



**DISEÑO E IMPLEMENTACIÓN DE UN EQUIPO
DIDÁCTICO DE UN CONTROLADOR LÓGICO
PROGRAMABLE (PLC), PARA LA
UNIVERSIDAD DON BOSCO**

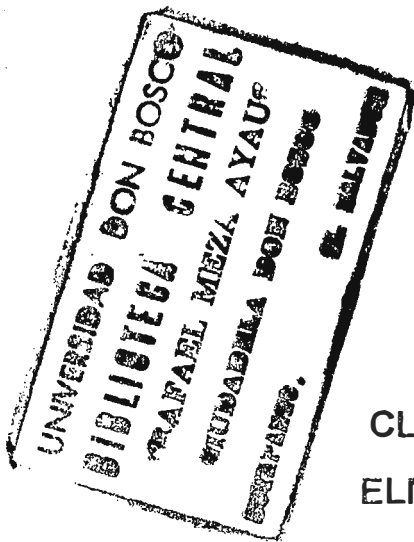
TRABAJO DE GRADUACIÓN
PREPARADO PARA LA FACULTAD
DE INGENIERÍA

PARA OPTAR AL GRADO DE:

INGENIERO ELECTRÓNICO

POR

**CLAUDIA MARÍA SANDOVAL MEJÍA
ELMER ERNESTO AVILÉS SÁNCHEZ**



JUNIO - 1998

SOYAPANGO - EL SALVADOR - CENTROAMERICA

UNIVERSIDAD DON BOSCO

RECTOR

ING. FEDERICO MIGUEL HUGUET RIVERA

SECRETARIO GENERAL

PBRO. PEDRO JOSE GARCIA CASTRO S.D.B.

DECANO DE LA FACULTAD DE INGENIERIA

ING. CARLOS GUILLERMO BRAN

ASESOR DEL TRABAJO DE GRADUACION

ING. JAIME JOEL VARELA

JURADO EXAMINADOR

ING. FEDERICO JOSÉ LAÍNEZ

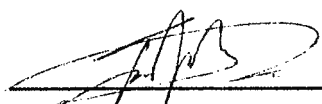
ING. ANSELMO VALDIZÓN


UNIVERSIDAD DON BOSCO


FACULTAD DE INGENIERIA

DEPARTAMENTO DE INGENIERIA ELECTRONICA

JURADO EVALUADOR DEL TRABAJO DE GRADUACION
“DISEÑO E IMPLEMENTACIÓN DE UN EQUIPO DIDÁCTICO
DE UN CONTROLADOR LÓGICO PROGRAMABLE (PLC),
PARA LA UNIVERSIDAD DON BOSCO”


ING. FEDERICO JOSÉ LAÍNEZ
JURADO


ING. ANSELMO VALDIZÓN
JURADO


ING. JAIME JOEL VARELA
ASESOR

DEDICATORIA

A **Dios Omnipotente**, que es El Padre Perfecto y Amigo fiel que me ilumina, protege y acompaña a cada instante. Por regalarme el don de la Vida y permitirme recorrerla al lado de una familia tan especial, manifestando su presencia en ellos. Por llenar cada segundo de mi existencia, con su Perfecto Amor. Porque ha puesto en mi camino las alegrías, premios y obstáculos precisos para que yo alcance cada una de mis metas y no me rinda jamás.

A **María Auxiliadora**, que me escogió para participar de su obra Salesiana desde mis primeros pasos, formando los principios y valores que me han permitido llegar hasta aquí. Porque ha sido mi mejor amiga y consejera, la Madre Perfecta que me protege con su manto celestial en cada instante de mi vida.

A *mis amados padres*, **Joel Sandoval Grijalva y Ana Luz Infante de Sandoval**, por renunciar a ellos y dedicarse a mí con tanto amor; por acompañarme y guiarme. Por celebrar hasta mis más pequeños triunfos, y perdonar todas mis faltas. Por que son el mejor ejemplo de amor, dedicación y responsabilidad que he tenido.

Gracias papás, por regalarme el mejor tesoro para el resto de mi vida: mi formación espiritual y profesional. Sinceramente, nadie lo podría haber hecho mejor que ustedes, por eso, este triunfo les pertenece.

A mis queridos hermanos: **Lya Verónica y Joel David**, pero especialmente a **Ana Beatriz**, por ser *mi segunda mamá*; por que ha sido tan amorosa, dedicada y comprensiva. Por predicar con el ejemplo, y mantener unida a mi familia.

A mi sobrinito, **David Ricardo Moran**, por existir. Y por venir a iluminar nuestras vidas con su sonrisa angelical.

A alguien muy especial, **Aarón Coreas Campos**, por ser siempre mi compañero y amigo perfecto. Por haberme dado la mano a lo largo de la carrera, y animarme en los momentos difíciles. Por todo el amor, comprensión, apoyo, y cuidados que me brinda.

Gracias Aarón por haber creído en mi.

AGRADECIMIENTOS

A mi compañero *Ernesto Avilés*, por su amistad y todo el empeño puesto en este trabajo.
Y por compartir sus conocimientos y este triunfo conmigo.

Al *Ing. Federico Lainez*, por enseñarnos a buscar la excelencia; y por todo el tiempo e interés invertidos en este trabajo.

Al *Ing. Jaime Varela* y al *Ing. Anselmo Valdizón*, por su apoyo, asesoría y buena voluntad.

Claudia María Sandoval I.

DEDICATORIA

La satisfacción más grande que puede tener un padre, es sentirse orgulloso de su hijo. Y la satisfacción más grande que puede tener un hijo, es saber que sus padres se sienten orgullosos de él.

Este documento lo dedico a mis padres, quienes me apoyaron en todo sentido para lograr terminar mi carrera universitaria. Se los dedico con todo mi corazón porque solo yo puedo dar testimonio de todos los esfuerzos y sacrificios que ellos hicieron por mí.

AGRADECIMIENTOS

Es imposible describir la satisfacción que siento al haber concluido mis estudios en Ingeniería. Es imposible también agradecer a tantas personas que me apoyaron para que siguiera adelante, luchando; dándome ánimos para alcanzar una meta, que para mi no fue nada fácil. Creo y siento la necesidad de decir, que sería injusto de mi parte no recordarlos en este documento.

A todas esas personas, que si bien no las menciono por temor a olvidar escribir sus nombres en esta página, solo les puedo decir, que no existen palabras que expresen mis sinceros agradecimientos; y que tengan la seguridad de que sus nombres siempre existirán en la lista de mi mente y en el libro de mi corazón.

Ernesto Avilés Sánchez.

INDICE GENERAL

<i>FIGURA</i>	<i>No. DE PAGINA</i>
PROLOGO	
I GENERALIDADES.....	i
II ANTECEDENTES.....	ii
III ALCANCES.....	iv
IV LIMITACIONES.....	v
INTRODUCCIÓN GENERAL.....	vi
CAPITULO 1	
SISTEMAS Y CIRCUITOS ELECTRÓNICOS DE CONTROL.....	1
1.1 SISTEMAS DE CONTROL DE LAZO CERRADO.....	3
1.1.1 CIRCUITOS ELECTRÓNICOS DE LAZO CERRADO.....	6
1.1.2 MODOS DE CONTROL DE LAZO CERRADO.....	9
1.1.2.1 CONTROL TODO O NADA.....	10
1.1.2.2 CONTROL PROPORCIONAL (P).....	13
1.1.2.3 CONTROL PROPORCIONAL-INTEGRAL (PI).....	18
1.1.2.4 CONTROL PROPORCIONAL-DERIVATIVO (PD).....	23
1.1.2.5 CONTROL PROPORCIONAL-INTEGRAL-DERIVATIVO (PID).....	26
1.1.3 CARACTERÍSTICAS TÍPICAS EN EL CONTROL DE PROCESOS INDUSTRIALES.....	30
1.1.3.1 RETARDO DE REACCIÓN DE PROCESO.....	30
1.1.3.2 TIEMPO MUERTO.....	31
1.1.3.3 ATRASO DE TRANSFERENCIA.....	32
1.1.4 TABLA RESUMEN DE LOS MODOS DE CONTROL.....	33
1.2 TRANSDUCTORES.....	34
1.2.1 CLASIFICACIÓN.....	35
1.3 CONVERTIDORES ANÁLOGO-DIGITAL.....	37
1.3.1 CONVERTIDOR SIMULTANEO A/D.....	37
1.3.2 CONVERTIDOR A/D DE CONTADOR.....	39
1.3.3 CONVERTIDOR CONTINUO A/D.....	40
1.3.4 CONVERTIDOR POR APROXIMACIÓN SUCESIVA A/D.....	44
1.4 MUESTREO Y MANTENIMIENTO.....	47
1.5 CONVERTIDORES DIGITAL-ANÁLOGO.....	49
1.5.1 DAC CON SALIDA DE VOLTAJE.....	49
1.5.2 DAC CON SALIDA DE CORRIENTE.....	51
1.6 ETAPA DE POTENCIA.....	53
1.6.1 TIRISTORES.....	53
1.6.2 RELÉS Y CONTACTORES.....	55
1.7 AISLAMIENTO ELÉCTRICO.....	60
1.7.1 ACOPLES ÓPTICOS.....	60

CAPITULO 2

EL CONTROLADOR LÓGICO PROGRAMABLE.....	62
2.1 HISTORIA DEL PLC.....	62
2.2 EL PLC.....	66
2.2.1 VENTAJAS COMPETITIVAS.....	66
2.2.2 CARACTERÍSTICAS DEL PLC.....	67
2.2.3 ESTRUCTURA DEL PLC.....	68
2.2.3.1 DIAGRAMA DE BLOQUES DE UN PLC.....	68
2.2.4 SECCIÓN PROCESADORA.....	69
2.2.5 FUENTE DE PODER.....	70
2.2.6 MEMORIAS.....	72
2.2.6.1 MEMORIAS ROM.....	73
2.2.6.2 MEMORIAS RAM.....	77
2.2.7 ENTRADAS/SALIDAS DE LOS PLCs.....	81
2.2.7.1 SISTEMAS DE ENTRADA/SALIDA.....	81
2.2.7.2 MÓDULOS DE ENTRDA/SALIDA.....	84
2.2.8 PROGRAMACIÓN DEL PLC.....	88
2.2.8.1 SOFTWARE PROCESADOR.....	88
2.2.8.2 SOFTWARE DEL USUARIO.....	90
2.2.8.3 PROGRAMACIÓN.....	95
2.2.8.4 TIPOS DE CARGADORES DE PROGRAMAS.....	96
2.2.9 INSTRUCCIONES BÁSICAS DEL PLC.....	97
2.2.9.1 REPRESENTACIÓN DEL LENGUAJE DE PROGRAMACIÓN....	99
2.2.10 VELOCIDAD.....	101
2.2.11 CUBIERTA DEL PLC.....	101
2.2.11.1 CARACTERÍSTICAS.....	101
2.2.11.2 INMUNIDAD AL RUIDO.....	103
2.3 EL ENTRENADOR DIDÁCTICO ASSI.....	106
2.3.1 DESCRIPCIÓN GENERAL.....	106
2.3.2 ETAPAS PRINCIPALES.....	107
2.3.3 LÓGICA DE FUNCIONAMIENTO DEL ASSI.....	111
2.3.4 CARACTERÍSTICAS DIDÁCTICAS.....	112
2.4 CONSTITUCIÓN DE LA CPU.....	113
2.4.1 ESTRECTURA DE LA CPU.....	113
2.4.2 ARQUITECTURA BÁSICA DE LA CPU.....	113
2.4.2.1 UNIDAD DE ACOPLAMIENTO Y SINCRONIZACIÓN.....	116
2.4.3 EL MICROPROCESADOR.....	117
2.4.3.1 CPU BASADO EN EL 6502.....	118
2.4.3.2 DIAGRAMA DE PINES DEL 6502.....	119
2.4.3.3 ARQUITECTURA DEL 6502.....	122
2.4.3.4 PROGRAMACIÓN DEL 6502.....	129
2.4.4 ROM PROGRAMABLE Y BORRABLE (EPROM).....	132
2.4.4.1 FUNCIONES DE LOS PINES DE CONTROL.....	133

2.4.4.2 PROGRAMACIÓN DE LA EPROM 2716.....	134
2.4.4.3 LECTURA DE LA EPROM 2716.....	135
2.4.5 INTERFACES.....	135
2.4.5.1 LA PIA 6520.....	136

CAPITULO 3

DISEÑO DE LA CPU DEL ENTRENADOR ASSI.....	140
3.1 MAPA DE MEMORIA DEL ENTRENADOR ASSI.....	140
3.1.1 DECODIFICACIÓN DEL MAPA DE MEMORIA.....	140
3.1.2 DECODIFICACIÓN DE DIRECCIONES.....	145
3.2 RELOJ DEL SISTEMA.....	150
3.3 CONTROL DE ENTRADA/SALIDA DE DATOS DEL ENTRENADOR ASSI.....	155
3.3.1 ESTRUCTURA DE LA PIA.....	155
3.3.2 REGISTROS DE LA PIA.....	155
3.3.3 DIRECCIONAMIENTO DE LOS REGISTROS DE LA PIA.....	157
3.3.4 INICIACIÓN DE LA PIA1.....	158
3.3.5 REGISTROS DE CONTROL DE LA PIA.....	159
3.4 BUFFER.....	162
3.5 MEMORIAS.....	165
3.5.1 MEMORIAS RAM.....	165
3.5.2 MEMORIAS ROM.....	166
3.6 CONTROL DEL VISUALIZADOR.....	168
3.6.1 DISEÑO.....	171
3.6.2 PROGRAMA.....	176
3.6.3 VENTAJAS DEL 2111.....	180
3.7 CONTROL DE TECLADO.....	182

CAPITULO 4

DESCRIPCIÓN Y FUNCIONAMIENTO DEL HARDWARE DEL ENTRENADOR ASSI.....	190
4.1 CIRCUITO DE CONTROL DE TECLADO Y VISUALIZADOR.....	191
4.2 CIRCUITO DE CONTROL DEL ÁREA DE MEMORIA.....	194
4.3 HABILITACIÓN DE LOS PUERTOS DE ENTRADA/SALIDA DEL ENTRENADOR.....	196
4.4 HABILITACIÓN DE LA PIA1 DE COMUNICACIÓN CON EL TECLADO Y EL VISUALIZADOR.....	196
4.5 CIRCUITOS DE TIEMPO.....	198
4.5.1 CIRCUITO DE GENERACIÓN DE LA SEÑAL DE RELOJ.....	198
4.5.2 EL CIRCUITO TEMPORIZADOR.....	198
4.6 DECODIFICADORES.....	202
4.7 EL CONVERTIDOR DIGITAL-ANÁLOGO.....	204
4.8 LISTA DE CIRCUITOS INTEGRADOS DEL ENTRENADOR ASSI.....	205

CAPITULO 5	
PROGRAMA MONITOR.....	207
5.1 RUTINAS Y SUBRUTINAS DEL PROGRAMA MONITOR.....	207
5.1.1 PROCEDIMIENTO INT.....	211
5.1.2 PROCEDIMIENTO EJE.....	214
5.1.3 PROCEDIMIENTO LER.....	216
5.1.4 PROCEDIMIENTO REG.....	218
5.2 EL LENGUAJE DE PROGRAMACIÓN "ELL".....	220
5.3 SIGNIFICADO DE BUFFERS Y REGISTROS.....	223
5.3.1 BUFFERS.....	223
5.3.2 REGISTROS.....	223
5.4 PROGRAMA MONITOR.....	226
CAPITULO 6	
MANUAL DE USUARIO.....	246
6.1 CARACTERÍSTICAS DEL ENTRENADOR.....	247
6.1.1 CARACTERÍSTICAS DE HARDWARE.....	247
6.1.2 CARACTERÍSTICAS DE SOFTWARE.....	247
6.2 COMO UTILIZAR EL ENTRENADOR ASSI.....	248
6.3 COMO ANEXAR UNA INSTRUCCIÓN.....	255
6.3.1 FORMATO DE LA SUBROUTINA.....	258
6.3.2 EJEMPLO: CREANDO LA INSTRUCCIÓN NOT.....	263
6.4 PUNTOS DE PRUEBA.....	
CONCLUSIONES.....	271
BIBLIOGRAFÍA.....	275
APÉNDICE A. EL AMPLIFICADOR OPERACIONAL.....	276
APÉNDICE B. SET DE INSTRUCCIONES DEL MICROPROCESADOR 6502.....	283
APÉNDICE C. DECODIFICADOR SN74LS138.....	286
APÉNDICE D. BUFFER SN74LS244.....	293
APÉNDICE E. DISPLAY HDSP-2111.....	300
APÉNDICE F. TRANSPARENT LATCHES AND EDGE-TRIGGERED FLIP-FLOPS SN74LS373...	317

INDICE DE FIGURAS

FIGURA	No. DE PAGINA
FIGURA 1.1 SISTEMA DE CONTROL	1
FIGURA 1.2 SISTEMA DE CONTROL DE LAZO ABIERTO	1
FIGURA 1.3. DIAGRAMA DE BLOQUES DE UN SISTEMA DE LAZO CERRADO	3
FIGURA 1.4. DIAGRAMA DE BLOQUES AMPLIADO DE UN SISTEMA DE LAZO CERRADO	5
FIGURA 1.5. DIAGRAMA DE LOS BLOQUES Y ETAPAS QUE CONSTITUYEN LOS CIRCUITOS ELECTRONICOS DE CONTROL	9
FIGURA 1.6. ACCION CORRECTORA DE UN CONTROL TODO O NADA.	9
FIGURA 1.7. GRAFICAS DE VALOR MEDIDO VRS. TIEMPO Y ACCION CORRECTORA VRS. TIEMPO.	11
FIGURA 1.8. ZONA DE ACTUACION DEL CONTROL TODO O NADA.	11
FIGURA 1.9. DIAGRAMAS DE TIEMPO QUE MUESTRAN EL EFECTO PRACTICO DE LA ZONA DE ACTUACION.	12
FIGURA 1.10. PORCENTAJE DE ACTUACION DEL DISPOSITIVO VRS. VARIABLE CONTROLADA	14
FIGURA 1.11. BANDA PROPORCIONAL ANGOSTA.	17
FIGURA 1.12. BANDA PROPORCIONAL MEDIANA.	17
FIGURA 1.13. BANDA PROPORCIONAL GRANDE.	18
FIGURA 1.14. COMPORTAMIENTO DEL CONTROL PROPORCIONAL-INTEGRAL.	20
FIGURA 1.15 COMPORTAMIENTO DEL CONTROL PROPORCIONAL INTEGRAL	21
FIGURA 1.16. CIRCUITO DE UN CONTROLADOR PROPORCIONAL INTEGRAL BÁSICO.	21
FIGURA 1.17 COMPORTAMIENTO DEL CONTROL PROPORCIONAL DERIVATIVO	24
FIGURA 1.18. CONTROL PROPORCIONAL DERIVATIVO	25
FIGURA 1.19. CONTROL PROPORCIONAL-INTEGRAL-DERIVATIVO.	26
FIGURA 1.20 GRAFICAS DE LA RESPUESTA EN EL TIEMPO DE UN CONTROLADOR PID.	29
FIGURA 1.20a. GRAFICA TIPICA DE LA VARIABLE VRS. TIEMPO EN UN SISTEMA CON RETARDO DE REACCION.	30
FIGURA 1.20b. GRAFICA VARIABLE VRS. TIEMPO LUEGO DE UN DISTURBIO CON ATRASO DE TRANSPORTE.	31
FIGURA 1.21. RESPUESTA CON ATRASO DE TRANSFERENCIA, CON 10% DE CAMBIO TOTAL.	32
TABLA 1.1. RELACION ENTRE LAS CARACTERISTICAS DEL PROCESO Y EL MODO DE CONTROL.	34
TABLA 1.2. TRANSDUCTORES PASIVOS.	35
TABLA 1.3. TRANSDUCTORES ACTIVOS.	36
FIGURA 1.22. CONVERTIDOR SIMULTANEO A/D.	38
FIGURA 1.23. CONVERTIDOR A/D DE CONTADOR.	40
FIGURA 1.24. FORMA DE ONDA EN LA ENTRADA ANALOGA Y LA SALIDA DEL DIVISOR.	41
FIGURA 1.25. EL CONVERTIDOR CONTINUO A/D	42
FIGURA 1.26. ACCION DE UN CONVERTIDOR A/D CONTINUO.	44
FIGURA 1.27. CONVERTIDOR POR APROXIMACIÓN SUCESIVA A/D.	45
FIGURA 1.28. OPERACIÓN DEL CONVERTIDOR POR APROXIMACIÓN SUCESIVA A/D.	45
FIGURA 1.29. CIRCUITO DE MUESTREO Y MANTENIMIENTO.	47
FIGURA 1.30. DAC CON SALIDA DE VOLTAJE	50
TABLA 1.4. SALIDA DEL DAC	50

FIGURA 1.31. DAC CON SALIDA DE CORRIENTE.	51
FIGURA 1.32. CIRCUITO CON TRIAC Y CONEXION DE LA FUENTE DE VOLTAJE Y LA CARGA; CON CIRCUITO DE DISPARO SIMPLE.	54
FIGURA 1.33. CONEXIÓN DE 2 POLOS EN SERIE.	59
FIGURA 1.36. ACOPLER OPTICOS COMUNES	61
TABLA 2.1. DESARROLLO CRONOLÓGICO DE LOS PLCs.	63
TABLA 2.2. COMPARACION DE SISTEMAS DE CONTROL.	65
FIGURA 2.1. DIAGRAMA DE BLOQUES SIMPLIFICADO DEL PLC.	69
FIGURA 2.2 DIAGRAMA DE LA SECCION PROCESADORA DEL PLC.	70
FIGURA 2.3. DIAGRAMA DE BLOQUE DE LA ROM.	74
TABLA 2.3. DATOS PROGRAMADOS.	75
FIGURA 2.4 DISTRIBUCION COMUN DE UNA OPERACIÓN DE LECTURA EN LA ROM	75
FIGURA 2.5. DIAGRAMA DE BLOQUE DE LA RAM.	78
FIGURA 2.6 DISTRIBUCION COMUN DE LA RAM ESTATICA.	79
FIGURA 2.7. SISTEMA DE ENTRADAS/SALIDAS PARALELAS	83
FIGURA 2.8. DIAGRAMA DE BLOQUES DE SISTEMAS SERIALES DE ENTRADA SALIDA	85
FIGURA 2.9. DIAGRAMA DE FLUJO DE LAS FUNCIONES BASICAS QUE REALIZA COMUNMENTE UN PROGRAMA MONITOR DE PLC.	90
FIGURA 2.10 FUNCION AND	97
FIGURA 2.11. LA FUNCION Y SE DIBUJA EN FORMA DE CONEXIONES EN SERIE.	100
FIGURA 2.12. LOS OPERANDOS SE CONSULTAN EN FILA Y EL RESULTADO SE COMBINA SEGÚN Y.	100
FIGURA 2.13. SUPRESION DE RUIDO.	104
FIGURA 2.14. DIAGRAMA GENERAL DEL PLC "ASSI"	107
FIGURA 2.15. ENTRENADOR DIDACTICO ASSI	108
FIGURA 2.16.FUNCIONES PRINCIPALES DE LA CPU.	110
FIGURA 2.17. LOGICA DE FUNCIONAMIENTO DEL ASSI	111
FIGURA 2.18. ARQUITECTURA BASICA DE LA CPU.	114
FIGURA 2.19. UNIDAD DE ACOPLAMIENTO.	117
FIGURA 2.20. CONFIGURACIÓN DEL SISTEMA BÁSICO DE CPU BASADO EN EL 6502	118
FIGURA 2.21. DIAGRAMA DE PINES DEL 6502	120
TABLA 2.4. NOMBRES Y FUNCIONES DE LOS PINES.	120
FIGURA 2.22. REGISTROS INTERNOS DEL 6502.	126
FIGURA 2.23 FORMAS DE ONDA DE DOS FASES.	127
FIGURA 2.24. FORMATO DE UNA INSTRUCCIÓN.	129
FIGURA 2.25. PASTILLA DE MEMORIA EPROM.	132
FIGURA 2.27 SIMBOLO DE LA EPROM 2716.	133
TABLA 2.5. FUNCION DE LOS PINES DE LA EPROM	133
FIGURA 2.27. DIAGRAMA DE TIEMPO DE PROGRAMACION DE LA EPROM 2716.	134
FIGURA 2.28. DIAGRAMA DE TIEMPO DE LECTURA DE LA EPROM.	135
FIGURA 2.29. DIAGRAMA DE PINES DE LA PIA 6520	137
TABLA 2.6. FUNCION DE LOS PINES DE LA PIA 6520	137
TABLA 2.6. FUNCION DE LOS PINES DE LA PIA 6520	139
FIGURA 3.1. MAPA DE MEMORIA	142
TABLA 3.1 DIRECCIONES DEL MAPA DE MEMORIA	143
FIGURA 3.2. SIMBOLO LOGICO DEL 74LS138	146
FIGURA 3.3. DIAGRAMA LOGICO DEL DECODIFICADOR 74LS138	146
TABLA 3.2. TABLA DE HABILITACION	147

TABLA 3.3. TABLA DE VERDAD DE 74LS138	147
FIGURA 3.4. CIRCUITOS DECODIFICADORES DE ENTRENADOR ASSI.	148
FIGURA 3.5. RELOJ DEL SISTEMA	150
FIGURA 3.6. SEÑALES DE DIRECCION Y DATOS.	151
FIGURA 3.7. SEÑALES DE LECTURA Y ESCRITURA.	152
FIGURA 3.8. RELACION DE TIEMPO DURANTE FASE 2	153
FIGURA 3.9. CIRCUITO DE RELOJ	154
FIGURA 3.10. CONDICION PARA LA SEÑAL RW	154
FIGURA 3.11. SECCIONES DE LA PIA.	155
FIGURA 3.12. PUERTOS A Y B DE LA PIA.	156
FIGURA 3.13. REGISTROS DE PIA Y SUS DIRECCIONES	158
TABLA 3.5. DIRECCIONAMIENTO DE LOS REGISTROS INTERNOS DE LA PIA	158
TABLA 3.6. ORGANIZACIÓN DE LOS REGISTROS DE CONTROL DE LA PIA	160
FIGURA 3.15. IC 74LS244	162
FIGURA 3.16. BUFFER OCTAL, MANEJADOR DE LINEA, NO INVERSOR CON SAIDA DE RES ESTADOS	163
FIGURA 3.17. DIAGRAMA LOGICO DEL 74LS244	163
FIGURA 3.18. SIMBOLO Y TABLA DE VERDAD DEL 74LS244	164
FIGURA 3.19. DIAGRAMA DE PINES DE LA 2114.	165
FIGURA 3.20. DIAGRAMA ESQUEMATICO DE MEMORIAS	167
TABLA 3.7. NIVELES LOGICOS	168
TABLA 3.8 a. SEÑALES DE CONTROL	169
TABLA 3.8 b. DIRECCIONAMIENTO DE RAM CARÁCTER	169
TABLA 3.8 c. FORMATO DE LOS DATOS DE LA RAM DE CARACTERES	169
TABLA 3.8 d. NIVELES LOGICOS PARA ACCESAR A LA CHARACTER RAM	169
TABLA 3.9. NIVELES LÓGICOS PARA ACCESAR A LA PALABRA DE CONTROL	171
TABLA 3.10. MODO DE OPERACIÓN	172
FIGURA 3.21. CIRCUITO DE CONTROL DEL VISUALIZADOR	172
TABLA 3.11. INFORMACION DE LA PALABRA DE CONTROL A.	173
TABLA 3.12. CONFIGURACION DE REGISTROS DE CONTROL.	173
TABLA 3.13. INFORMACION DE LA PALABRA DE CONTROL B.	174
TABLA 3.14. CONFIGURACION DE LA PALABRA DE CONTROL B.	174
TABLA 3.15. FORMATO PARA EL DATO DE LA CHARACTER RAM	175
TABLA 3.16. CODIGOS BINARIOS DE CARACTERES ASCII	175
FIGURA 3.22. FLUJOGRAMA CONTROL DEL VISUALIZADOR	177
TABLA 3.17. DIGITO SELECCIONADO	177
FIGURA 3.23. FLUJOGRAMA DE TECLADO	183
FIGURA 3.24. TECLADO HEXADECIMAL	184
FIGURA 3.25. DETECCION DE TECLA.	185
TABLA 3.18. TABLA DE DATOS 2 O "TABLA 2"	186
FIGURA 3.26. CIRCUITO DE CONTROL DE TECLADO	189
FIGURA 4.1 CONTROL DE TECLADO Y VISUALIZADOR	193
FIGURA 4.2 SELECCIÓN Y CONTROL DE MEMORIAS	195
FIGURA 4.3 CONTROL DE PIA-1 Y PIA-2	197
FIGURA 4.4 RELOJ DEL SISTEMA	199
FIGURA 4.5 CIRCUITO TEMPORIZADOR	200
FIGURA 4.6. SEÑALES DEL TEMPORIZADOR	201
FIGURA 4.7. DECODIFICADORES DEL ASSI	203

TABLA 4.1 SALIDA DEL DAC	205
FIGURA 4.7. CONVERTIDOR DIGITAL-ANÁLOGO	205
FIGURA 5.1. SUBRRUTINA DEL PROGRAMA MONITOR	210
FIGURA 5.2. SUBRRUTINA INT	213
FIGURA 5.3 PROCEDIMIENTO EJE.	215
FIGURA 5.4 PROCEDIMIENTO LER	217
FIGURA 5.5. PROCEDIMIENTO REG	219
TABLA 5.1 INSTRUCCIONES DEL ELL	220
TABLA 5.2. TABLA 7 DEL PROGRAMA MONITOR	226
TABLA 5.3 LISTA DE SUBRRUTINS DEL PROGRMA MONITOR	245
FIGURA 6.1 INTERPRETACION Y EJECUCION DE UNA INSTRUCCIÓN	256
TABLA 6.1 DIRECCIONES DE LAS SUBRRUTINAS	257

PROLOGO

I. GENERALIDADES

En los últimos treinta años el equipo electrónico de medición y control ha evolucionado en respuesta a requisitos cada vez más específicos destinados a mejorar la eficiencia y productividad en la industria. Las funciones primordiales que cubre un equipo de control son medir, indicar y controlar variables tales como caudal, presión, temperatura, nivel, composición, y otras. Para cumplir estas funciones se crearon los Controladores Lógicos programables "PLC"; dichos equipos pueden manejar desde una única variable hasta centenares de variables de una planta industrial completa.

El PLC consiste en una computadora industrial; es decir, una computadora creada para trabajar en ambientes de manufactura, a diferencia de otras computadoras que han sido diseñadas para trabajar en ambiente de oficina. Por lo tanto como cualquier computadora, el PLC requiere de la intervención humana para su programación; e independientemente del nivel de automatización, se deben considerar cuidadosamente el grado y los modos de capacitación del personal que lo va a programar. Y ya que el funcionamiento interno de cualquier equipo proporciona al usuario la ventaja de explotar al máximo las funciones que éste es capaz de realizar, se advierte la necesidad de un equipo didáctico adecuado que permita capacitar al usuario del PLC, no sólo desde el punto de vista del manejo externo, sino, tomando en cuenta, como factor importante, su estructura interna.

El diseño de un PLC como equipo didáctico debe cumplir con los siguientes lineamientos pedagógicos:

- Realizar las funciones básicas de un autómata programable.
- Permitir el estudio de la arquitectura interna del PLC.
- Demostrar como son procesadas las señales internamente.
- Proporcionar el correspondiente manual de usuario.

En la Universidad Don Bosco se cuenta con Controladores Lógicos Programables para aplicación industrial. Dicha institución hace uso de este equipo para la capacitación de personal en el área de automatización. Pero basándose en que es importante que cada uno de los lineamientos mencionados se cumplan completamente en un equipo didáctico para automatización industrial, es obvio, que la UDB necesita tanto de PLCs diseñados con fines industriales, como PLCs diseñados con fines pedagógicos, para lograr una completa capacitación en esta área.

II. ANTECEDENTES

Los computadores diseñados para el control de procesos son los autómatas programables o Controladores Lógicos Programables (PLC). Éstos, son controladores electrónicos cuyas funciones se almacenan como programas en una sección de control lógico llamada Unidad de Control y Procesamiento (CPU). Además poseen módulos de entrada salida que interconectan una variedad de sensores analógicos y digitales, a través de los cuales el PLC controla el funcionamiento de diferentes máquinas y procesos.

Los predecesores de los Autómatas Programables fueron los “controladores por conexiones”, cuyas funciones se realizaban cableando módulos lógicos. Los PLCs típicamente reemplazaron a los relés, temporizadores, contadores, contactores, etc., proveyendo además de funciones adicionales, como substracción, división y multiplicación. Actualmente se ofrecen en el mercado grandes PLCs que poseen Control Proporcional Integral Derivativo, o Control PID, que es un tipo de control de lazo matemático, el cual utiliza señales analógicas como entrada. El PLC requiere menos espacio y energía que un panel de relés y puede ser programado o reprogramado para diferentes operaciones.

Una de las empresas más reconocidas en cuanto a la producción de PLCs es SIEMENS. Cuyos primeros automatizadores programables fueron los SIMATIC S3, que se ofrecieron en 1975. En los años 1978/1979 fueron sustituidos por los SIMATIC S5, en los que se aplicaron por primera vez microprocesadores.

Luego se creó la serie S5”U”, estos aparatos son más robustos y compactos. Ahora se ha perfeccionado el sistema de automatización de SIMATIC. El resultado es la línea de aparatos S7 que poseen un módulo central con volumen de funciones incrementado, que puede contener además del procesador central, la memoria de programación y las interfaces de los aparatos de programación. Esto ha sido posible por la aplicación de procesadores estándar y otros componentes de alta integración (como por ejemplo memorias), con lo cual se ha conseguido una disposición mecánica más compacta, y se ha mejorado el rendimiento.

Ahora bien, ya se ha mencionado que debido a la importancia de los PLC en el área industrial, es necesario contar con el adecuado equipo para la capacitación del personal y los estudiantes de las diferentes ramas de la Ingeniería relacionadas con el

área. Actualmente en la Universidad Don Bosco se cuenta con PLC industriales los cuales se utilizan como entrenadores destinados al estudio de la programación y correcta utilización de los mismos. Los PLC utilizados como entrenadores en la UDB son precisamente los de la familia SIMATIC S5 de SIEMENS, más específicamente el S5-90U, S5-95U, S5-100U y S5-115U. El procesador central de estos PLC se programa en lenguaje de programación STEP 5, que consiste en una serie de instrucciones destinadas a resolver las distintas tareas de automatización, utilizando una interfaz que conecta al PLC con la computadora que contiene dicho lenguaje. Pero debido a que éste es un equipo diseñado para la industria y no con fines didácticos, no es posible estudiar las etapas internas que forman un PLC, ni la manera en que éste procesa, según su estructura lógica interna, las señales que recibe y genera durante el control de un proceso; por lo que se pretende con el desarrollo de este trabajo diseñar e implementar un equipo didáctico que muestre el funcionamiento interno de las diferentes etapas que constituyen un Controlador Lógico Programable.

III. ALCANCES

El alcance proyectado para este trabajo consiste en realizar los estudios y análisis técnicos necesarios para diseñar e implementar un equipo didáctico para un Controlador Lógico Programable. Para esto deberán cubrirse las siguientes áreas:

- En el área de microprocesadores se abarcarán los siguientes puntos:
 - Arquitectura Interna.
 - Programación del microprocesador.

Ambos enfocados al diseño de la Unidad de Control y Procesamiento (CPU).

Estudio de los circuitos digitales para realizar el diseño de las interfaces.

- Estudio de circuitos analógicos y de potencia para el diseño de los módulos de E/S.
- Estudio de la estructura interna del PLC.
- Realización del manual de usuario.

IV. LIMITACIONES

La realización de este trabajo, más específicamente, la implementación del entrenador, estará limitada hasta cierto punto, por la adquisición de los elementos electrónicos, ya que algunos deberán ser traídos desde el exterior del país.

Por lo tanto, debido a que no es posible estimar con exactitud el tiempo que consumirá dicha actividad, este factor podría verse alterado por la razón antes descrita.

INTRODUCCIÓN GENERAL

Muchas industrias se han visto en la necesidad de automatizar sus procesos de producción para obtener una mayor eficiencia y mejor calidad en sus productos. Para lograr esto, han introducido en sus maquinarias equipos electrónicos capaces de controlar de forma específica las variables más importantes. Estos equipos electrónicos que miden las variables, las procesan, toman una decisión y luego dan una respuesta al sistema para informar como será tratada la variable a controlar, son los Autómatas Programables, también denominados Controladores Lógicos Programables.

El Controlador Lógico Programable (PLC), es entonces un dispositivo electrónico capaz de controlar las variables de un determinado proceso.

Este proyecto consiste en realizar el diseño e implementación de un equipo didáctico de Controlador Lógico Programable (PLC) para la Universidad Don Bosco, denominado "Entrenador ASSI".

Debido su enfoque didáctico, el objetivo principal que se persigue con el desarrollo de este trabajo es dar a conocer el funcionamiento y arquitectura interna básica de los Autómatas Programables.

Los dos primeros capítulos son el resultado de un proceso de investigación y documentación que sienta las bases para el desarrollo y entendimiento del diseño del entrenador ASSI y de los PLCs en general. En éstos se estudia la teoría básica del control de procesos y de los elementos principales en la organización estructural de un PLC.

En el capítulo 1 se describen los *modos básicos* utilizado en el control industrial, denominados "Modo de Control de Lazo Abierto" y "Modo de Control de Lazo Cerrado". El capítulo inicia con una comparación entre ambos modos de control, luego se estudian las etapas que constituyen un sistema de control de lazo cerrado y los cinco diferentes modos en que se clasifica:

- Modo de Control Todo o Nada.
- Modo de Control Proporcional (P).
- Modo de Control Proporcional Integral (PI).
- Modo de Control Proporcional Derivativo (PD).
- Modo de Control Proporcional Integral Derivativo (PID).

Se explica en forma detallada, el funcionamiento, las características particulares y los diagramas que describen el comportamiento de cada modo de control. Luego se describen los tipos de circuitos básicos y sus respectivos análisis matemáticos, además los elementos de acople con el mundo exterior, utilizados en el control electrónico de sistemas. Estos son:

- Transductores.
- Convertidores Análogo/Digital y Digital/Análogo.
- Circuitos de aislamiento eléctrico.
- Elementos de potencia: Tiristores y Relés.

El capítulo 2 se dedica de forma específica al estudio de los Controladores Lógicos Programables; desarrollando los siguientes aspectos fundamentales:

- Características funcionales.
- Estructura.
- Sección Procesadora.
- Programación.

Finalmente y siempre dentro de este capítulo, se presenta una descripción general del Entrenador Didáctico ASSI, sus etapas principales, la lógica de funcionamiento y características didácticas en las que se basa su diseño. Se explica la estructura de la Unidad de Procesamiento Y Control (CPU), considerada el "Cerebro del Entrenador ASSI" cuyo componente principal es el microprocesador 6502; por lo que las últimas tres secciones están dedicadas al estudio de su funcionamiento, circuitería de apoyo, memorias y circuitos de interfaz, que se ocupan en el diseño del Entrenador.

Luego de estos dos capítulos, que trataban la teoría general de los Autómatas, se dedican los siguientes capítulos al estudio específico del entrenador ASSI.

El capítulo 3 denominado "Diseño de la CPU del entrenador ASSI", se dedica de forma específica al diseño de la Unidad de Procesamiento y Control (CPU), considerada el "Cerebro del entrenador ASSI", cuyo componente principal es el microprocesador 6502.

El capítulo inicia con una descripción general de la forma en que está estructurada la CPU del Entrenador ASSI. Luego se desarrollan de forma específica cada una de las etapas de la CPU; iniciando con un estudio de la distribución del mapa de memoria.

Dichas etapas son:

- Reloj del Sistema.
- Control y programación de la PIA.

- Memorias.
- Buffers.
- Control del visualizador, y
- Control del teclado.

Se incluyen las rutinas de inicialización de la PIA, de control del visualizador y de control del teclado.

En el capítulo 4, denominado "Descripción y funcionamiento del hardware del entrenador ASSI", se presentan los diagramas que describen completamente la circuitería del Entrenador. Para facilitar la comprensión del funcionamiento de ésta, se divide su estudio por etapas, separando los circuitos según su función específica dentro del Entrenador ASSI.

Luego el capítulo 5, denominado "Programa Monitor", comprende el estudio de las rutinas y subrutinas del programa monitor, y del lenguaje de programación creado para el Entrenador ASSI. Como complemento se presenta en este capítulo el *Programa Monitor* completo.

Finalmente en el capítulo 6, y tal como se menciona en el prefacio, se presenta el "Manual de Usuario", que comprende los siguientes puntos:

- Características del Entrenador.
- Como utilizar el Entrenador ASSI.
- Como anexar una instrucción.
- Puntos de Prueba.

Se espera que la información técnica y científica presentada, sirva como guía para la comprensión de la teoría básica de los circuitos y sistemas de control electrónico y su relación directa con los Automatas Programables. Además, que permita la comprensión del diseño, implementación y funcionamiento del Entrenador Didáctico ASSI.

Los Autores.

CAPITULO 1

SISTEMAS Y CIRCUITOS ELECTRONICOS DE CONTROL

La figura 1.1 describe la función básica de un sistema de control de procesos. Su función consiste en controlar un proceso, basándose en ciertos datos de entrada predeterminados, denominados referencia, para obtener el valor deseado a la salida.



FIGURA 1.1 SISTEMA DE CONTROL

Aunque en esta primera aproximación parece sencilla la aplicación de un sistema de control, deben considerarse otros factores que pueden provocar que el control no funcione de la forma deseada, a estos factores se les conoce como *disturbios*.

Los *disturbios* son los factores externos que afectan la salida del proceso, sobre las que el controlador no posee efecto alguno. (Ver figura 1.2).

Por lo tanto cuando se controla un proceso debe considerarse si es necesario o no, tomar en cuenta, el efecto que estos disturbios provocarán en el proceso.

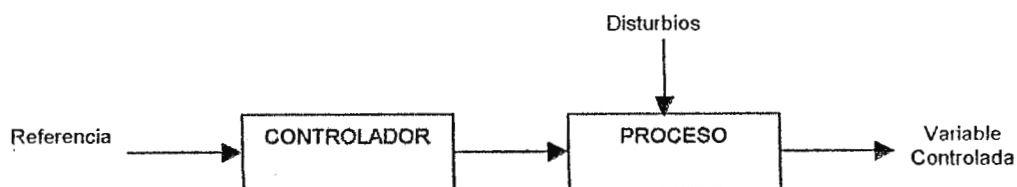


FIGURA 1.2 SISTEMA DE CONTROL DE LAZO ABIERTO

Como los disturbios están fuera de la acción controladora, para corregir el error que estos provocan es necesario comparar la salida deseada con la salida obtenida. Esta

comparación se lleva a cabo realimentando la salida hasta la entrada para compararla con un valor de referencia preestablecido, y obtener así la diferencia.

A partir de estos conceptos básicos es posible clasificar el control de procesos de la siguiente forma:

- **Sistemas de control de Lazo Abierto.** (Figura 1.2). Son los sistemas en los que la salida no es tomada en cuenta por el controlador; es decir que no es *retroalimentada*. En estos sistemas cada entrada de referencia corresponde a una condición de operación fija, por lo tanto la precisión en el sistema depende de la exactitud en la calibración del bloque controlador.
- **Sistemas de control de Lazo Cerrado.** (Figura 1.3). Es el control que se basa en la retroalimentación de la señal de salida hacia la entrada para compararla con un valor de referencia y basado en esta comparación generar una acción correctora. Estos sistemas poseen entonces la capacidad de corregir automáticamente cualquier diferencia entre el valor obtenido y el valor deseado, indiferentemente del disturbio que provoque la diferencia.

A la hora de decidir que tipo de control se aplicará aun sistema específico deben tomarse en cuenta las ventajas y desventajas de cada tipo de control. El *Sistema de Lazo Cerrado* posee la ventaja de ser prácticamente insensible a perturbaciones externas; pero el hecho de poseer un lazo de realimentación, significa que requiere más componentes que un sistema de Lazo Abierto lo cual lo vuelve más costoso. Además, como este tipo de sistema de control esta siempre autocorrigiendose, puede volverse inestable debido a que puede darse una sobrecorrección que haga oscilar al sistema. Por otro lado el *Sistema de Lazo Abierto* es más estable y menos costoso. Por lo tanto puede concluirse

que para sistemas cuyas entradas conocidas y que no existen perturbaciones es preferible aplicar un *Control de Lazo Abierto*. En cambio cuando en el sistema se presentan perturbaciones impredecibles se prefiere el *Control de Lazo Cerrado*.

1.1 SISTEMAS DE CONTROL DE LAZO CERRADO

Sistemas de Lazo Cerrado. (Autocorrector).

Los sistemas que poseen la habilidad de comparar el valor real de la variable que se está controlando con el valor deseado, y basándose en esta comparación son capaces de ejecutar una acción, son llamados "Sistemas de Lazo Cerrado". Este tipo de sistema se representa según el diagrama de bloques de la figura 1.3, el cual muestra el valor de la variable controlada siendo realimentada y comparada con el valor deseado (referencia).

Los Sistemas de Lazo Cerrado son capaces de tomar automáticamente una acción para corregir cualquier diferencia entre el valor real y el deseado, indistintamente de los factores que provoquen esta diferencia.

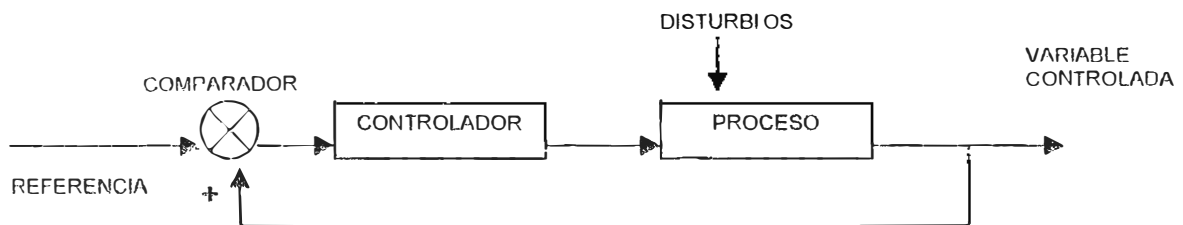


FIGURA 1.3. DIAGRAMA DE BLOQUES DE UN SISTEMA DE LAZO CERRADO

Diagrama de Bloques.

Para describir más adecuadamente los sistemas de lazo cerrado se utiliza el diagrama mostrado en la figura 1.4. En este diagrama la variable de proceso¹ que se está controlando se mide y envía a un comparador², para establecer la diferencia entre el valor medido de la variable y el valor de referencia. El comparador genera una señal de error, que representa dicha diferencia.

De forma más específica la señal de error es igual al valor medido menos el valor deseado, y se considera positiva si el valor medido es mayor que el valor de referencia; y negativa si es menor. El valor y polaridad de la señal de error se obtiene mediante la siguiente ecuación:

$$ERROR = VALOR MEDIDO - VALOR DE REFERENCIA.$$

Luego esta señal de error es recibida por el controlador², el cual a su vez genera una señal de salida que generalmente es ampliada para desarrollar la potencia suficiente para operar el dispositivo corrector final.

La relación entre la señal de salida del controlador y la señal de error depende del diseño y ajuste del controlador.

Si la señal de salida del controlador no posee la potencia necesaria para operar el dispositivo corrector final, puede ser necesario una amplificación. El dispositivo corrector final, puede ser por ejemplo un tiristor o un relé³, dependiendo del tipo de control que se ejecuta.

¹ Variable de Proceso: se refiere a temperatura, presión, velocidad de flujo de un fluido, concentración química, humedad, posición mecánica, velocidad mecánica, etc.

² El comparador y el controlador pueden ser mecánicos, eléctricos, neumáticos, etc.

³ Los tiristores y relés son descritos en la sección 1.6

Existen cinco diferentes modos de controlar un sistema, estos no dependen de si el controlador es eléctrico, mecánico o neumático, sino de que tan drásticamente y de qué manera el controlador reacciona ante la señal de error. De forma más específica, depende de la relación matemática entre la salida y la entrada del controlador (siendo la entrada la señal de error). Más adelante se describe cada modo de control.

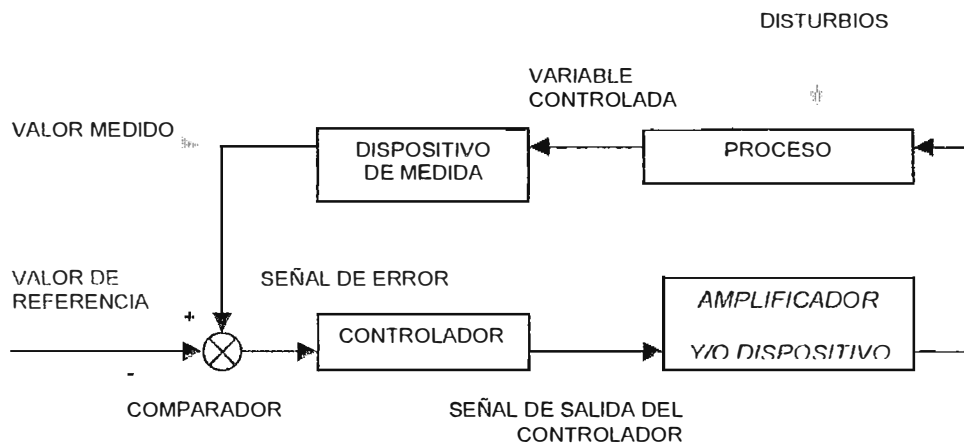


FIGURA 1.4. DIAGRAMA DE BLOQUES AMPLIADO DE UN SISTEMA DE LAZO CERRADO

Características de un Sistema de Lazo Cerrado.

