



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

Facultad de Ingeniería

CONTROL DE LAS FUNCIONES DE ROTACION Y ELEVACION
DE LOS SENSORES ULTRASONICOS DE LA
HERRAMIENTA SONIMPII

T E S I S
Que para obtener el título de
INGENIERO MECANICO ELECTRICISTA,
AREA ELECTRICA - ELECTRONICA
p r e s e n t a
MARIZA LUNA HERRERA



Director M. C. Isaac Omar López Vázquez

Codirector M. I. Lauro Santiago Cruz

México D. F.

2000

280364



Universidad Nacional
Autónoma de México



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

Agradezco:

A la UNAM y al IMP por haberme dado un lugar para desarrollarme como profesionista.

Todo el esfuerzo, trabajo y dedicación de mis directores de Tesis, Isaac Omar López Vazquez y Lauro Santiago Cruz.

A todos los maestros de la Facultad de Ingeniería, por su trabajo, esfuerzo y por todo el cúmulo de conocimientos que nos brindaron en sus clases con *mucho entusiasmo*.

Dedico este trabajo al ser humano mas bello e importante en mi vida, mi madre Angélica Herrera Rendón, por preocuparse, desvelarse, cuidarme y acompañarme en esta loca carrera de la vida.

A mis hermanos Fernando, Hermila y Asdrual, a mis Abuelitos que siempre me dieron su aliento, su amor y su apoyo.

A todos mis amigos de trabajo del IMP, que me brindaron su amistad, apoyo y compañía, durante el desarrollo de mi tesis, en especial a Alberto Flores Roa, por ofrecerme su paciencia, su tiempo y apoyo incondicional.

A Eric Rodriguez Tovar, que me dió todo su amor, todo su tiempo y el apoyo incondicional, para tener la fuerza de voluntad y el coraje para concluir mi tesis profesional.

A Dios que me enseñó que las batallas de la vida no las gana el más fuerte ni el más débil, sino el que decide ganarlas.

TABLAS	VII
ILUSTRACIONES	VIII
INTRODUCCIÓN	X

CAPITULO I.- " CONCEPTOS BÁSICOS"

1.1	Ultrasonido.....	1
1.1.1	Enlace básico de transmisión ultrasónica	1
1.1.2	Reflexión y refracción en las ondas sonoras en una Frontera entre dos materiales	2
1.1.3	Técnica de reflexión de pulso o de eco de pulso	4
1.1.4	Sensores	6
1.2	Características generales de los microcontroladores	9
1.2.1	Estructura funcional del microcontrolador	9
1.2.2	Instrucciones básicas del microcontrolador	11
1.3	Motores a pasos	14
1.3.1	Características esenciales del motor a pasos	15
1.3.2	Tipos de motores a pasos	15
1.3.3	Manejador de motores a pasos.....	16
1.3.4	Modos de excitación de los motores a pasos	16
1.3.5	Modos de manejadores de motores a pasos	17
1.3.6	Motores a pasos de la herramienta SONIMPII	17

CAPITULO II. - " SISTEMA SONIMPII"

2.1	Antecedente	20
2.2	Descripción general de la herramienta SONIMPII	20
2.3	Equipo de superficie	21
2.4	Equipo de fondo	23
2.5	Teoría de operación de la herramienta SONIMPII	26
2.5.1	Secuencia general de tiempos	27
2.5.2	Transmisión de datos	28
2.5.3	Ventana Sónica	28
2.5.4	Recepción de datos	29
2.6	Ordenes hacia la herramienta	29
2.7	Datos hacia la superficie	30
2.8	Procedimiento general para efectuar un registro sonar	31
2.9	Descripción general del cartucho electrónico	33
2.10	Descripción y funcionamiento del sistema de orientación del cartucho electrónico	34

2.10.1	Control de Rotación.....	35
2.10.2	Control de Elevación.....	36

CAPITULO III.-

" DISEÑO Y DESARROLLO DEL SISTEMA DE ORIENTACIÓN DEL CARTUCHO ELECTRÓNICO "

3.1	Descripción y análisis del sistema original de orientación del cartucho electrónico.....	38
3.1.1	Señales de entrada y salida de la sección Control de Rotación.....	39
3.1.2	Señales de entrada y salida de la sección Control de Elevación y Excitación de motores.....	46
3.1.3	Estructura interna del programa original de Rotación.....	48
3.1.4	Estructura interna del programa original de Elevación.....	55
3.2	Diseño y desarrollo del sistema de orientación modificado del cartucho electrónico.....	57
3.2.1	Fusión del sistema de orientación del cartucho Electrónico.....	59
3.2.2	Selección del microcontrolador.....	61
3.3	Constitución física del sistema de orientación modificado del cartucho electrónico.....	65
3.3.1	Sección A del sistema de orientación modificado.....	66
3.3.2	Sección B del sistema de orientación modificado.....	71
3.4	Desarrollo de algoritmos de control y diagramas de flujo del sistema de orientación.....	78
3.4.1	Rutinas de interrupción del programa Rot_Elev.....	87
3.5	Construcción de las tarjetas electrónicas del sistema de orientación modificado.....	90
3.6	Localización de fallas.....	92

CAPITULO IV.- " PRUEBAS DE LABORATORIO Y RESULTADOS "

4.1	Pruebas preliminares del sistema de orientación modificado.....	95
4.2	Pruebas finales del sistema de orientación modificado.....	100

INDICE

CONCLUSIONES.....	105
BIBLIOGRAFÍA.....	106
APÉNDICE A.....	A-0
A.1 Especificaciones técnicas de la herramienta SONIMPIL.....	A-1
A.2 Lista de componentes electrónicos de la sección A del sistema de orientación modificado.....	A-2
A.3 Lista de componentes electrónicos de la sección B del sistema de orientación modificado.....	A-3
A.4 Especificaciones técnicas del MCU AT90S8515.....	A-4
APÉNDICE B.....	B-0
B.1 Diagramas de flujo del programa Rot_Elev.....	B-1
APÉNDICE C.....	C-0
C.1 Listado del programa en lenguaje ensamblador Rot_Elev..	C-1
C.2 Listado del programa en C.....	C-19

3-1 Señales de entrada y salida de la sección Control de Rotación.....	46
3-2 Señales de entrada y salida de las secciones Control de Elevación y Excitación de motores.....	49
3-3 Ventajas y desventajas consideradas al fusionar el sistema de orientación.....	58
3-4 Señales de entrada y salida del sistema de orientación modificado.....	61
3-5 Características generales de algunos MCU's fabricados por Microchip.....	62
3-6 Características generales de algunos MCU's fabricados por Motorola.....	62
3-7 Características generales de algunos MCU's fabricados por Intel.....	63
3-8 Características generales de algunos MCU's fabricados por Atmel.....	63
3-9 Secuencia de las señales de salida del circuito SAA1027.....	73
3-10 <i>Parámetros de diseño de circuitos de conmutación para transistores en configuración emisor común</i>	74
3-11 Estados de los bits de control del registro SOLIPULSR.....	87
3-12 Estados de los bits de control del registro SOLIPULSE.....	88

Figura 1-1 Enlace básico de transmisión ultrasónica	2
Figura 1-2 Reflexión y refracción del sonido en una frontera de dos materiales.....	3
Figura 1-3 (a) Efecto de transmisión del pulso.....	5
Figura 1-4 Voltaje Hall, generado en la dirección Z, correspondiente a una corriente de excitación en X y un campo magnético en Y.....	7
Figura 1-5 Sensor de efecto Hall.....	8
Figura 1-6 Microinterruptor 11SX21-T.....	8
Figura 1-7 Estructura funcional del microcontrolador.....	10
Figura 1-8 Ciclo de instrucción de una arquitectura CISC.....	13
Figura 1-9 Ciclo de instrucción de una arquitectura RISC.....	14
Figura 1-10 Pulsos de entrada y las condiciones de los cambios de corriente en cada fase de excitación.....	17
Figura 1-11 Motor a pasos que efectúa el movimiento de rotación.....	18
Figura 1-12 Motor a pasos que efectúa el movimiento de elevación.....	19
Figura 2-1 Configuración general para la toma de un registro sonar.....	22
Figura 2-2 Componentes que integran el sistema de control, adquisición e interpretación de datos.....	23
Figura 2-3 Componentes que integran el equipo de fondo del sistema SONIMP II.....	24
Figura 2-4 Movimientos de rotación y elevación de la sonda mecánica.....	25
Figura 2-5 Diagrama a bloques de la herramienta de fondo.....	26
Figura 2-6 Ciclo de operación de la herramienta SONIMP II.....	28
Figura 2-7 Formato de los comandos hacia la herramienta.....	30
Figura 2-8 Formato de los datos enviados hacia la superficie.....	31
Figura 2-9 Gráfica tridimensional de una cavidad de almacenamiento.....	32
Figura 2-10 Partes que integran el cartucho electrónico.....	33
Figura 2-11 Diagrama a bloques de las funciones de control de rotación y elevación.....	35
Figura 3-1 Diagrama electrónico de la Sección Control de Rotación.....	40
Figura 3-2 Diagrama electrónico de la Sección Control de Elevación y Excitación de motores.....	41
Figura 3-3 Bus de datos bidireccional del sistema de control de Rotación.....	42
Figura 3-4 Sistema de interrupción de la sección Control de Rotación.....	44
Figura 3-5 Señales de entrada y salida al circuito monoestable de la señal AZS.....	45
Figura 3-6 Sistema de interrupción de la sección Control de Elevación.....	48
Figura 3-7 Diagrama de flujo de la estructura interna del programa de rotación...	50
Figura 3-8 Diagrama de flujo de la estructura interna de las rutinas de inicialización del programa de rotación.....	51
Figura 3-9 Localización del cero de referencia Zin.....	52
Figura 3-10 Diagrama de flujo de la estructura interna del programa principal de rotación.....	53
Figura 3-11 Diagrama de flujo de la rutina de Envío de pasos y sentido de giro de la estructura interna del programa de rotación.....	54

Figura 3-12 Diagrama de flujo de la estructura interna del programa de elevación.....	55
Figura 3-13 Localización del cero de referencia de la señal SW1.....	56
Figura 3-14 Flujo de datos hacia arriba y hacia abajo a través del bus.....	60
Figura 3-15 Diagrama a bloques de la Constitución física del sistema de orientación modificado del cartucho electrónico.....	66
Figura 3-16 Diagrama electrónico de la sección A del sistema de orientación modificado.....	67
Figura 3-17 Diagrama electrónico de la sección B del sistema de orientación modificado.....	72
Figura 3-18 Gráfica para determinar la resistencia de polarización R_x del circuito SAA1027.....	73
Figura 3-19 Configuración electrónica del circuito conmutador emisor común empleado para el cálculo de R_B y R_C	75
Figura 3-20 Circuitos de excitación de los motores.....	77
Figura 3-21 Diagramas de flujo del diseño total del sistema de orientación.....	79
Figura 3-22 Eliminación del ciclo principal de rotación y el ciclo principal de elevación.....	81
Figura 3-23 Señales de excitación obtenidas al realizarse un sobreflujo en el contador 0.....	82
Figura 3-24 Estructura interna del programa Rot_Elev.....	83
Figura 3-25 Rutinas de interrupción del programa Rot_Elev.....	89
Figura 3-26 Conectores de la tarjeta de Control de Rotación y sección A del sistema de orientación modificado.....	91
Figura 3-27 Conectores de la tarjeta Control de Elevación y Excitación de motores y de la sección B del sistema de orientación modificado.....	92
Figura 3-28 Dimensiones de las tarjetas de la sección A y sección B del sistema de orientación modificado.....	93
Figura 3-29 Tarjetas electrónicas del sistema de orientación modificado.....	94
Figura 4-1 Elementos utilizados en las pruebas preliminares.....	95
Figura 4-2 (a) Configuración de las terminales físicas del conector del puerto paralelo.....	97
Figura 4-2 (b) Señales de salidas generadas por el puerto paralelo.....	97
Figura 4-3 Señales de entrada al conector del simulador Sonda Mecánica.....	98
Figura 4-4 Elementos utilizados en el desarrollo de las pruebas finales.....	101
Figura 4-5 Circuito de preencendido.....	103
Figura 4-6 Sistema de orientación modificado montado en el cartucho electrónico de la herramienta SONIMPIL.....	104

INTRODUCCIÓN.

El Instituto Mexicano del Petróleo (IMP) ha apoyado tecnológicamente a Petróleos Mexicanos (PEMEX) en la realización de diversos proyectos relacionados a la industria petrolera.

Dentro de estos proyectos está el sistema SONIMPI, desarrollado por el Grupo de Instrumentación de Sistemas Artificiales de Producción (GISAP) que pertenece a la Gerencia de Geofísica del Instituto Mexicano del Petróleo.

Debido a que PEMEX necesitaba incrementar la capacidad de almacenamiento de hidrocarburo, le encomendó al IMP en el año de 1988 el desarrollo de un sistema de almacenamiento capaz de almacenar grandes volúmenes de petróleo crudo, ante esta problemática el IMP propuso un sistema subterráneo de almacenamiento construyendo cavidades en los domos salinos, mediante el proceso de lixiviación (disolución de la sal mediante el agua dulce).

El sistema SONIMPI es un sistema de medición que mediante la toma de registros sonares nos permite obtener la forma, las dimensiones y el volumen de las cavernas subterráneas en domos salinos durante la construcción y explotación de las cavidades de almacenamiento. Se basa en el principio del sonar para la determinación de las dimensiones en cualquier punto de la caverna.

El sistema SONIMPI fue modificado para seguir tomando registros sonares en las cavidades en explotación a través de la tubería de producción, surgiendo la herramienta SONIMP II, segunda versión del sistema.

El problema al que nos enfrentamos es aumentar el tiempo de vida útil de la herramienta SONIMP II, garantizando el funcionamiento óptimo del sistema de orientación del cartucho electrónico de dicha herramienta, ya que de no funcionar la herramienta, PEMEX tendría que contratar el servicio a empresas extranjeras, provocando un costo mayor en la realización de los registros. El cartucho electrónico es la parte de la herramienta que se encarga de generar, transmitir y recibir las señales ultrasónicas, así como el control de los motores a pasos para mover el transductor ultrasónico principal.

Hoy en día los dispositivos electrónicos avanzan aceleradamente debido al crecimiento de los sistemas de información y de comunicaciones, a la necesidad de contar con productos electrónicos más sofisticados, ocasionando que en un tiempo corto se hagan obsoletos.

Los dispositivos electrónicos que se usan en el sistema de orientación del cartucho electrónico de la herramienta SONIMP II, no son la excepción, algunos de ellos ya son obsoletos.

Los microcontroladores MC1468705G2 de la familia Motorola, se encuentran en el sistema de orientación del cartucho electrónico y se encargan de realizar el control total de las funciones de rotación y elevación del transductor ultrasónico principal,

éstos dispositivos ya no se encuentran en el mercado, si los microcontroladores se dañan, ya no se puede hacer el remplazo de los mismos, por lo que es necesario modificar el sistema de orientación con dispositivos electrónicos actuales que nos permitan seguir controlando dichos movimientos.

El objetivo que se pretende alcanzar durante el desarrollo de este trabajo es actualizar y optimizar el sistema de orientación del cartucho electrónico con productos de tecnología de vanguardia, que nos permita seguir utilizando la herramienta, y por consiguiente garantizar su funcionamiento.

El contenido de este trabajo abarca los siguientes temas:

- En el capítulo I se da una breve introducción de los dispositivos electrónicos principales y conceptos básicos de los términos empleados en esta tesis, tales como: ultrasonido, sensores, el microcontrolador y los motores a pasos.
- En el capítulo II se da una descripción general de las partes que integran la herramienta SONIMPII, su funcionamiento general y su teoría de operación. También se da una descripción más detallada del funcionamiento del sistema original de orientación del cartucho electrónico.
- En el capítulo III se explica la forma en la que se llevó a cabo el diseño y la implementación del sistema de orientación modificado del cartucho electrónico, así como también la constitución física y el desarrollo de los algoritmos de control de dicho sistema.
- En el capítulo IV se describen las pruebas realizadas y los resultados obtenidos del sistema de orientación modificado.
- Finalmente se presenta la bibliografía consultada y los apéndices generados durante el desarrollo de este trabajo.

CAPITULO I

CONCEPTOS BÁSICOS

En este capítulo se describen los conceptos básicos de los dispositivos electrónicos principales y de los términos empleados en el desarrollo de esta tesis, como son: ultrasonido, sensores y transductores, el microcontrolador y sus características que lo definen, motores a pasos y sus modos de excitación.

1.1 Ultrasonido

Ultrasonido, se refiere a las ondas sonoras a frecuencias mayores a las que corresponden a las ondas de frecuencia dentro del alcance del oído humano, es decir, a frecuencias superiores a los 20 kHz. Las ondas ultrasónicas se basan en las mismas leyes básicas del movimiento ondulatorio que las ondas sonoras a frecuencias más bajas, a diferencia de éstas, las ondas ultrasónicas poseen las siguientes ventajas:

- Las ondas de frecuencias más altas poseen longitudes de onda más cortas, por lo que es más fácil dirigir y enfocar un haz ultrasónico.
- Las ondas ultrasónicas pueden atravesar las paredes metálicas de tubos y recipientes, esto significa que el sistema ultrasónico de medición puede montarse externamente al fluido, sobre todo aquellos con propiedades corrosivas, radiactivas, explosivas o inflamables.
- El ultrasonido puede emitirse y propagarse a través de tejido biológico, por lo que se emplea en aplicaciones en el campo de la medicina.
- El oído humano no puede percibir el sonido de las ondas ultrasónicas, por lo que se aprovecha en aplicaciones militares por su silencio.

1.1.1 Enlace básico de transmisión ultrasónica

La base de cualquier sistema ultrasónico de medición se muestra en la figura 1.1, está formado por un transmisor ultrasónico, el medio de transmisión y un receptor ultrasónico. Los dispositivos de uso común para los transmisores y receptores son sensores de tipo piezoeléctrico, cuyo efecto piezoeléctrico es reversible, es decir, la energía mecánica puede convertirse en energía eléctrica y viceversa. El transmisor ultrasónico utiliza el efecto piezoeléctrico inverso (ecuación 1.1), y se excita con un voltaje sinusoidal V_s , entonces el cristal de transmisión experimenta una deformación sinusoidal correspondiente a la variable x , la vibración del cristal excita a las partículas del medio las cuáles se ponen en movimiento sinusoidal, agitándose y originando la vibración de otras partículas del medio, hasta llegar al otro extremo. Los desplazamientos sinusoidales de las partículas provocan una

presión o esfuerzo, que es detectada por el receptor ultrasónico, el cuál se comporta como un sensor de fuerza que utiliza el efecto piezoeléctrico directo (ecuación 1.2). La presión proveniente de los desplazamientos, origina una fuerza F sobre el área del cristal, produciendo así una carga q y una corriente i en función del tiempo.

Dicha corriente se aplica una carga de salida Z_L , lo cuál genera un voltaje de salida V_{SAL} .

Efecto inverso $x = dV$ (1.1)

Efecto directo $q = dF$ (1.2)

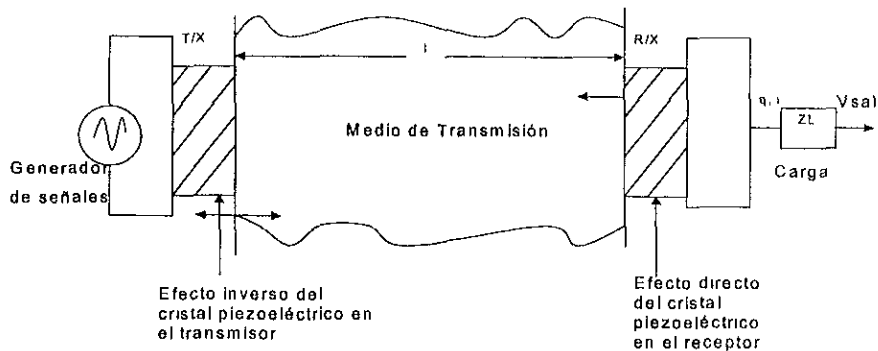


Figura 1-1. Enlace básico de transmisión ultrasónica.

1.1.2 Reflexión y refracción en las ondas sonoras en una frontera entre dos materiales

En la figura 1.2, podemos observar el fenómeno de reflexión y refracción del sonido en una frontera entre dos materiales. Se tiene una onda plana que incide en una frontera que separa dos materiales con diferentes impedancias características, parte de la onda incidente (I) se refleja (R) y parte de ella se transmite (T). Las leyes que relacionan los ángulos θ_i , θ_R , θ_T de incidencia, reflexión y transmisión son las mismas que para cualquier movimiento ondulatorio y se muestran en dicha figura. La potencia o intensidad W_R de la onda R sobre la potencia W_i de la onda incidente se le llama coeficiente de reflexión α_R (ecuación 1.4) y la razón de intensidad transmitida W_T sobre la intensidad incidente se le

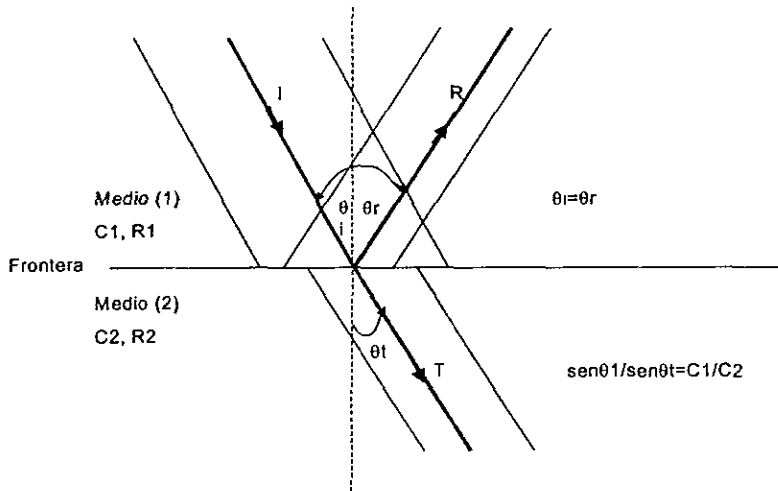


Figura 1.2. Reflexión y refracción del sonido en una frontera entre dos materiales.

llama coeficiente de transmisión α_T (ecuación 1.5). La impedancia característica de un medio está definida como el producto de la densidad ρ del medio y la velocidad del sonido c en dicho medio (expresión 1.3).

$$R_A = \rho c \quad (1.3)$$

$$\alpha_R = \frac{W_R}{W_i} \quad (1.4)$$

$$\alpha_T = \frac{W_T}{W_i} \quad (1.5)$$

Para ángulos de incidencia normal, donde los ángulos de incidencia, reflexión y transmisión son iguales a cero, se obtiene la siguiente expresión, en función de las impedancias características de cada medio:

$$\alpha_R = \frac{(R_2 - R_1)^2}{(R_2 + R_1)^2} \quad (1.6)$$

$$\alpha_T = \frac{4R_1 R_2}{(R_2 + R_1)^2} \quad (1.7)$$

De las expresiones 1.6 y 1.7¹ se puede obtener que la suma de los coeficientes es igual a uno, consecuencia de la conservación de la energía, y la suma de las potencias de la onda reflejante y de la onda de transmisión es igual a la potencia de la onda incidente. Si el valor de α_T es muy grande como es el caso para cuarzo/acero y acero/agua, significa que las ondas sonoras pueden emitirse desde transmisores de cristal de cuarzo a barras metálicas sólidas, y también a líquidos contenidos en tubos y recipientes metálicos. Si el valor de α_R es muy próximo a la unidad como es el caso para acero/aire y agua/aire (α_T es muy pequeño), significa que existe una reflexión casi perfecta de ondas sonoras en estas fronteras.

1.1.3 Técnica de reflexión de pulso o de eco de pulso

El principio de este método es empleado por el sistema SONIMP, para la determinación de los registros sonares en las cavidades de almacenamiento de hidrocarburos y se muestra en la figura 1.3(a). Este principio consiste de lo siguiente: el cristal piezoeléctrico emite un pulso ultrasónico en el material (1); si las impedancias características de los medios son sustancialmente diferentes, la mayor parte de la energía se reflejará en la frontera. El pulso reflejado se detecta por el cristal (que ahora actúa como receptor) y se mide el tiempo T_t entre los pulsos saliente y reflejado, donde T_t es el tiempo cuya distancia recorrida por el pulso es $2l$ y se puede definir por la siguiente expresión:

$$T_t = \frac{2l}{c} \quad (1.8)$$

Donde l es la distancia entre la interfaz del cristal y c es la velocidad del sonido en el material (1). Si se conoce c , puede calcularse l .

Durante el envío del pulso ultrasónico se presenta un fenómeno de reflexiones múltiples o ecos no deseados durante la medición. Parte del primer pulso reflejado se refleja en la frontera del material (1) y el cristal, y se refleja de nuevo en la frontera de los materiales (1) y (2), éste proceso se repite varias veces hasta desvanecerse la amplitud del pulso a causa de las pérdidas por atenuación en el material (1) y pérdidas por reflexión en las fronteras. En la figura 1.3(b), se muestra la señal observada durante el fenómeno de reflexiones múltiples del pulso ultrasónico.

Para lograr una medición confiable, la señal de pulso debe obedecer las siguientes condiciones:

- a) El ancho del pulso T_w debe ser mayor comparado con el periodo $1/f$ de la onda sonora:

$$T_w \gg 1/f \quad (1.9)$$

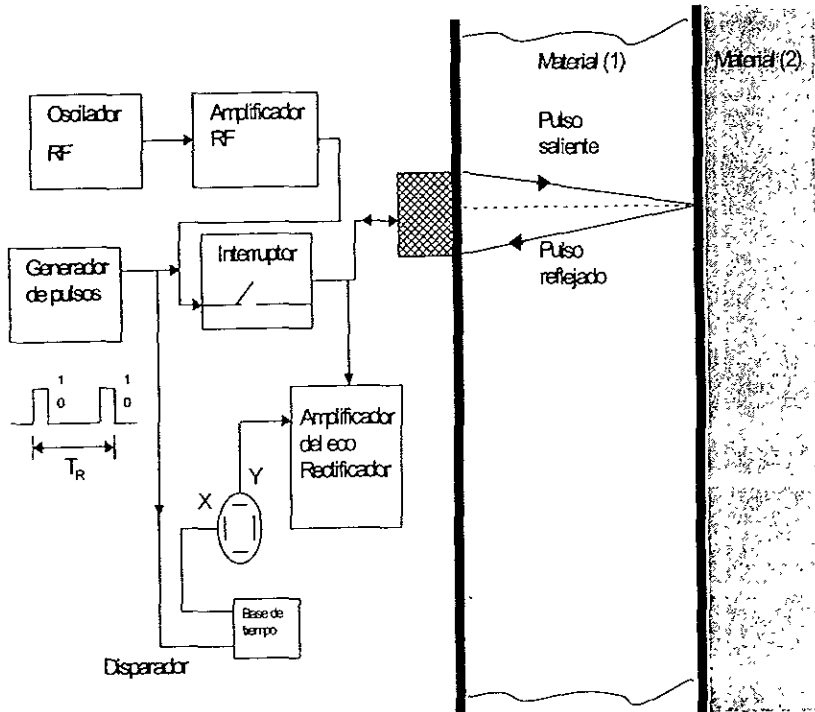
¹ Las expresiones 1.6 y 1.7 se obtuvieron del libro Sistemas de Medición, autor Jhon Bentley.

- b) El tiempo T_T de transición debe ser mayor comparado con el ancho del pulso T_W , para evitar interferencia entre pulsos salientes reflejados:

$$T_T \gg T_W \quad (1.10)$$

- c) El tiempo de repetición T_R entre los pulsos salientes sucesivos debe ser mayor comparado con el tiempo T_T , esto asegura que todas las reflexiones múltiples se atenúen antes de enviar el siguiente pulso saliente:

$$T_R \gg T_T \quad (1.11)$$



(a)

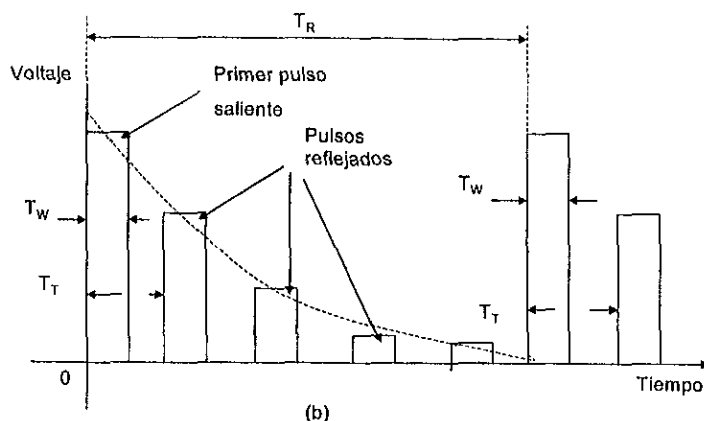


Figura 1-3. Técnica de reflexión de pulso. (a) Efecto de transmisión del pulso. (b) Señal observada en la reflexión de pulso.

1.1.4 Sensores

El sistema de orientación del cartucho electrónico de la herramienta SONIMP II, utiliza dos sensores de referencia, un sensor de efecto Hall y un microinterruptor, el primero se utiliza para generar una señal de índice cero con respecto al transductor horizontal, y el segundo se emplea para indicar que el transductor de elevación tiene una inclinación de 0° .

El sensor de efecto Hall es un sensor de estado sólido unipolar, modelo 103SR5A-1 fabricado por Honeywell, genera una tensión de salida de 5 voltios en presencia de un polo norte magnético.

El efecto Hall se genera cuando un campo magnético se aplica transversalmente a un material portador de corriente, obligando a los portadores móviles a acumularse en una cara del manejador. A consecuencia de esta acumulación se originará un campo eléctrico, representado esquemáticamente en la figura 1.4, donde la densidad de corriente J_x corresponde a la dirección x y la inducción magnética B tiene la dirección y . El coeficiente Hall R_H está definido para el semiconductor de tipo n por:

$$R_H = \frac{E_H}{J_x B_y} = \frac{1}{-en} \quad (1.12)$$

Donde n es la densidad de los electrones (m^{-3}) y e es la carga electrónica. Los semiconductores de tipo p presentan un coeficiente Hall positivo.

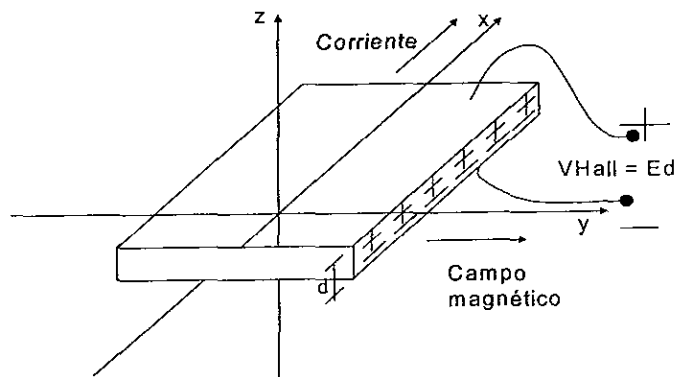


Figura 1.4. Voltaje Hall generado en la dirección z, correspondiente a una corriente de excitación en x y un campo magnético en y

El sensor de efecto hall, se encuentra en un encapsulado de aluminio enroscado para montarse mediante un tornillo, tiene un código de colores que nos permite realizar la conexión adecuada de sus terminales (figura 1.5). La terminal de salida es del tipo colector abierto, lo que hace necesario conectar una resistencia de carga activa, cuyo valor depende de la cargabilidad de salida, permitiendo el acoplamiento con circuitos digitales de familias como TTL, o CMOS.

El microinterruptor 11S21-T fabricado por Honeywell, es un pequeño interruptor mecánico, de bajo peso, con una gran capacidad eléctrica que se activa a presión, se encuentra cubierto de una resistente base de fenólica con un tiempo de vida de 1,000,000 operaciones, tiene tres terminales de metal, NC (normalmente cerrado), NA (normalmente abierto), y la terminal C (común) que permiten un rápido soldado en sus conexiones. Su capacidad eléctrica es de 5 amperios, 28 V en DC y 250 V en AC. En la figura 1.6 se muestra el microinterruptor 11S21-T.

Los datos técnicos de éstos sensores se muestran en la página de internet del fabricante (ver bibliografía).

Sensor de estado sólido, 103SR5A-1

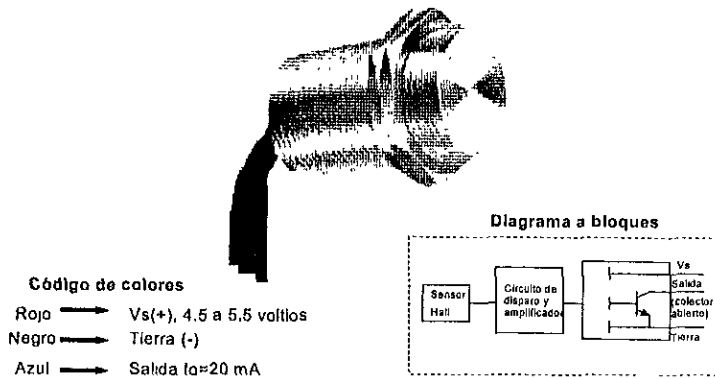


Figura 1.5. Sensor de efecto hall

Microinterruptor serie 11SM

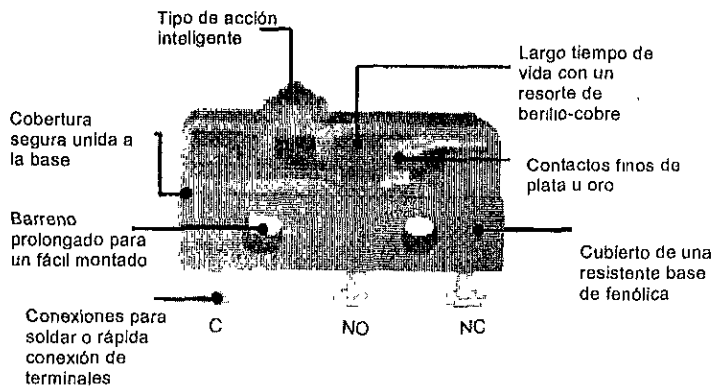


Figura 1.6.- Microinterruptor 11SX21-T

1.2 Características generales de los microcontroladores

El microcontrolador es un dispositivo electrónico digital que se define como una unidad de computadora pequeña (MCU: *Microcomputer Unit*) que está formado por una unidad central de procesamiento (*Central Processing Unit*: CPU), memorias para almacenar programas y datos, puertos bidireccionables de entrada y salida (E/S), temporizadores y contadores, sistemas de interrupción, puertos seriales, entre otras funciones, encapsulados en una sola pastilla. El MCU cuenta con un conjunto de instrucciones (software básico) que permiten interpretar programas diseñados para alguna aplicación específica escritos en lenguajes simbólicos directos (lenguaje ensamblador).

Los MCU's son dispositivos muy utilizados a nivel mundial en varias áreas de la industria, principalmente en la industria pesada, industria aeronáutica, en el campo de la medición y control, en el campo de la medicina.

Las características básicas que definen al MCU son: longitud de palabra (4, 8, 12, 16 bits), velocidad de ejecución de una instrucción, capacidad de memoria, número de instrucciones, número de acumuladores, número de registros de trabajo, modos de direccionamiento, capacidad de interrupción, capacidad de mandos periféricos.

1.2.1 Estructura funcional de los microcontroladores

Para comprender la estructura funcional del MCU, nos basaremos en el diagrama de la figura 1.7, que representa la estructura interna de un microcontrolador moderno mediante bloques funcionales. Los elementos funcionales del MCU son: memorias, unidades de proceso, unidades de control, unidades de entrada y salida, que se encuentran conectadas internamente a través de buses como son: el bus de direcciones, el bus de datos, el bus de programa, el bus de E/S y el bus de control.

La unidad de memoria: es el lugar donde se almacenan las instrucciones (memoria de programa) y los datos (memoria de datos), los buses de memoria de programa y datos se encuentran separados, esta característica define la arquitectura del MCU conocida como Harvard, cuya ventaja es acceder al dato y a la instrucción al mismo tiempo, a diferencia de los MCU's convencionales que utilizaban el mismo bus de datos. La unidad de memoria recibe una dirección a través del bus de direcciones y la envía al registro de direcciones de memoria, para realizar una operación de lectura o escritura de datos, ya sea obteniendo los datos desde el registro de datos de memoria, o bien, introduciendo datos en ella.

La unidad de proceso consta de una unidad aritmética y lógica (ALU), que realiza las operaciones aritméticas y lógicas con los datos de información que entran en ella mediante el bus de datos y el bus de direcciones.

CAPITULO I
CONCEPTOS BÁSICOS

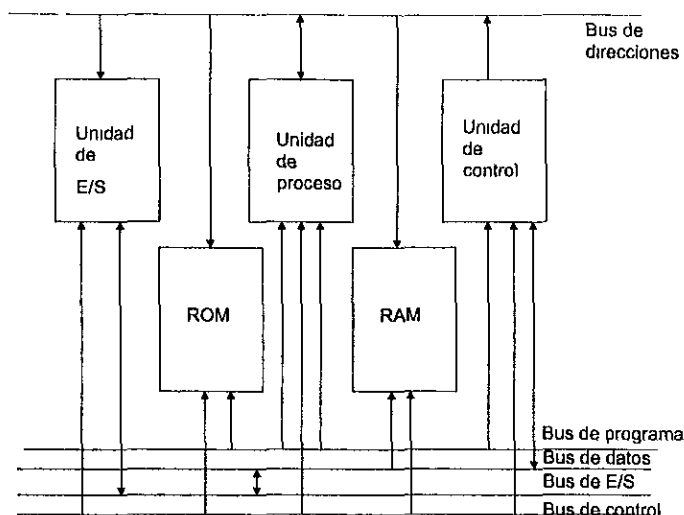


Figura 1-7. Estructura funcional del microcontrolador.

La unidad de control es el cerebro del sistema, en ella está contenido el contador del programa (PC) que lleva el conteo del programa y guarda la dirección de la instrucción que se está ejecutando; una pila (stack), que es un conjunto de posiciones de memoria que se pueden acceder de tal forma que el primero en salir es el último que entró, con su respectivo apuntador (stack pointer), que sirve para que la unidad de control lleve la cuenta de la posición de la pila. También se encuentra un registro de instrucciones en donde se almacenan las instrucciones obtenidas de la unidad de memoria; el decodificador de instrucciones (DI), que interpreta la información del código de operación de la instrucción y genera la secuencia de microinstrucciones.

El bus de direcciones, es el medio o camino por el cual pasan las direcciones proporcionadas por el contador de programa al realizar la búsqueda normal de alguna instrucción, o bien por un registro auxiliar proveniente de la unidad de control en caso de direccionamiento indirecto. La información contenida en el bus de direcciones se utiliza para direccionar celdas de memoria que contiene una instrucción o datos, registros de propósito general provenientes del ALU, o bien para direccionar el canal correspondiente a los puertos de E/S.

El bus datos, es el medio por el cual el flujo de información proporcionada por el registro de datos de memoria y por la unidad ALU, pasan a través de éste para comunicarse con las demás unidades del sistema, la información de los datos se

utiliza para obtener información del registro de datos de la memoria RAM, de los acumuladores, de los registros auxiliares de la unidad de control.

La información del Bus del programa las proporciona el registro de instrucción de memoria del programa en cada operación de búsqueda y dicha información se utiliza para leer o escribir en el contenido de esta memoria.

El bus de E/S, son puertos generalmente de 8 bits programables, que permiten el acceso de datos y salida para comunicarse con el medio exterior, están formados por circuitos de multiplexaje, que son seleccionados por medio del bus de direcciones y el bus de control para saber cuál o que dato se desea acceder o enviar.

1.2.2 Instrucciones básicas de los MCU's

El MCU requiere de un programa específico para el desarrollo de alguna aplicación particular, dicho programa se almacena en la memoria de programa en forma de códigos de máquina, para ejecutar el conjunto de instrucciones implementadas en él, cada instrucción se ejecuta secuencialmente en el orden en el que fueron introducidas. Existen varios tipos de instrucciones básicas en el MCU, que de acuerdo a su funcionalidad, número de palabras y ciclos de memoria que utiliza se pueden clasificar. La clasificación más general es por su funcionalidad y éstas pueden ser:

- a) Instrucciones relativas al acumulador, a los registros, incrementos, decrementos, desplazamientos a la izquierda/derecha, complementación.
- b) Instrucciones de intercambio de información entre memorias, registros y acumuladores.
- c) Instrucciones inmediatas de carga en registros, acumuladores, memoria.
- d) Instrucciones aritméticas y lógicas entre acumuladores o registros y memoria, sumas, restas, comparaciones, entre otras.
- e) Instrucciones de ruptura de secuencia, saltos condicionados, saltos a subrutinas.
- f) Instrucciones especiales y de control, puesta en operación, rutinas de interrupción, detener el sistema, entre otros.

