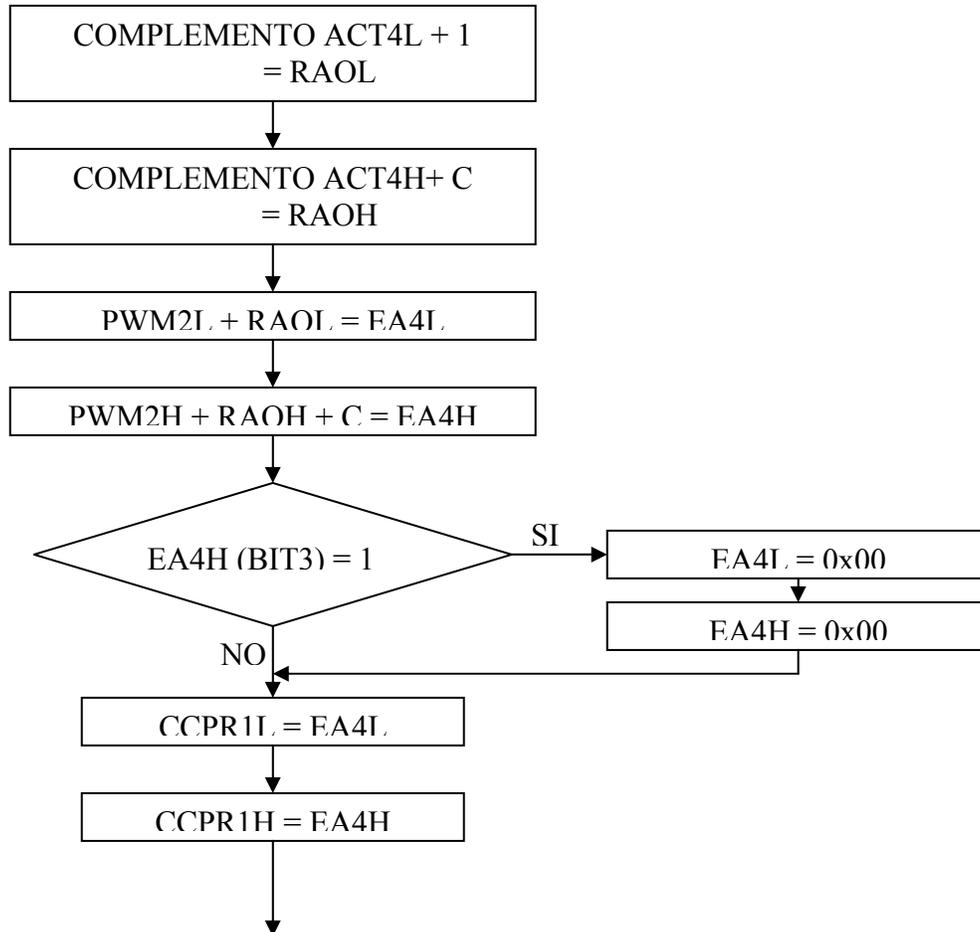
Una vez obtenidos estos valores (PWM1L y PWM1H), que es nuestra señal objetivo, se les restarán a las lecturas tomadas de ACT3L Y ACT3H para generar la palabra de error que será colocada en PWM1 para el control del actuador 3.

Se le hace un complemento de ACT3L y se suma 1, guardando el resultado en RAOL a, posteriormente se complementa ACT3H y se adiciona el acarreo de la operación anterior, guardando el resultante en RAOH.

Se suma PWM1L con RAOL y el resultado es guardado en EA3L, y después se suman el acarreo, PWM1H y RAOH, colocando lo obtenido en EA3H. El error siempre tiene que ser un valor mayor o igual a cero, con lo que cuando el bit 3 de EA3H sea 1 implicaría un valor negativo; en este caso se escribirá 0x00 en CCPR1L y en CCPR1H. Cuando el resultado de la resta arroje un valor de cero o positivo, EA3L se colocará en CCPR1L y EA3H en CCPR1H. Ésta es la señal que comanda el gato 3.

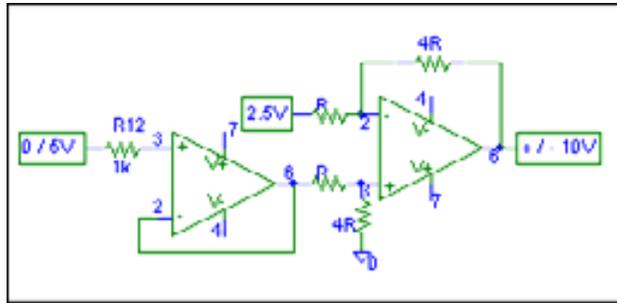


De igual forma, la señal de control para el actuador cuatro, estará generada por la señal de error proveniente de la señal objetivo (PWM2H – PWM2L) y la señal acondicionador del gato (ACT4H – ACT4L), guardando el resultado en EA4H y EA4L. También esta palabra tendrá que ser un valor mayor o igual a cero antes de ser escrita en CCPR2L y CCPR2H.

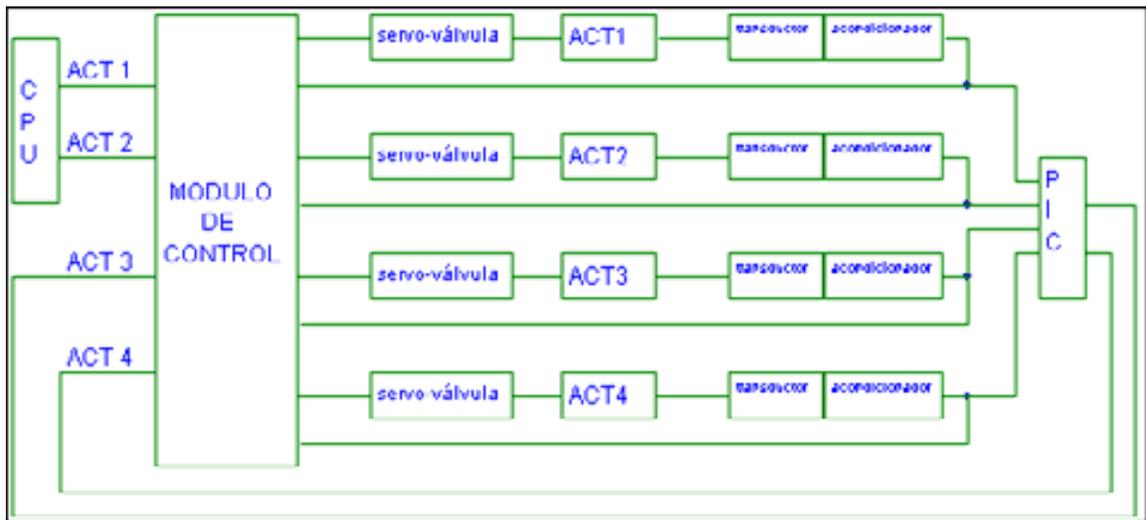


Una vez obtenida la señal del procesamiento digital, la señal tendrá que ser acondicionada para el módulo de control de los gatos, pues la señal otorgada por el PWM varía de 0 a 5V.

La señal del PWM es procesada en un seguidor para acoplar impedancias, luego mediante un restador, se le agrega un offset de $-2.5V$ y se utiliza la ganancia del amplificador para amplificar la señal en cuatro.



7.2.3 Diagrama del sistema digital.



7.2.4 Programa del microcontrolador.

```

list                p=18f4436
include "p18f4436.inc"

ALFA EQU 0X20      ;DECLARACION DE VARIABLES A UTILIZAR
ACT1L EQU 0X21
ACT1H EQU 0X22
ACT2L EQU 0X23
ACT2H EQU 0X24
ACT3L EQU 0X25
ACT3H EQU 0X26
ACT4L EQU 0X27
ACT4H EQU 0X28
SUML EQU 0X29
SUMH EQU 0X2A
ABL EQU 0X2B
ABH EQU 0X2C
ROSL EQU 0X2D
ROSH EQU 0X2E
PLAL EQU 0X2F
PLAH EQU 0X30
PHAL EQU 0X31
PHAH EQU 0X32
PAM EQU 0X33

```

```

PWM1L EQU 0X34
PWM1H EQU 0X35
PWM2L EQU 0X36
PWM2H EQU 0X37
TEMP EQU 0X38
RAL EQU 0X39
RAH EQU 0X3A
PNL EQU 0X3B
PNH EQU 0X3C
RAOL EQU 0X3D
RAOH EQU 0X3E
EA3L EQU 0X3F
EA3H EQU 0X40
EA4L EQU 0X41
EA4H EQU 0X42

ORG 0
MOVLW 0X80 ;ACTIVA TODAS LAS
MOVWF ADCON1 ;ENTRADAS ANALOGICAS
MOVLW 0XFF ;ACTIVA PUERTO B
MOVWF TRISB ;COMO ENTRADA
MOVLW 0X00 ;ACTIVA PUERTO D
MOVWF TRISD ;COMO SALIDA

MOVLW 0X00 ;PONE ALFA
MIVWF ALFA ;A CERO

INICIO
BTFS PORTB,0 ;VERIFICA BIT 0 DEL PUERTO B
INCF ALFA ;SI ES IGUAL A 1 INCREMENTA ALFA
BTFS PORTB,1 ;VERIFICA BIT 1 DEL PUERTO B
DECF ALFA ;SI EL BIT ES DECREMENTAR ALFA
MOVWF ALFA ;MUESTRA A ALFA
MOVWF PORTD ;POR EL PUERTO D
BTFS PORTB,2 ;VERIFICA EL ENTER
GOTO INICIO ;PARA PROSEGUIR CON EL PROGRAMA

ACT1
MOVLW 0X01 ;SELECCIONA EL CANAL 1
MOVWF ADCON0 ;DEL CONVERTIDOR AD
BTFS ADCON0,2 ;SE MANTIENEN HASTA
GOTO ACT1 ;QUE FINALICE LA CONVERSION
MOVWF ADRESL ;GUARDA EL REGISTRO BAJO
MOVWF ACT1L ;DEL ACTUADOR 1
MOVWF ARDESH ;GUARDA EL REGISTRO ALTO
MOVWF ACT1H ;DEL ACTUADOR 1
MOVLW 0X09 ;SELECCIONA EL CANAL 2
MOVWF ADCON0 ;DEL CONVERTIDOR AD
ACT2
BTFS ADCON0,2 ;SE MANTIENEN HASTA
GOTO ACT2 ;QUE FINALICE LA CONVERSION
MOVWF ADRESL ;GUARDA EL REGISTRO BAJO
MOVWF ACT2L ;DEL ACTUADOR 2
MOVWF ARDESH ;GUARDA EL REGISTRO ALTO
MOVWF ACT2H ;DEL ACTUADOR 2
MOVLW 0X11 ;SELECCIONA EL CANAL 3
MOVWF ADCON0 ;DEL CONVERTIDOR AD
ACT3
BTFS ADCON0,2 ;SE MANTIENEN HASTA
GOTO ACT3 ;QUE FINALICE LA CONVERSION
MOVWF ADRESL ;GUARDA EL REGISTRO BAJO
MOVWF ACT3L ;DEL ACTUADOR 3
MOVWF ARDES ;GUARDA EL REGISTRO ALTO
MOVWF ACT3H ;DEL ACTUADOR 3
MOVLW 0X19 ;SELECCIONA EL CANAL 4
MOVWF ADCON0 ;DEL CONVERTIDOR AD
ACT4
BTFS ADCON0,2 ;SE MANTIENEN HASTA
GOTO ACT4 ;QUE FINALICE LA CONVERSION
MOVWF ADRESL ;GUARDA EL REGISTRO BAJO
MOVWF ACT4L ;DEL ACTUADOR 4
MOVWF ARDES ;GUARDA EL REGISTRO ALTO

```

MOVWF	ACT4H	;DEL ACTUADOR 4
MOVFW	ACT1L	;SUMA LAS PARTES BAJAS
ADDWF	ACT2L	;DE LOS ACTUADORES 1 Y 2
MOVWF	SUML	;GUARDANDO EL RESULTADO
MOVFW	ACT1H	;SUMA LAS PARTES ALTAS Y EL ACARREO
ADDWFC	ACT2H	;DE LOS ACTUADORES 1 Y 2
MOVWF	SUMH	;Y GUARDA EL RESULTADO
BCF	STATUS,C	;LIMPIA EL ACARREO
MOVFW	SUML	;ROTA 1 A LA DERECHA
RRCF		;LA PARTE BAJA DE LA SUMA
MOVWF	ABL	;GUARDANDO EL RESULTADO
BCF	STATUS,C	;LIMPIA EL ACARREO
MOVFW	SUMH	;SE OBTIENE EL BIT
ANDLW	OX01	;MENOS SIGNIFICATIVO DE SUMH
RRSCF		;Y LO COLOCA
IORWF	ABL	;COMO EL BIT MAS SIGNIFICATIVO
MOVWF	ABL	;DE SUML
MOVFW	SUMH	;ROTA 1 A LA DERECHA
RRCF		;LA PARTE ALTA DE LA SUMA
MOVWF	ABH	;Y LO GUARDA
BCF	STATUS,C	;LIMPIA EL ACARREO
MOVFW	OX01	;SE RESTA EL OFFSET
ADDWF	ABL	;A LA PARTE BAJA
MOVWF	ROSL	;Y LO GUARDA
MOVFW	OXFE	;SE RESTA EL OFFSET
ADDWFC	ABH	;A LA PARTE ALTA
MOVWF	ROSH	;Y LO GUARDA
BTFS	ROSH,3	;VERIFICA SI LA RESTA ES
GOTO	NEGA	;NEGATIVA
		;O POSITIVA
MOVFW	ALFA	;SE MULTIPLICA ALFA
MULWF	ROSL	;POR LA PARTE BAJA SOF (SIN OFFSET)
MUVFW	PRODH	;GUARDA LA PARTE ALTA
MOVWF	PLAH	;DEL PRODUCTO
MOVFW	PRODL	;GUARDA LA PARTE BAJA
MOVWF	PLAL	;DEL PRODUCTO
MOVFW	ALFA	;MULTIPLICA ALFA
MULWF	ROSH	;POR LA PARTE ALTA SOF
MUVFW	PRODH	;GUARDA LA PARTE ALTA
MOVWF	PHAH	;DEL PRODUCTO
MOVFW	PRODL	;GUARDA LA PARTE BAJA
MOVWF	PHAL	;DEL PRODUCTO
ADDWF	PLAH	;SE SUMA LA PARTE ALTA DEL PRIMER PRODUCTO
MOVWF	PAM	;Y LA PARTE BAJA DEL SEGUNDO
MOVFW	OX00	;EL ACARREO SE
ADDWFC	PHAH	;SE SUMA A LA PARTE ALTA
MOVWF	PHAH	;Y SE GUARAD EN EL MISMO REGISTRO
BCF	STATUS,C	;PONE A CERO EL ACARREO
MOVWF	PLAL	;ROTA 4 A LA DERECHA
RRCF		;EL REGISTRO BAJO
BCF	STATUS,C	;
RRCF		;SEGUNDA ROTACION
BCF	STATUS,C	;
RRCF		;TERCER ROTACION
BCF	STATUS,C	;
RRCF		;CUARTA ROTACION
MOVWF	PLAL	;Y LO GUARDA EN EL MISMO REGISTRO
MOVLW	OX0F	;SE OBTIENEN LOS 4 BITS MENOS SIGNIFICATIVOS
ANDWF	PAM	;DEL REGISTRO MEDIO DEL PRODUCTO
RRSCF		;LOS COLOCA COMO LOS
RRSCF		;MAS SIGNIFICATIVOS
RRSCF		;ANTES DE ADICIONARLOS

	RRSCF		;AL REGISTRO BAJO
	IORWF	PLAL	;GUARDANDO EL RESULTADO
	MOVWF	PLAL	;EN EL MISMO REGISTRO
	BCF	STATUS,C	;PONE A CERO EL ACARREO
	MOVWF	PAM	;ROTA 4 A LA DERECHA
	RRCF		;EL REGISTRO MEDIO
	BCF	STATUS,C	;
	RRCF		;SEGUNDA ROTACION
	BCF	STATUS,C	;
	RRCF		;TERCER ROTACION
	BCF	STATUS,C	;
	RRCF		;CUARTA ROTACION
	MOVWF	PAM	;Y LO GUARDA EN EL MISMO REGISTRO
	MOVLW	OXOF	;SE OBTIENEN LOS 4 BITS MENOS SIGNIFICATIVOS
	ANDWF	PHAH	;DEL REGISTRO ALTO DEL PRODUCTO
	RRSCF		;LOS COLOCA COMO LOS
	RRSCF		;MAS SIGNIFICATIVOS
	RRSCF		;ANTES DE ADICIONARLOS
	RRSCF		;AL REGISTRO MEDIO
	IORWF	PAM	;GUARDANDO EL RESULTADO
	MOVWF	PAM	;EN EL MISMO REGISTRO
	BCF	STATUS,C	;LIMPIA EL ACARREO
	MOVLW	OXFF	;SE SUMA LA PROPORCION
	ADDWF	PLAL	;A LA PARTE BAJA
	MOVWF	PWM1L	;Y SE GUARDA
	MOVLW	OX01	;SE SUMA LA PROPORCION
	ADDWFC	PAM	;A LA PARTE MEDIA, AHORA ALTA
	MOVWF	PWM1H	;Y SE ALMACENA
	BTFS	PWM1H,3	;SE VERIFICA SI LA PALABRA
	CALL	SATS	;EXCEDIÓ LOS 11 BITS
	COMF	PLAL	;SE HACE NEGATIVA
	ADDLW	OX01	;LA PROPORCION
	MOVWF	RAL	;PARA PODER RESTARTARLA
	COMF	PAM	;
	MOVWF	TEMP	;
	MOVLW	OX00	;
	ADDWFC	TEMP	;
	MOVWF	RAH	;ALMACENA LA PARTE ALTA
	MOVLW	OXFF	;RESTA LA PROPORCION
	ADDWF	RAL	;A LA PARTE BAJA
	MOVWF	PWM2L	;Y LO GUARDA
	MOVLW	OX01	;RESTA LA PROPORCION
	ADDWFC	RAH	;A LA PARTE ALTA
	MOVWF	PWM2H	;Y LO ALMACENA
	BTFS	PWM2H,3	;REvisa SI LA RESTA ES
	CALL	SATI	;NEGATIVA
ERROR	COMF	ACT3L	;SE HACE NEGATIVA LA SEÑAL
	ADDLW	OX01	;DEL ACTUADOR PARA PODER
	MOVWF	RAOL	;RESTARTARLA A LA SEÑAL OBJETIVO
	COMF	ACT3H	;ALMACENANDO LA PARTE ALTA
	MOVWF	TEMP	;Y BAJA
	MOVLW	OX00	;
	ADDWFC	TEMP	;
	MOVWF	RAOH	;
	MOVWF	PWMIL	;SE RESTA LA PARTE BAJA
	ADDWF	RAOL	;DE LA SEÑAL OBJETIVO Y EL ACTUADOR
	MOVWF	EA3L	;Y SE GUARDA EL RESULTADO
	MOVWF	PWMIH	;SE RESTA LA PARTE ALTA
	ADDWFC	RAOH	;DE LA SEÑAL OBJETIVO Y EL ACTUADOR
	MOVWF	EA3H	;Y SE GUARDA EL RESULTADO
	BTFS	EA3H,3	;VERIFICA SI LA RESTA ES NEGATIVA

	CALL	ERN3	
	MOVFW	EA3L	;COLOCA LA PARTE BAJA DE LA SEÑAL DE ERROR
	MOVWF	CCPR1L	;EN LA PARTE BAJA DEL PWM1
	MOVFW	EA3H	;COLOCA LA PARTE ALTA DE LA SEÑAL DE ERROR
	MOVWF	CCPR1H	;EN LA PARTE ALTA DEL PWM1
	COMF	ACT4L	;SE HACE NEGATIVA LA SEÑAL
	ADDLW	0X01	;DEL ACTUADOR 4 PARA PODER
	MOVWF	RAOL	;RETARLA A LA SEÑAL OBJETIVO
	COMF	ACT4H	;ALMACENANDO LA PARTE ALTA
	MOVWF	TEMP	;Y BAJA
	MOVLW	0X00	;
	ADDWFC	TEMP	;
	MOVWF	RAOH	;
	MOVFW	PWM2L	;SE RESTA LA PARTE BAJA
	ADDWF	RAOL	;DE LA SEÑAL OBJETIVO Y EL ACTUADOR
	MOVWF	EA4L	;Y SE GUARDA EL RESULTADO
	MOVFW	PWM2H	;SE RESTA LA PARTE ALTA
	ADDWFC	RAOH	;DE LA SEÑAL OBJETIVO Y EL ACTUADOR
	MOVWF	EA4H	;Y SE GURADA EL RESULTADO
	BTFSC	EA4H,3	;VERIFICA SI LA RESTA ES NEGATIVA
	CALL	ERN4	
	MOVFW	EA4L	;COLOCA LA PARTE BAJA DE LA SEÑAL DE ERROR
	MOVWF	CCPR2L	;EN LA PARTE BAJA DEL PWM2
	MOVFW	EA4H	;COLOCA LA PARTE ALTA DE LA SEÑAL DE ERROR
	MOVWF	CCPR2H	;EN LA PARTE ALTA DEL PWM2
	GOTO	INICIO	;REGRESA A HASTA LA LECTURA DE LOS ;ACTUADORES
ERN3	CLRF	EA3L	;PONE EN CERO LA SEÑAL
	CLRF	EA3H	;DE ERROR
	RETURN		
ERN4	CLRF	EA4L	;PONE EN CERO LA SEÑAL
	CLRF	EA4H	;DE ERROR
	RETURN		
SATS	MOVLW	0X03	;COLOCA EL VALOR
	MOVWF	PWM1H	;MÁXIMO DE SATURACIÓN
	MOVLW	0XFF	;COMO VALOR
	MOVWF	PWM1L	;OBJETIVO
	RETURN		
SATI	MOVLW	0X00	;COLOCA EL VALOR
	MOVFW	PWM2H	;MINIMO DE SATURACIÓN
	MOVFW	PWM2L	;COMO VALOR OBJETIVO
	RETURN		
NEGA	COMF	ROSL	;SE HACE POSITIVO EL VALOR
	ADDLW	0X01	;DE LA PROPORCION PARA PODER MANIPULARLO
	MOVWF	PNL	;GUARDANDO ESTE VALOR
	COMF	ROSH	;LA PARTE SUPERIOR
	MOVWF	TEMP	;TAMBIEN TIENE QUE SER
	MOVLW	0X00	;CAMBIADA POR SU
	ADDWFC	TEMP	;REPRESENTACION POSITIVA
	MOVWF	PNH	;Y ALMACENADA
	MOVFW	ALFA	;SE MULTIPLICA ALFA
	MULWF	PNL	;POR LA PARTE BAJA DE ESTA PROPORCION
	MUVFW	PRODH	;GUARDA LA PARTE ALTA
	MOVWF	PLAH	;DEL PRODUCTO