➤ Desbalance:

La diferencia entre el valor medido y el valor de referencia que el sistema permite (que no puede corregir) usualmente se conoce como desbalance; por lo tanto un buen sistema tiene un desbalance bajo, es decir que reduce a cero o casi a cero la señal de error.

➤ Velocidad de Respuesta:

Otra característica es la velocidad de respuesta. Si ocurren circunstancias que saquen el valor medido de concordancia con el valor de referencia, un buen sistema debe restaurar rápidamente esta concordancia.

Oscilación y Estabilidad.

Es posible diseñar sistemas con desbalance bajo y velocidad alta de respuesta, pero a veces tienden a ser inestables, esto significa que el sistema produce variaciones violentas en el valor de la variable controlada a medida que busca el valor apropiado de salida del controlador. Esto se debe a que el sistema sobreacciona a un error, de este modo produce un gran error en la dirección opuesta. Trata entonces de corregir el nuevo error y nuevamente sobreacciona yéndose a la otra dirección. Cuando esto sucede, se dice que el sistema está oscilando. La oscilación se va amortiguando hasta que el sistema establece el valor correcto de la variable controlada, pero mientras esto ocurre, el proceso ha estado fuera de control, lo cual puede resultar en malas consecuencias.

A menor cambio en las oscilaciones de la variable controlada, más estable es el sistema. Un buen sistema de control es estable.

1.1.1 CIRCUITOS ELECTRÓNICOS DE LAZO CERRADO.

Los circuitos electrónicos de lazo cerrado para el control de procesos industriales pueden dividirse en tres bloques principales⁴:

- a. Bloque de entrada.
- b. Bloque de control.
- c. Bloque de salida.

a. Bloque de entrada:

Se encarga de la adquisición de datos. Está formado por los dispositivos que recogen la información que proviene del sistema y del operador. Este bloque es el

⁴ Los circuitos básicos que componen cada uno de estos bloques se desarrollan a lo largo del capítulo

responsable de convertir las señales de entrada a los niveles lógicos que el bloque de control puede interpretar y procesar.

El bloque de entrada esta compuesto por tres etapas principales:

- ◆ **Etapla Transductora:** Encargada de sensar las señales necesarias para controlar el sistema. Formada por elementos denominados *Transductores*.
- ◆ **Etapla de aislamiento eléctrico:** Que protege de sobrevoltajes, sobrecorrientes, ruido eléctrico, etc., que pueden dañar al resto del circuito de control. Una forma de aislamiento eléctrico es utilizar *Acoples Ópticos*. (Ver sec. 1.7).
- ◆ **Etapla convertidora de análogo a digital:** Generalmente la señal proveniente de los transductores es análoga, por lo que debe convertirse a los niveles lógicos requeridos por la sección controladora. Los circuitos que realizan esta función se denominan *Convertidores Análogo/Digital (ADC)*. (Ver sec. 1.3).

b. Bloque de control.

Encargado de la toma de decisiones basándose en la información aportada por el bloque de entrada: para enviar al bloque de salida las ordenes necesarias para controlar el sistema.

A la forma en que el bloque de control desarrolla su función se denomina *Modo de control*, el cual, depende de la manera en que este bloque reacciona ante la señal de error. (Ver sec. 1.1.2).

c. Bloque de salida.

Esta formado por los circuitos que toman las señales de salida del bloque de control, y las convierten a señales analógicas o digitales compatibles con el sistema que se controla.

Las etapas principales que puede requerir una etapa de salida son las siguientes:

- ◆ **Etapa convertidora de digital a análogo:** Las señales provenientes del bloque de control consisten en niveles lógicos digitales, por lo que a veces es necesario convertir los niveles lógicos a señales analógicas. Los circuitos que realizan esta función se denominan *Convertidores Digital/Análogo (DAC)*. (Ver sec. 1.5).
- ◆ **Etapa de aislamiento eléctrico:** Protege de sobrevoltajes, sobrecorrientes, ruido eléctrico, etc., que pueden dañar al resto del circuito de control. Una forma de aislamiento eléctrico es utilizar *Acoples Ópticos*, de forma similar que en el bloque de entrada. (Ver sec. 1.7).
- ◆ **Etapa de potencia:** El tipo de control requerido para un sistema puede ser de dos tipos:
 1. de variación continua, "*Control Proporcional*".
 2. de dos estados, "*Control Todo o Nada*".

Cada uno de estos modos de control se explica más adelante, pero el tipo de etapa de potencia depende del modo que desarrolle el bloque de control.

La etapa de potencia del Control Proporcional puede ser desarrollada cambiando continuamente el ángulo de disparo de un tiristor, (cambio proporcional). En este caso se requiere de un DAC que convierta la señal digital del controlador en la señal análoga que variará de forma proporcional.(Ver sec. 1.6.1).

La etapa de potencia del Control Todo o Nada puede desarrollarse utilizando algún tipo de interruptor, contactor, o relé. En este caso no es necesaria la etapa convertidora DAC, basta con amplificar la señal lógica al nivel requerido para activar los dispositivos mencionados. (Ver sec. 1.6.2).

La etapa de aislamiento eléctrica sí es requerida en ambos tipos de etapa de potencia. En la figura 1.5 se presenta un diagrama de bloques que describe a los sistemas electrónicos de control:

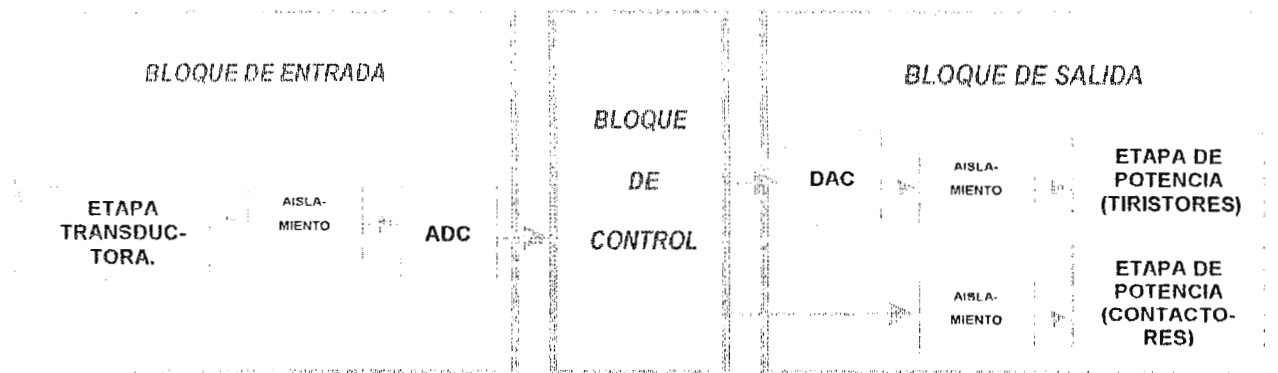


FIGURA 1.5. DIAGRAMA DE LOS BLOQUES Y ETAPAS QUE CONSTITUYEN LOS CIRCUITOS ELECTRONICOS DE CONTROL.

1.1.2 MODOS DE CONTROL DE LAZO CERRADO

Los cinco modos de control básicos en los sistemas industriales de lazo cerrado son: a. Todo o nada. b. Proporcional (P). c. Proporcional-Integral (PI). d. Proporcional-Derivativo (PD). e. Proporcional-Integral-Derivativo (PID). Independientemente de la naturaleza de los controladores; es decir, sin importar si son eléctricos, neumáticos, hidráulicos, o mecánicos; los principios involucrados son los mismos, aunque los métodos de implementación son completamente diferentes. A continuación se describe cada uno de estos sistemas.

1.1.2.1 CONTROL TODO O NADA.

En este modo se dispone solamente de dos posiciones o estados de operación en el dispositivo corrector final, por lo que también es conocido como "control de dos posiciones" o "control bang bang".

Si la señal de error es positiva, el controlador envía el dispositivo corrector final a una de las dos posiciones. Si la señal de error es negativa el controlador envía el dispositivo corrector final a la otra dirección. En la figura 1.6 se muestra el gráfico de la acción correctora para un control Todo o Nada ideal. La figura 1.7a muestra una gráfica típica de valor medido de la variable controlada versus tiempo, y en la figura 1.7b la acción correctora dibujada contra el mismo eje de tiempo. Se observa que el valor real de la variable controlada tiende a oscilar alrededor del valor de referencia. Esta es una característica universal del control Todo o Nada. El sobrepaso sucede porque el proceso no puede responder instantáneamente al cambio de la posición del dispositivo corrector final. Un buen sistema debe diseñarse de forma que mantenga una pequeña amplitud de las oscilaciones, pero esto tiende a producir una ocurrencia más frecuente, con lo que se agrava la otra desventaja del control Todo o Nada, que consiste en el desgaste del dispositivo corrector final producido por la frecuente operación.

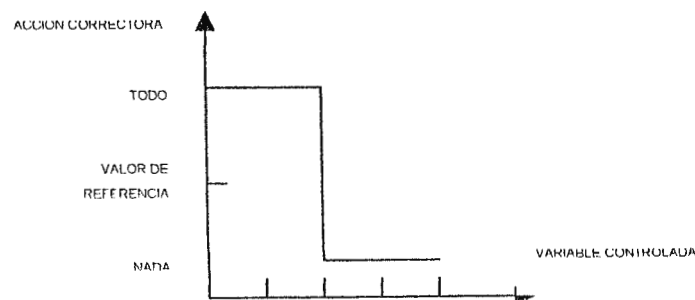


FIGURA 1.6. ACCION CORRECTORA DE UN CONTROL TODO O NADA.

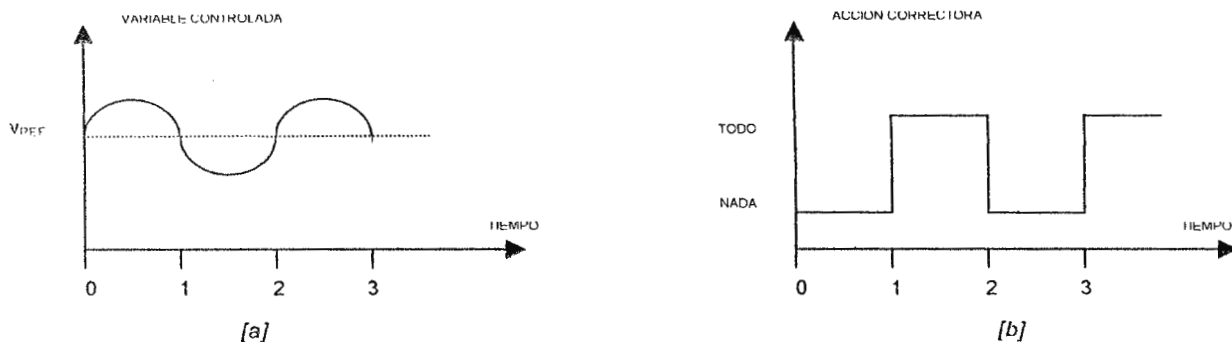


FIGURA 1.7. GRAFICAS DE VALOR MEDIDO VRS. TIEMPO Y ACCION CORRECTORA VRS. TIEMPO.

Zona de Actuación.

El comportamiento del control Todo o Nada que se presenta en las figuras 1.6 y 1.5 es ideal. En realidad, para lograr que un dispositivo vaya de una posición a otra es necesario atravesar cierto rango de valores tal como se muestra en la figura 1.8. Al menor rango de valores necesario, que se debe atravesar se denomina "Zona de Actuación".

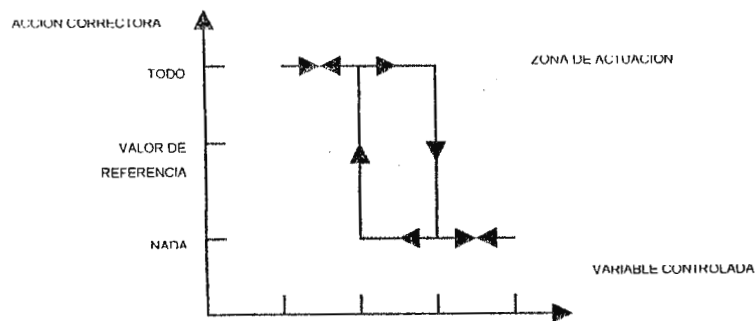


FIGURA 1.8. ZONA DE ACTUACION DEL CONTROL TODO O NADA.

La zona de actuación es una característica propia del control Todo o Nada en el que el valor medido debe pasar por encima del valor de referencia para que la señal de error alcance un valor positivo. Y de igual forma, para que la señal de error alcance un

valor negativo es necesario que el valor medido caiga por debajo del nivel de referencia; obteniendo así las dos posiciones del control Todo o Nada. El comportamiento de la variable se muestra en la figura 1.9a. En la figura 1.9b se presenta el gráfico de tiempo correspondiente, en donde el desfase entre la acción correctora y el cambio de la variable controlada se debe precisamente al efecto de la "Zona de Actuación".

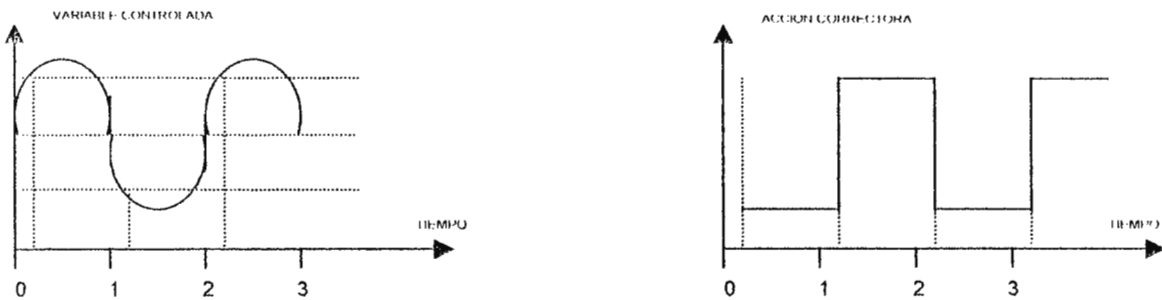


FIGURA 1.9. DIAGRAMAS DE TIEMPO QUE MUESTRAN EL EFECTO PRACTICO DE LA ZONA DE ACTUACION.

La Zona de Actuación presenta la desventaja de generar una oscilación mayor, pero a la vez ofrece una menor frecuencia de oscilación, con lo que se reduce el desgaste del dispositivo corrector, siendo esto una ventaja.

Generalmente la Zona de Actuación se presenta como un porcentaje de la plena escala:

$$\% \text{ ZONA DE ACTUACION} = \frac{(\text{VALOR}+) - (\text{VALOR}-)}{\text{Valor de Plena Escala}} \times 100\%$$

Muchos controladores Todo o Nada ofrecen una Zona de Actuación ajustable para que el usuario seleccione la cantidad según sus requerimientos.

En el modo de control Todo o Nada la acción correctora es decir la posición del dispositivo corrector final es la variable dependiente y la señal de error es la variable independiente.

1.1.2.2 CONTROL PROPORCIONAL (P).

A diferencia del control Todo o Nada en el que el dispositivo corrector final es forzado a tomar toda o ninguna posición; en el control Proporcional se tiene un rango de posiciones continuas posibles de respuesta. Se llama control Proporcional porque la posición que se genera es proporcional a la señal de error que la provoca.

El control proporcional se caracteriza porque la salida del bloque controlador es proporcional a la entrada.

En la figura 1.10 se presenta una gráfica del porcentaje de acción correctora (posición del dispositivo corrector final) versus el valor medido de la variable controlada. Si el valor medido de la variable controlada es demasiado pequeño se requiere de una acción correctora grande para que la variable medida alcance el valor de referencia deseado, y por el contrario, si el valor medido es muy grande la acción correctora debe ser casi o totalmente nula para que la variable controlada regrese al valor de referencia requerido. Pero, como el control es Proporcional, la respuesta será proporcional, es decir, que a medida que el valor medido sobrepase el valor de referencia, o, a medida que el valor medido disminuye alejándose del valor de referencia, se requiere una acción correctora mayor. Por lo tanto este tipo de control responde no solo al hecho de que el valor medido sea bajo o alto; sino también a la cantidad de error, es decir, que tan bajo o que tan alto sea. A mayor error, más drástica es la acción correctora. Esta es la diferencia esencial entre el control Proporcional y el control Todo o Nada. Entonces *Proporcional*

significa que la cantidad de corrección introducida está en proporción a la cantidad de error. En otras palabras, un determinado cambio porcentual en el error, ocasiona un correspondiente cambio porcentual en la posición de la válvula.

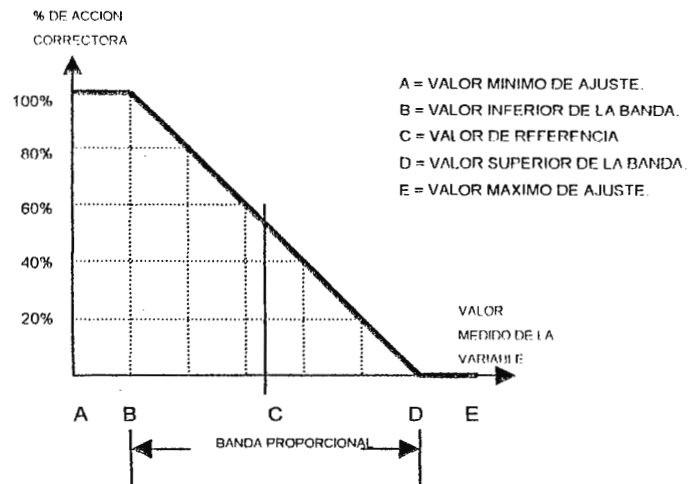


FIGURA 1.10. PORCENTAJE DE ACTUACION DEL DISPOSITIVO VRS. VARIABLE CONTROLADA

Banda Proporcional de Control.

En la figura 1.10 se observa que el control es variable entre los valores B y D. Cualquier valor de la variable controlada que se encuentre entre los puntos A y B genera una acción correctora total (100%), y cualquier valor de la variable controlada entre D y E genera una acción correctora nula (0%). Entonces al rango en que el control Proporcional puede generar una respuesta variable se le denomina "Banda Proporcional de Control", este rango es la diferencia entre D y B. La respuesta del dispositivo corrector es proporcional al cambio de la señal de error solo dentro de la banda; fuera de ésta la respuesta del controlador cesa, porque ha alcanzado sus límites.

El valor de la banda proporcional se expresa como un % de la totalidad del rango del controlador:

$$\%BANDA \text{ PROPORCIONAL} = \frac{\text{VALOR SUPERIOR DE LA BANDA} - \text{VALOR INFERIOR DE LA BANDA}}{\text{VALOR MAXIMO DE AJUSTE} - \text{VALOR MINIMO DE AJUSTE}}$$

o, para la figura 1.10:

$$\%BANDA \text{ PROPORCIONAL} = \frac{C - B}{D - A} \times 100\%$$

La banda proporcional de la mayoría de los controladores proporcionales, usualmente se puede ajustar entre poco porcentaje a no muchos cientos por ciento.

Respuesta según el tipo de Banda Proporcional.

Con el control proporcional puede existir una oscilación temporal, hasta que el controlador se acomode en el valor final de control, pero esta oscilación disminuye hasta desaparecer, si la Banda Proporcional se ajusta apropiadamente. Si la banda proporcional es muy pequeña pueden ocurrir oscilaciones. Entre más pequeña sea la banda proporcional, el controlador operará más parecido a un control Todo o Nada. La banda proporcional adecuada elimina la constante oscilación alrededor del valor de referencia; esta es una ventaja, ya que el control es más preciso, y se reduce el desgaste del dispositivo corrector final.

A continuación se presentan las respuestas típicas de un controlador Proporcional ante un disturbio en la carga. En los ejemplos presentados a continuación se supone que el disturbio ocasiona que el valor de la variable controlada disminuya alejándose del valor de referencia.

- ❖ **Banda Proporcional Angosta.** En la figura 1.11a se presenta una banda proporcional del 10% del rango total del controlador. En este caso la acción correctora es rápida, pero la variable oscila bastante antes de estabilizarse como se muestra en la figura 1.11b.

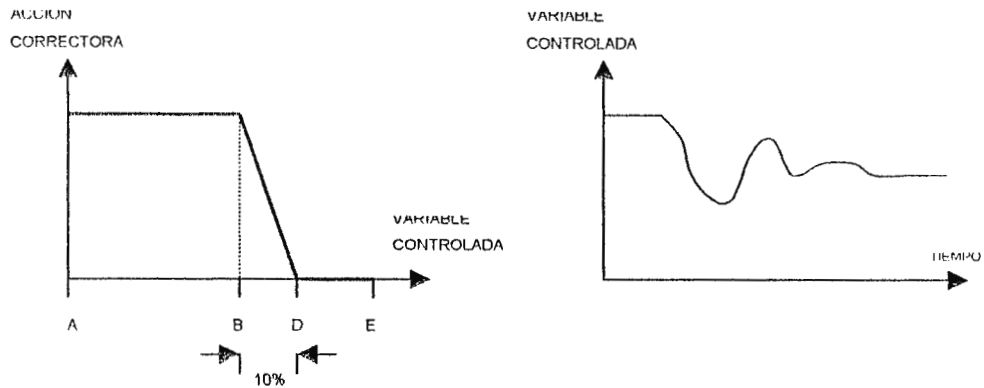


FIGURA 1.11. BANDA PROPORCIONAL ANGOSTA.

- ❖ **Banda Proporcional Mediana.** Un control proporcional con banda proporcional mediana, como la de la figura 1.12a, cuyo valor es del 50% del rango total, produce una respuesta que se aproxima lentamente al valor de control, eliminando casi totalmente la oscilación como se observa en la figura 1.12b.

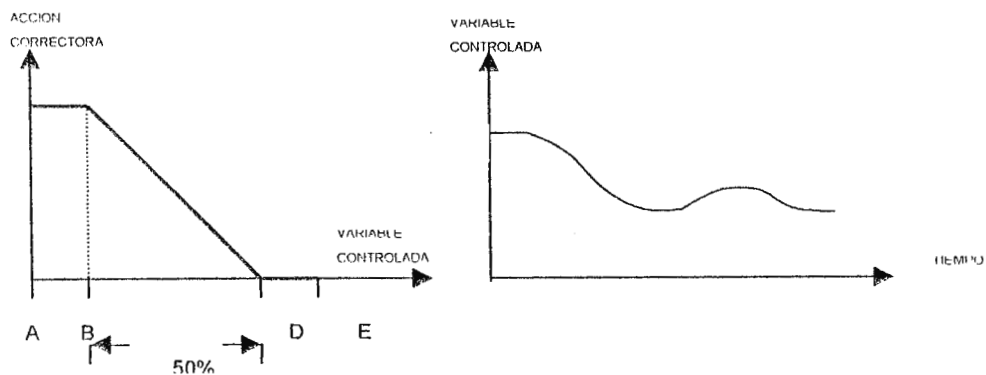


FIGURA 1.12. BANDA PROPORCIONAL MEDIANA.

❖ **Banda Proporcional Grande.** Un control con Banda Proporcional Grande como el de la figura 13a, con un valor del 200%, consume un tiempo largo en alcanzar el valor de control, pero tal como se observa en la figura 1.13b, la respuesta no experimenta oscilación.

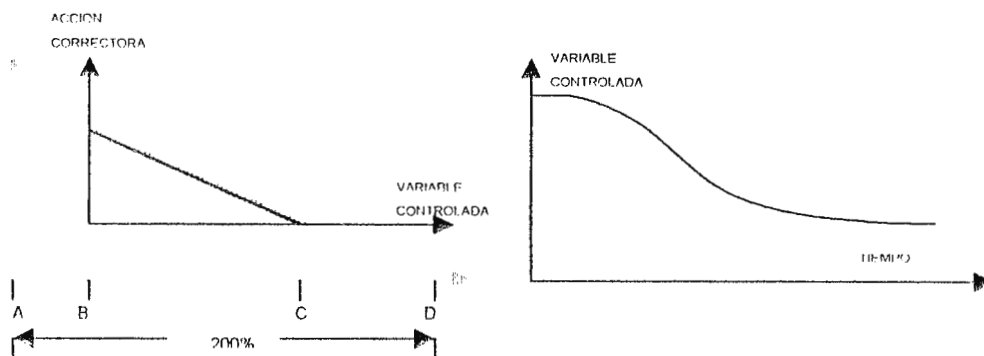


FIGURA 1.13. BANDA PROPORCIONAL GRANDE.

Los gráficos de los casos expuestos anteriormente muestran que después de un disturbio en la carga, la medida real de variable no regresa a su valor original de control. Esta es una característica del control Proporcional. Además a mayor banda proporcional, la diferencia es mayor entre los dos valores de control antes y después del disturbio. Este defecto del control Proporcional lo hace poco popular, y no muy útil. Puede entonces concluirse que el control Proporcional trabaja bien solo en los sistemas cuyos cambios en el proceso son muy lentos y pequeños.

Si el disturbio es lento la banda proporcional puede ser ajustada bastante estrecha, ya que no ocurrirá mucha oscilación por un cambio lento en el proceso, por lo que, si es posible ajustar la banda proporcional fina y estrecha, puede mantenerse pequeño el error permanente.

Desbalance.

En los ejemplos mencionados anteriormente se observa cómo a medida que aumenta el valor de la banda proporcional disminuyen las oscilaciones pero aumenta la diferencia entre el valor de referencia original y el valor de control.

A la diferencia entre los valores mencionados anteriormente se denomina *desbalance*.

1.1.2.3 CONTROL PROPORCIONAL-INTEGRAL (PI).

Ya que el control Proporcional elimina la oscilación en la variable de control, reduce el desgaste en el dispositivo corrector final; pero introduce un desbalance permanente en la variable medida. Esto vuelve a este tipo de control aplicable sólo en procesos en los cuales los cambios en la carga son pequeños y lentos, y la variación en el valor de referencia es pequeña.

Para procesos más comunes, con cambios grandes y rápidos en la carga en los que el valor de referencia puede variar considerablemente, se utiliza otro modo de control, el control Proporcional-Integral. En este tipo de control la acción correctora está determinada por dos factores:

- a. La magnitud de la señal de error, que es la parte *Proporcional*.
- b. La integral con respecto al tiempo de la señal de error; es decir la magnitud del error multiplicada por el tiempo que ha permanecido, que es la parte *Integral*.

En este tipo de control cualquier desbalance permanente del error se corregirá eventualmente a medida que pase el tiempo.

El controlador PI. es el más frecuentemente utilizado en el campo de la energía eléctrica.

El comportamiento del control Proporcional-Integral depende de la duración del tiempo, o más correctamente de la *Constante de Tiempo*. Cuando la constante de tiempo es grande, la integral es menos efectiva; y cuando es pequeña la parte integral es más efectiva.

En la figura 1.14 se describe el comportamiento del dispositivo corrector final (acción correctora) versus tiempo. En la figura 1.14b se observa el comportamiento de la variable controlada. Mientras la variable controlada es estable entre los instantes A y B, el dispositivo corrector final permanece también con una actuación constante. Cuando se da un disturbio en el instante B sobre la variable controlada, ésta tiende a alejarse del valor de referencia, elevando su valor (para este caso), entonces el dispositivo corrector final tiende a disminuir su actuación sobre la variable de forma proporcional hasta que ésta regresa al valor de referencia (en el instante C).

Como puede observarse con el control PI la variable no solo se estabiliza, como sucedía con el control proporcional, sino que además vuelve al valor de referencia.

El período de tiempo entre B y C es la constante de tiempo, y obviamente a menor constante de tiempo, más rápido se corregirá el error.

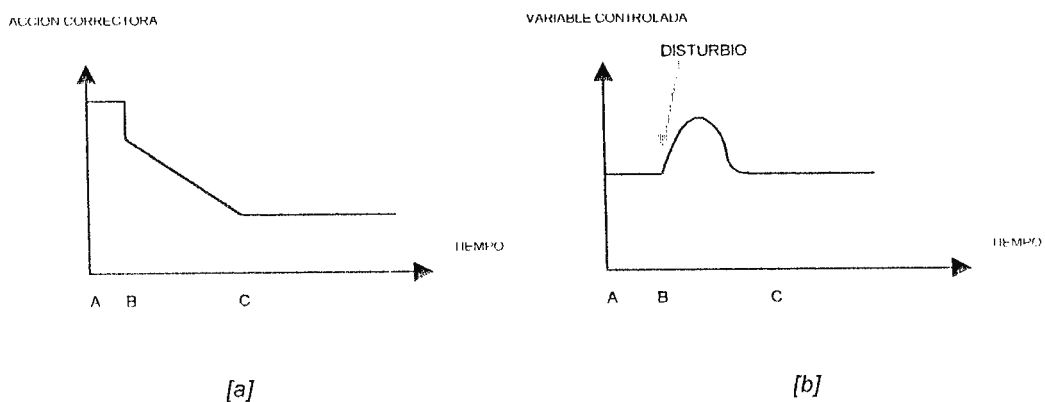


FIGURA 1.14. COMPORTAMIENTO DEL CONTROL PROPORCIONAL-INTEGRAL.

Las gráficas de la figura 1.14 son un poco idealizadas, ya que en la realidad la variable controlada no recupera su valor de referencia tan suavemente, sino que oscilará un poco.

En esta figura se observa el comportamiento tanto de la parte integral, como de la parte proporcional del controlador. La parte de control que es Proporcional se encarga de posicionar al dispositivo corrector en proporción al error existente, mientras que la parte integral mueve al dispositivo corrector en la misma dirección a medida que pasa el tiempo, ayudando así a reducir el desbalance. Luego el error se reducirá a cero, y se detendrá el dispositivo corrector final. El movimiento cesa, porque a medida que transcurre el tiempo la integral con respecto al tiempo del error no aumenta más, debido a que el error se vuelve cero.

Generalmente no se utiliza como referencia la constante de tiempo, sino su recíproco. Esta variable es denominada *Variable de Reposición*. Si la razón de reposición es baja (constante de tiempo alta) la reacción de la parte integral se vuelve lenta, es decir que la variable controlada regresa lentamente a su valor de referencia después de un disturbio. Y si la razón de reposición es alta (constante de tiempo baja) la reacción de la parte integral se vuelve rápida, es decir que la variable controlada regresa rápidamente a su valor de referencia después de un disturbio. Si la razón de reposición se ajusta muy alta, la variable controlada puede caer en una oscilación prolongada después de un disturbio

En la figura 1.15a se observa la respuesta de un control PI con una constante de tiempo de integración alta y en la figura 1.15b un control PI con una constante de tiempo de integración baja.

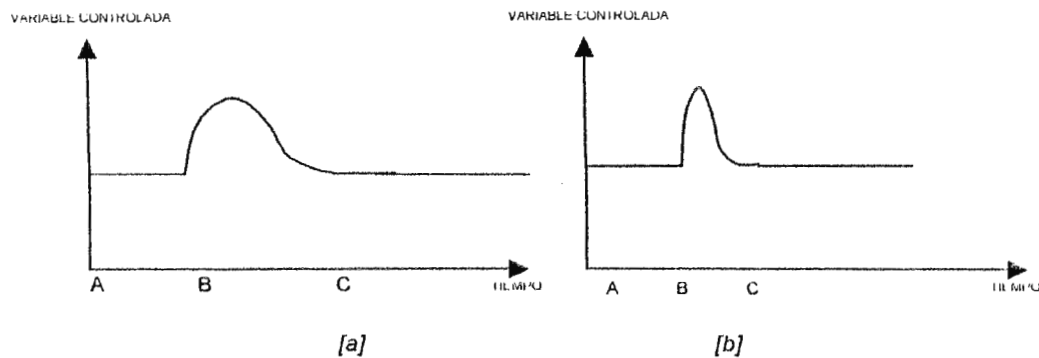


FIGURA 1.15 COMPORTAMIENTO DEL CONTROL PROPORCIONAL INTEGRAL

En general, con el control PI puede controlarse bastante bien cambios grandes en la carga y en el valor de referencia, sin oscilaciones prolongadas ni desbalances permanentes, y además una rápida recuperación después de un disturbio.

En sistemas eléctricos el circuito de un controlador proporcional integral básico, desarrollado con un amplificador inversor⁵, es el que se muestra en la figura 1.16.

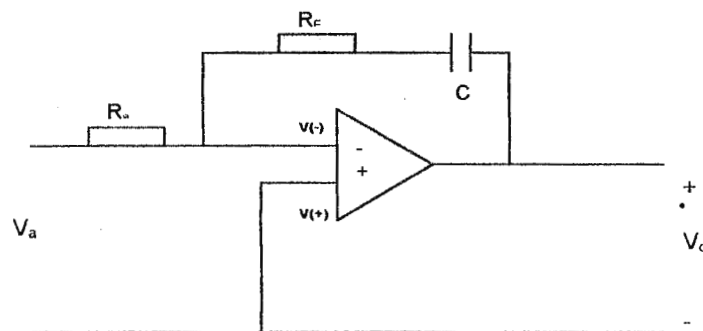


FIGURA 1.16. CIRCUITO DE UN CONTROLADOR PROPORCIONAL INTEGRAL BÁSICO.

⁵ Ver apéndice A.

El controlador integral-proporcional esta compuesto por dos etapas en su lazo de realimentación:

1. Etapa de amplificación proporcional.
2. Etapa integral.

La resistencia óhmica existente en el circuito de realimentación da lugar a la acción proporcional. El condensador proporciona la acción integral. Combinando estos elementos en el circuito de realimentación se genera la acción proporcional e integral.

Cada una de éstas genera un factor de salida.

1. ETAPA DE AMPLIFICACION PROPORCIONAL

Como
$$I_{Ra} = \frac{V_a - V_{(-)}}{R_a}$$

Y
$$I_{Rf} = \frac{V_{sal} - V_{(-)}}{R_f}$$

Además
$$I_{Ra} = I_{Rf}$$

Por lo tanto
$$\frac{V_a - V_{(-)}}{R_a} = - \frac{V_{sal} - V_{(-)}}{R_f}$$

como $V_{(-)} = 0$, para un Amp-op ideal

$$\frac{V_a}{R_a} = - \frac{V_{sal}}{R_f}$$

se obtiene
$$V_{sal} = -V_a \cdot \frac{R_f}{R_a}$$

2. ETAPA DE INVERSION

Como la capacitancia es definida como $C = \frac{dQ}{dt}$

entonces $i = \frac{CdV}{dt}$

como $V_{(-)} = 0$, entonces, $I_r = i_c$

$$\text{luego} \quad I_c = \frac{dQ}{dt} = \frac{CdV_C}{dt} = I_R$$

otra vez, como $V_{(-)} = 0$, se tiene $V_C = -V_{sal}$

$$\text{entonces} \quad I_c = -\frac{CdV_{sal}}{dt} = \frac{V_a}{R} = I_R$$

$$dV_{sal} = \frac{-1}{RC} V_a dt$$

$$V_{sal} = \frac{-1}{RC} V_a dt$$

$$V_{sal} = \frac{-1}{RC} \int V_a dt$$

$$\text{Luego de encontrar el} \quad V_{sal} = -R_f V_a - \frac{1}{RC} \int V_a dt$$

1.1.2.4 CONTROL PROPORCIONAL DERIVATIVO (PD).

El Control proporcional integral puede resultar ineficiente cuando los cambios en la variable controlada son muy rápidos, en estos casos es necesario un control que actúe según la rapidez de cambio de la señal de error, es decir, que tanto mayor sea la rapidez de cambio del error mayor será la acción correctora.

En los controles PD la parte derivativa del controlador prevé que el proceso sufrirá un gran cambio basándose en las medidas actuales, pues si la variable medida está cambiando muy rápidamente, es seguro que tratará de cambiar en una gran cantidad; entonces el controlador trata de anticipar el proceso aplicado.

En este tipo de control la constante de tiempo se denomina *Constante de Tiempo de Acción Derivada*. Su significado es contrario al de la constante de tiempo de acción

integral. Aquí cuanto mayor es la constante de tiempo, tanto mayor es también la importancia de la componente de acción derivada, es decir el valor de corrección. Este valor pasa a ser cero, cuando la variable de entrada deja de modificarse.

A veces se emplea el *Coefficiente De Acción Derivada* K_D , el cual coincide con la constante de tiempo derivada y se mide en segundos. De igual forma se emplea por analogía un *Coefficiente De Acción Proporcional* K , para el control P y un *Coefficiente De Acción Integral* K_i para el control PI.

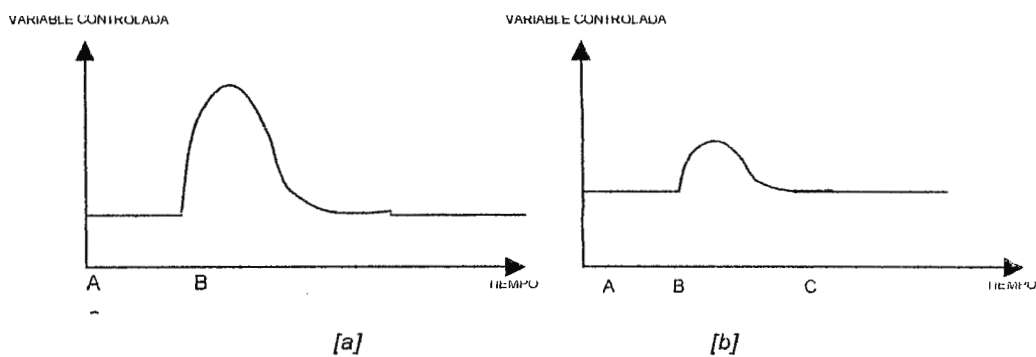


FIGURA 1.17 COMPORTAMIENTO DEL CONTROL PROPORCIONAL DERIVATIVO.

En la figura 1.17a se muestra la respuesta de la variable versus tiempo de un controlador PD con constante de tiempo de derivación pequeña, y en la figura 1.17b para una gran constante de tiempo de derivación.

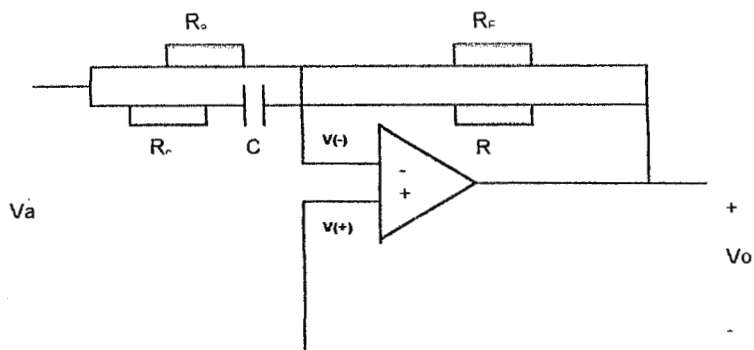


FIGURA 1.18. CONTROL PROPORCIONAL DERIVATIVO.

De forma específica, y siempre para sistemas eléctricos, el control proporcional derivativo se consigue con un amplificador inversor.

El controlador proporcional derivativo combina la respuesta de un amplificador y un diferenciador en un mismo circuito. La derivación se realiza mediante una capacitancia en la rama en paralelo del circuito de realimentación. En la rama en serie de este circuito debe disponerse tanto a la entrada del amplificador como a su salida una resistencia óhmica. En la figura 1.18 se muestra el circuito básico correspondiente.

La etapa proporcional ya se ha desarrollado en la sección anterior, por lo tanto ahora se desarrollara la etapa diferencial.

ETAPA DIFERENCIAL

La acción diferencial la determina C, el cual deja pasar al amplificador solo la porción cambiándole el voltaje de entrada.

Como $I_C = C \frac{dV_C}{dt}$ y $I_R = -I_C$

$$V_{sal} = R I_R = -R I_C$$

$$V_{sal} = -RC \frac{dV_a}{dt}$$

Ahora, igual que en el análisis anterior se obtiene V_{sal} total sumando el efecto de cada etapa:

$$V_{sal} = -\frac{R}{R_f} V_a - RC \frac{dV_a}{dt}$$

La etapa diferencial se encuentra compensada con C_C y R_C . El condensador C_C se encarga de reducir la presencia de ruidos de alta frecuencia. R_C limita la ganancia en altas frecuencias, reduciendo las demandas de corriente impuestas a la tensión de entrada.

1.1.2.5 CONTROL PROPORCIONAL-INTEGRAL-DERIVATIVO {PID}.

El control PI es el más adecuado para la mayoría, pero no todas, las situaciones de control. Esto se debe a que existen ciertas situaciones con ciertas características:

- Cambios muy rápidos en la carga.
- Grandes retardos de tiempo entre la aplicación de la acción correctora y el apareamiento de los resultados de dicha acción en la variable medida.

Ya sea que se presente una o ambas características, se suele aplicar un *control Proporcional-Integral-Derivativo*.

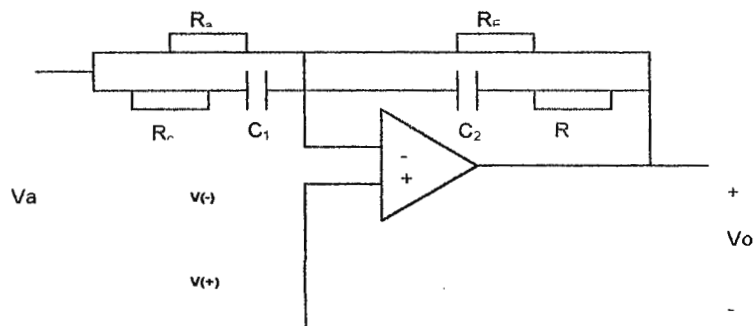


FIGURA 1.19. CONTROL PROPORCIONAL-INTEGRAL-DERIVATIVO.

En el control PID la acción correctora esta determinada por los tres siguientes factores:

- Factor Proporcional: magnitud del error.
- Factor Integral: la magnitud del error multiplicada por el tiempo que ha prevalecido, o integral con respecto al tiempo de error.
- Factor Derivativo: un rápido cambio en el error produce una acción correctora mayor que un cambio lento, o razón de cambio del tiempo de error.

De lo anterior se deduce que este control incluye tres componentes independientes: las componentes P, I y D.

Para sistemas eléctricos el control PID básico se obtiene empleando un amplificador inversor y un sencillo circuito de realimentación, ver figura 1.20. El funcionamiento de este tipo de controlador se explica combinando los conceptos de los controles PI y PD.

Sus elementos básicos son:

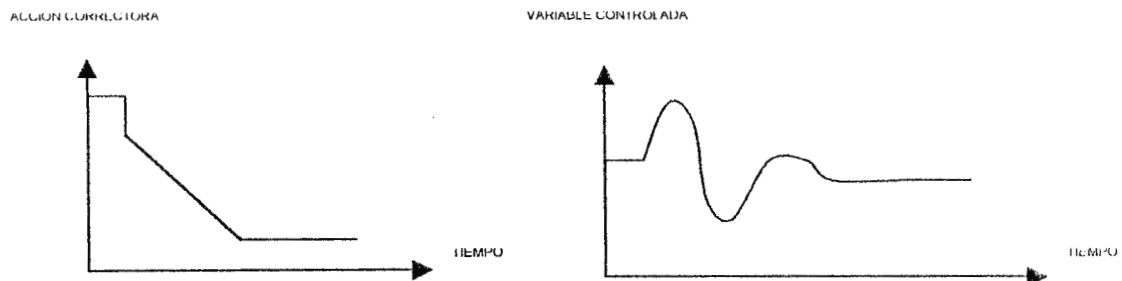
1. ETAPA DE AMPLIFICACION PROPORCIONAL.
2. ETAPA DE INTEGRACION.
3. ETAPA DERIVATIVA.

Además posee una etapa compensadora formada por R_C y C_C ; éstos se encargan de atenuar el ruido en altas frecuencias, generado por la etapa diferencial.

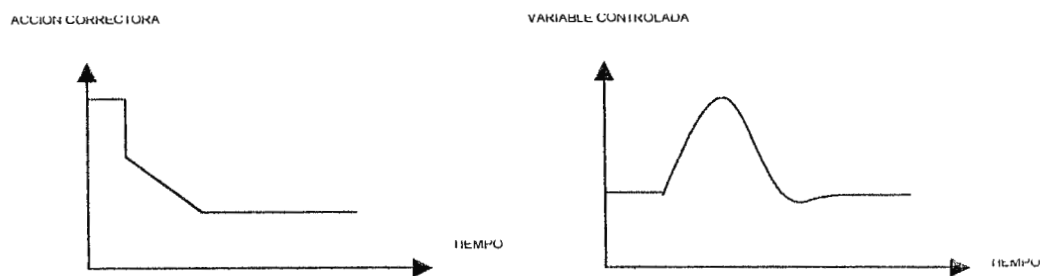
El V_{sal} total es la suma de las respuestas generadas por cada etapa.

$$V_{sal} = -\frac{R}{R_f} V_a - RC \frac{dV_a}{dt} - \frac{1}{RC} \int V_a dt$$

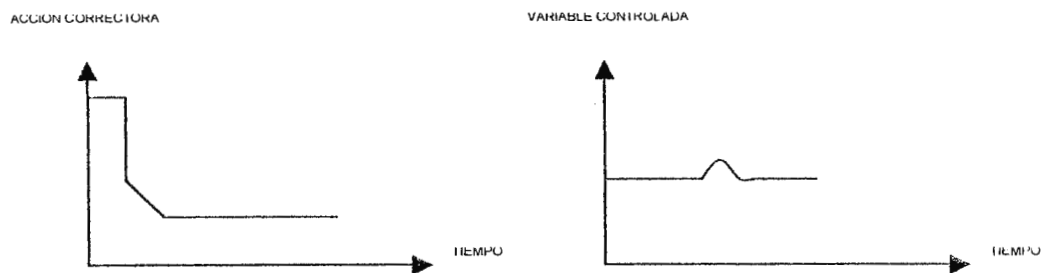
Se presentan a continuación tres pares de gráficas que muestran el efecto de cambiar la razón de tiempo, cambiando el tiempo de derivación en un controlador PID. En cada par, la primera gráfica corresponde a la acción correctora versus tiempo y la segunda a la variable controlada versus tiempo.



[a]. GRAN ERROR INICIAL POR SUPRESION DE PD.



[b]. PEQUEÑO ERROR INICIAL ANTE CORTA CONSTANTE DE TIEMPO DE DERIVACION.



[c]. MUY PEQUEÑO ERROR INICIAL ANTE UNA MAYOR CONSTANTE DE TIEMPO DE DERIVACION.

FIGURA 1.20 GRAFICAS DE LA RESPUESTA EN EL TIEMPO DE UN CONTROLADOR PID.

En la figura 1.20a se observa la respuesta del sistema cuando el control derivativo es suprimido, al existir un gran y rápido cambio en la carga. El resultado es un error inicial muy grande el cual lleva mucho tiempo en corregirse.

La figura 1.20b se introduce una corta constante de tiempo de derivación, de manera que el sistema reconozca el rápido cambio inicial en la variable controlada, presagiando así un gran cambio total. Con esto se consigue un error inicial pequeño que recobra rápidamente el valor de referencia.

Finalmente en la figura 1.20c se aumenta la constante de tiempo de derivación, con lo que se consigue un error inicial y tiempo de estabilización muy pequeños.

Si se selecciona una razón de tiempo demasiado alta, o sea, que la acción derivativa es muy alta, podrían ocurrir oscilaciones prolongadas en la respuesta del sistema. Por lo tanto, así como se tiene un límite en la razón de reposición, existe también un límite en el incremento de la razón de tiempo.

1.1.3 CARACTERÍSTICAS TÍPICAS EN LA RESPUESTA DE PROCESOS INDUSTRIALES

1.1.3.1 RETARDO DE REACCIÓN DE PROCESO.

Los procesos industriales requieren cierta cantidad de tiempo para responder a un cambio en su entrada. Es decir, siempre existe un retardo entre la aplicación de la *acción correctiva* y la aparición del resultado final. A este retardo se le llama *constante de tiempo de retardo*, o *retardo de reacción del proceso*.

Esta característica depende directamente de la constante de tiempo RC^6 . Entre más grande sea R , mayor será la *constante de tiempo*, y se necesitará más tiempo para alcanzar el valor de estabilización. Igualmente a mayor valor de C , mayor tiempo se necesita para llegar al valor final permanente.

Por lo general los retardos se miden en segundos, aunque ciertos procesos presentan retardos de 15 o 30 minutos y algunos de una hora o más.

En la figura 1.20a se muestra la gráfica típica de variable versus tiempo en un sistema con retardo de reacción. Grandes retardos de reacción de los procesos ayudan a prevenir sobrepasos de la medida de la variable.

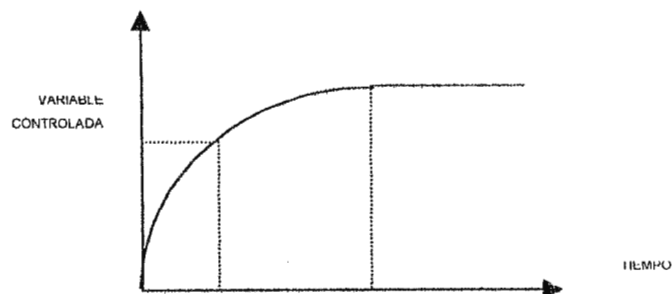


FIGURA 1.20a. GRÁFICA TÍPICA DE LA VARIABLE VRS. TIEMPO EN UN SISTEMA CON RETARDO DE REACCIÓN.

⁶ RC : Constante que depende de la resistencia (R) y la capacitancia (C).

1.1.3.2 TIEMPO MUERTO.

En ciertos procesos no se experimenta respuesta en la variable controlada durante cierto tiempo, después de que interviene el controlador. A este retraso se le llama *atraso de transporte*. Y a la cantidad real de tiempo que permanece indetectado el cambio en el dispositivo corrector se le llama *tiempo muerto*.

El nombre *atraso de transporte*, se debe al movimiento físico del medio controlado. Siempre se consume cierto tiempo para que los elementos respondan a una determinada orden, por lo que se puede decir que es un atraso mecánico, como por ejemplo el retraso que causa mover una serie de engranajes. El tiempo muerto puede llegar a ser un problema serio.

En la figura 1.20b se presenta un ejemplo de una gráfica de medida de variable versus tiempo después de un disturbio con atraso de transporte.