Las instrucciones poseen un ciclo de instrucción, que es el tiempo de búsqueda para ejecutar una instrucción, está formado por dos tiempos, el tiempo de Búsqueda (fetch) y el tiempo de Ejecución. El tiempo de Búsqueda es el tiempo utilizado para almacenar en el registro de instrucción IR de la UC el contenido de

CAPITULO I
CONCEPTOS BÁSICOS

la instrucción que se va a ejecutar; el PC se posiciona en la dirección de memoria donde se encuentra la instrucción, se guarda su contenido en el IR y se incrementa en uno para señalar a la dirección de la siguiente instrucción. Antes de ejecutarse la instrucción se lee una parte de la instrucción que se conoce como el código de la operación y se transfiere al decodificador de instrucciones, para saber de qué tipo es la instrucción y cuantos ciclos de máquinas se requieren para ejecutarse. El tiempo del ciclo de ejecución depende del número de palabras de cada instrucción y del número de los ciclos de memoria que cada tipo de instrucción necesite ejecutar totalmente.

Los MCU's convencionales fueron diseñados bajo la arquitectura CISC (Complex Instructions Set Computer: Conjunto de Instrucciones Complejas de Computadora), en la cuál el procesador del MCU obtiene la primera instrucción, ejecuta la operación de ésta y se va a la siguiente instrucción en la memoria y así sucesivamente permaneciendo en estado ocioso la parte aritmética de la instrucción. Los MCU's modernos están diseñados con la arquitectura RISC (Reduce Instruction Set Computer: Conjunto de Instrucciones Reducidas de Computadora), en la cuál durante el tiempo de ejecución de la instrucción, el procesador está obteniendo la siguiente instrucción, incrementando un número de instrucciones que pueden ser ejecutadas durante un período de tiempo dado, está forma de ejecución de los ciclos de instrucción se conoce como pipeline (recubrir).

En la arquitectura CISC su filosofía de diseño es maximizar el trabajo realizado por instrucción. Sus características principales son:

- Pocos registros
- Pocos operandos disponibles rápidamente
- Modificación de datos en registro o memoria
- Muchos modos de direccionamiento en instrucciones redundantes
- Dificultad para seleccionar un código de programa rápido
- Código de máquina de longitud variable y formato variable
- Dificultad para decodificar y buscar

En la figura 1.8 se muestran los ciclos de instrucción empleados para una arquitectura CISC.

CAPITULO I
CONCEPTOS BÁSICOS

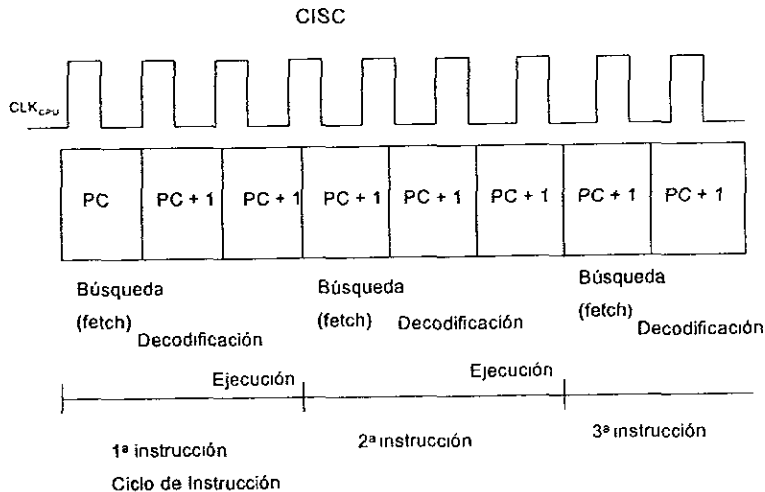


Figura 1.8.- Ciclo de instrucción de una arquitectura CISC.

En la arquitectura RISC su filosofía es minimizar ciclos de máquina por instrucción.

Sus características principales son:

- Muchos registros
- Muchos operandos disponibles rápidamente
- Modificación de datos únicamente en registros
- Pocos modos de direccionamiento y simples, no hay instrucciones redundantes
- Código rápido e *idealmente el mismo*
- Formato y longitud fija en el código de máquina
- Fácil búsqueda y decodificación

En la figura 1.9 se muestran los ciclos de instrucción empleado para una arquitectura RISC.

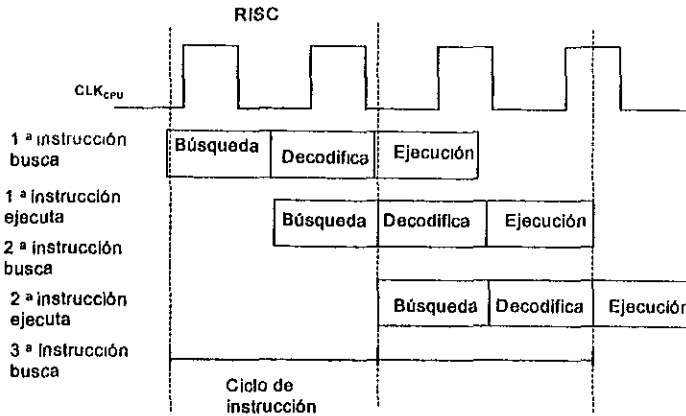


Figura 1.9.- Ciclo de instrucción de una arquitectura RISC.

Se requiere el uso de lenguajes de programación para el desarrollo de programas de aplicación del MCU, un programa es un conjunto de instrucciones que serán ejecutadas por el CPU del microcontrolador, una instrucción está formada por un conjunto de bits que son interpretados por la unidad de control. Los lenguajes de programación altamente utilizados en el diseño de sistemas basados en MCU's son los lenguajes orientados a las máquinas, conocidos como lenguajes ensambladores. El lenguaje ensamblador permite escribir programas utilizando símbolos (nemónicos) para designar operaciones, direcciones, y especificar datos constantes, los códigos de operación que emplea el lenguaje ensamblador no aparecen como números binarios, sino aparece indicado por un nombre asignado en función de las operaciones, y que es fácil de recordar. El programa desarrollado en lenguaje ensamblador pasa por un traductor (ensamblador) para convertirlo en lenguaje de máquina para que pueda ser ejecutado por el MCU.

1.3 Motores a pasos

El motor a pasos es un motor electromagnético que a partir de los cambios de excitación proporcionados por una señal de pulsos digitales gira a ángulos constantes. El motor a pasos no requiere de una retroalimentación para su control, es directamente controlado por la señal de pulsos, por lo que su circuito de control es muy simple. Debido a que resultan ser muy precisos y confiables, se emplean comúnmente en aplicaciones donde el posicionamiento resulta ser muy importante.

1.3.1 Características esenciales del motor a pasos

Las características que definen al motor a pasos son las siguientes:

- El ángulo de rotación del motor es directamente proporcional al número de pulsos de entrada.
- El ángulo de error por paso es despreciable y no acumulativo.
- Siempre necesitan de un circuito especial externo para controlarlo (*driver*: manejador) debido a que no se les puede conectar directamente a una fuente de alimentación.
- Grado por paso, este factor es el más importante al momento de escoger el motor para una aplicación. El grado por paso especifica el número de grados que el rotor girará por cada paso. Hay motores de 0.72°, 1.8°, 3.6°, 7.5°, 15° y hasta 90° por paso. Los grados por paso nos dan la resolución del motor.
- Resistencia por bobina, es una característica común de éstos motores, ya que determina la corriente que pasa por el motor, también como la curva de torque del motor y la máxima velocidad de operación.
- Capacidad de torque estático, que permite mantener al rotor en su posición cuando el motor no esta girando.
- Amplio rango de velocidades de giro proporcionales a la frecuencia de la señal de pulsos de entrada.

1.3.2 Tipos de motores a pasos

Los motores a pasos se clasifican en tres tipos según su estructura de construcción, y son:

- Reluctancia variable. Este tipo de motor cuenta con un rotor de multipolos de hierro suave, y un estator con embobinado laminado y gira cuando los dientes en el rotor son atraídos por los dientes del estator cuando se energizan electromagnéticamente. Generalmente el ángulo de paso es de 15°.
- Imán permanente. Este tipo de motor tiene un rotor magnetizado radialmente lo cual permite un torque de sostenimiento cuando el motor no está energizado. Hay dos tipo de imanes usados en la construcción del rotor, imán-alúnico y de ferrita. Los del imán-alúnico tienen ángulos de paso de 45° o 90°, los imanes de ferrita generalmente tienen ángulos de paso 7.5, 11.25, 15 y 18 grados.

- Híbrido. Este tipo de motor es una mezcla de los motores anteriores. El rotor y el estator tienen múltiples dientes, el rotor magnetizado radialmente está formado por un imán concéntrico alrededor de la flecha. Generalmente tienen una mayor precisión, un gran torque y los ángulos de paso son de 1.8° .

1.3.3 Manejador de motores a pasos

Un manejador de un motor a pasos genera las señales de excitación para provocar los cambios en el flujo de corriente suministrada en los embobinados del rotor. El manejador requiere de una señal de entrada de pulsos digitales, y genera la secuencia de las fases del motor, en cualquier sentido de giro (sentido de las manecillas del reloj o en contra). Dependiendo de las características del motor, el manejador es seleccionado de acuerdo a las características de capacidad de torque y velocidad del motor.

Existen tres formas para excitar los motores a pasos:

- Una fase.
- Dos fases.
- Una-dos fases.

1.3.4 Modos de excitación de los motores a pasos

- a) Una fase. En este modo de excitación los pulsos de corriente alimentan una fase a la vez para generar un paso. Tiene la ventaja de consumir poca energía y una buena precisión en el ángulo de fase. Sin embargo, la frecuencia de amortiguamiento es de valores muy grandes provocando resonancia en el motor, por lo que no es muy recomendable utilizarlo en aplicaciones donde el motor esté expuesto a fenómenos de vibración.
- b) Dos fases. En este modo de excitación los pulsos de corriente alimentan a dos fases el mismo tiempo para generar un paso. Tiene la ventaja de operar a un amplio rango de frecuencias, la frecuencia de amortiguamiento es muy baja, por lo que es el más utilizado.
- c) Una-dos fases. En este modo de excitación los pulsos de corriente alimenta una fase y dos fases a la vez, el valor de la corriente es 1.5 veces que la corriente suministrada en el modo de excitación de una fase. El ángulo de paso es dividido en dos y su rango de frecuencias de operación es dos veces más grande que los modos de excitación anteriores, por lo que su uso se recomienda para ángulos de paso muy pequeños.

CAPITULO I
CONCEPTOS BÁSICOS

Los pulsos de entrada y las condiciones de cambios de corriente en cada fase de los diferentes modos de excitación se muestran en la figura 1-10.

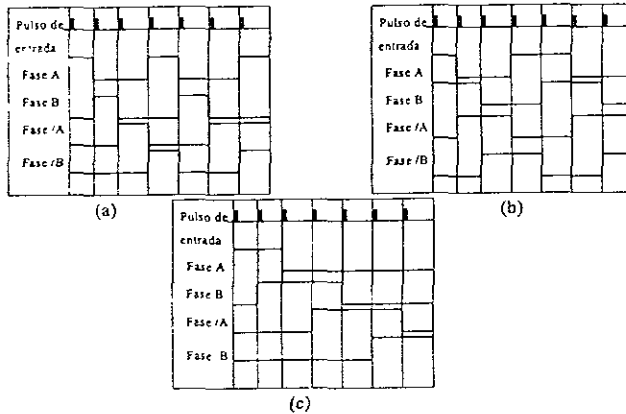


Figura 1-10. Los pulsos de entrada y las condiciones de cambios de corriente en cada fase de excitación (a) Una fase (b) Dos fases (c) Una-dos fases

1.3.5 Modos de Manejadores de motores a pasos

Hay dos modos de manejadores de motores a pasos, y se clasifican de acuerdo a la dirección del flujo de corriente a través de los embobinados del estator, el manejador unipolar y el manejador bipolar. El manejador unipolar el flujo de corriente es en un solo sentido, y es ampliamente usado por la sencillez de su configuración, en cambio el bipolar el flujo de corriente es de dos sentidos, y puede generar un gran torque de salida a bajas velocidades. El número de fases de los embobinados es la mitad con respecto al unipolar, por lo que el error del ángulo de paso generado por las resistencias por bobina y el apagado en las bobinas es pequeño. El costo del manejador bipolar es un poco más caro, debido a que requiere fuentes de voltaje para invertir la polaridad de la corriente de excitación.

1.3.6 Motores a pasos de la herramienta SONIMPIL

En la herramienta SONIMPIL se emplean dos motores a pasos para efectuar los movimientos de rotación y elevación de los transductores piezoeléctricos.

El motor de rotación del tipo imán permanente bipolar de cuatro fases, modelo PH268L-23, marca Vexta, se alimenta con 24 volts, 0.34 amperes por fase, una resistencia de embobinado por fase de 70.6 ohms/fase, una inductancia por fase de 124mH/fase, un torque de sostenimiento de 70.6 N-cm, una sola flecha, un

CAPITULO I
CONCEPTOS BÁSICOS

ángulo de paso de 1.8° y con un código de colores que se muestran en la figura 1-11.

El motor de elevación del tipo imán permanente bipolar de cuatro fases, modelo 4SH-24A56S, marca Airpax, se alimentan con 24 volts, 0.300 amperes por fase, resistencia de embobinado por fase 79 ohms/fase, inductancia por fase 150 mH/fase, torque de sostenimiento 600 N-m, ángulo de paso 1.8° , doble flecha, el motor de elevación se muestra en la figura 1-12.

Una vez definido algunos conceptos teóricos que nos servirán de apoyo para comprender la mayoría de los términos empleados durante el desarrollo de ésta tesis procederemos a describir y explicar el funcionamiento general de la herramienta SONIMPIL.

VEXTA PH268H-23

Código de colores

Fases	Color
Q1	Rojo
/Q1	Azul
Q2	Amarillo
/Q2	Café
24 V	Negro y Blanco

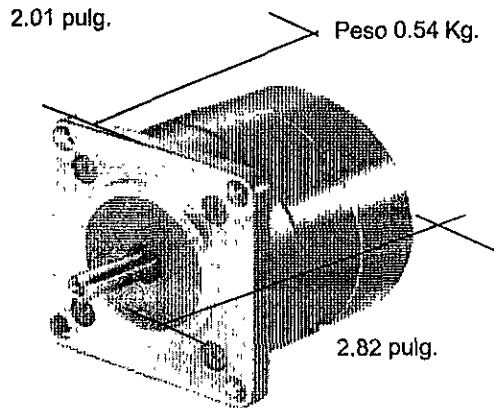
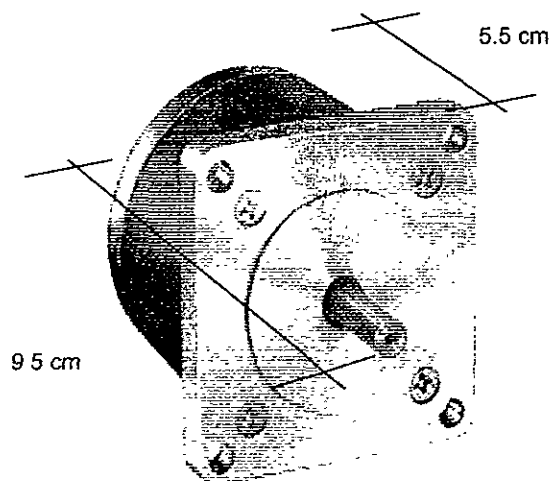


Figura 1-11. Motor a pasos que efectúa el movimiento de rotación.

4SH-24A56S AIRPAX



Código de colores

Fases	Color
Q1	Rojo
/Q1	Azul
Q2	Amarillo
/Q2	Café
24 V	Negro y
Blanco	

Figura 1-12. Motor a pasos que efectúa el movimiento de elevación.

CAPITULO II

SISTEMA SONIMPII

En este capítulo se describen las partes que integran el sistema SONIMPII y el ciclo de operación de la herramienta. También se da una explicación más detallada de las partes que integran el cartucho electrónico y su funcionamiento.

2.1 Antecedente

En el año de 1987, el Instituto Mexicano del Petróleo (IMP) firmó un convenio con Petróleos Mexicanos (PEMEX) en alianza con la compañía extranjera Schlumberger, para el desarrollo de una herramienta sonar que pudiera obtener la forma, las dimensiones y el volumen de las cavidades de almacenamiento que se estaban construyendo en el domo salino de Tuzandépetl, Veracruz.

Se estableció que la herramienta debería ser compatible con las unidades de registro que utilizaba PEMEX, que son unidades de registro CSU (Ciber System Unit), propios de la compañía Schlumberger.

El desarrollo de la herramienta SONIMPI se terminó con éxito en octubre del año de 1990, alcanzando los objetivos establecidos por PEMEX, al contar con una herramienta que les permitiera monitorear la forma de las cavidades de almacenamiento utilizadas como sistema de seguridad y almacenamiento de grandes cantidades de hidrocarburo.

Durante la explotación de las cavidades de almacenamiento, es posible que se presenten deformaciones en la forma de la cavidad al realizar procesos de llenado y vaciado del hidrocarburo, sobrepasando el diámetro especificado de las cavernas, por lo que en enero de 1993 se aprobó por parte de PEMEX, el desarrollo de una nueva herramienta sonar SONIMPII, que no sólo mejorara las características técnicas de la primera herramienta sino que también pudiera tomar registros a través de la tubería de revestimiento (TR).

Durante el diseño de la herramienta SONIMPII, se tuvo la necesidad de modificar el sistema de posicionamiento angular con giroscopios mecánicos, que elimine la interferencia magnética inducida por la presencia de TR, así como también el desarrollo de programas que permitiera interpretar el registro In Situ (el mismo día del registro).

2.2 Descripción general de la herramienta SONIMPII

El sistema SONIMPII es una herramienta de medición que se basa en el principio del sonar, nos permite obtener la forma, las dimensiones y el volumen de las cavidades de almacenamiento de hidrocarburos en explotación. Para determinar las dimensiones en cualquier punto de la caverna, se requiere conocer la

velocidad de propagación del sonido en el fluido y se determina el tiempo transcurrido desde la emisión del impulso ultrasónico hasta la detección de su respectivo eco.

El sistema SONIMP II se compone de dos partes, el equipo de fondo y el equipo de superficie. El equipo de superficie es la parte del sistema que controla la operación de la herramienta, envía órdenes hacia la herramienta para indicarle los movimientos que deberá realizar y recibe los datos generados por la herramienta para almacenarlos e interpretarlos a través de programas de cómputo desarrollados por Ingenieros del IMP. Como su nombre lo indica, el equipo de superficie se encuentra en la parte externa de la cavidad montado en una unidad móvil llamada BLUE (Basic Loggin Unit Electronic).

El equipo de fondo es la parte del sistema que se introduce a la cavidad por medio de un cable electromecánico de 7 conductores, utilizando el sistema de telemetría CTS (Cable Telemetry System). Está formado por sensores y transductores ultrasónicos que se encargan de la transmisión y recepción de la señal ultrasónica para efectuar los registros sonares, así como los circuitos electrónicos de control que realizan los movimientos de rotación y elevación de los transductores.

Los registros sonares se realizan tomando lecturas en forma horizontal y a diferentes profundidades, emitiendo la señal ultrasónica que es reflejada por las paredes de la cavidad para ser detectada en forma de ecos al regresar a la herramienta, contando el tiempo transcurrido desde la emisión hasta la recepción de la señal y conociendo la velocidad del sonido del medio se determina la distancia entre el punto de disparo de la señal y el punto de la cavidad que produjo el eco, de ésta forma se obtiene la forma completa de la cavidad, la cual se despliega a través de gráficas tridimensionales por medio del programa SONIMP_2. En la figura 2-1 se muestra la configuración general para la toma de un registro con el sistema sonar SONIMP II.

2.3 Equipo de superficie

El equipo de superficie está constituido por el sistema de control, adquisición e interpretación de datos y la unidad de registros BLUE, su función principal es recibir, mandar, procesar e interpretar la información que recibe del equipo de fondo y producir los resultados finales de la interpretación (In situ) de los datos, esta operación se lleva a cabo mediante el programa SONIMP_2, desarrollado en un lenguaje de programación de alto nivel y en un ambiente gráfico.

El sistema de control, adquisición e interpretación de datos, está compuesto de una microcomputadora AT tipo industrial compatible con IBM, un módulo de telemetría y sónico (MCTS-A), un osciloscopio digital de dos canales, una fuente de poder ininterrumpible (UPS), un inversor de CD a CA, una unidad de potencia 2kW y una impresora portátil.

El equipo de adquisición e interpretación de datos es alimentado con 120 VCA @ 60 Hz, con un consumo máximo de 2 kW.

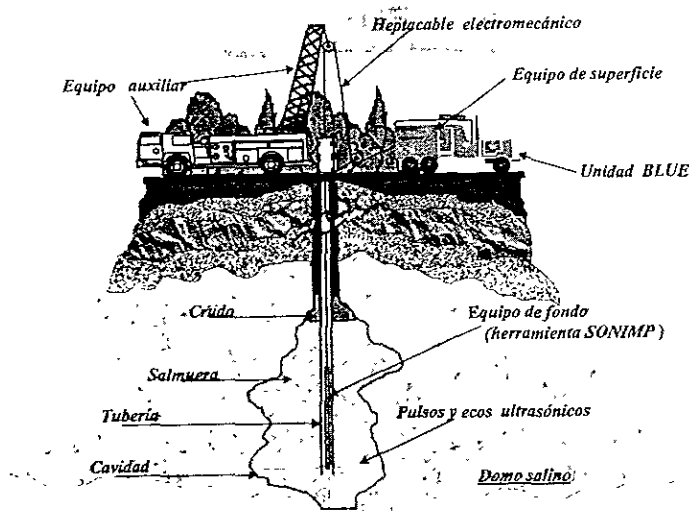


Figura 2-1. Configuración general para la toma de un registro sonar.

El sistema de control, adquisición e interpretación de datos se encuentran instalados en la unidad móvil BLUE y se muestran en la figura 2-2.

La microcomputadora de tipo industrial contiene el programa SONIMP_2 que controla los modos de operación de la herramienta, recibe la información proveniente del módulo de telemetría MCTS-A, para procesarla, interpretarla y desplegarla en el monitor o en la impresora portátil.

El programa de cómputo SONIMP_2, nos permite enviar una serie de comandos para controlar los diferentes modos de operación de la herramienta, como el control de los movimientos de rotación y elevación de los transductores, la activación de la transmisión de la señal ultrasónica, la frecuencia de operación que se aplica al transductor ultrasónico, ganancia de fondo para amplificar el eco reflejado por la pared de la cavidad, entre otros.

El proceso de interpretación del programa nos permite la obtención del volumen de la cavidad, así como una serie de gráficas tridimensionales que nos muestran las cavidades a diferentes cortes tanto horizontales como verticales, y a diferentes profundidades y orientaciones.

CAPITULO II

HERRAMIENTA SONIMP II

El módulo de comunicación de Telemetría y sónico del SONIMP II se utiliza como interface entre el cable del registro y la microcomputadora, recibe señales de la herramienta de fondo y se encarga de acondicionarlas para enviarlas al CPU de la computadora, también recibe la información introducida por el usuario mediante el uso del programa SONIMP_2.

El osciloscopio de dos canales se utiliza para observar la forma de la señal ultrasónica de recepción.

La fuente de poder ininterrumpible se utiliza para suministrar energía al sistema en caso de alguna interrupción o falla de la energía eléctrica, y poder respaldar la información de los registros sonares en un tiempo máximo de 10 minutos.

El inversor de CD a CA, se utiliza para alimentar la herramienta SONIMP II a un voltaje de alterna de 120 VCA @ 60 Hz, el inversor es alimentado por un banco de baterías de 12 volts que se encuentra en la unidad móvil BLUE, en caso de no contar con el inversor se realiza una conexión externa de alimentación, para proporcionarle energía a la herramienta.

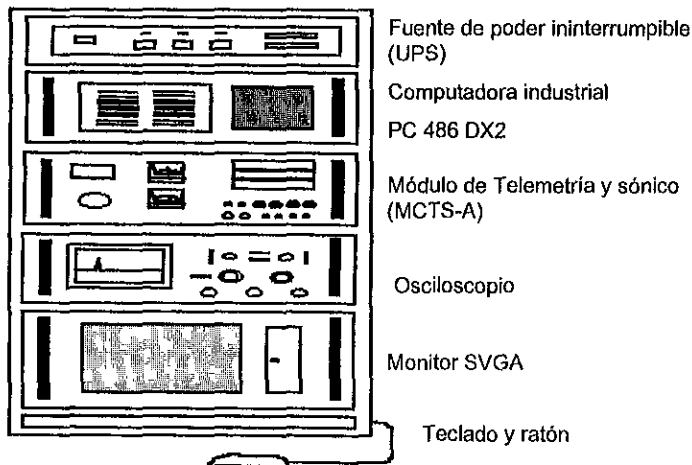


Figura 2-2. Componentes que integran el sistema de control, adquisición e interpretación de datos.

2.4 Equipo de fondo

El equipo de fondo (ver figura 2-3) es la parte del sistema SONIMP II que se introduce en el interior de la cavidad, por medio de un cable formado por siete conductores conocido como heptacable, por éste se transmiten las señales que

permiten la comunicación a distancia entre el equipo de fondo y el equipo de superficie utilizando el sistema de telemetría por cable CTS.

Como anteriormente se mencionó, el equipo de fondo están contenidos los sensores y transductores ultrasónicos, así como la instrumentación electrónica necesaria para el acondicionamiento, procesamiento y manejo de las señales ultrasónicas para efectuar la medición de la cavidad.

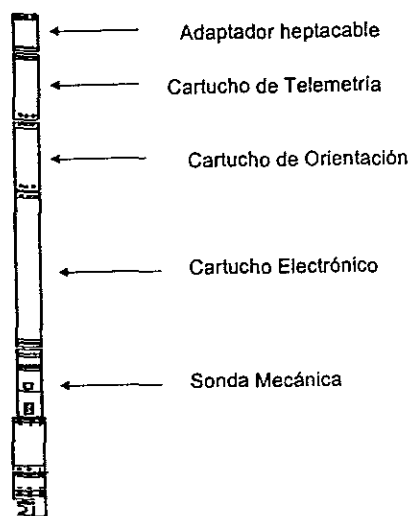


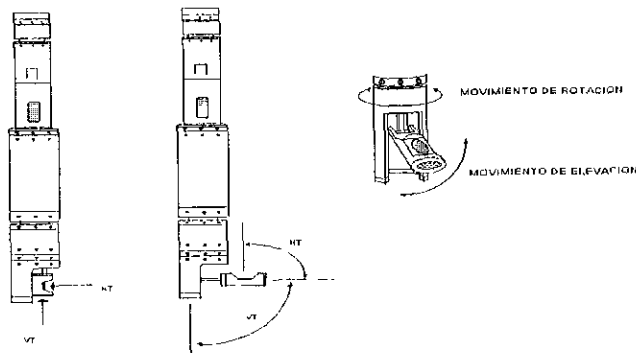
Figura 2-3. Componentes que integran el equipo de fondo del sistema SONIMP II.

El equipo de fondo está formado por cuatro secciones principales:

- a) Cartucho de Telemetría. Es un cartucho comercial modelo TCC-A (Telemetry Communications Cartridge) marca Schlumberger, modificado para trabajar con la herramienta SONIMP II, la modificación consistió en reducir la frecuencia de sincronización de 15 a 7.5 Hz. En el cartucho de telemetría se generan las señales del ciclo de operación de la herramienta, que se explican con mayor detalle en la Secuencia general de tiempos.
- b) Cartucho de Orientación. Está formado por un sistema de orientación no magnético que utiliza un giroscopio mecánico de la compañía Humprey Inc, para conocer la posición de la herramienta con respecto al norte magnético de la tierra dentro de la cavidad, en presencia de TR, a diferencia de su antecesora SONIMPI, que utiliza una brújula de estado sólido, para dar la referencia a la herramienta del norte magnético. La modificación del sistema de

orientación del cartucho se debió a la interferencia magnética inducida por la TR que provocaba que la brújula se desorientara.

- c) Cartucho electrónico. Está formado por un conjunto de circuitos electrónicos analógicos y digitales necesarios para desarrollar su función, la función básica de este cartucho es generar, transmitir y recibir las señales ultrasónicas, así como el control de los motores a pasos para mover el transductor ultrasónico.
- d) Sonda mecánica. La sonda mecánica está formada por sensores y transductores ultrasónicos, y por los motores a pasos que permiten mover y orientar a dichos transductores. La sonda opera en equilibrio de presión con el medio exterior, mediante el empleo de un pistón compensador de presión. El elemento principal es el transductor piezoeléctrico que se localiza en la parte inferior de la sonda, este transductor, es un transductor doble, porque contiene dos elementos piezoeléctricos, colocados 90° uno con respecto al otro, y de acuerdo a la dirección en la que emiten el haz ultrasónico, se les denomina como transductor horizontal (HT) y transductor vertical (VT). El transductor ultrasónico principal, tiene la capacidad de dirigir el haz ultrasónico en cualquier dirección de 0 a 359 grados, realizando un movimiento de rotación alrededor del eje vertical, y efectuar un movimiento de elevación de 0 a 90 grados, en la figura 2-4 se muestran los movimientos de elevación y rotación del transductor principal. La sonda tiene también un transductor que se emplea para la medición de la velocidad de propagación del sonido en el medio que llena la cavidad, en este caso salmuera, y forma parte de un velocímetro acústico.



2-4. Movimientos de rotación y elevación de la sonda mecánica.

2.5 Teoría de operación de la herramienta Sonimp II

La forma en la que interactúan las secciones del equipo de fondo que componen la herramienta SONIMP II, se puede presentar por medio de una serie de bloques funcionales para su fácil comprensión, que se muestran en la figura 2-5.

En esta figura, el cartucho de telemetría es la interfaz de comunicación donde pasan los datos que se envía a la superficie y las órdenes enviadas a la herramienta, dichos datos son enviados hacia el cartucho electrónico para procesarlas, acondicionarlas y distribuirías a las diferentes tarjetas contenidas en él.

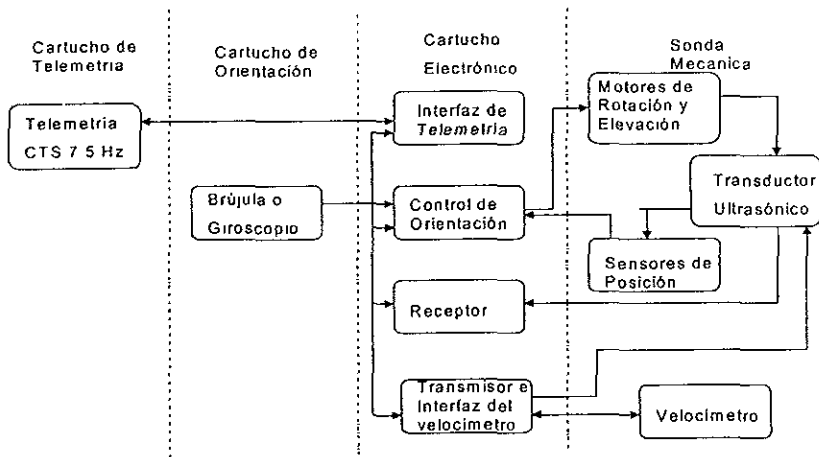


Figura 2-5. Diagrama a bloques de la herramienta de fondo.

La interfaz de telemetría acondiciona y genera las señales enviadas del cartucho de telemetría, para que los circuitos del cartucho electrónico puedan ejecutar las órdenes enviadas desde la superficie, es por decirlo así, el intérprete entre el cartucho de telemetría y el cartucho electrónico.

El cartucho de orientación envía al cartucho electrónico el valor de la posición angular de la herramienta con respecto al norte magnético de la tierra, el sistema de orientación puede utilizar el cartucho que contiene la brújula o el giroscopio según sea el caso, esto es porque la herramienta SONIMP II puede operar con o sin TR.

El control de orientación se encuentra en el cartucho electrónico y genera los pulsos de excitación que se envían a los motores a pasos para que el transductor ultrasónico principal realice los movimientos de rotación y elevación. También se

generan los datos que indican la posición real de los transductores, que se envía hacia arriba a través de la interfaz de telemetría.

El circuito transmisor envía a la sonda mecánica la señal para excitar al transductor ultrasónico, en forma de trenes de pulsos que van desde 1 hasta 128 pulsos dependiendo si el registro se hace con o sin TR, y a una frecuencia desde 150 hasta 250 kHz dependiendo del transductor instalado en la sonda. La señal de excitación se dispara cada 133.3 ms, con una amplitud de 150 volts.

El circuito receptor, recibe del transductor ultrasónico la señal reflejada por la pared de la cavidad, para acondicionarla, de tal forma que pueda ser transmitida por el cable del registro.

El circuito del velocímetro determina el dato necesario para el cálculo de la velocidad de propagación del sonido en el fluido de las cavernas.

El transductor ultrasónico es de tipo piezoeléctrico y se encarga de producir y detectar las ondas ultrasónicas que nos permiten hacer las mediciones de las cavidades, su frecuencia de resonancia varía dependiendo del transductor que se instale en la sonda.

Se tienen dos sensores de posición, el sensor de efecto Hall, y un microinterruptor, que se utilizan para tener una señal de referencia cero en los transductores. El sensor de efecto Hall genera la señal de índice cero para indicarle al transductor horizontal la posición de cero grados, y el microinterruptor genera una señal que le indica al transductor vertical cero grados de elevación. Ambos sensores se encuentran en la sonda mecánica.

2.5.1 Secuencia general de tiempos

La herramienta SONIMP II funciona mediante un ciclo de operación que está sincronizado con la frecuencia de la línea de 60 Hz, y tiene una duración de 133.3 ms. En la figura 2-6, se muestra un diagrama de tiempos del ciclo de operación de la herramienta SONIMP II.

El ciclo de operación se analiza en tres partes:

- a) Transmisión de datos. Es el tiempo que tarda la herramienta en enviar los datos hacia la superficie, con una duración de 1ms.
- b) Ventana sónica. Es el tiempo en el que se efectúa la transmisión y recepción de la señal ultrasónica y tiene una duración de 70 ms.
- c) Recepción de datos. Es el tiempo en el que la herramienta recibe las órdenes enviadas desde la superficie con una duración aproximada de 320 μ s.

1.5.2 Transmisión de datos

El ciclo de operación inicia con la señal de pulso GO que inicialmente se encuentra en estado bajo, se genera en el cartucho de telemetría con una duración de 20 μ s, es enviado al cartucho electrónico para indicarle que se prepare para enviar al equipo de superficie los datos obtenidos en el ciclo anterior.

Después de la bajada del pulso GO, el cartucho de telemetría genera la señal de reloj UCLK (up clock), para sacar los datos de los registros de corrimiento entrada paralelo/salida serie del cartucho electrónico, los datos a enviar son elevación, azimuth, velocidad del fluido y palabra de estado.

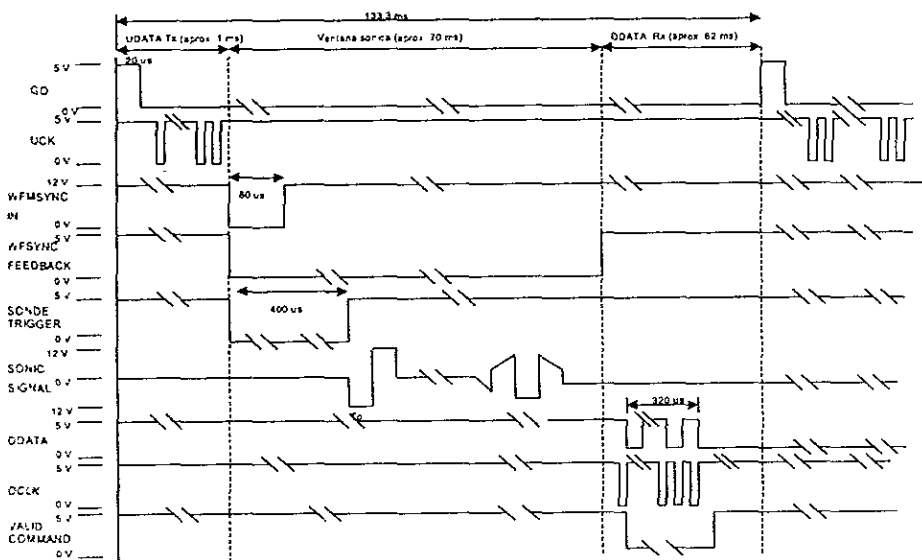


Figura 2-6. Ciclo de operación de la herramienta SonimpII.

Las señales GO y UCLK son generadas en el cartucho de telemetría, antes de llegar a los circuitos del cartucho electrónico pasan a través de la interfaz de telemetría donde son acondicionadas.

2.5.3 Ventana sónica

Después del envío de los datos hacia la superficie, se genera en el cartucho de telemetría la señal WFMSYNC (wave form sync), 1 ms después de la bajada del pulso GO. Dicha señal pasa a través de la interfaz de telemetría para generar dos señales, WFMSYNC FEEDBACK y SONDE TRIGGER, la primera señal determina

la duración de la ventana sónica y la segunda señal se utiliza principalmente para efectuar el disparo del transductor ultrasónico.

Después de 400 μ s a partir del inicio de la ventana sónica, se genera la señal SONIC SIGNAL. La señal SONIC SIGNAL sirve como referencia para llevar el conteo del tiempo (To) que tarda la señal sónica a partir de su envío hasta la recepción del eco reflejado por la pared de la caverna. Poco antes de finalizar la ventana sónica, se efectúa la medición de la velocidad de propagación del sonido en el fluido por medio del velocímetro acústico.

2.5.4 Recepción de datos

Al término de la ventana sónica, el cartucho de telemetría genera una señal DOWN SIGNAL o DSIG, dicha señal contiene las órdenes enviadas a la herramienta desde la superficie y es enviada a la interfaz de telemetría. A partir de la señal DSIG, la interfaz de telemetría genera dos señales DOWN DATA (DDATA) Y DOWN CLOCK (DCLK) que son utilizadas para enviar los datos a los registros de corrimiento entrada serie /salida paralelo hacia los diferentes circuitos de control del cartucho electrónico.

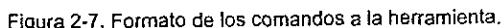
La señal VALID COMMAND, es generada en la interfaz de telemetría y se envía al mismo tiempo que se envían las señales DDATA y DCLK. Cada vez que se recibe una señal válida de la superficie la señal VALID COMMAND se habilita con un nivel alto de voltaje.

2.6 Órdenes hacia la herramienta

El equipo de superficie envía ordenes hacia el equipo de fondo para indicarle a la herramienta las funciones que debe realizar, como la posición deseada del transductor ultrasónico principal, la selección de dicho transductor, ganancia del velocímetro, la potencia aplicada al transductor, entre otras. Las órdenes hacia abajo DDATA (DOWN DATA) se envían en dos tramas de 32 bits en diferentes ciclos de operación diferentes.

Las señales DDATA y DCLK (DOWN CLK), provienen de la interfaz de telemetría y se envían en serie, sincronizados con la señal de reloj DCLK hacia los registros de corrimiento que convierten la información serial en paralelo para que sea utilizada por el resto de los circuitos digitales del cartucho electrónico.

Los registros de corrimiento están conectados en cascada y las señales DCLK y VALID COMMAND están conectadas en paralelo a cada uno de ellos. En la figura 2-7, se muestra el flujo de datos hacia abajo a través de los registros de corrimiento.



Los datos UDATA (UP DATA) se envían hacia la superficie en una trama de 32 bits, por cada ciclo de operación de la herramienta, para que sean procesados e interpretados por el equipo de superficie, los datos contienen la información de la posición del transductor, velocidad del fluido, así como las condiciones de los voltajes que alimentan a cada uno de los cartuchos de la herramienta.

Los datos se almacenan en los registros de corrimiento al pulso GO, son desplazados en serie y sincronizados con la señal de reloj UCLK hacia la interfaz de telemetría. Los registros de corrimiento están conectados en cascada y la señal de reloj se encuentra conectada en paralelo a cada uno de ellos. En la figura 2-8, se muestra el flujo de datos hacia arriba a través de los registros de corrimiento.