	MOVFW	PRODL	;GUARDA LA PARTE BAJA
	MOVWF	PLAL	;DEL PRODUCTO
	MOVFW	ALFA	;MULTIPLICA ALFA
	MULWF	PNH	;POR LA PARTE ALTA
	MUVFW	PRODH	;GUARDA LA PARTE ALTA
	MOVWF	PHAH	;DEL PRODUCTO
	MOVFW	PRODL	;GUARDA LA PARTE BAJA
	MOVWF	PHAL	;DEL PRODUCTO
	ADDWF	PLAH	;SE SUMA LA PARTE ALTA DEL PRIMER PRODUCTO
	MOVWF	PAM	;Y LA PARTE BAJA DEL SEGUNDO
MOVFW	OX00		;EL ACARREO SE
	ADDWFC	PHAH	;SE SUMA A LA PARTE ALTA
	MOVWF	PHAH	;Y SE GUARAD EN EL MISMO REGISTRO
	BCF	STATUS,C	;PONE A CERO EL ACARREO
	MOVWF	PLAL	;ROTA 4 A LA DERECHA
	RRCF		;EL REGISTRO BAJO
	BCF	STATUS,C	;
	RRCF		;SEGUNDA ROTACION
	BCF	STATUS,C	;
	RRCF		;TERCER ROTACION
	BCF	STATUS,C	;
	RRCF		;CUARTA ROTACION
	MOVWF	PLAL	;Y LO GUARDA EN EL MISMO REGISTRO
	MOVLW	OX0F	;SE OBTIENEN LOS 4 BITS MENOS SIGNIFICATIVOS
	ANDWF	PAM	;DEL REGISTRO MEDIO DEL PRODUCTO
	RRSCF		;LOS COLOCA COMO LOS
	RRSCF		;MAS SIGNIFICATIVOS
	RRSCF		;ANTES DE ADICIONARLOS
	RRSCF		;AL REGISTRO BAJO
	IORWF	PLAL	;GUARDANDO EL RESULTADO
	MOVWF	PLAL	;EN EL MISMO REGISTRO
	BCF	STATUS,C	;PONE A CERO EL ACARREO
	MOVWF	PAM	;ROTA 4 A LA DERECHA
	RRCF		;EL REGISTRO MEDIO
	BCF	STATUS,C	;
	RRCF		;SEGUNDA ROTACION
	BCF	STATUS,C	;
	RRCF		;TERCER ROTACION
	BCF	STATUS,C	;
	RRCF		;CUARTA ROTACION
	MOVWF	PAM	;Y LO GUARDA EN EL MISMO REGISTRO
	MOVLW	OX0F	;SE OBTIENEN LOS 4 BITS MENOS SIGNIFICATIVOS
	ANDWF	PHAH	;DEL REGISTRO ALTO DEL PRODUCTO
	RRSCF		;LOS COLOCA COMO LOS
	RRSCF		;MAS SIGNIFICATIVOS
	RRSCF		;ANTES DE ADICIONARLOS
	RRSCF		;AL REGISTRO MEDIO
	IORWF	PAM	;GUARDANDO EL RESULTADO
	MOVWF	PAM	;EN EL MISMO REGISTRO
	BCF	STATUS,C	;LIMPIA EL ACARREO
	MOVLW	OXFF	;SE SUMA LA PROPORCION
	ADDWF	PLAL	;A LA PARTE BAJA
	MOVWF	PWM2L	;Y SE GUARDA PARA EL SEGUNDO ACTUADOR (4)
	MOVLW	OX01	;SE SUMA LA PROPORCION
	ADDWFC	PAM	;A LA PARTE MEDIA, AHORA ALTA
	MOVWF	PWM2H	;Y SE ALMACENA AHORA PARA EL SEGUNDO
	BTFS	PWM2H,3	;SE VERIFICA SI LA PALABRA
	CALL	SATS	;EXCEDIÓ LOS 11 BITS
	COMF	PLAL	;SE HACE NEGATIVA
	ADDLW	OX01	;LA PROPORCION
	MOVWF	RAL	;PARA PODER RESTARLA
	COMF	PAM	;
	MOVWF	TEMP	;
	MOVLW	OX00	;

```

ADDWFC TEMP          ;
MOVWF RAH            ;ALAMCENA LA PARTE ALTA
MOVLW 0xFF           ;RESTA LA PROPORCION
ADDWF RAL            ;A LA PARTE BAJA
MOVWF PWM1L          ;Y LO GUARDA PARA EL ACTUADOR 3
MOVLW 0x01           ;RESTA LA PROPORCION
ADDWFC RAH           ;A LA PARTE ALTA
MOVWF PWM1H          ;Y LO ALMACENA PARA EL ACTUADOR 3
BTFSK PWM1H,3        ;REVISA SI LA RESTA ES
CALL SATI            ;NEGATIVA

GOTO ERROR           ;REGRESA A DONDE SE GENERA LA SEÑAL DE ERROR

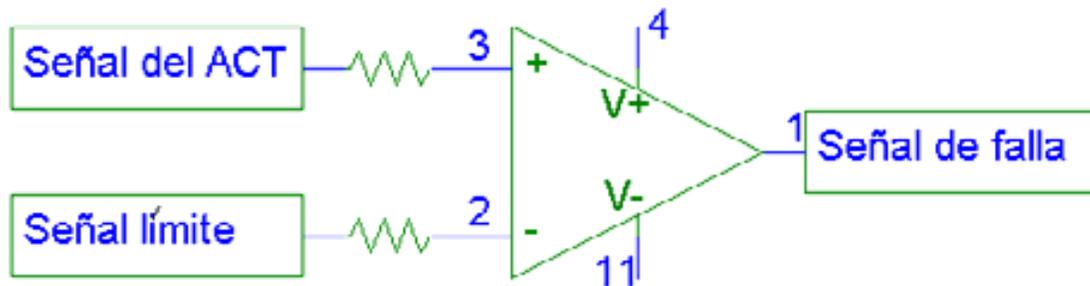
```

8. DISPOSITIVO DE DETECCIÓN DE LÍMITES EN OPERACIÓN.

Por razones de seguridad se desea que en el sistema ATS del Instituto de Ingeniería, exista un detector de los límites de operación tanto en carga como desplazamiento.

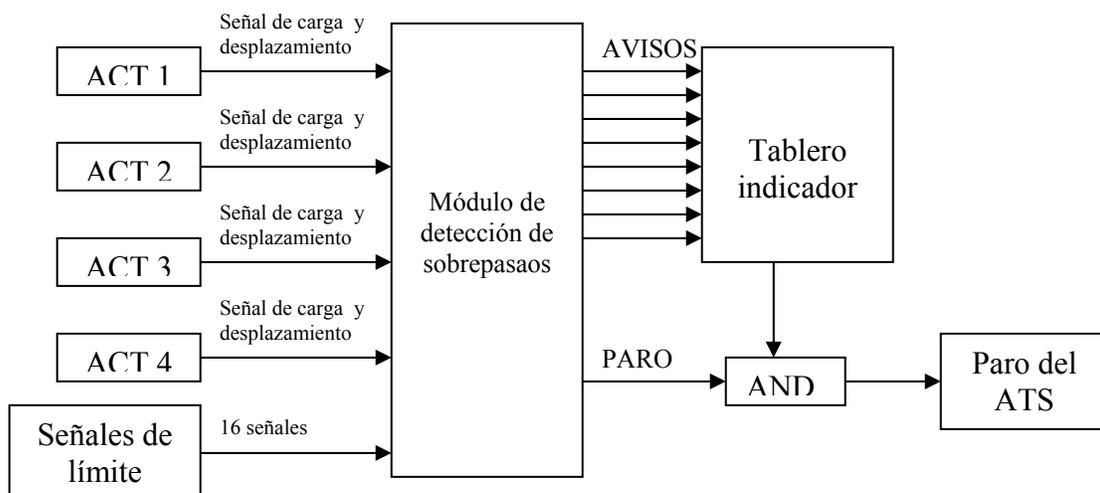
El ensayo realizado, pretende llevar a la estructura de concreto a punto de falla, por lo que el experimento puede considerarse de riesgo para el personal que realiza la prueba y para el equipo que hace fallar la estructura. Un detector de límites de operación es implementado como medida de seguridad, avisando al usuario cuando el sistema supera los valores de carga y desplazamiento establecidos para la prueba, y realizando un paro de la prueba si el operador así lo desea.

Los actuadores del sistema ATS cuentan con celdas de carga y LVDT's, las señales de estos transductores nos indican posición del pistón y la carga aplicada por el actuador sobre alguna estructura. Estas señales han de ser comparadas contra alguna otra, generada por el usuario con el fin de limitar la capacidad del actuador.



Cuando las señales de los transductores sobrepasen los límites establecidos por el operador, el sistema generara una señal de aviso indicando cual es el actuador que ha sobrepasado el límite y si este fue en carga o desplazamiento. Además, el detector de límites de operación puede ser programado para hacer un paro general de la prueba que se este llevando acabo con el ATS si es que así lo desea el operador.

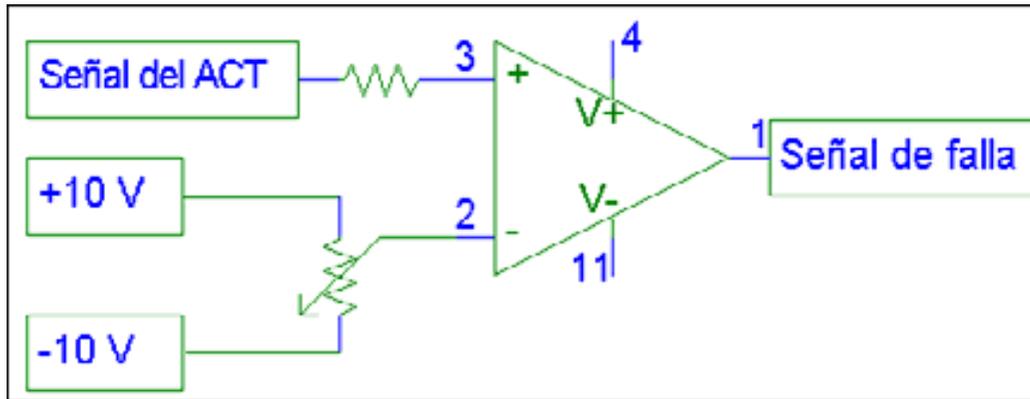
En el diagrama de bloques del sistema ATS junto con el detector de límites de operación queda constituido de la siguiente forma:



8.1 Sistema analógico-digital.

8.1.1 Diseño.

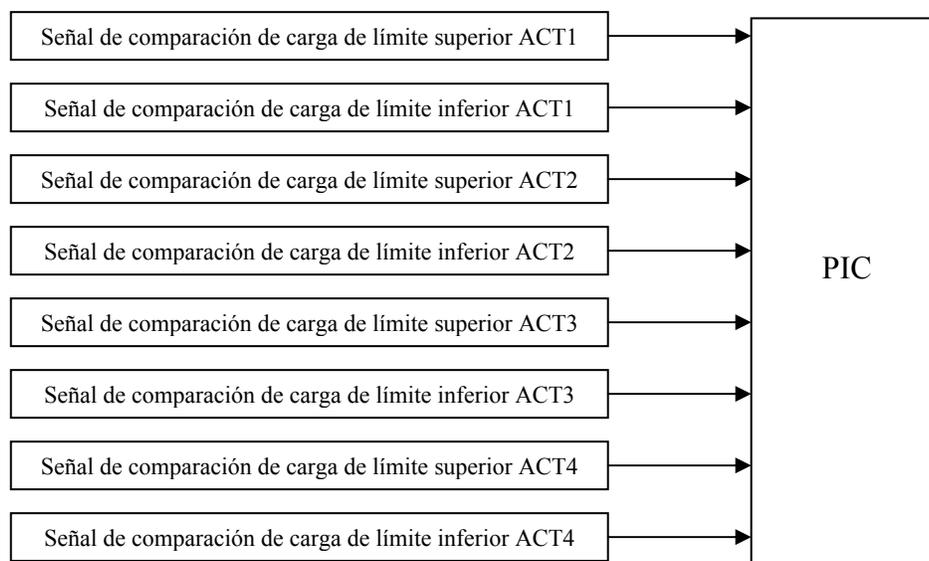
Las señales de los transductores ya acondicionadas; es decir, las señales de $\pm 10V$ pueden ser comparadas contra las señales de límite utilizando un amplificador operacional como comparador. La señal límite tendrá un valor $\pm 10V$ también.

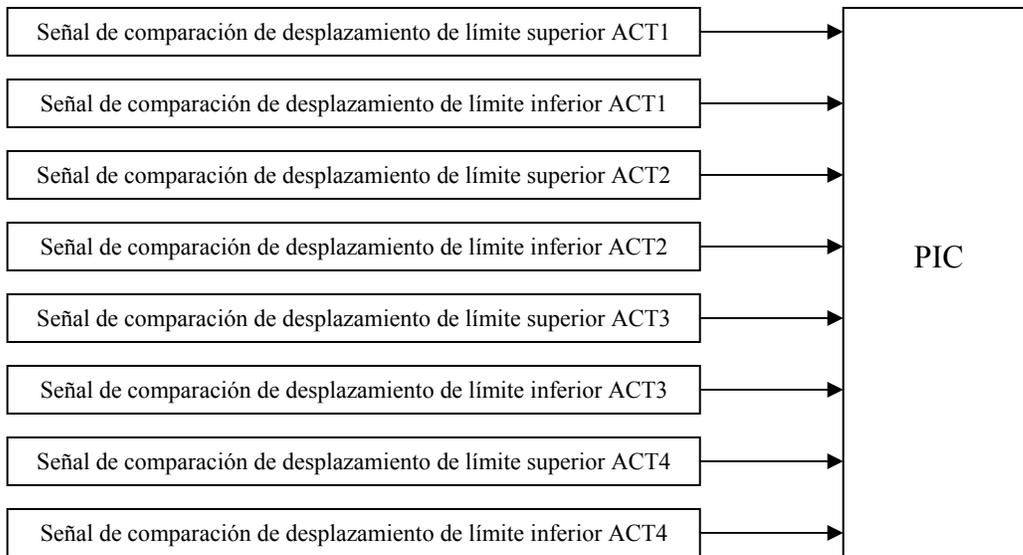


El valor de la señal límite es variable utilizando un potenciómetro.

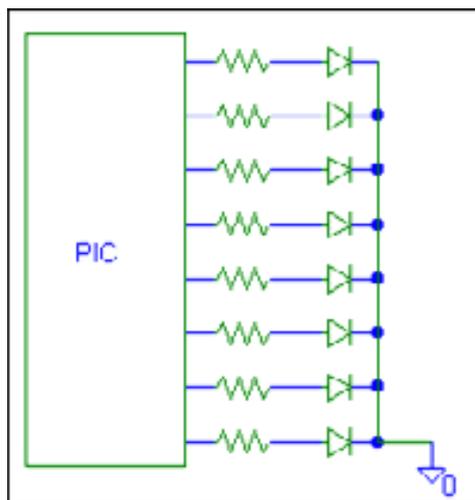
La delimitación de niveles es tanto para valores positivos, como para valores negativos, por lo que se declararan dos límites por variables, el superior y el inferior, dando como resultado el establecimiento de cuatro límites por actuador, dos para carga y dos para desplazamiento. Esto implica la implementación de cuatro comparadores por actuador.

Cuando el valor del transductor esté sobre el límite, el amplificador operacional entregará una señal de $+5V$ la cual será alimentada a un PIC16F873 en un puerto de propósito general. Se utilizará un PIC para el control de la carga y otro para el desplazamiento. Por lo que cada PIC recibirá ocho niveles de límite generados por los operacionales, cuatro de nivel superior y cuatro de nivel inferior.

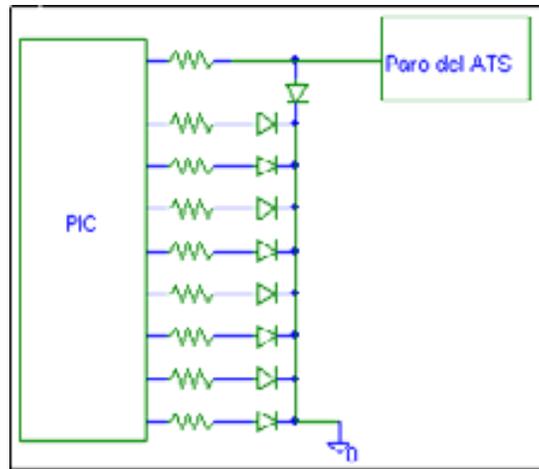




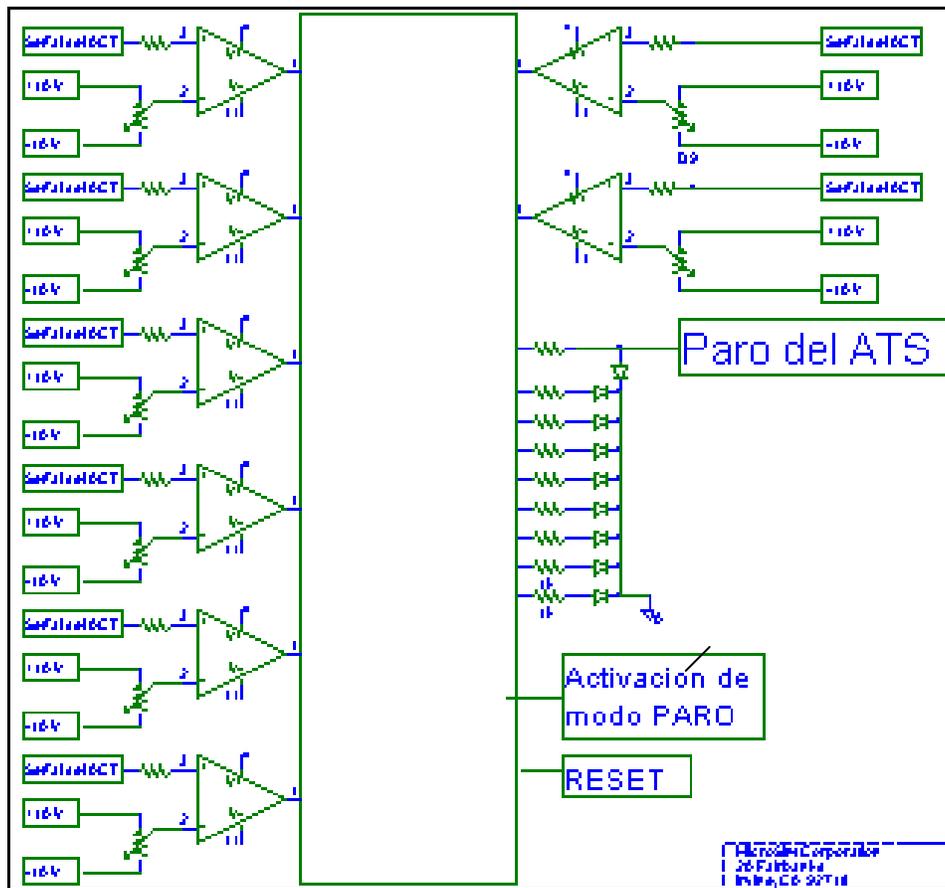
El principio de funcionamiento de los dos PIC's es el mismo. Un puerto del PIC recibe las cuatro señales de activación del paro del sistema para actuador, y el reset general. De no estar activada señal alguna de paro, el actuador solo se limitará a generar una señal luminosa intermitente de aviso (LED) por cada límite superado, y permanecerá en ese estado hasta que se active el reset.



De encontrarse funcionando para cualquier actuador y existiese un sobrepaso de límite en ese actuador, el equipo no solo mostrará la señal de aviso, además se generará una señal que active el sistema de paro del ATS.