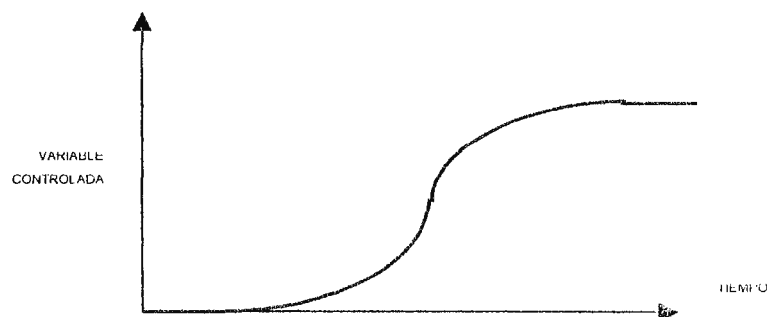


FIGURA 1.20b. GRAFICA VARIABLE VRS. TIEMPO LUEGO DE UN DISTURBIO CON ATRASO DE TRANSPORTE.

1.1.3.3 ATRASO DE TRANSFERENCIA.

Cuando existe más de una constante de tiempo RC en un sistema de control, lógicamente se experimenta un mayor retardo de tiempo, a este retardo se le llama *atraso de transferencia*. Esto provoca que la variable medida del proceso inicial responda muy lentamente a la acción correctora. En la figura 1.21 se presenta la respuesta de un sistema con *atraso de transferencia*, en el que la variable ha experimentado solamente el 10% del cambio total después de un minuto. Esto representa un problema en el control de sistemas porque es difícil prevenir sobrepasos en la medida de la variable.

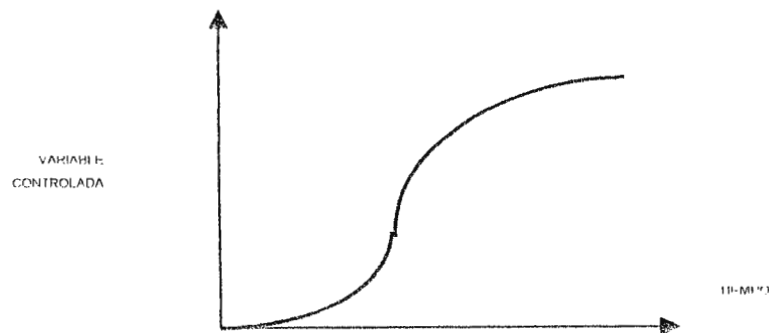


FIGURA 1.21. RESPUESTA CON ATRASO DE TRANSFERENCIA, CON 10% DE CAMBIO TOTAL.

1.1.4 TABLA RESUMEN DE LOS MODOS DE CONTROL

Las cinco características principales que determinan el modo de control más apropiado para un determinado sistema son:

- a. Magnitud del disturbio en la carga.
- b. Velocidad del disturbio en la carga.
- c. Tiempo de retardo.
- d. Tiempo muerto.
- e. Atraso de transferencia.

Basándose en éstas se ha desarrollado la tabla 1.1 que relaciona el modo de control con las características deseadas.

MODO DE CONTROL	MAGNITUD DEL DISTURBIO EN LA CARGA MAXIMO.	VELOCIDAD DEL DISTURBIO EN LA CARGA MAXIMO	RETARDO DE REACCION DE PROCESO MINIMO	TIEMPO MUERTO MAXIMO	ATRASO DE TRANSFERENCIA MAXIMO
TODO O NADA	Pequeño.	Lento.	Largo.	Muy corto.	Muy corto.
P	Pequeño.	Lento.	Largo o moderado.	Moderado.	Moderado.
PI	Cualquiera.	Lento.	Cualquiera.	Moderado.	Moderado.
PD	Pequeño.	Cualquiera.	Largo o moderado.	Moderado.	Moderado.
PID	Cualquiera.	Cualquiera	Cualquiera.	Cualquiera.	Cualquiera.

TABLA 1.1. RELACION ENTRE LAS CARACTERISTICAS DEL PROCESO Y EL MODO DE CONTROL.

1.2 TRANSDUCTORES

En el ámbito industrial se ha concluido que la mejor forma de medir la variable controlada es convirtiéndola en una señal de naturaleza eléctrica, debido a esto se han desarrollado sistemas electrónicos de control que están compuestos por ciertos elementos que se encargan de realizar las mediciones y registrar los resultados; de éstos los elementos principales son:

- Dispositivos de entrada. Se encarga de recibir la cantidad bajo medición y de entregarla al acondicionador de señal en forma de señal eléctrica proporcional a ella
- Acondicionador de señal o dispositivo procesador. Se encarga de amplificar, filtrar o modificar la señal para que sea compatible con el dispositivo de salida.
- Dispositivo de salida. Varían desde elementos que permitan un despliegue visual, como un display, un tubo de rayos catódicos, etc.; o también un registrador de cinta magnética que almacene los datos; hasta sistemas de control digital para el control de procesos.

La calidad de los sistemas electrónicos de medición depende de la exactitud y rapidez con que estos dispositivos midan e interpreten el valor de la variable controlada.

Los dispositivos de entrada encargados de convertir el valor de la variable que se controla en una señal de salida de la misma o diferente naturaleza se llaman *transductores*. Y si la señal de salida es eléctrica se llaman "*transductores eléctricos*".

Es pertinente, aclarar que se llama *transductor* al elemento como tal, y el termino *sensor* se aplica al transductor realizando la actividad de *senear* como parte de un sistema.

1.2.1 CLASIFICACION

Los transductores pueden clasificarse en pasivos y activos. Los pasivos son los que requieren de una fuente de potencia externa para poder actuar. Los activos son los que generan su propia potencia. A continuación se presentan dos tablas⁷, en las que se reúne a los transductores más comunes en la industria. La tabla 1.2 clasifica a los transductores según el parámetro eléctrico que utilizan, en la tabla 1.3 se presenta a los activos.

TRANSDUCTORES PASIVOS		
<i>Parámetro eléctrico y clase de transductor.</i>	<i>Principio de operación y naturaleza del dispositivo</i>	<i>Aplicación típica</i>
RESISTENCIA		
Dispositivo potenciométrico.	El posicionamiento de un cursor por medio de una fuerza eléctrica varía la resistencia eléctrica de un potenciómetro o de un circuito puente.	Presión, desplazamiento.
Galga extensiométrica resistiva.	La resistencia de un alambre o de un semiconductor se modifica por elongación debido a esfuerzos aplicados externamente.	Fuerza, torque, desplazamiento.
Medidor de alambre caliente.	La resistencia de un elemento caliente se varía por enfriamiento por medio de un chorro de gas.	Flujo de gas, presión de gases.
Termómetro de resistencia.	La resistencia de un alambre de metal puro con un gran coeficiente resistivo se cambia con la temperatura.	Temperatura, calor radiante.
Termistor.	La resistencia de un cierto óxido de metal con un coeficiente resistivo de temperatura negativo se cambia con la temperatura.	Temperatura.
Higrómetro resistivo.	La resistencia de una cinta conductora cambia con el contenido de la humedad.	Humedad relativa.
Celda fotoconductora.	La resistencia de la celda como un elemento de un circuito varía con la luz incidente.	Relé fotosensible.
CAPACITANCIA		
Galga de presión de capacitancia variable.	La distancia entre dos placas paralelas se varía por la aplicación de una fuerza externa.	Desplazamiento, presión.
Micrófono de condensador.	La presión del sonido varía la distancia entre una placa fija y un diafragma móvil.	Voz, música, ruido.

TABLA 1.2. TRANSDUCTORES PASIVOS.

⁷ Clasificación según William David Cooper.

Galga dieléctrica.	Variación de la capacitancia por cambios en el dieléctrico.	Nivel de líquidos, espesor.
INDUCTANCIA		
Transductor de circuito magnético.	La autoinductancia o inductancia mutua de una bobina excitada con ca varía cambiando su circuito magnético.	Presión, desplazamiento.
Detector de reluctancia.	La reluctancia de un circuito magnético se cambia variando la posición del núcleo de hierro de una bobina.	Presión, fuerza, desplazamiento, vibración.
Transformador diferencial.	El voltaje diferencial de dos devanados secundarios de un transformador se cambia variando la posición de un núcleo magnético por medio de una fuerza aplica externamente.	Desplazamiento, espesor.
Galga de corrientes inducidas.	La inductancia de una bobina se varía aproximándole un plato con corrientes inducidas.	Fuerza, presión, sonido.
Galga magnetométrica.	Las propiedades magnéticas se varían por presión y esfuerzos mecánicos.	Flujo, magnético, corriente.
VOLTAJE Y CORRIENTE		
Detector por efecto Hall.	Una diferencia de potencial se genera a través de una placa semiconductor (germanio) cuando el flujo magnético interactúa con una corriente aplicada.	Flujo magnético, corriente.
Cámara de ionización.	Flujo electrónico inducido por ionización de un gas debido a radiación radiactiva.	Conteo de partículas, Radiación.
Celda fotoemisiva.	Emisión electrónica debido a radiación incidente sobre superficies fotoemisivas.	Luz y radiación.
Tubo fotomultiplicador.	Emisión electrónica secundaria debido a la radiación incidente sobre un cátodo fotosensible.	Luz y radiación.

TABLA 1.2. TRANSDUCTORES PASIVOS (CONTINUACION)

TRANSDUCTORES ACTIVOS		
<i>Clase de transductor.</i>	<i>Principios de operación y naturaleza del dispositivo.</i>	<i>Aplicación típica.</i>
Termocupla y termopila.	Se genera una fem cuando la unión de dos metales o semiconductores diferentes se calienta.	Temperatura, flujo de calor, radiación
Generador de bobina móvil.	El movimiento de una bobina dentro de un campo magnético genera un voltaje.	Velocidad, vibración.
Detector piezoeléctrico.	Se genera una fem cuando se le aplica una fuerza externa a ciertos materiales cristalinos, tal como el cuarzo.	Sonido, vibración, aceleración, cambio de presión.
Celda fotovoltaica.	Se genera un voltaje de un dispositivo semiconductor cuando la energía, radiante estimula la celda.	Medidor de luz, celda solar.

TABLA 1.3. TRANSDUCTORES ACTIVOS.

1.3 CONVERTIDORES ANÁLOGO – DIGITAL

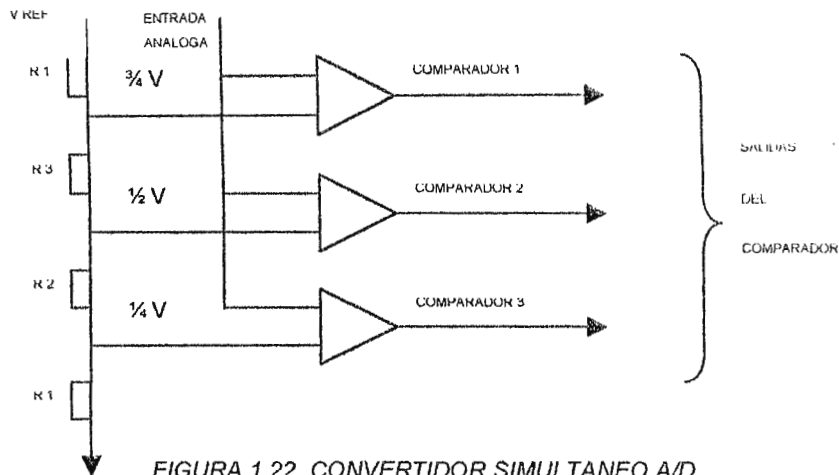
Luego de la obtención de los valores de la variable controlada a través de los transductores, en forma de señales eléctricas generalmente analógicas es necesario convertir el voltaje análogo a su equivalente en forma digital. Así la salida del convertidor A/D puede ser posteriormente interpretada por el sistema controlador.

Existen diferentes métodos para realizar la conversión A/D, los más comunes son:

- a. Convertidor simultáneo A/D.
- b. Convertidor A/D de contador.
- c. Convertidor continuo A/D.
- d. Convertidor por aproximación sucesiva A/D.

1.3.1 CONVERTIDOR SIMULTANEO A/D.

Se construye en base a circuitos comparadores. En la figura 1.22 se observa un circuito de este tipo, con tres comparadores, donde cada uno es alimentado desde un voltaje de referencia que proviene de un divisor de voltaje derivado de una fuente de precisión "V". Las cuatro resistencias proveen de voltajes de salida de $\frac{3}{4} V$, $\frac{1}{2} V$ y $\frac{1}{4} V$; mientras que el otro terminal del comparador es alimentado por el voltaje análogo desconocido.



En este circuito cada comparador se encenderá cuando su voltaje de referencia sea menor que el voltaje análogo, esto significa que si la entrada análoga es menor a $\frac{1}{4} V$ ninguno de los comparadores se encenderá; si solo el comparador C1 se enciende significa que $V_{\text{análogo}}$ está entre $\frac{1}{4} V$ y $\frac{1}{2} V$; de igual forma si se encienden solo c1 y c2, significa que $V_{\text{análogo}}$ está entre $\frac{1}{2} V$ y $\frac{3}{4} V$; finalmente si encienden los tres comparadores significa que $V_{\text{análogo}}$ está entre $\frac{3}{4} V$ y V . Se poseen entonces un total de cuatro salidas, que pueden ser codificadas en dos bits de información en código binario. Para aumentar la exactitud de la respuesta digital sería necesario aumentar el número de bits, para lo cual, lógicamente sería necesario aumentar el número de comparadores; por ejemplo para tres bits se necesitarían 2^3 salidas, de las cuales una es 0V, por lo tanto para las otras siete salidas se necesitarían siete comparadores.

Este sistema posee la ventaja de ser sencillo y veloz, pero a la vez posee la desventaja de volverse muy voluminoso cuando se desea una alta resolución en la respuesta.

1.3.2 CONVERTIDOR A/D DE CONTADOR.

Existe una forma de reducir a uno el número de comparadores, comparando la entrada análoga con un voltaje de referencia variable. Ver figura 1.23.

Si el voltaje de referencia incrementara en rampa (linealmente), las coincidencias entre él y el voltaje desconocido se pueden determinar en base al tiempo transcurrido desde que la rampa era cero.

Siendo el voltaje de referencia una variable controlada digitalmente por un convertidor D/A, el cual transforma el número digital con que se alimenta, a través de sus registros internos, en un valor análogo; se puede comparar la entrada análoga desconocida por medio de un circuito comparador. Si los registros no son iguales se modificará el número digital en los registros del DAC (convertidor análogo/digital) y su salida se comparará nuevamente. Una forma sencilla de controlar la conversión dentro del DAC, es arrancar al convertidor desde cero para luego contar el número de pulsos requeridos para llegar a un voltaje de salida igual a la señal análoga, ver figura 1.24.

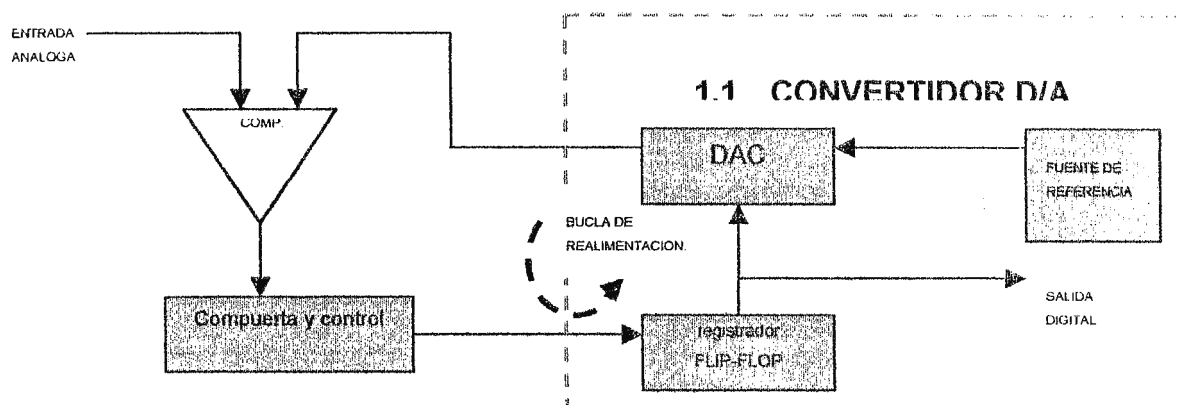


FIGURA 1.23. CONVERTIDOR A/D DE CONTADOR.

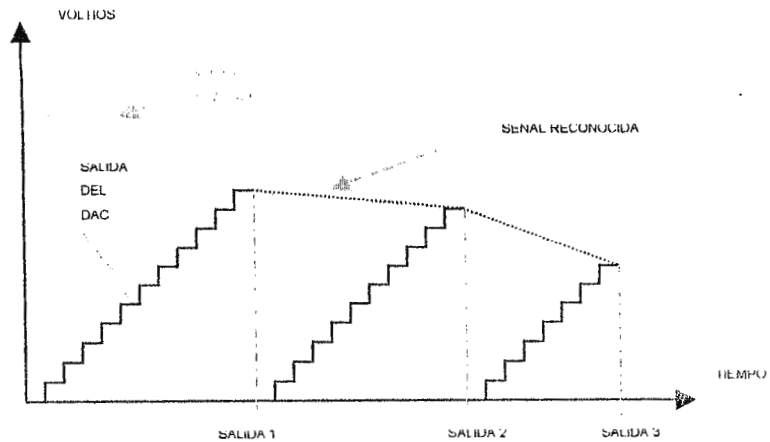


FIGURA 1.24. FORMA DE ONDA EN LA ENTRADA ANALOGA Y LA SALIDA DEL DIVISOR.

1.3.3 CONVERTIDOR CONTINUO A/D.

Las cuatro etapas básicas de un Convertidor Continuo A/D son:

- a. Etapa de contador reversible.
- b. Etapa convertidora D/A.
- c. Etapa comparadora.
- d. Etapa de sincronización y lógica de control.

En la figura 1.25 se observa un diagrama lógico simplificado de bloques del Convertidor Continuo D/A.

a. ETAPA DE CONTADOR REVERSIBLE.

Utilizando un contador reversible, es decir que cuenta hacia adelante y hacia atrás es posible seguir continuamente el voltaje analógico de la entrada, sin importar la dirección en que éste cambie. En la figura 1.25 se observa un contador que

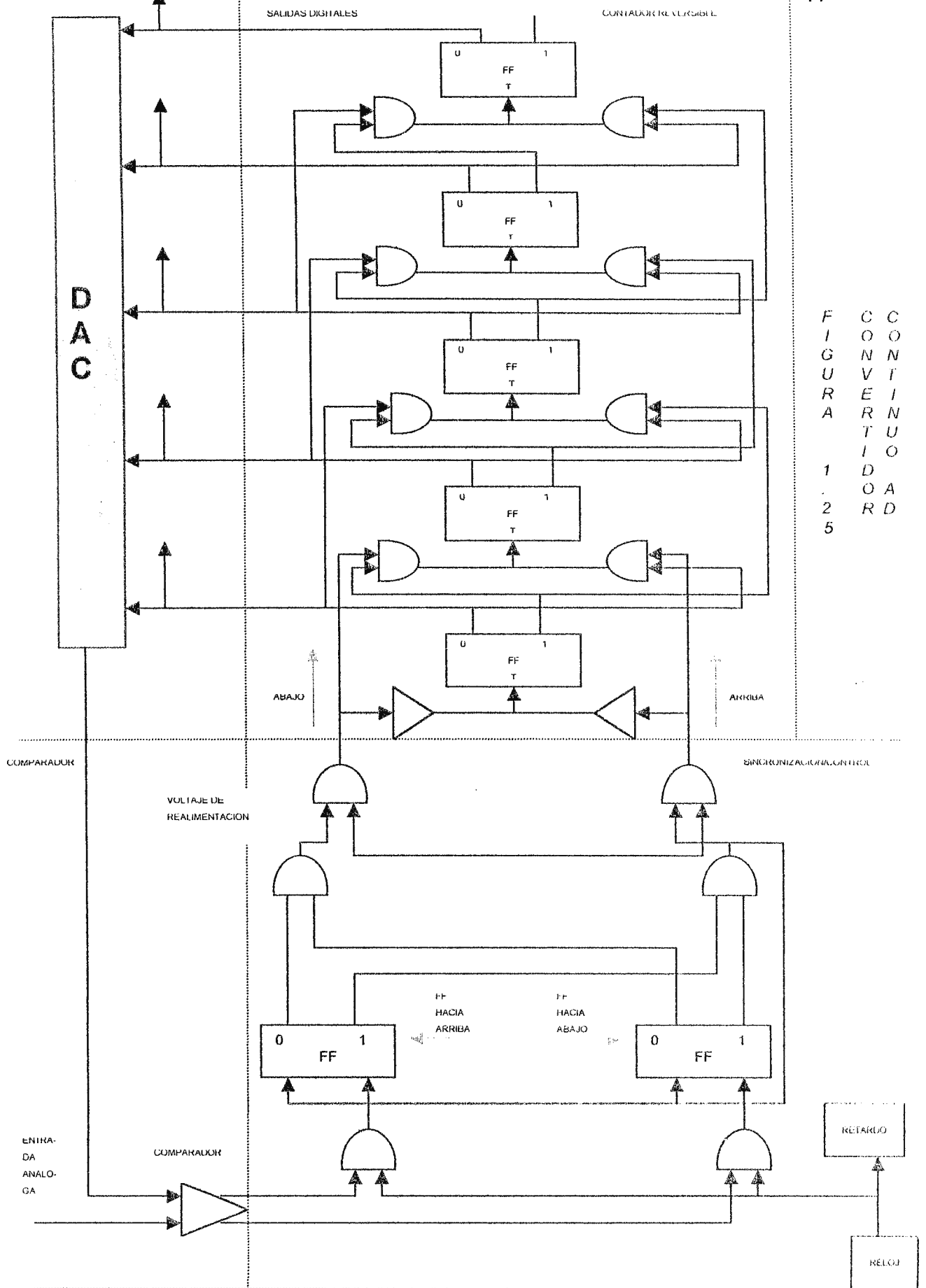


FIGURA 1.25
CONVERTIDOR
DIGITAL
ANALOGICO

ordinariamente cuenta hacia arriba cuando la entrada de disparo de un FF⁸ se conecta a la salida 1 del FF anterior, modificado para contar también hacia abajo conectando cada FF también a la salida 0 del FF anterior. Para sincronizar y comparar el conteo, este contador posee compuertas AND en los circuitos de disparo de los FF asegurando que las cuentas se acumulen en los momentos deseados.

b. ETAPA CONVERTIDORA D/A.

Para que la conversión sea exacta en esta etapa, se necesita una fuente de referencia que suministre un voltaje de precisión requerido. En la figura se ven las salidas de los FF conectadas directamente al DAC, cuyas entradas deben ser 1, por lo que debe existir la conversión apropiada entre estos dos circuitos, aunque en la figura no se muestre.

c. ETAPA COMPARADORA.

Esta etapa realiza una segunda comparación del voltaje análogo con la salida del DAC, así, teniendo dos niveles de salida, es posible determinar cuando el voltaje de entrada es más grande que el realimentado (V salida del DAC), conectando entonces la salida apropiada a la entrada de ajuste del FF_{arriba} a través de una compuerta. Así mismo, si el voltaje análogo es menor que el realimentado, se suministra a través de una compuerta, un voltaje de salida al otro terminal del FF_{abajo}.

d. ETAPA DE SINCRONIZACIÓN Y LÓGICA DE CONTROL.

El reloj que controla la sincronización del ciclo de medición, controla la transferencia de las señales de salida del comparador a los FF.

El contador sigue al voltaje analógico continuamente; ver figura 1.26. Esto se explica en los siguientes párrafos.

⁸ La abreviatura FF se utiliza para referirse a los circuitos digitales Flip-Flop.

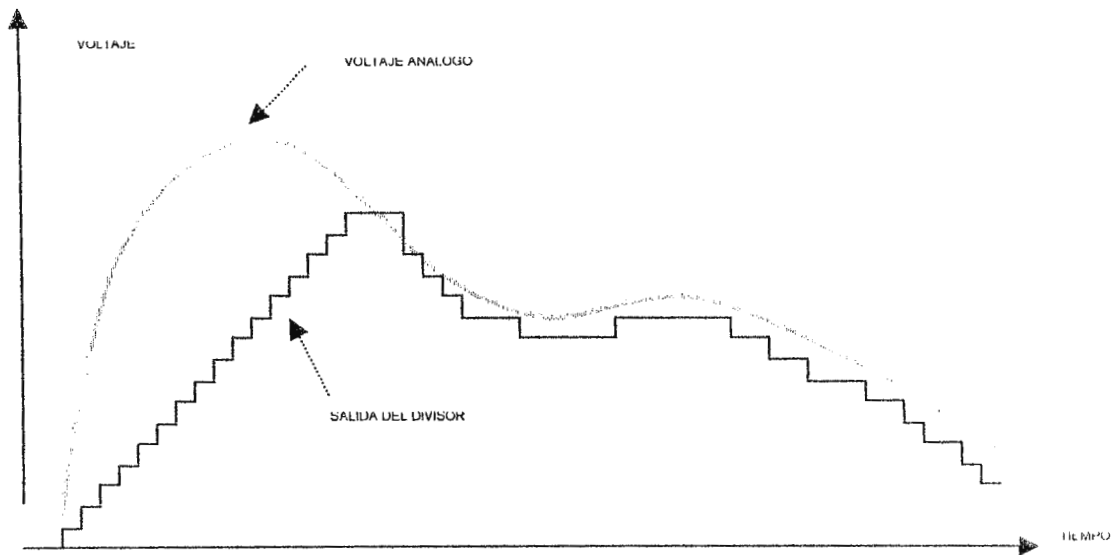


FIGURA 1.26. ACCION DE UN CONVERTIDOR A/D CONTINUO.

En el inicio del ciclo de medición con todos los FF a cero, se genera un pulso de reloj para mostrar la salida del comparador; en este momento, en condiciones normales el voltaje analógico de entrada será más grande que el voltaje realimentado, entonces se ajusta el FF_{arriba}. El pulso de reloj dispara el primer FF, cuya salida 0 se conecta al DAC; con el mismo pulso de reloj se condiciona la compuerta de disparo del siguiente FF. El voltaje analógico se compara entonces, con la salida del DAC, y si la entrada todavía es muy grande, el siguiente pulso de reloj redispara al FF_{arriba}, reiniciando el primer binario y disparando el siguiente. De esta forma se avanza en la cuenta de uno en uno. Comparando la entrada análoga con la salida del DAC, se repite el proceso hasta que ambas señales son iguales; y como en este momento la salida del comparador es cero se detiene la cuenta.

Cuando el reloj detecta que cambia a un valor inferior la señal análoga, ajusta el FF_{abajo}; y el pulso entra al contador en la entrada de disparo del primer binario, pero como

ahora la cuenta se lleva al lado de los ceros, el contenido del contador se reduce en uno. Como el DAC cae hasta la cantidad apropiada, la siguiente comparación determina cual FF será ajustado.

1.3.4 CONVERTIDOR POR APROXIMACIÓN SUCESIVA A/D.

Este convertidor es capaz de manejar señales continuas y discontinuas con muy buena resolución, regular velocidad y bajo costo. En la figura 1.27 se muestra el diagrama de bloques correspondiente.

Se utiliza un registro de control digital, un convertidor análogo con fuentes de referencia, un comparador, un lazo de control de tiempo y un registro de distribución que consiste en un contador en anillo, (con un uno circulando en su interior), para decidir qué paso toma lugar.

Se inicia ajustando a 1 el bit más significativo (MSB) de los registros de control y de distribución y a 0 el resto de los bits. De esta forma se registra (en el registro de distribución) que el ciclo comienza.

Se genera una salida de un medio de la fuente de referencia en el convertidor D/A; a la vez, entra un pulso en la cadena de retardo de tiempo, el cual es llevado a la compuerta a través de la salida del comparador cuando se han ajustado el convertidor y el comparador. Al ser ajustado en el registro de control el bit más significativo por medio de la cadena de tiempo, el MSB permanece en 1, o dependiendo de la salida del comparador, puede también ser reajustado a 0. Se traslada a la próxima posición en el registro de distribución el 1, conservando un récord del número de comparaciones hechas.

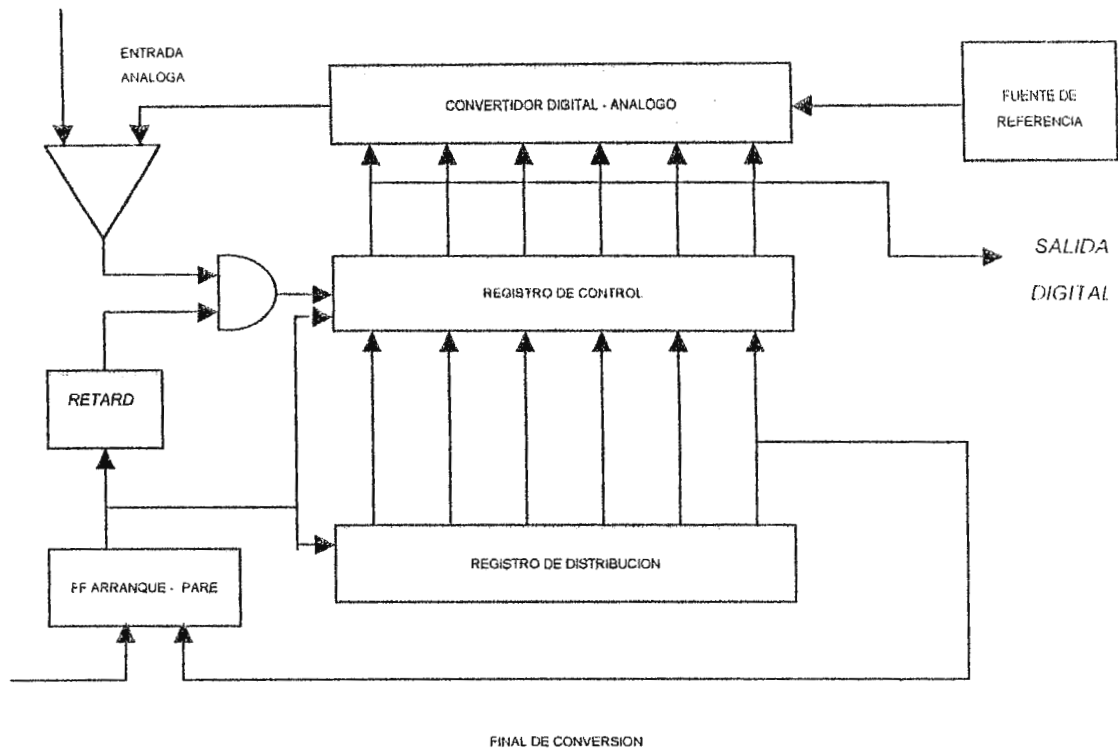


FIGURA 1.27. CONVERTIDOR POR APROXIMACIÓN SUCESIVA A/D.

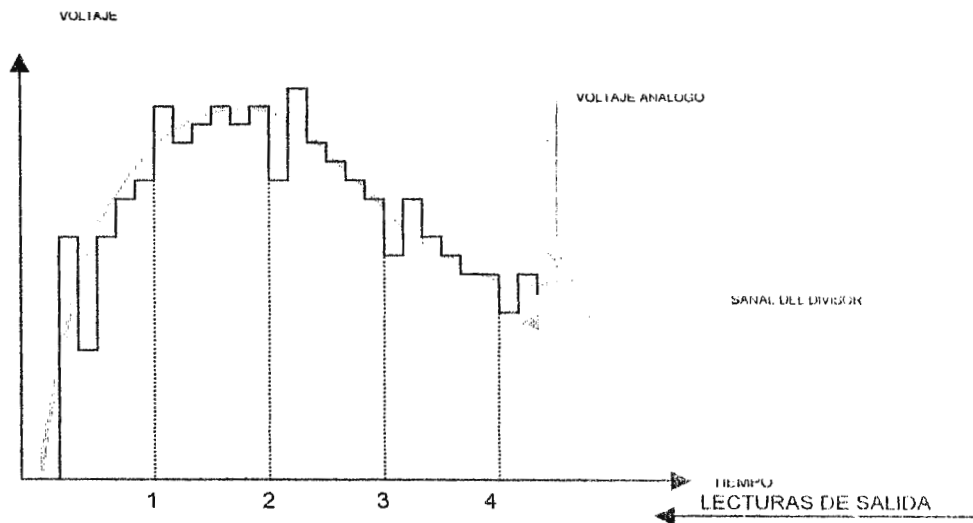


FIGURA 1.28. OPERACIÓN DEL CONVERTIDOR POR APROXIMACIÓN SUCESIVA A/D.

Se repite el proceso descrito hasta que la aproximación final es correcta, y el registro de distribución indica el final de la conversión.

La forma de onda que ilustra el proceso de la operación del convertidor se presenta en la figura 1.28.

1.4 MUESTREO Y MANTENIMIENTO

Cuando se trabaja con circuitos convertidores análogo – digital normales, como los descritos anteriormente por ejemplo, se tiene la desventaja de que dichos circuitos no son capaces de interpretar correctamente las señales de alta frecuencia, es necesario entonces implementar una etapa extra, que mantenga un nivel de salida estable para que el DAC sea capaz de realizar la conversión correctamente, a dicha etapa se le denomina de “muestreo y mantenimiento”.

El circuito de muestreo y mantenimiento puede ser implementado tal como se muestra en la figura 1.29, con un amp-op⁹ y transistores de actuación rápida y conmutación veloz. El amp-op se encarga de incrementar la corriente disponible en un condensador o para aislarlo de la carga de salida.

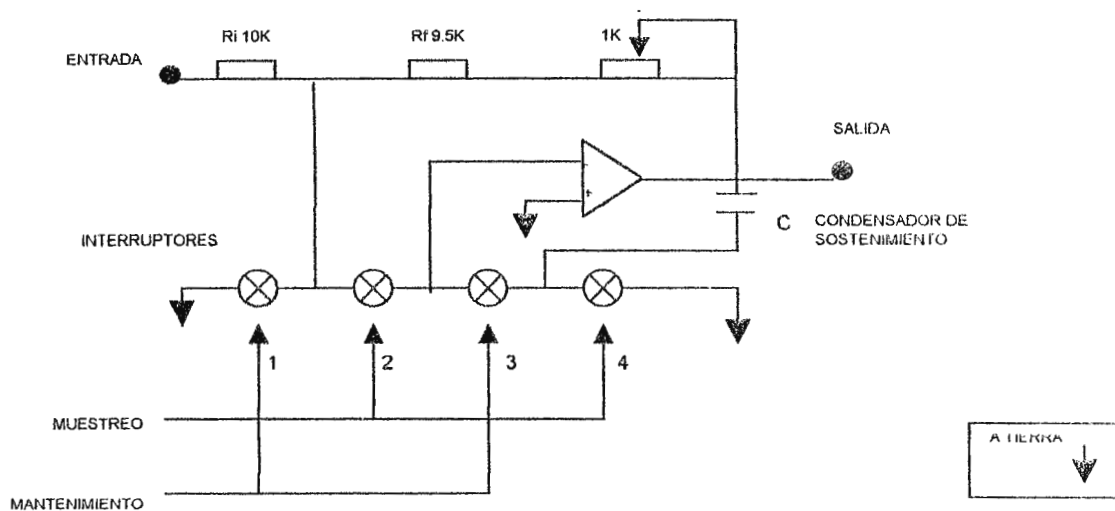


FIGURA 1.29. CIRCUITO DE MUESTREO Y MANTENIMIENTO.

⁹ Amp-op: amplificador operacional.

Los pulsos de muestreo y mantenimiento son complementarios, los interruptores 1 y 3 son operados por el de mantenimiento, mientras que el de muestreo controla los interruptores 2 y 4.

- *Muestreo.*

El condensador se carga a través del amp-op.

- *Mantenimiento.*

La forma de onda a la salida del circuito se muestra en la figura 1.29. El condensador se conmuta al lazo de realimentación a la vez que R_i y R_f se ponen a tierra.

Durante el muestreo y mantenimiento la entrada del amp-op se encuentra a unos pocos μV respecto a tierra (a excepción de la conmutación).

1.5 CONVERTIDORES DIGITAL-ANÁLOGO

Como se menciono anteriormente, después de la obtención de los valores de la variable controlada a través de los transductores, en forma de señales eléctricas generalmente analógicas, es necesario convertir el voltaje análogo a su equivalente en forma digital. Así, la salida del convertidor A/D puede ser posteriormente interpretada por el sistema controlador; pero luego de ser procesada, las salidas de los sistemas digitales internos se encuentran en algún tipo de código binario, por lo que muchas veces es necesario convertir dichas señales de salida, a una forma de señal análoga.

La conversión digital-análogo consiste en tomar un valor representado en código digital y convertirlo en una señal eléctrica análoga proporcional al valor digital.

1.5.1 DAC CON SALIDA DE VOLTAJE

El circuito básico de un tipo de convertido D/A de 4 bits se muestra en la figura 1.30. Se utiliza un circuito amplificador sumador con Amp-Op el cual produce una suma con valor asignado de las entradas A, B, C y D. Dichas entradas son digitales y se encuentran todas ya sea a 0V o a 5V.

El Amp-Op en esta configuración multiplica cada voltaje de entrada por la proporción de la resistencia de retroalimentación R_f a la resistencia de entrada R_{in} .

En la figura R_f es de 1 K ohm y las entradas varían de 1 a 8 K ohm. Como la resistencia en D es de 1 K ohm el Amp-Op, no atenúa el voltaje de entrada en D. El valor de C es 2 K ohm, en este caso la razón R_f/R_{in} es 0.5, por lo tanto el voltaje de entrada en C será atenuado un 50%; y así sucesivamente.

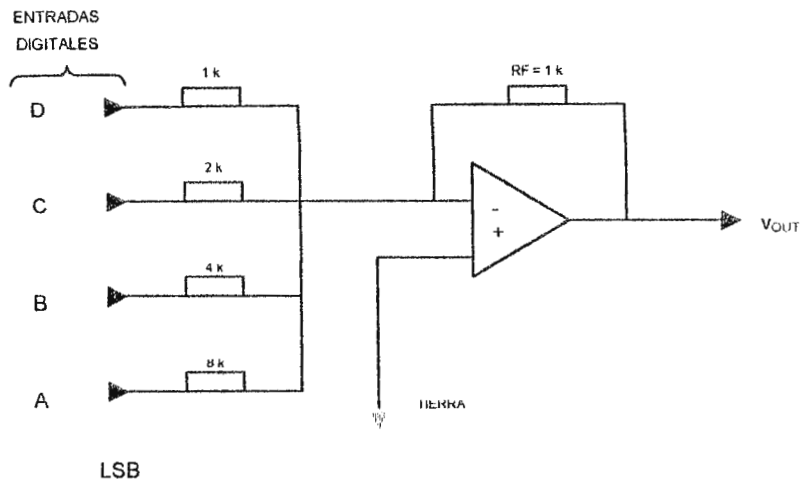


FIGURA 1.30. DAC CON SALIDA DE VOLTAJE

La salida del DAC es:

$$V_{out} = -(V_D + 0.5V_C + 0.25V_B + 0.125V_A)$$

Entonces, la conversión en este caso se regirá según la tabla 1.4.

D	C	B	A	V_{OUT} (V)
0	0	0	0	0
0	0	0	1	-0.625
0	0	1	0	-1.250
0	0	1	1	-1.875
0	1	0	0	-2.500
0	1	0	1	-3.125
0	1	1	0	-3.750
0	1	1	1	-4.375
1	0	0	0	-5.000
1	0	0	1	-5.625
1	0	1	0	-6.250
1	0	1	1	-6.875
1	1	0	0	-7.500
1	1	0	1	-8.125
1	1	1	0	-8.750
1	1	1	1	-9.375

TABLA 1.4. SALIDA DEL DAC

1.5.2 DAC CON SALIDA DE CORRIENTE.

Este circuito utiliza cuatro trayectorias de corriente en paralelo, tal como se muestra en la figura 1.31, estas trayectorias son controladas individualmente por interruptores semiconductores, los cuales son controlados a la vez por los niveles de voltaje en las entradas.

El circuito opera de forma tal, que la corriente que fluye por cada una de las trayectorias depende del voltaje de referencia y del valor de la resistencia en la trayectoria.

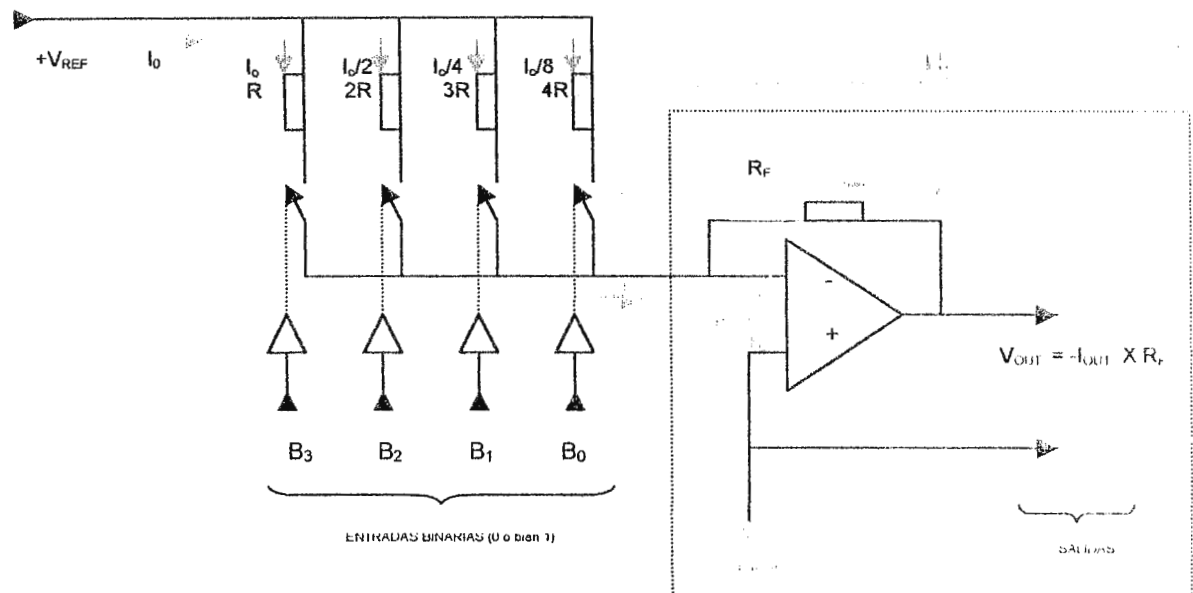


FIGURA 1.31. DAC CON SALIDA DE CORRIENTE.

La corriente total I_{OUT} depende del estado de los interruptores, (que son controlados por las entradas binarias) y de la corriente que fluye por cada trayectoria paralela. La corriente en las trayectorias, depende a su vez, del voltaje de referencia y la resistencia en cada trayectoria.

El cálculo de I_{OUT} se obtiene de:

$$I_{OUT} = (B_3 \times I_0) + (B_2 \times I_0/2) + (B_1 \times I_0/4) + (B_0 \times I_0/8)$$

donde

$$I_0 = V_{REF} / R$$

Para que I_{OUT} sea exacta, R_L debe ser prácticamente cero, o sea, un corto a tierra. Esto se logra conectando la salida I_{OUT} a la entrada negativa de un Amp-Op abierto, el cual esta virtualmente a tierra. El Amp-Op actúa como convertidor de corriente a voltaje. Como la retroalimentación negativa del Amp-Op hace circular a tierra una corriente I_F de igual valor a I_{OUT} a través de R_F , se genera un $V_{OUT} = I_{OUT} \times R_F$.

Finalmente se obtiene una salida análoga " V_{OUT} " proporcional a la entrada binaria, capaz de impulsar una amplia gama de cargas sin ser afectada.

1.6 ETAPA DE POTENCIA

Como se menciona en la sección 1.1.1, los elementos de potencia que se utilicen en la etapa de salida dependen del tipo de control que se desarrolle.

La etapa de potencia para los diferentes tipos de control Proporcional requiere de elementos que se acoplen a este cambio proporcional en la salida, como los tiristores. En cambio en el control Todo o Nada es más conveniente utilizar algún tipo de interruptor, contactor o relé.

A continuación se explican los elementos más comúnmente utilizados en las etapas de potencia de los circuitos de control de lazo cerrado.

1.6.1 TIRISTORES

En los modos de control P, PI, PD y PID muchas veces se requiere de una variación continua de la señal eléctrica que representa la variable controlada, en este caso no pueden utilizarse relés o contactos, pues poseen solo dos estados posibles.

Los tiristores de potencia, como los SCR y Triacs se acomodan bastante bien al control proporcional de la variable controlada. Puede hacerse variar el ángulo de conducción proporcionalmente al error entre la medida de la variable controlada y el valor de referencia. Esto varía en forma continua el flujo de corriente, proporcionando así un *Control P, PI, PD o PID*.

Entre las aplicaciones industriales de los tiristores se tiene por ejemplo el control de corriente en un elemento calefactor, o en un electroimán cuya fuerza

magnética de atracción debe ser variable, también a través de una bombilla incandescente en situaciones donde debe variarse la intensidad luminosa. Pero su aplicación más común es variar la corriente a través de los devanados de un motor para variar su velocidad.

Se describe a continuación el triac, el cual se emplea para alimentar cargas de AC.

El *triac* es un tiristor capaz de controlar el valor promedio de la corriente que fluye en una carga, variando el ángulo de disparo en la terminal *puerta*.

El triac actúa como un interruptor abierto que puede conducir corriente en dos direcciones (una a la vez) cuando es llevado a *conducción*, y como un interruptor cerrado que no permite el flujo de corriente entre sus terminales cuando es *bloqueado*.

La velocidad de trabajo de los triacs es mayor a la de los relés, pues tienen un retardo de tiempo de accionamiento del orden de los microsegundos.

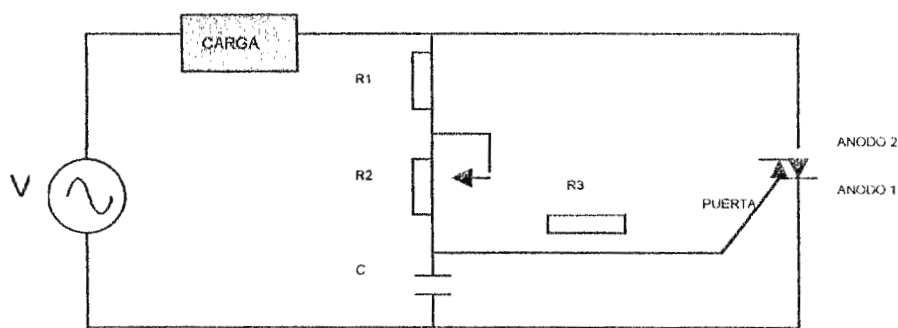


FIGURA 1.32. CIRCUITO CON TRIAC Y CONEXION DE LA FUENTE DE VOLTAJE Y LA CARGA; CON CIRCUITO DE DISPARO SIMPLE.

En la figura 1.32 se presenta un circuito con triac que muestra cómo se conecta a la fuente de voltaje y la carga; además se observa un circuito de disparo simple, donde el ángulo de disparo se ajusta por medio de un potenciómetro.

1.6.2 RELÉS Y CONTACTORES

En procesos industriales que aplican el modo de control Todo o Nada se utilizan generalmente dispositivos de dos estados capaces de controlar ciertos niveles de potencia denominados *Relés y Contactores*. Estos dispositivos presentan dos estados diferentes “cerrado” y “abierto”, por lo tanto su comportamiento es binario. Éstos son elementos de contacto, por lo que poseen un tiempo de retardo del orden de los milisegundos.

La diferencia entre un *Relé* y un *Contactor* radica en su construcción, el valor de las potencias que pueden controlar y la capacidad de interrupción de sus contactos. Los relés manejan pequeñas corrientes, en cambio los contactores son capaces de manejar grandes corrientes.

Para hacer que un relé, o contactor, magnético se energice, la corriente por la bobina debe aumentar por encima de un cierto valor, denominado *corriente de enganche*; luego una vez *enganchado* el relé o contactor, la corriente de la bobina deberá caer a un cierto valor bajo de corriente denominado *corriente de mantenimiento*.

- Contactores hidráulicos. Cuando la fuerza de accionamiento procede de un líquido (agua, aceite, etc.).
- b. Por la disposición de sus contactos:
- Contactores al aire. La ruptura se produce en el aire.
 - Contactores al aceite. La ruptura tiene lugar en un baño de aceite. Éstos permiten mayores intensidades con un tamaño no mayor que los contactores de aire, debido al efecto refrigerante del aceite. En los contactores de aceite se gastan rápidamente los contactos, por lo que necesitan un mayor mantenimiento, ya que debe renovarse periódicamente el aceite.
- c. Por la clase de corriente:
- Contactores de corriente continua.
 - Contactores de corriente alterna.
- d. Por los límites de tensión:
- Contactores de baja tensión. Hasta 1000 V.
 - Contactores de alta tensión. A partir de 1000 V en adelante.

El *contactor electromagnético* es el de mayor empleo. Existen dos disposiciones constructivas principales en los contactores electromagnéticos. Generalmente se utilizan los contactos principales, auxiliares y bobinas montadas sucesivamente en un eje convenientemente aislado para aplicaciones de mediana y gran potencia; esta disposición constructiva se denomina *contactor en barra*.

Se prefieren que los elementos estructurales del contactor constituyan un bloque compacto para aplicaciones de pequeña y mediana potencia; esta es la otra disposición constructiva y se denomina *contactor en bloque*.

Consideraciones sobre el empleo de contactores en corriente continua.

Para elegir un contactor para su empleo en corriente continua es necesario conocer la tensión y la corriente a cortar, pero también la constante de tiempo, expresada por

$$T = L / R.$$

del circuito controlado por el contactor.

Algunas magnitudes de la constante de tiempo son:

- Cargas no inductivas (hornos de resistencias): $L/R \leq 1$ ms.
- Motores Shunt: $L/R = 7.5$ ms.
- Motores Serie: $L/R = 10$ ms.
- Electroválvulas: $L/R \approx 50$ ms.
- Electroimanes de contactores: $L/R = 30 \dots 200$ ms.