CAPITULO II

HERRAMIENTA SONIMPII

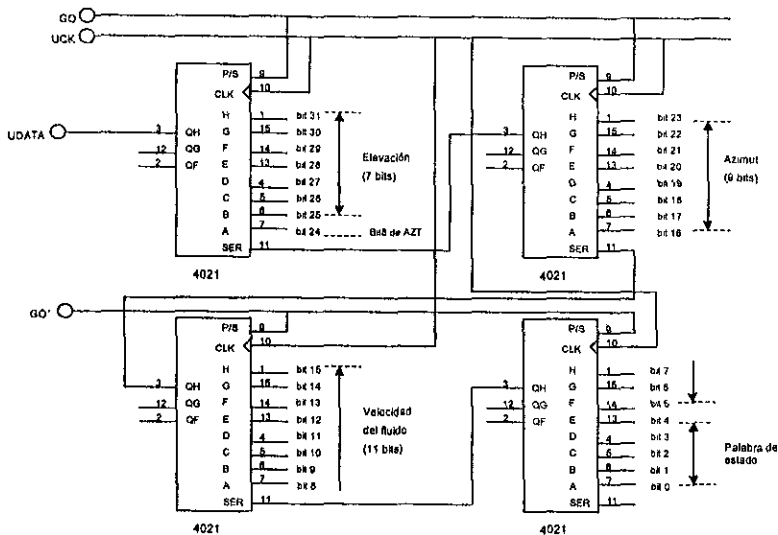


Figura 2-8. Formato de los datos enviados hacia la superficie.

2.8 Procedimiento general para efectuar un registro sonar

Para efectuar la toma de registros sonares, es necesario realizar una inspección de ciertos valores que deben ajustarse inicialmente, entre los valores de ajuste inicial están: el valor de la declinación magnética local, la ganancia de fondo dependiendo de las condiciones de la cavidad, la frecuencia del sintetizador, el tiempo para rechazar ecos no deseados, el número de pulsos de la señal de transmisión, ganancia del velocímetro, entre otros.

Cuando se desarma la herramienta ya sea para darle mantenimiento o por motivo de inspección, se efectúa la calibración de toda la herramienta, para efectuar los registros sonares con mayor seguridad y exactitud.

Los registros de la cavidad se realizan en cortes horizontales a varias profundidades llamados *scan*, con una separación entre cada corte de 2 o 5 metros, para obtener un corte horizontal se utiliza el transductor horizontal para realizar los movimientos de rotación y desde la pantalla de control se le envía la orden AUTO o MANUAL, para realizar las mediciones del radio de la cavidad a diferentes direcciones, ya sea de forma automática que controla el barrido horizontal hasta cubrir los 360 grados, o bien de forma manual que permite al usuario validar la medición, las mediciones del radio de la cavidad se hacen

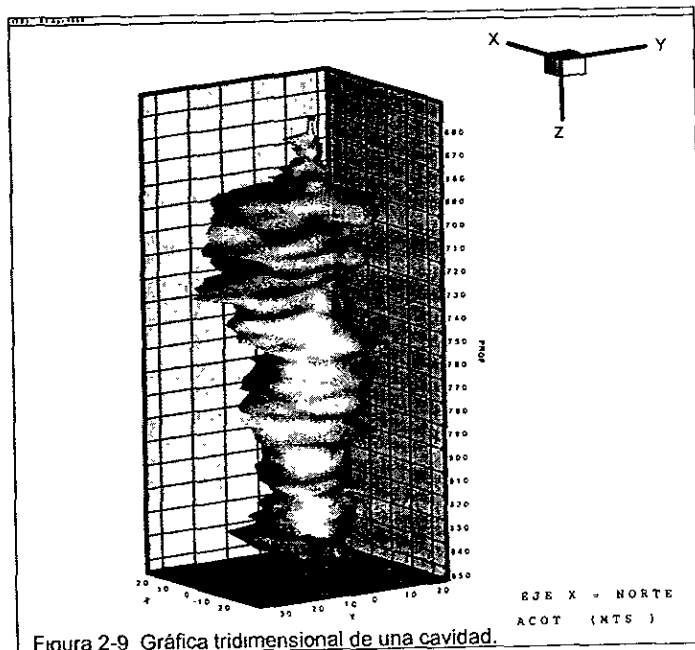
generalmente cada 10 o 15 grados. Durante el barrido de los cortes horizontales el transductor vertical se encuentra en cero grados de elevación.

Cuando se tiene la presencia de la tubería de producción, es necesario detectar primero los coples (el cople es la unión de acoplamiento entre cada sección de tubo que forma la TR) de la tubería operando con el transductor vertical para conocer las diferentes profundidades donde se localizan dichos coples, para no realizar mediciones a través de ellos, ya que en estas uniones el espesor de la tubería es el doble, provocando una atenuación en la señal ultrasónica.

Una vez detectado los coples hasta el final de la tubería TR, se baja la herramienta 1/2 metro de ésta para registrar el fondo de la cavidad, por medio de cortes inclinados a diferentes elevaciones del transductor. Una vez registrado el fondo se realizan los barridos horizontales de abajo hacia arriba para la obtención de la forma total de la cavidad.

Para registrar el techo es necesario quitar la tubería TR, y colocar la herramienta al inicio de la bóveda del techo, para efectuar varios cortes inclinados, utilizando el transductor vertical.

La información de los registros es interpretada y desplegada en tiempo real, por medio de gráficas tridimensionales, en la figura 2-9, podemos ver una gráfica tridimensional de una cavidad.



2.9 Descripción del Cartucho electrónico

El cartucho electrónico es la parte de la herramienta que se encarga de generar, transmitir y recibir las señales ultrasónicas, así como el control de los motores a pasos para mover el transductor ultrasónico principal. En la figura 2-10, podemos ver las partes que integran al cartucho electrónico.

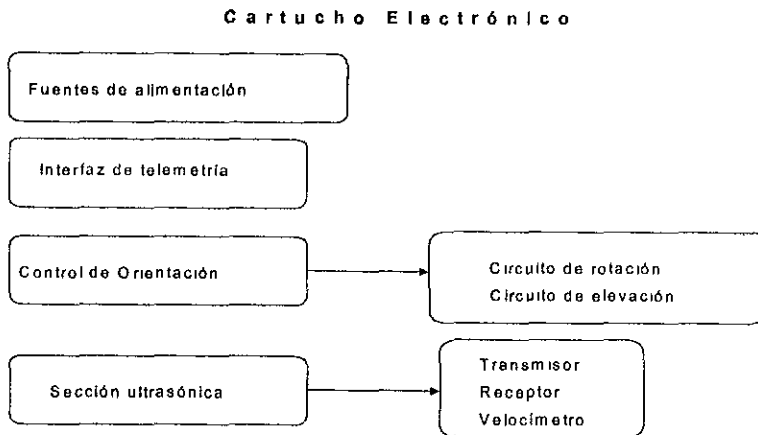


Figura 2-10. Partes que integran el cartucho electrónico.

En el cartucho electrónico están contenidas las fuentes de alimentación, que proporcionan voltaje de corriente directa a los circuitos electrónicos analógicos y digitales, a los transductores que se encuentran en la sonda mecánica y a los motores a pasos. Se tiene una fuente regulada de +5 volts, que alimenta a la mayoría de los circuitos digitales que se encuentran en los circuitos de control de orientación y de telemetría.

También se cuenta con dos fuentes reguladas de voltaje de +12 y -12 volts, que alimentan principalmente a los circuitos analógicos que se utilizan en el circuito receptor, en los amplificadores del velocímetro, y en los circuitos de excitación de los motores a pasos. Una fuente de 24 volts no regulada que se utiliza para alimentar al motor de elevación. Una fuente de 38 volts no regulada que se utiliza para alimentar al motor de rotación. Una fuente de 50 volts no regulada que se usa para excitar el transductor del velocímetro acústico. Por último una fuente de 150 volts no regulada que se usa para excitar el transductor ultrasónico principal.

El cartucho electrónico también está constituido por una sección ultrasónica. La sección ultrasónica está compuesta por un circuito transmisor, un circuito receptor

y por el circuito del velocímetro, las cuáles fueron previamente descritas en la sección 2.5.

Como anteriormente se explicó, el circuito transmisor genera la señal de excitación en forma de trenes de pulsos, la frecuencia de la señal depende del transductor piezoeléctrico que se esté utilizando, puede estar entre 100 y 625 kHz, el número de pulsos de la señal varía desde 1 pulso hasta 128 dependiendo del alcance que se desea tener del haz y de la presencia de TR, es decir, normalmente en presencia de TR se utilizan de 4 a 64 pulsos, y sin TR, de 1 a 32 pulsos.

El circuito receptor recibe la señal ultrasónica reflejada recibida y la acondiciona, de tal forma que pueda ser transmitida por el cable de registro, es decir, se amplifica la señal ultrasónica reflejada, para después detectar la envolvente y generar una señal por medio de una modulación de una portadora de 25 kHz. También tiene la función de generar el pulso de referencia T_0 cada vez que se dispara el transductor principal.

El circuito del velocímetro no sólo determina el dato necesario para el cálculo de la velocidad de propagación del sonido en el fluido de las cavernas, si no también genera la señal de excitación correspondiente al transductor del velocímetro.

La interfase de telemetría es la parte del sistema SOMIPPII que permite la comunicación entre el cartucho de telemetría y el cartucho electrónico, su función es acondicionar y decodificar las señales que le llegan del cartucho de telemetría para enviarlas hacia los circuitos digitales del cartucho electrónico. La señal de telemetría genera las señales DOWN CLOCK, DOWN DATA, VALID COMMAND, WAVEFORM HANDSHAKE O FEEDBACK, explicadas en la sección de secuencia general de tiempos, de este capítulo.

El sistema de orientación tiene como función principal orientar al transductor ultrasónico a la posición deseada e indicada desde la superficie, y controla los movimientos de rotación y elevación del transductor ultrasónico principal.

2.10 Descripción y funcionamiento del sistema original de orientación del cartucho electrónico

El sistema de original orientación está basado en dos microcontroladores MC1468705G2, (Micro Computer Unit, MCU) que realizan el control total de las funciones de rotación y elevación del transductor ultrasónico principal, así como el circuito de excitación de los motores a pasos. El transductor horizontal realiza los movimientos de rotación, y el transductor vertical realiza los movimientos de elevación. Ambas funciones de control tienen su propio programa para realizar en forma independiente los movimientos de rotación y elevación.

En la figura 2-11, se tiene un diagrama a bloques con los componentes que integran el sistema de orientación del cartucho electrónico, así como las señales de entrada y salida del sistema.

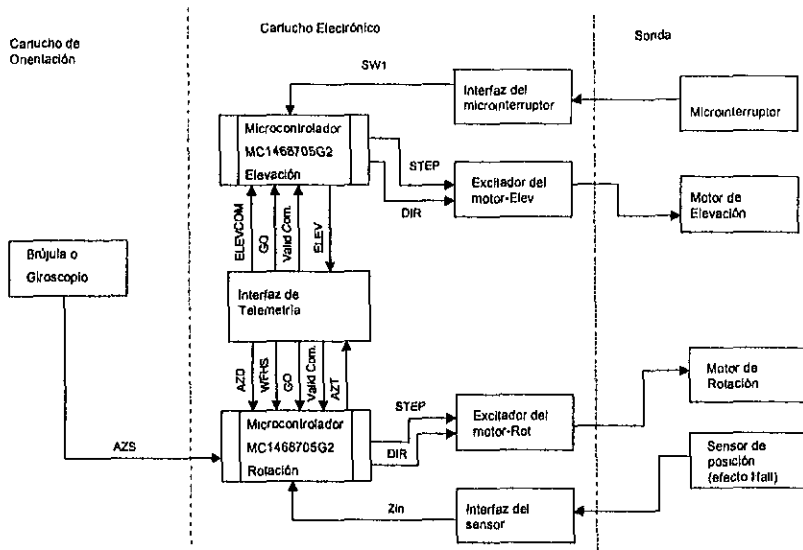


Figura 2-11. Diagrama a bloques de las funciones de control de elevación y rotación.

El MCU MC1468705G2, es un circuito integrado fabricado por la compañía Motorola con tecnología CMOS, contiene en el mismo circuito la unidad central de proceso (CPU), memoria RAM y ROM, líneas bidireccionales de entrada y salida. Este dispositivo es programable y se requiere de un programa diseñado para una aplicación específica, en este caso el control del movimiento de los motores. El MCU cuenta con 2KB de UV EPROM, 112 bytes en RAM, 32 líneas bidireccionales de entrada/salida, un contador interno de 8 bits con un divisor de frecuencia programable de 7 bits, una interrupción externa y una entrada externa del contador (TIMER).

2.10.1 Control de Rotación

El control de la rotación se realiza utilizando el MCU MC1468705G2, el excitador del motor, la interfaz del sensor de posición, y el dato del azimuth de la sonda proveniente del cartucho de orientación, su función es mantener el transductor ultrasónico orientado al azimuth correcto. Azimuth, es un término utilizado en la navegación, para tener una dirección de referencia con respecto a un objeto colocado en el horizonte o en el espacio. En nuestro caso, el término azimuth nos indica la posición del transductor horizontal con respecto al espacio en el que se encuentra inmerso.

El sensor de referencia, es un sensor de efecto Hall que nos sirve para tener una referencia de cero, cada vez que el transductor realiza una revolución se envía una señal de pulso Z_{in} .

El control de rotación tiene como información de entrada: el valor del azimuth deseado (AZD), proveniente de la interfaz de telemetría; el valor del azimuth de la sonda (AZS), proveniente del cartucho de orientación; el valor del pulso Z_{in} de referencia, proveniente de la interfaz del sensor de posición; y la señal WFHS, proveniente de la interfaz de telemetría.

Con el valor del azimuth de la sonda y con la señal de posición Z_{in} , se calcula el valor del azimuth del transductor, sumando ambos valores y se compara con el valor del azimuth deseado, dependiendo del valor de la comparación se envían las señales de pulso de avance (STEP) y sentido de giro (DIR) al circuito de excitación del motor de rotación hasta llegar a la posición deseada. Una vez colocado el transductor en la posición deseada, se envía el valor del transductor (AZT) hacia la superficie a través de los registros de corrimiento. La señal WFHS, es sensada por el MCU, el cual deja de enviar las señales STEP y DIR que excitan al motor para evitar que se induzca ruido en la recepción de la señal ultrasónica.

La señal de entrada AZS, proveniente del cartucho de orientación, llega al MCU en forma de trenes de pulsos a una frecuencia de 20 kHz cada 40 ms. El formato de la información de los trenes de pulsos se le conoce como (N+1), donde N corresponde al valor del azimuth de la sonda.

El circuito de excitación del motor de rotación, está formado por un circuito integrado (CI) SAA1027 comercial, el cual genera las secuencias de fases del motor a partir de dos señales de entrada, una señal en forma de trenes de pulsos STEP que le indica el avance al motor y la otra señal DIR que se usa para darle el sentido de giro al motor. Como salidas se tienen las señales de control para las diferentes fases del motor. Las fases de salida del circuito excitador pasan a través de una etapa de salida que está formada por cuatro transistores MOSFET de potencia contenidos en un módulo de potencia CPY400 fabricado por IR (International Rectifier).

1.10.2 Control de Elevación

El control de la elevación se realiza utilizando el MCU, el excitador del motor, y la interfaz del microinterruptor. La función del control de elevación es mantener al transductor ultrasónico vertical en la inclinación deseada.

El control de elevación recibe como señales de entrada: el dato de elevación deseada (ELEV_D), proveniente de la superficie; y la señal de posición cero (SW1), proveniente de la interfaz del microinterruptor. El microinterruptor se cierra cuando el transductor se encuentra en una posición de cero grados.

A partir de esta información, se obtiene el valor de elevación real del transductor (ELEV) y se compara con el valor de elevación deseada, dependiendo del valor de la comparación, el MCU envía la señal de pulso de avance STEP y la señal DIR de sentido de giro a la entrada del circuito excitador del motor a pasos, hasta lograr la posición deseada del transductor. El valor de elevación del transductor es colocado en los registros de corrimiento para ser enviado hacia la superficie.

El circuito de excitación del control de elevación es idéntico al del control de rotación SAA1027, las fases de salida del circuito de excitación se envían a una etapa de potencia formada por un arreglo de transistores Darlington que se encuentran en un circuito integrado UNL2001.

En este capítulo se describió cada una de las partes que integran el sistema SONIMP II, así como también la funcionalidad entre cada una de ellas, detallando un poco más en la descripción del cartucho electrónico por contener el sistema de orientación original de dicho cartucho.

CAPITULO III**DISEÑO Y DESARROLLO DEL SISTEMA DE ORIENTACIÓN DEL CARTUCHO ELECTRÓNICO**

Una vez que se ha descrito el sistema SONIMP II en general en el capítulo anterior, describiremos y explicaremos la forma en la que se llevó a cabo el diseño y la implementación del sistema de orientación del cartucho electrónico.

Primeramente se analizará el sistema total de orientación original, para determinar el número de entradas y salidas (E/S) que se emplearán en el nuevo sistema, la simplificación y/o reducción del número de componentes electrónicos de dicho sistema, y la selección del microcontrolador que realizará las funciones de rotación y elevación de los transductores.

Posteriormente, se mostrará el diseño electrónico del sistema de orientación modificado, el desarrollo de los algoritmos y diagramas de flujo diseñados para lograr el control de los movimientos de rotación y elevación, la construcción y el ensamble de las tarjetas electrónicas del sistema de orientación modificado del cartucho electrónico.

3.1. Descripción y análisis del sistema de orientación original del cartucho electrónico

El sistema de orientación del cartucho electrónico podemos analizarlo según su funcionalidad en tres secciones, Control de rotación, Control de elevación y Excitación de motores. Cada sección está formada por un conjunto de circuitos integrados con tecnología CMOS (complementario metal-óxido semiconductor), por las ventajas que éste ofrece, al tener un bajo consumo de energía y alta inmunidad al ruido.

La sección Control de Rotación, se encarga del control total para que el transductor ultrasónico horizontal ejecute los movimientos de rotación ordenados desde la superficie. El control del movimiento de rotación se realiza por medio de un MCU (MC1468705G2 Motorola) que contiene un programa diseñado específicamente para ello, también esta formado por circuitos de lógica secuencial, compuertas AND's utilizados para el sistema de interrupción, registros de corrimiento entrada paralelo/salida serie (P/S) y viceversa entrada serie /salida paralelo (S/P), para el flujo de datos de entrada y salida del valor del azimuth.

La sección Control de Rotación está contenida en una tarjeta electrónica cuyas dimensiones son de 8 pulgadas de largo x 2 de ancho, en ella están montados 8 dispositivos electrónicos y dos conectores que se utilizan para las líneas de E/S.

En la figura 3-1, se muestra el diagrama electrónico del sistema de Control de Rotación.

La sección Control de Elevación, controla los movimientos de elevación ejecutados por el transductor vertical. Está constituido por un MCU (MC1468705G2 Motorola) que contiene un programa diseñado para controlar las diferentes inclinaciones del transductor, cuenta con circuitos de lógica secuencial y compuertas NOR's para el sistema de interrupción, y los registros de corrimiento S/P y P/S para el flujo de los datos de elevación.

La sección de Excitación de motores, contiene los circuitos que realizan las secuencias de fases para excitar a los motores en el sentido de giro apropiado, así como también las etapas de potencia para el suministro de corriente. El circuito SAA1027, genera las secuencias de fase de los motores a pasos comerciales de dos fases. El motor de elevación utiliza una etapa de potencia formada por un arreglo de transistores Darlington contenidos en el circuito integrado ULN2001. La etapa de potencia del motor de rotación está formado por cuatro transistores MOSFET contenidos en un módulo de potencia CPY400. Cabe mencionar que la etapa de potencia de la sección Control de Rotación no se encuentra en el diagrama electrónico de la figura 3-2, porque se encuentra montada en el chasis de la herramienta.

La sección Control de Elevación y Excitación de motores se encuentran montados en la misma tarjeta electrónica cuyas dimensiones son de 8 pulgadas de largo y 2 de ancho, la cual está formada por nueve dispositivos electrónicos y dos conectores de E/S. En la misma figura 3-2, se muestra el diagrama electrónico de las secciones Control de Elevación y Excitación de motores.

3.1.1. Señales de entrada y salida de la sección Control de Rotación

En la figura 3-1 se muestra el diagrama electrónico de la sección Control de Rotación que actualmente está funcionando en el cartucho electrónico, con ayuda de éste diagrama describiremos e identificaremos las señales de entrada y salida de la sección Control de Rotación.

Las salidas de los datos en los registros de corrimiento están conectados al puerto A del MCU, utilizando el mismo bus de datos bidireccional, para la entrada del azimuth deseado (AZD) definida como señal de entrada y la salida del azimuth del transductor (AZT) definida como salida.

El MCU genera la señal OE, que es enviada al registro de corrimiento S/P, para que el dato pueda ser leído por el puerto A, también genera la señal GO', que es enviada al registro de corrimiento P/S para cargar el dato AZT en el registro. El dato del azimuth es de 9 bits, por lo que el bit más significativo se lee desde los registros de corrimiento de la sección Control de Elevación.

Doc Number	Doc Size	Doc Title	Doc Rev
9	8	109 CONTROL DE ROTACION	REV
9	8	PROYECTO CBL-5234 SONIMP LT	REV

CONFIDENTIAL - SECURITY INFORMATION

En la figura 3-3 podemos ver claramente el bus de datos bidireccional y las líneas de control enviadas por el MCU.

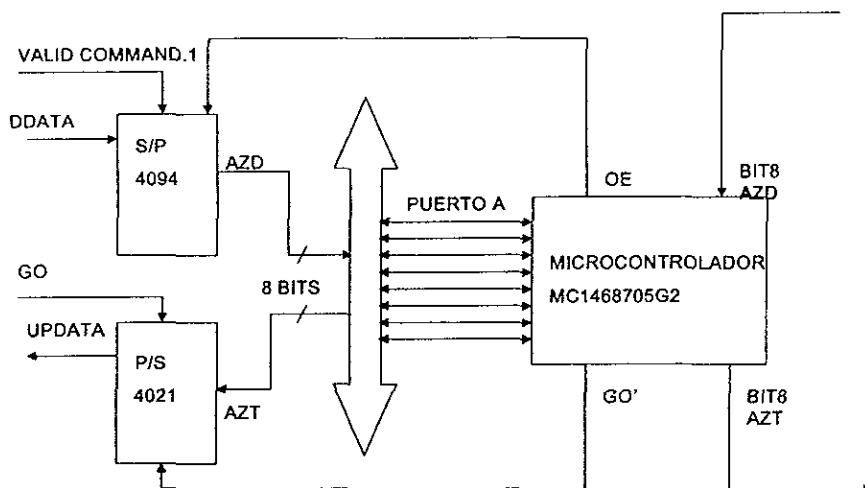


Figura 3-3. Bus de datos bidireccional del sistema de control de rotación.

Un bus de datos, podemos definirlo como un conjunto de cables o líneas de conexión compartidas, por las cuales se transfieren los datos, se utilizan para reducir el número de líneas o cables que se unen eléctricamente entre dispositivos eléctricos y/o distintas tarjetas.

La señal BIT8AZD es una señal de entrada al sistema, y proviene de la terminal 11 del registro de corrimiento (S/P) del sistema de control de elevación, para ser leído por el bit1 del puerto C del MCU. Es el bit más significativo del dato AZD.

La señal BIT 8 AZT es una señal de salida del sistema, y es generada por el MCU (bit0 del puerto D), para ser leído por el registro de corrimiento P/S (terminal 7) del sistema de control de elevación. Es el bit más significativo de AZT.

El dato DDATA IN, es una señal de entrada al sistema que contiene las órdenes enviadas desde la superficie, proviene de la terminal 10 del registro de corrimiento (S/P) del circuito de elevación y entra a la terminal 2 del registro de corrimiento (S/P) del circuito de rotación.

El dato UDATA IN, es una señal de entrada al sistema de control de rotación y

llega a la terminal 11 del registro de corrimiento (P/S), dicho dato contiene el valor de AZT.

El dato DDATA OUT es una señal de salida del sistema y sale de la terminal 10 del registro de corrimiento (P/S).

El dato UDATA OUT, es una señal de salida del sistema y sale de la terminal 3 del registro de corrimiento (P/S).

Las señales /DCLK y /UCLK son señales de entrada al sistema de control de rotación, ambas son señales de reloj que se utilizan para enviar datos hacia y desde la herramienta, a una frecuencia de 100 kHz, la primera señal llega a la terminal 3 del registro de corrimiento (S/P), y la segunda llega a la terminal 10 del registro de corrimiento (P/S).

La entrada Zin índice cero, es una señal de entrada al MCU y llega a la terminal 19 de éste. La señal Zin es una señal de pulso, que se activa cada vez que el transductor horizontal realiza una vuelta completa.

La señal WFHS, es una señal de entrada al sistema de control de rotación y llega a la terminal 16 del MCU.

Se tiene un sistema de interrupción formado por tres flip-flop's D, y dos compuertas AND's (ver figura 3-4); dependiendo de la señal de interrupción que se active, VALID COMMAND.1, GO o la señal AZS, señales de entrada al sistema de rotación, se realizará un cambio de estado de 1 a 0 lógico en la salida /Q de cada flip-flop correspondiente a la señal de interrupción, para activar la interrupción externa del MCU, esto es, porque el MCU se interrumpe a cada flanco de bajada presente en la terminal de interrupción externa.

A su vez, las salidas Q de los flip-flop's D cambian de estado de 0 a 1 lógico, para que la rutina de interrupción reconozca la señal de interrupción solicitada, y pueda ser identificada mediante una señal de salida que el MCU le envía a la terminal de reinicio del flip-flop para que sea desactivada.

Las señales que reconocen la señal de interrupción son INTVALIDCOMM.1, INTGO, INTAZS, que son señales de entrada al MCU (terminal 24, terminal 25 y terminal 26).

Las señales que identifican la señal de interrupción para desactivarla, son señales de salida del MCU y son IDENTGO, IDENTVALIDCOMM, IDENTAZS (terminal 30, terminal 31 y terminal 32).

Cabe mencionar que el sistema de interrupción fue diseñado de esta forma, porque el MCU cuenta con una interrupción externa, si las señales de entrada a

éste sistema estuvieran conectadas a la misma terminal de interrupción, el MCU no podría identificar la señal de interrupción que se está solicitando.

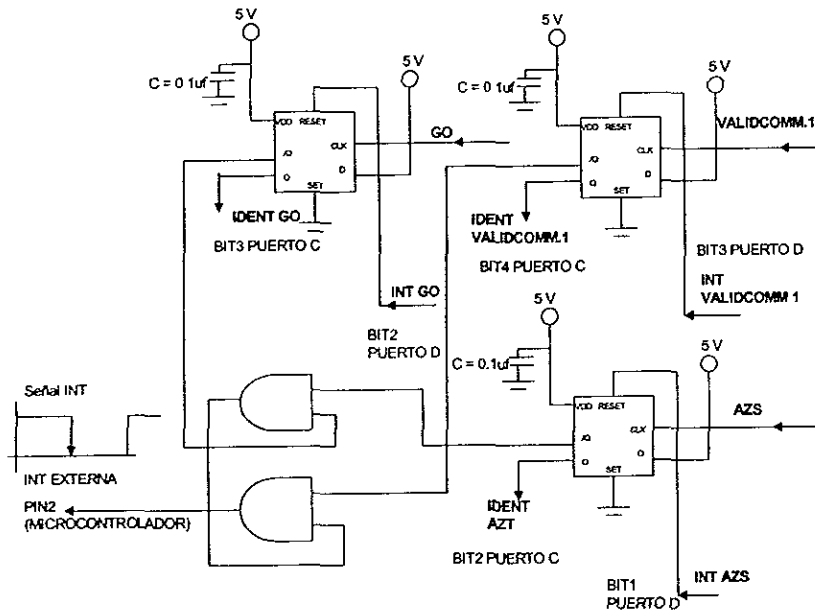


Figura 3-4. Sistema de interrupción del sistema de control de rotación.

El dato del azimuth de la sonda AZS, es una señal de entrada que llega en forma de trenes de pulsos hacia el Timer (terminal 37) del MCU, el timer se utiliza para llevar un conteo del número de pulsos enviados por el cartucho de orientación y obtener el dato del azimuth de la sonda.

En el diagrama electrónico de la figura 3-1, podemos observar que la señal AZS llega también a la entrada de un circuito monoestable (4538) en configuración redispensible, a su vez la señal /Q de salida del monoestable (terminal 7) es conectada a la señal de reloj de un Flip-flop D. Al primer flanco de subida de la señal AZS, el circuito monoestable envía un pulso como señal de salida (terminal 7), con una duración mínima 150 μ s.

La duración del pulso de salida del monoestable es de 150 μ s, y se dispara dependiendo del número de pulsos contenidos en la señal AZS, el periodo de esta señal es de 50 μ s, por lo que si en cada 3 pulsos de entrada al monoestable, éste recibe otro flanco de subida de AZS, el monoestable se disparará de nuevo, haciendo que el tiempo de duración sea dos veces el tiempo mínimo (300 μ s) y así sucesivamente, es decir, cada 150 μ s el monoestable se disparará siempre y

[illegible]

El circuito monoestable se utiliza para que el timer del MCU capture la señal AZS sin perder ningún pulso, y posteriormente solicitar una interrupción para actualizar el dato de entrada AZS. La configuración redispensible se usa para que la petición de la interrupción sea más rápida, ya que si la configuración fuera no redispensible, con tan solo un pulso de la señal AZS, el monoestable se tendría que esperar un tiempo de 18 ms ($360 \text{ pulsos} \times 50 \mu\text{s}$) para solicitar una interrupción, que es el tiempo máximo que tarda el cartucho de orientación para enviar 360 pulsos de la señal AZS.

- 45 -

Señales de entrada	Señales de salida
Bit8 AZD *	Bit8 AZT *
AZS *	AZT *
DDATA IN	DDATA OUT
UDATA IN	UDATA OUT
/UCLK	STEP *
/DCLK	DIR *
WFHS *	GO' *
Zin *	OE *
VALID COMM.1	IDENTVALIDCOMM.1 *
GO	IDENTGO *
AZD *	IDENTAZS *
INTVALIDCOMM.1 *	
INTGO *	
INTAZS *	
* señales de E/S del microcontrolador	

TABLA 3-1. Señales de E/S de la sección Control de Rotación.

3.1.2. Señales de entrada y salida de la sección Control de Elevación y Excitación de motores

A continuación describiremos y mencionaremos las señales de entrada y salida del sistema de control electrónico de elevación y del sistema de excitación de los motores a pasos, basándonos en el diagrama electrónico de la figura 3-2.

La señal MSW1 es una señal generada por la interfase del microinterruptor, es una señal de entrada al sistema de control de elevación y llega al bit4 del puerto C.

Las señales /UCLK y /DCLK, son señales de entrada al sistema de control de elevación, la primera entra a la terminal 10 del registro de corrimiento (P/S) y la segunda llega a la terminal 3 del registro de corrimiento (S/P).

El bit8 AZD es una señal de salida que proviene de la terminal 11 del registro de corrimiento (S/P) y es enviada al MCU de la sección Control de Rotación.

El bit8 AZT es una señal de entrada que proviene del sistema de rotación y llega a la terminal 7 del registro de corrimiento (P/S).

La señal UDATA IN es una señal de entrada que proviene del sistema de rotación y llega a la terminal 11 del registro de corrimiento (P/S), esta señal contiene el dato de elevación del transductor ELEV.

La señal DDATA OUT es una señal de salida que proviene de la terminal 10 del registro de corrimiento (S/P) hacia la terminal 2 del registro de corrimiento (S/P) del sistema de rotación, esta señal lleva la información de AZD y de elevación deseada ELEVCOM.

La señal UDATA OUT, es una señal de salida del sistema de elevación y llega a la terminal 11 del registro de corrimiento (P/S).

La señal DDATA IN, es una señal de entrada al sistema de elevación, y proviene del primer registro de corrimiento (S/P) que se encuentra montado en otra tarjeta electrónica del cartucho electrónico.

La señal GO' es una señal de entrada al sistema de elevación proveniente del sistema de control de rotación, llega a la terminal 9 del registro de corrimiento (P/S) de este sistema.

El dato de elevación deseada ELEVCOM sale del registro de corrimiento (S/P) y es leído en forma paralela por el puerto A del MCU, dicho dato es de 7 bits, debido a que el rango de medición es de 0 a 90°. El dato de elevación del transductor ELEV, sale del puerto B del MCU y es leído por el registro de corrimiento 4021, para ser enviado hacia la superficie.

El sistema de interrupción está formado por una compuerta NOR (4001) y dos flip-flop's D (74HC74). Las salidas Q de los flip-flop's D se conectan a la compuerta NOR, cuya salida está conectada a la terminal de interrupción externa del MCU (terminal 2). Dependiendo de la señal de interrupción que se active, VALID COMMAND.1 ó GO, se realizará un cambio de estado de 0 a 1 lógico en la salida Q de cada flip-flop, para activar la interrupción externa del MCU. La señal Q activada es leída por el MCU para que éste reconozca la interrupción solicitada, y ejecute la rutina de interrupción.

Las señales de interrupción VALID COMMAND.1 y GO están conectadas a la señal de reloj de cada flip-flop D que cargan el dato D a cada flanco de subida en la entrada de su reloj. Una vez reconocida la señal de interrupción, el MCU envía a los flip-flop's un 1 lógico para limpiarlos, las señales que se envían se llaman IDENT VALIDCOMMAND.1 (terminal 31), e IDENTGO (terminal 32). En la figura 3-6 se muestra el sistema de interrupción del circuito de elevación.

Las señales DIR_ELEV y STEP_ELEV son señales de salida del sistema de control de orientación, y están conectadas al arreglo de transistores Darlington contenidos en el circuito ULN2001 que se encuentra en el circuito de excitación de motores. El sistema de excitación de motores tiene como señales de entrada las señales DIR_ELEV, STEP_ELEV, DIR_ROT y STEP_ROT, generadas por los microcontroladores de cada sistema de control.

Las señales de salida del sistema de excitación, son las secuencia de fases de los motores a pasos generadas por los circuitos integrados SAA1027.

Las fases del circuito de excitación del motor de elevación Q1, Q2, Q3, y Q4, pasan a través de un arreglo de transistores Darlington que se encuentran contenidos en el circuito integrado ULN2001, para proporcionar la demanda de corriente requerida por el motor.

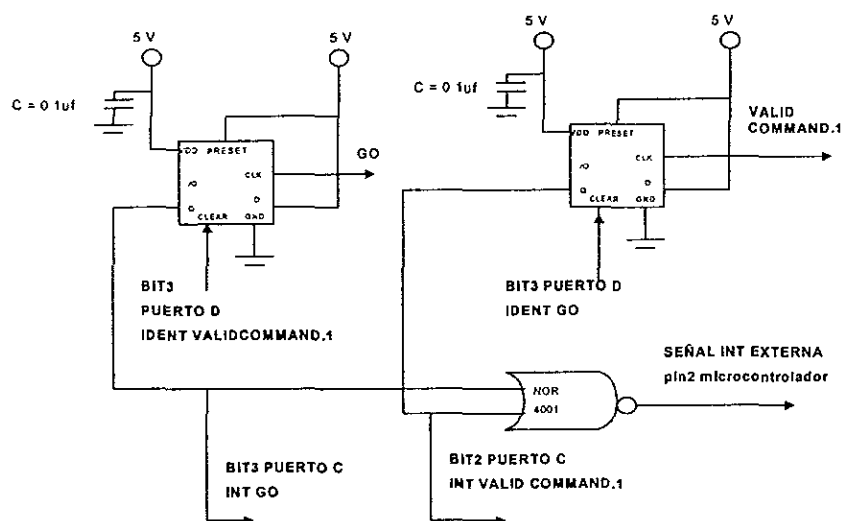


Figura 3 -6. Sistema de interrupción del circuito de control de elevación

Las fases de salida del circuito excitador del motor de rotación fase A, fase B, fase C y fase D, pasan por una etapa de potencia para suministrar una corriente de 0.34 amperios por cada fase del motor, la etapa de potencia está formada por cuatro transistores tipo MOSFET (metal-óxido semiconductor FET) contenidos en un módulo de potencia CPY400 diseñados específicamente para aplicaciones donde el modo de excitación de los motores es unipolar. En la tabla 3-2 se muestran las señales de entrada y salida del sistema de control de elevación y del sistema de excitación de motores.

3.1.3. Estructura interna del programa original de Rotación

El programa de rotación realiza el control del movimiento de rotación de la parte inferior de la sonda mecánica, las funciones principales que realiza el programa

Sistema de control de elevación		Sistema de excitación de motores	
Entrada	Salida	Entrada	Salida
MSW1*	DDATA OUT	DIR (rotación)	FASE A
/UCLK	UDATA OUT	STEP (rotación)	FASE B
/DCLK	STEP*	DIR (elevación)	FASE C
BIT8 AZD	DIR*	STEP (elevación)	FASE D
UDATA IN	IDENT VALIDCOMM.1*		FASE Q1
VALIDCOMM.1	IDENTGO*		FASE Q2
GO	ELEV*		FASE Q3
DDATA IN			FASE Q4
GO'			
INT			
VALIDCOMM.1*			
INTGO*			
ELEVCOM*			
BIT8 AZT			
STEP (rotación)			
DIR (rotación)			
* señales de E/S del microrcontrolador			

TABLA 3-2. Señales de E/S de las secciones Control de Elevación y Excitación de motores.

son, leer los datos AZD y AZS, calcular el valor AZT y generar el número de pulsos y sentido de avance correctos para colocar la herramienta en la posición deseada. El programa rotación fue desarrollado en lenguaje ensamblador TASM05 de motorola.

La estructura interna del programa de rotación está dentro de un ciclo condicionado a cumplir una expresión lógica, en la figura 3-7 se muestra a grandes rasgos la estructura interna del programa. A continuación desglosaremos cada uno de los bloques que forman la estructura interna del programa de rotación.

En el bloque de rutinas de Inicialización, se realizan cuatro rutinas, la primera rutina se llama POWRESET, en ella se definen e inicializan todas las variables en RAM que se van a requerir para guardar datos en forma temporal, los puertos de entrada/salida y los registros de control del contador. La segunda rutina se llama RETARINI, genera un retardo de 8.38 segundos, porque durante el encendido de la herramienta, los voltajes que alimentan a las tarjetas electrónicas tardan en establecer su valor nominal. Los bloques que forman las rutinas de inicialización se muestran en la figura 3-8.

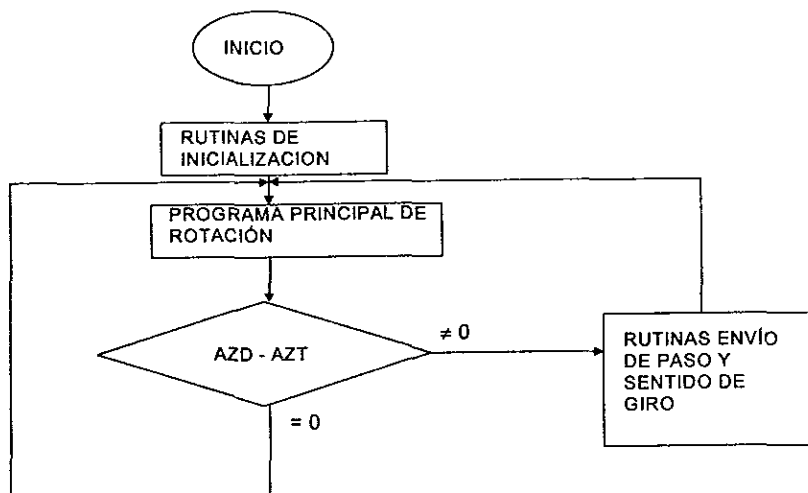


Figura 3-7. Diagrama de flujo de la estructura interna del programa de rotación

La tercera rutina CEROENCOD, se encarga de buscar el cero de referencia (Zin) al detectar la señal Zin proporcionado por el sensor de referencial.

La forma en la que se busca el cero de referencia es detectando el flanco de subida de la señal Zin, leyendo el cambio de estado en el bit7 del puerto C, si la señal Zin (ver figura 3-9) está en nivel bajo, se envían pulsos al motor en sentido positivo (en sentido de las manecillas del reloj) hasta detectar la subida del pulso, pero si la señal Zin está en nivel alto se envían pulsos positivos hasta detectar la bajada del pulso. Una vez detectada la bajada del pulso, se procede a detectar la subida del pulso enviando pulsos positivos, de tal forma que se obliga a encontrar el cero de referencia en el sentido de giro positivo.

La cuarta rutina se llama RETARELEV, se genera un retardo de 75 segundos, durante este tiempo el sistema de control de elevación ejecuta sus rutinas de inicialización.

En el siguiente bloque se tiene el Programa principal de rotación, que se muestra en el diagrama de flujo 3-10, en éste se realizan los cálculos para determinar el número de pulsos y el sentido de giro necesarios para colocar al transductor en la posición deseada. Está constituido de varias subrutinas que a continuación explicaremos.

El programa principal se inicia haciendo un sondeo de interrupciones, ejecutando

a función de interrupción que es solicitada, la rutina de interrupción realiza tres funciones: leer el dato AZD, escribir el dato AZD en el puerto A y actualizar el valor de la sonda AZS.