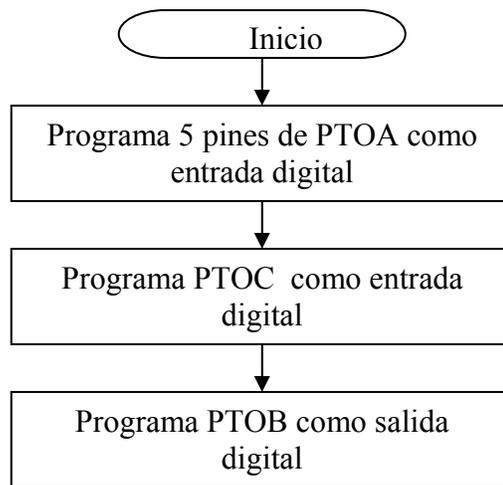
8.1.2 Diagrama del sistema.



8.1.3 Programa del PIC16F873 para el detector de límites de carga y desplazamiento.

8.1.3.1 Diagrama de flujo.

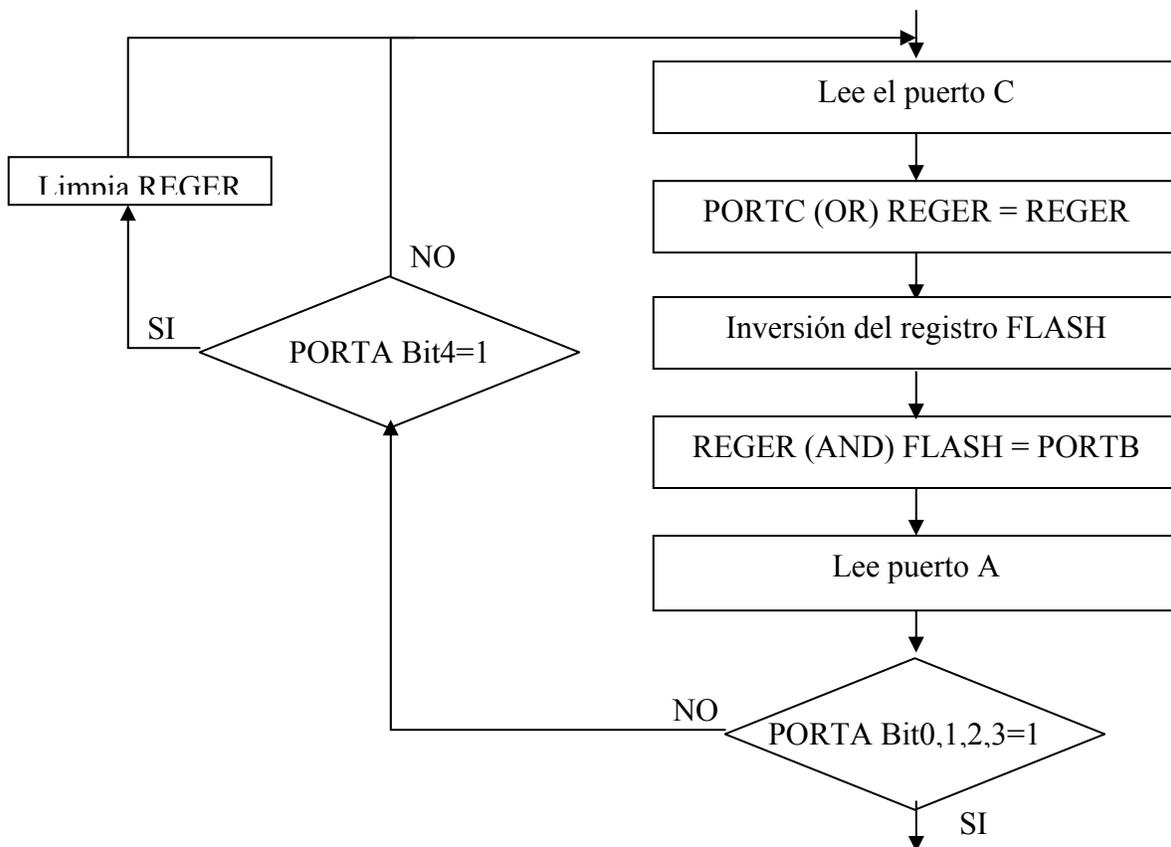
Se requiere de un total de trece entradas y nueve salidas. Para esto, se configura el puerto C como puerto B, como salida y los cinco primeros pines del puerto A como entradas.



Posteriormente se lee el puerto C, que es donde se conectan las señales de los comparadores, lo leído es colocado en un registro y posteriormente colocado en el puerto de salida, para indicar el actuador y tipo de sobrepaso (superior o inferior).

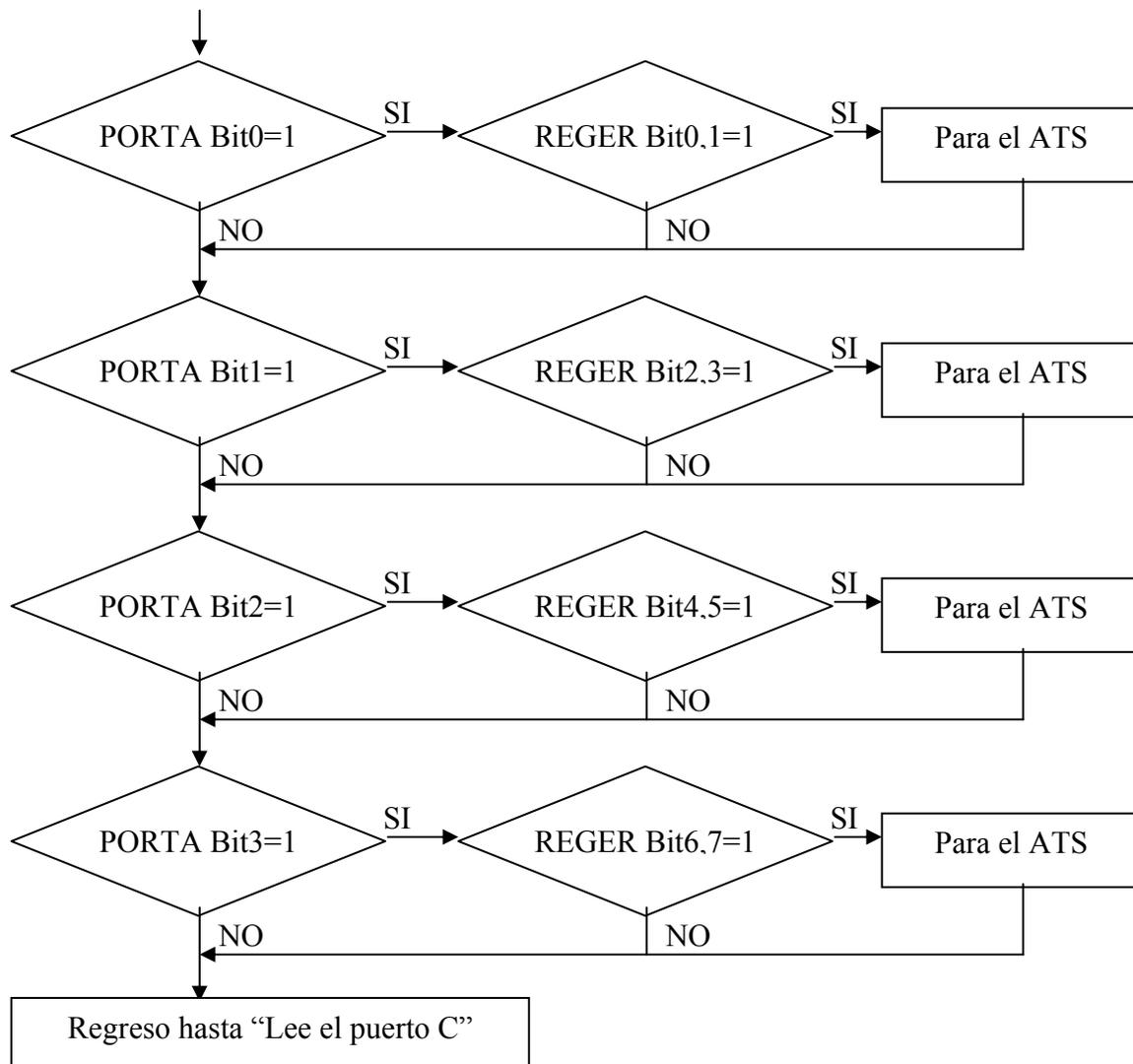
El registro donde se colocan las entradas es alternado con el valor 0x00 y los valores de entrada, esto con el fin de provocar un parpadeo de la luz del LED de la señal de salida.

Se lee el puerto A, que tienen las cuatro señales de paro, una por actuador y el reset. Si alguno de los actuadores está activado para trabajar en modo de paro el programa seguirá de lo contrario, el programa regresa a la lectura del puerto C.



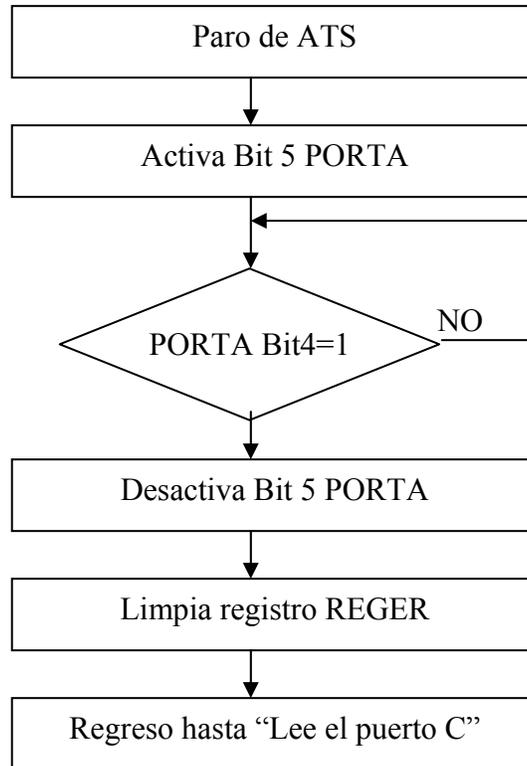
Si el reset fuera activado, se limpia el registro REGER borrando todo sobrepaso que hubiera ocurrido.

Se verifica cual de los cuatro actuadores está activado en modo paro y posteriormente si el sobrepaso ocurrido fue efectuado en ese actuador.



Cabe mencionar que si el fallo ocurrió tiempo atrás de la activación del modo paro, y el registro REGER no ha sido limpiado, el sistema tomará a este sobrepaso como el valor que activará el paro del equipo.

Una vez que se entró en el modo fallo, el PIC manda una señal que desactiva el ATS y permanece en este estado hasta que es puesto en alto el reset, regresando el programa a la lectura del puerto C y borrando el registro REGER.



El programa se repite de forma ininterrumpida todo el tiempo que esté funcionando el equipo

8.1.3.2 Programa.

```

list          p=16f873
include      "p16f873.inc"

FLASH       EQU    0X20
REGER       EQU    0x21
ESP         EQU    0X22

ORG 0
BSF  STATUS,RPO      ;CAMBIA DE BANKO
MOVLW 0XFF           ;PONE PUERTO C
MOVWF TRISC          ;COMO ENTRADA
MOVLW 0X06           ;DESACTIVA
MOVWF ADCON1         ;EL CONVERTIDOR
MOVLW 0X1F           ;PONE LOS PRIMEROS CINCO PINES DEL
MOVWF TRISA          ;PUERTO A COMO ENTRADAS
MOVLW 0X00           ;PONE AL PUERTO B
MOVWF TRISB          ;COMO SALIDA
BCF  STATUS,RPO      ;CAMBIO DE BANKO
CLRF  PORTA           ;LIMIA
CLRF  PORTC           ;LOS
CLRF  PORTB           ;PUERTOS
MOVLW 0X00
MOVWF REGER
MOVLW 0XFF           ;CARGA EL REGISTRO
MOVWF FLASH          ;QUE PONE EN ALTO Y BAJO LA SALIDA

INICIO      MOVFW  PORTC      ;LEE EL PUERTO C
  
```

```

XORLW 0X55 ;RECTIFICA LOS 1 DE NIVEL BAJO
IORWF REGER,F ;HACE UNA OR CON EL REGISTRO
REGER REGER ;MUEVE EL REEGISTRO AL ESPACIO
ANDWF FLASH,W ;PRENDE O APAGA EL BIT DE FALLO
MOVWF PORTB ;LO COLOCA EN EL PUERTO B
MOVFW PORTA ;LEE EL PUERTO A
ANDLW 0X0F ;LOS PRIMEROS CUATRO PINES
BTFSS STATUS,Z ;SI ESTA PRENDIDO
CALL PARO ;LLAMA A LA RUTINA PARO
BTFSC PORTA,4 ;VERIFICA EL PUERTO 4 DE A
CLRF REGER ;PARA APAGAR EL REGISTRO
MOVF FLASH ;VERIFICA EL VALOR DEL FLASH
BTFSS STATUS,Z ;PARA CONMUTARLO
GOTO BAJO ;ENTRE 0X00
GOTO ALTO ;Y 0XFF
GOTO INICIO

BAJO MOVLW 0X00 ;PONE A 0
MOVWF FLASH ;EL VALOR DE FLASH
GOTO INICIO ;PARA PONER A BAJO LA SALIDA B

ALTO MOVLW 0X93
MOVWF ESP
APGD DECSZ ESP
GOTO APGD
MOVLW 0XFF ;PONE A 1
MOVWF FLASH ;EL VALOR FLASH
GOTO INICIO ;PARA PONER A ALTO LA SALIDA B

PARO BTFSC PORTA,0 ;VERIFICA SI PARO 1 ESTA ACTIVADO
CALL VERIFCA0 ;VERIFICA QUE EL FALLO SEA EN ESE ACTUADOR
BTFSC PORTA,1 ;VERIFICA SI PARO 2 ESTA ACTIVADO
CALL VERIFICA1 ;VERIFICA QUE EL FALLO SEA EN ESE ACTUADOR
BTFSC PORTA,2 ;VERIFICA SI PARO 3 ESTA ACTIVADO
CALL VERIFICA2 ;VERIFICA QUE EL FALLO SEA EN ESE ACTUADOR
BTFSC PORTA,3 ;VERIFICA SI PARO 4 ESTA ACTIVADO
CALL VERIFICA3 ;VERIFICA QUE EL FALLO SEA EN ESE ACTUADOR
RETURN

ACTIVA BSF PORTA,5 ;PRENDE EL BIT DE PARO
BTFSS PORTA,4 ;Y PERMANECA AHÍ HASTA ACTIVAR
GOTO ACTIVA ;BIT 4 DEL PUERTO A
BCF PORTA,5 ;LIMPIA LA BANDERA DE PARO
CLRF REGER ;LIMPIA EL REGISTRO DE FALLAS
GOTO INICIO ;REGRESA AL INICIO

VERIFCA0 MOVFW REGER ;VERIFICA QUE EL PROBLEMA SEA
ANDLW 0X03 ;EN LAS ENTRADAS DEL ACTUADOR1
BTFSC STATUS,Z ;
RETURN ;
MOVWF PORTB ;
GOTO ACTIVA ;

VERIFICA1 MOVFW REGER ;VERIFICA QUE EL PROBLEMA SEA
ANDLW 0X0C ;EN LAS ENTRADAS DEL ACTUADOR2
BTFSC STATUS,Z ;
RETURN ;
MOVWF PORTB ;
GOTO ACTIVA ;

VERIFICA2 MOVFW REGER ;VERIFICA QUE EL PROBLEMA SEA
ANDLW 0X30 ;EN LAS ENTRADAS DEL ACTUADOR3
BTFSC STATUS,Z ;
RETURN ;
MOVWF PORTB ;
GOTO ACTIVA ;

```

```

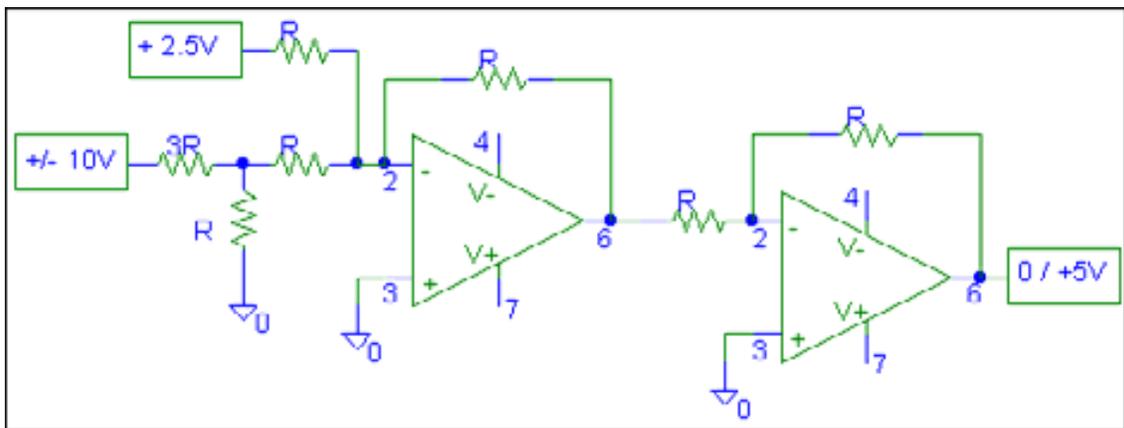
VERIFICA3      MOVFW REGER      ;VERIFICA QUE EL PROBLEMA SEA
ANDLW OXCO     ; EN LAS ENTRADAS DEL ACTUADOR4
BTFSK STATUS,Z
RETURN
MOVWF PORTB
GOTO ACTIVA
END

```

8.2 Sistema digital.

8.2.1 Diseño.

Los niveles de tensión proporcionados por los acondicionadores de las celdas de carga son de ± 10 V, mientras que el DAC del PIC trabaja de 0 a 5 V. La tensión se tiene que reducir a una cuarta parte, sumar un nivel de directa (off set) de 2.5 V dejando una señal que opera de 0 a 5 V, lista para ser procesada por el DAC del PIC. Esta señal queda invertida por la forma de operación del amplificador, para poder alimentar la señal al PIC se requiere de un amplificador inversor con ganancia unitaria.

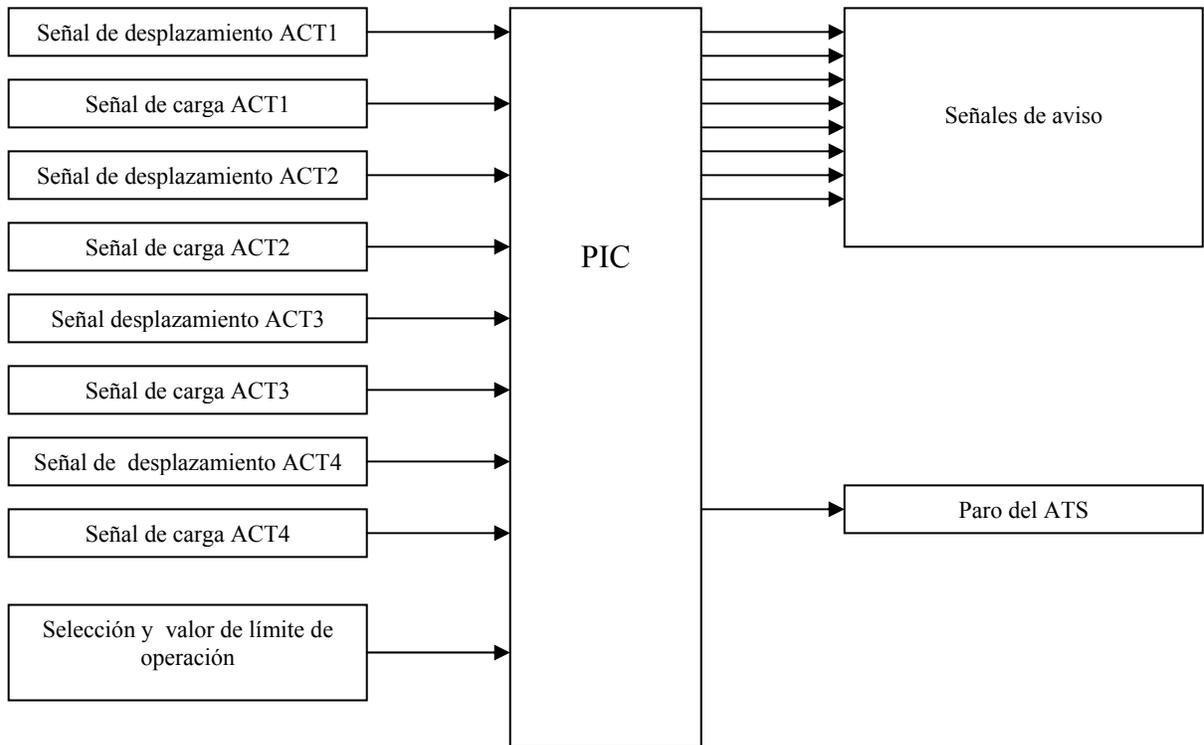


Finalmente se obtiene una señal que puede ser procesada por el CAD del PIC.

8.2.2 Programa para el PIC18F4439 para el detector de límites de carga y desplazamiento.

8.2.2.1 Diagrama de flujo.

Lo primero es configurar los puertos de entrada y salida; se requiere de ocho entradas AD, 14 salidas digitales y 5 entradas digitales. El PIC18F4439 cuenta con ocho entradas analógicas de 10 bits de resolución, por lo que se pueden procesar todas las ocho señales de los transductores, dos por actuador (carga y desplazamiento). Colocando los resultados en espacios de memoria identificados por el número de actuador, transductor y parte de la palabra de 10 bits que forman, ejemplo: AC3CL, es la parte baja de la señal de carga del actuador 3, AC4DH es la parte alta del actuador cuatro de la señal de desplazamiento.