Diferentes formas de conectar los contactores en corriente continua.

A continuación se presentan los diferentes esquemas bajo los que pueden conectarse los contactores para DC.

1. Conexión de dos polos en serie. (Ver figura 1.33).

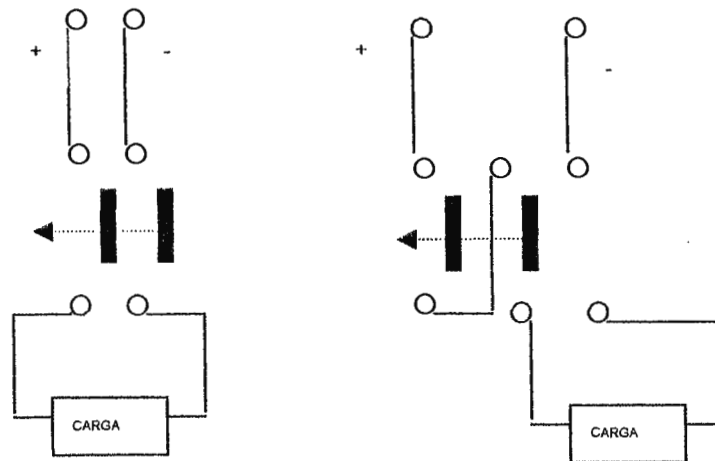


FIGURA 1.33. CONEXIÓN DE 2 POLOS EN SERIE.

2. Conexión de tres polos en serie. (Ver figura 1.34).
3. Conexión de cuatro polos en serie. (Ver figura 1.35).

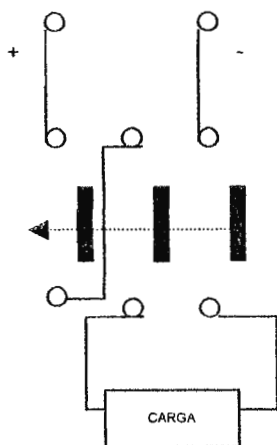


FIGURA 1.34. CONEXIÓN DE 3 POLOS EN SERIE.

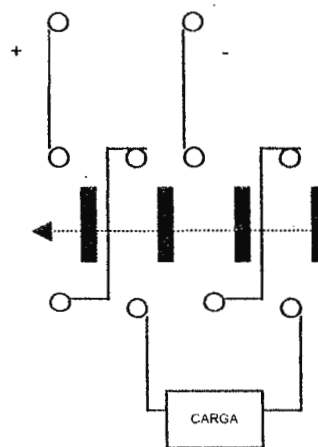


FIGURA 1.35. CONEXIÓN DE 4 POLOS EN SERIE.

1.7 AISLAMIENTO ELECTRICO.

El aislamiento eléctrico entre circuitos, significa que los circuitos no tienen conductores en común. Esto es necesario para evitar que el ruido eléctrico generado en un circuito sea transmitido al otro.

Un aislador óptico es básicamente una interface entre dos circuitos que operan (generalmente) a diferentes niveles de voltaje. Un uso industrial bastante importante es comunicar una señal entre dos circuitos que están aislados entre sí eléctricamente.

1.7.1 ACOPLES OPTICOS

Los acoples ópticos se utilizan en los sistemas de control para el aislamiento entre circuitos de adquisición de datos de alto voltaje y circuitos lógicos digitales de bajo voltaje, o entre circuitos lógicos y los dispositivos de la etapa de potencia.

Los acoples ópticos más comunes se presentan a continuación (ver figura 1.36):

- Conjunto "LED - FOTODIODO".
- Conjunto "LED - FOTRANSISTOR".
- Conjunto "LED - FOTODARLINGTON".

- Conjunto "LED - RESISTENCIA DEPENDIENTE DE LUZ".
- Conjunto "LUZ DE NEON - RESISTENCIA DEPENDIENTE DE LUZ".
- Conjunto "LAMPARA INCANDESCENTE - RESISTENCIA DEPENDIENTE DE LUZ".

Generalmente de todos los acoples ópticos presentados, el más utilizado debido a su eficiencia es el conjunto "LED – Fototransistor". El LED emite luz cuando por él circula corriente en polarización directa, con un voltaje de 1.0 a 2.2 V; esta luz activa la base del Fototransistor, comunicando de esta forma la señal sin necesidad de conexiones físicas. La práctica estándar que rige la construcción de este tipo de optoacopladores es construirlos con LEDs Infrarrojos y Fototransistores sensibles a luz infrarroja. Los elementos infrarrojos son más eficientes que los de luz visible, pues convierten más cantidad de energía en radiación infrarroja, y por lo tanto menos en calor. Además no son afectados por la luz del medio ambiente.

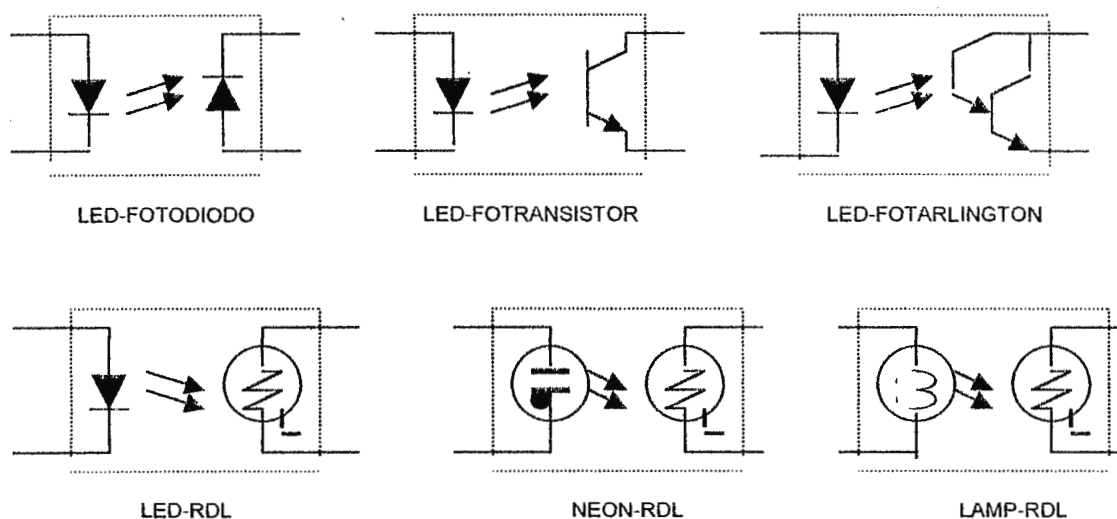


FIGURA 1.36. ACOPLER OPTICOS COMUNES.

CAPITULO 2

EL CONTROLADOR LOGICO PROGRAMABLE (PLC)

2.1 HISTORIA DEL PLC

Las industrias de proceso utilizan equipo electrónico que mide, indica y controla variables, tales como caudal, presión, temperatura, nivel y composición. Las aplicaciones de medición y control para las variables del proceso abarcan desde la indicación y/o regulación de una única variable del proceso hasta centenares de variables de una planta industrial completa, es con este fin que se han desarrollado los autómatas programables, estos típicamente reemplazan a los circuitos lógicos de control con relés, temporizadores, contadores, contactores, etc.

El principio de la evolución de los equipos de control se puede marcar desde “el control manual”, que consistía en utilizar elementos tales como válvulas locales y control hidráulico/neumático, luego combinando ambas técnicas se desarrollaron los controladores por válvulas solenoide. Pero todos estos procesos de control resultaban lentos, caros y de difícil mantenimiento, por lo que se desarrolló el “control por relés”. Sin embargo la progresiva complejidad de los procesos industriales exigía desarrollar procesos de control más eficiente, por lo que la aparición de los transistores redujo el tamaño y el costo de los equipos de control.

El siguiente paso fue la aplicación de la electrónica digital, inicialmente con las compuertas lógicas y luego con elementos más especializados como flip-flops's, decodificadores, multiplexores y unidades aritméticas lógicas, con lo que se redujo en gran

escala el tamaño y costo de mantenimiento de estos sistemas, a la vez que se incrementó su velocidad, precisión y exactitud.

Finalmente en 1977 se introdujeron los sistemas de control basados en microprocesadores, los cuales poseen la capacidad de manejar cualquier tipo de sistemas secuenciales, según el programa almacenado en su memoria. Dichos sistemas además de presentar en mayor escala las ventajas de los equipos basados en electrónica digital, permitieron el control de variables en tiempo real. Estos sistemas basados en microprocesadores son llamados en la actualidad "Autómatas programables" o "Controladores Lógicos Programables", (o simplemente "PLC").

El desarrollo cronológico de los Automatas Programables se resume en la tabla 2.1.

AÑO	DESARROLLO
1968	<ul style="list-style-type: none"> ▪ SE DESARROLLA EL CONCEPTO DE LOS PLC.
1969	<ul style="list-style-type: none"> ▪ HARDWARE DE LA CPU, CON INSTRUCCIONES LÓGICAS, DE 1K DE MEMORIA Y 128 PUERTOS.
1974	<ul style="list-style-type: none"> ▪ USO DE MULTIPROCESADORES, TIMESRS Y CONTADORES. OPERACIONES ARITMÉTICAS, 12 K DE MEMORIA Y 1024 PUNTOS E/S.
1976	<ul style="list-style-type: none"> ▪ SISTEMAS DE ENTRADA SALIDA REMOTOS.
1977	<ul style="list-style-type: none"> ▪ SE INTRODUCE EL USO DEL PLC BASADO EN MICROPROCESADORES.
1980	<ul style="list-style-type: none"> ▪ DESARROLLO DE MÓDULOS INTELIGENTES DE E/S, DESARROLLO DE FACILIDADES DE COMUNICACIÓN. MAYOR CAPACIDAD DEL SOFTWARE Y UTILIZACIÓN DE MICROCOMPUTADORAS COMO AUXILIARES PARA LA PROGRAMACIÓN.
1983	<ul style="list-style-type: none"> ▪ SE INTRODUCE EL PLC DE BAJO COSTO.
1985	<ul style="list-style-type: none"> ▪ DESARROLLO DE REDES DE TODO NIVEL PARA PLC. DESARROLLO DE LOS LENGUAJES DE PROGRAMACIÓN GRÁFICOS, Y DISTRIBUCIÓN JERÁRQUICA DEL CONTROL INDUSTRIAL.

TABLA 2.1. DESARROLLO CRONOLÓGICO DE LOS PLCs.

Desde 1980 a aumentado el mercado de PLCs pequeños (con aproximadamente 20 puntos de E/S digitales, y 500 pasos de programación) en respuesta la producción de las compañías japonesas que han desarrollado unidades muy pequeñas y de bajo costo, volviéndose muy competitivos en el mercado de los Automatas Programables.

El PLC en comparación con sus predecesores los sistemas de control basados en relés requiere menos espacio y energía; y puede ser programado o reprogramado para diferentes operaciones, éste consiste realmente en una computadora para aplicación industrial. El término aplicaciones industriales significa que el PLC debe ser diseñado para ser operado en un ambiente de manufactura, a diferencia de otras computadoras que son diseñadas para operar en ambientes de oficina.

Las secciones internas principales que conforman un PLC son: la sección de control lógico y los módulos de entrada y salida. La sección de control lógico se encarga de realizar los cálculos y funciones del proceso de datos según el programa destinado al control de la planta. Los módulos de entrada y salida interconectan una variedad de sensores analógicos y digitales.

Si se comprarán los PLCs con otros sistemas de control, se concluye que éstos son la mejor elección, a no ser que exista algún tipo de requerimiento especial, que vuelva necesaria la aplicación de algún otro tipo de sistemas. Su funcional diseño de hardware¹ y software² los hace atractivos, para la mayoría de aplicaciones de control automático.

La tabla 2.2 presenta un resumen de la comparación entre diferentes sistemas de control.

¹ Hardware: Los componentes físicos del PLC.

² Software: Conjunto de programas de todo tipo de los PLCs.

CARACTERÍSTICA	Precio por funciones.	Tamaño físico.	Velocidad de operación.	Inmunidad al ruido.	Capacidad de operaciones complejas.	Facilidad para cambio de función.	Facilidad de mantenimiento	Instalación
SISTEMA DE RELÉ	TOLERABLE.	GRANDE.	BAJA.	EXCELENTE.	NO.	MUY DIFÍCIL.	DIFÍCIL POR SUS CONTACTOS.	CONSUME TIEMPO EN DISEÑARLO.
LÓGICA DIGITAL	BAJO	MUY COMPACTO.	MUY RÁPIDA.	BUENA.	SÍ.	DIFÍCIL.	DIFÍCIL POR EL NÚMERO DE ICs.	CONSUME TIEMPO EN DISEÑARLO.
COMPUTADORAS	ALTO	TOLERABLE COMPACTO.	BASTANTE RÁPIDA.	REGULAR.	SÍ.	MUY SIMPLE.	DIFÍCIL.	CONSUME TIEMPO EN PROGRAMARLO
PLCs	BAJO.	MUY COMPACTO.	RÁPIDA.	BUENA.	SÍ.	BASTANTE SIMPLE.	BUENA.	SIMPLE DE PROGRAMAR E INSTALAR.

TABLA 2.2. COMPARACION DE SISTEMAS DE CONTROL.

2.2 EL PLC

Un controlador programable es definido por la NEMA (National Electrical Manufacturers Association of North America) como un "dispositivo electrónico digital que usa una memoria programable para guardar instrucciones e implementar funciones específicas tales como lógica, secuenciales, temporización, de conteo y aritmética, para el control de máquinas y procesos". Sin embargo, esta definición es demasiado general como para enmarcar la mayoría de los dispositivos de estado sólido utilizados en la manufactura, desde un simple temporizador hasta un programa de computadora. En consecuencia es mejor examinar las ventajas y características funcionales esenciales de un PLC que basarse solamente en esta definición.

2.2.1 VENTAJAS COMPETITIVAS.

La popularidad de los PLCs en aplicaciones industriales en el control de maquinaria y procesos radica básicamente en su facilidad de programación y eficacia. Lo que genera incrementos en la productividad, reducción de costos y mayor calidad.

Las principales ventajas que el PLC ofrece sobre los controles electromecánicos son:

1. Menor costo de materiales, mano de obra y mantenimiento.
2. Las modificaciones no afectan la estructura del sistema. Es decir que modificando solamente el programa controlador es posible realizar fácil y rápidamente cambios y adaptaciones sin afectar la estructura de la máquina o proceso que se controla.
3. Menor consumo de energía.
4. Reducción de hardware. Pues los PLCs utilizan circuitos de alta integración.

5. Transportabilidad de los programas. Es decir que un mismo programa puede ser cargado en diferentes controladores (compatibles) que realicen en determinado momento una misma función.

2.2.2 CARACTERISTICAS FUNCIONALES DE UN PLC

Las características funcionales más importantes de un PLC se describen a continuación:

1. Tiene un espacio programable por el usuario. Esta característica le permite al usuario escribir y cambiar programas en dicho espacio sin tener que solicitar al fabricante una programación específica.
2. Contiene funciones preprogramadas. Los PLCs contienen al menos funciones lógicas, contadores, temporizadores y memoria que el usuario puede acceder a través de algún tipo de lenguaje de programación orientado al control.
3. Posee un proceso para examinar la memoria, las entradas y las salidas. Esta característica permite determinar con precisión como responderá la máquina o proceso al programa.
4. Proporciona chequeo y diagnóstico de errores. Un PLC periódicamente realiza una prueba interna a la memoria, procesador y sistema de entradas/salidas para asegurarse de que éste está realizando correctamente el programa que controla la maquinaria o proceso.
5. Puede ser monitoreado. Los PLC están provistos de sistemas de monitoreo como luces indicadoras que muestran el estado de las entradas y salidas.

6. Su cubierta está diseñada apropiadamente. Los PLCs están diseñados para soportar la temperatura, humedad, vibración, y ruido que se encuentra presente en la mayoría de las fábricas.
7. Son diseñados para propósitos generales. Generalmente un PLC no es diseñado para una aplicación específica, pero puede manejar una amplia variedad de control de tareas efectivamente.

2.2.3 ESTRUCTURA DEL PLC

2.2.3.1 DIAGRAMA DE BLOQUES DE UN PLC

Un modelo simplificado de un PLC es mostrado en la figura 2.1.

La entrada convierte las señales de nivel alto que vienen de los dispositivos externos a señales de niveles lógicos³ que el procesador del PLC puede leer directamente. El procesador lógico lee estas entradas y decide que estados de salida deberán activarse en base al programa lógico del usuario. Los módulos de salida convierten los niveles lógicos de las salidas del procesador, en señales de nivel alto, necesarias para manejar una amplia variedad de dispositivos. El programa monitor se utiliza para introducir o modificar el programa de usuario en la memoria y para monitorear la ejecución de éste.

³ Nivel Lógico: Nivel de tensión compatible con los dispositivos electrónicos.

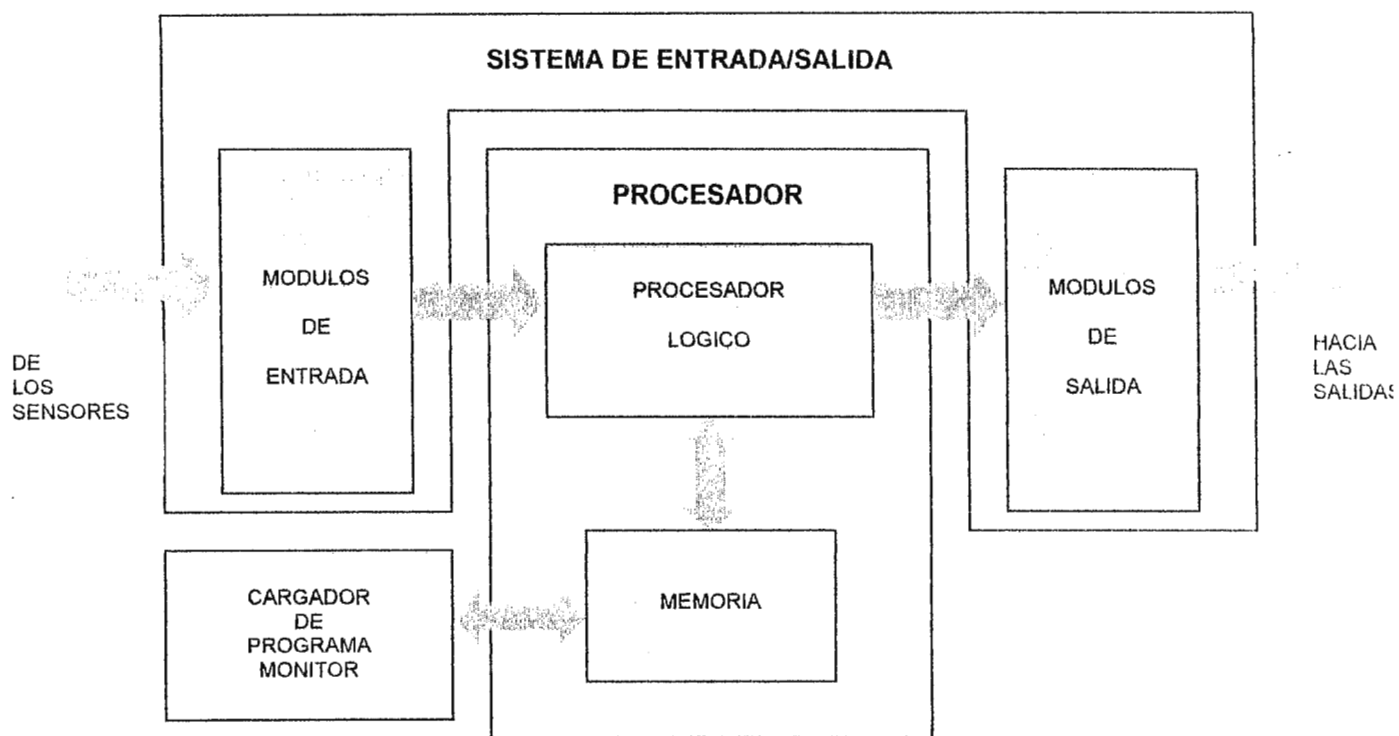


FIGURA 2.1. DIAGRAMA DE BLOQUES SIMPLIFICADO DEL PLC.

2.2.4 SECCION PROCESADORA

Un detallado diagrama de bloques de la sección procesadora de un PLC es mostrado en la figura 2. 2. Está sección esta constituida por cuatro elementos principales:

1. Fuente de poder.
2. Memoria.
3. Unidad de Control y Procesamiento (CPU).
4. Interfaz Entrada/Salida.

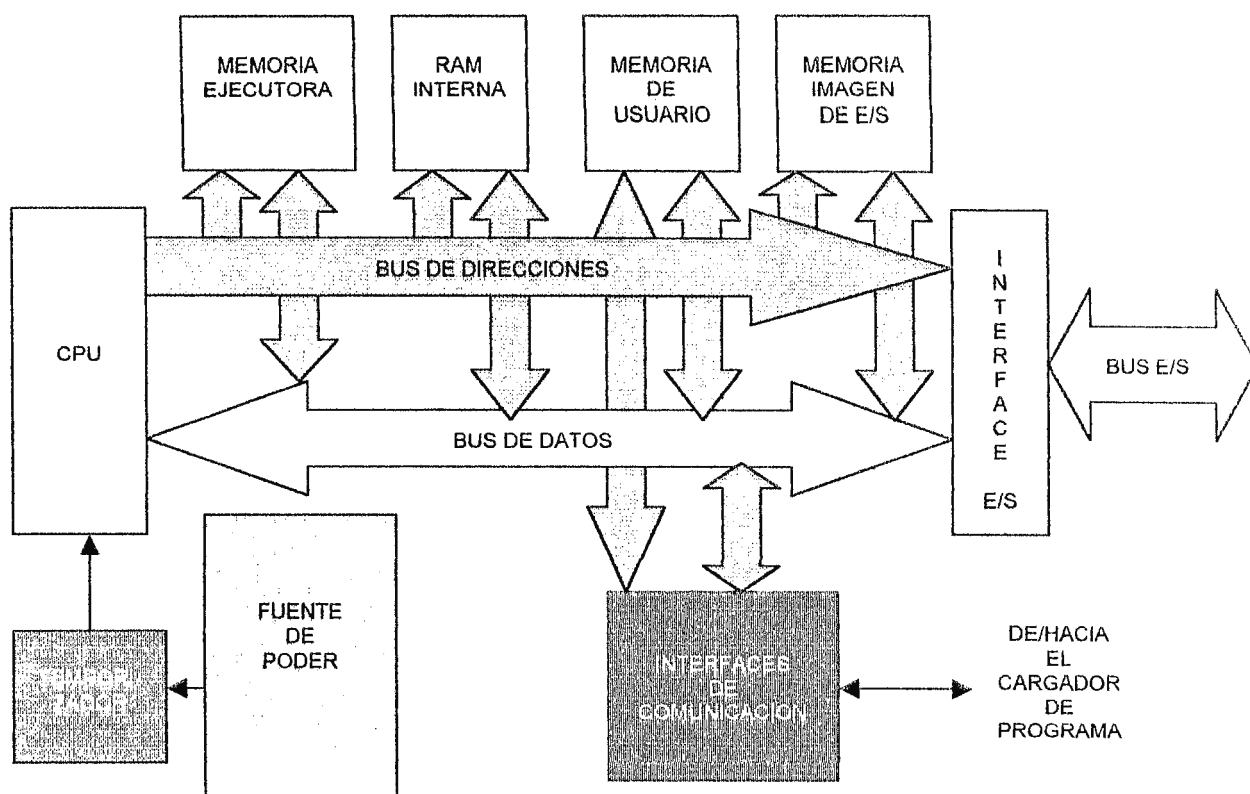


FIGURA 2. 2 DIAGRAMA DE LA SECCION PROCESADORA DEL PLC.

2.2.5 FUENTE DE PODER.

La función básica de la fuente de poder es convertir la potencia de alimentación en un nivel compatible con los dispositivos electrónicos que componen el PLC (típicamente +5V o +/-12V dc). La potencia suministrada es uno de los elementos más críticos de un PLC por dos razones:

1. Una falla en la fuente de poder del PLC puede causar que falle todo el sistema de control.

2. Generalmente contiene componentes de alto voltaje. Una falla en el aislamiento puede generar serios daños o fuego.

Dentro de las consideraciones que se deben tomar en cuenta para la fuente de poder, se contemplan las siguientes:

1. Debe poseer una cubierta que disipe el calor emitido por la fuente previniendo un sobrecalentamiento.
2. La fuente de poder debe ser probada por una agencia certificadora, tal como "Underwriters Laboratories" (UL) o la "Asociación Canadiense de Estándares" (CSA). Estas agencias desarrollan pruebas de temperatura y aislamiento eléctrico para los componentes de las fuentes de poder. Una marca "UL" o "CSA" indica que ésta cumple con los mínimos requerimientos estándar.
3. Deberá cumplir al menos un estándar de inmunidad al ruido. Dos de estos estándares son "NEMA⁴ ICS 2-230" (prueba que consiste en aplicar diferentes tipos de ruido eléctrico a la fuente), y la "IEEE⁵ Std. 472" (prueba que consiste en aplicar un impulso de alto voltaje). Algunas pruebas de ruido pueden ser desarrolladas por agencias de certificación, como UL y CSA. La fuente deberá ser capaz de proteger al PLC de variaciones en el voltaje como caídas de tensión y sobrecargas, las cuales son comunes en la industria.

⁴ NEMA: National Electrical Machine Association.

⁵ IEEE: Institute Engineering Electrical and Electronic.

2.2.6 MEMORIAS

Son dispositivos que sirven para almacenar datos binarios ya sea se forma temporal o permanente.

Las memorias son ampliamente utilizadas en sistemas de control automático basados en circuitos con microprocesadores. Se les puede clasificar según la función que desarrollen de la siguiente forma:

- Memoria de Datos.
- Memoria de Programa.

La información almacenada en la *Memoria de Datos* es de carácter temporal. Aquí los datos se guardan o recuperan según los requerimientos de las operaciones que se ejecutan en el programa. En la *Memoria de Programa* los datos se almacenan de forma permanente. En ella se encuentran las instrucciones en el orden lógico que el sistema en uso debe de seguir.

En los sistemas de control industrial basados en microprocesadores la función de las memorias es imprescindible, y consiste en almacenar los datos y programas que el microprocesador ocupará en algún proceso, permitiendo así, la versatilidad de las funciones a realizar.

Funciones Básicas de las Memorias

Todas las memorias obedecen a un formato general de funciones y basta conocer la lógica de operación de una para entender el funcionamiento de muchas más.

Para que una memoria sea funcional y reconocida por cualquier sistema, debe tener líneas de entrada y salida para una buena comunicación con la circuitería externa.

Las líneas de entrada salida se pueden dividir en tres grupos:

- *Bus de Datos*. Lleva la información desde y hacia la memoria.
- *Bus de Direcciones*. Se utiliza para direccionar⁶ la localización exacta donde se encuentra el dato, y así poder realizar las operaciones de escritura y lectura de una memoria.
- *Bus de Control*. Genera las señales necesarias para poder activar y acceder a la memoria, ya sea para operación de lectura o escritura.

Tipos de Memorias.

Hay una cantidad de memorias que difieren según la aplicación para la que fueron desarrolladas, como ya se menciono, las memorias pueden ser de almacenamiento temporal o permanente, a partir de aquí se les clasifica en:

- Memorias de Solo Lectura (ROM).
- Memorias de Lectura – Escritura (RAM).

2.2.6.1 MEMORIAS ROM

Las Memorias de solo Lectura son diseñadas de tal forma que la información almacenada en ellas no sea cambiada frecuentemente. A este tipo de memoria se le llama *No Volátil*⁷.

⁶ Direccionar: Colocar los niveles lógicos correspondientes a una dirección.

⁷ Volátil: termino para referirse a los dispositivos de memoria que requieren una tensión de alimentación constante para no perder la información almacenada. Si esta tensión se retira, los datos se pierden completamente.

La ROM es utilizada frecuentemente para guardar los programas de inicio, pues como en ella los datos no son volátiles la información no se pierde cuando se retira la alimentación.

En la figura 2.3 se presenta el diagrama de bloques básico de una memoria ROM. En esta figura se observa como la capacidad de la memoria se representa con los números que están dentro del bloque. Estos números indican que hay ocho localidades con capacidad de almacenar cuatro bits⁸ (8×4); esto significa que la capacidad de esta memoria es de 32 bit de almacenamiento.

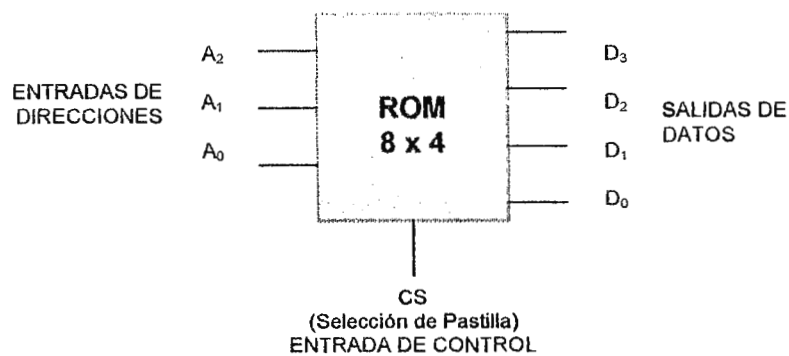


FIGURA 2.3. DIAGRAMA DE BLOQUE DE LA ROM.

En la tabla 2.3 se muestra la distribución de los datos dentro de la memoria suponiendo que la ROM ha sido programada con esta información.

⁸ Bit: Unidad de información en el sistema de números binarios. Puede tener el valor de 0 o 1.

LOCALIDAD	A ₂	A ₁	A ₀	D ₃	D ₂	D ₁	D ₀	D ₂ - D ₀
0	0	0	0	1	1	1	0	6
1	0	0	1	1	0	1	0	2
2	0	1	0	0	1	0	1	5
3	0	1	1	1	1	1	1	7
4	1	0	0	1	0	0	1	9
5	1	0	1	1	0	1	1	3
6	1	1	0	0	0	0	0	0
7	1	1	1	1	1	0	0	4

TABLA 2.3. DATOS PROGRAMADOS.

La obtención un dato (o lectura) de la memoria ROM se realiza en dos pasos:

Paso 1: Se coloca la dirección de la localidad que se quiere acceder.

Paso 2: Se colocan en los pines de control las señales necesarias para activar la memoria.

Asegurando así que el dispositivo que solicita la información, la reciba a través del bus de datos.

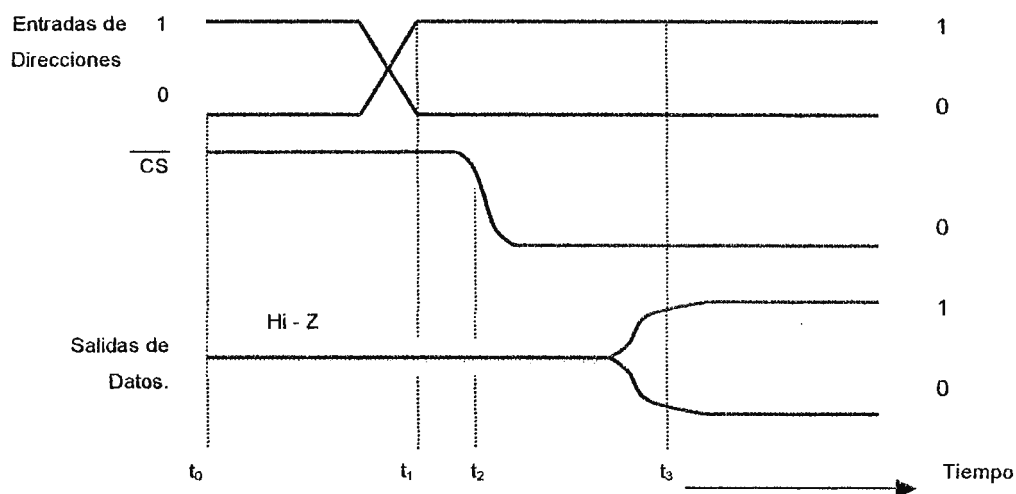


FIGURA 2.4 DISTRIBUCION COMUN DE UNA OPERACIÓN DE LECTURA EN LA ROM

El bus de control está integrado por una entrada **CS** como se muestra en la figura 2.3. CS, significa "selección de pastilla" (chip select), con un "1" lógico toda la memoria se desactiva y con un "0" lógico la memoria se activa. Para habilitar CS, la dirección seleccionada debe estar presente en los pines de dirección de la memoria, de lo contrario se obtendrá un dato erróneo.

En la figura 2.4 se describen los tiempos de acceso y las señales necesarias que deben estar presentes para una operación de lectura.

Tipos de ROM

Existen varios tipos de memorias ROM, éstas difieren entre sí, en la forma en que son programadas, borradas y reprogramadas. Los diferentes tipos de ROM se describen a continuación.

- **ROM PROGRAMABLES CON MASCARA (MROM).** Son programadas por el fabricante durante el proceso de manufactura según especificaciones de su cliente. Se conocen simplemente como ROM y presentan la desventaja de no poderse reprogramar.
- **ROM PROGRAMABLE (PROM):** Puede ser programada por el usuario una solo una vez, utilizando un grabador de memorias. Luego de su programación se comportará de igual forma que la MROM. Ejemplo de estas memorias es la 7641, con capacidad de 512 x 8.
- **ROM PROGRAMABLE Y BORRABLE (EPROM).** Estas memorias pueden ser borradas y reprogramadas por el usuario, con pulsos eléctricos. Para programarlas se requiere una circuiteria de soporte conocida como "Grabador de Memorias EPROM". El proceso de grabación consiste en aplicar una tensión DC mucho mayor que la de alimentación ($5 V_{DC}$), que puede estar en el orden de los 12 V_{DC} hasta 50 V_{DC} ; este valor depende de las

especificaciones técnicas de la memoria. Esta tensión deberá ser aplicada durante un intervalo de tiempo de aproximadamente 50 ms, si el tiempo es menor se corre el riesgo de que el dato no se grabe debidamente.

La EPROM presenta la desventaja de que para borrarla hay que extraerla del sistema en que se encuentra para ponerla en el borrador de memorias. Otra desventaja es que no se puede borrar el contenido de una celda en particular.

Un ejemplo de EPROM es la 2716 con una capacidad de 2K x 8.

- **ROM PROGRAMABLE Y BORRABLE ELECTRICAMENTE (EEPROM).** Esta memoria es una versión mejorada de la EPROM, con la ventaja de que se programa, borra y reprograma por pulsos eléctricos. Para borrarla no es necesario retirarla de la base en circuito del cual forma parte. Las celdas pueden ser borradas de forma particular y en conjunto. El tiempo de programación para este tipo de memoria es de 10 ms por palabra⁹ de datos, lo cual la hace más rápida en comparación con el tiempo requerido por las EPROM. Un ejemplo de EEPROM es la 2816 con una capacidad de 2K x 8.

2.2.6.2 MEMORIAS RAM

Las Memorias de Lectura – Escritura se utilizan en los sistemas de control con microprocesadores para almacenar datos de forma temporal. Para almacenar los datos en la RAM no se requieren dispositivos especiales, el mismo microprocesador puede almacenar un dato directamente con solo mandar las señales necesarias para el acceso a la memoria.

Una desventaja de las RAM es que son volátiles, y pierden toda información cuando se retire la fuente de voltaje que las alimenta.

⁹ Palabra: Grupo de bits en una memoria que representa instrucciones o datos de algún tipo.

El diagrama de bloques de una memoria RAM se muestra en la figura 2.5. Se observa como el bus de datos y el de direcciones se asemejan mucho al de la ROM. El bus de control esta constituido por entradas **R/W** y **CS**.

Por la forma en que están constituidas las celdas de una memoria RAM, estas se pueden clasificar de dos formas:

- RAM Estática.
- RAM Dinámica.

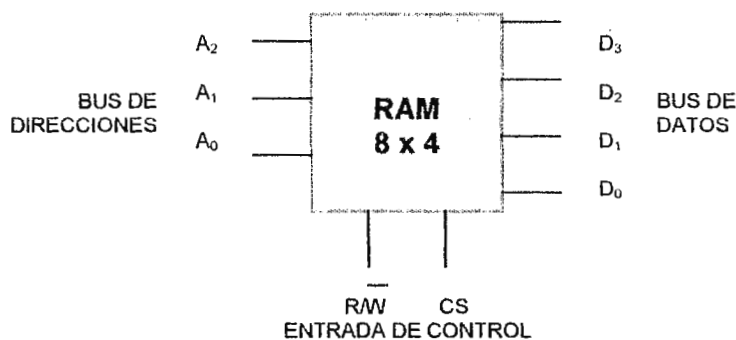
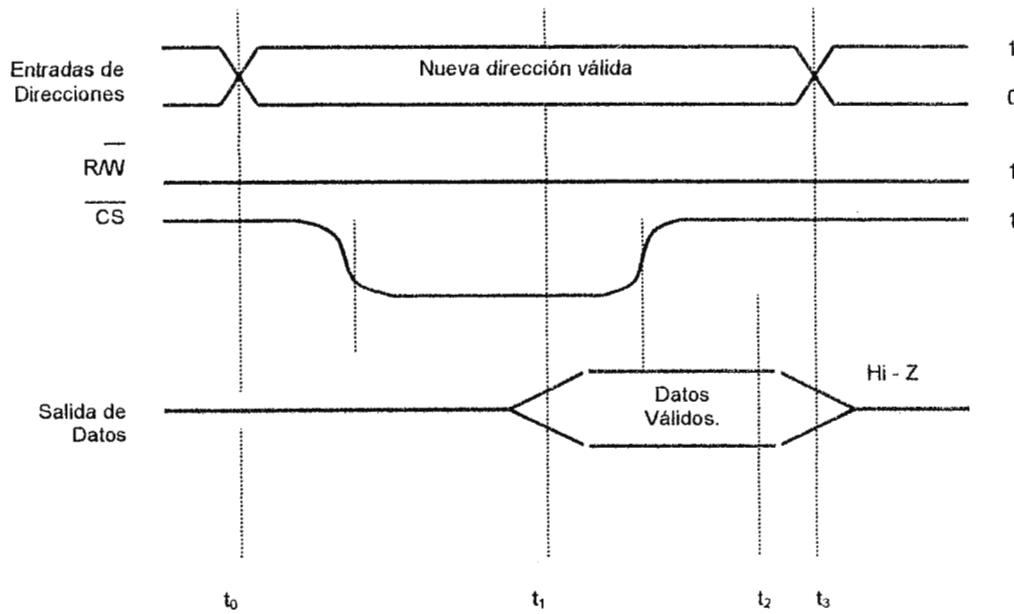


FIGURA 2.5. DIAGRAMA DE BLOQUE DE LA RAM.

- **RAM ESTÁTICA.** Una RAM estática tiene por celdas multivibradores biestables (Flip-Flop), los cuales pueden mantener un dato por tiempo indefinido siempre y cuando la fuente de voltaje que los alimenta este conectada.

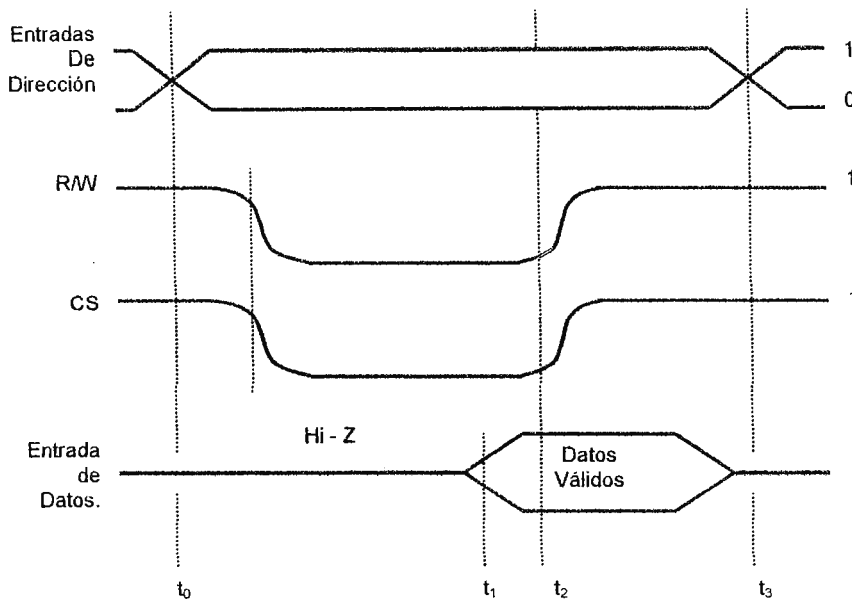
La figura 2.6 muestra los diagramas de tiempo para las operaciones de lectura escritura. En la figura 2.6a se muestra el ciclo de lectura, en el que se observa el valor que debe tener cada entrada, **R/W** debe de tener un 1 lógico, Y **CS** debe de tener un 0 lógico y la dirección debe estar presente y estable en el bus de direcciones, para asegurar la validez de los datos en el bus de datos. De

igual forma en el ciclo de escritura el bus de direcciones es estable, **CS** está activo y **R/W** pasa a un estado bajo, tal como se observa en la figura 2.6b.



[a] Ciclo de Lectura.

FIGURA 2.6 DISTRIBUCION COMUN DE LA RAM ESTATICA.



[b] Ciclo de Escritura.

FIGURA 2.6 DISTRIBUCION COMUN DE LA RAM ESTATICA.

■ **RAM DINAMICA.** Debido a la complejidad de su circuitería interna (Flip-Flop), las RAM estáticas se pueden volver muy costosas, para resolver esta desventaja se utilizan las memorias RAM Dinámicas.

La celda de una memoria RAM Dinámica básicamente está constituida por un Capacitor que es el elemento de almacenamiento y un transistor Mosfet que sirve como interruptor para el manejo de entrada y salida de los datos.

La RAM Dinámica aventaja a la RAM Estática, en el costo de producción, ya que sus celdas son mucho más fáciles de fabricar y por consiguiente es mucho más barata. La desventaja más grande que posee la RAM Dinámica es que la información que ella almacena debe de ser refrescada constantemente por períodos aproximados de 2 ms, debido al tiempo de

carga y descarga del capacitor. Este proceso de refrescamiento incrementa la circuitería para controlar a la memoria y el costo del sistema a implementar. Por el diseño de construcción de la celda Dinámica, ésta sólo podrá almacenar un bit por localidad.

2.2.7 ENTRADAS/SALIDAS DE LOS PLCs

2.2.7.1 SISTEMAS DE ENTRADA/SALIDA.

Existen tres sistemas de entradas salidas para los PLCs:

1. *Entradas/Salidas Directas.*
2. *Sistemas De Entradas/Salidas Paralelas.*
3. *Sistemas De Entradas/Salidas Seriales.*

Entradas/Salidas Directas. Como el nombre lo indica es la forma de tomar las señales directamente de las entradas/salidas del PLC. El número de señales que se utilizan con este método es igual al número de líneas de entrada/salida que posee el PLC. Este acceso generalmente es usado en muchos PLCs pequeños que tienen todos los circuitos de entrada/salida en la misma unidad física que el procesador, (a este sistema se le llama a veces "entrada/salida interna"). El bajo costo es la principal ventaja de este sistema. Sin embargo son poco flexibles, ya que debe cambiarse el procesador según cambia el número de entradas/salidas. Es recomendable utilizar este tipo de PLCs cuando no se requieren más de 64 entradas/salidas; de lo contrario, deben utilizarse sistemas orientados a buses (paralelos o seriales).

Sistemas De Entradas/Salidas Paralelas. En este sistema un bus E/S paralelo, sale de la interfaz E/S del procesador, y módulos individuales de entrada/salida se conectan

a este bus. Los módulos E/S contienen la circuitería necesaria para decodificar las señales del bus y convertirlas en los niveles de voltaje que pueden manejar las diferentes cargas de la maquinaria o proceso que se controla. Generalmente los módulos poseen cierto número de puntos que pueden utilizarse como entradas o salidas, a esta multiplicidad de entradas/salidas se le llama "Modularidad de los sistemas de E/S". Comercialmente la modularidad puede ser de 4, 8, 16 o 32 puntos. Cuanto mayor sea la cantidad de puntos E/S de un módulo, se reduce el costo por punto de E/S, y se reduce también el espacio requerido un número determinado de entradas/salidas. Además a mayor número de puntos asignados como salida, menor es el valor de las cargas que pueden ser manejadas. Un diagrama de bloques de un sistema de entradas/salidas paralelas es mostrado en la figura 2.7.

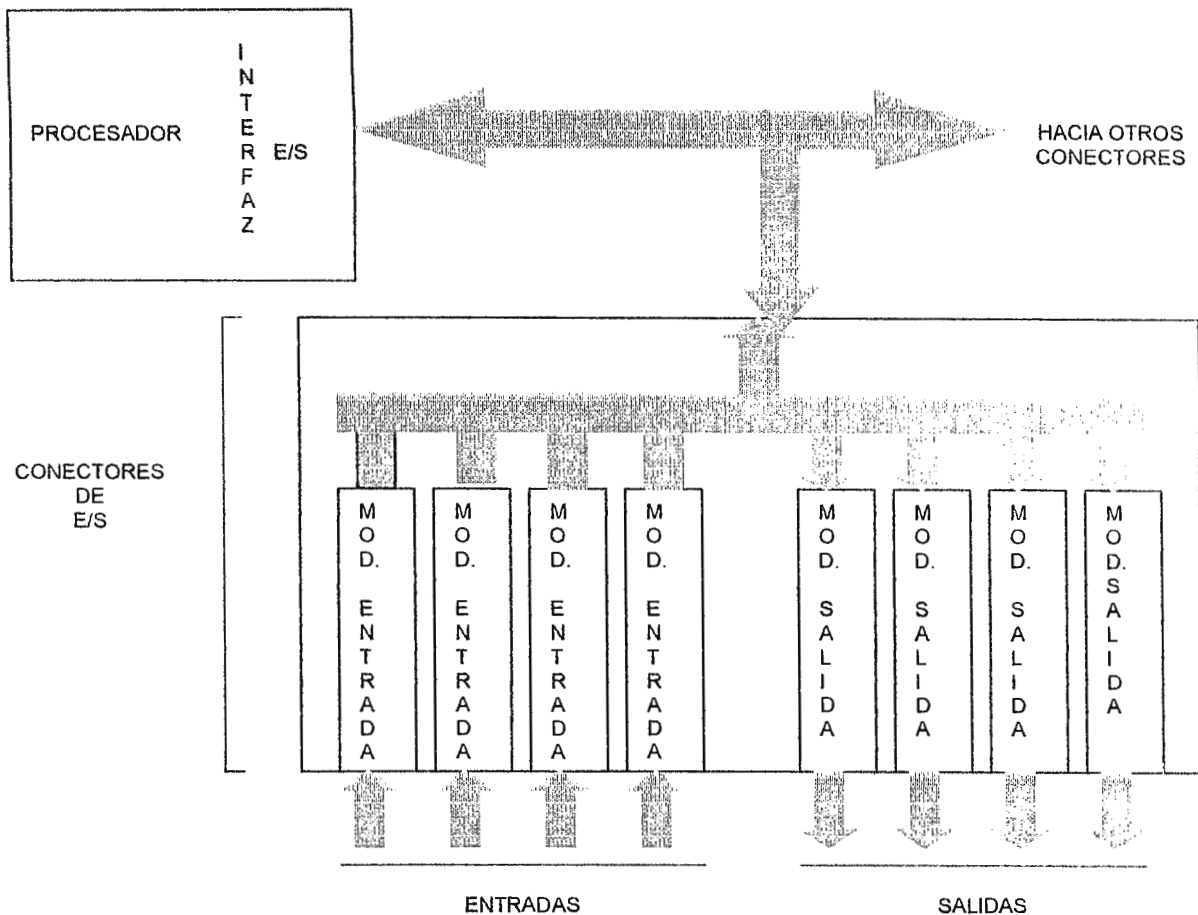


FIGURA 2.7. SISTEMA DE ENTRADAS/SALIDAS PARALELAS

Sistemas De Entradas/Salidas Seriales. La distancia que se puede cubrir con los sistemas paralelos es bastante limitada, generalmente menor a 15 metros. Si las máquinas son (por ejemplo) de 30 metros de larga, se necesitarán por lo menos dos PLCs. Con los sistemas seriales se resuelve este problema, al transmitir la información de las entradas/salidas sobre un enlace serial, el cual se extiende fácilmente sobre largas distancias (300-3000 mts). Generalmente un bus serial sale del procesador y se

conecta un bus paralelo a través de un convertidor serie-paralelo. Ya que el bus serial posee menos alambres que deberán conectarse a las cargas, es una aplicación más económica (ver figura 2.8). Algunos de los más nuevos modelos de PLCs usan solamente sistemas seriales donde cada módulo de entrada/salida se acoplan directamente a la conexión serial.