Posteriormente se realiza el cálculo del dato AZT en la rutina AZMTRANS, el dato AZT se calcula llevando el conteo de los grados girados (RBE) por el motor de rotación más el dato del azimuth de la sonda con respecto al norte magnético, es decir:

$$AZT = RBE + AZS \quad (3.1)$$

Debido a esta sumatoria, al dato AZT se le restan 360° , para tener un dato de salida en el rango de 0 a 359° .

RUTINAS DE INICIALIZACIÓN

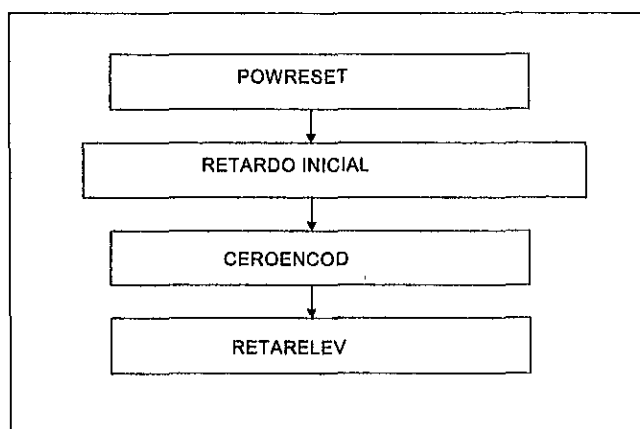


Figura 3-8. Diagrama de flujo de la estructura interna de las rutinas de inicialización del programa de rotación.

Si la señal WFHS, es detectada por el bit4 del puerto B, la rutina del programa principal se queda encerrada en un ciclo realizando el sondeo de las interrupciones y actualizando el dato AZT, hasta que el tiempo de duración de la señal de la ventana sónica WFHS haya finalizado, esto es para evitar la inducción de ruido durante la transmisión y recepción de la señal ultrasónica.

Una vez detectado la señal WFHS, se realiza el cálculo del valor de delta mediante la rutina DELTA, el cálculo es la diferencia entre el valor del azimuth deseado y el valor del azimuth del transductor, esto es:

$$\Delta = AZD - AZT \quad (3.2)$$

El valor de delta puede ser positivo, negativo o cero. Cuando el valor de delta es 0°, significa que el transductor se encuentra en la posición deseada. Si el valor de delta es negativo, el valor de AZD es menor que AZT, por lo que el transductor tendrá que desplazarse delta grados hacia el sentido negativo (contrario al de las manecillas del reloj); sin embargo, el cálculo de delta no nos indica la trayectoria más corta que debe de seguir el transductor, para ello se le suman 180 grados a delta y dependiendo del signo del resultado de la suma se define el sentido de giro y el número de pulsos necesarios para que el transductor se coloque en la posición deseada, es decir:

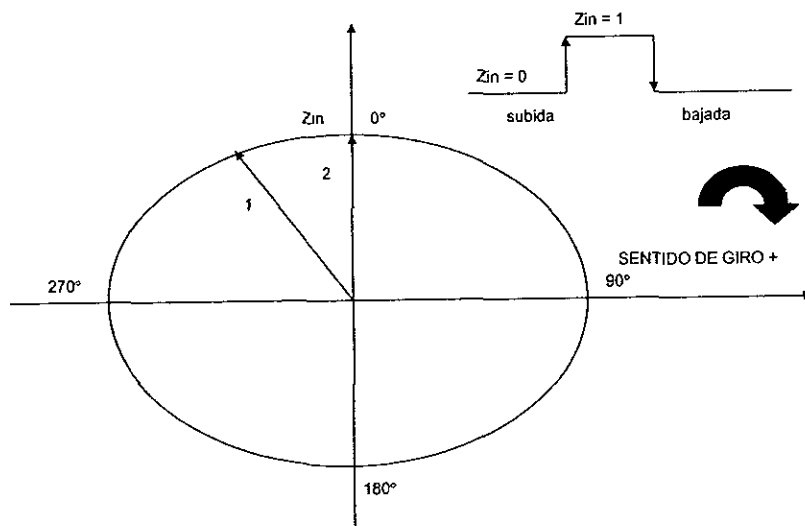


Figura 3-9. Localización del cero de referencia de la señal Zin.

Para

$$\Delta < 0$$

$$\Delta' = \Delta - 180^\circ \quad (3.3)$$

Si $\Delta' < 0$, entonces la trayectoria más corta es en sentido positivo.

Si $\Delta' > 0$, entonces la trayectoria más corta es en sentido negativo.

Para el caso cuando el valor de delta es positivo, se realiza el mismo procedimiento para el cálculo de la trayectoria más corta, a diferencia que en lugar de restarle el valor de 180 grados a delta se le suman, esto es:

Para $\Delta > 0$

$$\Delta' = \Delta + 180^\circ \quad (3.4)$$

Si $\Delta' < 0$, entonces el transductor se desplaza en sentido positivo.

Si $\Delta' > 0$, entonces el transductor se desplaza en sentido negativo.

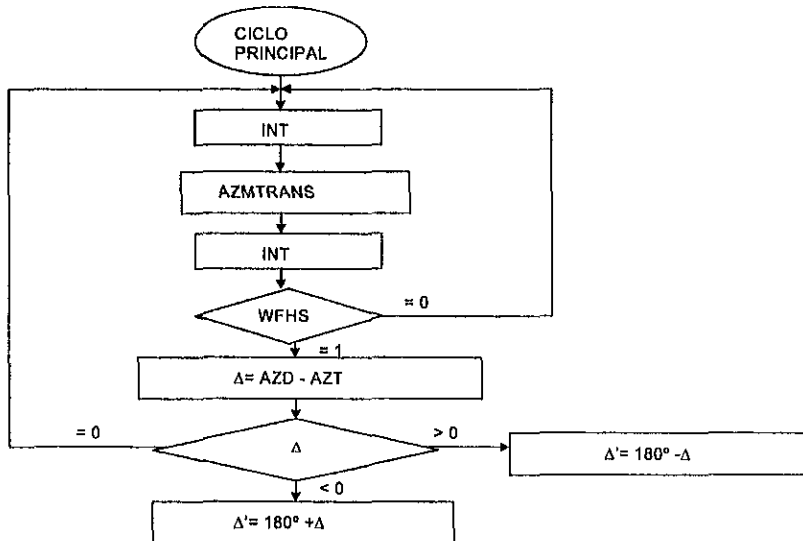


Figura 3-10. Diagrama de flujo de la estructura interna de las rutinas del programa principal de rotación.

En el siguiente bloque se tiene las rutinas de Envío de paso y sentido de giro, y se muestran en el diagrama de flujo 3-11, tienen como función generar el pulso de avance y el sentido de giro, se compone de tres rutinas, PPOS, PNEG y ACT_DATO.

La rutina PPOS se ejecuta cuando el valor de Δ' es negativo, en esta rutina se envía un pulso positivo cada 11.5 ms y se cuenta el número de pulsos enviados,

cada 5 pulsos enviados indica que el transductor se ha desplazado 1° ; cada vez que se desplaza 1° se habilitan dos banderas, una de ellas es para que el dato RBE se actualice y la otra es para indicar el sentido de giro. Al final de esta rutina se detecta si hay un cambio de estado en la señal de Zin, si se tiene un cambio de 0 a 1 lógico se ejecuta la rutina RESETENCOD, cuya función es inicializar el dato RBE, indicando que el motor de rotación paso por la marca de cero de referencia.

La rutina PNEG, se ejecuta cuando Δ' es positivo y realiza las mismas funciones que la rutina PPOS, solo que en éste caso, el sentido del pulso es negativo, y la rutina RESETENCOD se ejecuta cuando la señal Zin va de un estado de 1 a 0 lógico.

ENVIO DE PASOS Y SENTIDO DE GIRO

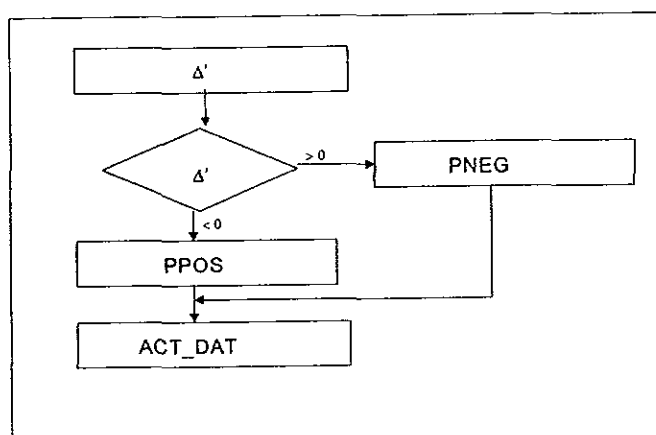


Figura 3-11 Diagrama de flujo de la rutina de Envío de pasos y sentido de giro de la estructura interna del programa de rotación

El ángulo de paso del motor de rotación, es de 1.8° por paso, dicho dato es un dato especificado por el fabricante, proporcionado en las especificaciones técnicas del motor de rotación modelo PH268L-23, fabricado por VEXTA. Lo anterior significa que por cada pulso de entrada el motor se desplaza 1.8° , sin embargo en las rutinas PPOS y PNEG, se requiere de 5 pulsos para lograr que el motor se desplace 1° , la razón por la cuál se necesitan 5 pulsos es porque se tiene un sistema de engranes conectado a la flecha del motor a una relación de 1:9.

El retardo de 11.5 ms, es el periodo de la señal de excitación STEP, la frecuencia por consiguiente es de 89 Hz.

La tercera rutina ACT_DAT, tiene como función principal actualizar el dato RBE. El dato RBE se actualiza dependiendo el sentido del movimiento que realizó el motor; si se enviaron 5 pulsos en sentido positivo, el valor RBE se incrementa en 1, pero si los pulsos se enviaron en sentido negativo, el valor de RBE se decrementa en 1.

Si el valor de RBE es de 359° y el motor se desplaza 1° en sentido positivo, el dato RBE se inicializa con cero grados, o bien, si el valor RBE es de 0° y el motor se desplaza 1° en sentido negativo, el valor RBE se inicializa con 359°.

3.1.4. Estructura interna del programa original de Elevación

El programa de elevación realiza el control del movimiento de elevación del transductor ultrasónico vertical, tiene como señales de entrada la señal de detector de cero SW1 y el dato ELEVCOM, y como señales de salida el dato de elevación real ELEV y las señales de excitación al motor, STEP y DIR. El programa de elevación se desarrolló en lenguaje ensamblador TASM05 de motorola.

En la figura 3-12, se muestra la forma en la que está estructurado el programa de elevación, podemos observar que al igual que la estructura interna del programa de rotación, se encuentra dentro de un ciclo condicionado a cumplir una expresión lógica. A continuación analizaremos bloque por bloque la estructura interna del programa elevación.

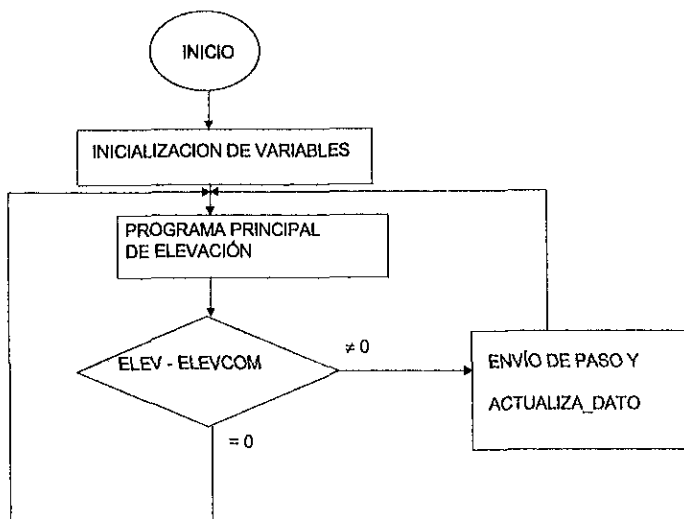


Figura 3-12. Diagrama de flujo de la estructura interna del programa de elevación.

En el bloque de Inicialización de variables, se definen y se limpian las variables en RAM y los puertos de entrada/salida, después se ejecuta un retardo inicial de 29.36 segundos.

En el siguiente bloque se tiene el Programa principal de elevación, el cuál comienza sensando el estado del bit4 del puerto C para saber el estado de la señal del microinterruptor SW1, si se tiene un 1 lógico, significa que el patín del transductor se encuentra cerrado, es decir, el transductor se encuentra en la posición de 0°, de lo contrario, si se tiene un 0 lógico en el bit4 del puerto C, el transductor aún no ha llegado a cero, por lo que se envían los pulsos negativos necesarios hasta lograr que el patín se cierre.

En la práctica al utilizar la herramienta en campo no siempre el microinterruptor se cierra, ya que por desajustes de piezas mecánicas no se logra hacer el contacto del transductor con el microinterruptor, por lo que, cuando el patín no se cierra y el valor ELEV alcanza el valor de cero, el programa de elevación seguirá enviando pulsos en sentido negativo indefinidamente, para evitar esto, se envían 255 pulsos en sentido negativo para detener el motor e iniciar con el ciclo normal de operación. En la figura 3-13, se muestra el sentido de giro del transductor vertical y la señal SW1 detectando cero grados de elevación.

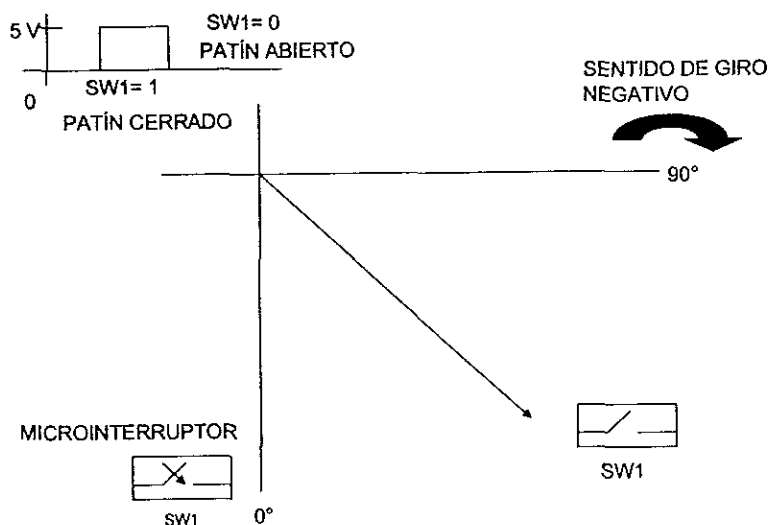


Figura 3-13. Localización del cero de referencia de la señal SW1.

Una vez que se coloca el transductor vertical en la posición de 0° , se habilitan las interrupciones para saber cual es el valor del dato ELEVCOM leído por el puerto A, o bien enviar el dato ELEV a través del puerto B. Posteriormente se compara el valor ELEVCOM con el valor ELEV, y dependiendo de ésta comparación se define el número de pulsos a enviar en el sentido de giro apropiado.

De la comparación:

Si el valor es $ELEV > ELEVCOM$, entonces se envía un pulso en sentido de giro negativo.

Si el valor es $ELEV < ELEVCOM$, entonces se envía un pulso en sentido de giro positivo.

Una vez que se definen el número de pulsos y el sentido de giro necesarios, se ejecuta un retardo de 10.2 ms para generar la frecuencia de la señal de excitación STEP, el bloque Envío de paso y actualiza dato, genera la frecuencia de la señal de excitación de 98 Hz, y actualiza el dato ELEV.

Cuando se termina de ejecutar el retardo, se solicita una interrupción por software, para llevar un conteo del número de pulsos enviados, cada 25 pulsos, el transductor se desplaza 1° y dependiendo del sentido de giro el valor ELEV se incrementa o decrementa en 1, la razón por la cual se envían 25 pulsos es porque se tiene un sistema de engranes conectado a la flecha del motor a una relación 1:45.

Como podemos ver, la estructura interna de ambos programas son diferentes, el programa de rotación está diseñado bajo una estructura lógica más ordenada por medio de llamados de subrutinas, en cambio el programa de elevación no utiliza subrutinas en su diseño.

3.2. Diseño y desarrollo del sistema de orientación modificado del cartucho electrónico

El control de las funciones del sistema de orientación se realiza en forma independiente, es decir, se tiene un MCU para cada movimiento de control; sin embargo un solo MCU tiene la capacidad de realizar ambas funciones de control, por lo que una de las formas en las que se puede optimizar el sistema, es realizar un software totalmente integrado que realice las funciones de movimiento de rotación y elevación del transductor ultrasónico principal. La integración del software implica la eliminación de componentes electrónicos en el sistema, y por consiguiente el sistema de orientación podría integrarse en una sola tarjeta electrónica, para decidir cuál es el camino más viable en el diseño del sistema de orientación, fue necesario hacer un análisis de las ventajas y desventajas que se tendrían al considerar tal integración. En la tabla 3-3, se muestran los aspectos

que se tomaron en cuenta en el análisis de las ventajas y desventajas al considerar la fusión del sistema.

	1 tarjeta	2 tarjetas
Desarrollo del programa	*	
Colocación de tarjetas en SONIMPI y SONIMPII		*
Tiempo de ensamble de tarjetas	*	
Pruebas de funcionamiento En laboratorio	*	
Localización de fallas en el mantenimiento	*	
Población o densidad de componentes en la tarjeta		*
Elementos externos de la tarjeta		*
Dimensiones de la tarjeta	*	
* Ventaja		

TABLA 3-3. Ventajas y desventajas consideradas al fusionar el sistema de orientación.

En el Desarrollo del programa es una ventaja tener un software totalmente integrado en un solo MCU, por el hecho de utilizar un MCU se tiene un programa total para la depuración del sistema, una tarjeta electrónica, y por consiguiente se tiene un menor número de dispositivos electrónicos; en cambio, si se tiene un software independiente en la realización de las funciones de rotación y elevación se tendrían dos microcontroladores, que implicaría tener dos tarjetas electrónicas, realizar la depuración en cada programa de control, y un mayor número de componentes.

En la Colocación de tarjetas en SONIMPI y SONIMPII, si el sistema se fusiona en una sola tarjeta, las señales de E/S de los conectores que se encuentran fijos en el cartucho electrónico de las herramientas SONIMP, se tendrían que cambiar, modificando el cableado e invirtiendo un mayor tiempo en la realización de pruebas. En cambio si el sistema de orientación utiliza las dos tarjetas, se tendrían una sustitución directa de las tarjetas y por consiguiente se discriminarían problemas en el cableado, siendo esto una desventaja en la utilización de una sola tarjeta.

Se tiene un menor tiempo de ensamble en el programa del sistema integrado y un menor tiempo en la realización de pruebas, porque no se tienen dos programas en forma independiente.

Las pruebas de funcionamiento en laboratorio se realizan en una sola tarjeta por lo que la probabilidad de falla es menor, en cambio al tener dos tarjetas el tiempo de pruebas aumenta y la probabilidad de falla también.

La Localización de fallas en una tarjeta disminuye, debido a la eliminación de posibles componentes electrónicos generadores de error, en cambio al tener las dos tarjetas se tiene el mismo número de componentes electrónicos que producen error.

La Población o densidad de componentes en las tarjetas, aumenta al integrarlo en una sola tarjeta, debido a que se tiene mayor población en las pistas y un mayor número de componentes, al tener las dos tarjetas del sistema se tiene igual o menor población de pistas y/o componentes.

En los Elementos externos de la tarjeta, se tendría un posible incremento de ellos al integrar el sistema en una sola tarjeta, en cambio con las dos tarjetas del sistema los elementos externos del cartucho no varían.

En las Dimensiones de la tarjeta, si se utiliza una tarjeta la dimensión de ésta puede ser igual o menor a las actuales.

3.2.1. Fusión del sistema de orientación del cartucho electrónico

Debido al análisis presentado con respecto a las ventajas y desventajas que se tienen al considerar la integración del programa en un solo MCU y los dispositivos electrónicos del sistema de orientación instalados en una sola tarjeta, podemos observar que la integración del sistema podría ser una solución más viable en nuestro diseño.

Sin embargo, se decidió realizar un sistema de control que realice ambas funciones de rotación y elevación mediante un solo MCU, desarrollando la constitución física del sistema en dos tarjetas electrónicas, porque se tiene una instalación directa en el cartucho de la herramienta, evitando la modificación de los conectores fijos que se encuentran montados en ella, ahorrándose el tiempo invertido en problemas de cableado y en la realización de pruebas. Cabe mencionar que se tienen más de dos herramientas SONIMPI y SONIMPII, en las cuales se les instalará dicho sistema de orientación, por lo cuál el fusionar la constitución física de este sistema no sería la solución más viable debido a que se tendrían que modificar los conectores fijos en los cartuchos electrónicos en cada herramienta.

Para hacer la integración del sistema de orientación, se deben definir las variables de entrada y salida, utilizando el total de las señales de entrada y salida (E/S) de los circuitos de Control de Rotación y Elevación descritas al inicio de este capítulo, para hacer una simplificación y/o eliminación de estas señales.

La simplificación y/o eliminación de señales de E/S se obtuvieron de la siguiente forma:

El sistema de control de rotación utiliza un bus bidireccional de 8 bits, usando dos líneas de control adicionales GO' y OE para leer y escribir los datos del azimuth a través del mismo puerto del MCU, de esta misma forma se puede utilizar un bus de datos bidireccional de 8 bits, para leer y escribir los datos del azimuth y de elevación provenientes de los registros de corrimiento utilizando el mismo puerto del MCU, añadiendo dos líneas más de control GO'' y OE' para los datos de elevación.

Al utilizar un mismo bus de datos, el bit8 de AZD y el bit8 de AZT podrán ser leídos directamente de los registros de corrimiento del dato elevación a través del bus, eliminándose 16 bits utilizados por el dato de ELEV y ELEVCOM, y los 2 bits más significativos del dato del azimuth.

En la figura 3-14 se muestra el bus de datos bidireccional, que permite leer y escribir los datos de azimuth y elevación.

Las señales de entrada y salida de la sección Excitación de motores no fueron modificadas, por lo que ésta sección permanecerá con el mismo diseño, en la siguiente tabla 3-4, se muestran las señales E/S del sistema de orientación fusionando las secciones Control de Rotación y Control de Elevación.

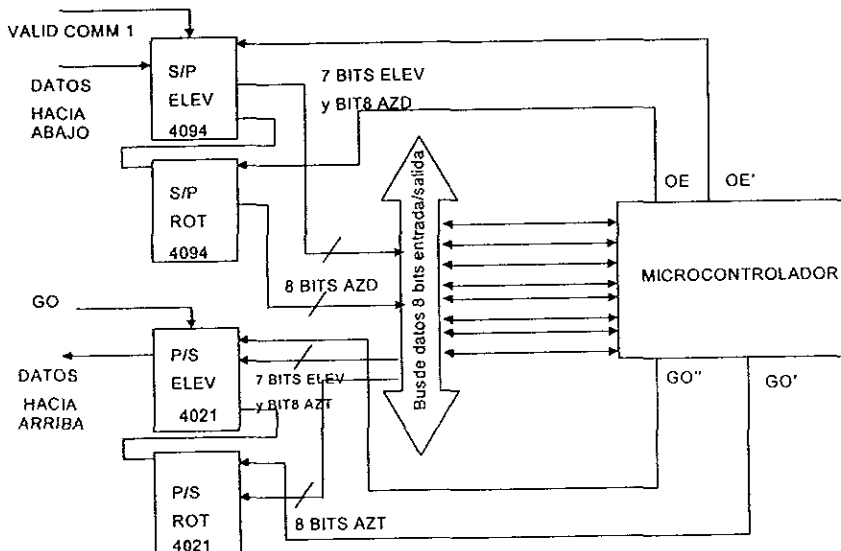


Figura 3-14. Flujo de datos hacia arriba y hacia abajo a través del bus

El número de líneas de E/S requeridas por el MCU son 26, es decir, 7 líneas de entrada, 11 líneas de salida, y 8 líneas bidireccionales de E/S. Una vez definido las señales E/S del sistema integrado, seleccionaremos el MCU que se encargará de realizar los movimientos de rotación y elevación del transductor ultrasónico principal.

Señales de entrada	Señales de salida	Señales de entrada /salida Por el bus de 8 bits
SW1*	GO**	AZD*
Zin*	GO**	AZT*
WFHS*	OE**	ELEVCOM*
AZS*	OE**	ELEV*
VALID COMMAND.1	DDATA OUT	
GO	UDATA OUT	
/UCLK	DIR_ROT*	
/DCLK	DIR_ELEV*	
UDATA IN	STEP_ROT*	
DDATA IN	STEP_ELEV*	
INT VALIDCOMMAND.1*	IDENT VALIDCOMMAND.1*	
INTGO*	IDENTGO*	
INTAZS*	IDENTAZS*	

* señales de E/S del microcontrolador

TABLA 3-4. Señales de E/S del sistema de orientación modificado.

3.2.2. Selección del microcontrolador

Los requerimientos mínimos que debe contar el MCU para nuestro diseño son: frecuencia máxima de operación de 1 MHz, contador de 8 bits, un sistema de interrupción externo, 26 líneas de entrada y salida, capacidad de memoria 1 KB en ROM y 100 Bytes en RAM.

Teniendo como referencia los requerimientos mínimos para desarrollar ambas funciones de control, podemos observar que el dispositivo de control a escoger no necesita ser muy poderoso; sin embargo, en la mayoría de los MCU's comerciales, si aumenta el número de puertos en él, sus características internas, la capacidad de memoria y sus funciones adicionales son de mayor capacidad.

En las siguientes tablas, podemos ver las características generales de algunos MCU's fabricados por los líderes en el mercado popular en México, Microchip, Motorola, Intel y Atmel. Para mayor información técnica de cada MCU, consultar las páginas de internet cuyas direcciones electrónicas se muestran en la Bibliografía.

DESARROLLO DEL SISTEMA DE ORIENTACIÓN

Familias 8bits series	Memoria de Programa (ROM) OTP bytes	Memoria de datos RAM Bytes	Frecuencia máxima de operación MHZ	Líneas de entrada y salida (I/O)	Contadores 8 y/o 16 bits
PIC12CXX	768 hasta 1792	25 hasta 80	4	6	1
PIC16C5X	576 hasta 3072	25 hasta 73	4,20	12,20	
PIC16CXX	896 hasta 7168	36 HASTA 192	20	13,22,33	1,2,3
PIC16CXXX	896 hasta 7168	36 hasta 192	8,10,20	13,22,52	1,3
PIC17CXXX	4096 hasta 32768	232 hasta 902	33	33,50	4

TABLA 3-5. Características generales de microcontroladores fabricados por MICROCHIP.

Familias 8 bits series	Memoria de programa ROM (Kbytes)	Memoria de datos RAM (bytes)	Frecuencia máxima de operación MHZ	Líneas de entrada y salida. (I/O)	Contadores 16 y/o 8 bits
MC68HCX05BXX	6 hasta 7.7	176 hasta 11.5 K	2.1, 4	24,26, 32	1
MC68HCX05CXX	8 hasta 16	176 hasta 352	4	31	1
MC68HCX05CDX X	32	352	2.1	31	1
MC68HCX05LXX	6 hasta 16	256 hasta 512	2.1	20,39	1
MC68HCX05PXX	2 hasta 4.5	128 hasta 256	2.1,2.5	21	1
MC68HCX05RXX	8 hasta 16	192 hasta 352	2.1	20,32	1
XC68HC08A2XX	32	1 hasta 2	8	48	1
PC68HC908JL3	4 (Flash)	128	8	23	1
PC68HC908MR24	24 (Flash)	512	8	44	1

TABLA 3-6. Características generales de microcontroladores fabricados por MOTOROLA.

Series MCS151/251	Memoria del programa ROM(KB)	Memoria de datos RAM (bytes)	Frecuencia máxima de operación MHz	Líneas de entrada y salida (I/O)	Contadores 16 y/o 8 bits
8X2515A	8	1K	32	16	3
8XC251SB	16	1k	32	16	3
8XC251SP	8	512	32	16	3
8XC251SQ	16	512	32	16	3
8XC251TA	8	1k	32	24	3
8XC251TB	16	1k	32	24	3
8XC251TP	8	512	32	24	3
8XC251TQ	16	512	32	24	3
Series MCS51					
80C31BH	4	128	32	12, 16	2
80C51BH	4	128	32	12, 16	2
87C51	4	128	32	12, 16	2
8XC51FA	8	256	32	12, 16	3
8XC52/54/58	8,16,32	256	32	12, 24, 33	3

TABLA 3-7. Características generales de microcontroladores fabricados por INTEL.

AVR Familias 8-bits	Memoria del programa ROM (Kbytes)	Memoria de datos RAM (bytes)	Frecuencia máxima de operación MHz	Líneas de entrada y salida (I/O)	Contadores 16 y/o 8 bits
AT90S1200	1		16	15	1
AT90S2313	2	128	20	15	2
AT90S4433	4	128	8	20	1
AT90S8515	8	512	4-20	32	2
AT90S8535	8	512	8	32	2
Atmega103	128	4k	6	48	3

Tabla 3-8. Características generales de microcontroladores fabricados por ATMEL.

Como podemos observar en las tablas previamente mostradas, los MCU's mencionados son productos de tecnología de vanguardia, buscando mejorar la velocidad de procesamiento, menor densidad de código, ejecución de instrucciones empleando el menor número de bytes, mayor eficiencia en la compilación, mayor desempeño, bajo costo, entre otros. De las tablas podemos ver también, que existe al menos un dispositivo de control de cada fabricante que

se ajusta a nuestras necesidades. Por ejemplo en la primera tabla 3-5, el MCU de la familia PIC16CXXX con 896 Bytes ROM, 192 Bytes en RAM, 33 líneas E/S; en la siguiente tabla 3-6, los MCU's MC68HCX05BXX y MC68HCX05CDXX. En nuestro caso, seleccionamos el MCU AT90S8515 fabricado por ATMEL, no solo porque es un MCU de los más modernos y nuevos en el mercado popular con arquitectura RISC, sino también porque se contaba con un kit de desarrollo y un número mayor de éstos dispositivos en el almacén de componentes de Geofísica, en donde me encuentro desarrollando éste trabajo.

Las características del MCU AT90S8515 son las siguientes:

Alto desempeño y baja potencia de consumo en arquitectura RISC

120 instrucciones

8 Kbytes de memoria interna programable Flash

- Interface serial de perifericos (SPI:serial interface peripheral) para grabar la memoria Flash.
- Capacidad de 100,000 $\frac{\text{escritura}}{\text{borrado}}$ ciclos

512 bytes de memoria EEPROM interna

- Duración de 100,00 $\frac{\text{escritura}}{\text{borrado}}$ ciclos

512 bytes SRAM interna

32 registros de propósito general de 8 bits

32 líneas programables de E/S

UART serial programable

Interface con SPI (Serial Peripheral Interface)

Voltaje de alimentación V_{CC} = 2.7 hasta 6.0 volts

Frecuencia máxima de operación 20MHz

Tiempo de ciclo de instrucción 50ns

1 contador de 8 bits con preescalador

1 contador de 16 bits con preescalador, modos de comparación y de captura

Función Modulación de ancho de pulso

Interrupciones externas e internas

Sistema de tiempo perro guardián (watchdog) programable

Un comparador analógico

Modos de apagado

Seguridad en el software

Encapsulados PDIP, PLCC y TQFP

En el apéndice A.4, se muestran las especificaciones técnicas del MCU AT90S8515, para una mayor referencia. A continuación, se mostrará el diseño total de la constitución física del sistema de orientación modificado, indicando así mismo el número de componentes electrónicos empleados en éste diseño y las señales de E/S.

3.3 Constitución física del sistema de orientación modificado del cartucho electrónico

Antes de describir cada uno de los componentes electrónicos que forman el diseño físico del sistema de orientación modificado y los cálculos realizados, explicaremos con la ayuda de un diagrama esquemático la funcionalidad y distribución de dicha constitución física.

En el diagrama esquemático mostrado en la figura 3-15 (a) Sección A, tenemos el primer bloque formado por los registros de corrimiento cuya función es la de la transferencia de datos de y hacia la superficie; el siguiente bloque está formado por el MCU cuya función es controlar los movimientos de ambos transductores; y por último tenemos un pequeño bloque formado por un circuito monoestable cuya función es generar un tiempo de retardo necesario para que la señal proveniente del cartucho de orientación pueda ser leída por el MCU.

En la figura 3-15 (b) Sección B, tenemos dos bloques, el primer bloque formado por los circuitos manejadores de los motores a pasos SAA1027 cuya función es generar la secuencia de fases de los motores y el segundo bloque formado por el circuito de potencia ULN2001 del motor de elevación cuya función es suministrar la demanda de corriente del motor.

A continuación describiremos las conexiones físicas e identificaremos las terminales de E/S previamente establecidas, para cada sección del sistema de

orientación modificado, basándonos en los diagramas electrónicos de cada una de las secciones.

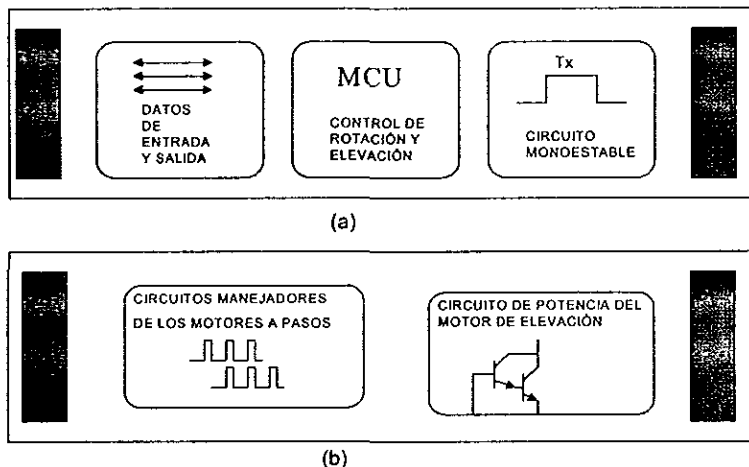


Figura 3-15. Diagrama a bloques de la constitución física del sistema de orientación modificado del cartucho electrónico. (a) Sección A. (b) Sección B.

3.3.1 Sección A del sistema de orientación modificado

La sección A (figura 3-16), se encarga de realizar las funciones de control de rotación y elevación y está conformada por los siguientes dispositivos electrónicos.

El dispositivo principal de control lo constituye el MCU, las señales de E/S están definidas de la siguiente forma:

El puerto A está conectado al bus de datos bidireccional de E/S, a través del bus pasan los datos AZD, AZT, ELEVCOM y ELEV provenientes de los registros de corrimiento entrada paralelo/salida serie y entrada serie/salida paralelo.

Al bit PB1 del puerto B se conecta la señal AZS proveniente del cartucho de orientación y del circuito monoestable.

Inicialmente el puerto B se utilizó para el control de las señales que reconocen e identifican las señales de interrupción (INTVALID COMMAND.1, INTGO, INTAZS, IDENT VALID COMMAND.1, IDENT GO, IDENT AZS), sin embargo el MCU cuenta con tres señales de interrupción externas que nos permitió eliminar 6 líneas más de control, y realizar tres rutinas de interrupción de forma independiente.

RECEIVED



Del puerto D se utilizan cinco bits para señales de entrada, el bit PD0 para sensar la señal WFHS, el bit PD2 funcionando como terminal de interrupción externa para la señal VALID COMMAND.1. El bit PD3 funcionando como terminal de interrupción externa para la señal GO, el bit PD6 para leer la señal de pulso SW1 y el bit PD7 para leer la señal de cero de referencia Zin.

El puerto C se utiliza para generar las señales de salida, el bit PC0 proporciona la señal de paso STEP_ELEV, el bit PC1 genera el sentido de giro DIR_ELEV, el bit PC2 la señal de paso STEP_ROT, el bit PC3 la señal DIR_ROT, el bit PC4 genera el pulso GO", el bit PC5 proporciona la señal de pulso GO', el bit PC6 la genera la señal OE' y por último el bit PC7 proporciona la señal OE.

La terminal 35 del MCU, es la terminal de interrupción externa de evento de captura, y se conecta como señal de entrada la salida Q proveniente de la terminal 6 del circuito monoestable.

El MCU utiliza un sistema de reinicio automático conectado a la terminal 10, mediante un arreglo RC externo, el MCU permanece en estado de reinicio mientras el terminal externo /RESET se encuentre en un nivel bajo ($V_{IO} \leq 0.8$ volts), y se desactiva al tener un nivel alto de voltaje mínimo ($V_{IH} \geq 2.7$ volts). El sistema de reinicio automático está formado por un arreglo externo Resistencia-Capacitor, donde el voltaje a la entrada de la terminal del /RESET alcanza un nivel alto de voltaje después de T segundos, éste tiempo de retardo es provocado por la carga y la descarga del capacitor a través de la resistencia y se calcula mediante la fórmula, (3.5):

$$T_x = R_x C_x \quad (3.5)$$

La expresión (3.5), se llama constante de tiempo del circuito, es decir, es el tiempo donde el valor de la corriente de salida decae 0.37 de su valor inicial.

El tiempo mínimo de retardo durante el encendido del /RESET es de 11 ms, que es el tiempo mínimo para asegurar que el voltaje en la entrada de la terminal del /RESET es mayor que el voltaje de umbral ($V_{pot}=1.8$ volts) necesario para realizar la operación de un reinicio automático. Fijando el valor de $R=100K\Omega$, y $T_x=15$ s, el valor de C_x , se obtiene despejando la fórmula (3.5):

$$C_x = \frac{T_x}{R_x} = \frac{15}{100K} = 150\mu f \quad (3.6)$$

El valor T_x se estableció de 15 s, porque durante el encendido de la herramienta, el voltaje de alimentación se suministra variando una perilla de control hasta colocarla a un valor de 250 volts de a.c., durante éste tiempo los voltajes de c.d. que alimentan a las tarjetas del cartucho electrónico no han alcanzado un valor

estable, esto provocaría estados indefinidos de voltajes en los circuitos digitales generando señales de salida no deseables. El diodo IN914, se conecta en paralelo a la resistencia R, para proteger al MCU de algún corto circuito en la fuente de poder, esto es, porque en algunas ocasiones el apagar la fuente de voltaje, ocurren pequeños transitorios que provocan un corto circuito en pequeños instantes de tiempo y si el diodo no estuviera, la carga almacenada en C se descargaría en la terminal /RESET del MCU suministrando corriente mayores a 10 mA, que dañarían dicha terminal.

La señal de reloj que alimenta al MCU está proporcionada por un cristal de cuarzo encapsulado de 4 MHz, al cuál se conectan dos capacitores de 22 pF de acuerdo con los datos técnicos del fabricante.

Los siguientes componentes electrónicos son los registros de corrimiento 4094 y 4021. El CI MC4094 es un registro de corrimiento retenedor (latched) de 8 bits (CMOS), que realiza la conversión de datos de entrada en serie hacia la derecha a salidas de datos en paralelo, la función del latch permite el almacenamiento temporal de la información binaria. Los datos seriales entran por la terminal D y son desplazados en los flancos positivos de la señal de reloj CLK, los bits de entrada son almacenados en un registro temporal siempre y cuando la señal STR (strobe) se encuentre en 1 lógico, dichos datos podrán ser leídos hasta que la señal de control OE (output enable) tenga un nivel 1 lógico, de lo contrario se tendrá una alta impedancia en las terminales de los datos de salida en paralelo. El CI cuenta con dos salidas seriales para sistemas en cascada, la salida serial Qs se utiliza para el cascadeo a altas velocidades y la salida /Qs se utiliza a bajas velocidades.

La frecuencia máxima de operación del CI es de 1.25 MHz, un sistema en cascada se considera a altas velocidades si el tiempo de la señal de reloj tiene un valor menor o igual a 5µs, un sistema se considera a bajas velocidades si el tiempo de la señal de reloj es mayor o igual a 15µs (dichas consideraciones son datos que proporciona el fabricante); en nuestra aplicación el tiempo de la señal de reloj DCLK es de 10µs, por consiguiente el sistema de cascadeo se conecta mediante la terminal /Qs. El bit más significativo se lee desde la terminal Q1 del registro y por consiguiente el bit menos significativo se lee desde la terminal Q8, esto es porque el bit menos significativo del dato DDATA es enviado primero.

El CI MC4021, es un registro de corrimiento retenedor de 8 bits, que realiza la conversión de los datos de entrada en paralelo en forma asíncrona a datos seriales de salida en forma sincronizada con la señal de reloj CLK. El CI tiene una terminal de control P/S (paralelo/serial), la cuál al estar en 1 lógico almacena los datos de entrada en forma paralela, al pasar a un nivel 0 lógico, realiza la transferencia de los datos en forma serial sincronizados con los flancos positivos de la señal CLK. También cuenta con tres terminales de bits de salida Q6, Q7, Q8, que permite leer el flujo de datos a partir de la terminal de salida seleccionada,

para esta aplicación el flujo de datos se lee desde el bit 8 del registro, y por medio de la terminal Q8 se hace el cascadeo de los registros. El bit más significativo se lee desde la terminal H, y el bit menos significativo se lee desde la terminal A. La frecuencia máxima de operación del CI es de 3 MHz.