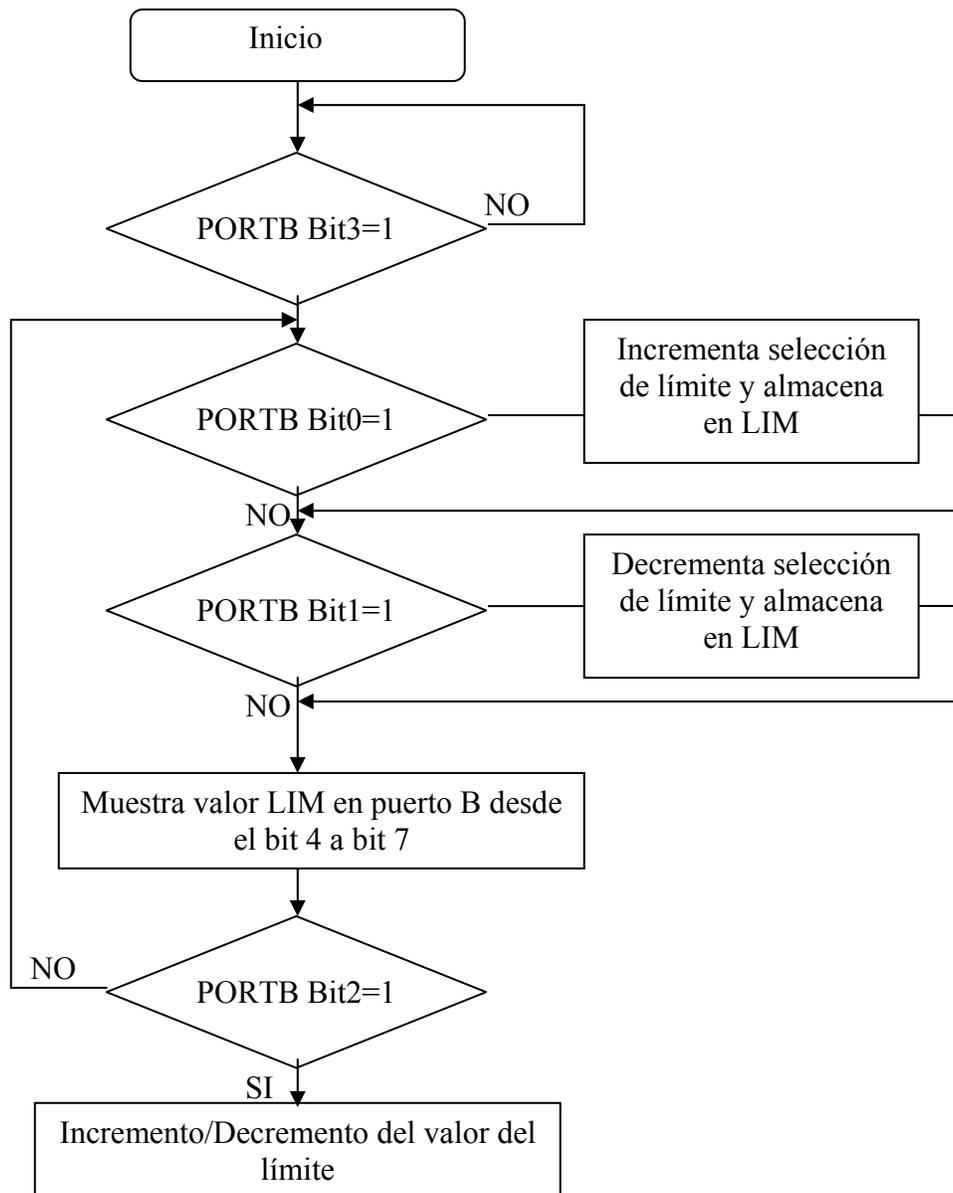
Mediante el uso de las cuatro entradas digitales se seleccionará el actuador, tipo de señal y nivel de operación. Los primeros cuatro bits del puerto B servirán para este propósito. El bit número 3 es el interruptor de inicio, mientras éste no esté en alto, el detector de límites no estará en función.

Lo primero es la selección del límite que se selecciona, en total son 16 los límites que se establecerán. Los pines 0 y 1 servirán para incrementar o decrementar la selección del límite. Mientras que los últimos 4 bits del puerto B desplegarán un número binario que muestra el nivel seleccionado. Los límites están representados de la siguiente manera.

Número	Límite
0000	Límite de carga superior ACT1
0001	Límite de carga inferior del ACT1
0010	Límite de desplazamiento superior ACT1
0011	Límite de desplazamiento inferior del ACT1
0100	Límite de carga superior ACT2
0101	Límite de carga inferior del ACT2
0110	Límite de desplazamiento superior ACT2
0111	Límite de desplazamiento inferior del ACT2
1000	Límite de carga superior ACT3
1001	Límite de carga inferior del ACT3
1010	Límite de desplazamiento superior ACT3
1011	Límite de desplazamiento inferior del ACT3
1100	Límite de carga superior ACT4

1101	Límite de carga inferior del ACT4
1110	Límite de desplazamiento superior ACT4
1111	Límite de desplazamiento inferior del ACT4

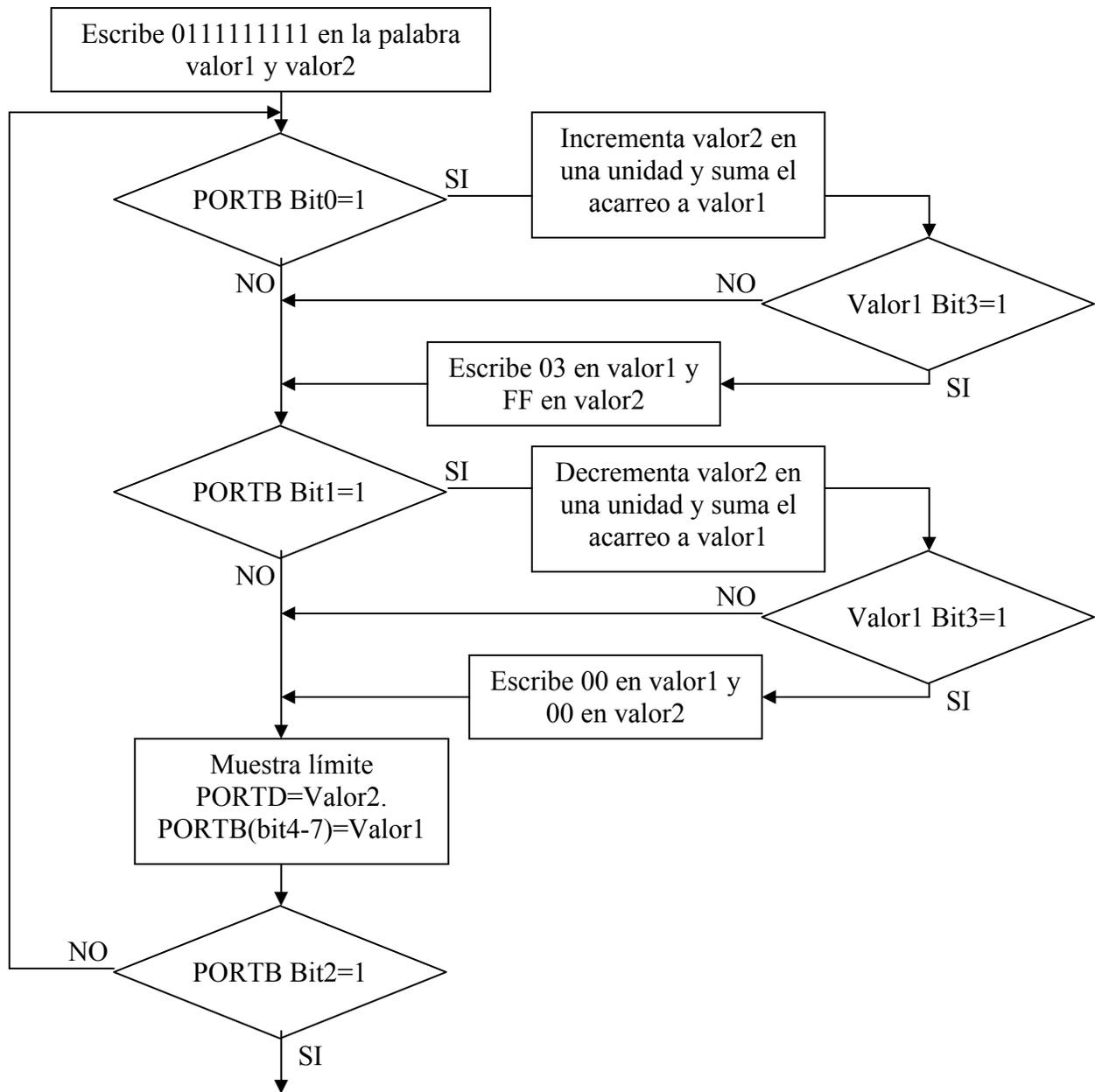
El bit 2 del puerto B funciona como botón de enter, una vez seleccionado el límite a establecer.



Cada límite está establecido por una palabra de 10 bits, por lo que se requiere de dos espacios de memoria para guardar el valor. Todos los límites inician con un valor de 01 1111 1111, que es el valor digital para cero carga y cero desplazamiento, a partir de este valor se establecerán los límites positivos y negativos del sistema. Los espacios de memoria de los límites son identificados por el número de actuador, tipo de límite y parte de la

palabra de 10 bits que forma, por ejemplo, el límite superior de desplazamiento del gato 3 estará contenido en los registros A3SDH y A3SDL.

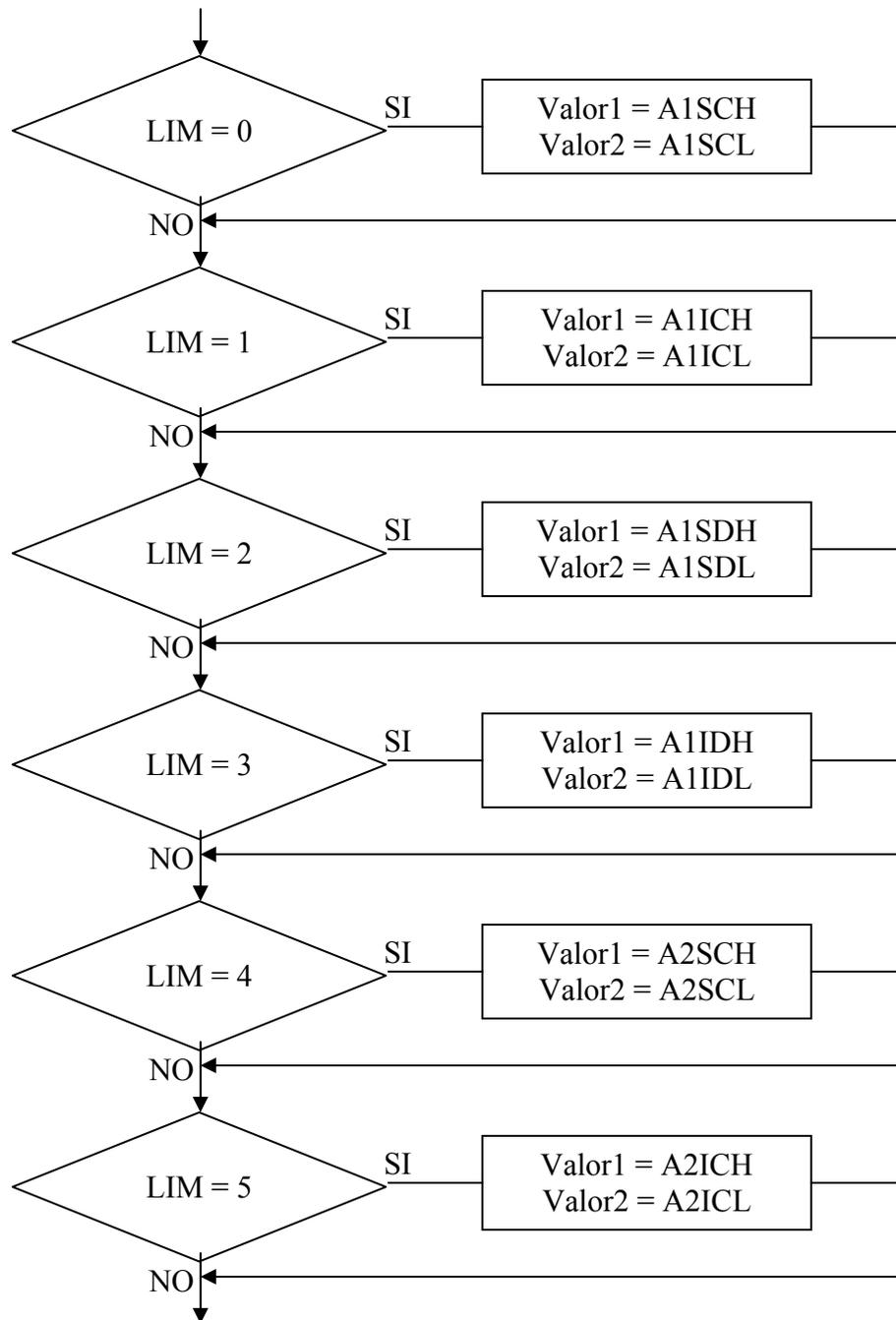
El incremento consta de sumar 0x01 a la parte baja del límite y se guarda en mismo registro, posteriormente el acarreo de adición se suma a la parte alta y se guarda. Si el bit 3 de la palabra alta es igual a 1 significa que se ha sobrepasado la palabra de 10 bits, por lo que se escribirá en forma automática 0xFF en la parte baja y 0x03 en la parte alta.

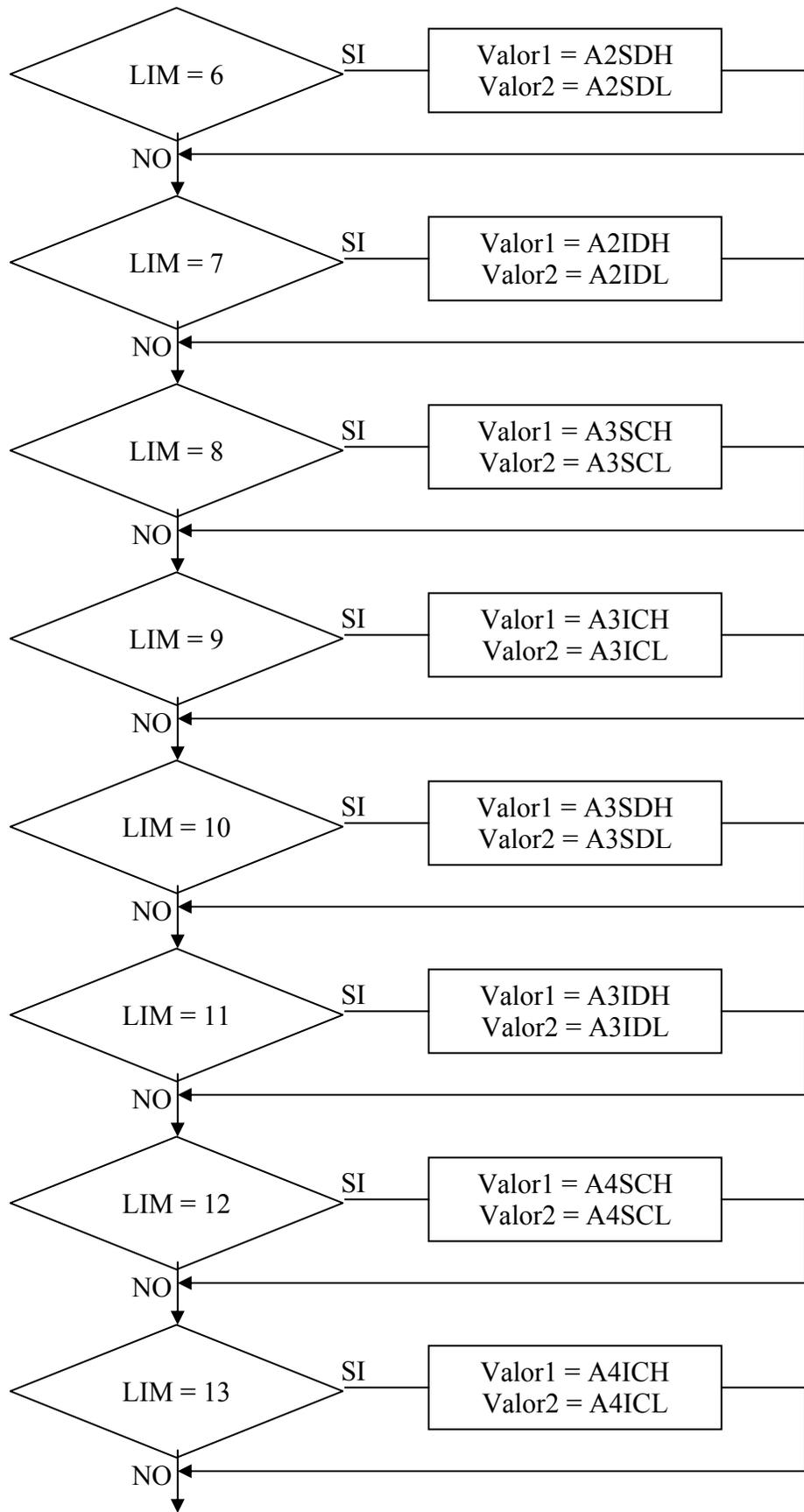


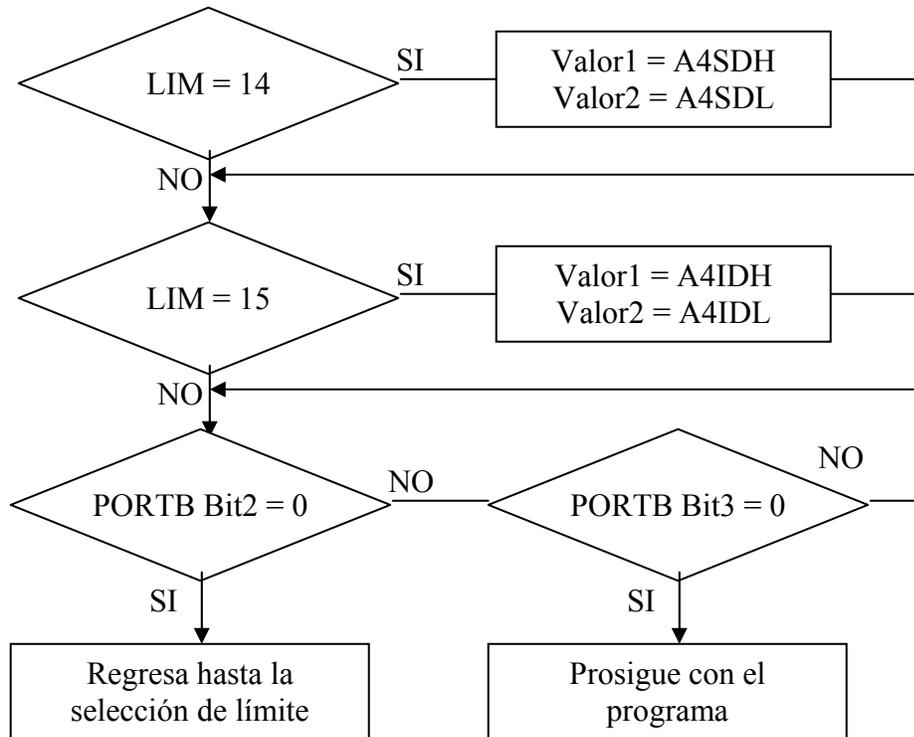
Para el decremento se sumara 0xFF a la parte baja, el resultado es almacenado en el mismo registro, el acarreo y 0xFF se adicionan a la parte alta, si el bit número 3 de esta nueva palabra resultara 1 que el límite inferior ha caído en un valor por abajo de cero, y se escribe 0x00 en ambos registros del límite.

Los 10 bits del nivel son desplegados en los en los últimos 4 bits del puerto B y el puerto D. Nuevamente el bit 2 del puerto B funciona como enter.

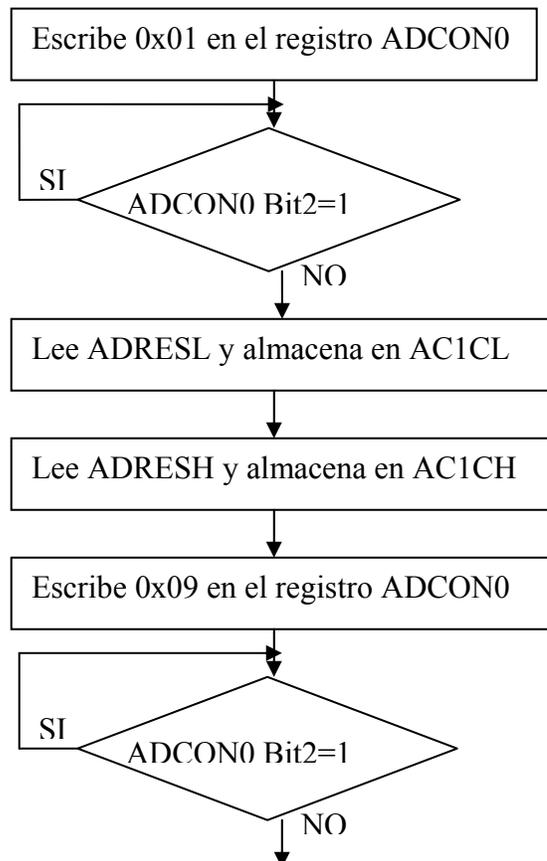
Se determinan los espacios de memoria donde serán escritos el valor 1 y 2 al hacer comparativos del valor LIM.