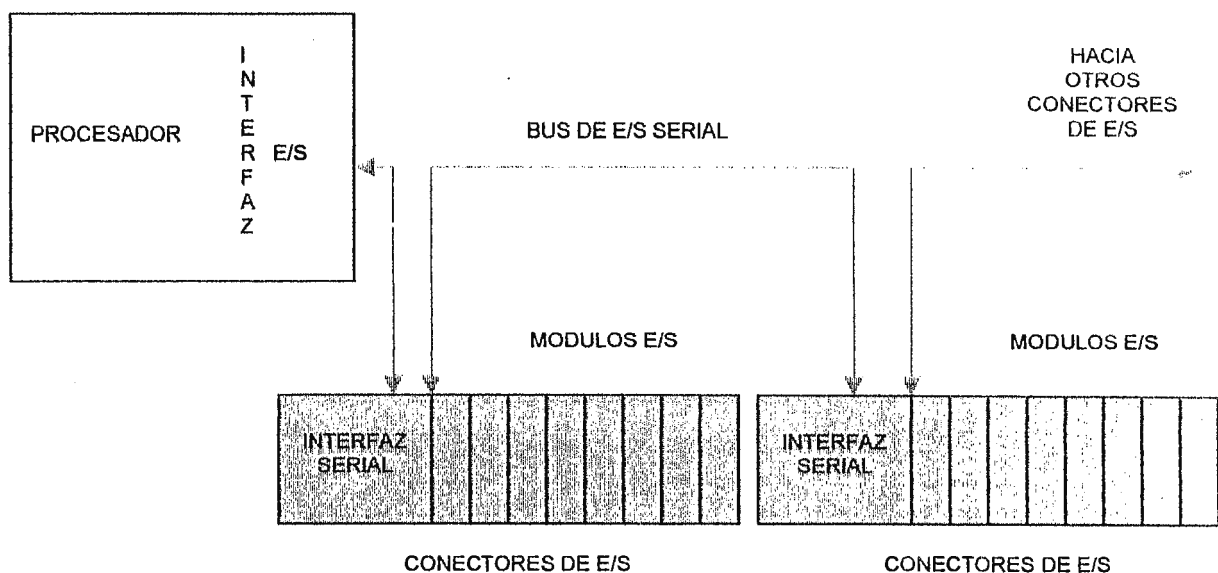


FIGURA 2.8. DIAGRAMA DE BLOQUES DE SISTEMAS SERIALES DE ENTRADA SALIDA

2.2.7.2 MODULOS DE ENTRADA/SALIDA.

Un módulo de entrada salida convierte señales y provee del aislamiento necesario entre las señales internas de bajo nivel y las de alto nivel del exterior. Existe una gran variedad de circuitos de entrada/salida capaces de manejar una cantidad considerable de carga y evaluar el estado de los sensores. La mayoría de estos circuitos de entrada/salida pertenecen a una de las cinco categorías que se mencionarán en esta sección. Los aislamientos eléctricos en los módulos de entradas/salidas poseen características de

seguridad para condiciones críticas. Si los aislamientos fallan el PLC podría calentarse y provocar un peligroso cortocircuito. Los aislamientos de un módulo de entrada/salida deberán resistir una tensión de prueba de 1000Vac rms más la tensión de trabajo del circuito de campo mínima por lo menos un minuto.

Los principales tipos de módulos de entrada/salida se describen a continuación.

- **Salidas "Pilot Duty" (Servicio Piloto).** Las salidas de este tipo generalmente se usan para manejar cargas con corrientes electromagnéticas elevadas tales como solenoides, válvulas, relés, y motores de arranque; estas cargas son altamente inductivas.
- **Salidas De Propósito General.** Generalmente son para bajos voltajes y corrientes pequeñas y se utilizan para manejar luces indicadoras u otras cargas no inductivas. Algunos de estos módulos incluyen supresión de ruido.
- **Entradas Discretas.** Circuitos de este tipo se usan para sensar el estado de interruptores, botoneras, y otros sensores discretos. La supresión de ruido es de gran importancia para prevenir falsas indicaciones; de entradas encendiéndose o apagándose; debido al ruido. Entre más inmune al ruido es una entrada, más lenta se vuelve debido al filtrado requerido. El nivel de inmunidad al ruido necesario versus la velocidad requerida deberán ser evaluadas por el usuario.
- **Entradas/Salidas Analógicas.** Las entradas analógicas se utilizan para manejar las señales provenientes de los dispositivos como termocuplas y sensores de presión, u otros elementos que provienen señales de voltaje o corriente derivadas de procesos variables. Las salidas pueden ser utilizadas para manejar dispositivos como voltímetros, válvulas, controladores de servomotores y otros. Los circuitos E/S analógicos poseen su propio convertidor digital/analógico (D/A) para las salidas o

analógico/digital (A/D) para las entradas. Dichos convertidores transforman la señal digital de los buses E/S a las señales analógicas requeridas por los dispositivos.

Existen cuatro características concernientes a los circuitos de E/S analógicas referentes a los niveles de voltaje y corriente:

- (1) Tiempo de conversión. Es la rapidez con que se transforma la señal proveniente de los dispositivos.
- (2) Precisión. Es una medida de la exactitud con que se puede sensar o manejar una señal analógica.
- (3) Desplazamiento. Es una medida de la precisión con que el convertidor cambia en función del tiempo o la temperatura.
- (4) Resolución. Es una medida de la cantidad de niveles de voltaje que el procesador es capaz de representar, lo cual se especifica en términos de bits. Un convertidor de 8 bits puede representar 256 (2^8) diferentes voltajes; uno de 16 bits es capaz de representar 4096 (2^{16}) niveles, etc.

La resolución y la precisión son totalmente independiente una de la otra. Así, es posible construir un circuito analógico con mucha precisión, pero muy baja resolución. Por ejemplo, el convertidor A/D de 1 bit llamado "detector de umbral", el cual activa o desactiva este único bit al detectar que cierto voltaje alcanza un valor específico; es entonces un circuito muy preciso, pero con baja resolución.

- **E/S De Propósitos Especiales.** Son utilizados en las interfaces de los PLCs para verificar tipos especiales de circuitos tales como servomotores, motores paso a paso, lazos PID (Proporcional Integral Derivativo), contadores de pulsos de alta velocidad, decodificadores de entrada, multiplexores para display, y teclados. Las ventajas de utilizar estos módulos de E/S radican en tres puntos específicos:

- (1) La velocidad del PLC incrementa porque este tiene que ejecutar menos niveles lógicos para controlar los dispositivos.
- (2) El PLC puede fallar, y el control de los dispositivos no se pierde porque los módulos de E/S no han fallado.
- (3) Una amplia variedad de dispositivos puede ser interfazada con el PLC.

Otra forma más sencilla de clasificar los módulos de entrada/salida es de acuerdo a la naturaleza de las señales que se manejan, es decir, análoga o digital. Estos módulos además se subdividen separando las entradas y salidas, generando así cuatro tipos de módulos los cuales se mencionan a continuación junto con las etapas básicas que los conforman:

- **Módulos de Entrada Digital:**
 - Etapa de aislamiento eléctrico.
 - Etapa de Enganche de Datos.
- **Módulos de Entrada Análogas:**
 - Etapa convertidora Análoga/Digital.
 - Etapa de aislamiento eléctrico.
 - Etapa Transductora.
- **Módulos de Salida Digital.**
 - Etapa de aislamiento eléctrico.
 - Etapa de potencia.
- **Módulos de Salida Análogos.**
 - Etapa convertidora de análogo a digital.
 - Etapa de aislamiento eléctrico.

- Etapa de potencia.

Muchos PLCs combinan en un solo módulo las E/S digitales y en otro las E/S análogas.

2.2.8 PROGRAMACIÓN DEL PLC

2.2.8.1 SOFTWARE PROCESADOR

El hardware de la CPU del PLC no difiere significativamente del de una computadora ordinaria, quien hace al PLC especial es el software. El programa monitor es el software del que es provisto el PLC por el fabricante, el cual se encarga de ejecutar el programa del usuario.

El programa monitor determina que funciones serán desarrolladas por el programa de usuario, como serán atendidas las entradas/salidas, y como se comportará el PLC durante su operación normal y fallas en el sistema.

Programa Monitor.

Un modelo simplificado del programa monitor se presenta en la figura 2.9. Como los diferentes PLCs pueden desarrollar estas funciones de diferentes maneras, esto generará grandes diferencias entre los programas monitores. Por ejemplo algunos PLCs pueden desarrollar el diagnóstico como una función aislada del programa monitor, en tanto que otros pueden desarrollar el diagnóstico en línea, es decir, mientras se desarrolla el programa de usuario.

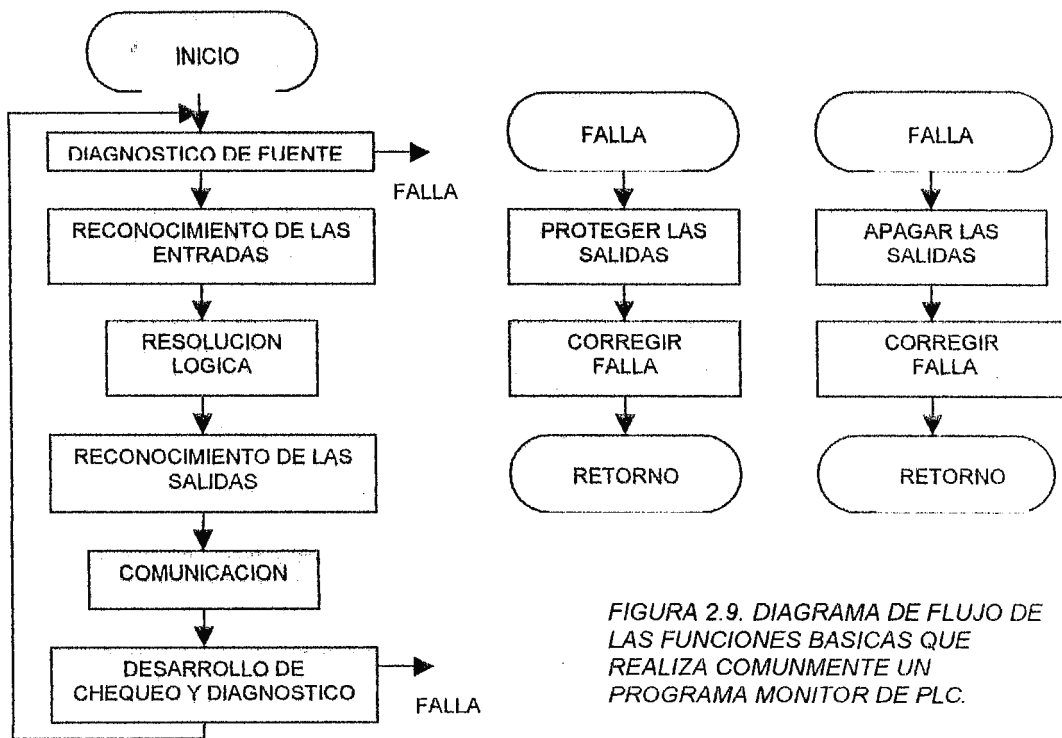


FIGURA 2.9. DIAGRAMA DE FLUJO DE LAS FUNCIONES BASICAS QUE REALIZA COMUNMENTE UN PROGRAMA MONITOR DE PLC.

Multitareas.

Algunos PLCs son capaces de realizar multitareas con un solo procesador. Existen PLCs en particular que utilizan Secuencias de Funciones Gráficas (SFCs) las cuales poseen una capacidad limitada de multitareas. Las multitareas toman diferentes formas, dos de las cuales son: Time-driven (Manejando los tiempos) y Event-driven (Manejando los eventos).

En algunas multitareas Time-driven el usuario escribe los programas y asigna entradas/salidas para cada tarea por separado. El usuario entonces puede configurar el programa para que desarrolle cada tarea separadamente. Esta característica permite el control de tiempos críticos en el sistema; tales como control de movimientos a altas

velocidades o detección de errores en la maquinaria que se controla; para lo que se requiere correr subrutinas¹⁰ muchas veces por segundo, a la vez que controla tareas no críticas; tales como dar servicio a las luces indicadoras, que consiste en subrutinas mucho más lentas.

Las multitareas Event-driven también llamadas "Interrupt-driven" (Manejando las interrupciones) se comportan de forma similar. En este caso el usuario define un evento en particular, tal como un estado de entrada variable o una salida apagándose, permitiendo que cada tarea se realice. Algunos sistemas multitareas poseen tareas específicas para acceder al control de una variable, como por ejemplo el control de los puertos de entrada salida. Este sistema se utiliza cuando en la programación de varias tareas es necesario acceder a las mismas variables. Pero puede ser difícil determinar que tarea utiliza una determinada variable al depurar un programa.

2.2.8.2 SOFTWARE DE USUARIO

Este es el software que se utiliza para escribir y guardar en la memoria del usuario en secuencia lógica las ordenes requeridas para controlar la maquinaria o proceso. El software de usuario está compuesto por lenguaje de programación y por los datos configurados a utilizarse. Los datos configurados contienen la información que se comunica al procesador, configurada de forma que sea comprensible para éste, dichos datos complementan las instrucciones dadas mediante el lenguaje de programación.

¹⁰ subrutina: Secuencia de instrucciones creada para realizar operaciones complejas.

Configuración.

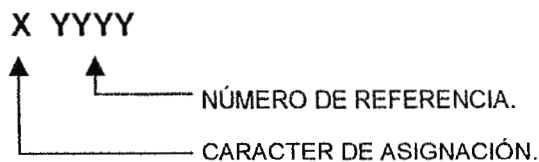
El típico proceso de configuración consiste en asignar los puntos de entrada/salida a ciertas clavijas¹¹ en particular, definir cuanta memoria y entradas/salidas tendrá el procesador, y asignar localidades de memoria específica para las diferentes tareas; y en general determinar las diferentes funciones que interactúan al cargar un programa. No todos los PLC poseen una configuración muy elaborada, pero su capacidad y eficiencia dependen mucho de este factor. Por ejemplo, un PLC que supervisa solo la memoria y entradas/salidas que están configuradas puede ser mucho más rápido que un PLC que supervisa toda la memoria y entradas/salidas independientemente de que éstas estén o no siendo utilizadas. Es posible que el usuario necesite conocer como se encuentra configurado el procesador, especialmente si desea reprogramarlo.

Lenguajes.

Puesto que en la industria de control moderna se solicita cada vez más capacidad para el control de interfaces, comunicaciones, adquisición de datos, y supervisión y control de procesos; se necesita que los lenguajes sean capaces de implementar estas funciones. Entonces es crucial considerar varios aspectos importantes, como los descritos a continuación, al diseñar un lenguaje.

Variables. Cuya función es permitir al usuario acceder a las entradas/salidas y los datos internos. Típicamente presentan el siguiente formato:

¹¹ Clavija: Terminal individual de cada línea que componen un bus o un puerto.



El caracter de asignación indica el tipo de variable, tal como “bit de entrada”, “palabra de salida”, “bit interno”, o “palabra interna”. El número de referencia indica la variable específica. Algunos PLCs permiten al usuario representar el mismo dato de diferentes formas, incluyendo bits, palabras, tablas de bits, tablas de palabras, archivos y bytes¹².

Los lenguajes que permiten que el mismo dato sea representado de diferentes formas son mucho más flexibles, y permiten al usuario escribir programas más entendibles y eficientes. Algunos lenguajes permiten al usuario utilizar nombres de variables en vez de números de referencia, pero esto vuelve al software más lento.

On-line versus Off-line. Un lenguaje “On-line” es el que puede ser programado de forma interactiva con el PLC. Un lenguaje “Off-line” requiere que el programa sea generado separadamente y luego cargado en el PLC. El usuario debe ser cuidadoso al utilizar un lenguaje “On-line”, cuando el PLC está controlando un proceso. Al cambiar inadvertidamente cualquier parte del programa puede estropearse el proceso que está siendo ejecutado por el PLC.

Control de flujo. Esta característica determina que tan fácil se tomarán las decisiones y se controlará la ejecución del programa. Algunos programas solo poseen instrucciones de salto condicional, mientras que otros permiten desarrollar subrutinas y lazos. No es recomendable utilizar demasiadas instrucciones de salto, pues esto vuelve al programa

¹² Byte: Terminó especial que se usa para palabra de ocho bits.

poco legible y difícil de depurar. (En algunos lenguajes las instrucciones de salto son evitadas a propósito).

Funciones. Las funciones con las que cuenta los lenguajes varían de un PLC a otro. Algunos lenguajes poseen solo las instrucciones mínimas: lógicas, temporizadores, y funciones de memoria; mientras que otros poseen instrucciones adicionales, como matrices, control de registros, funciones matemáticas, etc.

Mantenibilidad. Se refiere a la facilidad de depuración y modificación del lenguaje.

Velocidad. Algunos lenguajes tardan más en ejecutarse que otros. Utilizar un lenguaje eficaz en el control de aplicaciones en tiempos críticos puede volver muy difícil la programación.

Eficiencia. Esta es una medida de la cantidad de memoria que requiere un lenguaje para implementar una determinada función. La eficiencia generalmente se mide en "palabras de memoria por función". El precio de algunos PLCs se establece en base a su capacidad de memoria y eficiencia, pues estos factores determinan el hardware adicional requerido para realizar alguna función específica.

Ejemplos de lenguajes utilizados.

a. Lenguaje escalonado. Este es el principal lenguaje de PLC, el cual tiene muchas ventajas:

1. Este es fácilmente entendible por los obreros expertos familiarizados con control por relés. Sin embargo carecen por lo general, de una buena secuencia de instrucciones de control.
2. Estos despliegan gráficas de la ejecución del programa para mostrar la secuencia por medio de un diagrama escalonado, volviendo fácil su depuración.

3. Los programas se ejecutan rápidamente.

4. Son eficaces en el control secuencial.

b. Lenguaje Booleano. Este lenguaje generalmente se usa en PLCs muy pequeños. Las instrucciones Booleanas utilizadas son: AND, OR, NOT, y otras como STORE (guarde) y RECALL (llamada); utilizadas para describir un programa lógico. Este lenguaje no es de fácil depuración, a menos que se ejecute paso a paso para que el efecto en las entradas/salidas pueda visualizarse. Si bien es rápido, el lenguaje booleano carece de funcionalidad.

c. Lenguajes de alto nivel. Fortran, BASIC, y C, son algunos ejemplos. Estos lenguajes pueden ser muy poderosos usados de igual forma que en computadoras convencionales. Estos lenguajes permiten realizar un excelente control secuencial, son muy funcionales, y poseen una velocidad aceptable. Además permiten tener acceso a diferentes tipos de variables. Si bien, la CPU de algunos PLCs permite al usuario programar en lenguaje de alto nivel, otros PLCs ofrecen este amigable tipo de lenguaje a través de un módulo coprocesador que posee una CPU destinada especialmente a interpretar y ejecutar el lenguaje, separada del procesador principal. Pero por lo general los obreros que laboran en las plantas no lo conocen, y es difícil monitorear la ejecución de los programas en tiempo real.

d. Lenguajes de estado. Son por ejemplo las secuencias de funciones gráficas (SFCs) que permiten controlar el programa en términos de los estados y las condiciones de entrada/salida de la maquinaria, necesarias para realizar la transición de un estado al siguiente. Este lenguaje puede ser una herramienta que permita a un ingeniero describir un largo y complejo programa de control que ya se encuentra escrito en algún otro lenguaje, como el "escalonado". Una desventaja de este lenguaje es que todos los

estados de las máquinas y sus transiciones deben ser definidos exhaustivamente. Por lo general los obreros que laboran en las plantas no lo conocen, y es difícil monitorear la ejecución de los programas en tiempo real.

2.2.8.3 PROGRAMACION

La forma en que se introduce el programa de usuario es determinada por el "cargador de programa", es decir el equipo de programación de que está provisto el PLC. La mayoría del tiempo que se utiliza en programar y depurar, interactuando con el programa que está cargado en el PLC (programa monitor). Este programa puede brindar ayuda al usuario evitándole mayores esfuerzos. Debido a estas razones es importante que el usuario se familiarice con la variedad de funciones que ofrece el "cargador de programa". A continuación se describen las más comunes.

Programación. La primordial importancia del "cargador de programa" es generar un proceso para introducir programas. Esta función debe ser estudiada detenidamente puesto que determinará el tiempo requerido para escribir un programa. La mayoría de fabricantes de PLC ofrecen software's amigables al usuario. Además algunos tipos de "cargador de programa" contienen sofisticados programas depuradores que reducen el tiempo requerido para detectar y eliminar errores de los programas.

Monitoreo. Los "cargadores de programas" también están provistos de programas de monitoreo ejecutables en tiempo real (o muy cercanos al tiempo real). Ciertos "cargadores de programas" muestran las secuencias seguidas mediante un diagrama de relés, mientras que otros solo muestran al usuario la forma en que cambian las entradas/salidas.

Almacenamiento del programa. Algunos PLCs están provistos de equipo de lectura/escritura adicional, que permite guardar el programa de usuario, como por ejemplo disqueteras o caseteras. Se han desarrollado también productos que guardan una revisión histórica de los cambios efectuados en dicho programa, y ofrecen funciones de seguridad para prevenir cambios no autorizados en éste.

Documentación. Ciertos "cargadores de programas" permiten al usuario obtener una copia en disco o impresa de los programas.

2.2.8.4 TIPOS DE "CARGADORES DE PROGRAMAS".

"Hand-Held" (*cargadores manuales*). Estos son pequeños y baratos. Utilizan display¹³ de cristal líquido (LCD) o display de diodos emisores de luz (LED). Estas unidades se utilizan principalmente en aplicaciones donde el costo es de suma importancia, o para pequeñas y sencillas aplicaciones, estos cargadores generalmente poseen una limitada funcionalidad en cuanto a la información que pueden desplegar al usuario.

Computer-Based (*Cargadores basados en computadoras*). Actualmente los fabricantes de PLC ofrecen software's cargadores de programas compatibles con computadoras personales IBM, para los sistemas operativos MS-DOS o Windows. Algunos PLCs requieren de "tarjetas interfaz" para comunicarse con las computadoras.

¹³ Display (Visualizador): Elemento utilizado para exhibir o desplegar datos de forma visible.

2.2.9 INSTRUCCIONES BASICAS DEL PLC.

La programación consiste en la estructuración de una serie de instrucciones que el PLC tiene que ejecutar.

El conjunto de instrucciones básico que utiliza la mayoría de PLC' pueden dividirse en cinco grandes grupos:

- Instrucciones Aritméticas.
- Instrucciones Binarias.
- Instrucciones Digitales.
- Instrucciones de Flujo.
- Instrucciones de Transferencia de Datos.

Una instrucción puede dividirse en dos partes:

- Campo de operación: que es la representación del nombre de la instrucción a utilizar.
- Campo de operando: contiene las indicaciones necesarias para la ejecución de la operación, está define "con que " se va a realizar la función.

Ejemplo:

Para representa una operación de tipo Binaria *AND*, (figura 2.10).

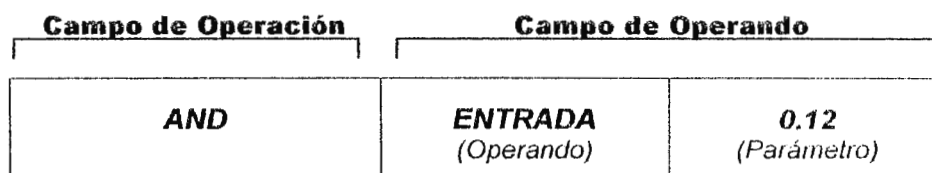


FIGURA 2.10

Instrucciones Aritméticas: Son todas aquellas que realizan las operaciones aritméticas básicas con números enteros.

- SUMA.
- RESTA.
- COMPARACION. (Mayor que, Menor que e Igual que)
- INCREMENTO Y DECREMENTO. (de uno en uno).

Instrucciones Binarias: Son todas aquellas que realizan las operaciones binarias básicas bit a bit.

- AND
- OR
- NOT
- EX - OR
- DESPLAZAMIENTO Y ROTACION. (de bits en una palabra).

Instrucciones Digitales: Son todas aquellas que realizan todas las operaciones binarias básicas entre números enteros.

- AND
- OR
- NOT
- EX - OR

Instrucciones de flujo: Son todas aquellas que realizan las operaciones para estructurar un programa.

- SALTOS CONDICIONALES

- SALTOS INCONDICIONALES.
- SALTOS A SUBRRUTINAS.

Instrucciones de Transferencia de datos: Son todas aquellas que realizan operaciones de transferir los datos entre diferente áreas de memoria.

- TRANSFERENCIA DE REGISTRO A MEMORIA.
- TRANSFERENCIA DE MEMORIA A REGISTROS.
- TRANSFERENCIA DE ENTRADAS A MEMORIA.
- TRANSFERENCIA DE MEMORIA A SALIDAS.
- TRANSFERENCIA DE REGISTROS A SALIDAS.
- TRANSFERENCIA DE ENTRADAS A REGISTROS.

2.2.9.1 REPRESENTACIÓN DEL LENGUAJE DE PROGRAMACIÓN

Se han desarrollado distintos modos de representación para la programación de PLC, con el objetivo de facilitar su utilización. Las dos técnicas más utilizadas son:

◆ **Plano de contactos:**

Este tipo de representación es la que se utiliza para el lenguaje escalonado, (ver sección 2.2.8.2). Se basa en los esquemas de circuitos utilizados en electrotecnia; las operaciones de combinación (lógicas) se realizan en forma de conexiones en serie y en paralelos de contactos. Está constituido por una serie de nudos y bifurcaciones conectadas entre un nudo derecho y uno izquierdo. Entre los distintos nudos están insertadas las entradas y salidas del sistema a controlar.

Esta representación facilita la programación a los usuarios habituados a diseñar sistemas de mando con contactores o relés.

En la figura 2.11 se presenta un ejemplo de representación en Plano de Contactos, correspondiente al lenguaje de programación STEP 5 de SIEMENS.

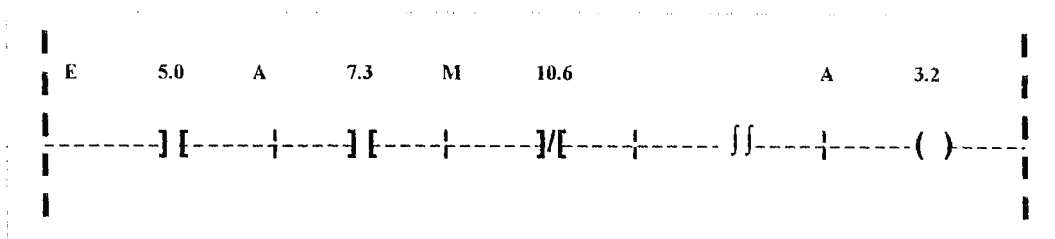


FIGURA 2.11. LA FUNCION Y SE DIBUJA EN FORMA DE CONEXIONES EN SERIE.

◆ Lista de instrucciones:

Este tipo de representación es la que se utiliza para los lenguajes booleanos, (ver sección 2.2.8.2). Está constituido por un lenguaje de programación específico. Se compone de diferentes instrucciones cada una de las cuales representa una orden para el procesador del autómatas. Las diferentes instrucciones están compuestas de abreviaturas mnemotécnicas que identifican la función a ejecutar.

Este modo de operación se parece a los lenguajes de programación de microprocesadores y computadoras personales.

En la figura 2.12 se presenta un ejemplo de representación en Lista de instrucciones, correspondiente al lenguaje de programación STEP 5 de SIEMENS.

0000	:U	E	5.0
0001	:U	A	7.3
0002	:UN	M	0.6
0003	:=	A	3.2
0004	:***		

FIGURA 2.12. LOS OPERANDOS SE CONSULTAN EN FILA Y EL RESULTADO SE COMBINA SEGÚN Y.

2.2.10 VELOCIDAD

La construcción de la CPU determina la flexibilidad del PLC (si este podrá ser modificado y mejorado en el futuro) y sobre todo su velocidad. La velocidad es expresada en términos de que tan rápido el PLC realiza las operaciones de reconocimiento de periféricos y la envía a la memoria; a este factor se le conoce como "tiempo de reconocimiento", y es medida generalmente en milisegundos por millón de palabras. Un PLC rápido es más caro que un PLC lento. Así es importante escoger un PLC con un tiempo de reconocimiento apropiado para una determinada aplicación.

Es importante hacer notar que muchos PLCs desarrollados comercialmente especifican su tiempo de reconocimiento solo en base a contactos y bobinas. Por lo que un programa que use otras funciones como temporizadores, contadores, y funciones matemáticas, puede tomar mucho tiempo para ejecutarse.

2.2.11 CUBIERTA DEL PLC

La forma en que el PLC y sus sistemas de entrada/salida son empacados, es muy importante para determinar si éste es aplicable a una determinada situación. Si el PLC debe instalarse directamente en el piso, o, próximo a la máquina o proceso que controla, deberá tener una cubierta con características diferentes a las de una computadora personal que se encuentra en un ambiente de oficina. La constante vibración, el ruido eléctrico y las variaciones de temperatura en el ambiente que se encuentran en la mayoría de industrias afectará al PLC si su cubierta no ha sido diseñada correctamente.

2.2.11.1 CARACTERÍSTICAS

Las principales características que debe poseer la cubierta de un PLC son:

1. Sistema de extracción de calor.
2. Montaje.
3. Alambrado.

1. Sistema De Extracción De Calor

Como es común con los equipos electrónicos, un PLC debe ser provisto de “disipadores de calor”, para mantener la temperatura interna lo más baja posible, tomando en cuenta los siguientes factores a la hora de diseñar su cubierta.

- *Ventilación.* Un método comúnmente utilizado para disipar el calor en un PLC es enfriándolo mediante corrientes de aire. Dichas corrientes deberán generarse mediante un dispositivo colocado muy próximo al PLC.
- *Generación de Aire Frío.* Un PLC deberá ser provisto de un “Generador de aire frío”, como ventiladores diseñados especialmente para funcionar durante largos periodos de tiempo.
- *Disipadores de Calor.* Se usan para remover el calor de dispositivos específicos, como por ejemplo la fuente de poder. En algunas fuentes, los transistores consumen el 40% de la potencia utilizada por el PLC.
- Los disipadores son construidos de materiales con alta conductividad térmica, como el aluminio.

2. Montaje

Los PLCs pueden ser enclavijados (colgados en algún dispositivo colocado en una pared) o montados en un panel. A bajas temperaturas es más común que los PLCs sean enclavijados, pero se requiere de un disipador de calor.

3. Alambrado

El costo de alambrado en una instalación puede ser mayor que el costo del hardware.

Si el alambrado no se realiza correctamente, a la larga se elevarán grandemente los costos de mantenimiento. Los dos sistemas más comúnmente utilizados son:

- *Alambrado fijo.* Generalmente utilizado para conectar la fuente de poder del PLC, o para alambrear pequeños PLCs con E/S directas.
- *Alambrado removible.* Generalmente usado para módulos E/S.

2.2.11.2 INMUNIDAD AL RUIDO.

Los sistemas de control están expuestos al ruido eléctrico, lo puede generar una serie de fallas, provocando un bajo rendimiento. Afortunadamente los problemas de ruido generalmente pueden prevenirse por medio de ciertas técnicas.

1. **Puesta a Tierra.** Es una de las formas más comunes y efectivas de inmunizar un equipo al ruido. Todos los equipos electrónicos requieren de un buen aterrizamiento para funcionar correctamente. Una técnica recomendada por los fabricantes es conectar a tierra el chasis mediante alambres largos para minimizar la impedancia debido a ruido de alta frecuencia.
2. **Aislamiento.** Deben separarse internamente los dispositivos que generan ruido de los que son susceptibles al ruido. Transformadores de aislamiento pueden ser utilizados en todas las fuentes de potencia. Además, los alambrados lógicos (tales como buses), deben colocarse físicamente separados del resto de las conexiones.

3. Supresión. Un efectivo camino de inmunización al ruido, es reducir el ruido en sí. Las salidas controladas por fase pueden encenderse en varios puntos debido a la forma del Vac, provocando señales con mucho rizado el cual genera ruido de alta frecuencia. Para evitar esto se sustituyen estas salidas, por salidas de cruce por cero, las cuales generan menos ruido, ya que se activan solo en los cruces por cero del Vac. Otro método de eliminación de ruido es colocar supresores de ruido; como fuentes de potencia; a la entrada de dispositivos susceptibles al ruido.

Como ejemplo, en la figura 2.13a se muestra la técnica de filtrado del ruido en la entrada y en la figura 2.13b un supresor de ruido de cargas DC.

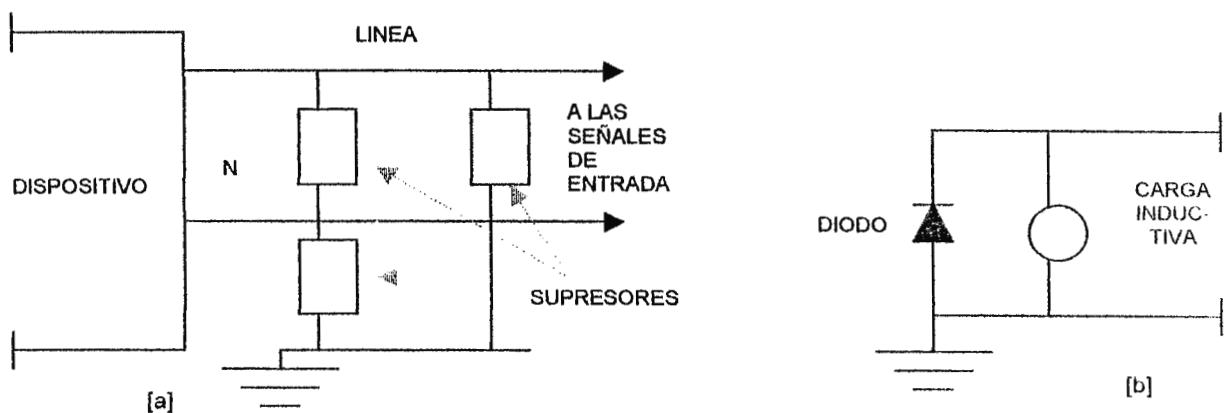


FIGURA 2.13. SUPRESION DE RUIDO.

Muchos PLCs son diseñados con cierto grado de inmunidad al ruido. Pero en ciertas aplicaciones se requieren precauciones adicionales. El filtrado de ruido en las entradas incrementará la inmunidad de algunos circuitos de entrada o fuentes de poder.

Los supresores pueden ser arreglos de capacitores con resistencias, o varistores de metal-oxido. Los supresores entre la línea viva y neutro (o entre +dc y -dc) mejoran la inmunidad al ruido. En muchos casos este método es más efectivo que eliminar el ruido en la fuente, sobretodo cuando se trabaja con cargas dc inductivas. En estos casos un diodo se conecta en paralelo, como se muestra, limitando la sobretensión que ocurre cuando la carga es desactivada.

2.3 EL ENTRENADOR DIDACTICO ASSI

Ahora que se ha desarrollado de forma general la teoría básica de los Controladores Lógicos Programables, se procede a detallar de forma específica en que consiste el "Entrenador ASSI"¹⁴, describiendo y explicando, la estructuración, lógica de funcionamiento y características didácticas de éste.

La realización del diseño, tanto del software como del hardware del entrenador se tratará en el siguiente capítulo, con un enfoque altamente técnico; pero antes es necesario explicar "que es el Entrenador ASSI".

2.3.1 DESCRIPCION GENERAL

El "Entrenador ASSI" es un equipo didáctico para el estudio del "Controlador Lógico Programable" desarrollado para la Universidad Don Bosco, con el fin de mostrar la arquitectura interna y las etapas que constituyen al PLC.

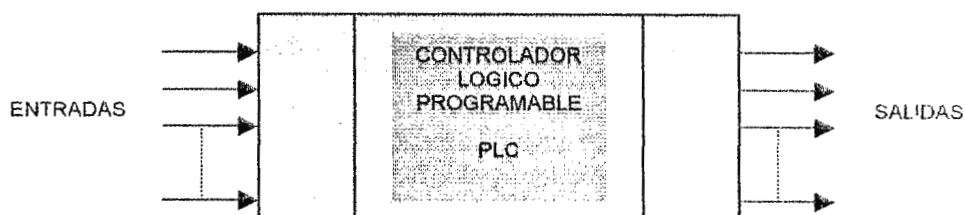


FIGURA 2.14. DIAGRAMA GENERAL DEL PLC "ASSI"

¹⁴ ASSI, es una sigla creada para la identificación del equipo didáctico; y corresponde a las iniciales de los apellidos de los integrantes del grupo.

El ASSI está basado en un sistema de microprocesador, que permite el control del estado de una serie de salidas en función del estado de una serie de entradas (ver figura 2.14).

Por medio de la programación se introducen las órdenes que regulan el estado de las salidas basándose en el estado de las entradas, lo cual lo vuelve bastante flexible, ya que es capaz de controlar una variedad de sistemas simplemente cambiando el programa.

2.3.2 ETAPAS PRINCIPALES

El ASSI posee las funciones básicas de un PLC las cuales se pueden resumir en: procesar la información que se encuentra en sus entradas y dar una respuesta lógica a través de sus salidas. Para que el ASSI realice estas operaciones es necesario programar cada una de las etapas que lo conforman. Cada etapa está constituida por bloques de circuitos electrónicos diseñados para realizar una tarea específica.

El entrenador didáctico se ha dividido en cuatro etapas básicas, éstas se muestran en la figura 2.15.

Explicación del diagrama de bloques:

- *Módulo del programador.* Es el encargado de comunicar al usuario con el PLC de forma directa.

Éste está formado por un teclado y un visualizador, y se utiliza como interfaz entre el usuario y la máquina.

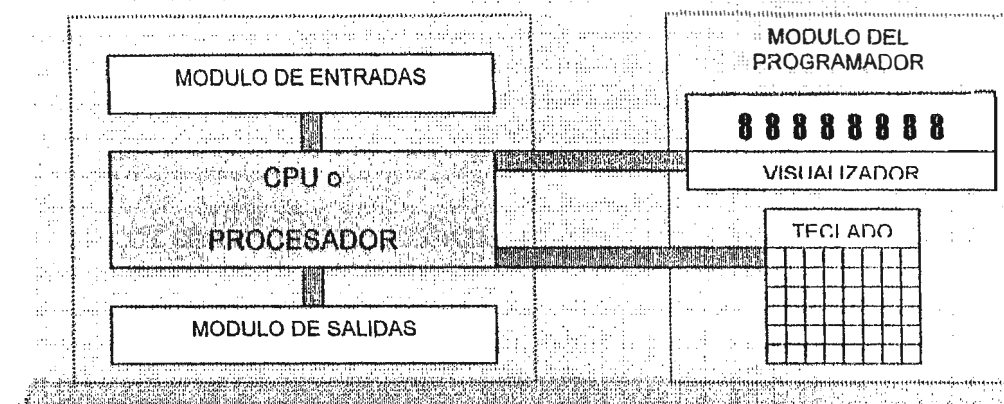


FIGURA 2.15. ENTRENADOR DIDACTICO ASSI

El *teclado* es utilizado para entrar, cambiar y borrar programas guardados en la memoria del usuario.

El *visualizador* muestra al usuario el resultado de cada tecleo cuando se entra o cambia un programa. Éste también provee información cuando existe alguna anomalía con la circuitería.

- *Módulo de entrada:* Es el encargado de transferir la información del mundo exterior al procesador. Las entradas básicas de un Autómata Programable son digitales, y son las que permiten al PLC ASSI adquirir el estado del sistema a controlar.
- *Módulo de salida:* Es el encargado de transferir la información del procesador al mundo exterior. Al igual que las entradas, las salidas más básicas de un PLC son digitales y constituyen el recurso mediante el cual el ASSI es capaz de gobernar los dispositivos externos.

- *CPU (Unidad de Control y Procesamiento) o PROCESADOR*: Se encarga de transferir, procesar y decodificar los datos provenientes del mundo exterior, para tomar una decisión lógica según el programa monitor.

El programa monitor es el encargado de controlar todas las operaciones del entrenador didáctico, éste reside dentro del módulo de la CPU permanentemente. Entre las funciones que realiza están:

1. Proveer al usuario la facilidad de programar el PLC.
2. Controlar las operaciones del entrenador didáctico.
3. Dar a conocer los resultados de los programas que se están ejecutando.
4. Informar de los errores de conexión entre el PLC y el mundo exterior.
5. Supervisar las funciones del PLC.

La CPU o procesador, se encarga de procesar toda la información que el PLC capta por medio del módulo de entrada, para generar en el módulo de salida la respuesta para la que ha sido programado, realizando las siguientes funciones en base al programa:

- Leer el estado de las entradas.
- Procesar las entradas utilizando memorias internas.
- Utilizar temporizadores, contadores y registros de desplazamiento para procesamientos complejos.
- Gobernar las salidas.

Para la realización de este proceso se enumeran a continuación las cuatro funciones principales de la CPU; éstas se muestran en la figura 2.16.

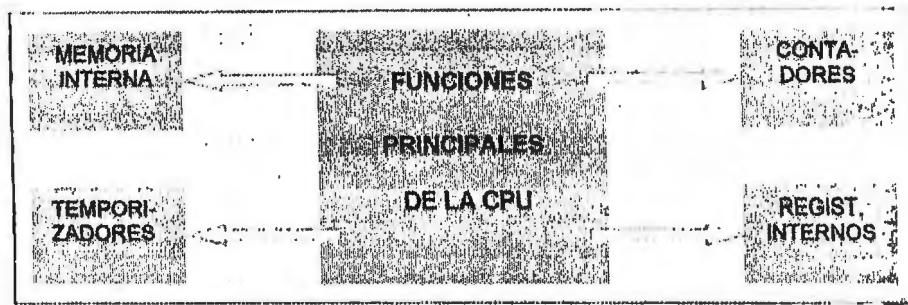


FIGURA 2.16. FUNCIONES PRINCIPALES DE LA CPU.

Cada una de ellas se describe a continuación:

a. *Memorias Internas.*

La memoria interna del ASSI está constituida por:

- Memoria de Programa.
- Memoria de Datos.

La *Memoria de Programa* incorpora el programa que desarrolla el usuario para la aplicación específica en cuestión. Él puede cargar en esta memoria diferentes programas, para diferentes aplicaciones.

La *Memoria de Datos* es utilizada por el ASSI para memorizar todos los valores de las entradas y las salidas digitales y analógicas, asimismo es utilizada por el programador para memorizar sus propios datos.

b. *Temporizadores.*

Los temporizadores son los encargados de realizar las funciones de tiempo con las que se realizan y supervisan los desarrollos de los contadores. Las funciones de tiempo se utilizan para los tiempos de espera, los de supervisión y la construcción de impulsos.

c. *Contadores.*

Mediante las funciones de contador se realiza la cuenta directamente por el procesador central.

d. *Registros Internos.*

El ASSI pone a la disposición del usuario una serie de registros de trabajo para la memorización de los datos intermedios. Se utilizan para almacenar datos, direcciones, códigos de instrucción e información acerca de las condiciones de diversas operaciones. Algunas se emplean como contadores que pueden ser controlados por el software (instrucciones de programa) para llevar registro de actividades, como el número de veces que una cierta secuencia de instrucciones se ha ejecutado.

2.3.3 LOGICA DE FUNCIONAMIENTO DEL ASSI

Para explicar la lógica de funcionamiento del ASSI se utilizará el esquema presentado en la figura 2.17.

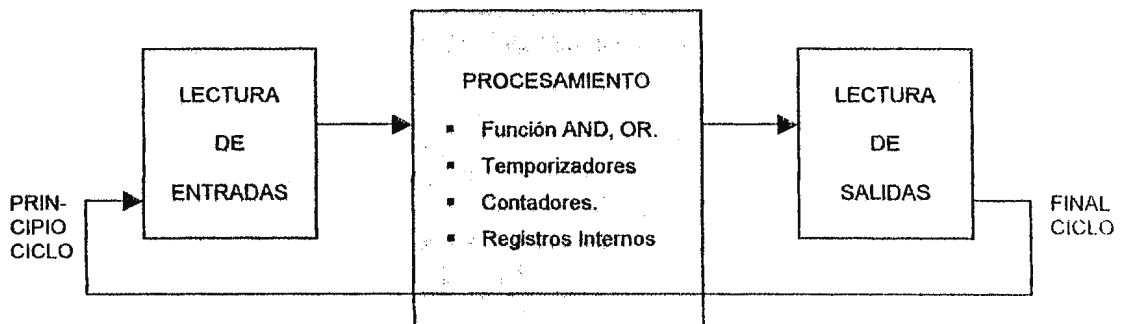


FIGURA 2.17. LOGICA DE FUNCIONAMIENTO DEL ASSI

En el interior del ASSI reside el programa, escrito por el usuario, que es ejecutado por el microprocesador. Este programa está constituido por una secuencia de instrucciones que se ejecutan de manera cíclica, desde la primera hasta la última, recomenzando luego desde el principio.

Estas instrucciones, en general, realizarán una lectura de las entradas. Posteriormente se efectuarán los procesamientos matemáticos, lógicos, de contaje, etc., para calcular el valor de las salidas.

Al final el valor calculado de las salidas será utilizado como comando de los dispositivos de salida efectivos del ASSI.

2.3.4 CARACTERISTICAS DIDACTICAS

Recordando que el ASSI es un equipo diseñado con fines didácticos, es importante enumerar las características didácticas que posee:

- Simula las instrucciones básicas de un PLC, para permitir la programación del mismo.
- Permite el estudio de la arquitectura interna de un PLC, mostrando las etapas que lo constituyen por medio de diagramas esquemáticos.
- Muestra como son procesadas las señales dentro del PLC por medio de puntos de prueba, en los cuales se observan las señales internas del entrenador.
- Posee una guía de usuario, la cual contiene la información necesaria para utilizarlo.
- Posee módulos de simulación de procesos industriales. Para facilitar el entendimiento de la programación del PLC, de forma que el usuario pueda crear programas destinados a controlar los procesos industriales simulados en estos módulos.

2.4 CONSTITUCION DE LA CPU

2.4.1 ESTRUCTURA DE LA CPU.

Como ya se ha mencionado la CPU o Procesador se vale de cuatro funciones principales, (ver sección 2.3.2.1), que son:

- Las Memorias Internas.
- Los Temporizadores.
- Los Contadores.
- Los Registros Internos.

Los Temporizadores y Contadores son funciones ofrecidas por el programa monitor de la CPU, el cual se ejecuta mediante la interacción de las Memorias con el Microprocesador, siendo este último el verdadero cerebro de la CPU. Los Registros Internos son áreas de memoria, puestas a disposición del usuario a través del programa monitor.

Entonces para comprender completamente el funcionamiento de la CPU es necesario conocer el hardware que lo constituye.

2.4.2 ARQUITECTURA BASICA DE LA CPU.

En esta sección se desarrolla la organización que regirá el diseño de la CPU del ASSI.

Las funciones inteligentes son desempeñadas; según el programa monitor; por la "Unidad de Control y Procesamiento" (CPU), que contiene básicamente cuatro unidades elementales, cuya interconexión se presenta en la Figura 2.18.

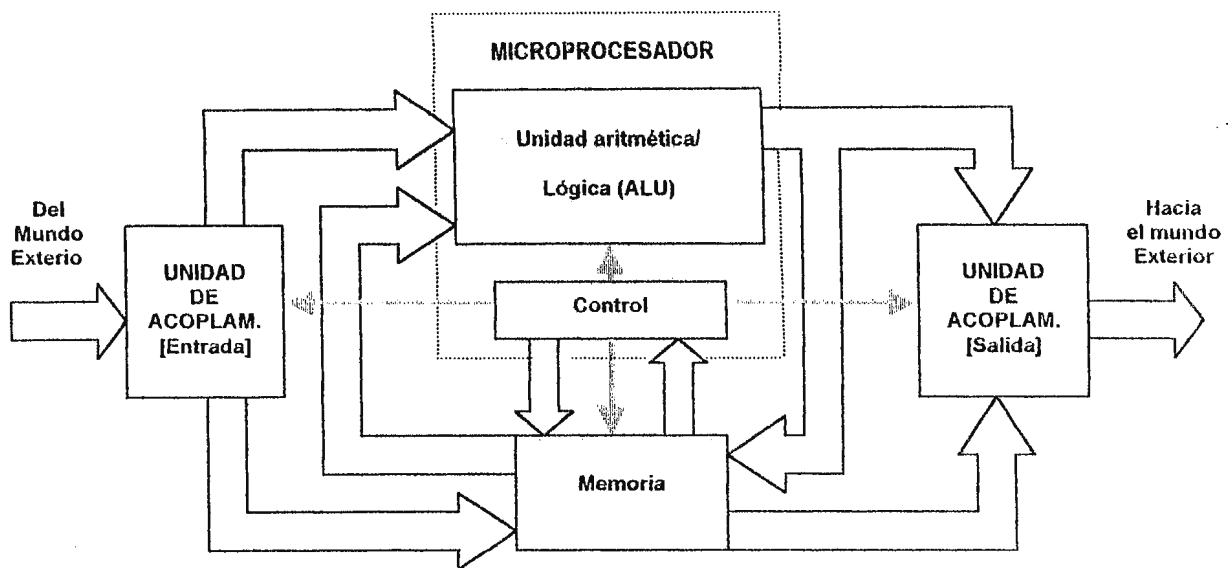


FIGURA 2.18. ARQUITECTURA BASICA DE LA CPU.

Las flechas en este diagrama indican la dirección en la cual los datos, información o señales de control fluyen. Se usan dos tipos de flechas; las gruesas representan líneas paralelas de información y las delgadas representan señales de control que utilizan una o unas cuantas líneas.

La función de cada unidad se desarrolla a continuación.

♦ UNIDAD DE CONTROL.

Está formada por circuitos lógicos y de distribución que generan las "señales de distribución y control", necesarias para dirigir la operación de las otras unidades, ejecutando así cada instrucción en un programa.

Para leer las instrucciones, esta unidad envía una dirección y un comando de lectura a la memoria correspondiente, luego la palabra de instrucción (grupo de dígitos en código binario) que esta almacenada en la localidad de la memoria, se transfiere a la unidad de

control. Entonces la palabra es decodificada por los circuitos lógicos de la unidad de control para determinar que instrucción se solicita. Con esta información la unidad genera las señales necesarias para ejecutar la instrucción.

◆ *UNIDAD ARITMÉTICA Y LÓGICA (ALU).*

Las operaciones lógicas y aritméticas son realizadas por la ALU, pero es la unidad de Control la que determina que tipo de operación se realizará, utilizando para esto 'Señales de Control'. La ALU utiliza los datos que provienen de la unidad de memoria o de la unidad de entrada; y luego de realizar las operaciones almacena los datos en memoria o los transfiere a la unidad de salida.

◆ *UNIDAD DE MEMORIA.*

Además de almacenar los resultados de la ALU, almacena palabras de instrucción y datos, del programa que será ejecutado.

Al igual que en la ALU, es la Unidad de Control, la que con operaciones de lectura o escritura controla su operación; accedendo mediante el código de operación adecuado a una localidad determinada. (Ver sección 2.4.4).