Otro componente electrónico es el monoestable MC14538, que se utiliza para generar un pulso de salida de duración fija T_x a partir de una señal de disparo de entrada, la señal de disparo puede ser un flanco positivo o negativo, la duración del pulso es la constante de tiempo del arreglo externo $R_x C_x$, y se calcula mediante la expresión (3.5).

De las especificaciones técnicas el valor mínimo que puede tomar R_x es de $4\text{ k}\Omega$, y los valores del capacitor C_x se encuentra en el rango $5000\text{ pf} \leq C_x \leq 100\text{ }\mu\text{ f}$. La duración del pulso de salida está limitado por el siguiente rango, $10\text{ }\mu\text{s} \leq T_x \leq 10\text{ s}$.

El CI monoestable puede operar en dos formas, en configuración disparable y en configuración redisparable. La primera configuración genera el pulso T_x al primer disparo de entrada $+T$ o $-T$ según sea el caso. La segunda configuración genera un pulso cuya duración varía dependiendo del número de disparos en la señal de entrada válidos, es decir, a cada flanco de subida (o bajada) el pulso de salida se mantiene en alto hasta que ya no detecte otro disparo en la entrada. La configuración empleada en el sistema de orientación modificado es en configuración redisparable, y el cálculo de la constante de tiempo se determinó de la siguiente forma:

De las especificaciones técnicas del fabricante se recomienda el valor de R_x de $100\text{ k}\Omega$, la duración del pulso se fija a un valor de $150\text{ }\mu\text{s}$ (3 veces el período de la señal AZS), por lo que a partir de estos datos el valor del capacitor C_x se obtiene de la fórmula (3.5):

Despejando C_x de la ecuación (3.5):

$$C_x = \frac{T_x}{R_x} \rightarrow C_x = 0.0015\text{ }\mu\text{f} \quad (3.7)$$

En la terminal 4 de entrada del monoestable se tiene una que permite fijar el nivel lógico de voltaje, lo que evita estados indefinidos en la señal de entrada, que provocaría valores de voltaje no deseados. Generalmente el valor de dicha resistencia se recomienda de entre $1\text{ k}\Omega$ y $100\text{ k}\Omega$, mientras mayor sea su valor, se disipa menos potencia y se limita el flujo de corriente hacia la carga, que en nuestro caso actúan como carga la terminal 4 del monoestable y la terminal 3 del MCU.

Los capacitores que se conecta entre la fuente de alimentación y tierra se llaman capacitores de desacoplo o de paso, se utilizan para eliminar las espigas de ruido

provenientes de las fluctuaciones de la fuente cuyo tiempo de duración es tan corto que no pueden ser filtradas por el regulador de voltaje. Dichos capacitores almacenan la energía suficiente para suministrar la energía a unos pocos CI's durante la espiga de corriente, manteniendo el voltaje de alimentación próximo a su valor nominal. El número y el valor de los capacitores de desacople dependen de la familia lógica empleada, y se recomiendan de 0.1 μ f.

3.3.2 Sección B del sistema de orientación modificado

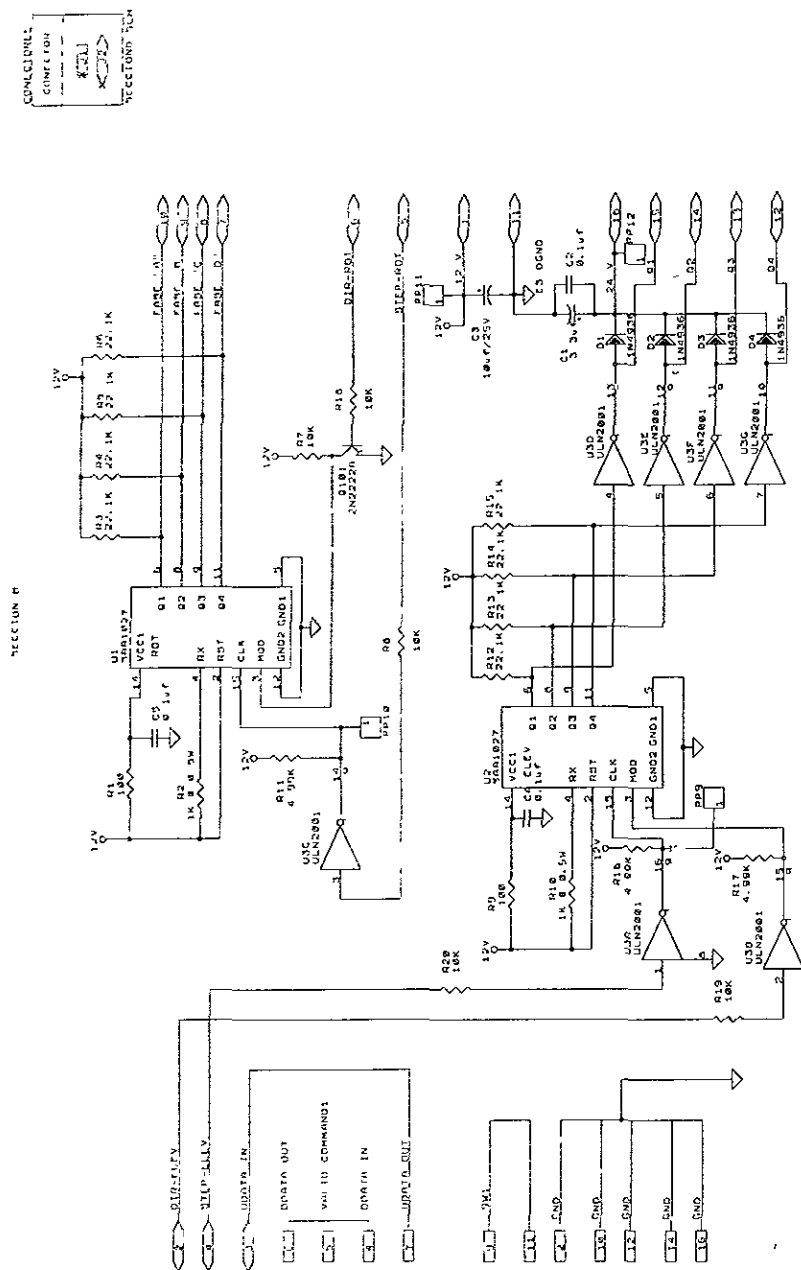
La sección B (figura 3-17), está formada por los circuitos manejadores de los motores de rotación y elevación, y por los circuitos que suministran la demanda de corriente hacia los motores.

Los CI's SAA1027 son circuitos manejadores de motores a pasos, que generan las secuencias de las señales que se conectan a los embobinados de los motores pasos para excitarlos y generar su movimiento, estas señales son pulsos de corriente cuyo valor de frecuencia es proporcional a la velocidad a la que se encuentran operando. El CI es un dispositivo comercial, que genera cuatro señales de salida con la secuencia requerida para un motor a pasos bifásico; opera en dos sentidos de giro, sentido en dirección a las manecillas de reloj (sentido negativo) y en sentido contrario a las manecillas de reloj (sentido positivo). Requiere de dos señales de entrada, la terminal 15 (CLK) cuya señal de entrada es la señal digital cuadrada generada por el MCU y la terminal 3 (MOD) cuya señal controla el sentido de giro. Si presenta un 0 lógico girará en sentido positivo y si presenta un 1 lógico girará en sentido contrario. Los estados de las secuencias de los pasos cambian a cada transición positiva presentada en la señal de entrada a la terminal CLK (terminal 15), dicha señal es generada por el MCU para lograr un desplazamiento angular en los motores. En la siguiente tabla 3-9 se muestra la secuencia de las señales de salida considerando el sentido de giro.

Este CI se puede alimentar con un voltaje que va desde 9.5 hasta 18 volts, la corriente máxima de salida es de 500 mA, el mínimo voltaje de entrada para lograr un 1 lógico es de 7.5 V, el máximo valor de entrada que se requiere para lograr un nivel 0 lógico es de 4.5 V. Las señales de entrada a las terminales CLK y MOD requieren de un nivel de voltaje mayor a 7.5 V, por lo que se utilizan los transistores del circuito ULN2001.

La resistencia de 100 Ω (R1) y el capacitor de 0.1 μ f (C5) conectados a la terminal Vcc1, son valores especificados por el fabricante, se utilizan para evitar distorsiones en las secuencias lógicas de las fases de salida, que se producen en los embobinados del motor debido a los cambios de la señal de corriente.

Figura 3-17. Diagrama electrónico de la sección B del sistema de orientación modificado del cartucho electrónico.



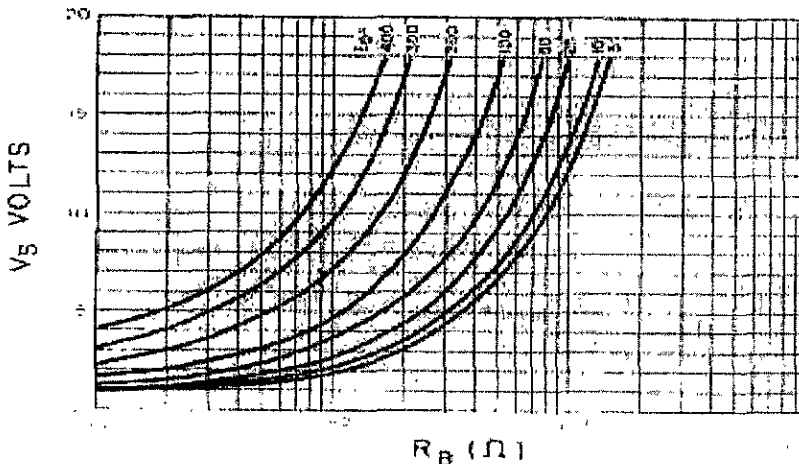
INSTITUTO MEXICANO DEL PETROLEO	
GERENCIA DE GEOFISICA	
Título	
TBI EXCITACION DE MOTORES	
Size	Document Number
0	PROYECTO 01-05512
REV	SONIMP II
0001	FECHAS 14-03-08 05-01-08

Diagrama electrónico de la sección B del sistema de orientación modificado

Secuencia de pasos	M = L (sentido positivo)				M = H (sentido negativo)			
	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4
0	L	H	L	H	L	H	L	H
1	H	L	L	H	L	H	H	L
2	H	L	H	L	H	L	H	L
3	L	H	H	L	H	L	L	H
0	L	H	L	H	L	H	L	H
L = 0 lógico				H = 1 lógico				

TABLA 3-9. Secuencia de las señales de salida del circuito SAA1027.

La resistencia externa (R_{10}) conectada a la terminal 4, es la resistencia de polarización que se requiere para controlar la demanda de corriente del motor, dependiendo de su valor será la corriente que fluirá a través de las terminales de salida. El valor de la resistencia R_x se determina a partir de una tabla proporcionada por el fabricante, y conociendo el valor del voltaje de polarización y el valor de la corriente de salida a los motores (I_Q). En la siguiente figura 3-18 se muestra la gráfica para determinar el valor de la resistencia R_x .

Figura 3-18. Gráfica para determinar la resistencia $R_B = R_x$ de polarización para el circuito SAA1027.

El valor de R_x en este diseño se escogió de $1k\Omega$ a $\frac{1}{2} W$, a partir del voltaje de polarización de 12 volts, y una corriente mínima de carga $I_Q = 5 \text{ mA}$.

Las señales de entrada a las terminales 15 y 3 del SAA1027, provienen del CI ULN2001, cuyo contenido es un arreglo de transistores en configuración Darlington empleado para elevar el nivel de voltaje de las señales STEP y DIR generadas por el MCU. Los transistores funcionan en este caso como circuitos de conmutación, por lo que las regiones de operación que nos interesan son saturación y corte. Para el cálculo de la resistencia de base R_B y la resistencia de colector R_C se deben cumplir las condiciones de la tabla 3-10.

Condiciones de corte	Condiciones de saturación
$I_B \leq 0$ (3.8)	$I_{Bmin} \geq \frac{I_{Csat}}{h_{FEmin}}$ (3.12)
$V_{BE} \leq 0$ (3.9)	$V_{BEsat} = 0.7 \text{ v (silicio)}$ (3.13)
$I_C = I_{fuga}$ (3.10)	$V_{CC} \gg V_{Cesat}$ (3.14)
$V_{CC} \approx V_{CE}$ (3.11)	$R_C = \frac{V_{CC}}{I_{Csat}}$ (3.15)
	$V_{BB} > \frac{(R_B)(V_{CC})}{(R_C)(h_{FEmin})} + V_{BEsat}$ (3.16)

Tabla 3-10. Parámetros de diseño de circuitos de conmutación para transistores en configuración emisor común.

Los parámetros de diseño son fáciles de obtenerse considerando los voltajes de polarización de las terminales del transistor para cada región. En la región de corte, las terminales base-emisor y base-colector deben estar polarizadas en inversa, y en la región de saturación las terminales base-emisor y base-colector en directa. En la figura 3-19 se muestra la configuración emisor-común de un transistor.

Para calcular el valor de las resistencias R_B y R_C , limitamos el valor de la corriente de salida menor a 4.5mA (valor máximo de corriente de entrada, dato proporcionado por el fabricante) para proteger al circuito SAA1027, por lo que el valor lo fijamos a 2.5 mA, los valores de la ganancia de corriente h_{FEmin} y voltaje colector-emisor de saturación V_{Cesat} , se obtiene de las especificaciones técnicas.

Cálculando R_B y R_C , tenemos lo siguiente:

* Los parámetros de diseño se obtuvieron del libro Electrónica IV, mostrado en la parte de Bibliografía.

Datos conocidos:

$$I_{Csat} = 2.5 \text{ mA}$$

$$V_{BB} = 5 \text{ V}$$

$$V_{CE} \approx 12 \text{ V}$$

$$h_{FEmin} = 400$$

$$V_{Cesat} = 1.2 \text{ V}$$

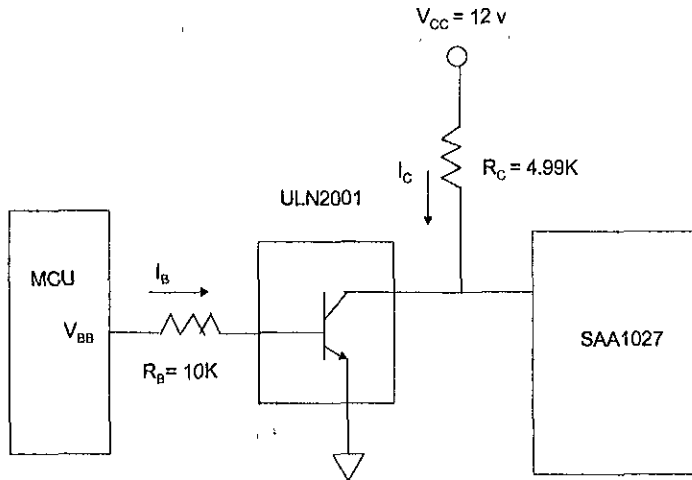


Figura 3-19. Configuración electrónica del circuito conmutador emisor común empleado para el cálculo de R_B y R_C .

De la expresión (3.15), obtenemos R_C :

$$R_C = \frac{V_{CC}}{I_{Csat}} = \frac{12}{2.5m} = 4.8K\Omega \quad (3.17)$$

De la expresión (3.12), se obtiene I_B :

$$I_B \geq \frac{I_{Csat}}{h_{FEmin}} \geq \frac{2.5m}{400} \geq 6.25\mu A \quad (3.18)$$

De la expresión (3.16) y despejando R_B , se tiene:

$$R_B \geq \left(\frac{V_{BB} - V_{BEsat}}{V_{CC}} \right) (R_C h_{FEmin}) \left(\frac{5-0.7}{12} \right) [(4.8k)(400)] \quad (3.19)$$

$$R_B (6.88k\Omega) \quad (3.20)$$

R_B , es el valor máximo que nos permite saturar al transistor, sin embargo, para asegurar la saturación, aumentamos la corriente en la base, para sobresaturar al transistor disminuyendo el valor de R_B . De las especificaciones técnicas del CI seleccionamos el valor de la $I_B = 500 \mu A$, con dicho valor calculamos R_B , a partir de la expresión (3.19), obteniendo un valor de $R_B \approx 10k\Omega$.

Las terminales de salida (Q1, Q2, Q3, Q4) de los CI's SAA1027 se conectan a las bases de cada transistor de potencia contenidos en el CI ULN2001, como se observa en la figura 3-17. El arreglo de transistores darlington contenidos en el CI ULN2001 se emplea para proporcionar corriente al motor de elevación y el módulo CPY400H contiene cuatro transistores MOSFET para proporcionar corriente al motor de rotación, cabe mencionar que éste último circuito no se encuentra en el diagrama electrónico de la figura 3-17, porque se encuentra montado en el chasis del cartucho electrónico.

Las resistencias de base R_B para cada transistor del CI ULN2001 fueron calculadas a partir de los datos técnicos del fabricante, es decir, para $I_B \geq 500\mu A$ se tiene $I_C = 350 \text{ mA}$, por lo que se tiene una $I_B = 511 \mu A$ (ver expresión 3.21). A partir de estos datos se obtiene R_B :

$$R_B = \frac{(V_B - V_{BEsat})}{I_B} = \frac{(12 - 0.7)}{511} = 22.1k\Omega \quad (3.21)$$

El circuito de excitación de los motores empleados se muestra en la figura 3-20, la resistencia R_M y la fuente de alimentación V_X , son elementos externos a nuestro sistema, y se encuentran montados en otra tarjeta electrónica de la herramienta SONIMP II, como los embobinados de los motores que se encuentran montados en la sonda mecánica de la herramienta, sin embargo para la explicación del cálculo de sus valores, es necesario apoyarnos de dicha figura.

La corriente máxima que consume el motor de elevación es de 0.303 amperios por fase y el de rotación es de 0.34 amperios por fase, por lo que la corriente total que se debe suministrar a cada motor es, dos veces la corriente de fase. La corriente total del sistema de motores es la suma de las corrientes de ambos motores, es decir:

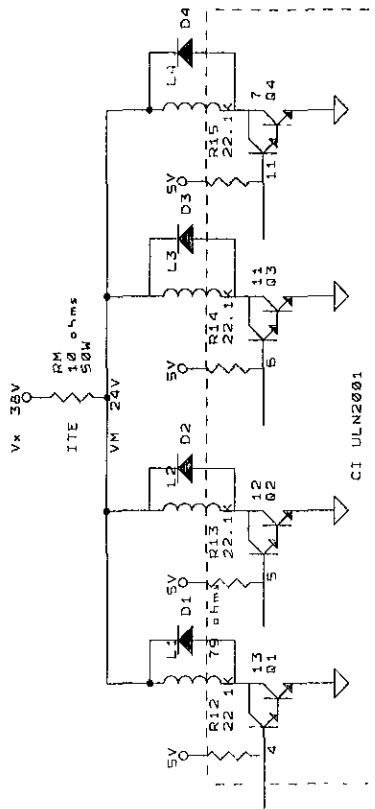
$$I_T = I_{TR} + I_{TE} \quad (3.22)$$

sustituyendo en la siguiente expresión:

$$I_T = 2(0.340) + 2(0.3030) = 1.286 \text{ A} \quad (3.23)$$

Figura 3-20. Circuitos de excitación de los motores a pasos.

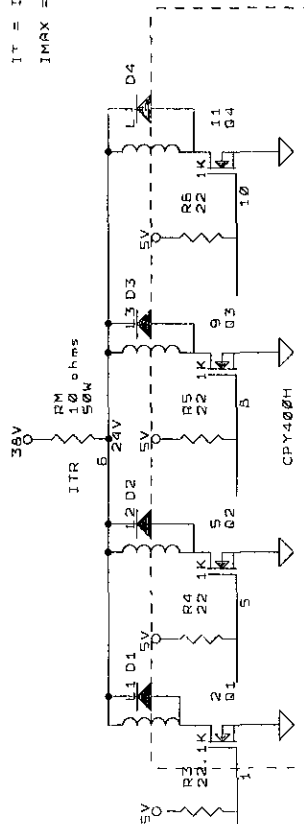
Circuito de excitación del motor de elevación



Circuito de excitación del motor de rotación

$$I^* = I^*_{TR} + I^*_{TE}$$

$$I^*_{MAX} = 1.4 A$$



INSTITUTO MEXICANO DEL PETROLEO		
Title	Circuitos de Excitación de motores	
Size	Document Number	PIV
A	Cartucho Electronico	
Date:	feb-2000	Sheet 01

Circuitos de excitación de los motores a pasos del sistema de orientación modificado del cartucho electrónico.

donde:

I_T , es la corriente total del sistema de motores

I_{TR} , es la corriente total del motor de rotación

I_{TE} , es la corriente total del motor de elevación

El valor de $V_X = 38$ volts, $I_{MAX} = 1.286$ A y el voltaje de polarización de los motores es de $V_M = 24$ V (especificado por el fabricante), por lo que calculando la resistencia R_M , se tiene:

$$R_M = \frac{(V_X - V_M)}{I_{MAX}} = \frac{(38 - 24)}{1.286} = 10.88\Omega \quad (3.19)$$

Por lo que eligiendo un valor comercial menor o igual a R_M calculado, tenemos que $R_M = 10 \Omega$.

Los circuitos de excitación de los motores empleados en el sistema de orientación, son de la forma más utilizada por su diseño simple y económico, sin embargo posee una gran desventaja debido a la gran disipación de potencia en la resistencia R_M , cuyo cálculo es el siguiente:

$$P_{R_{MAX}} = R I_{MAX}^2 = (10)(1.4)^2 = 19.6 \text{ w} \quad (3.23)$$

Una vez que se hizo el del hardware del sistema de orientación modificado, se diseñaron los algoritmos de control y el programa de control en lenguaje ensamblador, empleados para lograr el funcionamiento adecuado de los motores a pasos.

3.4 Diseño de algoritmos de control y diagramas de flujo del sistema de orientación

Inicialmente, los algoritmos de control para efectuar los movimientos de rotación y elevación, fueron diseñados bajo el mismo esquema de diseño utilizado en el programa de control de rotación a través de llamados de subrutinas, la unión de ambos programas se hizo mediante un ciclo principal condicionado a cumplir dos expresiones lógicas. La estructura del software para el sistema de rotación, se muestra en la figura 3-21.

La estructura lógica del diseño del sistema de orientación, se puede describir mediante la siguiente secuencia de pasos:

1. En el inicio del programa se efectúa un retardo inicial de 16 segundos, durante el cuál los voltajes que alimentan a las tarjetas electrónicas se estabilizan a su valor nominal.
2. Inicialización y definición de variables en memoria RAM, puertos de E/S, registros de control para el contador, configuración de los registros de interrupción.
3. Ejecución de la rutina Cero-Elevación, para colocar al transductor vertical a una inclinación de 0° .
4. Ejecución de la rutina Cero-Rotación, para colocar al transductor horizontal en la posición de cero de referencia.
5. Ejecución del ciclo principal del programa de rotación, para colocar al transductor horizontal en la posición ordenada desde la superficie. En este ciclo principal se hacen los cálculos para determinar el número de pulsos y el sentido de giro necesarios para mover al motor de rotación hasta lograr la siguiente expresión lógica:

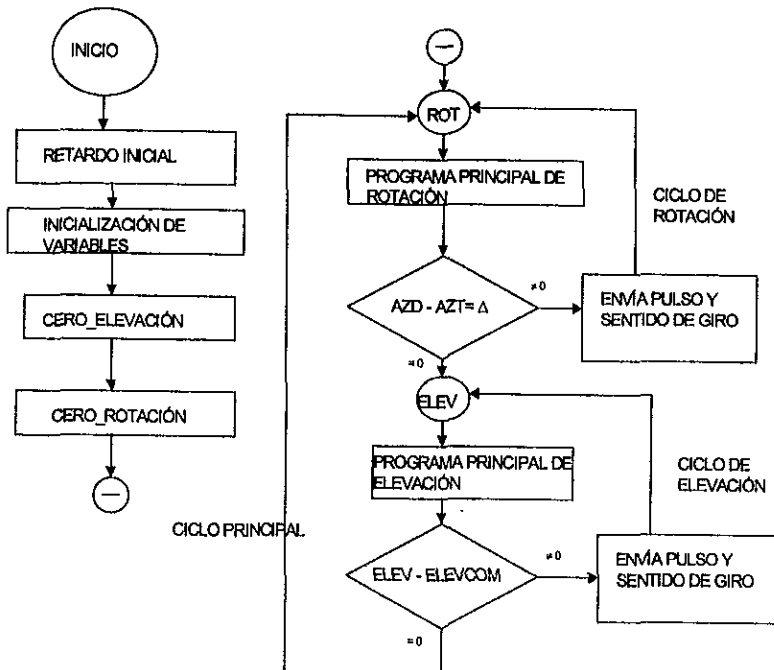


Figura 3-21. Diagrama de flujo del diseño total del sistema de orientación.

$$AZD - AZT = 0 \quad (3.24)$$

6. Ejecución del ciclo principal del programa de elevación, para colocar al transductor vertical en la posición deseada. En este ciclo principal se hacen los cálculos para determinar el número de pulsos y el sentido de giro necesarios para mover al motor de elevación hasta lograr que se cumpla la siguiente expresión lógica:

$$ELEV - ELEVCOM = 0 \quad (3.25)$$

Con esta secuencia de pasos se realizó el programa Rot_Elev, en lenguaje ensamblador AVR Assembler versión 1.2. El programa de desarrollo se muestra en el apéndice C.

Los bloques del diagrama de flujo, contienen las mismas rutinas utilizadas en el programa de rotación, a excepción de la rutina Cero_Elevación y la sección del ciclo principal de elevación que se implementaron mediante el desarrollo de subrutinas.

El desarrollo de este diseño no fue del todo satisfactorio, cuando se le envía una orden desde la superficie para mover el motor de rotación, el programa Rot_Elev se mantiene encerrado en el ciclo del programa principal de rotación hasta lograr que se cumpla la expresión lógica (3.24); mientras tanto, el ciclo principal de elevación tiene que esperarse a que se cumpla tal expresión para poder ejecutarse, en caso de recibir una orden desde la superficie. De la misma forma el ciclo principal de rotación tendría que esperarse hasta que se cumpla la expresión (3.25) para poder ejecutarse de nuevo.

Cuando la herramienta se encuentra funcionando en campo, el valor de AZS no siempre se mantiene fijo, generalmente varía entre un rango de $\pm 1^\circ$, por lo que esta variación origina que el valor de AZT no sea constante, y la expresión (3.24) del ciclo principal de rotación sea diferente de cero, provocando que el transductor horizontal esté siempre en movimiento, lo cual no permitiría realizar un movimiento de elevación, hasta que el valor de AZS se mantuviera constante. La solución a este problema, se resolvió mediante un solo ciclo principal Rot_Elev, que realiza el movimiento de ambos motores sin depender de las condiciones de las estructuras lógicas (3.24) y (3.25), es decir, eliminando el ciclo principal de rotación y el ciclo principal de elevación. En la figura 3-22 se muestra el diseño modificado del sistema de orientación.

Las rutinas PPOS y PNEG (explicadas en la sección 3.1.3) contenidas en el bloque Envío pulso y sentido de giro, ejecutan un retardo definido, para generar la señal de excitación STEP. Durante este retardo el MCU se encuentra en un estado "ocioso", esperando a que se terminé de ejecutar dicho retardo. Debido a esto, se cambió la filosofía del diseño en la estructura lógica del programa, con el

objetivo de hacer más eficiente el programa de control Rot_Elev, aprovechando el tiempo del retardo generado por ambas rutinas.

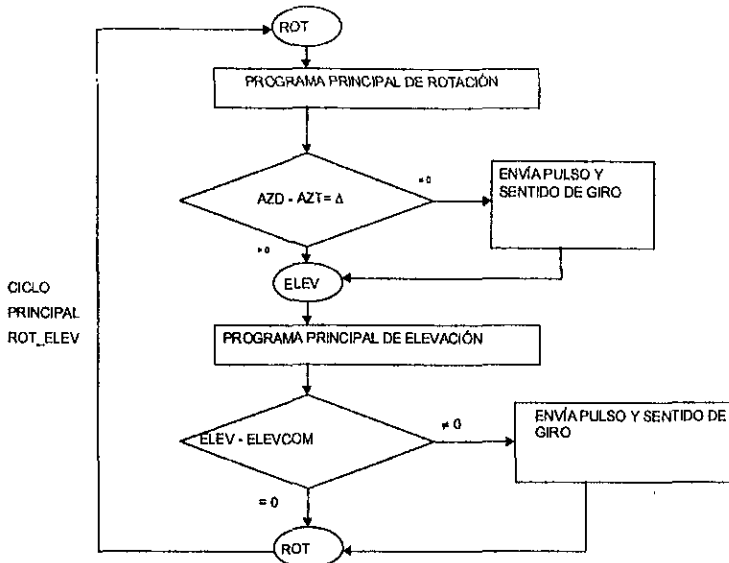


Figura 3-22. Eliminación del ciclo principal de rotación y el ciclo principal de elevación.

Con ésta nueva filosofía de diseño, la estructura interna del programa Rot_Elev resulta ser un poco más compleja, bajo éste esquema de diseño se utiliza el contador0 para generar un retardo de 10ms, durante el cual el programa realiza los cálculos necesarios para determinar el número de pulsos y sentido de giro apropiados para ambos movimientos de control. Cuando la cuenta del contador0 cambia de FFH a 00H, se tiene un sobreflujo en el contador0 y se habilita una interrupción por sobreflujo para ejecutar una rutina que genera los pulsos de avance para rotación y elevación siempre y cuando se soliciten. De esta forma, durante el tiempo de retardo generado en las rutinas PPOS y PNEG, el MCU se encuentra ejecutando otras tareas.

En la figura 3-23 se observa las señales de salida de excitación a los motores, obtenidas al generarlas por medio de la interrupción por sobreflujo.

Cabe mencionar que al usar esta forma de diseño, ambos motores se excitan a la misma frecuencia de operación, porque el contador1 es utilizado para hacer la captura de la señal AZS, si estuviera disponible dicho contador se podrían manejar dos frecuencias diferentes, pero en esta aplicación en particular, esta restricción no alteró el funcionamiento de los motores.

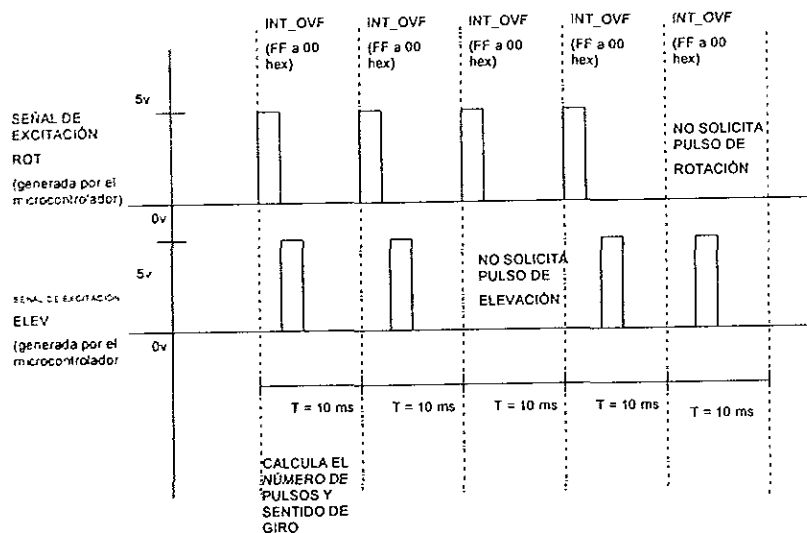


Figura 3-23. Señales de excitación obtenidas al realizarse un sobreflujo en el contador0.

A continuación describiremos los algoritmos de control del programa Rot_Elev bajo éste nuevo esquema de diseño, apoyándonos en el diagrama a bloques de la figura 3-23, en donde se muestra la estructura interna del programa Rot_Elev.

1. - INICIALIZACIÓN

En el bloque INICIALIZACIÓN se ejecutan cuatro rutinas de inicio, POWRESET, RETARINI, CERO_ELEV y CERO_ROT. Por medio de llamados de subrutinas se ejecutan las rutinas de inicialización en el siguiente orden:

1.1 POWRESET. En ella se inicializan y se definen las variables en memoria RAM, puertos de entrada y salida, registros de control para el contador configuración de los registros de interrupción.

1.2 RETARINI. Se genera un retardo inicial de 16.7 ms.

1.3 CERO_ELEV. En esta rutina se coloca en el registro del contador0, el número 63H, y se preescala el contador0 a 256, para generar el tiempo de retardo de 10ms, y solicitar los pulsos necesarios en cada sobreflujo, para colocar el transductor vertical a una inclinación de cero grados. Cada vez que se ejecuta la rutina de interrupción por sobreflujo (INT_OVF), se carga de nuevo el registro del contador0 con el valor de 63 Hex. Para detectar el cero de elevación, se realiza de la misma forma que el programa de Elevación

(descrito en la sección 3.1.4), detectando la señal SW1 del microinterruptor, la diferencia ahora es que para generar los pulsos de excitación, se solicitan a través del registro SOLIPULSE, cuyos bits de control se explicarán en la estructura del bloque MOVER_ELEV.

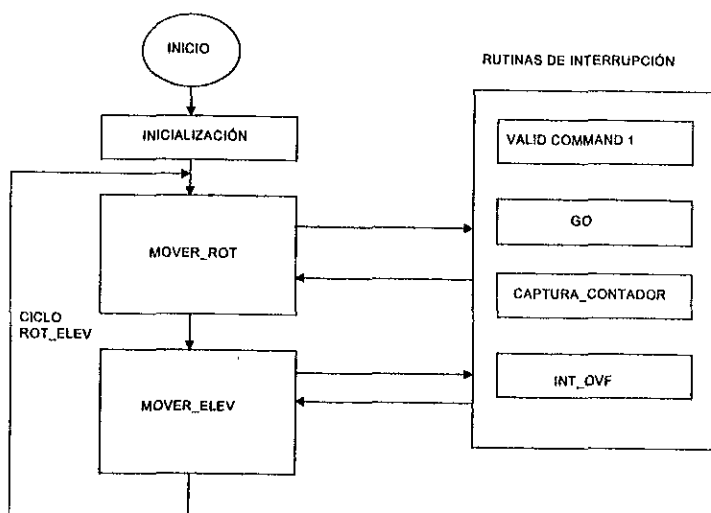


Figura 3-24 Estructura interna del programa Rot_Elev.

1.4 CERO_ROT. En esta rutina se busca la marca de referencia de cero, de la misma forma que el programa de rotación (descrito en la sección 3.1.3) detectando la señal de pulso Zin, la diferencia en ésta rutina es que para solicitar los pulsos de excitación se utiliza el registro SOLIPULSR, cuyos bits de control se explicarán en la estructura de la sección MOVER_ROT.

El cálculo del valor del registro de contador se determinó de la siguiente manera:

El ciclo de máquina del MCU es de 0.25us, y se desea generar una interrupción cada 10 ms, por lo que se necesita saber cuantos ciclos de máquina se requiere para generar un tiempo de 10 ms.

Los ciclos de máquina se obtienen dividiendo:

$$K = \frac{T_{\text{ciclo_máquina}}}{T_{\text{deseado}}} \quad (3.26)$$

donde:

K = número de ciclos para generar 10 ms.

Tciclo_máquina = ciclo de máquina del MCU, en segundos.

Tdeseado = tiempo de retardo de 10 ms.

Haciendo el cociente de la expresión (3.26), el valor de K es:

$$K = 40,000 \text{ ciclos de máquina} \quad (3.27)$$

La cuenta máxima del contador es de 256, por lo que es necesario aumentar el valor del ciclo de máquina, para ello se utiliza el preescalador del contador0, para dividir la frecuencia de operación. La frecuencia a la que opera el MCU es de 4 MHz, al dividirla 256 veces, la frecuencia de operación es de 15.625 kHz, por consiguiente el ciclo de máquina es de 64 μ s, haciendo el cálculo de la expresión (3.26), tenemos:

$$K = \frac{10\text{ms}}{64\mu\text{s}} \Rightarrow K = 156.25 \text{ ciclos de máquina} \quad (3.28)$$

Redondeando el valor de K a un valor entero:

$$K = 9DH \quad (3.29)$$

Si obtenemos el complemento del valor 157, se tiene que el valor de K¹:

$$K^1 = 63H \quad (3.30)$$

donde:

K' = es el número al cuál el contador0 debe iniciar su conteo para ejecutar 157 ciclos de máquina.

2. - MOVER_ROT

El bloque MOVER_ROT, está formado por la estructura de control que realiza los movimientos de rotación del transductor horizontal y podemos describirla mediante la siguiente secuencia de pasos:

- 2.1 Actualiza el valor de la bandera ZFLAG leyendo la señal Zin, para saber el estado actual de la señal de referencia cero, si por algún motivo el programa llegará a perder el conteo del número de grados avanzados por el motor, y éste pasa por la marca cero, inmediatamente coloca el valor de AZT a cero grados.

2.2 Calcula el valor del AZT, mediante la expresión (3.1) previamente definida:

$$AZT = AZS + RBE \quad (3.1)$$

donde:

RBE = es el número de grados girados por el motor de rotación.

AZS = es el valor del azimuth de la sonda.

2.3 Calcula el valor delta Δ , mediante la expresión (3.2), previamente definida:

$$\Delta = AZD - AZT \quad (3.2)$$

donde:

AZD = azimuth deseado.

AZT = azimuth del transductor.

Si $\Delta = 0$, entonces el programa continua en el bloque MOVER_ELEV.

Si $\Delta > 0$, entonces calcula $\Delta = \Delta - 180^\circ$ (3.3)

Si $\Delta < 0$, entonces calcula $\Delta' = \Delta + 180^\circ$ (3.4)

2.4 Cuando $\Delta' < 0$, se solicitan grados en sentido positivo y cuando $\Delta' > 0$ se solicitan grados en sentido negativo, mediante el registro SOLIPULSR, cuyos bits de control son los siguientes:

El BIT1 del registro se utiliza para solicitar un grado, es decir, 5 pulsos.

El BIT2 del registro, determina el sentido de giro.

El BIT0 del registro, indica 1 grado realizado

En la tabla 3-11, se muestran los estados de los bits de control del registro SOLIPULSR.

Una vez, que se solicita 1 grado en cualquiera de los sentidos, se lee de nuevo el estado de la señal de referencia Zin, para actualizar la bandera ZFLAG.

2.5 Actualización del dato RBE. Dependiendo del valor de los bits de control, se actualiza el dato de la siguiente forma:

Si $\text{bit1} \& \text{bit0} = 1$ y el $\text{bit2} = 0$, entonces el dato RBE se incrementa en 1 grado.

Si $\text{bit1} \& \text{bit0} = 1$ y el $\text{bit2} = 1$, entonces el dato RBE se decrementa en 1 grado.

Si $\text{bit1} \& \text{bit0} = 0$, no se actualiza el dato RBE y el programa continua en el bloque MOVER_ELEV.

3. – MOVER_ELEV

El bloque MOVER_ELEV contiene la estructura interna de control que realiza los movimientos de elevación del transductor vertical, su contenido puede describirse mediante la siguiente secuencia de pasos:

3.1 Lee el estado actual de la señal SW1 del microinterruptor, para colocar el valor de elevación real (ELEV_R) a cero grados.

3.2 Si los valores de ELEV_R y ELEV_D son igual a cero, entonces se solicitan 10 grados en sentido negativo utilizando el registro SOLIPULSE, para detener al transductor vertical, en caso de no detectarse la señal del microinterruptor.

3.3 Se calcula la siguiente expresión:

$$\text{ELEV}_R - \text{ELEV}_D = \Delta_{\text{elev}} \quad (3.31)$$

donde:

ELEV_R = el valor de elevación del transductor.

ELEV_D = el valor de elevación deseada.

Si el valor $\Delta_{\text{elev}} = 0$, entonces el programa continua en el bloque MOVER_ROT cerrando el ciclo principal Rot_Elev, para iniciar de nuevo.

3.4 Si $\Delta_{\text{elev}} > 0$, entonces se solicitan grados en sentido positivo.

Si $\Delta_{\text{elev}} < 0$, entonces se solicitan grados en sentido negativo.

La solicitud de pulsos, se realiza mediante el registro SOLIPULSE, cuyos bits de control se definen a continuación:

EL BIT1, se utiliza para solicitar 1 grado, es decir, 25 pulsos.

EL BIT2, se utiliza para determinar el sentido de giro.

EL BIT0, indica 1 grado realizado.

BIT2	BIT1	BIT0	Resultado
0	0	0	No solicita grado
0	1	0	Solicita grado (5 pulsos), en sentido positivo.
0	1	1	Avanzó 1 grado, se enviaron 5 pulsos en sentido positivo.
0	0	0	No solicita grado.
1	1	0	Solicita grado (5 pulsos), en sentido negativo.
1	1	1	Avanzó 1 grado, se enviaron 5 pulsos en sentido negativo.