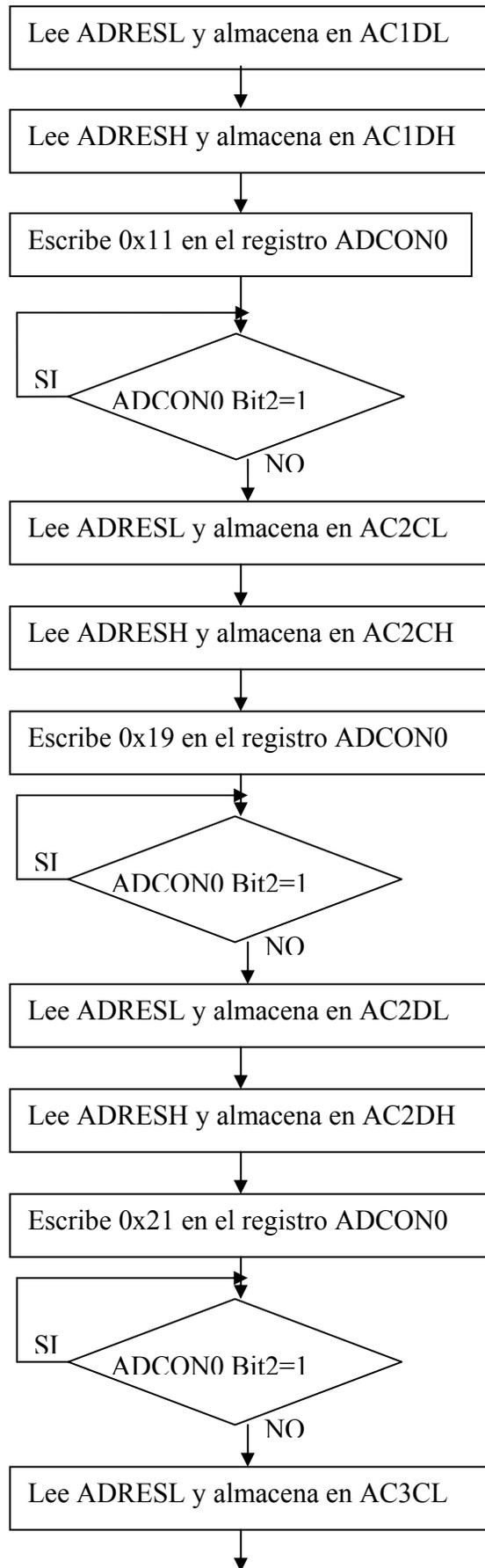


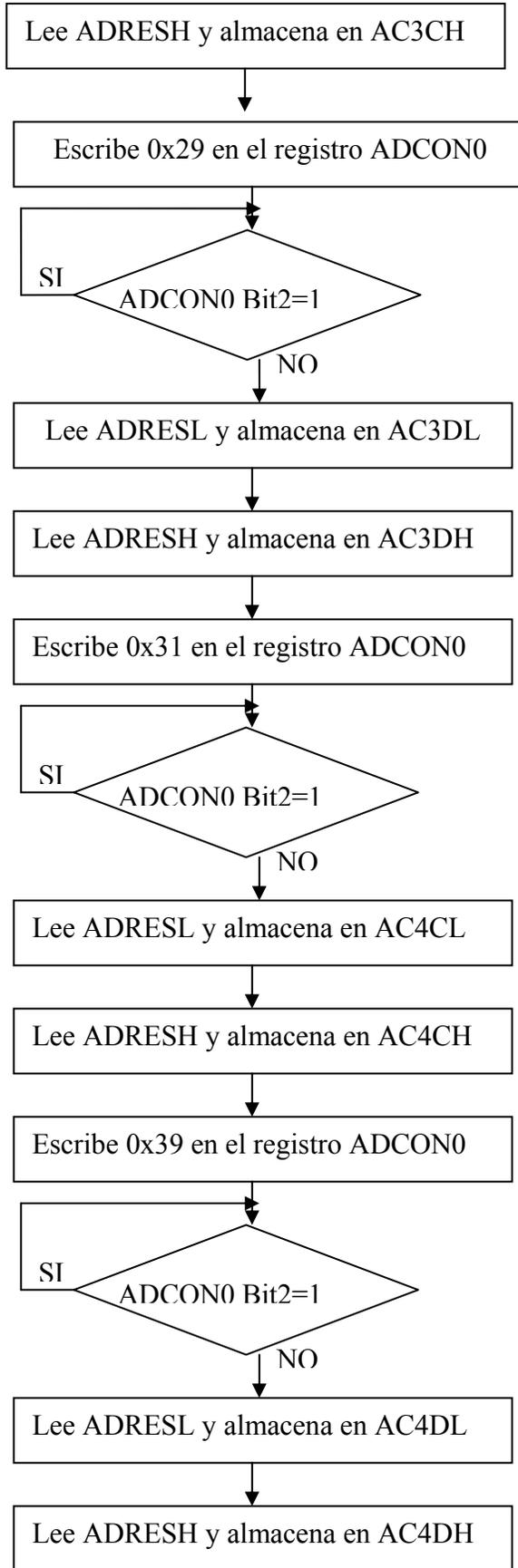




Ya definidos los límites, se tienen que leer y almacenar los valores de las señales provenientes de los actuadores.







Cuando entra en operación el sistema se compara cada una de las partes bajas de las señales de entrada y las de sus respectivos límites; es decir, se compara:

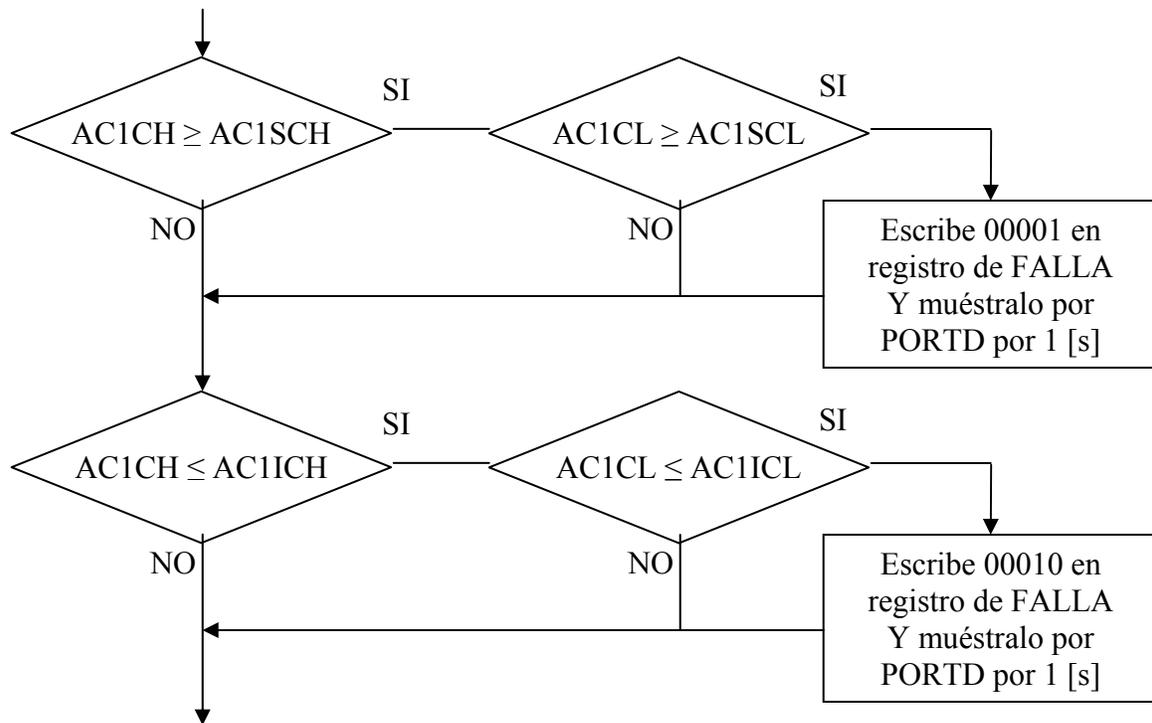
Señal	Relación con el límite
AC1CH	\geq AC1SCH
	\leq AC1ICH
AC1DH	\geq AC1SDH
	\leq AC1IDH
AC2CH	\geq AC2SCH
	\leq AC2ICH
AC2DH	\geq AC2SDH
	\leq AC2IDH
AC3CH	\geq AC3SCH
	\leq AC3ICH
AC3DH	\geq AC3SDH
	\leq AC3IDH
AC4CH	\geq AC3SCH
	\leq AC3ICH
AC4DH	\geq AC3SDH
	\leq AC3IDH

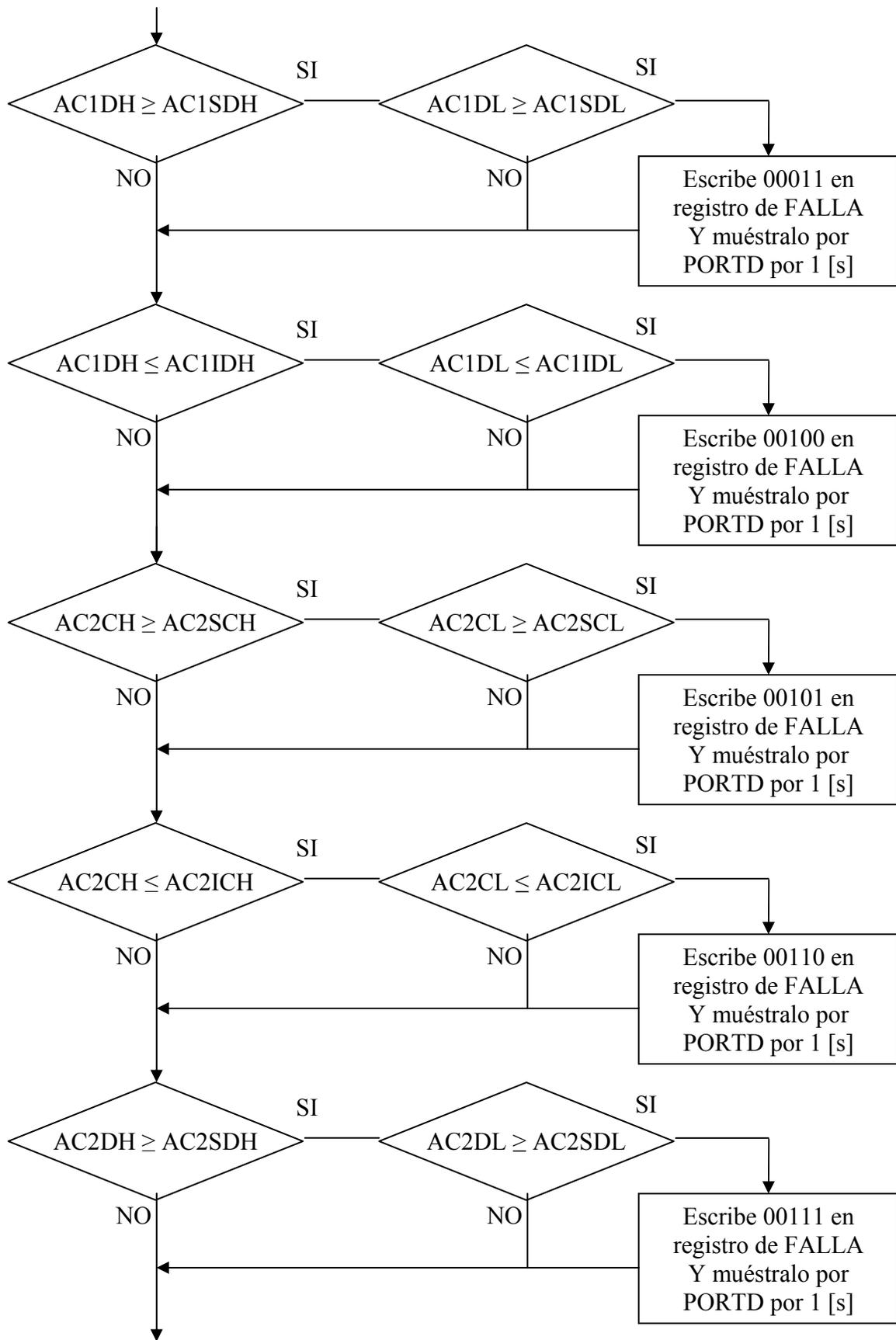
Si alguna comparación resulta cierta se procede a la comparación de la parte baja de la señal y la respectiva palabra menos significativa del límite superado.

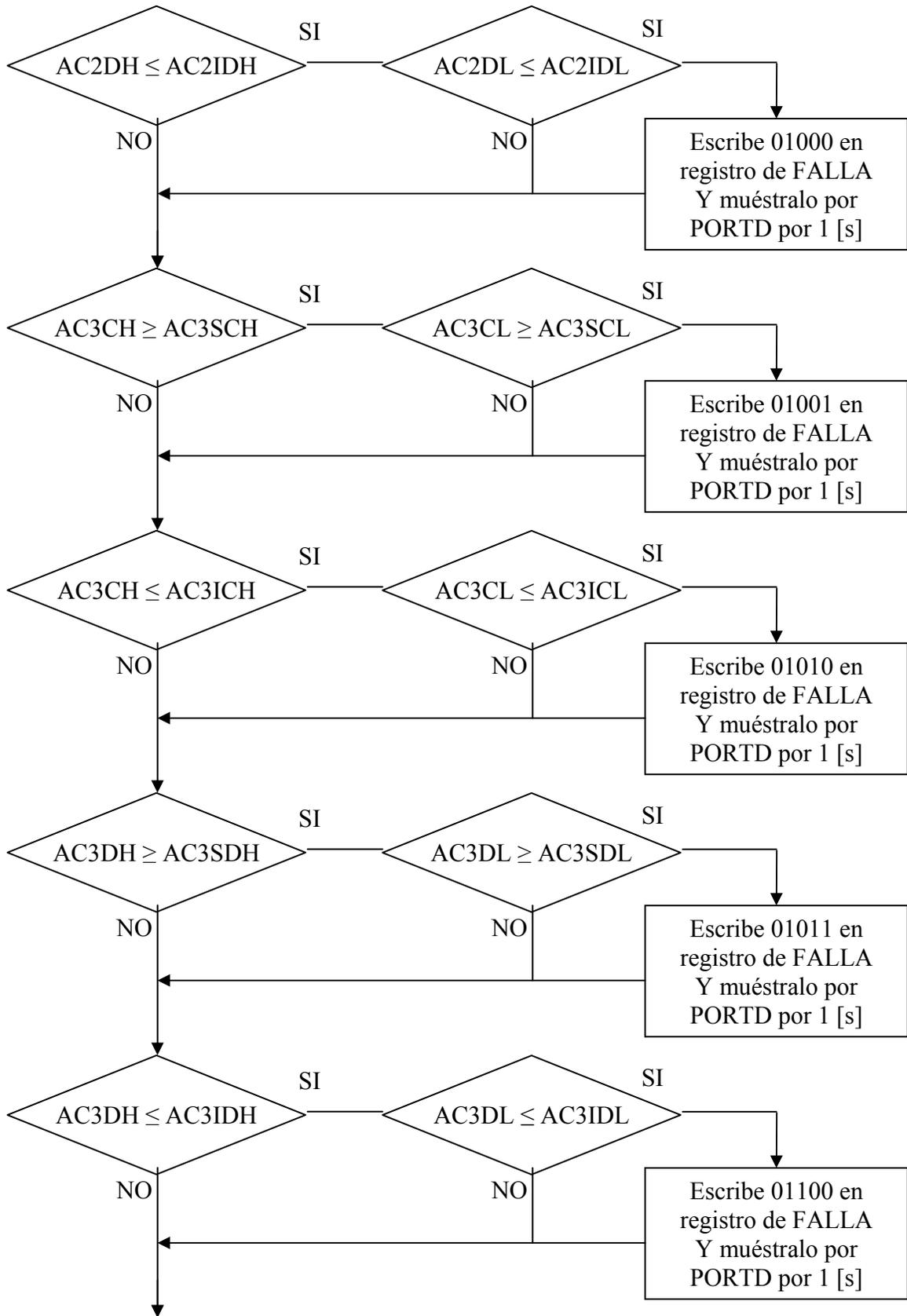
Señal	Relación con el límite
AC1CL	\geq AC1SCL
	\leq AC1ICL
AC1DL	\geq AC1SDL
	\leq AC1IDL
AC2CL	\geq AC2SCL
	\leq AC2ICL
AC2DL	\geq AC2SDL
	\leq AC2IDL
AC3CL	\geq AC3SCL
	\leq AC3ICL
AC3DL	\geq AC3SDL
	\leq AC3IDL
AC4CL	\geq AC3SCL
	\leq AC3ICL
AC4DL	\geq AC3SDL
	\leq AC3IDL

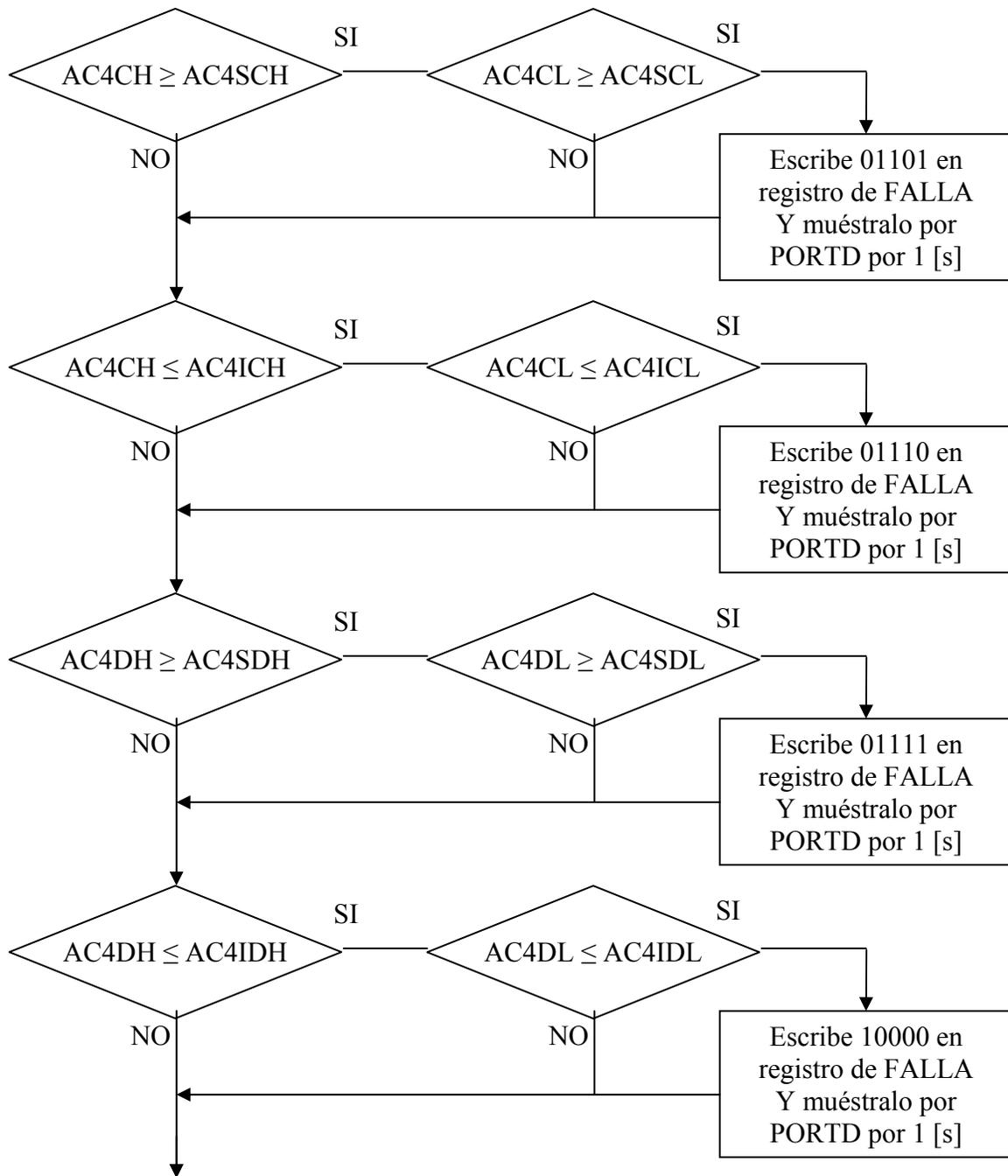
De no resultar la comparación igual, se escribirá una palabra de 5 bits guardada en el registro FALLA para indicar el tipo de sobrepaso ocurrido.