◆ *MICROPROCESADOR.*

El microprocesador es un circuito integrado que contiene la ALU y la Unidad de Control. Este puede ser considerado el *cerebro* de la computadora. (Ver sección 2.4.3).

◆ *UNIDAD DE ACOPLAMIENTO.*

Se encarga de sincronizar la transmisión de información digital entre la CPU y los dispositivos externos de entrada/salida.

Es esta unidad la que recibe y transmite los datos e instrucciones entre la CPU y el mundo exterior, proceso que es controlado por la Unidad de Control. (Ver secciones 2.4.2.1 y 2.4.5).

2.4.2.1 UNIDAD DE ACOPLAMIENTO Y SINCRONIZACION

Generalmente los dispositivos de entrada/salida no poseen las mismas características operativas que la CPU. Se entiende como características operativas, el nivel de las señales lógicas, la velocidad, el modo de transmisión, formato de datos, etc. Esto se debe a que los periféricos son sistemas digitales que poseen una unidad de control con un generador de impulsos diferente al que se encuentra en la CPU. (Ver sección 2.4.5).

Esta incompatibilidad es eliminada utilizando circuitos especiales destinados a acoplar la CPU y sus periféricos. Al conjunto de circuitos que realizan esta función se les llama Unidad de Acoplamiento, la cual se encarga de sincronizar la transmisión información digital entre la CPU y dispositivos externos de entrada/salida; como el teclado y los visualizadores.

Descripción del Modo de Funcionamiento.

Cuando el microprocesador se comunica con el periférico, la información y las señales de control de la CPU son recibidas por la unidad de acoplamiento, quien la transfiere al periférico, como se muestra en la figura 2.19. Luego el periférico le comunica a la unidad de acoplamiento que ya aceptó la información, entonces ésta pasa la información a la CPU para que se envíe una nueva información si es necesario.

Cuando es el periférico el que envía la información a la CPU, el proceso se realiza de forma similar.

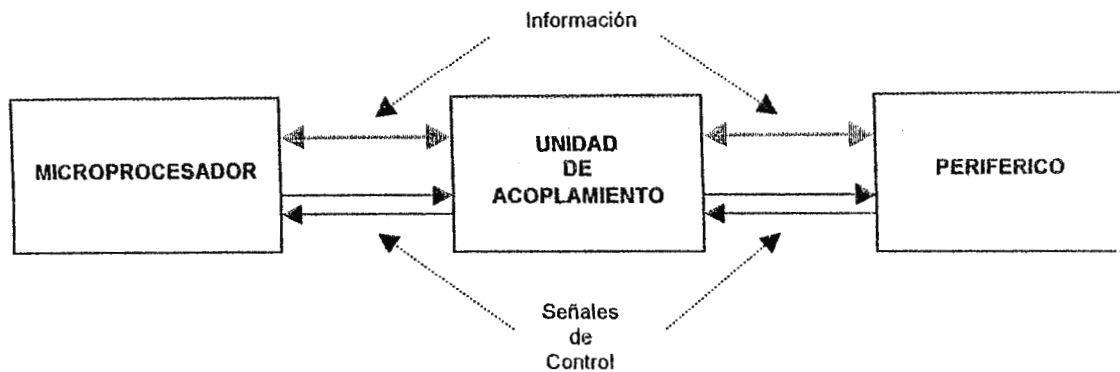


FIGURA 2.19. UNIDAD DE ACOPLAMIENTO.

2.4.3 EL MICROPROCESADOR

Esta etapa, está dedicada al estudio de las características del microprocesador que se utilizará en el desarrollo del entrenador ASSI.

Como ya se ha mencionado repetidas veces, el enfoque de este trabajo es altamente didáctico, y de forma más específica, se ha enfocado de acuerdo a los programas de estudios de la UDB, la cual ha basado sus prácticas de laboratorio de microprocesadores, en la programación del microprocesador 6502 fabricado por la Compañía Rockwell, siendo esta una razón por la que se escogió éste como base del CPU del entrenador ASSI.

Además, el 6502 posee las características de ser sencillo y versátil. Y como las funciones que desempeñará el ASSI son las más básicas de un PLC; temporización, conteo y secuencia; se ha considerado que este microprocesador cumple con los requerimientos de capacidad y velocidad requeridos para ejecutar estas funciones.

2.4.3.1 CPU BASADO EN EL 6502

El 6502 es un microprocesador de 8 bits, con reloj interno. En la figura 2.20 se muestra la configuración del sistema básico de CPU basado en el 6502, donde éste constituye el núcleo del sistema y es el centro de todas las transferencias de datos. Las interfaces con el usuario son un teclado, como entrada, y un visualizador, como salida. Para comunicarse con ellos, el 6502 utiliza como unidad de acoplamiento el circuito integrado de interfaz PIA (Interfaz Adaptadora de Periféricos 6520), la cual contiene los buffers y registros de control adecuados para almacenar y transferir entradas y salidas; además puede manipular interrupciones y enviarlas al 6502 a través del bus de control. En este sistema todos los circuitos integrados utilizan una fuente de alimentación común de 5V.

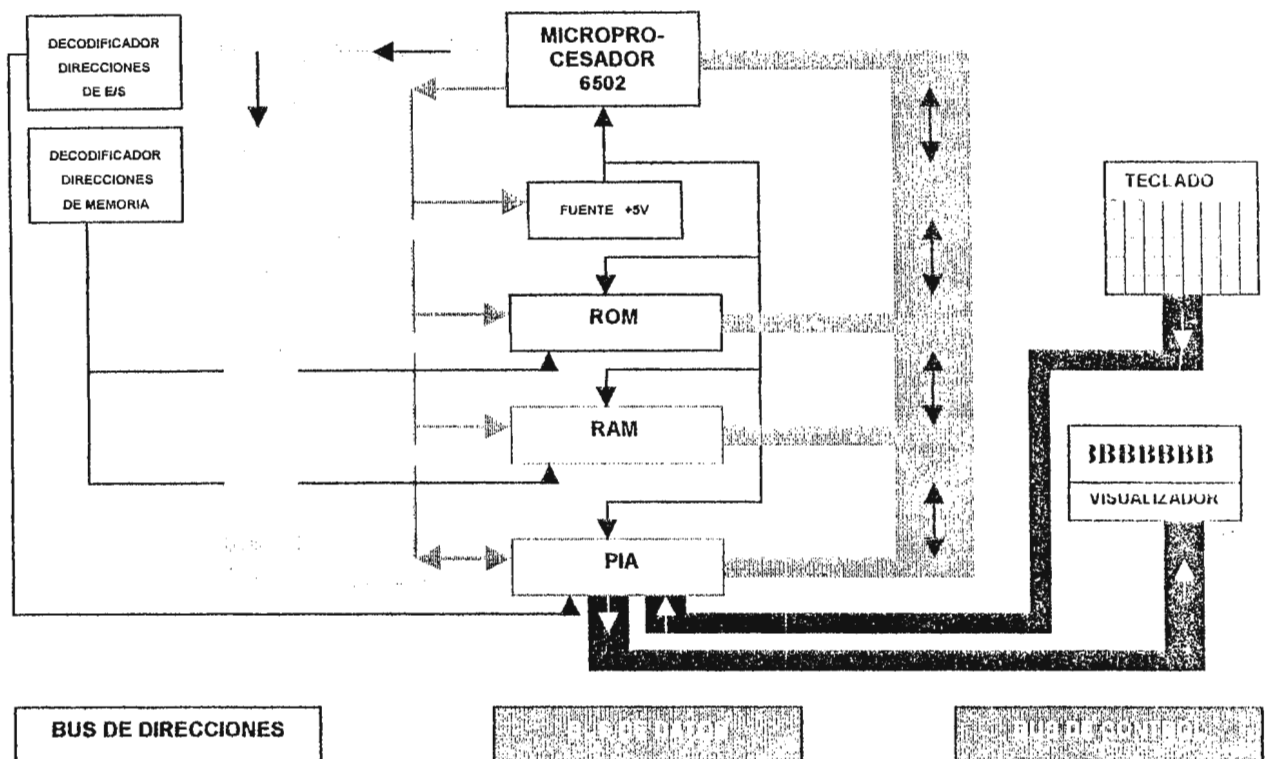


FIGURA 2.20. CONFIGURACIÓN DEL SISTEMA BÁSICO DE CPU BASADO EN EL 6502

Buses del 6502

1. Posee un bus de datos de 8 bits para comunicarse con la unidad de acoplamiento, la RAM o la ROM.
2. Tiene 16 líneas de dirección que forman el bus de direcciones. En la figura 2.20 se muestran algunas de estas líneas separadas del resto del bus, para indicar las señales de selección para la RAM, ROM y entradas/salidas de la PIA.
3. La acción entre el microprocesador, las memorias y la PIA es coordinada por las líneas de control.

2.4.3.2 DIAGRAMA DE PINES DEL 6502

El microprocesador 6502 se encuentra encapsulado en doble línea (DIP), éste se muestra en la figura 2.21, y en la tabla 2.4 se describen las funciones de cada pin.

El 6502 opera con un reloj de 1 ó 2 MHz.

Las entradas y salidas que posee este microprocesador se agrupan de la siguiente forma:

- Entradas/salidas de reloj.
- Conexiones de alimentación.
- Entradas/salidas del bus de datos.
- Salida del bus de direcciones.
- Entradas/salidas de control.

La dirección de flujo se muestran en la figura 2.21 por medio de flechas.

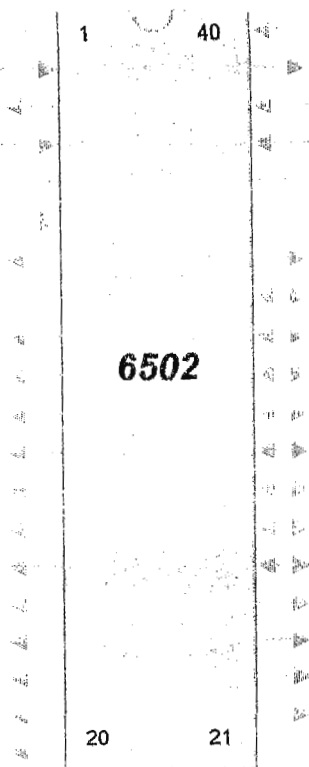


FIGURA 2.21. DIAGRAMA DE PINES DEL 6502

PIN No.	NOMBRE	DESCRIPCION	TIPO
9-20,22-25	AB0-AB15	Bus de direcciones.	Salida.
33-26	DB0-DB7	Bus de datos.	Triestado/bidirec.
2	RDY	Control listo.	Entrada.
3,39	$\phi 1, \phi 2$	Reloj(al sistema)	Salida.
4	$\overline{\text{TRQ}}$	Petición de Interr.	Entrada.
7	SYNC	Ciclo de búsqueda de código de op.	Salida.
34	$\overline{\text{R/W}}$	Lectura/ecritura.	Salida.
37	$\phi 0$	Reloj de CPU.	Entrada.
38	SO	Señaliz. de sobrefl.	Entrada.
40	RES	Reinicialización.	Entrada.
6	$\overline{\text{NMI}}$	Interrup. No enm.	Entrada.
1,21	V_{cc}, V_{ss}	Alimentación.	Entrada.

TABLA 2.4. NOMBRES Y FUNCIONES DE LOS PINES.

A continuación se describe la función de cada uno de los pines correspondientes a la figura 2.21 y tabla 2.4 del 6502.

- **TIERRA (V_{ss})(pines 1 y 21).** Estos pines están conectados a la parte negativa de la fuente de alimentación de 5V.
- **LISTO (RDY)(pin 2).** Señal de entrada que permite al usuario utilizar ciclo a ciclo el microprocesador en todos los ciclos, excepto en los de escritura. Una transición negativa al estado bajo durante la fase 1 detendrá al microprocesador y las líneas de dirección reflejarán la dirección actual que se está buscando. Esta condición permanecerá en la fase siguiente, en la cual RDY está en nivel bajo. Esta característica permite al

microprocesador conectarse por interfaces con PROM lentas, así como proporcionar rápidos accesos directos a memoria (DMA). Si la señal RDY está en el nivel bajo durante un ciclo de escritura, se ignora hasta la siguiente operación de lectura.

- **RELOJ ($\phi 0$ y $\phi 2$)(pines 37 y 30).** El 6502 requiere de un reloj de dos fases. Este microprocesador posee un generador interno de reloj, pero la frecuencia se controla externamente. Debe conectarse un cristal a las patillas 37 y 39 para fijar la frecuencia del oscilador.
- **RELOJ ($\phi 1$)(pin 3).** Este pin saca la fase 1 de la señal de reloj para sincronizar eventos de un sistema basado en el 6502.
- **PETICIÓN DE INTERRUPCIÓN (IRQ)(pin 4).** En el 6502 se solicita una secuencia de interrupción con un nivel BAJO en esta entrada. El microprocesador completará la instrucción actual que se está ejecutando antes de reconocer la petición. Entonces el registro de estado y la dirección actual son almacenados en memoria (específicamente en la PILA). La entrada RDY deberá ser alta para que se reconozca cualquier interrupción.
- **INTERRUPCIÓN NO ENMASCARABLE (NMI)(pin 6).** Con un flanco negativo en NMI se solicita al microprocesador una secuencia de interrupción no enmascarable. La NMI es una interrupción incondicional. El registro de estado y la dirección actual son almacenados en la pila.
- **SYNC (pin 7)** Esta línea permite la identificación de los ciclos en que el 6502 realiza una búsqueda de código de operación. La línea SYNC se activa durante el ciclo 1 de búsqueda de un código de operación y permanece alta durante el resto de ese ciclo. Si en RDY se da un flanco negativo durante el ciclo 1 cuando SYNC está en alto, el procesador se detendrá en su estado actual, y permanecerá así hasta que RDY alcance

un nivel alto. Debido a esto la señal SYNC puede ser utilizada para controlar que RDY sea la causa de la ejecución de una instrucción.

- **ALIMENTACIÓN (V_{cc})(pin 8).** La parte positiva de la fuente de alimentación de 5V se conecta a este pin.
- **LÍNEAS DE DIRECCIÓN (AB_0 - AB_{16})(pines 9-20,22-25).** Son los pines que se encuentran conectados al bus de direcciones del sistema, y se utilizan para direccionar dispositivos como ROM, RAM y dispositivos E/S.
- **LÍNEAS DE DATOS (DB_0 - DB_7)(pin 26-33).** El bus de datos bidireccional del sistema esta conectado a estos pines. Estas líneas se utilizan para transferir datos entre memoria y los dispositivos de entrada salida.
- **CONTROL DE LECTURA/ESCRITURA (R/W)(pin 34).** Cuando el 6502 esta leyendo o escribiendo, esta salida es una línea de control que señala a un dispositivo externo (RAM, ROM, Periférico, etc.).Una salida alta indica lectura, y una baja escritura.
- **SEÑALIZADOR DE SOBREFLUJO (SO)(pin 38).** Indica que la última operación ocasiona un desborde o sobreflujo. Una transición negativa en esta entrada pone a 1 el bit de "sobreflujo"en el registro de estado del microprocesador.
- **REINICIALIZACIÓN (RES)(pin 10).** Cuando ocurren fallas en la alimentación se utiliza esta entrada para reiniciar el microprocesador. Así cuando esta línea este en baja se inhibe la lectura o escritura del 6502.

2.4.3.3 ARQUITECTURA DEL 6502

El microprocesador 6502 está compuesto internamente por los siguientes elementos:

- Un bus de datos de 8 bits, (DB_0 - DB_7).

- Un bus de direcciones de 16 líneas (AB_0 - AB_{15}) que se conectan al bus de direcciones del sistema.
- Un contador de programa de 16 bits (PCH y PCL).
- Dos registros de índice de 8 bits (X e Y).
- Un acumulador de 8 bits.
- Un registro de estado del procesador de 8 bits (registro PS), el cual contiene siete señales.
- Ocho señales de control.

Las señales de control de entrada son: reinicialización,

- reinicialización
- petición de interrupción,
- interrupción no enmascarable,
- listo y
- sobreflujo.

Las señales de control de salida son:

- SYNC,
 - reloj y
 - lectura/escritura.
- La Unidad Aritmética Lógica (ALU).
 - Un registro puntero de Pila.

Registros.

El 6502 es un microprocesador orientado a memoria, esto es porque tiene pocos registros internos, por lo que debe acceder a memoria constantemente, (ver figura 2.22).

Estos registros son:

1. *Acumulador (A)*. Posee 8 bits y es el centro de la mayoría de las operaciones en el 6502. Está asociado a la ALU, pues los resultados de las operaciones aritméticas y lógicas se almacenan en el acumulador.
2. *Contador de programa (PC)*. Registro de 16 bits que contiene la dirección de la próxima instrucción que se va a realizar.
3. *Registros X e Y*. Registros de 8 bits, de propósito general, es decir que pueden almacenar cualquier tipo de datos. Se utilizan comúnmente para contar pasos de programa o para proporcionar un valor índice en el cálculo de direcciones efectivas.
4. *Registro de Puntero de Pila (S)*. Registro de 8 bits que define la posición actual del último dato introducido a la Pila en la RAM. El byte de orden superior de este registro esta siempre en 01H por especificaciones del fabricante.
5. *Registro de estado del procesador (PS)*. Registro de 8 bits que contiene los siete señaladores que utiliza el microprocesador 6502; éstos se muestran en la figura 2.22.

El registro de estado en el 6502 se utiliza para determinar si se cumple una determinada condición de bifurcación y para indicar el modo de operación del procesador.

- C. SEÑALIZADOR DE ARRASTRE. Se pone a 1 si se genera arrastre en la última instrucción de suma. También se utiliza en la resta como indicador de préstamo, poniéndose a 1 cuando no ha habido préstamo.
- Z. SEÑALIZADOR DE CERO. Se pone a 1 cuando el resultado de una operación es cero.
- I. INDICADOR DE INHABILITACION DE IRQ. Sirve para habilitar o deshabilitar la entrada de interrupción IRQ. Cuando la entrada de petición de interrupción está inhabilitada se pone a 1.
- D. INDICADOR DE MODO DE OPERACIÓN. Indica si el 6502 está en modo de operación binario o de aritmética decimal. Un 1 indica que se interpretarán los operandos como BCD cuando se sume o se reste; de lo contrario, (0), es modo binario.
- B. SEÑALIZADOR DE RUPTURA. Indica con un 1 si la última interrupción fue provocada por BRK, es decir interrupción por software". Si se encuentra en 0, significa que la última interrupción fue por hardware (NMI, IRQ, o RES).
- N. SEÑALIZADOR NEGATIVO. Se utiliza para indicar un resultado negativo. Si el resultado de la última operación fue negativo se pondrá a 1.

Unidad Aritmética Lógica (ALU).

La ALU es la sección del microprocesador que se encarga de realizar las operaciones aritméticas y lógicas. Estas incluyen: Suma.

Resta.	OR.
Incremento.	OR- exclusiva.
Decremento.	

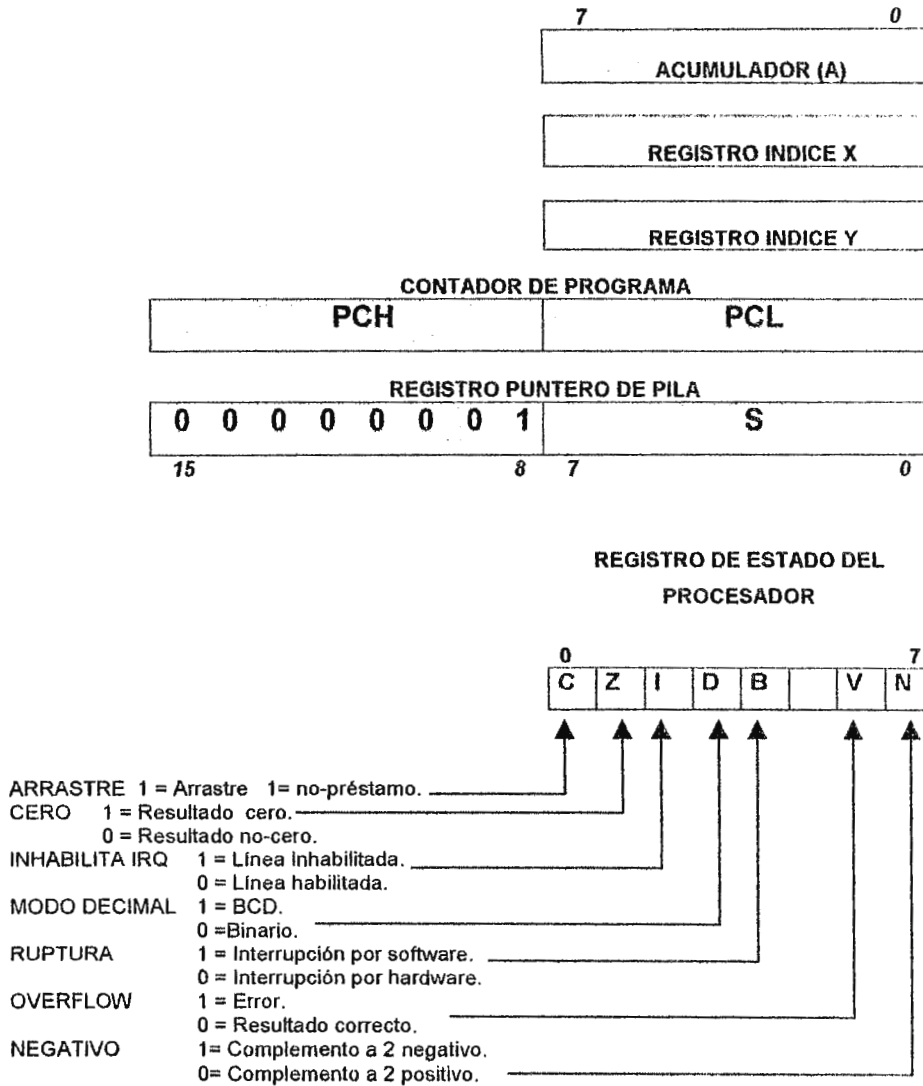


FIGURA 2.22. REGISTROS INTERNOS DEL 6502.

Registro de Instrucciones.

El microprocesador 6502 realiza ciclos con la siguiente secuencia: busca-decodifica-ejecuta. Durante la etapa de búsqueda, un código de operación es almacenado en el registro de Instrucciones, luego éste es decodificado por la sección decodificadora de instrucción, la cual genera las señales de control internas y externas adecuadas.

Reloj del Sistema.

El 6502 posee un reloj interno que genera una señal de reloj de dos fases. Este reloj alimenta el bloque de control de temporización que sincroniza las operaciones del microprocesador.

En los pines 3 y 39 del circuito integrado se encuentran disponibles las señales de la fase 1 y 2 del reloj, respectivamente. Las instrucciones en el 6502 se ejecutan entre 2 y 7 ciclos de máquina.

Las formas de onda del reloj del sistema y un ciclo de máquina se muestran en la figura 2.23. Se observa que cuando $\phi 1$ está en ALTO, $\phi 2$ está siempre en baja.

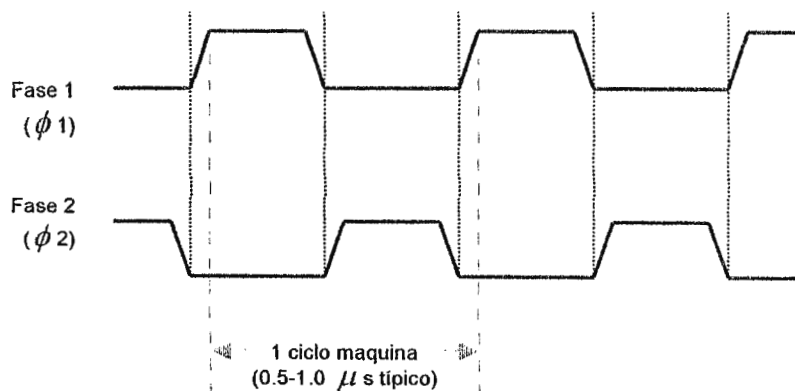


FIGURA 2.23 FORMAS DE ONDA DE DOS FASES.

Interrupciones.

El 6502 posee cuatro tipos de interrupción¹⁵; tres provocadas por hardware, es decir por un dispositivo externo que envía una señal baja a un pin determinado; y una por software, es decir que la operación del microprocesador es interrumpida por una instrucción del programa. Las interrupciones se presentan a continuación según su prioridad¹⁶:

- **RES:** Reinicialización. (Realizada por hardware). Se utiliza para reinicializar al microprocesador cuando éste se conecta a la alimentación, o para reinicializarlo utilizando un conmutador una vez que ha sido conectado a la alimentación.
- **NMI:** Interrupción no enmascarable. (Realizada por hardware). Responde a un flanco negativo, el cual la activa. Luego de realizar la rutina de servicio de interrupción ejecuta la instrucción RTI (vuelta de interrupción), con lo cual devuelve el control al programa principal.
- **Instrucción BRK.** (Realizada por Software). Esta instrucción activa el registro del el señalizador de ruptura "B" del registro de estado, lo que indica que no es una interrupción provocada por hardware. Se encarga de cargar en el contador de programa la dirección de comienzo de la rutina de tratamiento de la interrupción de ruptura.
- **IRQ:** Petición de interrupción. (Realizada por hardware). Esta interrupción utiliza el señalizador "I" para que otra petición de interrupción con la misma prioridad no sea atendida. Al finalizar la interrupción, utiliza la instrucción RTI, (ver apéndice B), para devolver el control al programa principal.

¹⁵ Interrupción: Interrumpir la ejecución del programa regular y ocasionar que el microprocesador salte a un programa especial, denominado "rutina de servicio de interrupciones".

¹⁶ Prioridad: En caso de que dos o más interrupciones se activen al mismo tiempo.

2.4.3.4 PROGRAMACION DEL 6502.

En la memoria del microprocesador se almacena el programa que se encarga de controlar todas las funciones que éste desarrolla. El programa no es más que un conjunto de instrucciones ordenadas de manera lógica y secuencial. Las instrucciones son los ordenes específicos que se le dan al microprocesador para que ejecute la operación deseada.

Una instrucción se divide en dos partes fundamentales el Opcode y el Operando; ver figura 2.24.

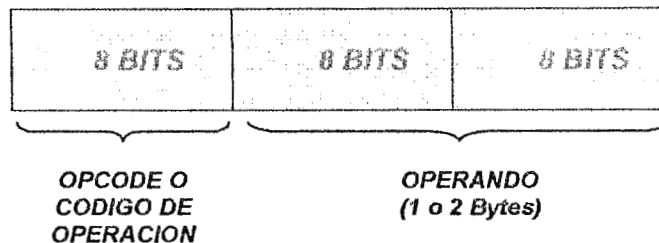


FIGURA 2.24. FORMATO DE UNA INSTRUCCIÓN.

- *El Opcode* es el código de la instrucción seleccionada. Éste se puede representarse de dos formas:
 1. *Representación Mnemotécnica*: por lo general es la abreviatura del nombre de la función en inglés, representada por tres letras mayúsculas (este es el caso para el microprocesador 6502), ejemplo:

Instrucción: to Index Transfer Accumulator X

Mnemotécnico: **TAX**. (Ver Apéndice B).
 2. *Representación Numérica*: consiste en un número en código hexadecimal el cual posee una longitud de un byte si el microprocesador posee un bus de datos de 8 bits (este es el caso para el microprocesador 6502), ejemplo:

Instrucción: Transfer Accumulator to Index X

Código hexadecimal: **AA**. (Ver Apéndice B).

- *El Operando* contiene datos o indica al microprocesador la dirección donde están ubicados los datos a ser procesados. Su longitud puede ser de 1 a 2 bytes.

Tanto el Opcode como el Operando son introducidos a la memoria en su representación en hexadecimal. Una instrucción completa puede tener el siguiente formato:

En Mnemotécnico: **TAX 59**

En hexadecimal: **AA 59**

Donde 59 es el dato a operar.

Modos de direccionamiento

La mayoría de las instrucciones en el 6502 poseen más de una representación numérica de código de operación, debido a que existen diferentes métodos de direccionamiento de la memoria o de los registros. Las diferentes representaciones numéricas de código de operación se presentan en el apéndice B.

Como ejemplo, la instrucción LDA (Load Accumulator) para cargar un dato en el acumulador posee las representaciones numéricas A9, AD y A5, correspondientes a diferentes modos de direccionamiento.

Los modos de direccionamiento del 6502 son:

- *Implicito*. Consiste en un solo byte de opcode. Esta instrucción no necesita especificar un operando para la ejecución. Por lo general se trata de comandos para alterar el contenido de los registros, incrementar o reducir los registros o transferir información entre ellos.

- *Inmediato*. Estas instrucciones requieren de un operando para su ejecución. Consiste en 2 bytes, uno para el opcode y otro para el operando, el cual contiene los datos a procesar durante la operación.
- *Absoluto*. Consiste en tres bytes, uno de opcode y dos de operando, el cual contiene la dirección donde se encuentra el dato a procesar. El primer byte del operando especifica la parte baja de la dirección y el segundo byte especifica la parte alta de la misma.
- *Indexado*. Existen dos registros índice en el 6502; "X" e "Y". La dirección donde se encuentra el dato a procesar se obtiene sumando el contador del programa con el dato contenido en el registro índice que se esté empleando.
- *Relativo*. Este modo de direccionamiento se emplea en instrucciones de bifurcación e iteración. La dirección donde se encuentra el dato a procesar se obtiene sumando el operando, cuya longitud es de un byte, al byte menos significativo del contador de programa.
- *De página cero o Directo*. El dato a operar se encuentra localizado en la página cero (direcciones de la 0000 a 00FF del microprocesador). Utiliza un operando de un byte de largo el cual indica la posición de memoria específica dentro de esta página en donde se encuentra el dato.
- *Indirecto*. Este modo de direccionamiento utiliza un operando de dos bytes de longitud. Este operando consiste en la dirección de memoria donde se encuentra el byte menos significativo de la dirección donde se encuentra el dato a procesar; el byte más significativo se encuentra en la dirección siguiente a la especificada por el operando.

2.4.4 ROM PROGRAMABLE Y BORRABLE (EPROM)

Estas memorias pueden ser borradas y reprogramadas por el usuario. Para programarlas se requiere una circuitería de soporte conocida como "Grabador de Memorias EPROM". El proceso de grabación consiste en aplicar una tensión DC mucho mayor que la de alimentación ($5 V_{DC}$), que puede estar en el orden de los $12 V_{DC}$ hasta $50 V_{DC}$; este valor depende de las especificaciones técnicas de la memoria. Este voltaje deberá ser aplicado durante un intervalo de tiempo de aproximadamente 50 ms, si el tiempo es menor se corre el riesgo de que el dato no se grabe debidamente.

El proceso de borrado consiste en exponer las celdas a rayos ultravioleta durante un tiempo promedio de 15 a 20 minutos. Las memorias EPROM poseen una ventana que permite el paso de la luz ultravioleta hacia la superficie de las celdas, (ver figura 2.25).

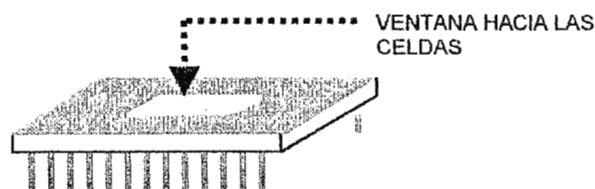


FIGURA 2.25. PASTILLA DE MEMORIA EPROM.

La EPROM presenta la desventaja de que para borrarla hay que extraerla del sistema en que se encuentra para ponerla en el borrador de memorias. Otra desventaja es que no se puede borrar el contenido de una celda en particular. La EPROM que se utilizará en el Entrenador ASSI es la 2716 con una capacidad de $2K \times 8$.

En la figura 2.26 se presenta el símbolo de la EPROM 2716 y en la Tabla 2.5 la función de sus pines.

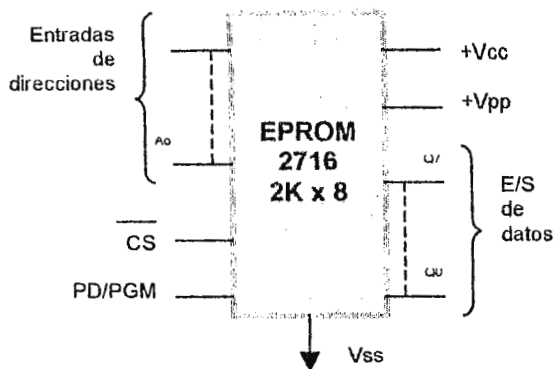


FIGURA 2.27 SIMBOLO DE LA EPROM 2716.

PIN	FUNCION
A0 – A10	Entradas de Dirección.
Q0 – Q7	Entradas/ Salidas de Datos.
PD/PGM	Programa/Potencia Apagada. (Habilitación de Pastilla)
CS	Selección de Pastilla.
Vpp	Entrada de Programación de +25V.
Vcc	Potencia (+ 5V).
Vss	Tierra.

TABLA 2.5. FUNCION DE LOS PINES DE LA EPROM

2.4.4.1 FUNCIONES DE LOS PINES DE CONTROL

La memoria 2716 posee tres pines de control, los cuales se encargan de seleccionar los distintos modos u operaciones del dispositivo, dichas funciones son Lectura y Escritura. Esto pines se describen a continuación:

- **CS (PIN 20).** Este pin es el encargado de selección de pastilla, si en la entrada CS existe un uno lógico, la pastilla esta desactivada y es en este momento en el cual se pueden introducir datos a la memoria. Si existe un cero lógico en CS, la pastilla esta activada, la salida de datos provee la información que se encuentra en la dirección seleccionada (A₀– A₁₀), el dato puede ser leído.
- **PD/PGM (PIN 18).** Para poder introducir datos a la memoria es necesario aplicar un pulso de 10 ms (en estado alto), en la entrada PGM.
- **V_{pp} (PIN 21).** Este pin es la entrada de voltaje de programación que debe alimentarse con 25 V al momento de enviar el pulso de programación.

2.4.4.2 PROGRAMACIÓN DE LA EPROM 2716

Antes de enviar el pulso de programación a la entrada PD/PGM, hay que hacer válida la dirección en la que se desea programar y enviar el dato a introducir. Una vez que las entradas de datos y direcciones son válidas, y los voltajes en CS y Vpp han sido fijados (uno lógico y 25 voltios respectivamente), se manda un pulso de aproximadamente 10 ms a la entrada PD/PGM para programar el dato en la EPROM. El dato es programado en la dirección seleccionada, cuando PD/PGM regresa a cero lógico.

El diagrama de tiempo correspondiente se muestra en la figura 2.27

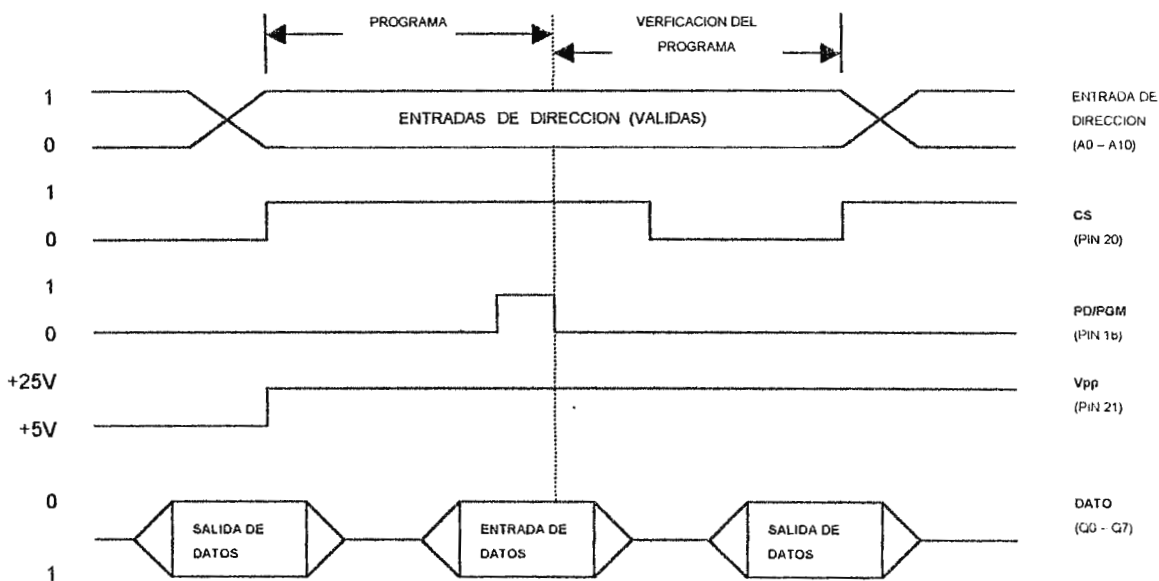


FIGURA 2.27. DIAGRAMA DE TIEMPO DE PROGRAMACION DE LA EPROM 2716.

2.4.4.3 LECTURA DE LA EPROM 2716

Hay que direccionar la memoria en la localidad que se quiere leer. CS debe estar a uno lógico mientras tanto el pin PD/PGM debe estar a cero lógico y Vpp debe estar a cinco voltios cuando se leen los datos de la EPROM. CS debe caer de un uno a un cero lógico. El diagrama de tiempo correspondiente se muestra en la figura 2.28

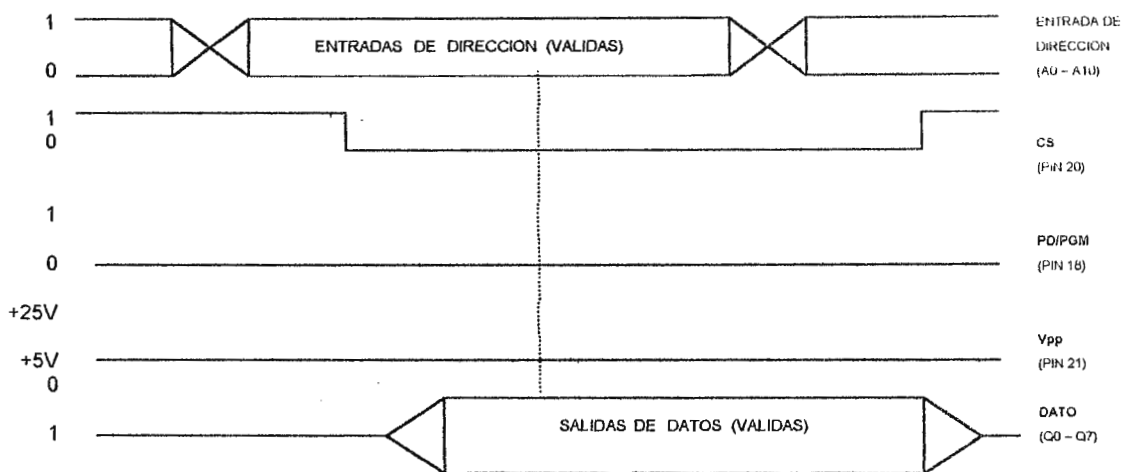


FIGURA 2.28. DIAGRAMA DE TIEMPO DE LECTURA DE LA EPROM

2.4.5 INTERFACES

Un dispositivo de interfaz es el que permite al microprocesador comunicarse con los sistemas periféricos tales como: teclados, impresoras, monitor, dispositivos de estado sólido, relés, transductores, etc.

Estos dispositivos son diseñados para ser totalmente compatibles con una determinada familia de microprocesadores. Las características propias de cada dispositivo de interfaz, varían según los requerimientos del microprocesador para el cual han sido

fabricados. Las funciones que desempeñan son controladas a través de un programa que ejecuta el microprocesador.

Debido a que la CPU del entrenador ASSI está basada en el microprocesador 6502, se ocuparán dos INTERFACES ADAPTADORAS DE PERIFERICOS (Peripheral Interface Adapter, PIA) 6520.

2.4.5.1 LA PIA 6520

Es una interfaz de propósito general, con entradas y salidas (E/S) programables diseñada para trabajar con el microprocesador 6502. El propósito de tener entradas y salidas programables, es simplificar las conexiones entre el microprocesador y los sistemas periféricos. La PIA 6520 se muestra en la figura 2.29.

La PIA 6520 posee 16 líneas de E/S las cuales se dividen en dos grupos de ocho, llamados Puerto A y Puerto B. Las líneas de estos puertos pueden ser programadas ya sea para introducir datos a la PIA o para sacar datos de ella. Esto va a depender de la operación que este ejecutando que puede ser de escritura o de lectura.

Pines de la PIA 6520

En la figura 2.29 se muestra el diagrama de pines de la PIA, y en la tabla 2.6 sus correspondientes funciones.

- LINEAS DE CONEXIÓN AL BUS DE DATOS DEL μP . Consiste en 8 líneas bidireccionales, que van desde D_0 hasta D_7 permitiendo la transferencia de datos entre la PIA y el microprocesador.

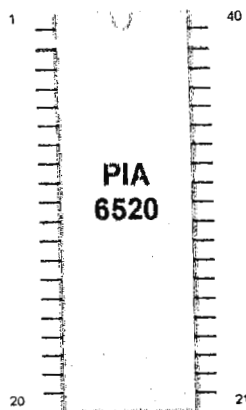


FIGURA 2.29. DIAGRAMA DE PINES DE LA PIA 6520.

PINES DE LA PIA 6520			
1	V _{SS}	21	R/W
2	PA ₀	22	CS ₀
3	PA ₁	23	CS ₂
4	PA ₂	24	CS ₁
5	PA ₃	25	E
6	PA ₄	26	D ₇
7	PA ₅	27	D ₆
8	PA ₆	28	D ₅
9	PA ₇	29	D ₄
10	PB ₀	30	D ₃
11	PB ₁	31	D ₂
12	PB ₂	32	D ₁
13	PB ₃	33	D ₀
14	PB ₄	34	Reset
15	PB ₅	35	RS ₁
16	PB ₆	36	RS ₀
17	PB ₇	37	IRQB
18	CB ₁	38	IRQA
19	CB ₂	39	CA ₂
20	V _{CC}	40	CA ₁

TABLA 2.6. FUNCION DE LOS PINES DE LA PIA 6520

- LINEAS DE CONEXIÓN AL BUS DE DIRECCIONES DEL μ P. La PIA posee cinco pines que se conectan al bus de direcciones del μ P:
 - Pines de selección de pastilla. CS₀, CS₁, y CS₂.
 - Pines de selección de registro. RS₀ y RS₁.
- LINEAS DE CONEXIÓN AL BUS DE CONTROL DEL μ P. La PIA posee cinco pines que se conectan a cuatro líneas del bus de control del μ P, para esto se conectan IRQA e IRQB a una sola línea. Los pines mencionados son:
 - Pin de Lectura/Escritura, R/W. Se conecta a la línea R/W del 6502.
 - Pin de Habilitación, E. Se conecta a la línea ϕ 2 del 6502.
 - Pin de Reinicio. Se conecta a la línea de reinicio del 6502.

- Pines de solicitud de interrupción, IRQA e IRQB. Se utilizan para enviar señales de solicitud de interrupción al 6502. Ambas se conectan a la línea IRQ del μP .
- PUERTO A. Cada línea del puerto A ($PA_0 - PA_7$) puede programarse como entrada o salida. Existe un registro interno de 8 bits en la PIA denominado DDRA (Data Direction Register Port A), el cual es direccionado desde el microprocesador; cada bit del DDRA corresponde a una línea del puerto. Si se carga un 1 lógico en el bit correspondiente a una determinada línea, ésta se comportará como salida, y si se carga un 0 lógico se comportará como entrada.
- PUERTO B. Tal como en el puerto a, en el puerto B ($PB_0 - PB_7$) cada línea puede ser programada como entrada o como salida, en este caso por un registro interno denominado DDRB.
- LINEAS DE INTERRUPCION Y CONTROL. Son cuatro:
 - Pines de Interrupción: Son los pines CA_1 y CB_1 , estos actúan solo como entradas para interrumpir a la PIA desde los dispositivos periféricos.
 - Pines de Interrupción/Control: Son los pines CA_2 y CB_2 . CA_2 puede usarse como línea de interrupción si se programa como entrada, y también puede utilizarse como señal de sincronismo entre la PIA y un equipo periférico si se programa como salida. CB_2 puede usarse como línea de interrupción si se programa como entrada, o como línea de control si se programa como salida.

Conexión entre el μP 6502 y la PIA 6520

El μP y la PIA se comunican por medio del bus de datos, bus de direcciones y bus de control. En la figura 2.30 se muestra un diagrama del alambrado entre el μP y la

PIA. Las conexiones incluyen un bus bidireccional de datos de 8 bits, tres líneas de selección de pastilla, dos líneas de selección de registro, una línea de lectura/escritura, una línea de habilitación, y una línea de reinicio.

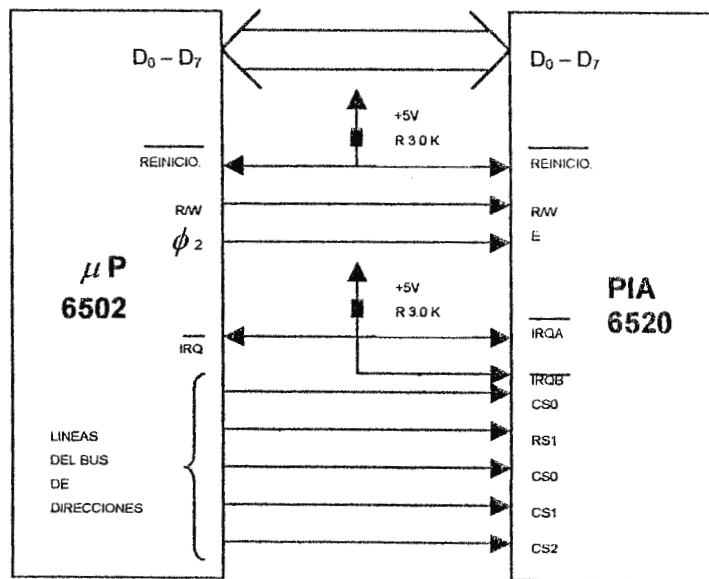


FIGURA 2.30. DIAGRAMA DE CONEXIONES ENTRE EL μP 6502 Y LA PIA 6520.

CAPITULO 3

DISEÑO DE LA CPU DEL ENTRENADOR ASSI

3.1 MAPA DE MEMORIA DEL ENTRENADOR ASSI

En la figura 3.1 se presenta la distribución del Mapa de Memoria del "Entrenador ASSI". Los criterios tomados en cuenta para su diseño son:

1. Las líneas del bus de direcciones que se utilizan para seleccionar los circuitos de interfaz dependen del diseño propio de cada sistema, es decir que no existen reglas ni métodos específicos. Pero al momento del diseño debe tenerse sumo cuidado en los siguientes puntos:
 - Que una misma dirección no seleccione dos dispositivos diferentes.
 - Que no se utilicen direcciones reflejo para seleccionar dos dispositivos diferentes.
2. Para el caso específico del microprocesador 6502 deben reservarse las paginas 00 y 01 para operaciones propias de las instrucciones de integrado.
3. Debe tomarse en cuenta que ciertos dispositivos necesitan unos cuantos bytes de memoria, como la PIA (Interface Adaptadora de Periféricos) que requiere solamente de 4 ubicaciones, en cambio la MROM (ROM del Programa Monitor) necesita varias paginas dependiendo del tamaño del programa.

3.1.1 DECODIFICACION DEL MAPA DE MEMORIA

Basándose en estos criterios se seleccionaron las líneas 15, 14, 12 y 02 del bus de direcciones para acceder a los diferentes dispositivos del sistema.

Estas líneas se agrupan en dos conjuntos que se conectan a dos decodificadores de 3 a 8 líneas (74LS138).

Las líneas 15, 14 y 12 se utilizarán para seleccionar las áreas de memoria grandes, y con las líneas restantes se seleccionan otros dispositivos.

-MROM

-UROM

-RAM

-DAC¹

-PIA 2.

TEMPORIZADOR.

Las líneas 14, 12 y 02 se utilizarán para seleccionar las siguientes áreas de memoria:

-PIA 1.

-LATCH.

¹ El DAC, la PIA2 y el Temporizador no requieren de un área de memoria grande, pero para aprovechar a máximo los recursos se conectaron en el mismo decodificador de la MROM, UROM, y RAM.