TABLA 3-11. Estados de los bits de control del registro SOLIPULSR.

En la tabla 3-12, se muestran los estados de los bits de control del registro SOLIPULSE.

3.5 Actualización del dato ELEVR.

Si $\text{bit1} \& \text{bit0} = 1$ y el $\text{bit2} = 0$, entonces el dato ELEVR se incrementa en 1 grado.

Si $\text{bit1} \& \text{bit0} = 1$ y el $\text{bit2} = 1$, entonces el dato ELEVR se decrementa en 1 grado.

Si el $\text{bit1} \& \text{bit0} = 0$, no se actualiza el dato ELEVR y el programa continua en el bloque MOVER_ROT.

Si se desea tener un conocimiento más detallado de la estructura total del programa Rot_Elev, puede consultar los diagramas de flujo que se encuentran en el apéndice B.

3.4.1 Rutinas de interrupción del programa Rot_Elev

El programa Rot_Elev, ejecuta cuatro rutinas de interrupción, mostradas en la figura 3-25.

Al tener un flanco de subida en la terminal 14 de interrupción externa 0 (INT0), la

BIT2	BIT1	BIT0	Resultado
0	0	0	No solicita grado
0	1	0	Solicita grado (25 pulsos), en sentido positivo.
0	1	1	Avanzó 1 grado, se enviaron 25 pulsos en sentido positivo.
0	0	0	No solicita grado.
1	1	0	Solicita grado (25 pulsos), en sentido negativo.
1	1	1	Avanzó 1 grado, se enviaron 25 pulsos en sentido negativo.

TABLA 3-12. Estados de los bits de control del registro SOLIPULSE.

rutina de interrupción VALID COMMAND.1 se ejecuta, dicha rutina configura el puerto A (bus de datos) como entrada para leer las órdenes enviadas desde la superficie, genera dos señales de salida OE y OE', la primera se envía al registro de corrimiento S/P de rotación para leer los datos de AZD, la segunda se envía al registro de corrimiento S/P de elevación para leer los datos de ELEVD.

La siguiente rutina GO, se ejecuta con un flanco de subida en la terminal 15 de interrupción externa 1 (INT1) del MCU, genera dos señales de salida GO y GO' para escribir los datos obtenidos durante cada ciclo de operación en el puerto A y poder ser leídos por los registros de corrimiento P/S respectivamente. Los datos obtenidos son el valor de AZT y ELEVR.

La siguiente rutina de interrupción se llama Contador1 de Captura (Timer1 Capture). Su función es actualizar el dato del azimuth de la sonda AZS.

Para actualizar el dato AZS es necesario contar el número de pulsos enviados por el cartucho de orientación, el conteo de los pulsos lo realiza el contador de 16 bits, cuya señal es sincronizada con el reloj interno del MCU, esto es, el contador1 de 16 bits, utiliza la terminal 3 para leer una señal externa de reloj, la cual es leída en los flancos de subida del reloj interno del CPU (cada 0.25 us). El contador1 lleva la cuenta del número de pulsos leídos por la terminal 3 y la almacena en el registro del contador1 de 16 bits (TCNT1). Cuando se tiene un flanco de subida en la terminal 35 ICP (input capture pin), el valor del registro TCNT1 se transfiere al

registro ICR1(registro del evento de captura), y se habilita la interrupción Contador1 de captura. Al activarse dicha rutina, el dato del registro ICR1 es el valor del número de pulsos enviados por la señal AZS, la cuál fue leída por la terminal externa del contador1 (terminal 3). El dato AZS llega en formato n+1, por lo que se le resta 1 grado para obtener el valor real de AZS.

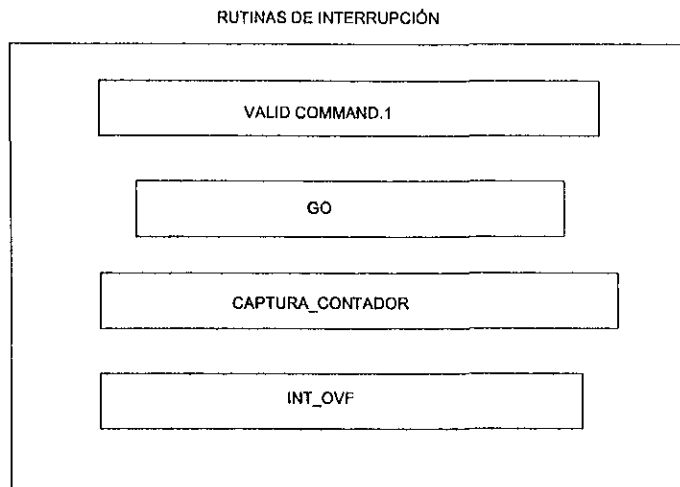


Figura 3-25. Rutinas de interrupción del programa Rot_Elev.

Cabe mencionar que al contar con dos terminales de interrupción externa y una terminal de Contador de captura, ya no es necesario utilizar el sistema externo de interrupción utilizado en el circuito de control de rotación, debido a que las señales de petición de interrupción VALID COMMAND.1, GO y AZS, se conectan directamente a las terminales del MCU, ejecutándose una rutina para cada una de éstas señales. La ventaja que se tuvo al contar con estas tres terminales es eliminar CI's en la tarjeta electrónica y señales de E/S.

La última rutina de interrupción es INT_OVF, se ejecuta a cada sobreflujo del contador 0, su función es generar los pulsos de excitación del motor rotación y elevación cada 10 ms. Al inicio de esta rutina se inicializa el registro del contador 0 al valor de 63Hex, para generar inmediatamente el siguiente retardo. Posteriormente se lee el estado del bit1 de SOLIPULSR para generar un pulso de excitación para el motor de rotación si es solicitado, una vez que se detecta este bit, se lee del registro SOLIPULSE el estado del bit1 para generar el pulso de excitación del motor de elevación siempre y cuando haya sido solicitado.

3.5. Construcción de tarjetas electrónicas

Se construyeron dos tarjetas electrónicas para implementar el sistema de orientación modificado. La primera tarjeta está compuesta por los elementos de la sección A mostrados en el diagrama electrónico de la figura 3-16. La segunda tarjeta está compuesta por los elementos de la sección B mostrados en el diagrama electrónico de la figura 3-17.

Los pasos efectuados en la construcción y ensamble de las tarjetas electrónicas son los siguientes:

- 1) Desarrollo de los diagramas electrónicos. En los diagramas electrónicos se debe especificar el consumo de corriente y los voltajes de alimentación de cada tarjeta, indicar las señales de E/S para cada conector, y el tipo de dispositivos electrónicos. El consumo de corriente de la tarjeta sección A es de 50 mA con un voltaje de alimentación de 5 volts. El consumo de corriente de la tarjeta sección B es de 0.70 A con voltajes de alimentación de 12 y 24 volts. La tarjeta de la sección Control de Rotación (figura 3-1) será sustituida por la tarjeta de la sección A, por consiguiente la tarjeta de la sección Control de Elevación y Excitación de motores (figura 3-2) será sustituida por la tarjeta de la sección B, por lo que las señales de E/S en los conectores J1 y J2 de cada tarjeta se modificaron para lograr una sustitución directa de las nuevas tarjetas, sin modificar las señales de los conectores que se encuentran fijos en el *cartucho electrónico de la herramienta SONIMP II*. En la figura 3-26, se muestra la distribución de las señales de E/S de los conectores fijos J1 y J2 de la sección Control de Rotación y la sección A, y en la figura 3-27 se muestran los conectores J1 y J2 de la sección Control de Elevación y Excitación de motores y de la sección B.
- 2) Distribución electrónica de las tarjetas. La distribución de las tarjetas sección A y sección B fueron desarrolladas por personal experto del IMP dedicado a la distribución de tarjetas y la fabricación de circuitos impresos. Para el desarrollo de la distribución, fue necesario entregar los diagramas electrónicos de ambas tarjetas, la posición de cada conector, las dimensiones de las tarjetas y la posición de los barrenos de los tornillos en las tarjetas, en la siguiente figura 3-28 se muestra las dimensiones de las tarjetas y la posición de los conectores y de los tornillos.
- 3) Fabricación de circuitos impresos. Una vez supervisado y corregida la distribución electrónica de cada tarjeta, se solicitó hacer la fabricación de los circuitos impresos a una empresa externa al IMP especializada en la fabricación de éstos. El tiempo de fabricación invertido por tarjeta fue de 2 a 3 semanas.

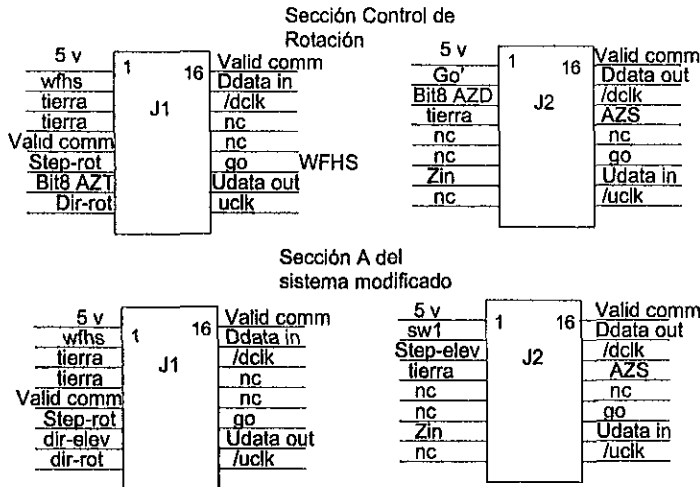


Figura 3-26. Conectores de la tarjeta Control de Rotación y sección A del sistema de orientación modificado.

- 4) Soldado y armado de tarjetas. Antes del soldado de los componentes electrónicos, se realizó una inspección de las pistas en los circuitos impresos, de tal forma que no hicieran contacto entre ellas. Se soldaron las bases de tipo industrial para cada CI, las resistencias, los capacitores, y el MCU de montaje superficial. Para el soldado del MCU se empleó una soldadura y pasta especial para circuitos de montaje superficial, cuya soldadura al hacer contacto con la pasta de soldar se funde a muy bajas temperaturas, por lo que se debe tener cuidado en limpiar perfectamente dicha pasta para evitar que se desprenda de la tarjeta.
- 5) Finalmente se colocan los CI's en las tarjetas y se limpian con alcohol isopropílico para quitar la grasa de la pasta de las soldaduras.

En la figura 3-29, se muestran las tarjetas electrónicas construidas del sistema de orientación modificado del cartucho electrónico. Es importante mencionar que el sistema de orientación modificado puede implementarse en una sola tarjeta, y que el desarrollo de su distribución electrónica está realizada y lista para su fabricación en circuito impreso, sin embargo no se construyó debido a que el empleo de dos tarjetas y la sustitución directa de ambas, es una solución más práctica y sencilla, evitando fallas al modificar el cableado en los conectores internos del cartucho.

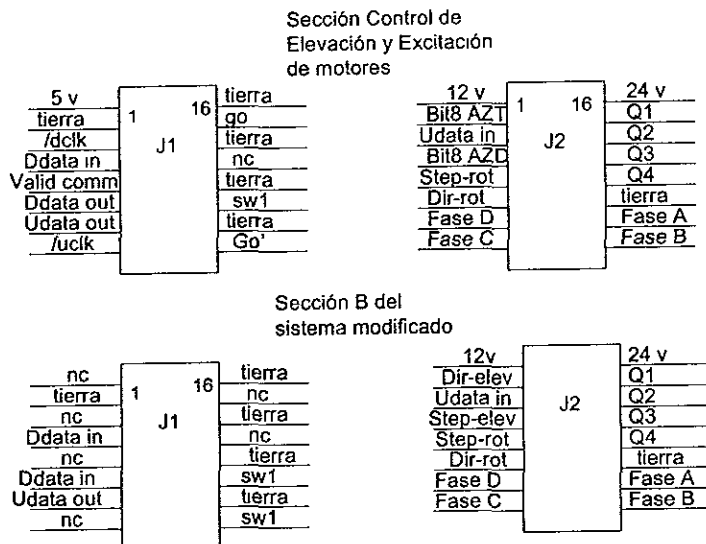


Figura 3-27 Conectores de la tarjeta Control de Elevación y Excitación de motores y sección A del sistema de orientación modificado.

3.6 Localización de fallas

Los puntos de prueba de las tarjetas electrónicas de la sección A y sección B del sistema de orientación, se muestran en los diagramas electrónicos de las figuras 3-16 y 3-17. Los puntos de prueba nos sirven para la detección de fallas en el sistema o bien para inspeccionar de una forma rápida y sencilla el funcionamiento correcto del sistema. A continuación se identifican las señales que deben observarse en cada punto de prueba de ambas tarjetas.

En PP1, se detecta el voltaje de alimentación de la tarjeta A $V_{CC} = 5\text{ V}$.

En PP2, se detecta el nivel de voltaje del RESET en la terminal 10 del MCU, $V_{CC} > 2.7\text{ V}$.

En PP3, se observa la señal STEP_ELEV, una señal cuadrada a una frecuencia de 100 Hz.

En PP4, se observa la señal STEP_ROT, una señal cuadrada a una frecuencia de 100 Hz.

En PP5, se observa el pulso OE en el registro de corrimiento Elev (S/P).

En PP6, se observa el pulso OE' en el registro de corrimiento Rot (S/P).

En PP7, se observa el pulso GO en el registro de corrimiento Elev (P/S).

En PP8, se observa el pulso GO' en el registro de corrimiento Rot (P/S).

En PP9, se observa la señal cuadrada STEP_ELEV en el circuito SAA1027.

En PP10, se observa la señal cuadrada STEP_ROT en el circuito SAA1027.

En PP11, se observa el voltaje de polarización de la tarjeta B, $V_{ee} = 12 \text{ V}$.

En PP12, se observa el voltaje de alimentación de los motores, $V_m = 24 \text{ V}$.

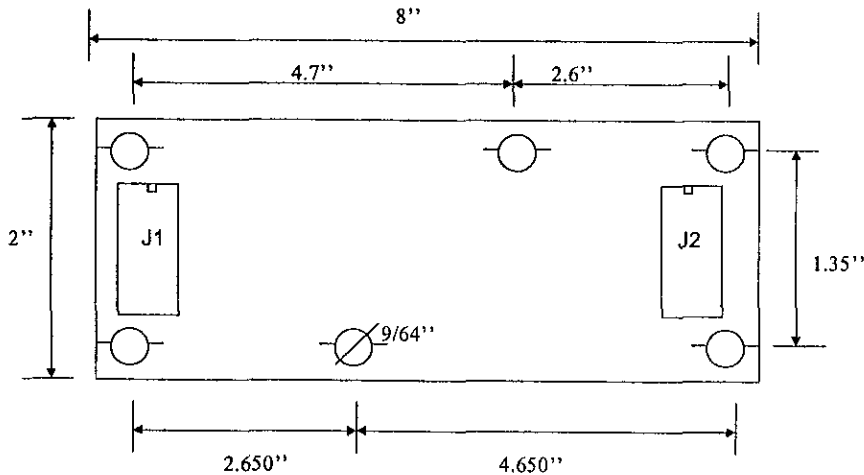


Figura 3-28. Dimensiones de las tarjetas en pulgadas de la sección A y sección B.

CAPITULO III

DESARROLLO DEL SISTEMA DE ORIENTACIÓN

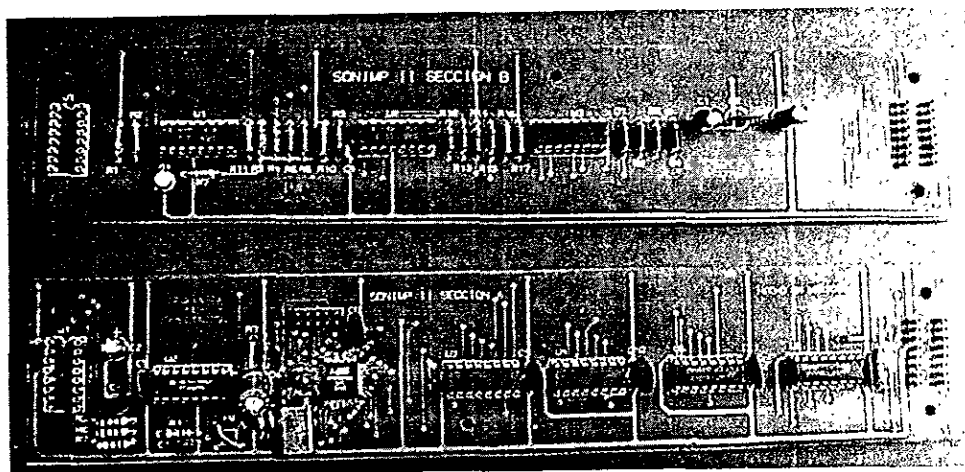


Figura 3-29. Tarjetas electrónicas del sistema de orientación modificado del cartucho electrónico

CAPITULO IV

PRUEBAS DE LABORATORIO Y RESULTADOS

En este capítulo se explica el desarrollo de las pruebas preliminares y finales del sistema de orientación y la descripción de las herramientas empleadas en la realización de las pruebas, así como los resultados obtenidos en dichas pruebas.

4.1 Pruebas preliminares del sistema de orientación modificado

Las primeras pruebas del sistema de orientación modificado se hicieron mediante una PC (computadora personal), los circuitos digitales del sistema implementados en tarjetas prototipo, el kit AVR y el simulador de la sonda mecánica. En la figura 4-1 se muestran los elementos empleados y las conexiones entre ellos en la realización de las primeras pruebas del sistema.

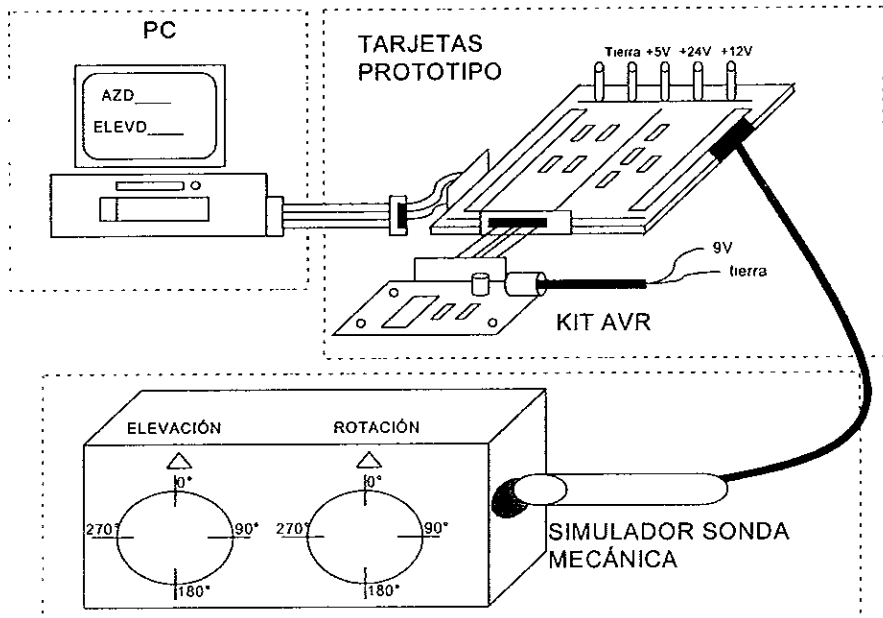


Figura 4-1. Elementos utilizados en las pruebas preliminares.

La PC se utilizó para el desarrollo de un programa de simulación en lenguaje C (ver apéndice B), el programa genera las señales de entrada del sistema de orientación y lee los datos de salida ELEV y AZT a través del puerto paralelo. Las señales de entrada y la configuración de las terminales físicas del puerto paralelo se muestran en la figura 4-2.

El programa despliega un menú con tres opciones de operación:

- Envío de dato AZD
- Envío del dato ELEVD
- Salida del programa

Envío de datos para introducir el dato AZD y el dato ELEVD desde el teclado, y salida del programa. El envío de datos DDATA corresponde a una cadena de 16 bits que contiene el dato de AZD y ELEVD, dichos datos se envían serialmente sincronizados con la señal de reloj DCLK, los primeros 9 bits corresponden al valor del azimuth y los siete bits restantes corresponden al dato de elevación, dicha cadena de bits es leída por los registros de corrimiento S/P. Los datos de ELEV y AZT generados por el MCU pasan por los registros de corrimiento P/S para ser leídos a través de la terminal 13 del puerto paralelo y son desplegados en la pantalla de la PC, para saber si el dato generado es correcto. Las señales de entrada WFHS, DDATA, DCLK, UCLK, VALID COMMAND, GO, son generadas al mismo tiempo por el puerto paralelo, y la señal de salida del sistema UDATA siempre es leída a través del puerto.

El programa no genera las señales de entrada Zin, SW1 y AZS requeridas por el sistema de orientación, las señales de referencia cero provienen de un *dip switch*, y son controladas manualmente. Para generar la señal AZS utilizamos un generador de señales que nos permitió variar el número de pulsos cada 40 ms.

El siguiente bloque de la figura 4-1, está formado por el sistema de orientación implementado en tarjetas prototipo y por el kit AVR. En las tarjetas prototipos se colocaron los registros de corrimiento y los circuitos de excitación de los motores.

El kit AVR se utilizó para programar, leer y verificar la memoria Flash del MCU, y también para iniciar la operación del programa Rot_Elev, obteniendo las señales de entrada y salida a través de los puertos.

El kit AVR, es una tarjeta electrónica de desarrollo, diseñada especialmente para introducir a los usuarios de una forma rápida y sencilla al diseño de aplicaciones con el MCU de la familia AVR, implementando diseños de programas pequeños para la obtención rápida de información o señales de salida deseados, logrando que el usuario se familiarice con el equipo. El kit AVR cuenta con los siguientes componentes, una fuente de alimentación regulada, 8 microinterruptores (*push-buttons*) de uso general, 8 led's indicadores de uso general, los puertos del MCU

pueden accesarse mediante conectores tipo *header*, un convertidor de nivel RS-232 de uso general y programación serial para familias AT90S/AT89S AVR's.

El kit AVR cuenta con tres programas de desarrollo, AVR Studio, AVR Prog y AVR assembler. El programa AVR Studio permite al usuario ejecutar los programas, simulando el conjunto de instrucciones de un programa en un archivo objeto generado por el AVR ensamblador o por el compilador ICCA90 para MCU AVR's, o bien, la ejecución de programas en circuito emulador AVR.

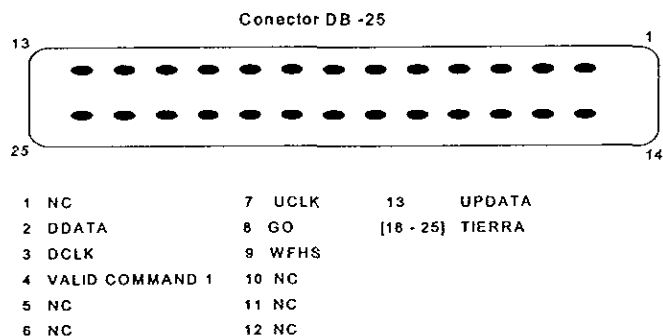


Figura 4-2 (a) Configuración de las terminales físicas del conector del puerto paralelo.

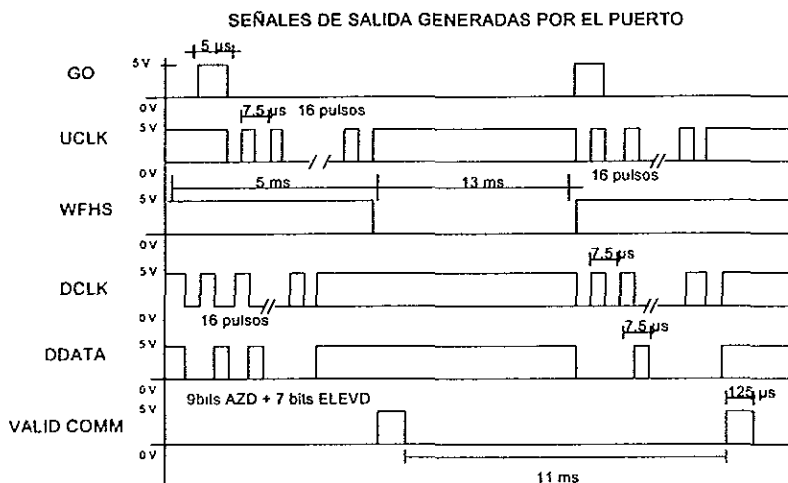


Figura 4-2 (b). Señales de salida generadas por el puerto paralelo

El programa AVR Prog es empleado para programar, leer, verificar la memoria Flash y EEPROM. El programa AVR assembler traduce el código de instrucciones en lenguaje ensamblador a código objeto de la familia AT90S MCU's. Los tres programas pueden instalarse en Windows 3.11, 95 ó NT.

El último bloque está formado por el simulador Sonda Mecánica, el simulador está formado por el sistema de engranes que utiliza el cartucho de la sonda mecánica de la herramienta y por los motores de elevación y rotación. En la parte externa tiene dos discos graduados para observar el desplazamiento de cada motor. Las señales de entrada requeridas por el simulador se conectan mediante un conector, en la figura 4-3 se muestran las terminales del conector así como las señales de entrada a éste. Las señales de entrada al simulador son las fases Q1, Q2, Q3, Q4 del motor de elevación, las fases A, B, C, D del motor de rotación y el voltaje de polarización de 24 volts.

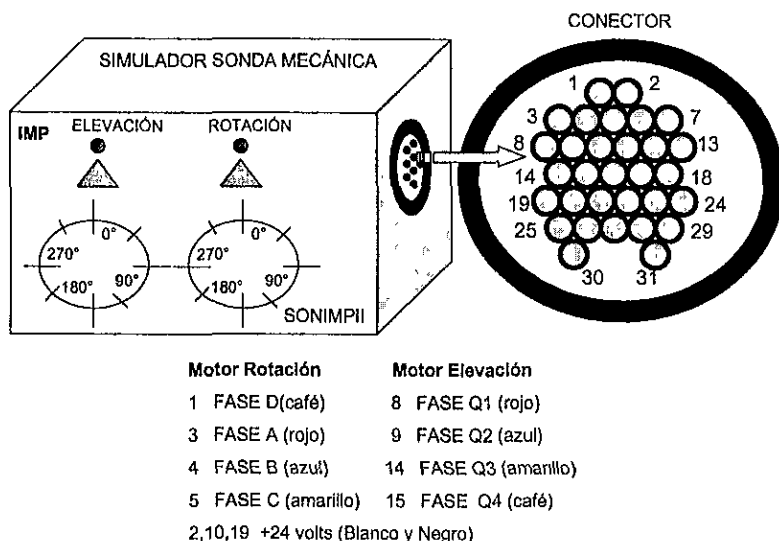


Figura 4-3. Señales de entrada al conector del simulador Sonda Mecánica.

Las primeras pruebas se efectuaron de la siguiente forma:

- Ejecución de la rutina de inicialización Cero_elev. Se enciende el sistema de orientación y después de un retardo inicial de 16 ms, se ejecuta la rutina Cero_elev, la cuál genera trenes de pulsos en sentido negativo, simulando desplazar el motor de elevación hacia cero grados de inclinación, observando en el disco de la sonda mecánica un desplazamiento de 45 grados en sentido de las manecillas del reloj sin el envío de la señal SW1 para simular que el

transductor vertical no llega a hacer contacto con el sensor de posición. Se inicia de nuevo la ejecución de la rutina, pero en este caso la señal SW1 se activa simulando que el transductor ha llegado a la posición de cero grados.

- b) Ejecución de la rutina de inicialización Cero_rot. Una vez que el disco del motor de elevación se detiene, se envían trenes de pulsos en sentido positivo al motor de rotación hasta detectar la señal del sensor de posición Zin. El disco del motor de rotación se desplaza hacia el sentido contrario a las manecillas del reloj hasta activarse la señal Zin a través del dip-switch. Simulando encontrar la marca de cero de referencia en el disco graduado.
- c) Envío del dato AZS. Inicialmente probamos con varios valores de AZS manteniendo fijo dicho valor, observando en la perilla del motor de rotación desplazarse en un sentido o en otro hasta colocarse en el cero magnético terrestre y observando en la pantalla de la PC la variación del valor AZT, es decir, si $AZS = 350^\circ$ la perilla gira 10° grados en sentido en contra de las manecillas del reloj hasta colocarse en 0° y en la pantalla el dato AZT se observa que va desde 350 hasta 0 grados, o bien, si el dato $AZS = 45^\circ$, entonces la perilla gira 45 grados en sentido a las manecillas del reloj hasta colocarse en la posición cero magnético, y por consiguiente se observa en la pantalla de la PC el dato AZT que varía desde 45 hasta 0 grados. Posteriormente del generador de señales variamos el número de pulsos, simulando el comportamiento real de la herramienta, es decir, el valor proveniente de la brújula o giroscopio AZS no siempre se mantiene en un valor estable debido a que la herramienta se encuentra en caída libre.
- d) Envío del dato AZD. En esta prueba se observa el funcionamiento del motor de rotación. Introducimos el dato AZD desde el teclado con incrementos de 10, es decir, 10, 20, 30, 40 hasta 360° . Después se probaron varios recorridos en forma aleatoria, desde el mínimo desplazamiento 0 a 1° hasta barridos desde 0 hasta 360° , observando en el disco del motor el sentido de giro correcto y la trayectoria más corta en el desplazamiento angular del motor. También se probó el estado de la señal Zin durante el movimiento del motor, es decir, si por alguna causa el MCU llegará a perder la cuenta correcta del valor AZT durante el desplazamiento deseado del motor, al pasar por la marca Zin, el valor AZT se inicializa con 0° y comienza a efectuarse el desplazamiento y la cuenta correcta del dato AZT.
- e) Envío del dato ELEVD. Introducimos el dato ELEVD desde el teclado, variando su valor en incrementos de 10, desde 0 hasta 90° , observando el desplazamiento angular en el disco del motor de elevación. Así mismo, realizamos barridos en forma aleatoria, introduciendo el valor ELEVD desde el mínimo desplazamiento de 0 a 1° , hasta barridos desde 0 hasta 90° . Los desplazamientos angulares del motor de elevación se observaron en el disco

graduado al igual que los sentidos correctos para cada valor de ELEVD. Si durante el movimiento del motor de elevación la señal SW1 se llegará a activar, el valor de ELEV se inicializa con 0° y el MCU comenzará de nuevo la cuenta correspondiente al valor ELEVD.

- f) Envío de los datos AZD y ELEVD. Se introducen los valores AZD y ELEVD desde el teclado para efectuar barridos en forma aleatoria, para observar el movimiento de ambos motores simultáneamente.

Los resultados esperados fueron satisfactorios, el sentido de giro, el seguimiento de la trayectoria más corta, el movimiento de los motores simultáneamente, se realizaron correctamente; sin embargo, el realizar este tipo de pruebas antes de implementar el sistema en tarjetas en circuito impreso, nos permitió detectar varias fallas en el sistema, permitiendo la depuración del programa de control Rot_Elev.

Una de las fallas detectadas durante la realización de las pruebas fue que el valor del dato AZT, desplegaba valores fuera del rango establecido, es decir $AZT > 359^\circ$, al principio se ajustó el valor de AZS en el rango de 0 hasta 359° en la rutina de Entrada de captura (ICP), sin embargo el MCU seguía generando el mismo problema y se creía que lo estaba provocando alguna instrucción de condición al efectuar el cálculo del valor de AZT pero no fue así. El problema ocurría cuando se ejecutaba la suma $AZT = RBE + AZS$ y se solicitaba el envío de los datos a la entrada del pulso GO, en ese instante la variable AZT almacenada en SRAM contiene el valor de la suma, y es el dato que es enviado. El problema se resolvió almacenando el valor de AZT en otra variable AZTREAL al final de la rutina de AZMTRANS, para enviar el valor calculado anteriormente, sin alterar el cálculo del valor de AZT que actualmente se esté ejecutando.

4.2 Pruebas finales del sistema de orientación modificado

Las pruebas finales del sistema de orientación se realizaron con las tarjetas electrónicas de la sección A y sección B del sistema de orientación modificado.

Inicialmente se probaron las tarjetas con las señales de E/S generadas por el programa en lenguaje C a través del puerto paralelo, y conectando las señales de salida al simulador Sonda Mecánica, para efectuar la misma secuencia de pasos de las pruebas realizadas en la sección 4.1.

Una de las fallas detectada fue que durante el envío del dato AZT, el programa en C no leía el valor del dato generado por el MCU, sin embargo, el motor de elevación si efectuaba el movimiento correcto, por lo que el problema era de una conexión física errónea o faltante en la tarjeta de la sección A, se encontró que los registros de corrimiento P/S no estaban conectados en cascada, y se realizó la conexión.

Una vez que se probó el sistema de orientación con las tarjetas electrónicas y se obtuvieron el movimiento de los motores, el sentido de giro y la ejecución de las rutinas de inicialización correctamente, se procedió a instalar las tarjetas electrónicas en la herramienta SONIMPII.

La herramienta SONIMPII, se conectó al simulador de Telemetría SONIMP y a la unidad móvil BLUE para introducir los datos de ELEVD y AZD, así como también el simulador Sonda Mecánica para observar el desplazamiento de los motores, las conexiones físicas y los elementos empleados para las pruebas finales del sistema de orientación modificado se muestran en la siguiente figura 4-4.

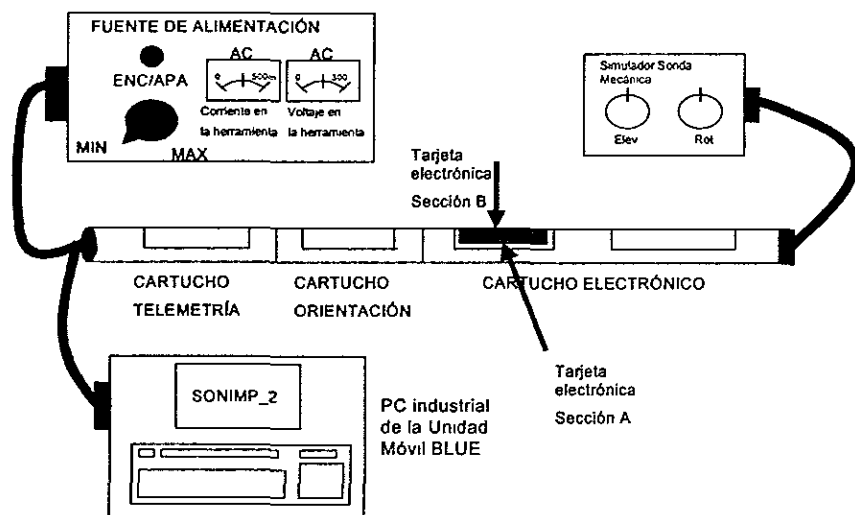


Figura 4-4. Elementos utilizados en el desarrollo de las pruebas finales

Una vez efectuada dicha conexión se instalaron las tarjetas electrónicas en el cartucho electrónico de la herramienta SONIMPII y se encendió la herramienta variando la perilla del simulador de Telemetría SONIMP hasta observar que la aguja del galvanómetro marcara 250 voltios de a.c., supervisando que el consumo de corriente de la herramienta no exceda 300 mA.

En algunas ocasiones el sistema de orientación ejecutaba las rutinas Cero_elev y Cero_rot, pero en el momento de activar la señal de Zin, el MCU generaba valores de AZT aleatorios mayores a 359°, generando el error que se presentaba en el sistema instalado en las tarjetas prototipo. El problema se resolvió inicializando el registro de datos del contador de 16 bits al inicio de la rutina Cero_rot.

Sin embargo, se detectó una falla más crítica en el sistema, algunas veces el MCU operaba sin ningún problema y otras veces no operaba. Inicialmente se pensó que el problema era ocasionado por soldaduras frías en las tarjetas, por lo que se soldaron una vez más, terminal por terminal de cada dispositivo y el resultado fue el mismo, seguía sin operar el sistema. Posteriormente se creyó que el problema era ocasionado por la señal de entrada en la terminal RESET del MCU, por lo que se efectuaron varios RESET en forma manual, y el MCU seguía sin operar.

Por último se supuso que el problema se debía a la variación del voltaje V_{CC} de alimentación al MCU, debido a que el suministro de voltaje aumenta de forma gradual hasta llegar a 250 volts de a.c., durante esta variación de voltaje, los voltajes que alimentan a las tarjetas del cartucho electrónico no llegan de forma instantánea, tardan pocos segundos para lograr establecer su valor. Debido a esto, se desmontaron las tarjetas de la herramienta para probar el suministro de voltaje variando el rango de 0 hasta 5 volts con una fuente de voltaje, y una vez más no funcionó. Se probaron las tarjetas con una fuente de voltaje constante de 5 volts, y el sistema funcionó sin ningún problema. Por lo que se desarrolló un circuito de preencendido, tomando en cuenta los siguientes parámetros en su diseño:

$$V_{on} > 2.7 \text{ V y } I_{out} \geq 10 \text{ mA.}$$

V_{on} , voltaje mínimo de polarización del MCU.

I_{out} , corriente que consume la tarjeta sección A.

Para lograr que la fuente de voltaje de 5 volts se suministrara en forma casi instantánea, se empleo un comparador de voltaje, cuyo voltaje de encendido fuera mayor a 2.7 volts y suministrara una demanda de corriente mayor o igual a 10 mA.

El comparador de voltaje que seleccionamos en base a los parámetros de diseño es el LM111JG de Texas Instruments, cuyo valor mínimo de voltaje para operar es de 3.5 volts y una corriente máxima de 50 mA, datos obtenidos del fabricante. En la figura 4-5 se observa la configuración empleada del circuito de encendido.

En el circuito de encendido se realizó el cálculo de la resistencia de colector empleando la formula (3.15) de la tabla 3-10.

$$R_c = \frac{V_{CC}}{I_{Csat}} = \frac{5\text{volts}}{10\text{mA}} = 500\Omega \quad (4.1)$$

El circuito de encendido, se probó en una tarjeta prototipo, debido a que la tarjeta de la sección A ya se había construido.

Una vez efectuadas estas adecuaciones se montaron las tarjetas en el cartucho electrónico de la herramienta y se efectuaron las mismas pruebas realizadas en las pruebas iniciales, a diferencia que las señales de entrada son generadas por el simulador de Telemetría y los datos de ELEVD, AZD y AZS, fueron introducidos por la computadora que se encuentra montada en la unidad móvil BLUE del sistema SONIMPII. En la figura 4-6, se muestran las tarjetas electrónicas montadas en la herramienta.

Los resultados de las pruebas finales fueron los esperados, a cada orden enviada desde la PC se obtuvo el desplazamiento angular y el sentido de giro deseado, la ejecución de las rutinas de inicialización, el seguimiento de la trayectoria más corta, el movimiento de ambos motores simultáneamente, la posición de cada motor y el sentido de giro esperados, se ejecutaron correctamente.

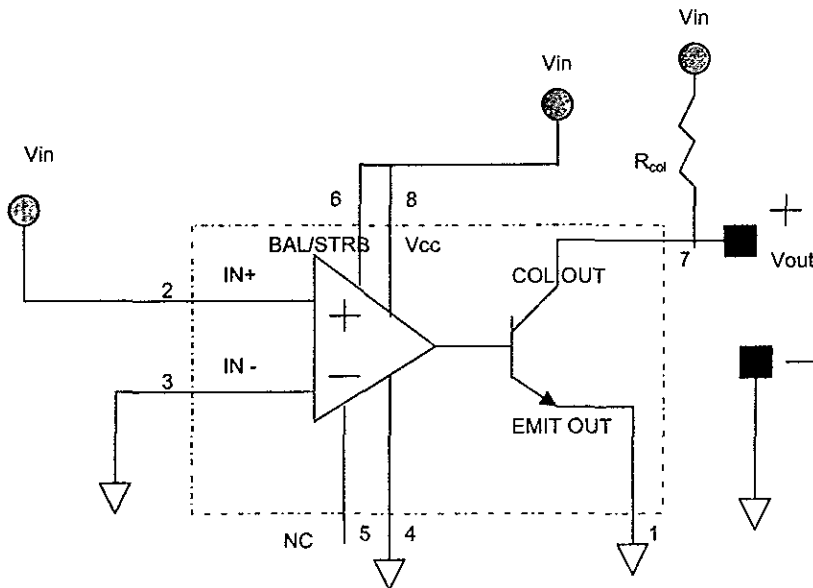
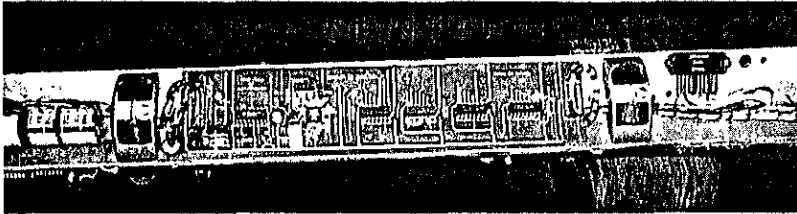


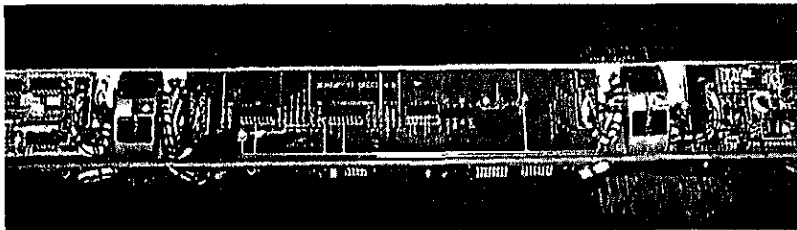
Figura 4-5. Circuito de preencendido de la tarjeta sección A del sistema de orientación.