Palabra	Límite superado
00001	Límite superior de carga ACT1
00010	Límite inferior de carga ACT1
00011	Límite superior de desplazamiento ACT1
00100	Límite inferior de desplazamiento ACT1
00101	Límite superior de carga ACT2
00110	Límite inferior de carga ACT2
00111	Límite superior de desplazamiento ACT2
01000	Límite inferior de desplazamiento ACT2
01001	Límite superior de carga ACT3
01010	Límite inferior de carga ACT3
01011	Límite superior de desplazamiento ACT3
01100	Límite inferior de desplazamiento ACT3
01101	Límite superior de carga ACT4
01110	Límite inferior de carga ACT4
01111	Límite superior de desplazamiento ACT4
10000	Límite inferior de desplazamiento ACT4





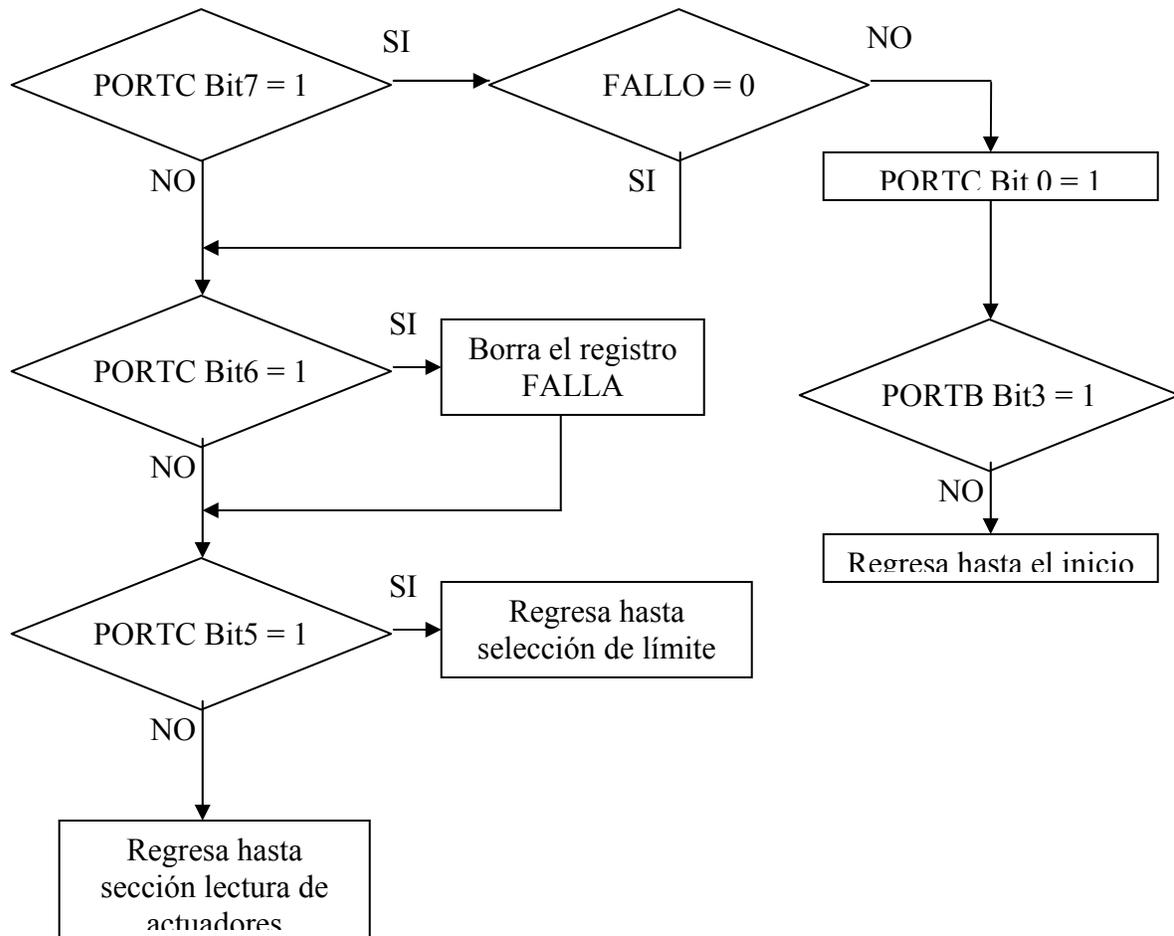




Las comparaciones se harán una a una, y cuando un exista un fallo la palabra será colocada en el puerto D y mantenida por un segundo, rescribiéndose el registro FALLA con cada límite superado y mostrando en forma secuencial éstos. Si el bit 6 del puerto C está en alto, se realizará una limpieza del registro FALLA.

Mientras que el bit 7 del puerto C funcionará como el interruptor del sistema de paro. Cuando este bit no esté activado, el sistema solo enviará las señales de aviso mediante el registro FALLA. Cuando esté activado y FALLA sea distinto de cero, se genera una señal

de paro, mandando a alto el bit 0 del puerto C y se mantendrá ahí hasta que el bit 3 del puerto B sea activado, reiniciando el programa.



Durante la ejecución del programa, si el bit 5 del puerto C es puesto a 1, el programa regresará hasta la selección de límite, esto con el fin de poder variar los límites durante la operación del equipo.

8.2.2.2 Programa

```

list      p=18f4436
include   "p18f4436.inc"

LIMTS    EQU    0X20
LIMH     EQU    0X21
LIML     EQU    0X22
A1SCH    EQU    0X23
A1SCL    EQU    0X24
A1ICH    EQU    0X25
A1ICL    EQU    0X27
A1SDH    EQU    0X28
A1SDL    EQU    0X29
A1IDH    EQU    0X2A
A1IDL    EQU    0X2B
  
```

```

A2SCH EQU 0X2C
A2SCL EQU 0X2D
A2ICH EQU 0X2E
A2ICL EQU 0X2F
A2SDH EQU 0X30
A2SDL EQU 0X31
A2IDH EQU 0X32
A2IDL EQU 0X33
A3SCH EQU 0X34
A3SCL EQU 0X35
A3ICH EQU 0X36
A3ICL EQU 0X37
A3SDH EQU 0X38
A3SDL EQU 0X39
A3IDH EQU 0X3A
A3IDL EQU 0X3B
A4SCH EQU 0X3C
A4SCL EQU 0X3D
A4ICH EQU 0X3E
A4ICL EQU 0X3F
A4SDH EQU 0X40
A4SDL EQU 0X41
A4IDH EQU 0X42
A4IDL EQU 0X43
AC1CL EQU 0X44
AC1CH EQU 0X45
AC1DL EQU 0X46
AC1DH EQU 0X47
AC2CL EQU 0X44
AC2CH EQU 0X45
AC2DL EQU 0X46
AC2DH EQU 0X47
AC3CL EQU 0X48
AC3CH EQU 0X49
AC3DL EQU 0X4A
AC3DH EQU 0X0B
AC4CL EQU 0X4C
AC4CH EQU 0X4D
AC4DL EQU 0X4E
AC4DH EQU 0X4F
FALLO EQU 0X50

ORG 0
MOVLW 0X80 ;ACTIVA TODAS LAS
MOVWF ADCON1 ;ENTRADAS ANALOGICAS
MOVLW 0X0F ;ACTIVA 4 BITS DEL PUERTO B
MOVWF TRISB ;COMO ENTRADA Y 4 COMO SALIDA
MOVLW 0X00 ;ACTIVA PUERTO D
MOVWF TRISD ;COMO SALIDA
MOVLW 0XE0 ;COLOCA LOS ÚLTIMOS TRES BITS
MOVWF TRISC ;DEL PORT C COMO ENTRADAS
CLRF FALLO ;LIMPIA LOS REGISTROS DE FALLA

INICIO BTFS PORTB,3 ;VERIFICA BIT 3 DEL PUERTO B
GOTO INICIO ;SI ES IGUAL A 1 INICIA EL PROGRAMA
SELIM BTFS PORTB,0 ;VERIFICA BIT 0 DEL PUERTO B PARA
CALL MAYLIM ;LLAMA EL INCREMENTO DE SELECCION DEL LÍMITE
ML BTFS PORTB,1 ;VERIFICA BIT 1 DEL PUERTO B PARA
CALL MENLIM ;EL DECREMENTO DE LA SELECCION DEL LÍMITE
MOVLW 0X10 ;RECORRE EL VALOR DE SELECCIÓN DE
MULWF LIMITS ;LÍMITE EN CUATRO BITS
MOVFW PRODL ;PARA PODER SER DESPLEGADO EN LOS
MOVWF PORTB ;ÚLTIMOS CUATRO BITS DEL PUERTO B

BTFS PORTB,2 ;VERIFICA EL BIT 2 DEL PUERTO B
GOTO SELIM ;PARA PODER PROCEGUIR CON EL PROCESO

```

	MOVLW	0XFF	;COLOCA EL VALOR DE 01 1111 1111 COMO
	MOVWF	LIML	;VALOR INICIAL DEL LÍMITE
	MOVLW	0X01	;
	MOVWF	LIMH	;
LÍMITE	BTFSC	PORTB,0	;VERIFICA EL BIT 0 DEL PUERTO B PARA
	CALL	INCLIM	;EL INCREMENTO DEL VALOR DEL LÍMITE
	BTFSC	PORTB,1	;VERIFICA EL BIT 1 DEL PUERTO B PARA
	CALL	DECLIM	;EL DECREMENTO DEL VALOR DEL LÍMITE
	MOVFF	LIML,PORTD	;
	MOVLW	0X10	;RECORRE EL VALOR DE LA PARTE ALTA
	MULWF	LIMH	;DEL LÍMITE EN CUATRO BITS
	MOVWF	PRODL	;PARA PODER SER DESPLEGADO EN LOS
	MOVWF	PORTB	;ÚLTIMOS CUATRO BITS DEL PUERTO B
	BTFSS	PORTB,2	;VERIFICA EL BIT 2 DEL PUERTO B PARA
	GOTO	LÍMITE	;PODER PROSEGUIR EN EL PROGRAMA
UNO	MOVLW	0X00	;VERIFICA EL LÍMITE SELECCIONADO
	CPFSEQ	LIMITS	;ES EL 0
	GOTO	DOS	;SI NO VERIFICARA SI EL SIGUIENTE
	MOVFF	LIML,A1SCL	;SI ES EL LÍMITE COLOCA EL VALOR DEL
	MOVFF	LIMH,A1SCH	;LÍMITE EN LOS ESPACIOS ASIGNADOS
	GOTO	SIGUE	;Y PROSIGUE CON EL PROGRAMA
DOS	MOVLW	0X01	;VERIFICA EL LÍMITE SELECCIONADO
	CPFSEQ	LIMITS	;ES EL 1
	GOTO	TRES	;SI NO VERIFICARA SI EL SIGUIENTE
	MOVFF	LIML,A1ICL	;SI ES EL LÍMITE COLOCA EL VALOR DEL
	MOVFF	LIMH,A1ICH	;LÍMITE EN LOS ESPACIOS ASIGNADOS
	GOTO	SIGUE	;Y PROSIGUE CON EL PROGRAMA
TRES	MOVLW	0X02	;VERIFICA EL LÍMITE SELECCIONADO
	CPFSEQ	LIMITS	;ES EL 2
	GOTO	CUATRO	;SI NO VERIFICARA SI EL SIGUIENTE
	MOVFF	LIML,A1SDL	;SI ES EL LÍMITE COLOCA EL VALOR DEL
	MOVFF	LIMH,A1SDH	;LÍMITE EN LOS ESPACIOS ASIGNADOS
	GOTO	SIGUE	;Y PROSIGUE CON EL PROGRAMA
CUATRO	MOVLW	0X03	;VERIFICA EL LÍMITE SELECCIONADO
	CPFSEQ	LIMITS	;ES EL 3
	GOTO	CINCO	;SI NO VERIFICARA SI EL SIGUIENTE
	MOVFF	LIML,A1IDL	;SI ES EL LÍMITE COLOCA EL VALOR DEL
	MOVFF	LIMH,A1IDH	;LÍMITE EN LOS ESPACIOS ASIGNADOS
	GOTO	SIGUE	;Y PROSIGUE CON EL PROGRAMA
CINCO	MOVLW	0X04	;VERIFICA EL LÍMITE SELECCIONADO
	CPFSEQ	LIMITS	;ES EL 4
	GOTO	SEIS	;SI NO VERIFICARA SI EL SIGUIENTE
	MOVFF	LIML,A2SCL	;SI ES EL LÍMITE COLOCA EL VALOR DEL
	MOVFF	LIMH,A2SCH	;LÍMITE EN LOS ESPACIOS ASIGNADOS
	GOTO	SIGUE	;Y PROSIGUE CON EL PROGRAMA
SEIS	MOVLW	0X05	;VERIFICA EL LÍMITE SELECCIONADO
	CPFSEQ	LIMITS	;ES EL 5
	GOTO	SIETE	;SI NO VERIFICARA SI EL SIGUIENTE
	MOVFF	LIML,A2ICL	;SI ES EL LÍMITE COLOCA EL VALOR DEL
	MOVFF	LIMH,A2ICH	;LÍMITE EN LOS ESPACIOS ASIGNADOS
	GOTO	SIGUE	;Y PROSIGUE CON EL PROGRAMA
SIETE	MOVLW	0X06	;VERIFICA EL LÍMITE SELECCIONADO
	CPFSEQ	LIMITS	;ES EL 6
	GOTO	OCHO	;SI NO VERIFICARA SI EL SIGUIENTE
	MOVFF	LIML,A2SDL	;SI ES EL LÍMITE COLOCA EL VALOR DEL
	MOVFF	LIMH,A2SDH	;LÍMITE EN LOS ESPACIOS ASIGNADOS
	GOTO	SIGUE	;Y PROSIGUE CON EL PROGRAMA
OCHO	MOVLW	0X07	;VERIFICA EL LÍMITE SELECCIONADO

	CPFSEQ	LIMITS	;ES EL 7
	GOTO	NUEVE	;SI NO VERIFICARA SI EL SIGUIENTE
	MOVFF	LIML,A2IDL	;SI ES EL LÍMITE COLOCA EL VALOR DEL
	MOVFF	LIMH,A2IDH	;LÍMITE EN LOS ESPACIOS ASIGNADOS
	GOTO	SIGUE	;Y PROSIGUE CON EL PROGARMA
NUEVE	MOVLW	0X08	;VERIFICA EL LÍMITE SELECCIONADO
	CPFSEQ	LIMITS	;ES EL 8
	GOTO	DIEZ	;SI NO VERIFICARA SI EL SIGUIENTE
	MOVFF	LIML,A3SCL	;SI ES EL LÍMITE COLOCA EL VALOR DEL
	MOVFF	LIMH,A3SCH	;LÍMITE EN LOS ESPACIOS ASIGNADOS
	GOTO	SIGUE	;Y PROSIGUE CON EL PROGARMA
DIEZ	MOVLW	0X09	;VERIFICA EL LÍMITE SELECCIONADO
	CPFSEQ	LIMITS	;ES EL 9
	GOTO	ONCE	;SI NO VERIFICARA SI EL SIGUIENTE
	MOVFF	LIML,A3ICL	;SI ES EL LÍMITE COLOCA EL VALOR DEL
	MOVFF	LIMH,A3ICH	;LÍMITE EN LOS ESPACIOS ASIGNADOS
	GOTO	SIGUE	;Y PROSIGUE CON EL PROGARMA
ONCE	MOVLW	0X0A	;VERIFICA EL LÍMITE SELECCIONADO
	CPFSEQ	LIMITS	;ES EL A
	GOTO	DOCE	;SI NO VERIFICARA SI EL SIGUIENTE
	MOVFF	LIML,A3SDL	;SI ES EL LÍMITE COLOCA EL VALOR DEL
	MOVFF	LIMH,A3SDH	;LÍMITE EN LOS ESPACIOS ASIGNADOS
	GOTO	SIGUE	;Y PROSIGUE CON EL PROGARMA
DOCE	MOVLW	0X0B	;VERIFICA EL LÍMITE SELECCIONADO
	CPFSEQ	LIMITS	;ES EL B
	GOTO	TRECE	;SI NO VERIFICARA SI EL SIGUIENTE
	MOVFF	LIML,A3IDL	;SI ES EL LÍMITE COLOCA EL VALOR DEL
	MOVFF	LIMH,A3IDH	;LÍMITE EN LOS ESPACIOS ASIGNADOS
	GOTO	SIGUE	;Y PROSIGUE CON EL PROGARMA
TRECE	MOVLW	0X0C	;VERIFICA EL LÍMITE SELECCIONADO
	CPFSEQ	LIMITS	;ES EL C
	GOTO	CATOR	;SI NO VERIFICARA SI EL SIGUIENTE
	MOVFF	LIML,A4SCL	;SI ES EL LÍMITE COLOCA EL VALOR DEL
	MOVFF	LIMH,A4SCH	;LÍMITE EN LOS ESPACIOS ASIGNADOS
	GOTO	SIGUE	;Y PROSIGUE CON EL PROGARMA
CATOR	MOVLW	0X0D	;VERIFICA EL LÍMITE SELECCIONADO
	CPFSEQ	LIMITS	;ES EL D
	GOTO	QUIN	;SI NO VERIFICARA SI EL SIGUIENTE
	MOVFF	LIML,A4ICL	;SI ES EL LÍMITE COLOCA EL VALOR DEL
	MOVFF	LIMH,A4ICH	;LÍMITE EN LOS ESPACIOS ASIGNADOS
	GOTO	SIGUE	;Y PROSIGUE CON EL PROGARMA
QUIN	MOVLW	0X0E	;VERIFICA EL LÍMITE SELECCIONADO
	CPFSEQ	LIMITS	;ES EL E
	GOTO	DISE	;SI NO VERIFICARA SI EL SIGUIENTE
	MOVFF	LIML,A4SDL	;SI ES EL LÍMITE COLOCA EL VALOR DEL
	MOVFF	LIMH,A4SDH	;LÍMITE EN LOS ESPACIOS ASIGNADOS
	GOTO	SIGUE	;Y PROSIGUE CON EL PROGARMA
DISE	MOVFF	LIML,A2IDL	;SE COLOCA EL VALOR DEL
	MOVFF	LIMH,A2IDH	;LÍMITE EN LOS ESPACIOS ASIGNADOS
SIGUE	BTFS	PORTB,2	;VERIFICA EL BIT 2 DEL PUERTO B PARA
	GOTO	SELIM	;PARA PROGRACION DE OTRO LÍMITE
	BTFS	PORTB,3	;VERIFICA EL BIT 3 DEL PUERTO B PARA
	GOTO	SIGUE	;PROSEGUIR EN EL PROGRAMA
LECT	CALL	READ	;LEES LAS SEÑALES DE LOS ACTUADORES
SUC1	MOVFW	A1SCH	;VERIFICA SI LA PARTE ALTA VALOR DE CARGA
	CPFSLT	AC1CH	;DEL GATO 1 ES INFERIOR AL LÍMITE SUPERIOR

```

INC1      GOTO   SCA1      ;EN CASO DE SOBERPASO VERIFICA LA PARTE BAJA
          MOVFW  A1ICH   ;VERIFICA SI LA PARTE ALTA VALOR DE CARGA
          CPFSGT AC1CH   ;ES MENOR AL LÍMITE INFERIOR
          GOTO   ICA1      ;EN CASO DE SOBREPASO VERIFICA LA PARTE BAJA
SUD1      MOVFW  A1SDH   ;VERIFICA LA PARTE ALTA VALOR DESPLAZAMIENTO
          CPFSLT AC1DH   ;DEL GATO 1 SEA INFERIOR AL LÍMITE SUPERIOR
          GOTO   SDA1      ;EN CASO DE SOBERPASO VERIFICA LA PARTE BAJA
IND1      MOVFW  A1IDH   ;VERIFICA LA PARTE ALTA VALOR DESPALAZAMI-
          CPFSGT AC1DH   ;MIENTO SEA MENOR AL LÍMITE INFERIOR
          GOTO   IDA1      ;EN CASO DE SOBREPASO VERIFICA LA PARTE BAJA

SUC2      MOVFW  A2SCH   ;VERIFICA SI LA PARTE ALTA VALOR DE CARGA
          CPFSLT AC2CH   ;DEL GATO 2 ES INFERIOR AL LÍMITE SUPERIOR
          GOTO   SCA2      ;EN CASO DE SOBERPASO VERIFICA LA PARTE BAJA
INC2      MOVFW  A2ICH   ;VERIFICA SI LA PARTE ALTA VALOR DE CARGA
          CPFSGT AC2CH   ;ES MENOR AL LÍMITE INFERIOR
          GOTO   ICA2      ;EN CASO DE SOBREPASO VERIFICA LA PARTE BAJA
SUD2      MOVFW  A2SDH   ;VERIFICA LA PARTE ALTA VALOR DESPLAZAMIENTO
          CPFSLT AC2DH   ;DEL GATO 2 SEA INFERIOR AL LÍMITE SUPERIOR
          GOTO   SDA2      ;EN CASO DE SOBERPASO VERIFICA LA PARTE BAJA
IND2      MOVFW  A2IDH   ;VERIFICA LA PARTE ALTA VALOR DESPALAZAMI-
          CPFSGT AC2DH   ;MIENTO SEA MENOR AL LÍMITE INFERIOR
          GOTO   IDA2      ;EN CASO DE SOBREPASO VERIFICA LA PARTE BAJA

SUC3      MOVFW  A3SCH   ;VERIFICA SI LA PARTE ALTA VALOR DE CARGA
          CPFSLT AC3CH   ;DEL GATO 3 ES INFERIOR AL LÍMITE SUPERIOR
          GOTO   SCA3      ;EN CASO DE SOBERPASO VERIFICA LA PARTE BAJA
INC3      MOVFW  A3ICH   ;VERIFICA SI LA PARTE ALTA VALOR DE CARGA
          CPFSGT AC3CH   ;ES MENOR AL LÍMITE INFERIOR
          GOTO   ICA3      ;EN CASO DE SOBREPASO VERIFICA LA PARTE BAJA
SUD3      MOVFW  A3SDH   ;VERIFICA LA PARTE ALTA VALOR DESPLAZAMIENTO
          CPFSLT AC3DH   ;DEL GATO 3 SEA INFERIOR AL LÍMITE SUPERIOR
          GOTO   SDA3      ;EN CASO DE SOBERPASO VERIFICA LA PARTE BAJA
IND3      MOVFW  A3IDH   ;VERIFICA LA PARTE ALTA VALOR DESPALAZAMI-
          CPFSGT AC3DH   ;MIENTO SEA MENOR AL LÍMITE INFERIOR
          GOTO   IDA3      ;EN CASO DE SOBREPASO VERIFICA LA PARTE BAJA

SUC4      MOVFW  A4SCH   ;VERIFICA SI LA PARTE ALTA VALOR DE CARGA
          CPFSLT AC4CH   ;DEL GATO 4 ES INFERIOR AL LÍMITE SUPERIOR
          GOTO   SCA4      ;EN CASO DE SOBERPASO VERIFICA LA PARTE BAJA
INC4      MOVFW  A4ICH   ;VERIFICA SI LA PARTE ALTA VALOR DE CARGA
          CPFSGT AC4CH   ;ES MENOR AL LÍMITE INFERIOR
          GOTO   ICA4      ;EN CASO DE SOBREPASO VERIFICA LA PARTE BAJA
SUD4      MOVFW  A4SDH   ;VERIFICA LA PARTE ALTA VALOR DESPLAZAMIENTO
          CPFSLT AC4DH   ;DEL GATO 4 SEA INFERIOR AL LÍMITE SUPERIOR
          SDA4      ;EN CASO DE SOBERPASO VERIFICA LA PARTE BAJA
IND4      MOVFW  A4IDH   ;VERIFICA LA PARTE ALTA VALOR DESPALAZAMI-
          CPFSGT AC4DH   ;MIENTO SEA MENOR AL LÍMITE INFERIOR
          GOTO   IDA4      ;EN CASO DE SOBREPASO VERIFICA LA PARTE BAJA
          ;52 INSTRUCIONES DESDE LA LECTURA

SHOW      MOVFW  FALLO   ;MUESTRA LA PALBRA DE FALLO POR EL PUERTOD
          MOVWF  PORTD   ;

          BTFSC  PORTC,7 ;VERIFICA SI EL BIT DE PARO ESTA ACTIVADO
          CALL  PARO     ;
          BTFSC  PORTC,6 ;VERIFICA SI EL BIT DE LIMPIEAR EL REGISTRO
          CLRF  FALLO   ;FALLA ESTA ACTIVADO
          BTFSC  PORTC,5 ;VERIFICA SI SE QUIERE REASIGNAR UN NUEVO
          GOTO  SELIM   ;VALOR A UN LÍMITE
          GOTO  LECT    ;O SIGUE CON LA OPERACION NORMAL