MAPA DE MEMORIA DE LA CPU
DEL ENTRENADOR ASSI

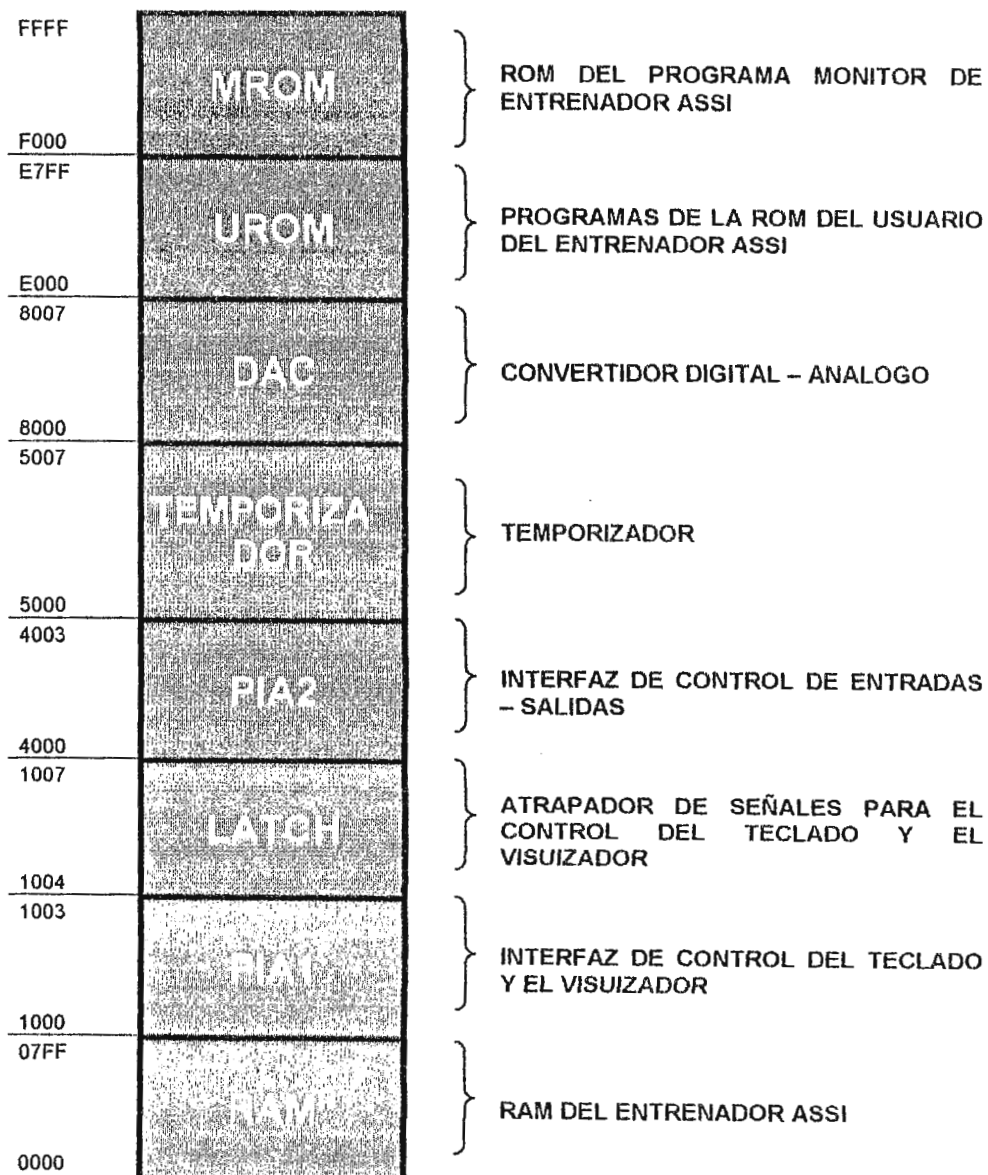


FIGURA 3.1. MAPA DE MEMORIA

En la tabla 3.1 se presentan las direcciones específicas dedicadas para cada área del mapa de memoria.

DIRECCIONES DEL MAPA DE MEMORIA																	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	HEXA.	
0	0	X	0	X	X	X	X	X	X	X	X	X	0	X	X	0000	RAM
0	0	X	0	X	X	X	X	X	X	X	X	X	1	X	X	07FF	
0	0	X	1	X	X	X	X	X	X	X	X	X	0	X	X	1000-3	PIA1
0	0	X	1	X	X	X	X	X	X	X	X	X	1	X	X	1004-7	LATCH
0	1	X	0	X	X	X	X	X	X	X	X	X	0	X	X	4000-4	PIA2
0	1	X	0	X	X	X	X	X	X	X	X	X	1	X	X		
0	1	X	1	X	X	X	X	X	X	X	X	X	0	X	X	5000-7	PIA3
0	1	X	1	X	X	X	X	X	X	X	X	X	1	X	X		
1	0	X	0	X	X	X	X	X	X	X	X	X	0	X	X	8000-7	PIA4
1	0	X	0	X	X	X	X	X	X	X	X	X	1	X	X		
1	1	X	0	X	X	X	X	X	X	X	X	X	0	X	X	E000	PIA5
1	1	X	0	X	X	X	X	X	X	X	X	X	1	X	X	E7FF	
1	1	X	1	X	X	X	X	X	X	X	X	X	0	X	X	F000	PIA6
1	1	X	1	X	X	X	X	X	X	X	X	X	1	X	X	FFFF	

TABLA 3.1 DIRECCIONES DEL MAPA DE MEMORIA

A continuación se explica la función específica de las líneas AB15, AB14, AB12 y AB02 en la selección de cada área de memoria.

□ DIRECCIONAMIENTO DE LA RAM:

La RAM se ubica desde la 0000 hasta la 07FF, para seleccionarla se utilizan los bits 15, 14 y 12 del bus de direcciones, de forma que siempre que estos se encuentren en "0" se direcciona la RAM.

□ DIRECCIONAMIENTO DE PIA 1:

La PIA 1 se utiliza para controlar el teclado y el visualizador; ésta se ubica desde la 1000 hasta la 1003, para seleccionarla se utilizan los bits 14, 12 y 02 del bus de direcciones. Con los bits AB14=0 y AB12=1 se ubica la página 10 y con AB02=0 se limitan las cuatro ubicaciones 1000, 1001, 1002, y 1003.

□ **DIRECCIONAMIENTO DEL LATCH:**

El Latch se selecciona de igual forma y con las mismas líneas que la PIA, con la diferencia de que con $AB02=1$ se seleccionan las siguientes ubicaciones: 1004, 1005, 1006 y 1007.

□ **DIRECCIONAMIENTO DE PIA 2:**

La PIA 2 se utiliza para controlar las entradas y salidas del PLC; ésta se ubica desde la 2000 hasta la 2003, para seleccionarla se utilizan los bits 15, 14, 12 del bus de direcciones, con los valores 010.

□ **DIRECCIONAMIENTO DEL TEMPORIZADOR:**

El Temporizador se selecciona de igual forma y con las mismas líneas que la PIA 2, con la diferencia de que con $AB12=1$ se selecciona la ubicación: 3000.

□ **DIRECCIONAMIENTO DEL DAC:**

El DAC es el convertidor que se utiliza para sacar señales análogas del PLC se selecciona de igual forma y con las mismas líneas que la RAM, con la diferencia de que con $AB15=1, AB14=0$ y $AB12=0$ se ubica en la dirección 4000.

□ **DIRECCIONAMIENTO DE LA UROM:**

La UROM es la ubicación de memoria reservada para cargar programas realizados por el usuario. Ésta se ubica desde la E000 hasta la E7FF, para seleccionarla se utilizan los bits 15, 14 y 12 del bus de direcciones, de forma similar que la RAM. Con $AB15=1, AB14=1$ y $AB12=0$ se selecciona las paginas E0 - EF.

□ **DIRECCIONAMIENTO DE LA MROM:**

La MROM es la ubicación de memoria que contiene el programa monitor. Ésta se ubica desde la F000 hasta la F7FF, para seleccionarla se utilizan los bits 15, 14 y 12

del bus de direcciones, de forma similar que en la RAM, y en la MROM. Con AB15=1, AB14=1 y AB12=1 se selecciona la pagina F.

3.1.2 DECODIFICACION DE DIRECCIONES

Debido a que se utilizan tres líneas; AB02, AB11 y AB14; para seleccionar los dispositivos de interfaz y tres líneas; AB11, AB12 y AB14; para seleccionar los módulos de memoria, se utilizarán dos decodificadores 74LS138 IC9 e IC13 para el direccionamiento con cada conjunto de líneas del bus de dirección.

La implementación de los circuitos decodificadores permitirá activar cada dispositivo solamente en la incidencia de un código de entrada único. En la figura 3.2 se presenta el símbolo lógico del decodificador 74LS138 y en la figura 3.3 el diagrama lógico equivalente interno.

FUNCIONAMIENTO DEL 74LS138.

El decodificador 74LS138 posee salidas de compuerta NAND, de modo que sus salidas son bajas activas.

Líneas del 74LS138.

- Entradas: Las líneas A, B y C son las entradas del integrado, a través de las cuales se aplica el código a decodificar.
- Salidas: La salida se obtiene en las líneas $Y_7 - Y_0$.

- **Habilitación:** Las entradas G1, G2A y G2B son entradas activas separadas combinadas en función AND.

En el apéndice D se presentan las hojas técnicas del 74LS138.

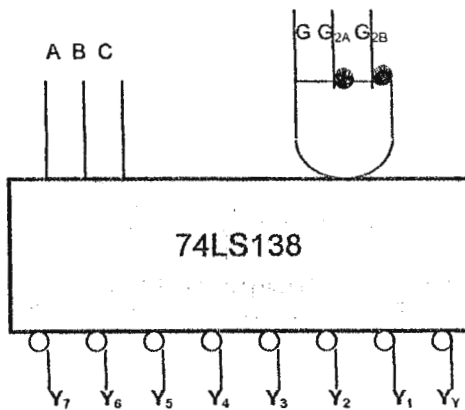


FIGURA 3.2. SIMBOLO LOGICO DEL 74LS138

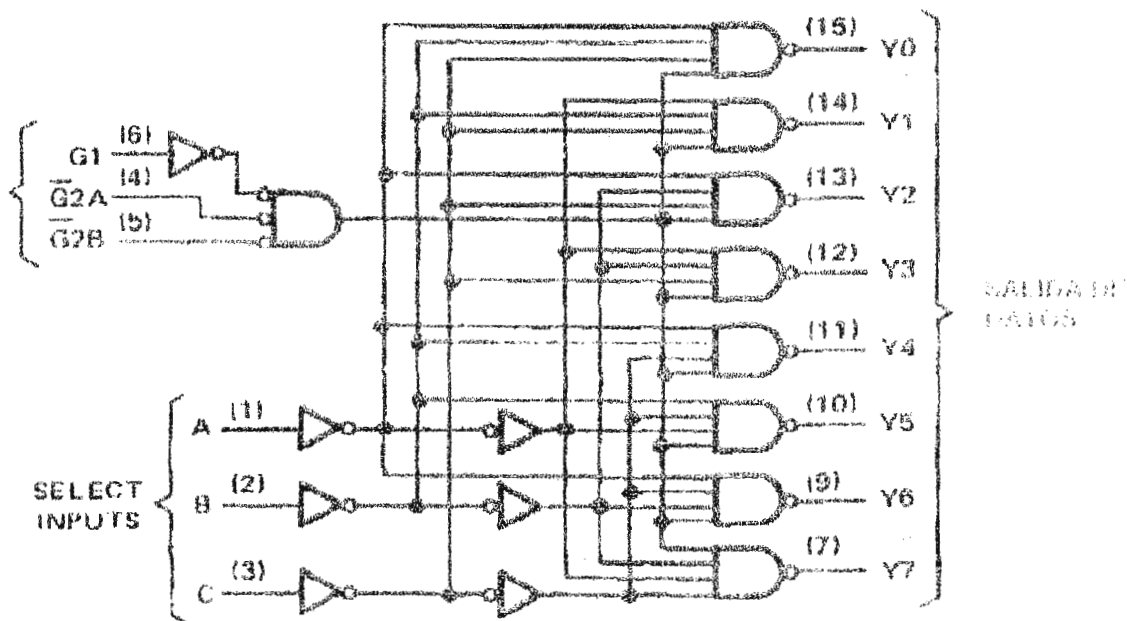


FIGURA 3.3. DIAGRAMA LOGICO DEL DECODIFICADOR 74LS138

El funcionamiento de los IC 74LS138 se resume en las tablas 3.2 y 3.3. En ésta se observa que se necesita un 0 lógico en G2A y G2B; y un 1 lógico en G1 para habilitar cada integrado.

G1	G2A	G2B	Salidas (Todas Altas)
1	0	0	Responder al código de entrada A ₂ , A ₁ , A ₀
X	X	1	Desactivada
X	1	X	Desactivada
0	X	X	Desactivada

TABLA 3.2. TABLA DE HABILITACION.

ENTRADAS					SALIDAS							
HABILITACION		SELECCION										
G1	G2	C	B	A	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
X	H	X	X	X	H	H	H	H	H	H	H	H
L	X	X	X	X	H	H	H	H	H	H	H	H
H	L	L	L	L	L	H	H	H	H	H	H	H
H	L	L	L	H	H	L	H	H	H	H	H	H
H	L	L	H	L	H	H	L	H	H	H	H	H
H	L	L	H	H	H	H	H	L	H	H	H	H
H	L	H	L	L	H	H	H	H	L	H	H	H
H	L	H	L	H	H	H	H	H	H	L	H	H
H	L	H	H	L	H	H	H	H	H	H	L	H
H	L	H	H	H	H	H	H	H	H	H	H	L

TABLA 3.3. TABLA DE VERDAD DE 74LS138.

En la figura 3.4 se observan las líneas G2A y G2B conectadas a tierra de forma permanente, y G1 está conectado al reloj ϕ 2, de forma que la decodificación de las señales se encuentra sincronizada con el sistema.

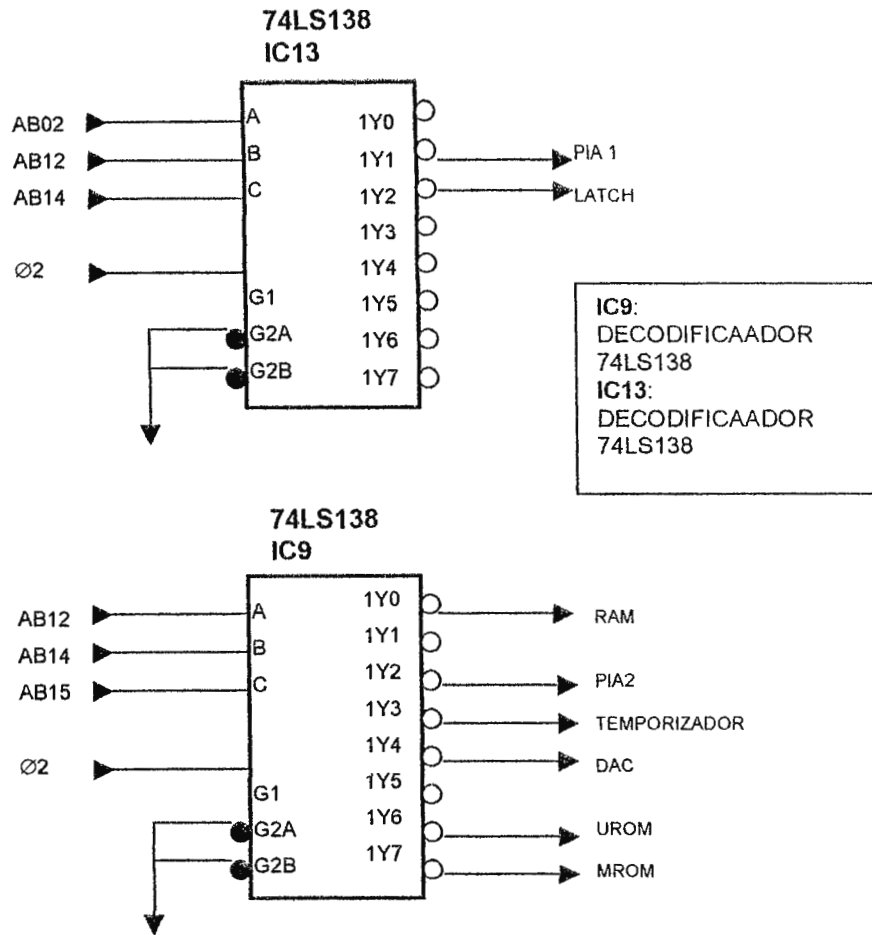


FIGURA 3.4. CIRCUITOS DECODIFICADORES DE ENTRENADOR ASSI.

Las entradas de selección del 74138, (A, B, C), están conectadas a los pines del bus de dirección AB12, AB14 y AB15 para IC9, y AB02, AB12, AB14 para IC13.

La tabla 3.4 muestra el estado de los pines de dirección, cuando son generadas las direcciones por el microprocesador. Como muestra la tabla de verdad dependiendo de la información en los pines A, B, y C, así será la salida seleccionada.

Las salidas 1Y0 y 1Y1 son activadas con las direcciones 0000 hasta 07FF, cualquier dirección generada y que se encuentre en este rango activará la memoria RAM. Esto quiere decir que hay ocho páginas destinadas (00 – 07) a la RAM. Como cada página

posee 256 datos (FF_{HEX}), entonces existen 2048 direcciones reservadas para este dispositivo.

La misma lógica se utiliza para acceder la UROM, la MROM y la PIA del sistema.

DIRECCION	Estado de los pines			Salida 138	
	AB15	AB14	AB13		
0000	0	0	0	1Y0	RAM
0700	0	0	0	1Y1	
E000	1	1	0	1Y4	UROM
E700	1	1	0	1Y5	
F000	1	1	1	1Y6	MROM
F700	1	1	1	1Y7	

TABLA 3.4. ESTADO DE LOS PINES DE DIRECCION.

3.2 RELOJ DEL SISTEMA

Los intervalos de tiempo durante los cuales tienen lugar las operaciones en la CPU son generados por un sistema de reloj de dos fases a una frecuencia de 1 MHz.

El reloj del sistema está compuesto por dos fases, ϕ_1 y ϕ_2 , que ocurren dentro de un ciclo de reloj. Ambas fases son importantes pues de ellas se derivan las diferentes señales de control. (Ver figura 3.5).

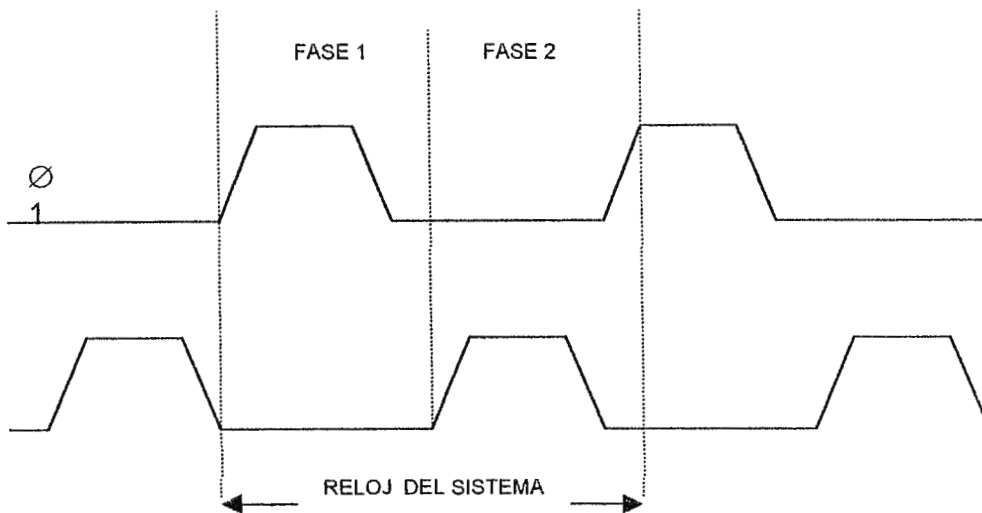


FIGURA 3.5. RELOJ DEL SISTEMA

- **Estabilización de direcciones y datos:**

ϕ_1 y ϕ_2 se utilizan para definir el tiempo en que las direcciones y los datos se hallan en los buses adecuados. Las direcciones se estabilizan durante la fase 1 y permanecen estables el resto del ciclo. Los datos se estabilizan durante la fase 2, y permanecen estables solo el pequeño intervalo de tiempo restante de ϕ_2 . (Ver figura 3.6).

Mientras $\phi 2$ es baja, la dirección del bus de direcciones ya es estable, tal como se indica en la figura 3.6, pues se estabilizó durante la fase 1.

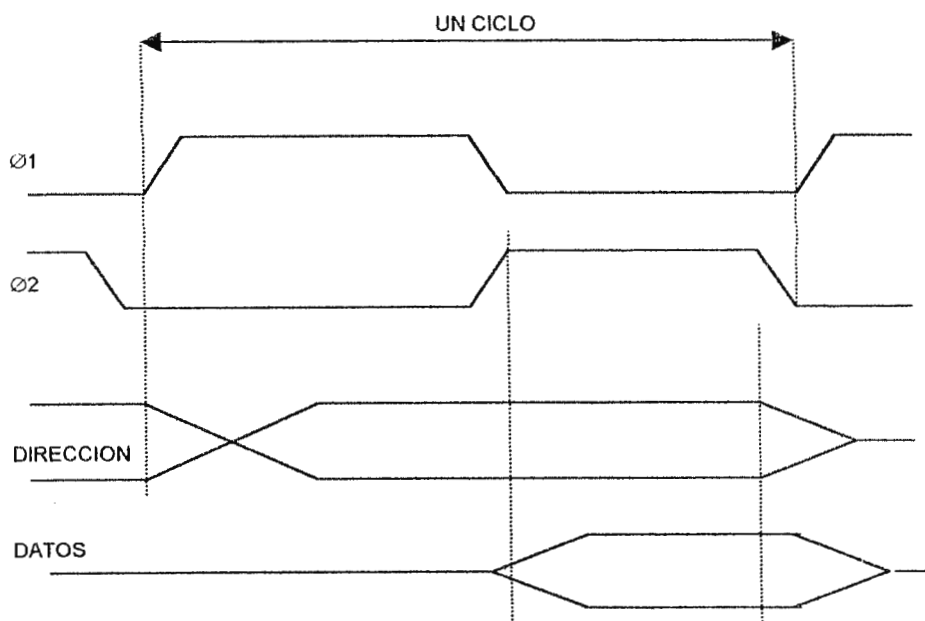
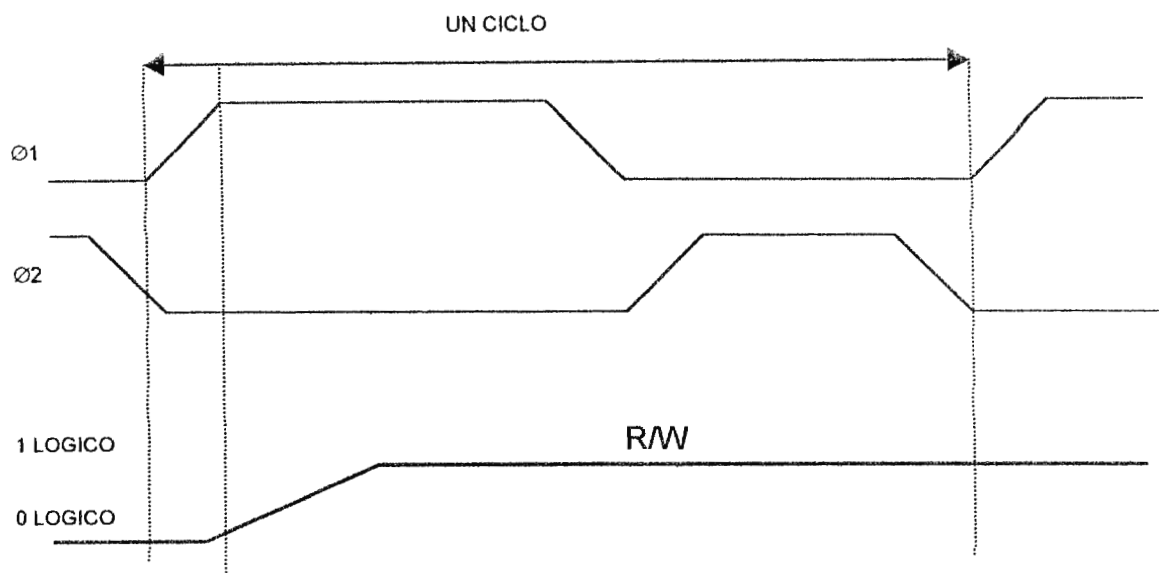


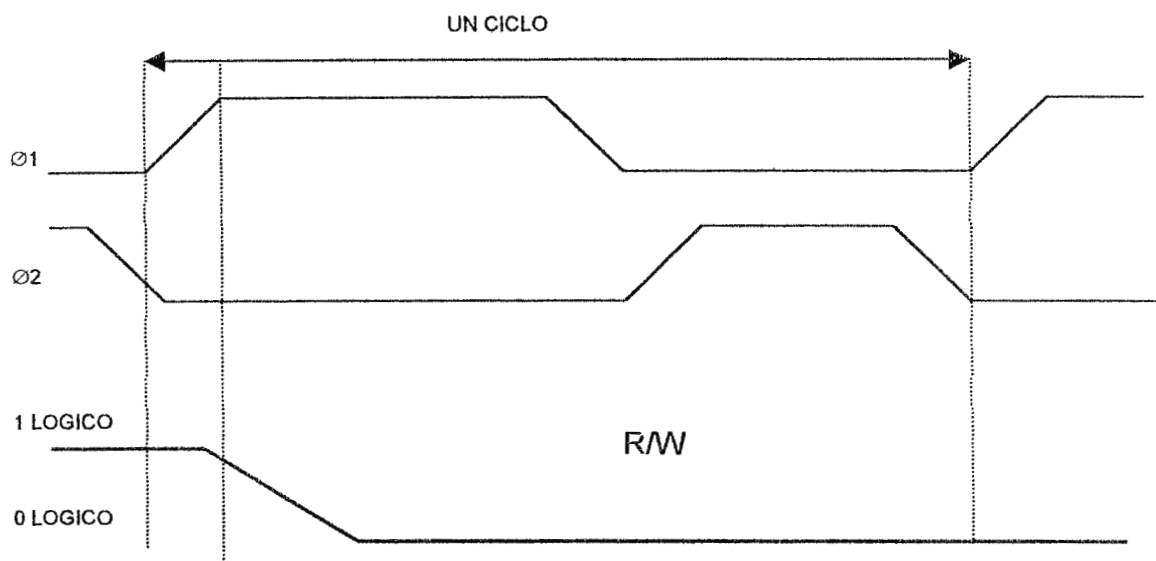
FIGURA 3.6. SEÑALES DE DIRECCION Y DATOS.

La CPU debe capturar los datos durante el breve intervalo de tiempo mencionado, para lo que requiere varias señales de control.

READ/WRITE ($\overline{R/W}$) es una de las señales de control producidas por la CPU, La señal $\overline{R/W}$ es baja cuando a CPU escribe datos en la memoria y es alta cuando los lee. En la figura 3.7 se puede observar las señales $\overline{R/W}$. En (a) se presentan las condiciones cuando la CPU lee, y en (b) cuando escribe. En estas figuras se observa el intervalo de tiempo antes de la estabilización de la señal, después que la segunda fase se vuelve baja.



(a) R/W cuando la CPU esta leyendo.



(b) R/W cuando la CPU esta escribiendo.

FIGURA 3.7. SEÑALES DE LECTURA Y ESCRITURA.

La dirección se halla presente en la posición 0 lógico de $\phi 2$ y permanece hasta el flanco posterior de $\phi 2$. Los datos solo se hallan presentes durante la parte de 1 lógico de $\phi 2$.

Las líneas de dirección cambian cuando se produce el flanco posterior de la señal de reloj $\phi 2$ y se coloca otra dirección en el bus de direcciones durante el intervalo $\phi 1$. Los datos del bus de datos indicados en la figura 3.8 se estabilizan y permanecen estables después de un tiempo denominado T_{DS} durante $\phi 2 = 1$; T_{DS} es el tiempo de fijación de los datos (*DATA SETUP TIME*) de 150 ns especificado por el fabricante; una vez que el reloj se torna alto. Los datos permanecerán estables además durante cierto periodo de tiempo, denominado T_H , al comenzar el próximo ciclo. T_H es el tiempo de retención requerido por la CPU para escribir o leer datos.

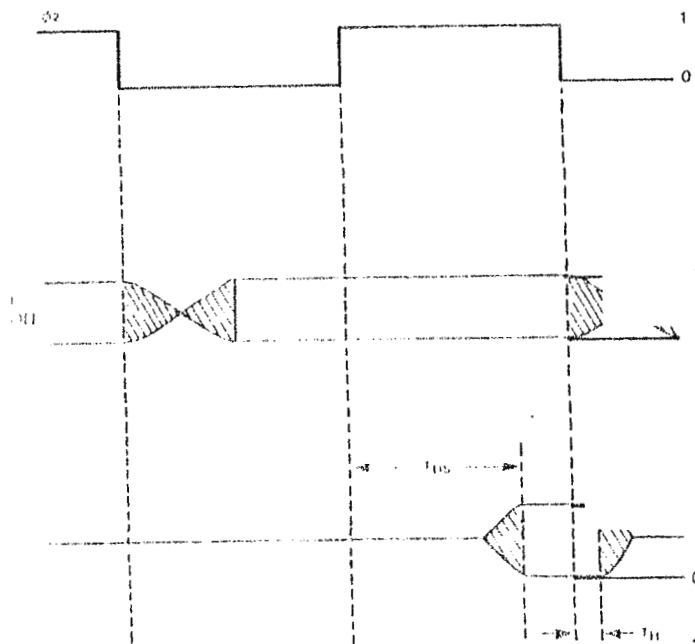


FIGURA 3.8. RELACION DE TIEMPO DURANTE FASE 2

Para establecer el tiempo correcto de observación de los datos el fabricante recomienda implementar el circuito de la figura 3.9, el cual combina en función NAND las señales ϕ_2 y R/W , y cuya señal de salida ($R-R/W-B$) se muestra en la figura 3.10.

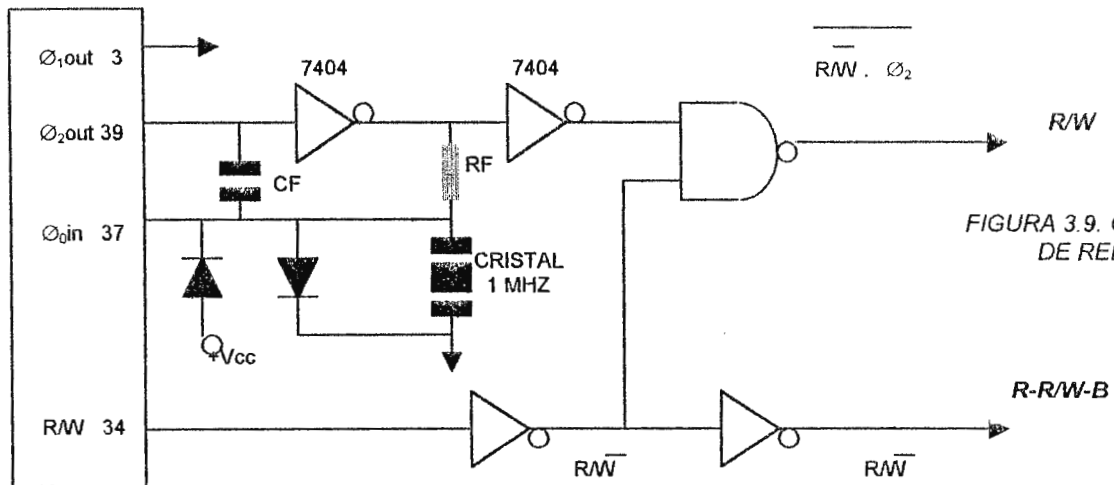


FIGURA 3.9. CIRCUITO DE RELOJ

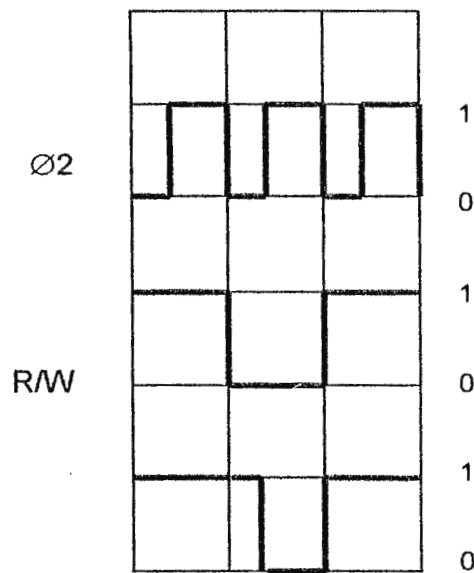


FIGURA 3.10. CONDICION PARA LA SEÑAL R/W

3.3 CONTROL DE ENTRADA/SALIDA DE DATOS DEL ENTRENADOR ASSI

La entrada de los datos a la CPU se controla desde el teclado y la visualización de éstos, así como su resultado se presenta al usuario en el visualizador HDSP-2111.

Para poder realizar estas funciones el ASSI utiliza la PIA 6520, denominada PIA1. Este capítulo se dedica al estudio de la forma en que se programa la PIA1 por medio del μ P 6502 en el entrenador ASSI.

3.3.1 ESTRUCTURA DE LA PIA

La PIA1 se divide en dos secciones, una conectada a la CPU y la otra conectada a ambos dispositivos periféricos, tal como se muestra en la figura 3.11.

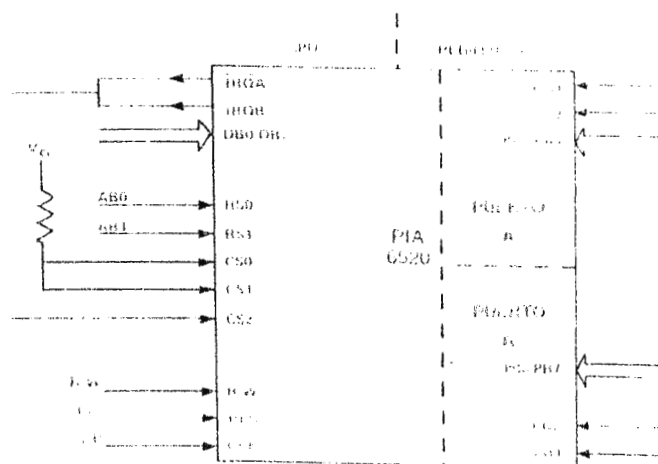


FIGURA 3.11. SECCIONES DE LA PIA.

3.3.2 REGISTROS DE LA PIA

La PIA posee dos puertos de salida: A y B, tal como se muestra en la figura 3.12.

Cada puerto consta de tres registros principales de 8 bits;

- REGISTROS DE CONTROL; CRA y CRB.
- REGISTROS DE DIRECCION DE DATOS; DDRA y DDRB.
- REGISTRO DE SALIDA A PERIFERICOS; ORA y ORB.

Estos registros se seleccionan mediante RS0 y RS1, los cuales se conectan a AB0 y AB1. Los *registros de control* permiten controlar desde la CPU el funcionamiento de la PIA, ya que los datos pueden almacenarse en estos registros y la CPU puede leer dichos datos desde tales registros.

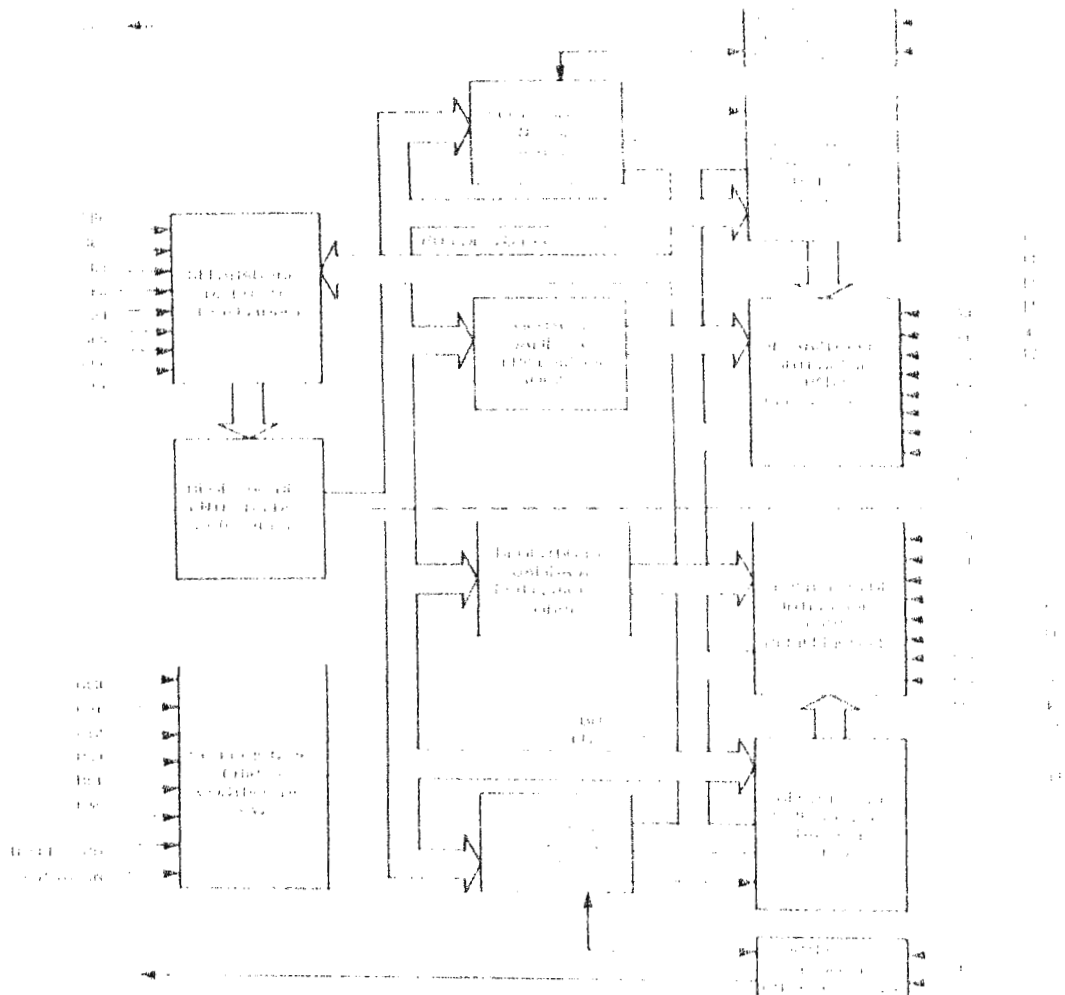


FIGURA 3.12. PUERTOS A Y B DE LA PIA.

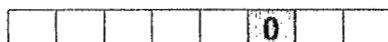
3.3.3 DIRECCIONAMIENTO DE LOS REGISTROS DE LA PIA

El registro de salida a periféricos y el de dirección de datos de cada puerto comparten una dirección. Los registros de control CRA y CRB poseen una dirección cada uno. Ver figura 3.13.



FIGURA 3.13. REGISTROS DE PIA Y SUS DIRECCIONES.

El bit 2 de cada registro de control se utiliza para seleccionar el registro de salida o el registro de dirección de datos, de la siguiente forma:



SELECCIONA REGISTRO DE DIRECCION DE DATOS



SELECCIONA REGISTRO DE SALIDA

Las condiciones necesarias en las líneas de selección de registros RS1 y RS0 y los bits de acceso del registro de control para el direccionamiento de los registros internos de la PIA se presentan en la tabla 3.5.

LINEAS DE SELECCIÓN DE REGISTROS		BIT DE ACCESO DEL REGISTRO DE CONTROL DDR		REGISTRO SELECCIONADO
RS1	RS0	CRA-1	CRB-2	
0	0	1	X	REG. DE SALIDA A
0	0	0	X	REG. DE DIR. DE DATOS
0	1	X	X	REG. DE CONTROL A.
1	0	X	1	REG. DE SALIDA B.
1	0	X	0	REG. DE DIRECCION B.
1	1	X	X	REG. DE CONTROL B.

X = 0 ó 1

TABLA 3.5. DIRECCIONAMIENTO DE LOS REGISTROS INTERNOS DE LA PIA.

3.3.4 INICIACIÓN DE LA PIA1

Generalmente una puerta se inicia como *salida* y otra como *entrada* cuando se restablece el 6502. A continuación se presenta un ejemplo para la iniciación del puerto A como entrada y el puerto B como salida para el control del teclado y visualización:

```

LDA #$00
STA CRA
STA CRB
LDA #$00
STA DDRA
LDA #$FF
STA DDRB
LDA #$02
STA CRA
STA CRB
LDA ORA
STA ORB

```

} Selecciona DDRA y DDRB
 } Programa el puerto A como entrada como entrada y el puerto B como salida.
 } Selecciona ORA y ORB
 } Carga el dato desde A y lo saca por puerto B.

El sistema de la CPU en el ASSI debe estar reiniciando la PIA1 alternadamente para permitir el acceso desde el teclado y el control del visualizador continuamente.

⇒ Iniciación de ambos puertos como salida para el control del visualizador:

LDA #\$00	}	Selecciona DDRA y DDRB
STA CRA		
STA CRB	}	Programa ambos puertos como salidas
LDA #\$FF		
STA DDRA		
STA DDRB	}	Selecciona ORA y ORB
LDA #\$02		
STA CRA		
STA CRB	}	Saca los datos correspondientes a través de los puertos A y B.
LDA #\$DATO1		
STA ORA		
LDA #\$DATO2		
STA ORB		

⇒ Iniciación para el control del teclado:

LDA #00	}	Selecciona CRA.
STA \$CRA		
LDA #0F	}	Programa el PA. (PA0-PA3 = Salid.) (PA4-PA7 = Ent.)
STA \$DDRB		
LDA #04	}	Habilita la ORA.
STA \$ORA		

3.3.5 REGISTROS DE CONTROL DE LA PIA

Cada línea de control de estado de interrupción posee una línea de control de entrada CA1, CB1 (bit 0 y 1 de CRA y CRB), y una línea de control bidireccional CA2, CB2 (bits 3, 4 y 5 de CRA y CRB).

En la tabla 3.6 se presenta la función de cada bit de los registros de control.

CRA							
7	6	5	4	3	2	1	0
IRQA 1	IRQA 2	CONTROL CA2			ACCESO A DDRA	CONTROL CA1	

CRB							
7	6	5	4	3	2	1	0
IRQB 1	IRQB 2	CONTROL CB2			ACCESO A DDRB	CONTROL CB1	

TABLA 3.6. ORGANIZACIÓN DE LOS REGISTROS DE CONTROL DE LA PIA.

Función de los bits de los registros de control:

- BIT 7: El estado de IRQA1 o IRQB1 se fija con transiciones en la línea de control (CA1, CB1) y se limpia mediante la lectura de los registros de salida (A y B).
- BIT 6: Igual que el bit 7, salvo que se fija por transiciones en la línea de control (CA2, CB2).
- BIT 5: Determina si la línea de control (CA2, CB2) es una entrada (0) o una salida (1)
- BIT 4: Entrada de las líneas de control (CA2, CB2): determina si el bit 6 se fija mediante transiciones de arriba hacia abajo o abajo hacia arriba. Salida de las líneas de control (CA2, CB2): determina si las líneas de control (CA2, CB2) es un pulso o un nivel.

- BIT 3: Entrada de las líneas de control (CA2, CB2): si es 1 permite interrumpir la salida de bit 6. Salida de las líneas de control (CA2, CB2): 0 = línea de control baja, 1 = línea de control alta.
- BIT 2: Selecciona el registro de direccionamiento de datos con 0, o el registro de salida con 1.
- BIT 1: Determina si el bit 7 se fija mediante transiciones de arriba abajo o transiciones de abajo arriba en la línea de control 1.
- BIT 0: Si es 1, permite interrumpir la salida del bit 7 del registro de control²

La conexión entre la PIA1 y los periféricos se realiza a través buffers 74LS244 (ver sección 3.4), como se muestra en la figura 3.15.

² Tomado de "Entrenador de Microprocesador, Conceptos y Aplicaciones", Buck Engineering.

3.4 BUFFER

El visualizador HDSP-2111 (IC4) pertenece a la tecnología CMOS, debido a esto, para controlarlo desde la PIA1 (IC2), de tecnología TTL, no es recomendable realizar una conexión directa debido a la diferencia en las características eléctricas entre la PIA (que impulsa la señal de salida) y el 2111 (que recibe la señal). Se utiliza entonces en la CPU como circuitería de acople entre la PIA1 y el 2111 dos IC 74LS244. (ver figura 3.15). De igual forma se ocupa el 74LS244 para conectar la PIA1 con el teclado.

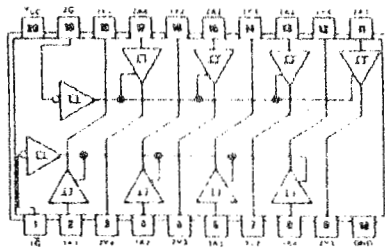


FIGURA 3.15. IC 74LS244

El 74LS244 es un circuito integrado que contiene ocho compuertas separadoras denominadas "buffers" de estado triple³. Este separador octal y manejador de línea está diseñado para recibir y transmitir las señales provenientes del bus de direcciones. (Ver hojas técnicas en apéndice E).

El circuito equivalente interno se presenta en la figura 3.16.

El 74LS244 es un TTL de estado triple que permite que haya tres posibles estados de salida: *alta*, *baja* y de alta impedancia (Hi-Z). En el estado Hi-Z la salida es

³ Compuertas de estado triple son las que además de los estados lógicos 1 y 0, poseen un estado de alta impedancia o "HI-Z".

una terminal flotante que no es ni baja ni alta. En realidad en Hi-Z la salida no es exactamente un circuito abierto, pero tiene una resistencia de varios megohmios.

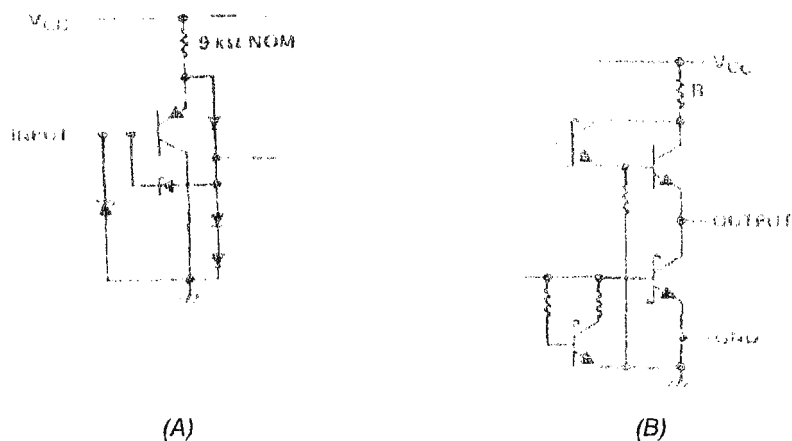


FIGURA 3.16. BUFFER OCTAL, MANEJADOR DE LINEA, NO INVERSOR CON SAIDA DE TRES ESTADOS. (A) CIRCUITO EQUIVALENTE DE CADA ENTRADA (B) CIRCUITO EQUIVALENTE DE CADA SALIDA

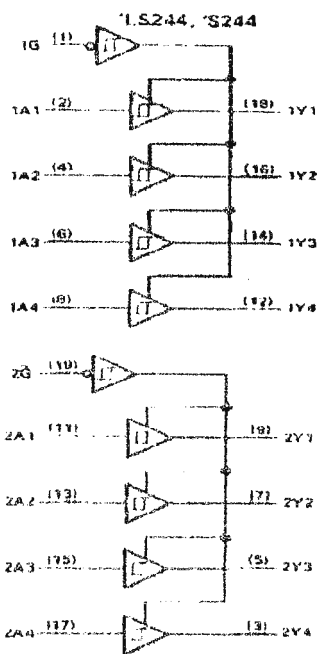


FIGURA 3.17. DIAGRAMA LOGICO DEL 74LS244.

En la figura 3.17 se presenta el diagrama lógico del 74LS244. Con un 0 lógico en 1G se habilitan las salidas 1Y1 – 1Y4, es decir que los buffer transmitirán la señal que se encuentre en 1A1 – 1A 4, sea este un 1 o un 0. Con un 1 lógico en 1G se produce el estado de alta impedancia. De igual forma se comporta 2G para controlar las salidas 2Y1 - 2Y4. Esto se resume en la tabla de verdad que se presenta de la figura 3.18.

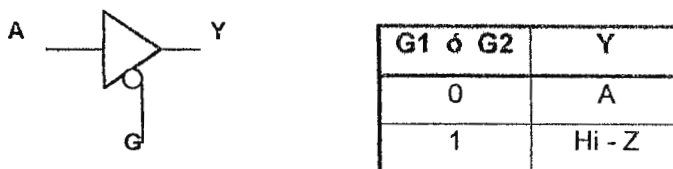


FIGURA 3.18. SIMBOLO Y TABLA DE VERDAD DEL 74LS244.

3.5 MEMORIAS

El sistema de la CPU del ASSI utiliza memorias RAM 2114 y memorias EPROM 2716.

3.5.1 MEMORIAS RAM

Las memorias RAM que se utilizarán en la CPU de la ASSI son memorias 2114, las cuales están clasificadas dentro de las *RAM Estáticas*, debido a que no necesita ciclos de refrescamiento para mantener la información. El diagrama de pines de la 2114 se muestra en a figura 3.19

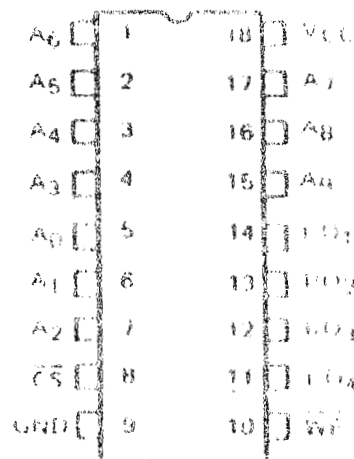


FIGURA 3.19. DIAGRAMA DE PINES DE LA 2114.

- A₀–A₉: pines de dirección.
- I/O₀–I/O₃: pines de datos.
- CS: Selección de pastilla.

- WE: pin de escritura-lectura. WE = 0; escritura. WE = 1; lectura.

La capacidad individual de las 2114 es de 1K X 4. Se utiliza la técnica de expansión del bus de datos, conectando los pines de dirección de dos memorias en paralelo, para lograr almacenar una palabra de 8 bits, en una sola dirección; obteniéndose un banco de memoria con capacidad de 1K X 8, denominado *banco A*. (Ver figura 3.20).

La línea de control RAM-A es generada por el decodificador de memorias IC9, éste se encargan de seleccionar el banco de memoria cuando es solicitado por el microprocesador, a través del bus de direcciones. (Ver sección 4.2)

3.5.2 MEMORIAS ROM.

Las memorias ROM que se utilizan en el ASSI son EPROM 2716, que posee una capacidad de 2K X 8; destinadas a almacenar el programa monitor, en la memoria MROM y los programas de usuario, en la memoria UROM, los cuales son de carácter permanente. La línea de control es generada por el decodificador IC9. (Ver sección 4.2)

Las memorias EPROM son detalladas ampliamente en la sección 2.4.4.

El diagrama esquemático de las memorias se muestra en la figura 3.20.