CAPITULO IV

DESARROLLO DE PRUEBAS Y RESULTADOS



(a)



(b)

Figura 4-6. Sistema de orientación modificado montado en el cartucho electrónico de la herramienta SONIMP II. (a) Tarjeta Sección A. (b) Tarjeta Sección B.

CONCLUSIONES

Durante el desarrollo de este trabajo se modificó el sistema de orientación del cartucho electrónico de la herramienta SONIMP II, teniendo un control de las funciones de rotación y elevación de los sensores ultrasónicos de dicha herramienta. Los resultados obtenidos durante las pruebas del sistema fueron satisfactorios, es decir, se tiene un control en los movimientos de rotación y elevación de los transductores ultrasónicos con una resolución mínima de 1°.

Las ventajas principales presentadas por el sistema de orientación modificado con respecto al sistema original son:

- Software totalmente integrado en un solo MCU
- Menor número de componentes electrónicos
- Menor consumo de corriente
- Mayor facilidad en detección de fallas

Es importante mencionar que el sistema de orientación puede reducirse tanto en componentes electrónicos como en las dimensiones de las tarjetas, si es que en un futuro se requiere disminuir el tamaño del cartucho electrónico. El código utilizado en el desarrollo del programa de control es menor al código utilizado por el sistema original y utiliza instrucciones de MCU's modernos, por lo que puede reemplazarse fácilmente por otro MCU del mercado actual, invirtiendo poco tiempo en el desarrollo del código, suponiendo que en el peor de los casos dicho MCU ya no saliera al mercado.

Con el diseño, desarrollo y construcción de este sistema se pudieron alcanzar dos objetivos primordiales, el primer objetivo fue garantizar el funcionamiento óptimo del sistema de orientación del cartucho electrónico utilizando tecnología de vanguardia, ya que de no funcionar la herramienta, PEMEX tendría que contratar el servicio de los registros sonares a empresas extranjeras, ocasionándole un costo mayor comparado con el que se tiene al emplear dicha herramienta. El segundo objetivo fue la experiencia que obtuve en la realización de un programa de control empleando un MCU moderno y en la construcción física de un sistema en el campo de la ingeniería electrónica que se encuentra funcionando en la industria petrolera.

BIBLIOGRAFÍA

Páginas del Web

<http://www.motorola.com>
<http://www.microchip.com>
<http://www.intel.com>
<http://www.atmel.com>
<http://www.honeywall.com>

Diseño de sistemas digitales y microprocesadores
John P. Hayes
Ed. Mc Graww Hil.

Circuitos Electrónicos, discretos e integrados
Donald L. Schilling
Charles Belove.
2 edición
1991.

AVR Enhanced RISC Microcontroller
Atmel Corporation.
Data book
1997.

Flash Microcontroller Development tools
User's guide
1997, Atmel Corporation.

Manual Stepping Motors
Handbook
Vexta

Microprocesadores y microcomputadoras
Serie: Mundo electrónico
Autores: Antonio Alabau Muñoz, Francisco Angeles Corts, José A. Escamilla Ballester.
3ª edición, 1980.

Manual de Ingeniería Electrónica
Donald G. Fink
Donald Christiansen.
Vol II, materiales, dispositivos, componentes y montajes.
Mc Graw Hill. 1992

Sistemas de Medición
Principios y aplicaciones
John P. Bentley
2ª Edición.

APÉNDICE A

En este apéndice se muestran las especificaciones técnicas de la herramienta y del MCU AT90S8515, también la lista de los componentes electrónicos empleados en el desarrollo del sistema de orientación modificado.

A.1 Especificaciones técnicas de la herramienta

TEMPERATURA MÁXIMA DEL MEDIO	70 GRADOS C
PRESION EXTERNA MÁXIMA	3000 PSI
PROFUNDIDAD MÁXIMA DE OPERACIÓN	2000 METROS
ALCANCES	RADIOS DE 70 METROS SIN TP Y EN SENO DE SALMUERA
	RADIOS DE 50 METROS CON TP Y EN SENO DE SALMUERA
RESOLUCIÓN DEL MOV. DE ROTACIÓN	± 1 GRADO
RESOLUCIÓN DEL MOV. DE ELEVACIÓN	± 1 GRADO
DIÁMETRO MÁXIMO DE LA HERRAMIENTA	6"
LONGITUD TOTAL	7.8 METROS
PESO	150kg.
TIPO DE FLUIDO	SALMUERA
TIPO DE TRANSDUCTOR	PIEZOELECTRICO
FRECUENCIA DEL TRANSDUCTOR	SONDA MECÁNICA I, ,VARIABLE DE 150 A 250 kHz. SONDA MECÁNICA II 340 kHz.
SISTEMA DE TELEMETRÍA	C T S
COMPATIBILIDAD	IBM-AT
PROGRAMA DE CONTROL E INTERPRETACIÓN	SONIMP_2
ALIMENTACIÓN REQUERIDA	250 VCA ±10%
FRECUENCIA DE ALIMENTACIÓN	50-63 Hz
CONSUMO DE CORRIENTE	250 a 340 mA

**A.2 Lista de componentes de la tarjeta electrónica Sección A
del sistema de orientación del cartucho electrónico**

Resistencias	Valor (Ω)	Descripción
R1, R2, R3	100K	Carbón
Capacitores	Valor (f)	Descripción
C1	1.5 n	Cerámico
C2, C3, C5, C8, C9, C10, C11, C12	0.1 μ	Cerámico
C4	10 μ / 25V	Electrolítico
C6, C7	22 p	Tantalio
C13	150 μ / 25V	Electrolítico
Diodos	Valor	Descripción
D1	1N914	VPI=75 V
Cristal		
4 MHz	K11SA	
Circuitos integrados		
U1	AT90S8515	Microcontrolador Atmel
U2	MC4538	Monoestable
U3	MC4094	Registro de corrimiento S/P
U4	MC4094	Registro de corrimiento S/P
U5	MC4021	Registro de corrimiento P/S
U6	MC4021	Registro de corrimiento P/S

**A.3 Lista de componentes de la tarjeta electrónica Sección B
del sistema de orientación del cartucho electrónico**

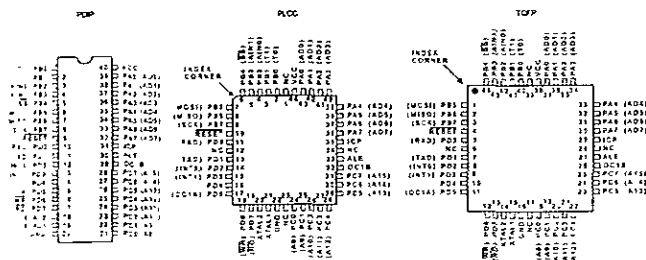
Resistencias	Valor (Ω)	Descripción
R1, R9	100	Carbón
R2, R10	1 K	Carbón
R3, R4, R5, R6	22.1 K	Precisión
R7, R8	10 K	Carbón
R11	4.99 K	Precisión
R12, R13, R14, R15	22.1 K	Precisión
R16, R17	4.99 K	Precisión
R18, R19, R20	10 K	Carbón
Capacitores	Valor (f)	Descripción
C1	3.3 μ / 25 V	Electrolítico
C2, C4, C5	0.1 μ	Cerámico
C3	10 μ / 25 V	Electrolítico
Diodos	Valor	Descripción
D1	IN4936	Ultra-rápida conmutación, VPI=400V, $I_o=1$ A
D2	IN4936	
D3	IN4936	
D4	IN4936	
Transistor	Tipo	
Q1	2N2222A	Transistor N, B=300
Circuitos integrados		
U1	SAA1027	Excitador de motores
U2	SAA1027	Excitador de motores
U3	ULN2001	Transistores Darlington

A.4 Especificaciones técnicas del MCU AT90S8515.

Features

- Utilizes the AVR[®] RISC Architecture
- AVR - High-performance and Low-power RISC Architecture
 - 118 Powerful Instructions - Most Single Clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Up to 8 MIPS Throughput at 8 MHz
- Data and Nonvolatile Program Memory
 - 4K/8K Bytes of In-System Programmable Flash
 - Endurance: 1,000 Write/Erase Cycles
 - 256/512 Bytes of SRAM
 - 256/512 Bytes of In-System Programmable EEPROM
 - Endurance: 100,000 Write/Erase Cycles
 - Programming Lock for Flash Program and EEPROM Data Security
- Peripheral Features
 - One 8-bit Timer/Counter with Separate Prescaler
 - One 16-bit Timer/Counter with Separate Prescaler
 - Compare, Capture Modes and Dual 8-, 9- or 10-bit PWM
 - On-chip Analog Comparator
 - Programmable Watchdog Timer with On-chip Oscillator
 - Programmable Serial UART
 - Master/Slave SPI Serial Interface
- Special Microcontroller Features
 - Low-power Idle and Power Down Modes
 - External and Internal Interrupt Sources
- Specifications
 - Low-power, High-speed CMOS Process Technology
 - Fully Static Operation
- Power Consumption at 4 MHz, 3V, 25°C
 - Active: 3.0 mA
 - Idle Mode: 1.0 mA
 - Power Down Mode: <1 μ A
- I/O and Packages
 - 32 Programmable I/O Lines
 - 40-pin PDIP, 44-pin PLCC and TQFP
- Operating Voltages
 - 2.7 - 6.0V (AT90S4414-4 and AT90S8515-4)
 - 4.0 - 6.0V (AT90S4414-8 and AT90S8515-8)
- Speed Grades
 - 0 - 4 MHz (AT90S4414-4 and AT90S8515-4)
 - 0 - 8 MHz (AT90S4414-8 and AT90S8515-8)

Pin Configurations



Rev 0841ES-04/99



8-bit AVR[®]
Microcontroller
with 4K/8K
bytes In-System
Programmable
Flash

AT90S4414
AT90S8515



Note: This is a summary document. For the complete 101-page document, please visit our web site at www.atmel.com or e-mail at literature@atmel.com and request literature #0841E.

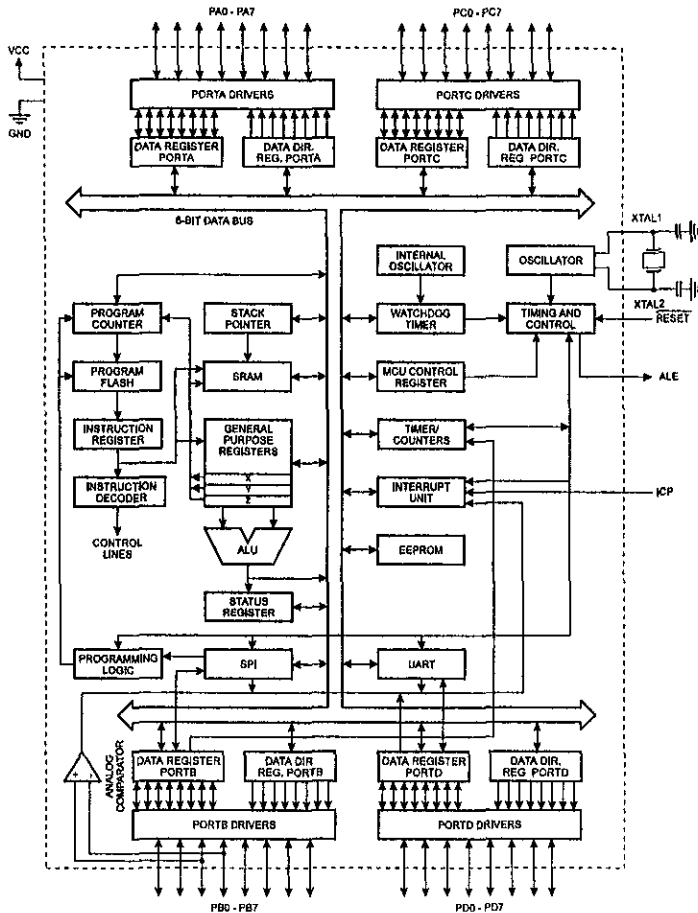


Description

The AT90S4414/AT90S8515 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the AT90S4414/8515 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

Block Diagram

Figure 1. The AT90S4414/8515 Block Diagram



The AVR core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The AT90S4414/8515 provides the following features: 4K/8K bytes of In-System Programmable Flash, 256/512 bytes EEPROM, 256/512 bytes SRAM, 32 general purpose I/O lines, 32 general purpose working registers, flexible timer/counters with compare modes, internal and external interrupts, a programmable serial UART, programmable Watch-dog Timer with internal oscillator, an SPI serial port and two software selectable power saving modes. The Idle Mode stops the CPU while allowing the SRAM, timer/counters, SPI port and interrupt system to continue functioning. The power down mode saves the register contents but freezes the oscillator, disabling all other chip functions until the next external interrupt or hardware reset.

The device is manufactured using Atmel's high density non-volatile memory technology. The on-chip in-system programmable Flash allows the program memory to be reprogrammed in-system through an SPI serial interface or by a conventional nonvolatile memory programmer. By combining an enhanced RISC 8-bit CPU with In-System Programmable Flash on a monolithic chip, the Atmel AT90S4414/8515 is a powerful microcontroller that provides a highly flexible and cost effective solution to many embedded control applications.

The AT90S4414/8515 AVR is supported with a full suite of program and system development tools including: C compilers, macro assemblers, program debugger/simulators, in-circuit emulators, and evaluation kits.

Comparison Between AT90S4414 and AT90S8515

The AT90S4414 has 4K bytes of In-System Programmable Flash, 256 bytes of EEPROM and 256 bytes of internal SRAM.

The AT90S8515 has 8K bytes of In-System Programmable Flash, 512 bytes of EEPROM and 512 bytes of internal SRAM

Table 1 summarizes the different memory sizes for the two devices.

Table 1. Memory Size Summary

Part	Flash	EEPROM	SRAM
AT90S4414	4K bytes	256 bytes	256 bytes
AT90S8515	8K bytes	512 bytes	512 bytes

Pin Descriptions

VCC

Supply voltage

GND

Ground

Port A (PA7..PA0)

Port A is an 8-bit bidirectional I/O port. Port pins can provide internal pull-up resistors (selected for each bit). The Port A output buffers can sink 20mA and can drive LED displays directly. When pins PA0 to PA7 are used as inputs and are externally pulled low, they will source current if the internal pull-up resistors are activated. The Port A pins are tri-stated when a reset condition becomes active, even if the clock is not active.

Port A serves as Multiplexed Address/Data input/output when using external SRAM.

Port B (PB7..PB0)

Port B is an 8-bit bidirectional I/O port with internal pull-up resistors. The Port B output buffers can sink 20 mA. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not active.



Port C (PC7..PC0)

Port C is an 8-bit bidirectional I/O port with internal pull-up resistors. The Port C output buffers can sink 20 mA. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not active. Port C also serves as Address output when using external SRAM.

Port D (PD7..PD0)

Port D is an 8-bit bidirectional I/O port with internal pull-up resistors. The Port D output buffers can sink 20 mA. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not active.

RESET

Reset input. A low level on this pin for more than 50 ns will generate a reset, even if the clock is not running. Shorter pulses are not guaranteed to generate a reset.

XTAL1

Input to the inverting oscillator amplifier and input to the internal clock operating circuit.

XTAL2

Output from the inverting oscillator amplifier

ICP

ICP is the input pin for the Timer/Counter1 Input Capture function.

OC1B

OC1B is the output pin for the Timer/Counter1 Output CompareB function

ALE

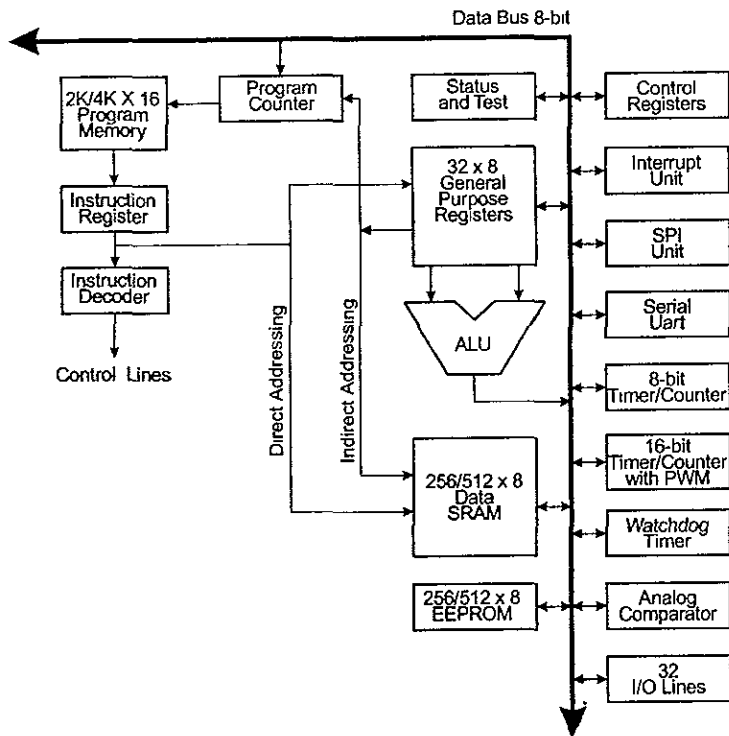
ALE is the Address Latch Enable used when the External Memory is enabled. The ALE strobe is used to latch the low-order address (8 bits) into an address latch during the first access cycle, and the AD0-7 pins are used for data during the second access cycle.

Architectural Overview

The fast-access register file concept contains 32 x 8-bit general purpose working registers with a single clock cycle access time. This means that during one single clock cycle, one ALU (Arithmetic Logic Unit) operation is executed. Two operands are output from the register file, the operation is executed, and the result is stored back in the register file - in one clock cycle.

Six of the 32 registers can be used as three 16-bits indirect address register pointers for Data Space addressing - enabling efficient address calculations. One of the three address pointers is also used as the address pointer for the constant table look up function. These added function registers are the 16-bits X-register, Y-register and Z-register.

The ALU supports arithmetic and logic functions between registers or between a constant and a register. Single register operations are also executed in the ALU. Figure 2 shows the AT90S4414/8515 AVR Enhanced RISC microcontroller architecture

Figure 2. The AT90S4414/8515 AVR Enhanced RISC Architecture


In addition to the register operation, the conventional memory addressing modes can be used on the register file as well. This is enabled by the fact that the register file is assigned the 32 lowest Data Space addresses (\$00 - \$1F), allowing them to be accessed as though they were ordinary memory locations.

The I/O memory space contains 64 addresses for CPU peripheral functions as Control Registers, Timer/Counters, A/D-converters, and other I/O functions. The I/O Memory can be accessed directly, or as the Data Space locations following those of the register file, \$20 - \$5F.

The AVR uses a Harvard architecture concept - with separate memories and buses for program and data. The program memory is executed with a two stage pipeline. While one instruction is being executed, the next instruction is pre-fetched from the program memory. This concept enables instructions to be executed in every clock cycle. The program memory is in-system programmable Flash memory.

With the relative jump and call instructions, the whole 2K/4K address space is directly accessed. Most AVR instructions have a single 16-bit word format. Every program memory address contains a 16- or 32-bit instruction.

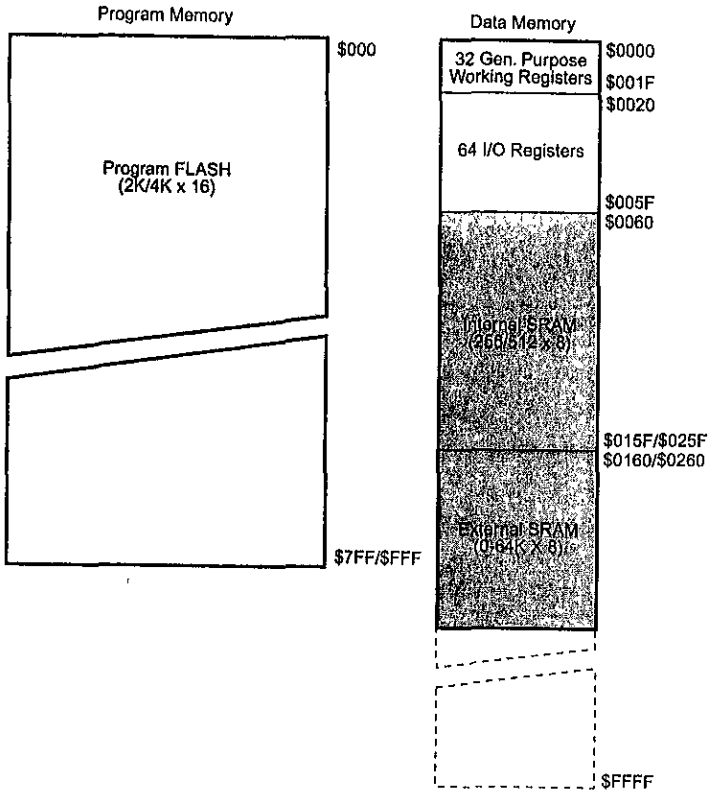
During interrupts and subroutine calls, the return address program counter (PC) is stored on the stack. The stack is effectively allocated in the general data SRAM, and consequently the stack size is only limited by the total SRAM size and the usage of the SRAM. All user programs must initialize the SP in the reset routine (before subroutines or interrupts are executed). The 16-bit stack pointer SP is read/write accessible in the I/O space.

The 256/512 bytes data SRAM can be easily accessed through the five different addressing modes supported in the AVR architecture.

The memory spaces in the AVR architecture are all linear and regular memory maps.

A flexible interrupt module has its control registers in the I/O space with an additional global interrupt enable bit in the status register. All the different interrupts have a separate interrupt vector in the interrupt vector table at the beginning of the program memory. The different interrupts have priority in accordance with their interrupt vector position. The lower the interrupt vector address the higher the priority.

Figure 3. Memory Maps



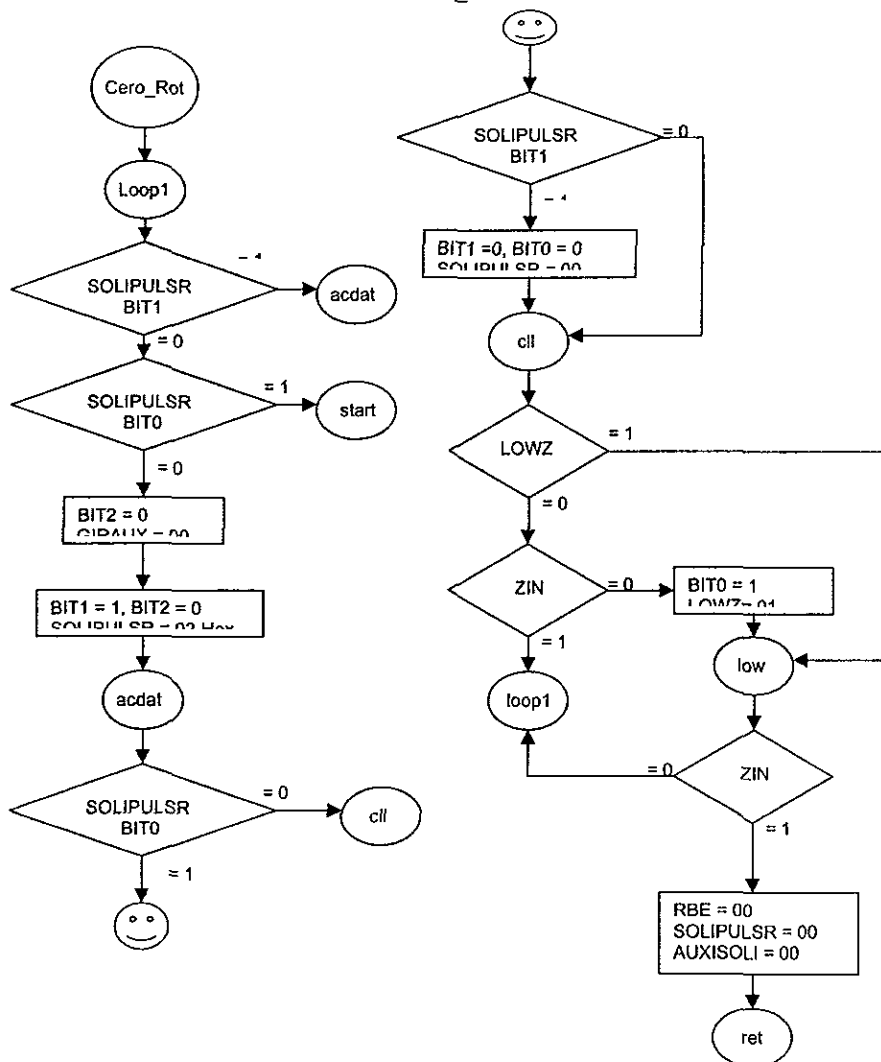
APÉNDICE B

En este apéndice se muestran los diagramas de flujo del programa Rot_Elev, incluyendo las rutinas de interrupción del sistema de orientación modificado.

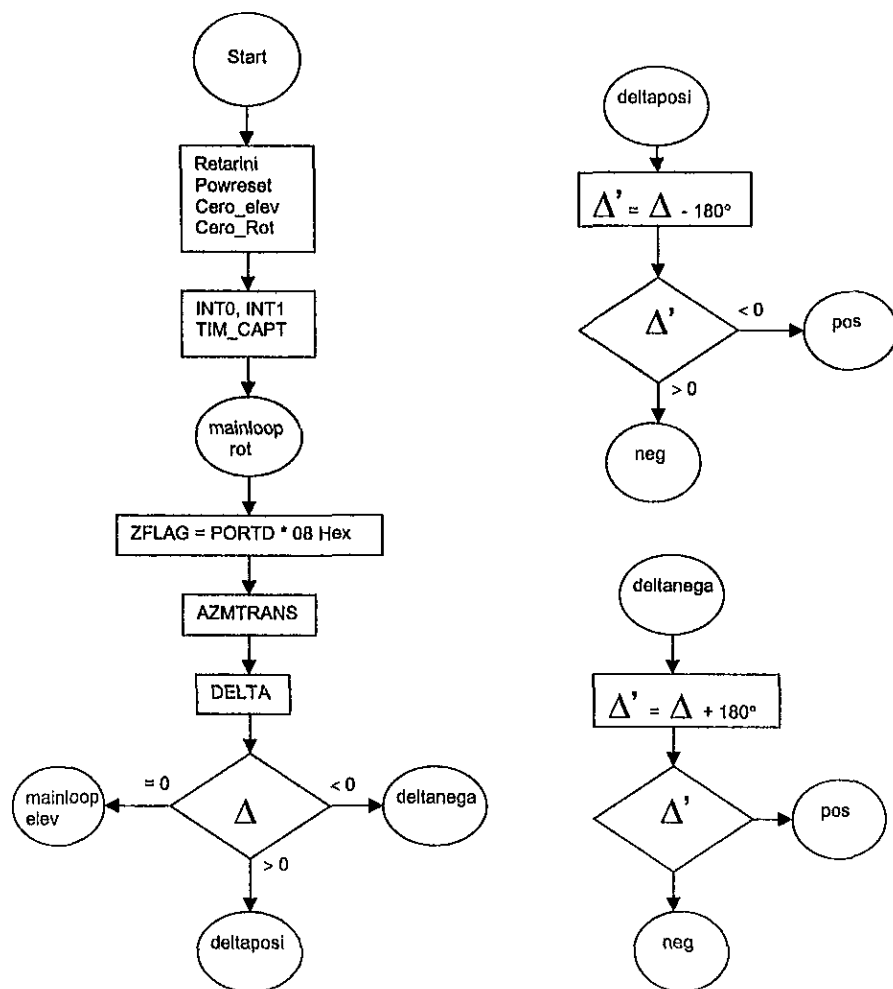
Rutina Cero_Elev



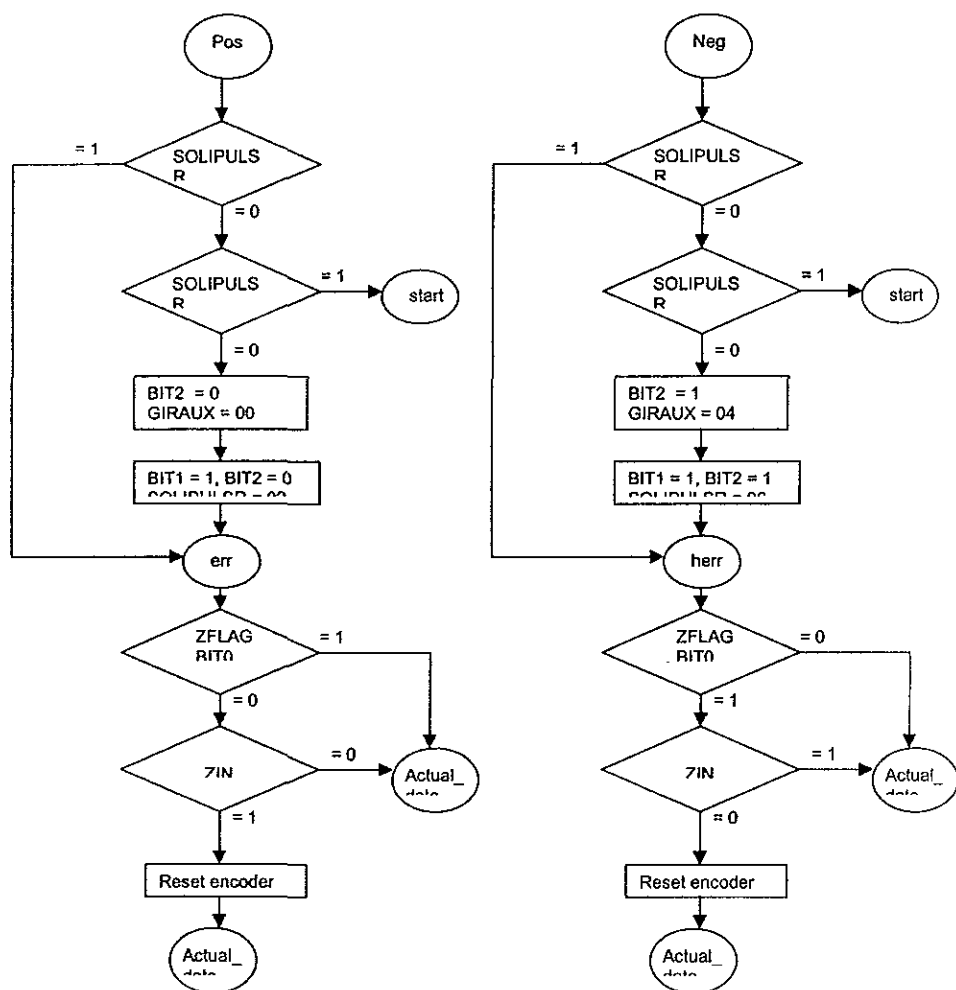
Rutina Cero_Rot



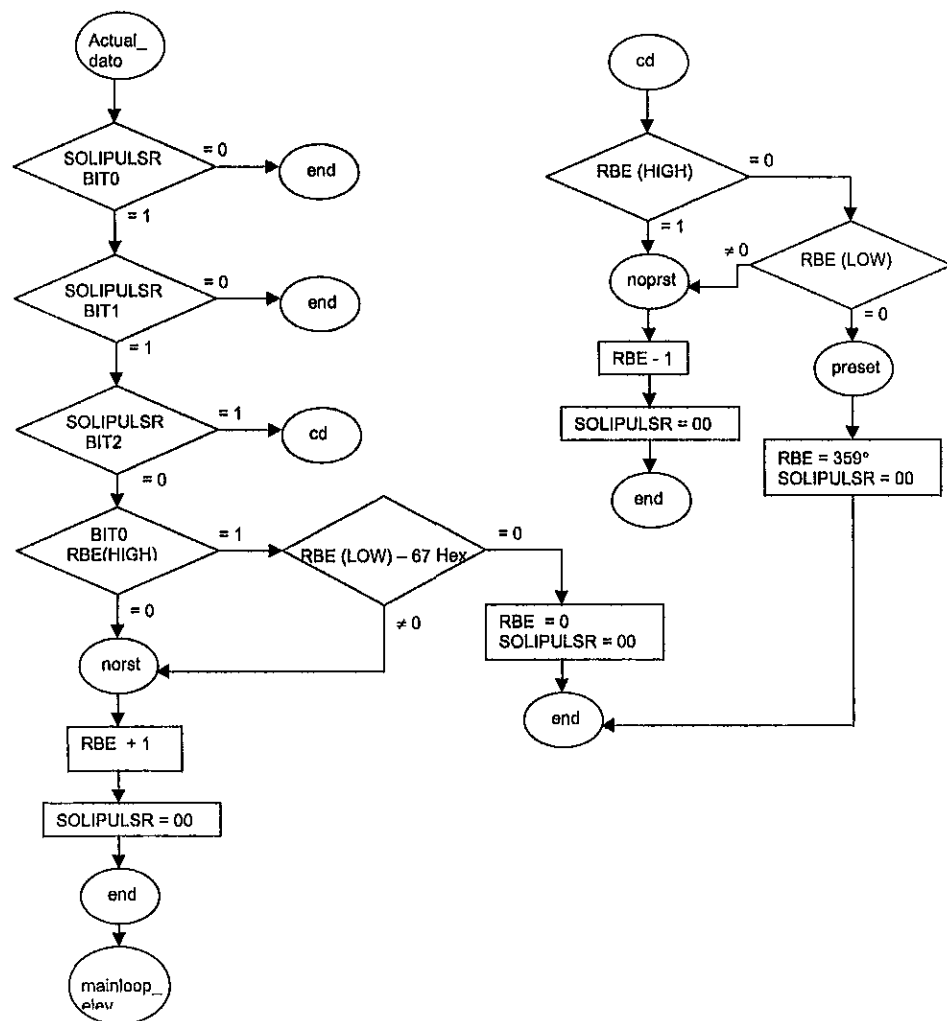
Programa_principal ROT_ELEV (a)



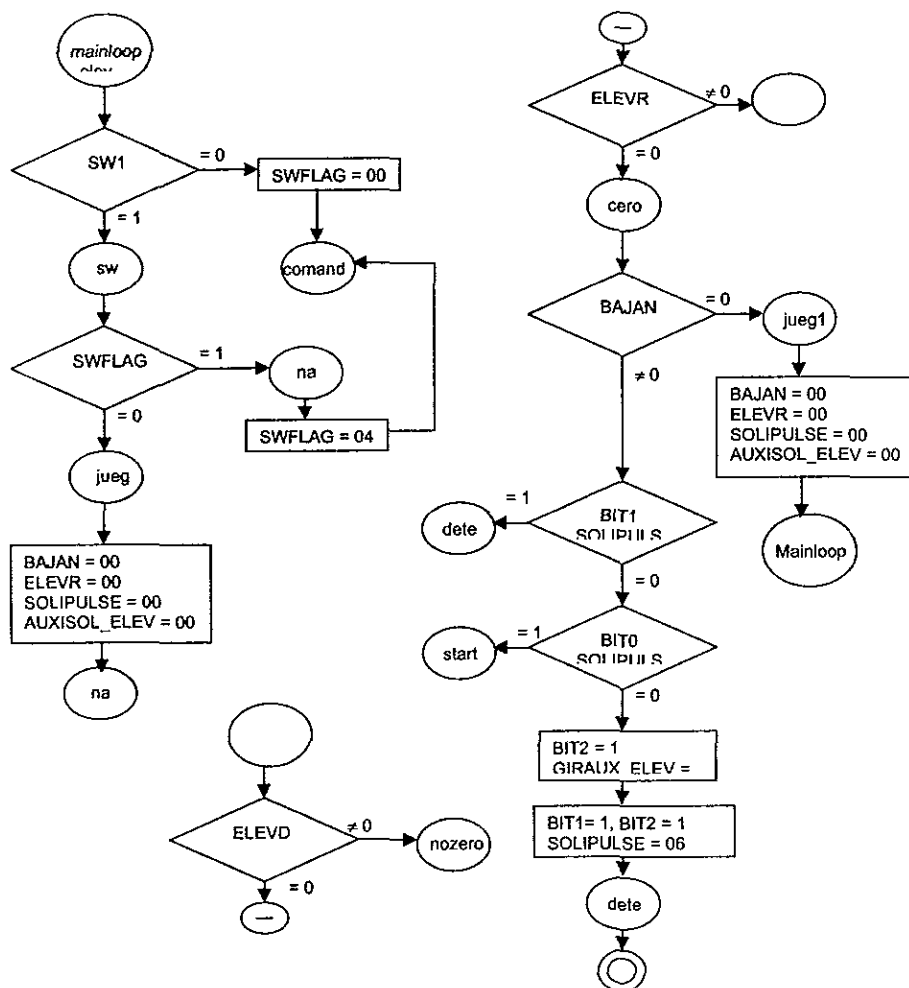
Programa_principal ROT_ELEV (b)



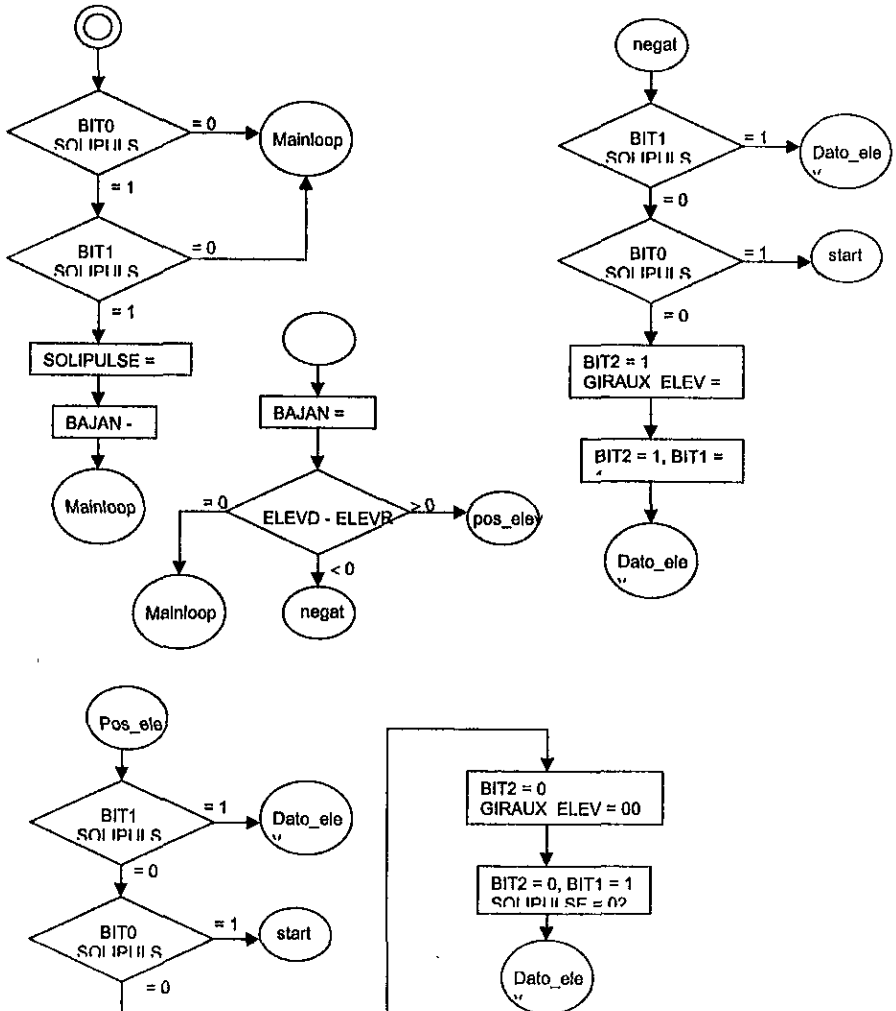
Programa_principal ROT_ELEV (c)



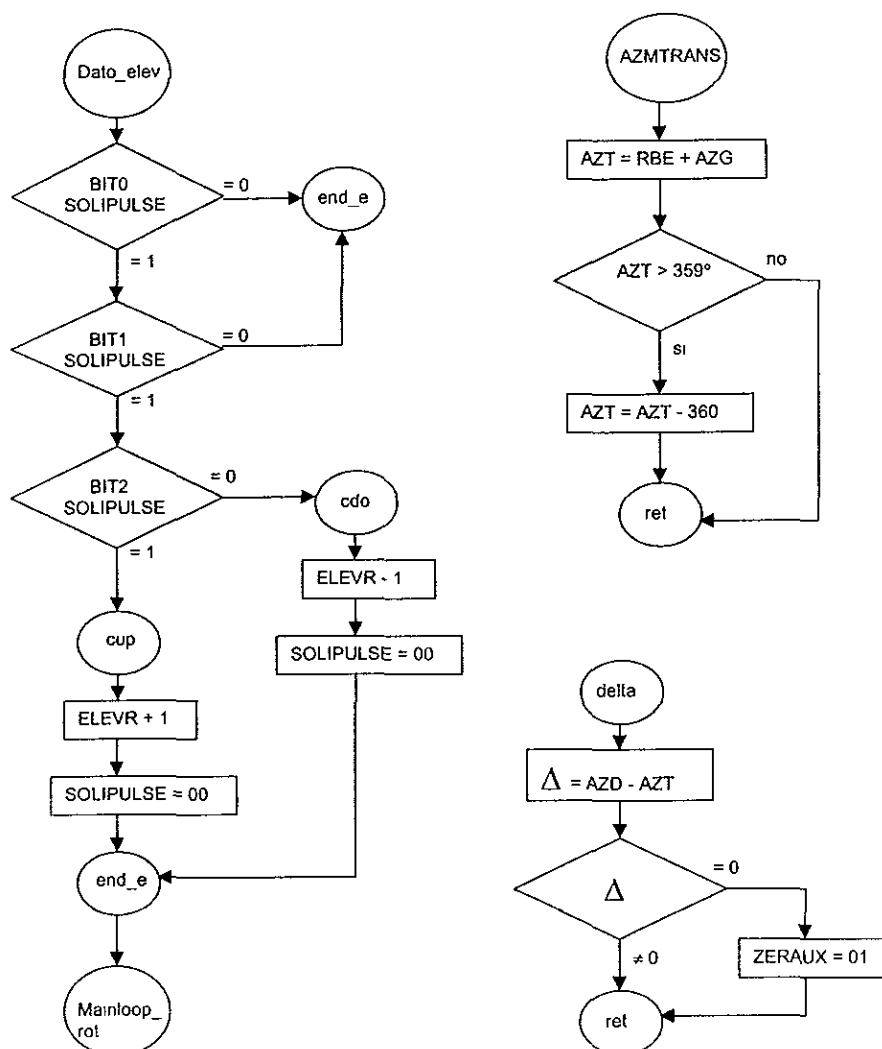
Programa_principal ROT_ELEV (d)



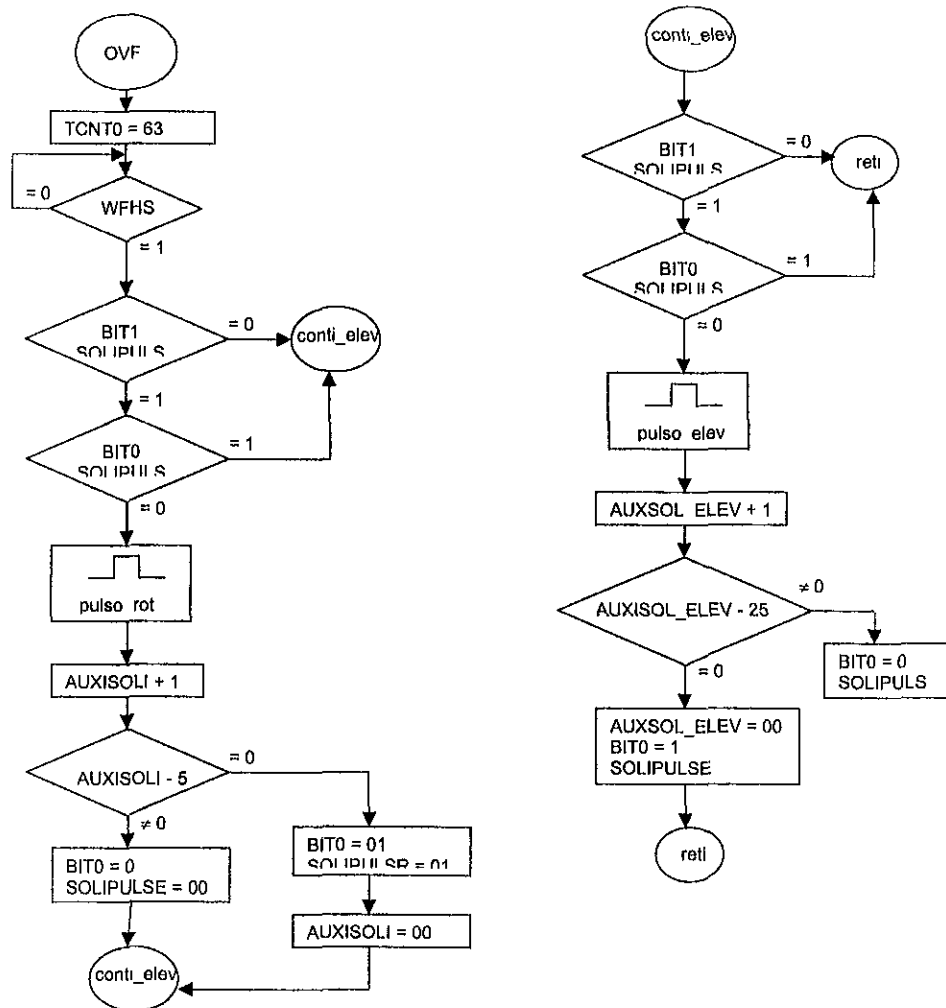
Programa_principal ROT_ELEV (e)



Programa_principal ROT_ELEV (f)



Programa_principal ROT_ELEV (g)



APÉNDICE C

En este último apéndice se muestra el listado del programa Rot_Elev en lenguaje ensamblador y el programa en C de la comunicación por puerto paralelo.