```

END

,*****

```

INCLIM      MOVLW  0X01      ;INCREMENTA EN UNA UNIDAD EL LIMTE
            ADDWF  LIML      ;COMENZANDO POR LA PARTE BAJA DEL LÍMITE
            MOVWF  LIML      ;ALMACENA LA PALABRA
            MOVLW  0X00      ;SE SUMA EL ACARREO
            ADDWFC LIMH      ;A LA PARTE ALTA
            MOVWF  LIMH      ;LA PALABRA ES ALMACENADA
            BTFSS  LIMH,2    ;VERIFICA QUE EL LÍMITE NO PASE DE 10 BITS
            RETURN          ;Y SALE DE LA SUBROUTINA
            MOVLW  0XFF      ;SI EL LÍMITE HA SOBREPASADO LOS 10 BITS
            MOVWF  LIML      ;SE ALMACENA EL MAXIMO VALOR EN LA PARTE
            MOVLW  0X03      ;BAJA
            MOVWF  LIMH      ;Y ALTA
            RETURN          ;Y SALE DE LA SUBROUTINA

```

```

DECLIM      BCF      STATUS,4 ;LIMPA LA BANDERA N
            MOVLW  0X01      ;Y DECREMENTA EN UNA UNIDAD EL VALOR
            SUBWF  LIML,1    ;DE LA PARTE BAJA DEL NIVEL
            BTFSS  STATUS,4  ;VERIFICA SI EL RESULTADO NO ES NEGATIVO
            RETURN          ;SI NO ES NEGATIVO REGRESA AL PROGRAMA
            BCF      STATUS,4 ;LIMPIA LA BANDERA N
            MOVLW  0X01      ;Y DECREMENTA EN UNA UNIDAD EL VALOR
            SUBWF  LIMH,1    ;DE LA PARTE ALTA
            BTFSS  STATUS,4  ;VERIFICA SI EL RESULTADO NO ES NEGATIVO
            RETURN          ;SI NO ES NEGATIVO REGRESA AL PROGRAMA
            MOVLW  0X00      ;SI EL RESULTADO ES NEGATIVO
            MOVWF  LIML      ;LA PARTE BAJA SERA 0
            MOVWF  LIMH      ;Y LA PARTE ALTA SERA 0
            RETURN          ;Y REGRESA AL PROGRAMA

```

```

MAYLIM      INCF      LIMTS    ;INCREMENTA EN 1 EL VALOR DE SELECCION
            MOVLW  0X10      ;SI VERIFICA QUE EL LÍMITE MÁXIMO
            CPFSLT LIMTS    ;SELECCIONADO SEA 1111 "DESPLAZAMIENTO INF"
            GOTO   MAXLVL    ;EN ACT 4, DE CONTRARIO, ESTE SE DEBE FIJAR
            RETURN          ;REGRESA A LA RUTINA NORMAL

```

```

MAXLVL      MOVLW  0X0F      ;COLOCA EL VALOR DE 1111 "DEPLAZAMIENTO INF"
            MOVWF  LIMTS    ;COMO EL VALOR DE LIMTE SELECCIONADO
            GOTO   ML       ;REGRESA A LA RUTINA NORMAL

```

```

MENLIM      MOVLW  0X00      ;VERIFICA QUE EL MENOR LÍMITE SELECCIONADO
            CPFSEQ LIMTS    ;SEA 0000 "CARGA SUP ACT1"
            GOTO   DECRE    ;DE SER OTRO LÍMITE
            RETURN          ;SE PUEDE PROCEDER A DECREMENTAR

```

```

DECRE      DECF      LIMTS    ;LA SELECCIÓN.
            RETURN          ;

```

```

READ      MOVLW  0X01      ;SELECCIONA EL CANAL 1
            MOVWF  ADCONO    ;DEL CONVERTIDOR AD
AC1C      BTFSC  ADCON0,2    ;SE MANTIENEN HASTA
            GOTO   AC1C     ;QUE FINALICE LA CONVERCION
            MOVFW  ADRESL    ;GUARDA EL REGISTRO BAJO DE
            MOVWF  AC1CL    ;CARGA DEL ACTUADOR 1
            MOVFW  ARDESH    ;GUARDA EL REGISTRO ALTO DE
            MOVWF  AC2CH    ;CARGA DEL ACTUADOR 1
            MOVLW  0X09      ;SELECCIONA EL CANAL 2
            MOVWF  ADCONO    ;DEL CONVERTIDOR AD
AC1D      BTFSC  ADCON0,2    ;SE MANTIENEN HASTA
            GOTO   ACID     ;QUE FINALICE LA CONVERCION
            MOVFW  ADRESL    ;GUARDA EL REGISTRO BAJO DE
            MOVWF  AC1DL    ;DESPLAZAMIENTO DEL ACTUADOR 1
            MOVFW  ARDESH    ;GUARDA EL REGISTRO ALTO DE

```

```

MOVWF AC1DH ;DESPLAZAMIENTO DEL ACTUADOR 1
MOVLW 0X11 ;SELECCIONA EL CANAL 3
MOVWF ADCONO ;DEL CONVERTIDOR AD
AC2C BTFSC ADCON0,2 ;SE MANTIENEN HASTA
GOTO AC2C ;QUE FINALICE LA CONVERCION
MOVFW ADRESL ;GUARDA EL REGISTRO BAJO DE
MOVWF AC2CL ;CARGA DEL ACTUADOR 2
MOVFW ARDESH ;GUARDA EL REGISTRO ALTO DE
MOVWF AC2CH ;CARGA DEL ACTUADOR 2
MOVLW 0X19 ;SELECCIONA EL CANAL 4
MOVWF ADCONO ;DEL CONVERTIDOR AD
AC2D BTFSC ADCON0,2 ;SE MANTIENEN HASTA
GOTO AC2D ;QUE FINALICE LA CONVERCION
MOVFW ADRESL ;GUARDA EL REGISTRO BAJO DE
MOVWF AC2DL ;DESPLAZAMIENTO DEL ACTUADOR 2
MOVFW ARDESH ;GUARDA EL REGISTRO ALTO DE
MOVWF AC2DH ;DESPLAZAMIENTO DEL ACTUADOR 2
MOVLW 0X21 ;SELECCIONA EL CANAL 5
MOVWF ADCONO ;DEL CONVERTIDOR AD
AC3C BTFSC ADCON0,2 ;SE MANTIENEN HASTA
GOTO AC3C ;QUE FINALICE LA CONVERCION
MOVFW ADRESL ;GUARDA EL REGISTRO BAJO DE
MOVWF AC3CL ;CARGA DEL ACTUADOR 3
MOVFW ARDES ;GUARDA EL REGISTRO ALTO DE
MOVWF AC3CH ;CARGA DEL ACTUADOR 3
MOVLW 0X29 ;SELECCIONA EL CANAL 6
MOVWF ADCONO ;DEL CONVERTIDOR AD
AC3D BTFSC ADCON0,2 ;SE MANTIENEN HASTA
GOTO AC3D ;QUE FINALICE LA CONVERCION
MOVFW ADRESL ;GUARDA EL REGISTRO BAJO DE
MOVWF AC3DL ;DESPLAZAMIENTO DEL ACTUADOR 3
MOVFW ARDESH ;GUARDA EL REGISTRO ALTO DE
MOVWF AC3DH ;DESPLAZAMIENTO DEL ACTUADOR 3
MOVLW 0X31 ;SELECCIONA EL CANAL 7
MOVWF ADCONO ;DEL CONVERTIDOR AD
AC4C BTFSC ADCON0,2 ;SE MANTIENEN HASTA
GOTO AC4C ;QUE FINALICE LA CONVERCION
MOVFW ADRESL ;GUARDA EL REGISTRO BAJO DE
MOVWF AC4CL ;CARGA DEL ACTUADOR 4
MOVFW ARDESH ;GUARDA EL REGISTRO ALTO DE
MOVWF AC4CH ;CARGA DEL ACTUADOR 4
MOVLW 0X39 ;SELECCIONA EL CANAL 8
MOVWF ADCONO ;DEL CONVERTIDOR AD
AC4D BTFSC ADCON0,2 ;SE MANTIENEN HASTA
GOTO AC4D ;QUE FINALICE LA CONVERCION
MOVFW ADRESL ;GUARDA EL REGISTRO BAJO DE
MOVWF AC4DL ;DESPLAZAMIENTO DEL ACTUADOR 4
MOVFW ARDESH ;GUARDA EL REGISTRO ALTO DE
MOVWF AC4DH ;DESPLAZAMIENTO DEL ACTUADOR 4

```

RETURN

;66 INSTRUCCIONES 8 ENTRADAS

,*****

```

SCA1 MOVFW A1SCL ;VERIFICA SI LA PARTE BAJA DE VALOR DE CARGA
CPFSLT AC1CL ;DEL GATO 1 ES MAYOR AL LIM.INF. SUPERIOR
GOTO FSCA1 ;EN CASO DE SOBERPASO ESCRIBE LA FALLA
FSCA1 GOTO INC1 ;SI NO REGERSA A LA SIGUIENTE COMPARACION
MOVLW OX01 ;ESCRIBE EL NÚMERO CORRESPONDIENTE DE LA
IORWF FALLO ;FALLA EN EL REGISTRO CORRESPONDIENTE
MOVWF FALLO ;SIN ALTERAR LOS ANTERIORES ESTADOS
GOTO INC1 ;REGRESA A LA COMPRACAIION SIGUIENTE.

ICA1 MOVFW A1ICL ;VERIFICA SI LA PARTE BAJA DE VALOR DE CARGA
CPFSGT AC1CL ;DEL GATO 1 ES MENOR AL LIM.INF. INFERIOR
GOTO FICA1 ;EN CASO DE SOBERPASO ESCRIBE LA FALLA

```

FICA1	GOTO SUD1 MOVLW OXO2 IORWF FALLO MOVWF FALLO GOTO SUD1	;SI NO REGERSA A LA SIGUIENTE COMPARACION ;ESCRIBE EL NÚMERO CORRESPONDIENTE DE LA ;FALLA EN EL REGISTRO CORRESPONDIENTE ;SIN ALETRAR LOS ANTERIORES ESTADOS ;REGRESA A LA COMPRACAION SIGUIENTE.
SDA1	MOVFW A1SDL CPFSLT AC1DL GOTO FSDA1 GOTO IND1	;VERIFICA SI LA PARTE BAJA DE VALOR DE DESP. ;DEL GATO 1 ES MAYOR AL LIM.INF. SUPERIOR ;EN CASO DE SOBERPASO ESCRIBE LA FALLA ;SI NO REGERSA A LA SIGUIENTE COMPARACION
FSDA1	MOVLW OXO3 IORWF FALLO MOVWF FALLO GOTO IND1	;ESCRIBE EL NÚMERO CORRESPONDIENTE DE LA ;FALLA EN EL REGISTRO CORRESPONDIENTE ;SIN ALTERAR LOS ANTERIORES ESTADOS ;REGRESA A LA COMPRACAION SIGUIENTE.
IDA1	MOVFW A1IDL CPFSGT AC1DL GOTO FIDA1 GOTO SUC2	;VERIFICA SI LA PARTE BAJA DE VALOR DE DESP. ;DEL GATO 1 ES MENOR AL LIM.INF. INFERIOR ;EN CASO DE SOBERPASO ESCRIBE LA FALLA ;SI NO REGERSA A LA SIGUIENTE COMPARACION
FIDA1	MOVLW OXO4 IORWF FALLO MOVWF FALLO GOTO SUC2	;ESCRIBE EL NÚMERO CORRESPONDIENTE DE LA ;FALLA EN EL REGISTRO CORRESPONDIENTE ;SIN ALETRAR LOS ANTERIORES ESTADOS ;REGRESA A LA COMPRACAION SIGUIENTE.
SCA2	MOVFW A2SCL CPFSLT AC2CL GOTO FSCA2 GOTO INC2	;VERIFICA SI LA PARTE BAJA DE VALOR DE CARGA ;DEL GATO 2 ES MAYOR AL LIM.INF. SUPERIOR ;EN CASO DE SOBERPASO ESCRIBE LA FALLA ;SI NO REGERSA A LA SIGUIENTE COMPARACION
FSCA2	MOVLW OXO5 IORWF FALLO MOVWF FALLO GOTO INC2	;ESCRIBE EL NÚMERO CORRESPONDIENTE DE LA ;FALLA EN EL REGISTRO CORRESPONDIENTE ;SIN ALTERAR LOS ANTERIORES ESTADOS ;REGRESA A LA COMPRACAION SIGUIENTE.
ICA2	MOVFW A2ICL CPFSGT AC2CL GOTO FICA2 GOTO SUD2	;VERIFICA SI LA PARTE BAJA DE VALOR DE CARGA ;DEL GATO 2 ES MENOR AL LIM.INF. INFERIOR ;EN CASO DE SOBERPASO ESCRIBE LA FALLA ;SI NO REGERSA A LA SIGUIENTE COMPARACION
FICA2	MOVLW OXO6 IORWF FALLO MOVWF FALLO GOTO SUD2	;ESCRIBE EL NÚMERO CORRESPONDIENTE DE LA ;FALLA EN EL REGISTRO CORRESPONDIENTE ;SIN ALETRAR LOS ANTERIORES ESTADOS ;REGRESA A LA COMPRACAION SIGUIENTE.
SDA2	MOVFW A2SDL CPFSLT AC2DL GOTO FSDA2 GOTO IND2	;VERIFICA SI LA PARTE BAJA DE VALOR DE DESP. ;DEL GATO 2 ES MAYOR AL LIM.INF. SUPERIOR ;EN CASO DE SOBERPASO ESCRIBE LA FALLA ;SI NO REGERSA A LA SIGUIENTE COMPARACION
FSDA2	MOVLW OXO7 IORWF FALLO MOVWF FALLO GOTO IND2	;ESCRIBE EL NÚMERO CORRESPONDIENTE DE LA ;FALLA EN EL REGISTRO CORRESPONDIENTE ;SIN ALTERAR LOS ANTERIORES ESTADOS ;REGRESA A LA COMPRACAION SIGUIENTE.
IDA2	MOVFW A2IDL CPFSGT AC2DL GOTO FIDA2 GOTO SUC3	;VERIFICA SI LA PARTE BAJA DE VALOR DE DESP. ;DEL GATO 2 ES MENOR AL LIM.INF. INFERIOR ;EN CASO DE SOBERPASO ESCRIBE LA FALLA ;SI NO REGERSA A LA SIGUIENTE COMPARACION
FIDA2	MOVLW OXO8 IORWF FALLO MOVWF FALLO GOTO SUC3	;ESCRIBE EL NÚMERO CORRESPONDIENTE DE LA ;FALLA EN EL REGISTRO CORRESPONDIENTE ;SIN ALETRAR LOS ANTERIORES ESTADOS ;REGRESA A LA COMPRACAION SIGUIENTE.
SCA3	MOVFW A3SCL CPFSLT AC3CL GOTO FSCA3 GOTO INC3	;VERIFICA SI LA PARTE BAJA DE VALOR DE CARGA ;DEL GATO 3 ES MAYOR AL LIM.INF. SUPERIOR ;EN CASO DE SOBERPASO ESCRIBE LA FALLA ;SI NO REGERSA A LA SIGUIENTE COMPARACION
FSCA3	MOVLW OXO9	;ESCRIBE EL NÚMERO CORRESPONDIENTE DE LA

	IORWF	FALLO	;FALLA EN EL REGISTRO CORRESPONDIENTE
	MOVWF	FALLO	;SIN ALTERAR LOS ANTERIORES ESTADOS
	GOTO	INC3	;REGRESA A LA COMPRACAION SIGUIENTE.
ICA3	MOVFW	A3ICL	;VERIFICA SI LA PARTE BAJA DE VALOR DE CARGA
	CPFSGT	AC3CL	;DEL GATO 3 ES MENOR AL LIM.INF. INFERIOR
	GOTO	FICA3	;EN CASO DE SOBERPASO ESCRIBE LA FALLA
	GOTO	SUD3	;SI NO REGERSA A LA SIGUIENTE COMPARACION
FICA3	MOVLW	OXOA	;ESCRIBE EL NÚMERO CORRESPONDIENTE DE LA
	IORWF	FALLO	;FALLA EN EL REGISTRO CORRESPONDIENTE
	MOVWF	FALLO	;SIN ALETRAR LOS ANTERIORES ESTADOS
	GOTO	SUD3	;REGRESA A LA COMPRACAION SIGUIENTE.
SDA3	MOVFW	A3SDL	;VERIFICA SI LA PARTE BAJA DE VALOR DE DESP.
	CPFSLT	AC3DL	;DEL GATO 3 ES MAYOR AL LIM.INF. SUPERIOR
	GOTO	FSDA3	;EN CASO DE SOBERPASO ESCRIBE LA FALLA
	GOTO	IND3	;SI NO REGERSA A LA SIGUIENTE COMPARACION
FSDA3	MOVLW	OXOB	;ESCRIBE EL NÚMERO CORRESPONDIENTE DE LA
	IORWF	FALLO	;FALLA EN EL REGISTRO CORRESPONDIENTE
	MOVWF	FALLO	;SIN ALTERAR LOS ANTERIORES ESTADOS
	GOTO	IND3	;REGRESA A LA COMPRACAION SIGUIENTE.
IDA3	MOVFW	A3IDL	;VERIFICA SI LA PARTE BAJA DE VALOR DE DESP.
	CPFSGT	AC3DL	;DEL GATO 3 ES MENOR AL LIM.INF. INFERIOR
	GOTO	FIDA3	;EN CASO DE SOBERPASO ESCRIBE LA FALLA
	GOTO	SUC4	;SI NO REGERSA A LA SIGUIENTE COMPARACION
FIDA3	MOVLW	OXOC	;ESCRIBE EL NÚMERO CORRESPONDIENTE DE LA
	IORWF	FALLO	;FALLA EN EL REGISTRO CORRESPONDIENTE
	MOVWF	FALLO	;SIN ALETRAR LOS ANTERIORES ESTADOS
	GOTO	SUC4	;REGRESA A LA COMPRACAION SIGUIENTE.
SCA4	MOVFW	A4SCL	;VERIFICA SI LA PARTE BAJA DE VALOR DE CARGA
	CPFSLT	AC4CL	;DEL GATO 4 ES MAYOR AL LIM.INF. SUPERIOR
	GOTO	FSCA4	;EN CASO DE SOBERPASO ESCRIBE LA FALLA
	GOTO	INC4	;SI NO REGERSA A LA SIGUIENTE COMPARACION
FSCA4	MOVLW	OXOD	;ESCRIBE EL NÚMERO CORRESPONDIENTE DE LA
	IORWF	FALLO	;FALLA EN EL REGISTRO CORRESPONDIENTE
	MOVWF	FALLO	;SIN ALTERAR LOS ANTERIORES ESTADOS
	GOTO	INC4	;REGRESA A LA COMPRACAION SIGUIENTE.
ICA4	MOVFW	A4ICL	;VERIFICA SI LA PARTE BAJA DE VALOR DE CARGA
	CPFSGT	AC4CL	;DEL GATO 4 ES MENOR AL LIM.INF. INFERIOR
	GOTO	FICA4	;EN CASO DE SOBERPASO ESCRIBE LA FALLA
	GOTO	SUD4	;SI NO REGERSA A LA SIGUIENTE COMPARACION
FICA4	MOVLW	OXOE	;ESCRIBE EL NÚMERO CORRESPONDIENTE DE LA
	IORWF	FALLO	;FALLA EN EL REGISTRO CORRESPONDIENTE
	MOVWF	FALLO	;SIN ALETRAR LOS ANTERIORES ESTADOS
	GOTO	SUD4	;REGRESA A LA COMPRACAION SIGUIENTE.
SDA4	MOVFW	A4SDL	;VERIFICA SI LA PARTE BAJA DE VALOR DE DESP.
	CPFSLT	AC4DL	;DEL GATO 4 ES MAYOR AL LIM.INF. SUPERIOR
	GOTO	FSDA4	;EN CASO DE SOBERPASO ESCRIBE LA FALLA
	GOTO	IND4	;SI NO REGERSA A LA SIGUIENTE COMPARACION
FSDA4	MOVLW	OXOF	;ESCRIBE EL NÚMERO CORRESPONDIENTE DE LA
	IORWF	FALLO	;FALLA EN EL REGISTRO CORRESPONDIENTE
	MOVWF	FALLO	;SIN ALTERAR LOS ANTERIORES ESTADOS
	GOTO	IND4	;REGRESA A LA COMPRACAION SIGUIENTE.
IDA4	MOVFW	A4IDL	;VERIFICA SI LA PARTE BAJA DE VALOR DE DESP.
	CPFSGT	AC4DL	;DEL GATO 4 ES MENOR AL LIM.INF. INFERIOR
	GOTO	FIDA4	;EN CASO DE SOBERPASO ESCRIBE LA FALLA
	GOTO	SHOW	;SI NO REGERSA A LA SIGUIENTE COMPARACION
FIDA4	MOVLW	OX10	;ESCRIBE EL NÚMERO CORRESPONDIENTE DE LA
	IORWF	FALLO	;FALLA EN EL REGISTRO CORRESPONDIENTE
	MOVWF	FALLO	;SIN ALETRAR LOS ANTERIORES ESTADOS
	GOTO	SHOW	;REGRESA A LA COMPRACAION SIGUIENTE.

```

-*****
,
PARO      MOVLW 0x00      ;VERIFICA QUE EXISTA UNA FALLA
          CPFSEQ FALLAS  ;
          RET           ;
          BSF  PORTB,0   ;ACTIVA LE BIT DE PARO
LOPP      BTFSS PORTB,3  ;Y PERMANECE AHI HASTA QUE SE
          GOTO LOOP     ;HASTA QUE ACTIVE EL BIT 3 DE PORTB
          CLRF FALLA    ;LIMPIA EL REGISTRO DE FALLA
          BCF  PORTB,0   ;LIMPIA LE BIT DE PARO
          GOTO INICIO    ;VA AL INICIO