APÉNDICE C

C.1 Programa en lenguaje ensamblador Rot_Elev

```

*****
;
;
; Este programa realiza las funciones de control del movimiento de Rotación
; y de Elevación de la sección inferior de la sonda
;
;
;
;
;
;
*****

.device AT90S8515

;*****Declaración de variables y constantes

.include "iodef.inc"

;*****

        rjmp reset

.org INT0addr
        rjmp EXT_INT0      ;Interrupción externa INT0 (valid comm)

.org INT1addr
        rjmp EXT_INT1      ;Interrupción externa INT1 (Go)

.org ICP1addr
        rjmp TIM1_CAPT     ;Interrupción ICP (señal AZG)

.org OVF0addr
        rjmp TIM0_OVF      ;Interrupción Overflow0

;*****
;
;
; Rutina POWRESET
;
; En esta rutina se inicializa los puertos de entrada/salida,
; así como las variables del sistema.
;
;
;
;*****

.def temp=r18

```

```

def ini=r19
def config=r20
POWRESET:

    ser r16
    out ddra,r16      ;programa el puerto A
    out ddrC,r16      ;y el puerto C como salidas
    ldi r17,0x20       ;programa el bit 5 como salida del puerto D
    out ddrd,r17      ;y el bit 2 como entrada para habilitar INTO

;*****Iniciación del vector de interrupción OVF*****

    ldi config,0x02    ;Se configura la
    out TIMSK,config   ; interrupcion OVF timer0

;*****Iniciación de variables RAM*****

    clr ini             ;inicializa
    sts RBEL,ini        ;
    sts RBEH,ini        ;
    sts AZDL,ini        ;
    sts AZDH,ini        ;
    sts AZTL,ini        ; las
    sts AZTH,ini        ;
    sts AZGL,ini        ;
    sts AZGL,ini        ;
    sts DELTAH,ini      ;
    sts DELTAH,ini      ; variables
    sts COUNTENCOD,ini ,
    sts ENCOD,ini       ;
    sts ELEVR,ini       ;
    sts ELEVD,ini       ;
    sts LOWZ,ini        ;
    sts BAJAN,ini       ;
    sts ZEROING,ini     ; SRAM
    sts SWFLAG,ini      ;
    sts ZERAUX,ini       ;
    sts GIRAUX,ini       ;
    sts ZFLAG,ini        ;
    sts GIRAUX_ELEV,ini ; del
    sts SOLIPULSR,ini    ;
    sts SOLIPULSE,ini    ;
    sts AUXISOL,ini      ;
    sts AUXSOL_ELEV,ini , sistema
    sts REALL,ini        ;
    sts REALH,ini        ;
    ret

```

```

*****
;* Rutina Cero Elevación
;*
;* Su funcion es colocar el transductor de Elevación a cero grados
;*
*****

.def flag=r16
.def calc=r17
.def conf=r18

Cero_elev:
    sei
    ldi conf,0x04                ;prescalador CK/256(0x04)
    out TCCR0,conf
    ldi conf,0x63
    out TCNT0,conf
    sbic pind,sw1                ;Si SW=0 entonces
    rjmp skeep                   ;en via pulso a paso -
    otra:ldi yl,0x2D              ;hasta encontrar el cero Elevacion.
again:sbi portc,dir2             ;sentido de giro -
    lds calc,SOLIPULSE           ;Paso en sentido negativo
    sbrc calc,1
    rjmp det
    sbrc calc,0
    rjmp start
    sbi portc,dir2               ;Sentido negativo
    lds calc,GIRAUX_ELEV
    ori calc,0x04                ;Coloca un b2=1
    sts GIRAUX_ELEV,calc
    lds calc,SOLIPULSE
    ori calc,0x06                ;en
    sts SOLIPULSE,calc           ; sentido de giro -
    det:sbic pind,sw1            ;Si sw=1 entonces
    rjmp skeep                   ;se tiene cero elevación.
    lds conf,SOLIPULSE           ;Si b1=1 & b0=1
    sbrc conf,0                  ; entonces
    sbrc conf,1
    rjmp again                   ; avanza un grado.
    clr zl
    sts SOLIPULSE,zl
eshora:dec yl                    ;Si SW=0
    tst yl                       ; avanza 45 grados
    brne again
skeep:ldi conf,0x19
    clr zl
    sts ELEVR,zl
    sts SOLIPULSE,zl
    sts AUXSOL_ELEV,zl
    ret

```

```

.....
; Rutina CEROENCOD
;
; Rutina en la que el sistema se encarga de encontrar el cero del
; encoder
.....

```

```

def flag=r16
def res =r17
def conf=r18

```

```

;*****Cero del encoder

```

CEROENCOD:

```

loop1:
    sei ;
    ldi conf,0x00 ;
    out TCNT1H,conf ;
    out TCNT1L,conf ;
    out ICR1L,conf
    out ICR1H,conf
    sts AZGL,conf
    sts AZGH,conf
    nop ;habilita interrupción global
    lds conf,SOLIPULSR ;
    sbrc conf,1 ;Si b1=1 esta enviando un grado
    rjmp acdat ;
    sbrc conf,0 ;Si b1=0 & b0=0 solicita un pulso
    rjmp start ;Si b1=0 & b0=1 error
    cbi portc,dir1 ;Solicita 1 grado
    clr yl ;
    sts GIRAUX,yl ; b2=0
    lds res,SOLIPULSR ;
    ori res,0x02 ; en
    andi res,0x0B ;
    sts SOLIPULSR,res ; sentido de giro +
    acdat:lds conf,SOLIPULSR ;
    sbrc conf,0 ;Si b0=1 & b1=0
    sbrc conf,1 ;
    rjmp cll ; entonces sigue enviando pulsos
    clr res ;Si b1=1 $ b0=1
    sts SOLIPULSR,res ;se envió un grado y se limpia el registro
    cll:lds flag,LOWZ ;
    sbrc flag,0 ;Si Z esta en bajo, detectara la subida
    rjmp low ;
    sbic pind,Zin ;Si Z esta en alto,se envian n pulsos+
    rjmp loop1 ;hasta detectar la bajada
    ldi r19,0x01 ;
    sts LOWZ,r19 ;Detecto la bajada

```

```
low: sbis pind,Zin      ;se envía a n pulsos+,hasta que Z suba
      rjmp loop1       ;Si no hay subida,se repite el proceso
      ldi res,0x00      ;
      sts RBEL,res      ;Si hay subida, se detectó el cero, y
      sts RBEH,res      ; se resetean
      sts SOLIPULSR,res  ; las
      sts AUXISOLI,res   ; variables RBE y COUNTENCOD
      ret
```

```
*****
;* Rutina RETARINI
```

```
;* Rutina de retardo de inicialización del sistema
```

```
;* El valor del retardo es de: RegX*TCNT0*CK/1024
```

```
;* o bien; 256*256*CK/1024
```

```
;* si, f=4Mhz entonces Tret= 16.78 seg
```

```
.def read=r16
```

```
.def const=r17
```

```
*****Retardo inicial
```

```
RETARINI:
```

```
    cli
    ldi const,0x05      ;Se ajusta el preescalador CK/1024(poner 0x05)
    out tccr0,const     ;configurando TCCR0
    clr xl              ;se detectaran 256 pase por cero
    out tcnt0,xl        ;se pone en ceros el TCNT0
    in read,tifr        ;
    clr read            ;resetea el bit TOV0
loop: in read,tifr      ;
    sbrc read,tov0     ;espera un cero en TCNT0
    rjmp loop          ;
    ldi r18,0x02       ;se resetea el bit TOV0
    out tifr,r18        ;
    dec xl             ;conteo descendente
    brne loop          ;espera a que X=0
    ret
```

```
;//*****Inicio del programa*****//
```

RESET.

.def temp=r16

cseg

***** Inicializa Stack Pointer *****

```
ldi    temp,low(RAMEND)
out    SPL,temp
ldi    temp,high(RAMEND)
out    SPH,temp
```

```
start:rcall POWRESET      ;inicializa puertos y variables
rcall retarini            ;retardo inicial del sistema
rcall cero_elev           ;rutina que busca el cero de elevación
rcall ceroencod           ;rutina que busca el cero del encoder
```

def config=r17

*****Iniciación de los vectores de interrupción INT0,INT1*****

```
ldi config,0xC0           ;se configura el registro GIMSK
out GIMSK,config          ;para habilitar INT0 y la INT1
ldi config,0x0f           ;se configura el MCUCR para generar INT0,INT1
out MCUCR,config          ;con flanco de subida 0x0F
```

*****Iniciación del vector interrupción ICP*****

```
ldi config,0x0A           ;la interrupción del timer/Counter1 y OVF timer0
out TIMSK,config          ;evento de captura ICP se habilita
ldi config,0x07           ;se configura la entrada de captura
out TCCR1B,config         ;en flanco de bajada (falling edge)
                          ;y sin el preescalador, CK/1
```

```
*****
*
* Inicio del ciclo normal de operación del programa Rotación_Elevacion
*
*****
```

```
.def conf=r16
def clear=r17
```

```
.def prueba=r18
.def opera=r19
.def suma=r20
.def suma1=r25
.def gimst=r23
.def gocyc=r24
```

mainloop_rot:

```
sei
in r16,pind ;guarda
andi r16,0x80 ; estado
sts ZFLAG,r16 ; anterior de Zin
rcall azmtrans ;cálculo del Azimut del transductor
rcall delta ;cálculo del valor delta
lds prueba,DELTAH ;compara el valor de delta
tst prueba ;
brmi deltanega ;prueba si es negativo
brne deltaposi ;
lds prueba,DELTAL ;
tst prueba ;
brne deltaposi ;prueba si es positivo
rjmp mainloop_elev ;si es cero,continua con el cálculo del dato Elevación
deltaposi:clc ;limpia el bit del carry
lds r23,DELTAL ;
subi r23,0xB4 ;delta es positivo
lds r24,DELTAH ; se le resta
sbci r24,0x00 ; 180 grados
brcc neg ;si aún es positivo envía paso -
pos:lds conf,SOLIPULSR ;de lo contrario envía paso +
sbrc conf,1 ;Si b1=1 esta enviando un grado
rjmp err ;
sbrc conf,0 ;Si b1=0 & b0=0 solicita un pulso
rjmp error ;Si b1=0 & b0=1 error
cbi portc,dir1 ;Solicita 1 grado
clr yl ;
sts GIRAUX,yl ;b2=0
lds opera,SOLIPULSR ;
ori opera,0x02 ;en
andi opera,0x0B ;
sts SOLIPULSR,opera ;sentido de giro +
err:in prueba,pind ;
andi prueba,0x80 ;
lds opera,ZFLAG ;
sbrc opera,7 ;Calcula el estado de Zin
rjmp finpos ;
sbrs prueba,7 ;
rjmp finpos ;
rcall resetencod ;entonces RBE =00
rjmp actual_dato ;
```

```

finpos: sts ZFLAG, prueba ;
      rjmp actual_dato ;
deltanega: clc ; limpia el bit del carry
      lds r23, DELTA ; delta es negativo
      ldi suma, 0xB4 ; se
      add suma, r23 ; le
      lds yh, DELTAH ; suma
      ldi suma1, 0x00 ; 180 grados
      adc suma1, yh ;
      brmi pos ; si aún es negativo envía paso +
neg: lds conf, SOLIPULSR ; de lo contrario envía paso -
      sbrc conf, 1 ; Si b1=1 espera a que envíe un grado
      rjmp herr ;
      sbrc conf, 0 ; Si b1=0 & b0=0 solicita un pulso
      rjmp error ; Si b1=0 & b0=1 error
      sbi portc, dir1 ; Solicita 1 grado
      lds conf, GIRAUX ;
      ori conf, 0x04 ; Coloca un b2=1
      sts GIRAUX, conf ;
      lds opera, SOLIPULSR ;
      ori opera, 0x06 ; en
      sts SOLIPULSR, opera ; sentido de giro -
herr: in prueba, pind ;
      andi prueba, 0x80 ; Calcula el estado de Zin
      lds opera, ZFLAG ;
      sbrc opera, 7 ;
      rjmp finneg ; Si Zin=1
      sbrc prueba, 7 ; entonces
      rjmp finneg ;
      rcall resetencod ; RBE =00
      rjmp actual_dato ;
finneg: sts ZFLAG, prueba ;

```

.* Esta rutina actualiza el dato del encoder RBE

```

.def rbest=r19
.def encodst=r20
.def suma2=r23
.def rbesth=r24
.def coloca=r25

```

```

actual_dato.
      lds coloca, SOLIPULSR
      sbrc coloca, 0 ; Si b1=1 & b0=1 & b2=0
      rjmp end ; entonces RBE + +
      sbrc coloca, 1 ; Si b0=1 & b1=1 & b2=1
      rjmp end ; entonces RBE - -
      sbrc coloca, 2 ; brinca

```

```

rjmp cd ; a cd
lds rbesth,RBEH ;
sbrs rbesth,0 ;Si bit0 de RBE >= 359
rjmp norst ;
lds rbest,RBEL ;
cpi rbest,0x067 ;entonces RBE=00
breq reset1 ;
norst:clc ;Si RBE<359
lds rbest,RBEL ;
ldi suma2,0x01 ;incrementa
add rbest,suma2 ; cuenta
sts RBEL,rbest ;
clr xl ; RBE + 1
adc rbesth,xl ;
sts RBEH,rbesth ;
clr yl ;
sts SOLIPULSR,yl ;
rjmp end ;
reset1:clr yl ;RBE=00
sts RBEL,yl ;
sts RBEH,yl ;
sts SOLIPULSR,yl ;
rjmp end ;
cd:lds rbest,RBEL ;Si RBE = 00
lds rbesth,RBEH ;
sbrs rbesth,0 ;entonces RBE= 359
rjmp noprst ;
lds rbest,RBEL ;
tst rbest ;
breq preset ;
noprst:clc ;Si RBE diferente 359
lds rbesth,RBEH ;
lds rbest,RBEL ;decrementa
subi rbest,0x01 ;
sts RBEL,rbest ; cuenta
sbci rbesth,0x00 ;
sts RBEH,rbesth ; RBE - 1
clr yl ;
sts SOLIPULSR,yl ;
rjmp end ;
preset:ldi coloca,0x67 ;
sts RBEL,coloca ;
ldi coloca,0x1 ;
sts RBEH,coloca ;
clr yl ;
sts SOLIPULSR,yl ;
end:rjmp mainloop_elev ;fin del ciclo
error:rjmp start

```

```

.def band=r16
.def zero=r17

```

```
def elevauxst=r18
```

```
mainloop_elev:
```

```
    sbic pind,sw1      ;sondeo del switch SW
    rjmp sw            ;
    clr xl             ;si es cero, pone la bandera en cero
    sts swflag,xl      ;
    rjmp comando       ;y termina el sondeo de switches
sw:lds band,swflag     ;si es uno el switch SW,checa la bandera
    sbrc band,0        ;
    rjmp na            ;
jueg:lds zero,ZEROING  ;pone bandera de avance de juego
    ori zero,0x01
    sts ZEROING,zero   ;
    clr xl             ;si la bandera=0
    sts BAJAN,xl       ;resetea
    sts ELEVR,xl       ; los datos de elevacion
    sts ZEROING,xl     ;quita bandera de avance de juego
na:lds band,SWFLAG     ;
    ori band,0x40      ;no hubo transición en SW
    sts SWFLAG,band    ;
    rjmp comando       ;
jueg1:clr zl           ;si la bandera=0
    sts BAJAN,zl       ;resetea
    sts ELEVR,zl       ; los
    sts AUXSOL_ELEV,zl ; datos
    sts SOLIPULSE,zl   ; de elevación
    lds band,SWFLAG    ;
    ori band,0x40      ;no hubo transición en SW
    sts SWFLAG,band    ;
    rjmp mainloop_rot
```

```
.def calc=r16
.def elevdst=r18
.def zero=r19
.def elevrst=r17
.def bajanst=r20
.def clear=r23
```

```
comando:
```

```
    sei                ;
    sbis pind,WFHS     ;Si existe WFHS
    rjmp comando       ;el sistema no opera
    lds elevdst,ELEVD  ;prueba el dato
    tst elevdst        ;deseado de elevd
    brne nozero_elev   ;si es cero
```

```

    lds elevrst,ELEVR    ;prueba el dato
    tst elevrst          ; de elevr
    brne nozero_elev     ;si es cero
cero:lds bajanst,BAJAN   ;prueba el valor de
    tst bajanst          ; bajan
    breq jueg1           ; bajan = 0
    lds calc,SOLIPULSE   ;Paso en sentido negativo
    sbrc calc,1           ;
    rjmp dete            ;
    sbrc calc,0           ;
    rjmp start           ;
    sbi portc,dir2
    lds calc,GIRAUX_ELEV ;
    ori calc,0x04         ;Coloca un b2=1
    sts GIRAUX_ELEV,calc ;
    lds calc,SOLIPULSE   ;
    ori calc,0x06         ;en
    sts SOLIPULSE,calc    ; sentido de giro -
dete:lds conf,SOLIPULSE  ;Si b1=1 & b0=1
    sbrc conf,0           ; entonces
    sbrc conf,1           ;
    rjmp wpr             ; avanza un grado.
    clr yl
    sts SOLIPULSE,yl
    dec bajanst
    sts BAJAN,bajanst
    wpr:rjmp mainloop_rot ;
nozero_elev:ldi xl,0x0A   ;Bandera de regreso a cero
    sts BAJAN,xl          ;
    lds elevdst,ELEVD     ;
    lds elevrst,ELEVR     ;se compara
    cp elevdst,elevrst     ; Elevd - Elevr
    brne next_elev        ;Si no es cero
    rjmp mainloop_rot     ;Si es cero, regresa inicio del loop
next_elev:brmi negat      ;Si Elevr > Elevd envia paso negativo
pos_e:lds calc,SOLIPULSE  ;Si Elevr < Elevd envia paso positivo
    sbrc calc,1           ;Paso ++
    rjmp ere              ;Si b1=0 & b0=0 solicita un grado
    sbrc calc,0           ;Si b1=0 & b0=1 error
    rjmp error_e          ;Si b1=1 se esta enviando un pulso, no se puede
    cbi portc,dir2        ; solicitar un grado
    lds zero,ZEROING      ;Checa bandera de avance de juego
    sbrc zero,0           ;Se solicita un
    rjmp ere              ; grado
    clr yl                ; en
    sts GIRAUX_ELEV,yl    ; sentido positivo
    lds calc,SOLIPULSE   ;
    ori calc,0x02         ;
    andi calc,0x0B        ;
    sts SOLIPULSE,calc    ;
    ere:rjmp Dato_elev     ;actualiza dato
negat:lds calc,SOLIPULSE  ;Paso --
    sbrc calc,1           ;Si b1=0 & b0=0 solicita un grado

```

```

rjmp eren      ;Si b1=0 & b0=1 error
sbrc calc,0    ;Si b1=1 se esta enviando un pulso, no se puede
rjmp error_e   ;
sbi portc,dir2 ;
lds zero,ZEROING ;Checa bandera de avance de juego
sbrc zero,0    ; Se solicita un
rjmp eren      ;
lds calc,GIRAUX_ELEV ; grado
ori calc,0x04  ; en
sts GIRAUX_ELEV,calc ;Coloca un b2=1
lds calc,SOLIPULSE ;
ori calc,0x06  ; en
sts SOLIPULSE,calc ; sentido de giro -
eren:rjmp Dato_elev ;actualiza dato

```

.def datelev=r20

.def sumel=r23

;****rutina actualiza dato de elevación***

Dato_elev:

```

    lds datelev,SOLIPULSE ;
    sbrs datelev,0        ;Si b1=1 & b0=1 & b2=0
    rjmp end_e            ;entonces RBE ++
    sbrs datelev,1        ;Si b0=1 & b1=1 & b2=1
    rjmp end_e            ;entonces RBE - -
    sbrc datelev,2        ;
    rjmp cdo              ;
cup:lds elevrst,ELEVR     ;incrementa cuenta
    ldi sumel,0x01        ; de elevacion real
    add sumel,elevrst     ;Elevacion + 1
    sts ELEVR,sumel       ;
    clr yl                ;ya se envió un grado
    sts SOLIPULSE,yl      ; se limpia el registro
    rjmp end_e            ;
cdo:lds elevrst,ELEVR     ;decrementa la cuenta
    subi elevrst,0x01     ; de elevacion real
    sts ELEVR,elevrst     ;Elevacion - 1
    clr yl                ;ya se envió un grado
    sts SOLIPULSE,yl      ; se limpia el registro
    rjmp end_e            ;va al inicio del ciclo
error_e:rjmp start        ;error regresa al inicio del programa
end_e:rjmp mainloop_rot   ;va al inicio del ciclo

```

include "rutint inc"

```

*****
;
;

```

```

;* Rutina AZMTRANS
;* Esta rutina calcula el azimut del transductor
;* a partir del dato del encoder(RBE) y del giroscopio(AZG).
;*****

.def rbest=r16
.def azgst=r17
.def rbesth=r18
.def azgsth=r19
.def aztst=r20
.def aztsth=r23

;*****Azimut del transductor
;

azmtrans:

    clc                ;limpia bit carry
    lds rbest,RBEL     ;
    lds azgst,AZGL      ;Calculo
    add rbest,azgst     ;
    sts AZTL,rbest      ; del
    lds rbesth,RBEH     ;
    lds azgsth,AZGH     ; AZT=RBE + AZG
    adc rbesth,azgsth   ;
    sts AZTH,rbesth     ;
    clc                ;limpia bit carry
    lds aztst,AZTL      ;
    subi aztst,0x68     ;Compara el valor de AZT
    lds aztsth,AZTH     ;
    sbci aztsth,0x01    ;Si AZT > 359
    brcs sigue         ;
    sts AZTL,aztst      ; entonces
    sts AZTH,aztsth     ;
    sts REALH,aztsth    ; AZTreal=AZT - 360
    sts REALL,aztst     ;
    ret                ;
sigue:sts REALL,rbest   ;Escribe el dato a enviar
    sts REALH,rbesth    ; del AZT
    ret                ; en REAL

;*****
;*
;* Rutina RESETENCOD
;* En esta rutina se reinicializa las variables del encoder.
;*****

.def girauxst=r16

;*****Rutina reset_encoder
;

```

resetencod:

```

    clr xl          ;Se encuentra
    sts RBEL,xl     ; en cero grados
    sts RBEH,xl     ; por lo que
    sts SOLIPULSR,xl ; los registros
    sts AUXISOLI,xl ; se limpian
    ret             ;

```

```

;*****
;
; Rutina DELTA
; Esta rutina hace el cálculo del valor de delta, que está definido como la
; diferencia entre el azimut deseado y el azimut del transductor.
; DELTA= AZD-AZT
;*****

```

```

.def flag=r16
.def azdst=r17
.def azdsth=r18
.def aztst=r19
.def aztsth=r20

```

;*****Cálculo de delta

delta:

```

    clr xl          ;se limpia bandera
    sts ZERAUX,xl   ; de cero delta
    clc             ;
    lds azdst,AZDL  ;Calcula el valor de delta
    lds aztst,AZTL  ;
    sub azdst,aztst ; DELTA = AZD - AZT
    brne nozero     ;
    lds xl,ZERAUX   ;Si DELTA = 0 entonces
    ori xl,0x01     ; pone bit0=1 en ZERAUX
    sts ZERAUX,xl   ;
nozero: sts DELTA,azdst ;Si DELTA != 0
    lds azdsth,AZDH ;
    lds aztsth,AZTH ; no hace nada
    sbc azdsth,aztsth ;
    sts DELTAH,azdsth ,
    ret

```

Rutinas de interrupción

```

*****
;
;*
;* Rutina de Interrupción_rot_elev
;*
;*
;*
;
*****

```

```

.def aztlst=r0
.def azthst=r1
.def elevrst=r2
.def calc1=r3
.def azdlst=r4
.def azdhst=r5
.def elevdst=r6
.def dato=r7

```

```

;*****Rutinas de interrupción Valid Comm*****
;

```

EXT_INT0:

```

    in r22,sreg      ;guarda el estado
    push r22         ; del SREG en el stack
    clr xl           ;
    out ddra,xl      ;se programa como entrada el portA
    sbi portc,OE      ;habilito el
    nop              ; OE de
    nop              ; rotacion
    in azdlst,pina    ;leo el dato
    sts AZDL,azdlst   ; de rotacion AZDL
    cbi portc,OE      ;deshabilito OE de rotacion
    sbi portc,OE1     ;habilito
    nop              ; OE'
    nop              ; de elevacion
    in elevdst,pina   ;lee dato de ELEV y AZDH
    cbi portc,OE1     ;deshabilito
    nop              ;
    nop              ; el OE'
    lsr elevdst       ;
    clr xl           ;
    sts AZDH,xl       ;
    lds azdhst,AZDH   ;
    rol azdhst        ;
    sts AZDH,azdhst   ;
    clc              ;
    lds azdhst,AZDH   ;
    ldi r21,0x5B      ;
    mov r8,r21        ;
    cp elevdst,r8      ;compara el dato de elevd
    brcs dato_ok      ;Si Elevd > 90

```

```

ldi r21,0x5A      ; entonces Elevd = 90
mov elevdst,r21    ;
dato_ok:sts ELEVd,elevdst ;
ser xl             ;
out ddra,xl        ;se programa el portA como salida
pop r22            ;
out sreg,r22       ;Recupera el estado
reti               ; del SREG

```

.....*Rutina pulso GO*.....

EXT_INT1:

```

in r22,sreg        ;guarda el estado
push r22           ; del SREG en el stack
lds aztlst,REALL   ;leo el dato de rotacion
out porta,aztlst   ;
sbi portc,GO       ;envia el pulso go'
nop               ;se envian los datos hacia el registro
nop               ;de corrimiento 4021 (ROT)
nop               ;a traves de porta
cbi portc,GO       ;se
nop               ;
nop               ; limpia
nop               ; el pulso go'
lds elevrst,ELEVR  ;
lsl elevrst        ;desplazo los dato a la izq.
lds azthst,REALH   ;leo el dato del Azimut transductor
ldi r21,0x01       ;parte alta
mov r8,r21         ;
and azthst,r8      ;se envian el dato de Elevr y AZTH hacia el
or elevrst,azthst  ;registro de corrimiento 4021 (ELEV)
out porta,elevrst  ;a traves del porta
sbi portc,GO1      ;envia
nop               ;
nop               ; el
nop               ; pulso go"
cbi portc,GO1      ;se limpia el pulso go"
pop r22            ;Recupera el estado
out sreg,r22       ; del SREG
reti               ;

```

.....*Rutina señal del giroscopio AZG*.....

```

def tfirst=r0
def regcountl=r1
def regcounth=r2

```

.def temp1=r4

TIM1_CAPT:

```

    in r22,sreg           ;guarda el estado
    push r22             ; del SREG en el stack
    in regcountl,ICR1L    ;Lee el dato del Input
    in regcounth,ICR1H    ; Capture Register
    ldi r21,0x01          ;
    mov r5,r21            ;Si el bit0=0 ICRH
    and regcounth,r5      ;
    sbrc regcounth,0      ;entonces no hace comparación
    rjmp espw            ;
    clc                  ;Si bit0=1
    ldi r21,0x68          ; entonces
    mov r5,r21            ; compara
    cp regcountl,r5       ;
    brcs espw            ;
    ldi r21,0x00          ;
    mov r5,r21            ;Si ICR > 359
    sts AZGH,r5           ;entonces
    sts AZGL,r5           ; el dato del giroscopio AZG = 0
    rjmp si_no           ;
espw:clc                 ;limpia bit carry
    ldi r21,0x01          ;ICR < 360
    mov temp1,r21         ;se resta un grado
    sub regcountl,temp1   ; ICR - 1
    sts AZGL,regcountl    ;
    ldi r21,0x00          ;
    mov r6,r21            ;AZG = ICR - 1
    sbc regcounth,r6      ;
    sts AZGH,regcounth    ;
si_no:clr xl             ;
    out TCNT1H,xl         ;resetea el registro del timer1
    out TCNT1L,xl        ;
    pop r22              ;Recupera el estado
    out sreg,r22          ; de SREG
    reti                 ;

```

.def envio=r0

.def bandf=r1

.def conf1=r2

.def wish=r3

.def conf=r4

TIM0_OVF:

```

in r22,sreg          ;guarda el estado
push r22             ;del SREG en el stack
ldi r21,0x63         ;Resetea el
mov conf,r21         ;
out TCNT0,conf       ; timer0
sbis pind,WFHS       ;si existe WFHS el sistema
rjmp conti_elev      ;no opera
lds envio,SOLIPULSR  ;
sbrs envio,1         ;
rjmp conti_elev      ;
sbrs envio,0         ;
rjmp conti_elev      ;Si b1=1 & b0=1 se movio un grado,actualiza dato ROT
cbi portc,step1      ;Si b1=1 & b0=0 envia pulso a rotacion
nop                  ;
nop                  ;Envia
sbi portc,step1      ;
nop                  ;
nop                  ; pulso
cbi portc,step1      ;
lds bandf,AUXISOLI   ; al motor de rotacion
inc bandf
sts AUXISOLI,bandf   ;
ldi r21,0x06         ;
mov r5,r21           ;
cp bandf,r5          ;Compara el contador de pulsos
breq solip           ;Si es igual a 5 resetea y actualiza dato ROT
ldi r21,0x0E         ;
mov r5,r21           ;
lds envio,SOLIPULSR  ;
and envio,r5         ;Pone al bit0=0
sts SOLIPULSR,envio  ;
rjmp conti_elev      ;
solip:ldi r21,0x00    ;
mov conf,r21         ;
sts AUXISOLI,conf    ; Limpia el contador de pulsos
ldi r21,0x01         ;
mov envio,r21        ;
lds bandf,SOLIPULSR  ;Pone b0=1 para actualizar dato ROT
or envio,bandf       ;
sts SOLIPULSR,envio  ;
ldi r21,0x04         ;Si el b2=0 entonces sentido +
mov conf,r21         ;
lds bandf,GIRAUX     ;Si el b2=1 entonces sentido -
and bandf,conf       ;
sts GIRAUX,bandf     ;
conti_elev:lds wish,SOLIPULSE ;
sbrs wish,1         ;
rjmp yasi            ;
sbrs wish,0         ;
rjmp yasi            ;
cbi portc,step2      ;Envia
nop                  ;
nop                  ;

```

```

sbi portc,step2      ; pulso
nop                  ;
nop                  ;
cbi portc,step2      ;
lds bandf,AUXSOL_ELEV      ; al motor de elevación
inc bandf            ;
sts AUXSOL_ELEV,bandf      ;
ldi r21,0x19          ;
mov r5,r21            ;
cp bandf,r5           ;
breq yanosoli         ;
ldi r21,0x0E          ;
mov r5,r21            ;
lds envio,SOLIPULSE      ;
and envio,r5           ;Pone al bit0=0
sts SOLIPULSE,envio      ;
rjmp yasi             ;
yanosoli:ldi r21,0x00      ;
mov conf,r21           ;
sts AUXSOL_ELEV,conf      ; Limpia el contador de pulsos
ldi r21,0x01           ;
mov envio,r21          ;
lds bandf,SOLIPULSE      ;Pone b0=1 para actualizar dato ROT
or  envio,bandf         ;
sts SOLIPULSE,envio      ;
ldi r21,0x04           ;Si el b2=0 entonces sentido +
mov conf,r21           ;
lds bandf,GIRAUX_ELEV      ;Si el b2=1 entonces sentido -
and bandf,conf         ;
sts GIRAUX_ELEV,bandf      ;
yasi:pop r22           ;Recupera el estado
out sreg,r22           ; de SREG
reti                   ;

```

C.2 Código del programa en C, generación de las señales de entrada al sistema de orientación

```

#include<stdio.h>
#include<conio.h>

```

```

unsigned int  rotdown,rotup;
unsigned int  eleup,eledown;
unsigned int  datadown;
unsigned char salpto;

```

```
char selec;
```

```
main()
```

```

{
    int opcion;

    clrscr();
    inicializa();
    menu();
    while(1)
    {
        if (kbhit())
        {
            opcion = getch();
            switch (opcion)
            {
                case '3':
                    clrscr();
                    gotoxy(34,20);
                    printf("SALIDA");
                    sleep(2);
                    clrscr();
                    exit(0);
                    break;

                case '1':
                    gotoxy(15,10); printf ("      ");
                    gotoxy(15,10); scanf ("%u",&rotdown);
                    if (rotdown>=360) rotdown = 359
                    ;
                    gotoxy(15,10); printf (" %3u",rotdown);
                    break;

                case '2':
                    gotoxy(15,12); printf ("      ");
                    gotoxy(15,12); scanf ("%u",&eledown);
                    if (eledown>90) eledown = 90;
                    gotoxy(15,12); printf (" %2u", eledown);
                    break;

                default:
                    break;
            }
        }
        envia_datos ();

        salpto &= 0x7f; /* wfhs */
        outportb (0x378,salpto);
        delay(70);
        salpto |= 0x80;
    }
}

```

```

    outportb (0x378,salpto);

    recibe_datos ();

    gotoxy(38,10); printf(" %4u",rotup);
    gotoxy(38,12); printf(" %4u",eleup);
    delay (63);

}

}

menu()
{
    gotoxy(15,2); printf ("INSTITUTO MEXICANO DEL PETROLEO");
    gotoxy(10,4); printf ("PRUEBAS DE ROTACION Y ELEVACION");
    printf ("\n\n 1- ROTACION.\n 2- ELEVACION. \n 3- SALIR");
    gotoxy(5,10); printf ("ROTDOWN =");
    gotoxy(5,12); printf ("ELVDOWN =");
    gotoxy(30,10); printf ("ROTUP =");
    gotoxy(30,12); printf ("ELEUP =");

}

inicializa()
{
    rotup=rotdown=eleup=eledown=0;
    datadown=0;
    salpto= 0x02|0x20|0x80 | 0x10; /* dclk,uclk, wfhs, udata */

}

envia_datos ()
{
    int i, temp;

    salpto &= 0xfb; /* valid command */
    outportb (0x378,salpto);

    datadown = (eledown << 9) & 0xfe00;
    datadown |= (rotdown & 0x1ff);
    for (i=0;i<16;i++)
    {

```

```

        salpto &= 0xfe;
        salpto |= (char) (datadown & 0x1);
        outportb (0x378,salpto);
        salpto &= 0xfd;
        outportb (0x378,salpto);
/*      delay(1);
*/      salpto |= 0x02;
        outportb (0x378,salpto);
/*      delay (1);
*/      datadown >>= 1;
    }
    salpto |= 0x04;
    outportb (0x378,salpto); /* valid command */
}

recibe_datos ()
{
    int temp,i;
    unsigned char dato;

    salpto |= 0x40;          /* go */
    outportb (0x378,salpto);
    outportb (0x378,salpto);
/*      delay(1);
*/      salpto &= 0xbf;
        outportb (0x378,salpto);
        delay (1);

    temp=0;
    for(i=0;i<16;i++)
    {
        dato = inportb (0x379);
        dato >>= 4;
        dato &= 0x1;

        temp <<= 1;
        temp |= (int) dato;

        salpto &= 0x0f;
        outportb (0x378,salpto);
/*      delay (1);
*/      salpto |= 0x020;
        outportb (0x378,salpto);

```