```

9. RESULTADOS

Se implementó el equipo analógico para el dispositivo de proporcionalidad de carga y el analógico-digital para el dispositivo de detección de límites en operación, debido al menor tiempo en desarrollo de ambos. No pudiendo hacer un comparativo físico contra su respectivo dispositivo alternativo.

No obstante, el sistema ATS operó bajo las condiciones preestablecidas de carga, cumpliendo con la proporcionalidad pedida por el usuario para sus pruebas. Y el dispositivo de detección de límites desempeñó efectivamente su función, inclusive en la primera prueba realizada con el dispositivo, los cálculos de trabajo para el sistema ATS implicaron una sobre carga la cual fue detectada y además del aviso de ésta, se interrumpió la función del sistema.

Se pudiera esperar el mismo éxito con los dispositivos alternativos, pues desarrollo entre una y otra solución parte de un mismo planeamiento de solución.

9.1 Dispositivo de detección de límites.

En dispositivo analógico- digital la captura y detección de sobrepasos está a cargo de la parte analógica. Por lo que se podría decir que el tiempo de respuesta es inmediato.

En el microcontrolador la velocidad de ejecución de un comando es de 1 a $3\mu\text{s}$ dependiendo la instrucción y condiciones implícitas para ésta.

La parte digital tiene la siguiente forma:

Tarda **$3\mu\text{s}$** entre la lectura y la muestra de la falla.

Lleva $4\mu\text{s}$ si ninguna de las entradas de paro está activada y $5\mu\text{s}$ si alguna está activada

Cuando se procede a la verificación de los paros que están activados toma $2\mu\text{s}$ la verificación si no es el paro indicado y $3\mu\text{s}$ si lo es. Toma $4\mu\text{s}$ más si el fallo no ocurrió ahí y $7\mu\text{s}$ si se activa el paro.

Se verifican los cuatro bits de control de paro por lo que el máximo tiempo se presenta cuando la falla ocurre en el último actuador y se verificaron los tres gatos anteriores, dando un total de $31\mu\text{s}$ hasta llegar la habilitación del bit de paro.

Dando un total teórico de:

$$3\mu\text{s}(\text{de lectura y muestra}) + 5\mu\text{s}(\text{interruptor de paro}) + 31\mu\text{s}(\text{verificación falla y activacion de paro}) = \mathbf{39\mu\text{s}}$$

Si no hubiese ningún paro activado, el tiempo que toma en terminarse un ciclo de programa completo es de:

$$3\mu\text{s}(\text{de lectura y muestra}) + 4\mu\text{s}(\text{interruptor de paro}) + 2\mu\text{s}(\text{Limpieza del registro}) + 4\mu\text{s}(\text{conmutación de luz de salida}) + 152\mu\text{s}(\text{cuando esta apagado}) \text{ ó } 4\mu\text{s}(\text{cuando esta encendido}) = \mathbf{165\mu\text{s}(\text{apagado}) / 17\mu\text{s}(\text{encendido})}$$

El encendido y apagado significa, la forma en que se muestra la falla a la salida, se le da una intermitencia a la salida, cuando no es un paro, para darle una menor intensidad a la señal luminosa; una vez que se presenta un paro del equipo, la señal es activada permanentemente, con lo que se presenta una intensidad luminosa mayor.

Cuando uno de los interruptores de paro esta activados, pero no se presenta falla alguna el tiempo es de:

$$3\mu\text{s}(\text{de lectura y muestra}) + 5\mu\text{s}(\text{interruptor de paro}) + 30\mu\text{s}(\text{verificación falla y retorno}) + 2\mu\text{s}(\text{Limpieza del registro}) + 4\mu\text{s}(\text{conmutación de luz de salida}) + 152\mu\text{s}(\text{cuando esta apagado}) \text{ ó } 4\mu\text{s}(\text{cuando esta encendido}) = \mathbf{195\mu\text{s}(\text{apagado}) / 47\mu\text{s}(\text{encendido})}$$

En el sistema que totalmente digital al igual que el otro microcontrolador la velocidad de ejecución de un comando es de 1 a 3 μs .

Para el caso del dispositivo de detección de límites digital ya una vez puesto en marcha; es decir, sin la parte de la carga de valores de los límites, se cuneta con la siguiente estructura.

El tiempo de respuesta variará dependiendo de las condiciones de entrada.

La ejecución de la lectura de un canal analógico lleva 9 μs en comandos y 12.86 μs de tiempo de conversión, dando un tiempo de 21.86 μs por señal. En total la lectura tomará 174.88 μs

La verificación con cada límite toma 3 μs si la comparación entre las partes altas de la señal y el límite no cumplen la condición establecida, 10 μs si las partes altas cumplen pero las bajas no y de 13 μs si se presenta una falla.

Se verifican los 16 límites pero el caso extremo solo puede arrojar 8 condiciones para que se genere falla y por consiguiente las otras 8 solo tomarán tiempos de 3 μs , dando como máximo un tiempo de 136 μs

Dos microsegundos para desplegar la falla en el puerto B y siete más para activar el bit de paro.

Así que teóricamente:

- Dando como resultado un tiempo máximo de muestra de la falla de:

174.88 μ s(de lectura)+136 μ s(comprobación con 8 fallas)+2 μ s(muestra de la falla)= **312.88 μ s**

- Dando como resultado un tiempo máximo de paro del ATS:

$$312.88\mu\text{s} + 7\mu\text{s}(\text{activación del bit de paro}) = \mathbf{319.88\mu\text{s}}$$

- Para el caso cuando no se presente falla alguna y paro, el ciclo sistema estará censando con el siguiente periodo:

$$174.88\mu\text{s}(\text{de lectura}) + 48\mu\text{s}(\text{sin sobrepasos}) + 2\mu\text{s}(\text{muestra de la falla}) + 7\mu\text{s}(\text{verificación de paro}) + 2\mu\text{s}(\text{limpieza o no al registro falla}) + 2\mu\text{s}(\text{salto configuración de límite}) + 2\mu\text{s}(\text{regreso a lectura}) = \mathbf{214.88\mu\text{s}}$$

Cabe mencionar algunas diferencias que resaltan entre uno y otro dispositivo:

- Se requiere de dos dispositivos analógicos digitales para la detección de los límites, uno para los de carga y otro para los de desplazamiento. Mientras que se requiere solamente de uno digital.
- En el sistema analógico-digital la selección de paro puede ser individual, es decir, un interruptor por actuador, mientras que en el digital el paro es general.
- La intermitencia de la señal de salida solo se presenta para el dispositivo analógico-digital. En caso de no hacer este modo de funcionamiento, el periodo de lectura del dispositivo caería hasta 42 μ s aproximadamente.

No obstante el principio de funcionamiento es el mismo. Se establecen límites de operación, se detectan sobrepasos a los límites, se comunica al usuario que límite fue el rebasado y se hace un paro del sistema si así fue previsto.

Claramente se pudo observar la mayor rapidez de respuesta del sistema analógico-digital sobre el puramente digital, pues la etapa de captura de los valores analógicos representa una un incremento representativo en éste.

Sin embargo un muestreo mayor a 3kHz es aceptable para el sistema ATS, porque éste es el tiempo de respuesta del módulo de control.

La resolución está dada en el sistema digital por señal y el número de bits del convertidor. La señal analógica de entrada es de 5V y se tiene un DAC de 10bits, por lo que la resolución es:

$$5\text{V} / 1024 = 4.882\text{mV}$$

Que representan 0.09765 de tonelada por cada incremento de un bit para el actuador de 50T y de 0.1952 en los de 100 T. Así los límites que se establezcan tendrán que ajustarse al valor más aproximado dado por esta resolución.

Y no obstante de tratar de ser un sistema puramente digital se requirió de amplificadores operacionales para la etapa de acoplamiento de señales.

En el sistema analógico-digital la resolución es infinita y aquí se depende del mecanismo analógico que ajuste el valor, en nuestro caso se utilizaron potenciómetros de precisión con diales numéricos.

Los potenciómetros eran de 10 vueltas, con los extremos a +/- 10V, por lo que el derivado central oscilará entre estos dos valores, que cuando el dial marca 5, se obtienen 0 Volts en el derivado central.

Para la fijación de obtiene de la siguiente ecuación.

$$5/x = \text{valor máximo de carga o desplazamiento/limite}$$

Donde x es el valor que se adicionará o restará al dial. Ejemplo, si deseo colocar como valor límite de carga un valor de -20T en un actuador de 50T, la ecuación quedaría de la siguiente manera.

$$\frac{5}{x} = \frac{50T}{-20T}$$

$$x = \frac{5(-20T)}{50T}$$

$$x = -2$$

Por lo que hay que restarle 2 al dial del potenciómetro colocando su valor en 3 y así queda establecido el límite.



9.2 Dispositivo de proporcionalidad de carga.

El sistema analógico de proporción de carga tiene un tiempo de respuesta prácticamente instantáneo, mientras que el dispositivo digital teóricamente operaría así:

- Le tomaría 87.44µs en la lectura de las de las cuatro señales de carga.
- 7µs para la suma de las cargas horizontales.
- 14µs para el acomodo de los valores de la suma.

- 8µs en la resta del offset si el resultado es positivo y 9µs si el resultado es negativo, para el tratamiento de un resultado negativo se suman 10µs más.
- Luego son 17µs en la multiplicación.
- 36µs en el acomodo de las registros producto de la multiplicación.
- 9µs suma de la proporción sin no hay saturación, da haberla la operación lleva 7µs más.
- 16µs para la inversión y resta de la proporción cuando no hay saturación, cuando exista una se adicionan 5µs más.
- 8µs inversión del valor del act3 para realizar la resta contra la señal objetivo
- 8µs de la resta si el resultado no es negativo y 5µs más cuando el resultado sea negativo.
- 4µs en la colocación del valor en el PWM1
- 8µs inversión del valor del act4 para realizar la resta contra la señal objetivo
- 8µs de la resta si el resultado no es negativo y 5µs más cuando el resultado sea negativo.
- 4µs en la colocación del valor en el PWM2
- 2µs para el regreso a la lectura de los actuadores.

Con esto se puede estimar un tiempo de **236.44µs** en la respuesta cuando la suma de cargas horizontales es positiva, y de **247.44µs** cuando es negativa.

Se incrementará el periodo en **7µs** cuando exista una saturación superior, **5µs** con una saturación inferior, **5µs** si la resta entre la señal del actuador 3 y el objetivo fuera negativa y **5µs** si la si ocurriera lo mismo en el actuador 4.

El caso extremo nos entregaría un tiempo de ejecución del programa de **269.44µs** que representa una frecuencia de 3.711 kHz, cumpliendo con el requerimiento de una frecuencia mayor al dado por la respuesta del módulo de control del equipo.

De igual manera que en el sistema anterior la resolución esta dada por señal y el número de bits del convertidor. La señal analógica de entrada es de 5V y se tiene un DAC de 10bits, por lo que la resolución es:

$$5V / 1024 = 4.882mV$$

Representando 0.09765 de tonelada por cada incremento de un bit para el actuador de 50T. Los valores objetivo se fijarán en múltiplos de ésta resolución. Y como en la multiplicación por alfa solo se toma la parte entera del resultado, que es un redondeo por truncamiento, puede que en algunos momentos el valor objetivo sea menor al deseado en la prueba. A su vez el valor de alfa está fijado por la resolución dada para el sistema, que era de 0.0625 con un formato $Q_i=4$ en una palabra de 8 bits, lo que dejaba como valores más próximos a los calculados:

$$0.8125 < \text{alfa} < 15.9375$$

10. CONCLUSIONES

Con base en los resultados arrojados en el capítulo anterior, se llega a la conclusión de que todos los equipos desarrollados cumplen con las especificaciones y condiciones de uso deseadas.

Para el dispositivo de detección de límites analógico-digital contando las instrucciones de programa y sabiendo el tiempo que tardan en realizarse cada una, los tiempos dados son de $39\mu\text{s}$ desde la toma de la muestra hasta que se genera la señal de paro, un ciclo de 165 ó $17\mu\text{s}$ entre toma de muestras si no hay interruptor alguno de paro (la variación es producto del encendido y apagado del led para generar la menor intensidad de luz en éste cuando solo se trata de un aviso) y 195 ó $47\mu\text{s}$ en el tiempo de muestreo para cuando hay al menos un interruptor de paro activado.

De igual manera para el sistema totalmente digital se contabilizaron las instrucciones y se obtuvieron tiempos de $312.88\mu\text{s}$ para la muestra de una falla a partir del muestreo, $319.88\mu\text{s}$ para el paro del ATS y de $214.88\mu\text{s}$ entre la toma de muestras cuando no hay paro alguno.

El módulo de control del ATS tiene una frecuencia de respuesta de 3kHz ($333.33\mu\text{s}$), y todas las respuestas generadas por los sistemas diseñados tienen una frecuencia mayor.

Se implementó el dispositivo analógico-digital obteniéndose los resultados esperados de su funcionamiento.



El módulo consta de tres estructuras sobresalientes, la primera es donde se localizan los interruptores que indicarán si el actuador esta trabajando en la modalidad de paro o aviso, son ocho interruptores, uno de carga y otro desplazamiento por cada actuador.



En la siguiente estructura se localizan cuatro potenci6metros por actuador para la selecci6n de los l6mites (carga y desplazamiento superior e inferior) y los dos leds de aviso.



Y es en la última parte donde se encuentra el botón de reset y el paro de emergencia.



En el dispositivo de proporcionalidad de carga el sistema analógico es considerado un sistema de parámetros concentrados en el cual no importando su tamaño el tiempo entre señal de entrada y salida es prácticamente nulo. Tiene un valor del factor alfa entre 0.8 y 16, con una resolución “infinita”.

En el caso del dispositivo digital los tiempos son de **236.44 μ s** en la respuesta cuando la suma de cargas horizontales es positiva, y de **247.44 μ s** cuando es negativa.

La máxima ejecución de rutina de programa nos entregaría un tiempo de 269.44 μ s que representa una frecuencia de 3.711 kHz, cumpliendo con el requerimiento de una frecuencia mayor al dado por la respuesta del módulo de control del equipo. Respecto al valor del factor alfa, este se encontrará entre 0.8125 y 15.9375 con una resolución de 0.0625.

Finalmente se implementó el dispositivo analógico por ser más sencillo de ensamblar y presentar ventajas como el tiempo de respuesta “inmediato” y resolución “infinita” del factor alfa.

En cuanto tiempo de respuesta de los dispositivos desarrollados para el ATS todos cumplen con un tiempo menor al que del módulo de control del equipo, por lo que para este caso, ambos sistemas resultan adecuados para la implementación. Un incremento de la frecuencia de respuesta del módulo de control del equipo, implicaría para los dispositivos digitales, en principio un cambio en la estructura del programa tratando de reducir el tiempo entre muestreos y respuesta, y de no alcanzar lo deseado se requeriría de un procesador más rápido.

Para la lectura de las señales la resolución los sistemas digitales desarrollados cumplen con los parámetros requeridos en principio y solo dependerá del usuario si ésta es aceptable para sus futuras pruebas, en caso de no satisfacer las necesidades del usuario, se requiere de un cambio de microcontrolador por uno que presente la resolución requerida. En el caso del valor de alfa, se requiere de un cambio en la lógica del programa, y un incremento en tamaño de programa que representa un incremento en memoria de programa, que algunas ocasiones implica un cambio de microcontrolador.

Con respecto al dispositivo analógico, para mejorar la resolución, solo se requerirá de un dispositivo que presente con mayor exactitud el valor de la resistencia del potenciómetro, tanto para el dispositivo de detección de límites analógico-digital como para el alfa del dispositivo de proporción de cargas.

Se hace uso de un microcontrolador para el control de un equipo, de no ser que se requiera controlar un mayor número de equipos mediante un solo dispositivo, no es recomendable hacer el circuito integrado.

Los componentes analógicos siguen haciéndose presentes para el acondicionamiento de las señales a la entrada y salida, por lo que las soluciones digitales desarrolladas no implican reducción en el número de componentes, por lo tanto no existe un cambio drástico en el tamaño del dispositivo.

Los dispositivos analógicos presentan la ventaja de un tiempo de respuesta “inmediato” y una resolución “infinita” lo que hace más fácil su implementación en un sistema similar al ATS, o futuras pruebas bajo otras condiciones. Mientras que los dispositivos digitales requerirán de un mayor análisis y tratamiento para su adaptación.

Sin embargo, el dispositivo de detección de límites analógico-digital es una clara muestra de cómo un sistema híbrido puede ser la mejor solución. Mientras que para el sistema digital, la lectura y comparación de las señales implica una parte considerable del atraso en la respuesta, el micro representa un ahorro en espacio y componentes en la parte de muestra de fallas y activación de paro.

Lo que demuestra que tanto los sistemas analógicos como los digitales cuentan con sus ventajas y lo conveniente es sacar provecho a ambos para un diseño óptimo.

BIBLOGRAFIA.

1. FRANCO SERGIO. [Libro] Diseño con amplificadores operacionales y circuitos integrados analógicos. Tercera edición. Lugar, Mc Graw-Hill. Año
2. MATA H. GLORIA, SÁNCHEZ E VICTOR, GOMEZ G. MANUEL. [Libro] Análisis de sistemas y señales con computo avanzado, México, Facultad de Ingeniería, UNAM, 2002.
3. DOMÍNGUEZ G. ANTONIO. JIMÉNEZ R. JOSE. SÁNCHEZ B. ALFREDO. Problemas y cuestiones de tecnología industrial. Principio de servosistemas[En línea] Junta de Andalucía, <www.juntadeandalucia.es/averroes/ies_sierra_magina/d_tecnologia/LIBRO/pdf/servopri.pdf >
4. WIKIPEDIA, Microcontrolador,[En línea] Última modificación 6 de Marzo del 2008 <<http://es.wikipedia.org/wiki/Microcontrolador>>
5. MICROCHIP, Pic18fxx39 Data Sheet, Pic. [En Línea]. <http://www.hitech.com.au/resources/pic18_datasheets.php#PIC18F4439>
6. MICROCHIP, Pic16f87X Data Sheet, Pic. [En Línea]. <http://www.eupmt.es/cra/inform/16F87X_Data_Sheet.pdf >