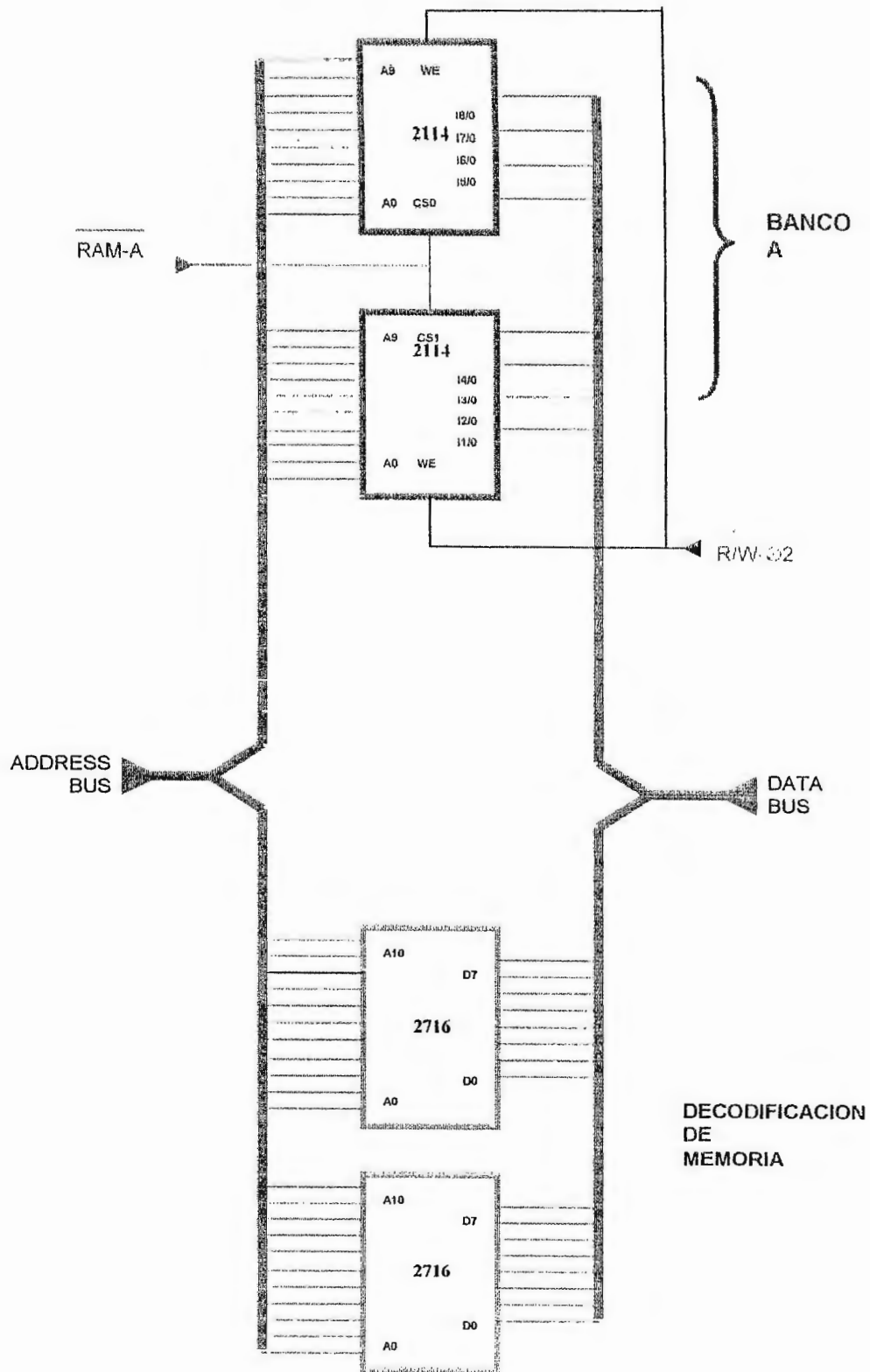


FIGURA 3.20. DIAGRAMA ESQUEMATICO DE MEMORIAS.

3.6 CONTROL DEL VISUALIZADOR

El control de visualizador se realiza mediante una rutina del programa monitor. El visualizador del ASSI es el HDSP-2111 (IC4) fabricado por HEWETT PACKARD, Las hojas técnicas correspondientes se presentan en el apéndice E.

Se escogió el HDSP-2111 debido a su diseño para aplicaciones con microprocesadores. Este consiste en un visualizador alfanumérico de 28 pines con ocho dígitos, cada uno formado por una matriz de 5 x 7.

El HDSP-2111 posee tres modos de operación:

- CHARACTER. Accesa a 128 caracteres ASCII guardados de forma permanente en memoria ROM.
- UDC. Permite al usuario definir sus propios caracteres y guardarlos en memoria.
- FLASH. Permite programar cada matriz para que se encienda a frecuencias determinadas.

El ASSI utiliza el modo CHARACTER. Para seleccionar el modo de operación deseado se utiliza la tabla de verdad de "Niveles Lógicos de Acceso a Memoria" mostrado en la tabla 3.7.

SECCIÓN DE MEMORIA	FL	A ₄	A ₃	A ₂ A ₁ A ₀
Flash RAM	0	X	X	Dir. Char
Registro de direccionamiento UDC	1	0	0	No Importa
UDC RAM	1	0	1	Dir. Fila
Registro de Control	1	1	0	No Importa
RAM CHARACTER	1	1	1	Dir, Char.

TABLA 3.7. NIVELES LOGICOS

MODO CHARACTER.

En la tabla 3.8 se presentan los niveles lógicos necesarios para utilizar la RAM de caracteres del HDSP-2111. Durante un acceso normal, la señal de (Habilitación de interruptores) CE, y la señal de (Lectura) RD o de (Escritura) WR se mantienen en cero. Sin embargo datos erróneos pueden ser escritos en las RAM si los valores en las líneas de direccionamiento se encuentran inestables cuando CE=0, dependiendo de los niveles lógicos de WR y RD.

RST	CE	WR	RD
1	0	0	0
		0	1
		1	0
		1	1

TABLA 3.8 a. SEÑALES DE CONTROL.

FL	A ₄	A ₃	A ₂	A ₁	A ₀
1	1	1	CHARACTER RAM		

TABLA 3.8 b. DIRECCIONAMIENTO DE RAM CARÁCTER.

D7	D6	D5	D4	D3	D2	D1	D0
0	128 ASCII CODE						
1	X	X	X	UDC CODE			

TABLA 3.8 c. FORMATO DE LOS DATOS DE LA RAM DE CARACTERES.

DIG 0	DIG 1	DIG 2	DIG 3	DIG 4	DIG 5	DIG 6	DIG 7
000	001	010	011	100	101	110	111

VISUALIZADOR

0 = 0 LOGICO; 1 = 1 LOGICO; X = NO IMPORTA.

TABLA 3.8 d. NIVELES LOGICOS PARA ACCESAR A LA CHARACTER RAM.

Las líneas de direccionamiento A_0 - A_2 se utilizan para selección de localizaciones del carácter de la RAM. Los tipos de datos que pueden guardarse en cada localización son: códigos ASCII o una dirección específica de la UDC RAM (User Defined Character RAM).

El bit D7 es utilizado para diferenciar entre los caracteres ASCII y la dirección de la UDC RAM. Cuando el bit D7 = 0 se activa el decodificador ASCII, cuando D7 = 1 se habilita la UDC RAM. Dependiendo de la condición de D7, los bits D0 – D6 son usados como entradas de direcciones de la UDC RAM.

REGISTRO DE CONTROL

La tabla 3.9 muestra como se accesa al registro de control. Este registro de 8 bits desarrolla cinco funciones:

- Control de brillo.
- Control del Modo de Flash RAM.
- Control de intensidad.
- Control de Modo de Prueba.
- Limpiar.

Cada función es independiente de las otras.

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	
C	S	B	BI	F	B	B	B	
					0	0	0	30%
					0	0	1	40%
					0	1	0	50%
					0	1	1	60%
					1	0	0	70%
					1	0	1	80%
					1	1	0	90%
					1	1	1	100%

BRIGHTNESS CONTROL LEVELS

0 DISABLE FLASH
 1 ENABLE FLASH
 0 DISABLE BLINKING
 1 ENABLE BLINKING
 0 NORMAL OPERATION X IS NORMAL
 X START SELF TEST, RESULT GIVEN IN X
 0 FAILED X PASSED

0 NORMAL OPERATION
 1 CLEAR FLASH AND CHARACTER RAMS

CONTROL WORD DATA FORMAT
 0 LOGIC 0; 1 LOGIC 1; X DO NOT CARE

TABLA 3.9. NIVELES LÓGICOS PARA ACCESAR A LA PALABRA DE CONTROL.

3.6.1 DISEÑO

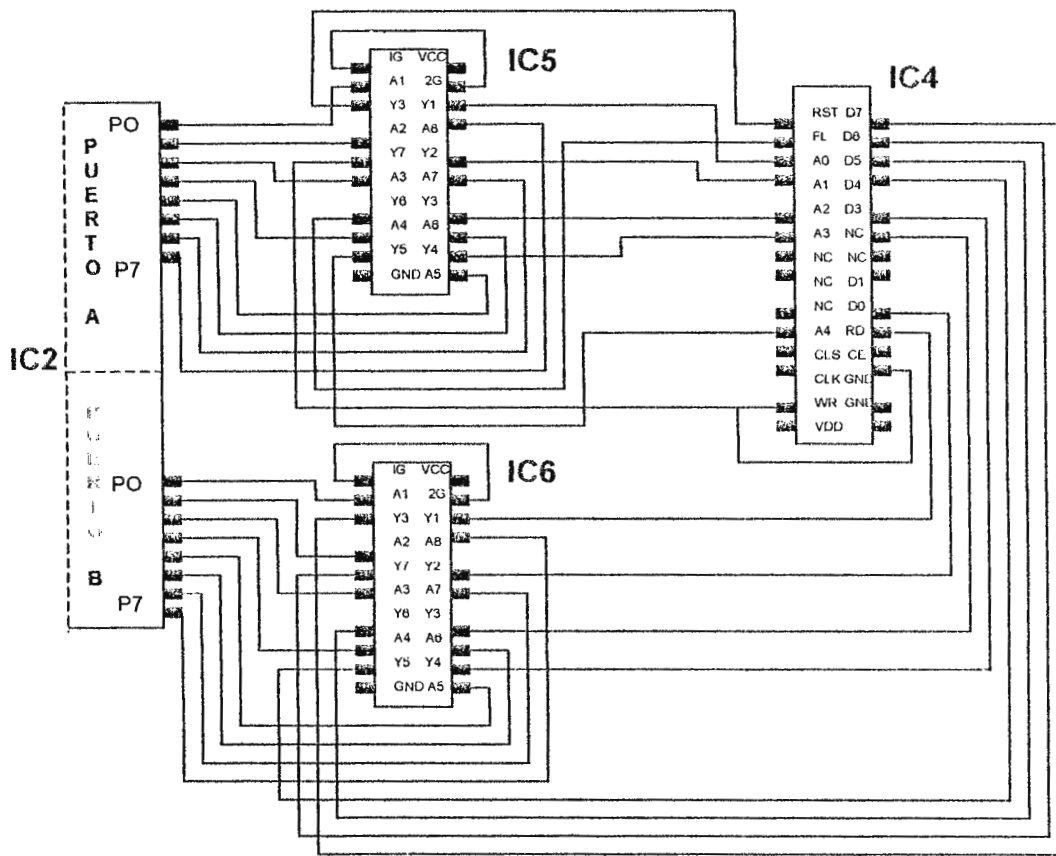
En la figura 3.21 se observa el circuito de control del visualizador. El control se realiza a través de los puertos de la PIA 6520 denominada PIA1 (IC2), y de dos buffers 74LS244 (IC6 e IC8), encargados de acoplar las señales TTL con CMOS. La función específica de éstos se detalla ampliamente en la sección 3.4.

Basándose en el circuito de control de visualizador HDSP-2111, el acceso para su operación se ha dividido en dos partes: el acceso al registro de control y el acceso a la memoria interna del visualizador (modo carácter).

Los accesos son controlados por el modo de operación (ver tabla 3.10), el puerto A de la PIA1 se encarga de proporcionar esta información, la palabra A para el registro de control y la palabra B para la memoria interna del Visualizador.

NUMERO BINARIO								HEX	MODO SELECCIONADO
RST	C/W	FL	A4	A3	A2	A1	A0		
1	0	1	0	0	0	0	0	A 0	REG. DE DIRRCCION UDC
1	0	1	0	1	0	0	0	A 8	RAM DE UDC
1	0	1	1	0	0	0	0	B 0	REGISTRO DE CONTROL
1	0	1	1	1	0	0	0	B 8	CHARACTER
1	0	0	X	X	0	0	0	- -	FASH RAM

TABLA 3.10. MODO DE OPERACIÓN



IC2 = PIA
 IC5 = 74LS244
 IC6 = 74LS244
 IC4 = HDSP-2III

FIGURA 3.21. CIRCUITO DE CONTROL DEL VISUALIZADOR

Palabra de control A.

PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
RST	CE/WR	FL	A4	A3	A2	A1	A0
1	0	1	1	0	X	x	x

TABLA 3.11. INFORMACION DE LA PALABRA DE CONTROL A.

El código hexadecimal de esta palabra puede ir desde "B0" hasta "B7", debido a que no importa el contenido de los bits A_2 , A_1 y A_0 .

Una vez accesado el registro de control se procede a guardar en él la configuración deseada. Esta configuración es enviada a través del puerto B y contiene la siguiente información:

PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0	HEXA.	% DE BRILO
C	S1	S0	BL	F	B2	B1	B0		
0	0	0	0	0	0	0	0	00	100
0	0	0	0	0	0	0	1	01	80
0	0	0	0	0	0	1	0	02	53
0	0	0	0	0	0	1	1	03	40
0	0	0	0	0	1	0	0	04	27
0	0	0	0	0	1	0	1	05	20
0	0	0	0	0	1	1	0	06	13
0	0	0	0	0	1	1	1	07	0

TABLA 3.12. CONFIGURACION DE REGISTROS DE CONTROL.

De la tabla 3.12 se seleccionó un porcentaje de brillo del 53%, al cual corresponde el código hexadecimal 02.

- Palabra de control B. (Ver tabla 3.13).

PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
RST	CE/WE	F	A4	A3	A2	A1	A0
1	0	1	1	1	0	0	0

TABLA 3.13. INFORMACION DE LA PALABRA DE CONTROL B.

El código hexadecimal de esta palabra puede variar desde B8 hex hasta B7 hex, los bit A0, A1 Y A2 determinan a posición de carácter en el visualizador, ver tabla 3.18. Si se quiere acceder al dígito número 5 del visualizador, solo basta mandar el código hexa "B4".

NUMERO BINARIO								HEX	POSICION								No. DE REGISTRO EN EL VISUALIZADOR
RST	CW	FL	A4	A3	A2	A1	A0		0	1	2	3	4	5	6	7	
1	0	1	1	1	0	0	0	B0	■								DIG 1
1	0	1	1	1	0	0	1	B1		■							DIG 2
1	0	1	1	1	0	1	0	B2			■						DIG 3
1	0	1	1	1	0	1	1	B3				■					DIG 4
1	0	1	1	1	1	0	0	B4					■				DIG 5
1	0	1	1	1	1	0	1	B5						■			DIG 6
1	0	1	1	1	1	1	0	B6							■		DIG 7
1	0	1	1	1	1	1	1	B7								■	DIG 8

TABLA 3.14. CONFIGURACION DE LA PALABRA DE CONTROL B.

Una vez a palabra B ha sido colocada en el Puerto A de la PIA1 se procede en el puerto B a enviar el dato que se quiere mostrar en el dígito seleccionado, la información en el puerto B es la que se muestra en la tabla 3.15.

PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
D7	D6	D5	D4	D3	D2	D1	D0
0	128 CODIGOS ASCII						

TABLA 3.15. FORMATO PARA EL DATO DE LA CHARACTER RAM

Desde PB0 hasta PB6 manejan el campo del carácter ASCII, la información puesta aquí debe corresponder al código binario de carácter seleccionado. En la tabla 3.16 pueden verse las combinaciones posibles.

ASCII Character Set HDSP-210X, HDSP-211X, HDSP-250X Series

		D7	D6	D5	D4	D3	D2	D1	D0	CHARACTER					
		0	0	0	0	0	0	0	0	X					
		0	0	1	1	0	1	1	0	X					
		0	1	0	1	0	1	0	1	X					
		1	1	0	1	0	1	0	1	X					
		1	1	1	1	0	1	0	1	X					
		1	1	1	1	1	1	0	1	X					
		1	1	1	1	1	1	1	1	X					
D4	D3	D2	D1	D0	COLUMN										
↓	↓	↓	↓	↓	↓	0	1	2	3	4	5	6	7	8F	
ROW															
0000	0	[Matrix visualization]													UNDEFINED CHARACTERS
0001	1	[Matrix visualization]													
0010	2	[Matrix visualization]													
0011	3	[Matrix visualization]													
0100	4	[Matrix visualization]													
0101	5	[Matrix visualization]													
0110	6	[Matrix visualization]													
0111	7	[Matrix visualization]													
1000	8	[Matrix visualization]													
1001	9	[Matrix visualization]													
1010	A	[Matrix visualization]													
1011	B	[Matrix visualization]													
1100	C	[Matrix visualization]													
1101	D	[Matrix visualization]													
1110	E	[Matrix visualization]													
1111	F	[Matrix visualization]													

TABLA 3.16. CODIGOS BINARIOS DE CARACTERES ASCII.⁴

⁴ La tabla se ha obtenido de las hojas de datos del fabricante que se encuentran en línea en Internet, para ofrecer una detallada representación de cómo se forman los datos en cada matriz del visualizador.

3.6.2 PROGRAMA

Para explicar la forma en que el uP6502 controla al 2111 a través de la PIA1, se presenta un programa que realiza las siguientes funciones:

- Inicia la PIA1 con PA y PB como salida.
- Salva los registros que se pueden alterar en el proceso.
- Coloca los tributos en los registros internos del Visualizador.
- Muestra la información presente en el Buffer del Visualizador.

La subrutina A es la encargada de presentar la información requerida al visualizador. El flujograma de esta rutina puede observarse en la figura 3.22.

Se ha dividido el flujograma en cuatro partes:

- La primera parte (A) es la inicialización. Aquí los puertos de la PIA1 A y B son configurados como salidas. Se guarda el contenido de los registros a utilizar para no alterar el proceso de otro programa que haya llamado a éste. Se carga al registro X con "08". El registro X es utilizado como puntero que lleva el conteo de la posición del dígito al cual se va a escribir.
- La segunda parte: Se colocan los atributos del Visualizador, primero se envía al puerto B de la PIA1 "02_{HEX}", con esto se le dice al Visualizador que tenga un brillo del 53% y que la operación sea normal.

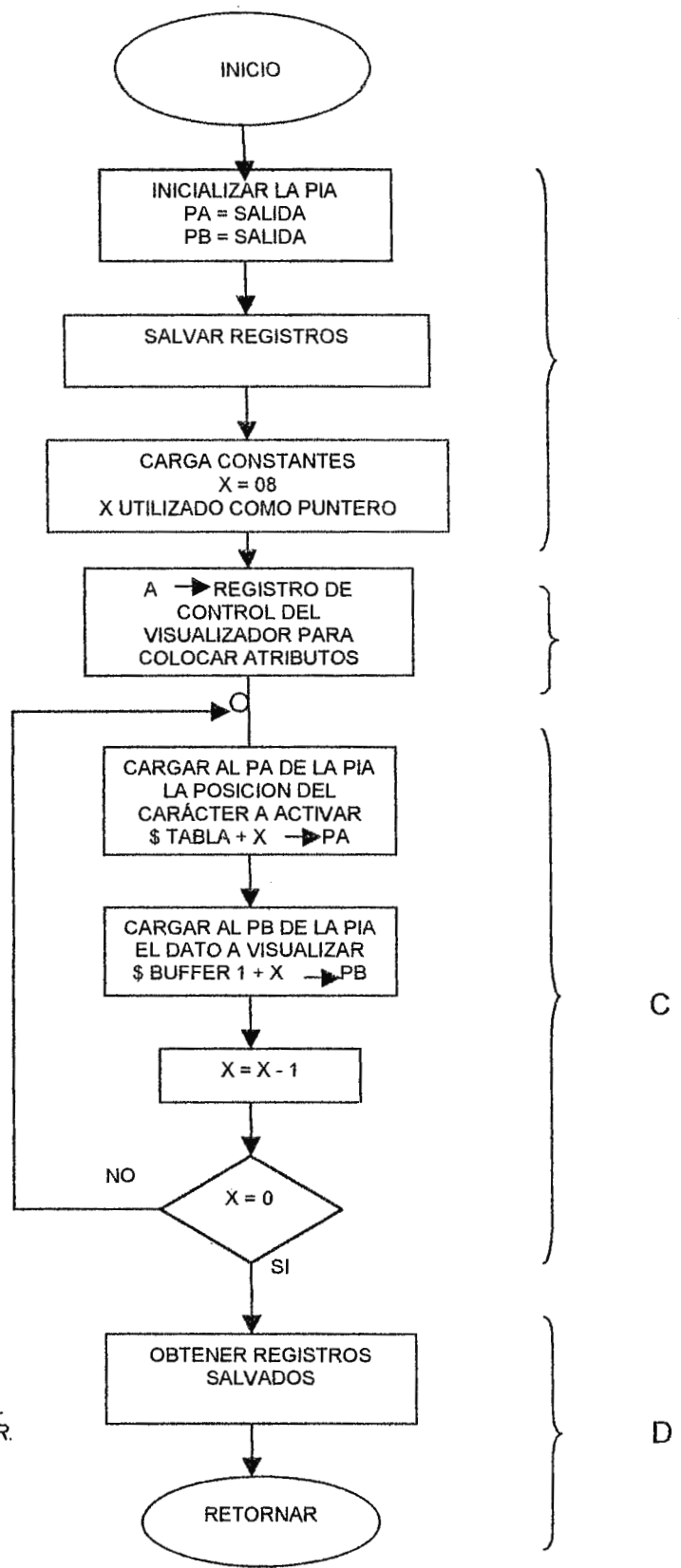


FIGURA 3.22.
FLUJOGRAMA
CONTROL DEL
VISUALIZADOR.

Luego se manda al puerto A, la *palabra de control A* "B0 hex" que se encarga de acceder al registro de control. Cuando esto sucede el Visualizador se configura con el dato que se encuentra en el puerto "B".

Hasta aquí el visualizador está listo para trabajar en operación normal y con una brillantez del 53%.

- Tercera parte: Es la encargada de escribir el contenido del Buffer de salida en el Visualizador.

El Buffer de salida es una región de a memoria RAM que se encuentra en la página 0 y cuya dirección es \$0031 hasta \$0038. Aquí los programas vienen a depositar los mensajes que ellos quieren mandar al Visualizador.

Primero se carga al puerto A de la PIA1 la posición del carácter a activar, esta es la palabra de control B. La posición del carácter a activar es una constante fija que esta almacenada en a memoria ROM cuya dirección es \$E501, a esta dirección se le ha llamado *Tabla 1*. El contenido de *Tabla 1* es:

TABLA 1: Configuración del visualizador.

F501	B8
F502	B9
F503	BA
F504	BB
F505	BC
F506	BD
F507	BE
F508	BF

Como se observa en la *Tabla 1* posee los códigos hexadecimales para ir seleccionando el dígito al cual se va a escribir. (ver tabla 3.14).

La tabla 3.17 muestra el código hexadecimal de *Tabla 1* correspondiente al dígito seleccionado.

VISUALIZADOR	DIG0	DIG1	DIG2	DIG3	DIG4	DIG5	DIG6	DIG7
CODIGO HEXADECIMA	B8	B9	BA	BB	BC	BD	BE	BF

TABLA 3.17. DIGITO SELECCIONADO.

El programa debe acceder en forma ordenada tanto al Buffer de salida como a *Tabla 1*.

A la PIA1 se manda primero el dato de *tabla 1* al puerto A y luego al puerto B se manda el código del carácter que se quiere mostrar al puerto B.

El registro x que sirve como puntero, toma importancia cuando por medio de él se accesa en forma indexada (ver modo en el direccionamiento capítulo 2) al Buffer y a la *tabla 1*.

Ejemplo:

Si asumimos que el registro x es igual 08_{HEX} , entonces la instrucción:

LDA #(TABLA 1+X)

se lee así:

“Cargar a acumulador a constante almacenada en la dirección $\$E501+08$ ”.

Si operamos $E500+08=E508$; entonces el valor almacenado en la dirección E508 que es “BF_{HEX}”, es llevado al acumulador. Luego el acumulador es enviado al puerto A.

De la misma forma se obtiene de la instrucción:

LDA # (BUFFER1+X)

que es el código de carácter que se desea enviar.

En este caso este carácter será visualizado en el dígito D8 del Visualizador.

Siguiendo con el programa, una vez mandados los contenidos de *Tabla 1* y Buffer 1 accedados. El programa decrementa en “uno” (1) el “registro x”, verifica si esa operación es igual a “cero” (0). De no serlo el programa extraerá con ayuda del “registro x” los siguientes códigos almacenados en *tabla 1* y Buffer 1. Si el resultado es cero, significa que todo el mensaje ha sido desplegado en el Visualizador.

- Cuarta Parte:

En esta parte se reponen los valores de los registros que se almacenaron al principio y se da la instrucción de retorno.

3.6.3 VENTAJAS DEL 2111

- No hace falta estar refrescando la información para almacenar los datos ya que posee memoria interna y un reloj de refrescamiento.
- Tiene capacidad para regular el brillo.
- Es altamente compatible con microprocesadores.
- Acepta códigos ASCII.

RUTINA A: Visualiza el contenido del buffer de visualizador.

F000	A9	00		LDA	#00
F002	8D	01	08	STA	\$CRA
F005	8D	03	08	STA	\$CRB
F008	A9	FF		LDA	#FF
F00A	8D	00	08	STA	#DDRA
F00D	8D	02	08	STA	3DDRB
F010	A9	04		LDA	#04
F012	8D	01	08	STA	\$CRA
F015	8D	03	08	STA	\$CRB
F018	8A			TXA	
F019	48			PHA	
F01A	A2	08		LDX	#08
F01C	A9	02		LDA	#02
F01F	8D	02	08	STA	\$ORB
F021	A9	B0		LDA	#B0
F023	8D	00	08	STA	\$ORA
F026	BD	00	F5	A: LDA	# (TAB1 + X)
F029	8D	00	08	STA	\$ORB
F02C	B5	30		LDA	# (BUF1 + X)
F02F	8D	02	08	STA	\$ORB
F031	CA			DFX	
F032	D0		F3	BNF	A:
F034	48			PLA	
F035	AA			TAX	
F036	60			RTS	

3.7 CONTROL DE TECLADO

El flujograma correspondiente a la rutina "Sensor Teclado" se presenta en la figura 3.23, y se explica a continuación.

I Parte

Se encarga de inicializar la PIA1 y las constantes utilizadas por los registros. El puerto A de la PIA1 es inicializado así:

- Desde PA0 hasta PA3 salida, estas líneas serán válidas para operaciones de escritura.
- Desde PA4 hasta PA7 entrada, estas líneas serán válidas para operaciones de lectura.

El "*Registro Y*" es utilizado como puntero y lleva el conteo de la tecla máxima.

El "*Registro Acumulador*" se carga con "EF", este valor servirá para ir habilitando una columna del teclado a la vez. Solo la parte baja de esta palabra contiene la información válida para el teclado.

II Parte.

Es la parte del programa que interactúa directamente con el teclado. La lógica es ir desplazando un cero o nivel bajo en cada columna y al mismo tiempo ir leyendo las filas para ir buscando si el cero generado en la columna pasa a alguna de las filas, este proceso indica si hay una tecla activada.

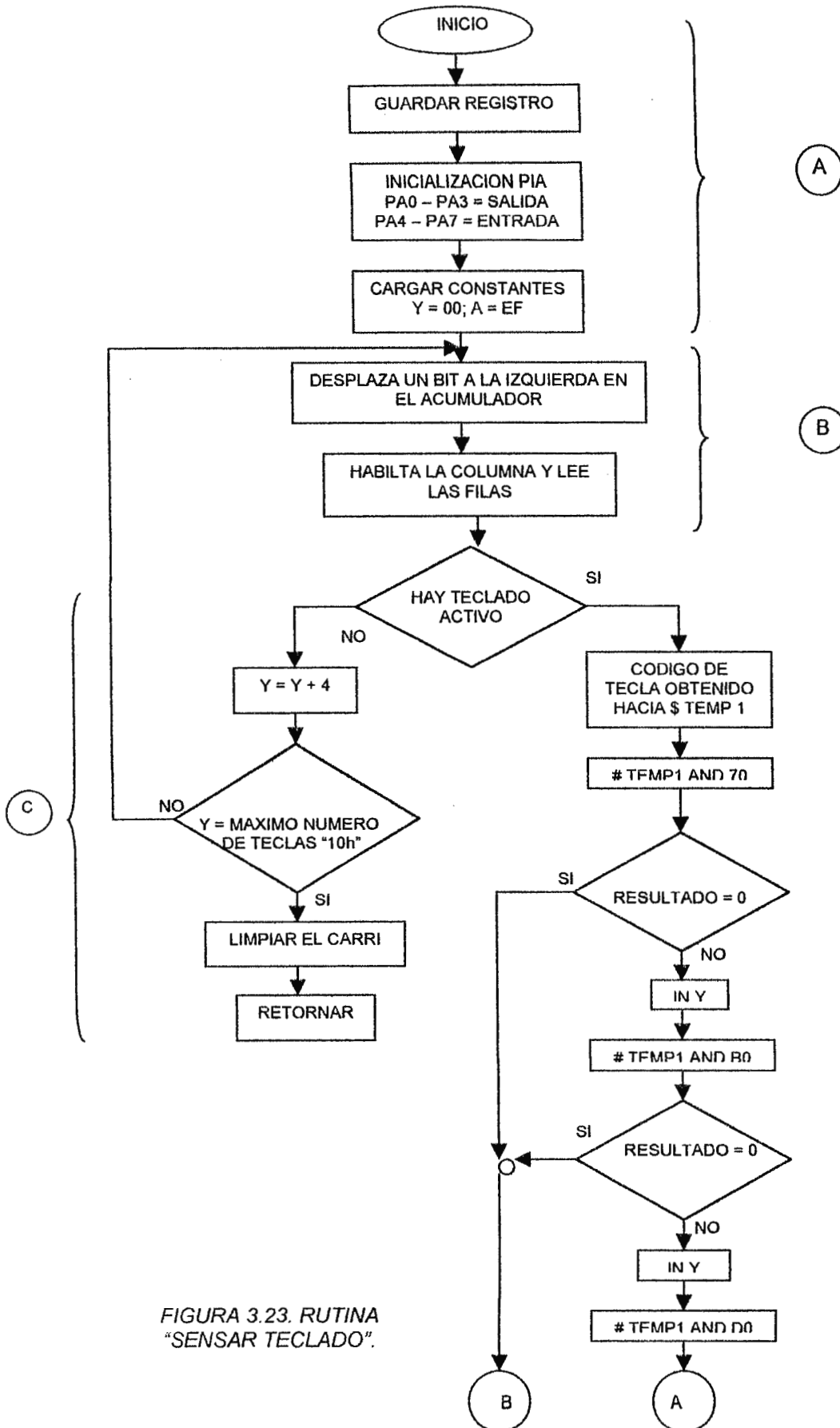


FIGURA 3.23. RUTINA "SENSAR TECLADO".

Detalladamente, la forma en que se habilitan las columnas y se leen las filas es la siguiente:

- ✓ El acumulador es cargado con EF_{HEX} , y se desplaza un bit hacia la derecha obteniendo 77_{HEX} , este valor es mandado al puerto A como se muestra en la figura 3.24.
- ✓ Se observa de la figura 3.24 que en el puerto A solo está presente la parte baja de 77_{HEX} Que es 7, ya que la parte baja (PA0 – PA3) de la PIA1 está configurada como salida y la parte alta; (PA4 – PA7); como entrada.
- ✓ El bit 0 colocado en PA3 habilita toda la columna 4 donde se encuentran las teclas A, B, C y D. Si alguna de estas teclas es presionada el 0 podría desplazarse por la fila de la misma tecla.

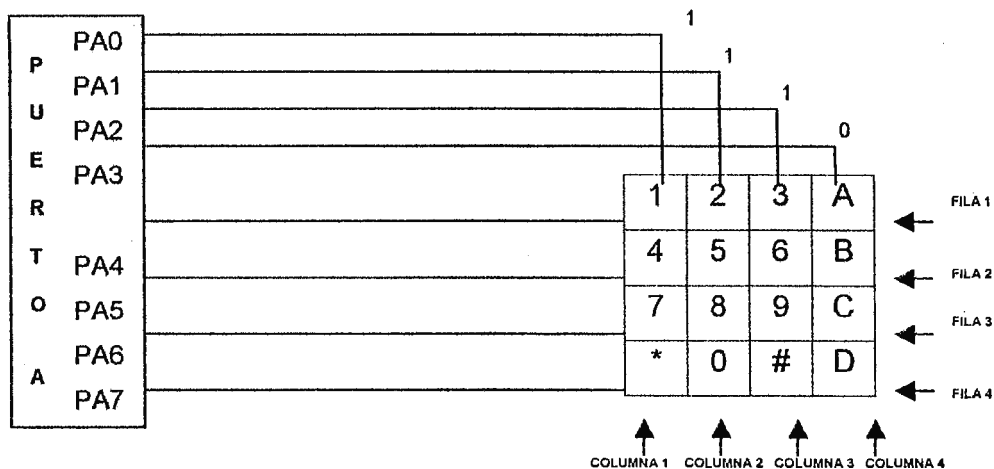


FIGURA 3.24. TECLADO HEXADECIMAL.

- ✓ Suponiendo que la tecla C es presionada, entonces el 0 de la columna 4 se desplazaría por la fila 3 como se muestra en la figura 3.25.
- ✓ Después de haber mandado el código y haber habilitado la columna 4, inmediatamente se procede a leer el contenido de las filas del teclado.

En el registro acumulador se guarda la lectura de las filas si la parte alta del registro acumulador presenta a todos los bits en estado alto entonces no hay tecla activada, y cualquiera de los bits es bajo, entonces hay tecla activada.

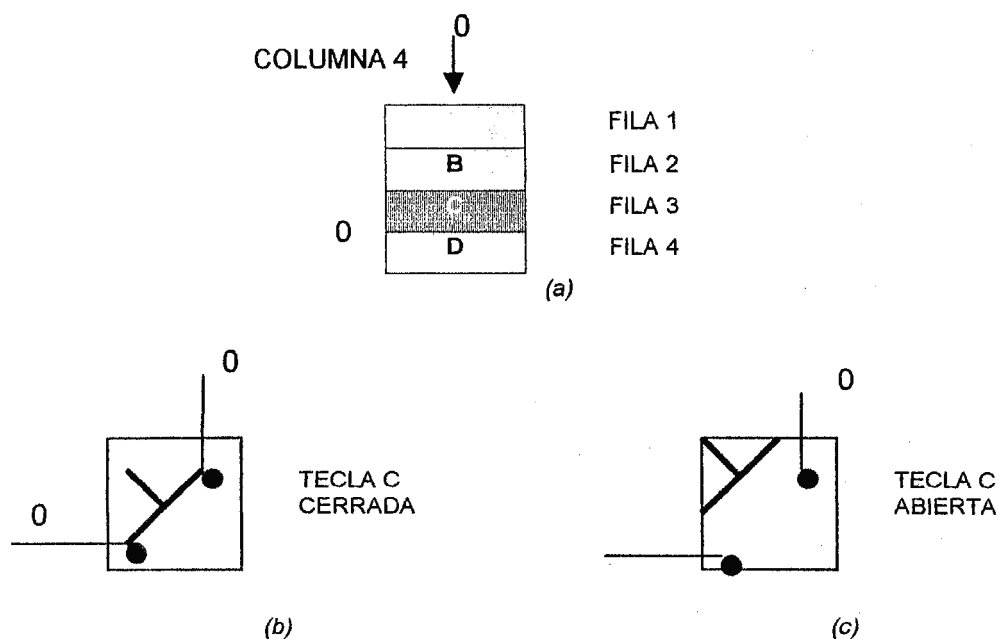


FIGURA 3.25. DETECCION DE TECLA.

III Parte

El programa sigue su lógica cuando no hay una tecla activa.

El puntero "Y" es incrementado cuatro veces, esto se debe a que todas las teclas poseen un dato almacenado en memoria que representa su valor, a esta región de memoria se le ha llamado *Tabla de datos 2* y se encuentra en la dirección \$F509. Entonces "Y" es el puntero de esta tabla de datos.

"Y" es inicializado con 0 y cada vez que una columna es habilitada no hay tecla presionada; "Y" debe moverse al siguiente bloque de datos, ver tabla 3.18.

Como el número máximo de teclas es de 16 (10_{HEX}), se le pregunta a "Y" si ya llego a ese valor. De ser así, significa que al sentir el teclado no hubo tecla activada, el programa limpia la bandera activada avisando a otras subrutinas que no hubo tecla activada, y la subrutina B regresa al programa que lo llamo.

En caso contrario, cuando "Y" no ha llegado a su valor máximo, el programa regresa a la parte "B" del mismo.

F509	0D	Y = 0	1 ^a . COLUMNA
F50A	0C	Y = 1	
F50B	0B	Y = 2	
F50C	0A	Y = 3	
F50D	0E	Y = 4	2 ^a . COLUMNA
F50D	09	Y = 5	
F50E	06	Y = 6	
F50F	03	Y = 7	
F510	00	Y = 8	3 ^a . COLUMNA
F511	08	Y = 9	
F512	05	Y = A	
F513	02	Y = B	
F514	0F	Y = C	4 ^a . COLUMNA
F515	07	Y = D	
F516	04	Y = E	
F517	01	Y = F	

TABLA 3.18. TABLA DE DATOS 2 O "TABLA 2"

IV Parte

Cuando hay una tecla activada, el código binario obtenido en la parte alta del registro acumulador es depurado por dos máscaras binarias AND #F0 y EOR #F0. La primera anula completamente la parte baja de registro A, que es la parte que no interesa. La segunda obtiene el inverso de la parte alta del registro acumulador y luego es almacenado en la dirección \$0009 (constante TEMP1).

Siguiendo con el ejemplo de la tecla "C", el código obtenido en el acumulador una vez leído el PA será:

X = "1" ó "0"

AH	AL
1011	XXXX

✓ Aplicando las máscaras binarias:

```

AND  1011  XXXX
      1111  XXXX
      -----
      0011  0000
  
```

```

EOR  1011  0000
      1111  0000
      -----
      0100  0000
  
```

- ✓ El resultado obtenido en el acumulador es, 40_{HEX}; que es el almacenado en *TEMP 1*.
- ✓ Al depurar a registro acumulador con las dos máscaras, **AND** y **EOR**, se le dice al programa si hay tecla activada, y él decide si pasar a la parte 3 o a la parte 4 del programa.

- ✓ Una vez que ya está el código binario válido en *TEMP1* se procede a aplicar nuevas máscaras binarias con el fin de encontrar el valor de "Y" correspondiente al valor del dato almacenado en la *TABLA 2*, que corresponde a la salida seleccionada.
- ✓ Según la tabla 2, la tecla C se encuentra en la dirección \$F50A "0C".
- ✓ El dato almacenado en \$0009 es 40_{HEX}. Como "C" se encuentra en la primera columna entonces el valor de "Y" es igual a "00".
- ✓ Aplicando la primera máscara a TEMP se obtiene

```

AND    0100  0000
        0111  0000
                
        0100  0000

```

- ✓ Como el resultado no es igual a cero, entonces se incrementa "Y" en 1, y se aplica la segunda máscara:

```

AND    0100  0000
        1011  0000
                
        0000  0000

```

- ✓ El resultado obtenido es igual a cero, entonces el programa salta a cargar al acumulador el valor de la *TABLA 2* correspondiente a la tecla seleccionada.

A = [\$ Tabla2 + Y]

A = [\$F509 + 01]

A = [\$ F504]

A = 0C

V Parte

En la parte 5 del flujograma, se obtienen los datos de los registros guardados, el CARRY es fijado en alto para avisar a otros programas que se ha oprimido una tecla que será almacenada en la constante #tecla que se encuentra en la dirección \$0010. Después de la rutina B, retorna el programa que la llama.

En la figura 3.26 se presenta el circuito de control del teclado.

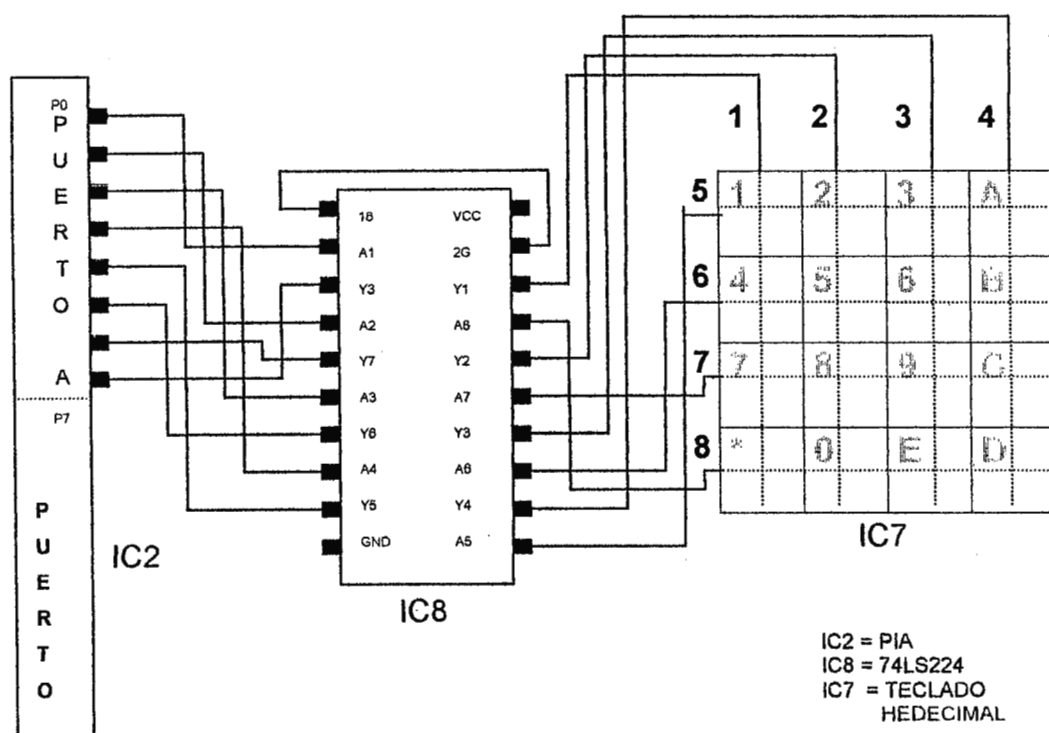


FIGURA 3.26. CIRCUITO DE CONTROL DE TECLADO.

CAPITULO 4

DESCRIPCION Y FUNCIONAMIENTO DEL HARDWARE DEL ENTRENADOR ASSI

El entrenador ASSI es un PLC creado con fines puramente didácticos, por lo que carece de la robustez necesaria para funcionar correctamente en un ambiente industrial. Su verdadero objetivo es demostrar la lógica que rige el diseño del CPU de un PLC típico y como se manejan las señales internamente.

Basándose en esta idea básica, es posible enumerar una serie de criterios tomados en cuenta a la hora de diseñar el hardware del Entrenador ASSI.

- Utilizar un microprocesador cuya programación sea, desde el punto de vista didáctico, accesible, es decir de fácil comprensión al usuario.
- Ya que se está diseñando un equipo para la UDB, se decidió utilizar el microprocesador que se utiliza en los equipos de laboratorios de dicha Universidad. Éste es el μ P 6502, que posee la capacidad suficiente para convertirse en el cerebro de un equipo destinado a controlar procesos industriales.
- Que su diseño permita un fácil reconocimiento de las diferentes etapas que conforman el hardware del CPU de un PLC típico.
- Que sus etapas de entrada salida sean capaces de demostrar como un PLC es diseñado para tratar las señales con lógica digital básica.
- Que el hardware permita el reconocimiento de las señales internas del PLC, especialmente las de reloj y selección de los diferentes dispositivos.

- Que su hardware sea representativo de un sistema, diseñado en base a un Mapa de Memoria, diseñado para PLC.

Tomando en cuenta estos criterios se diseñó el hardware del PLC denominado "Entrenador ASSI", a cuya descripción se dedica este capítulo.

A continuación se desarrollan las etapas principales y circuitos integrados utilizados para el diseño e implementación del PLC.

4.1 CIRCUITO DE CONTROL DE TECLADO Y VISUALIZADOR.

El circuito correspondiente se presenta en la figura 4.1

Como interfaz con el Usuario se utiliza un teclado hexadecimal (IC7), y un visualizador HDSP2111 (IC4).

Como se observa en la figura 4.2, el puerto A de la PIA1 (IC2) es compartido entre el visualizador y el teclado. Para el visualizador, el puerto A se convierte en la salida desde el sistema, por medio de la que se programará éste; y para el teclado, en entrada de datos hacia el sistema, por lo tanto es necesario habilitar un solo dispositivo a la vez. Para esto se utilizan dos buffer 74LS244; IC8 para el teclado e IC6 para la sección de control del visualizador. Se utiliza además un tercer 74LS244 (IC5) para habilitar la entrada de datos del visualizador. Los datos son transmitidos al visualizador a través del puerto B de la PIA1.

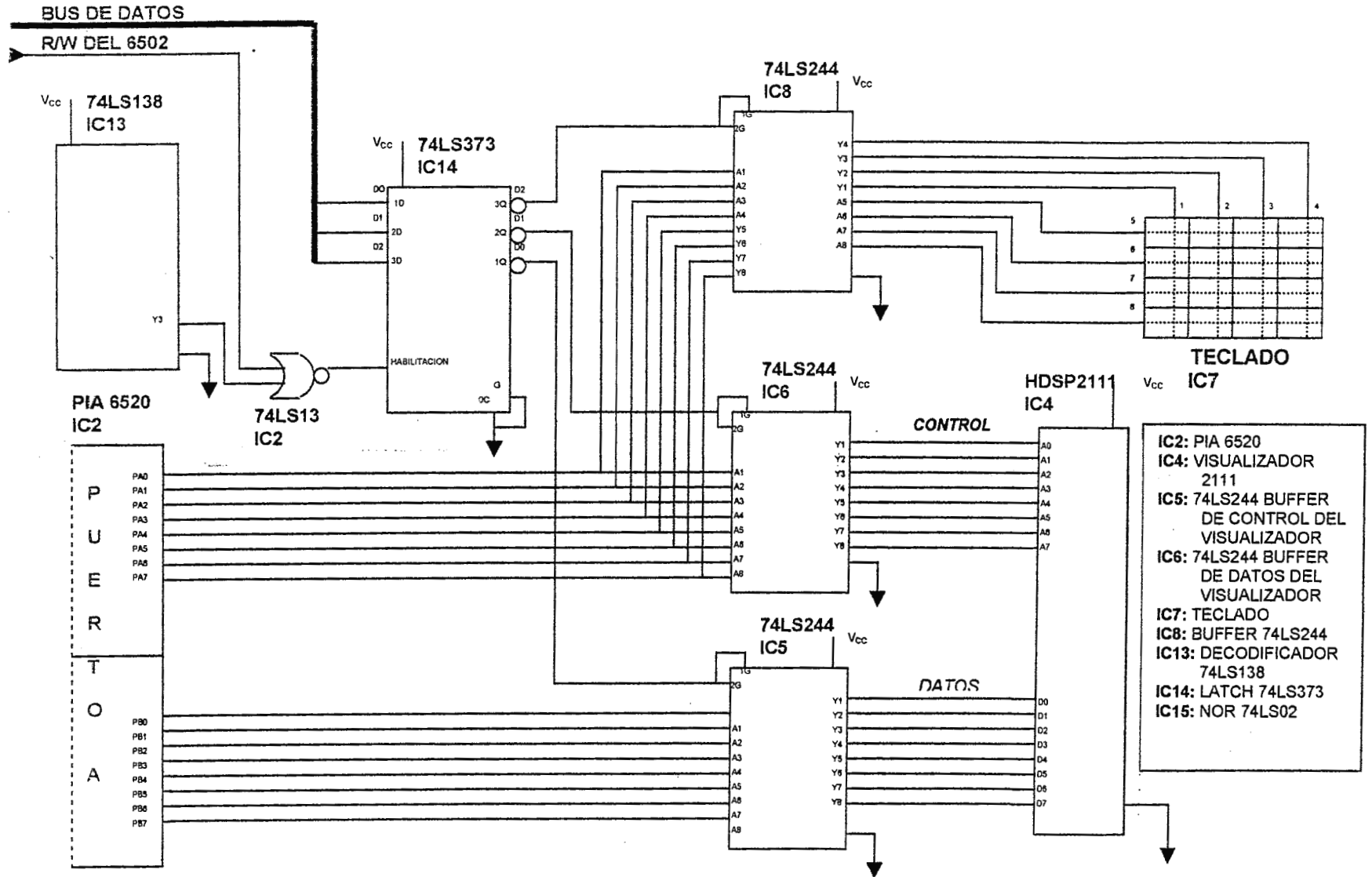


FIGURA 4.1 CONTROL DE TECLADO Y VISUALIZADOR

Los 3 buffer son controlados con un latch 74LS373 (IC14) que recibe las señales de escogitación de dispositivos desde el μ P6502 a través de las líneas D0, D1y D2 del bus de datos.

El latch posee la dirección 1004 en el mapa de memoria, con la que el μ P6502 accesa al área de interfaz con el usuario. Pero debido a que es por medio de esta circuiteria que se programará el μ P 6502, debe coincidir la habilitación de latch (IC14) con la señal R/W proveniente del microprocesador.

Tanto la señal de escritura como la señal de habilitación a la salida del decodificador (Y3) 74LS138 (IC13) son bajas activas, mientras que el pin CS del 74LS373 es alto activo, por lo que se requiere entonces un arreglo lógico que habilite con 1 al 74LS373 cuando R/W y Y3 sean 0, esto se logra con una compuerta NOR 74LS02 (IC15).

4.2 CIRCUITO DE CONTROL DEL AREA DE MEMORIA

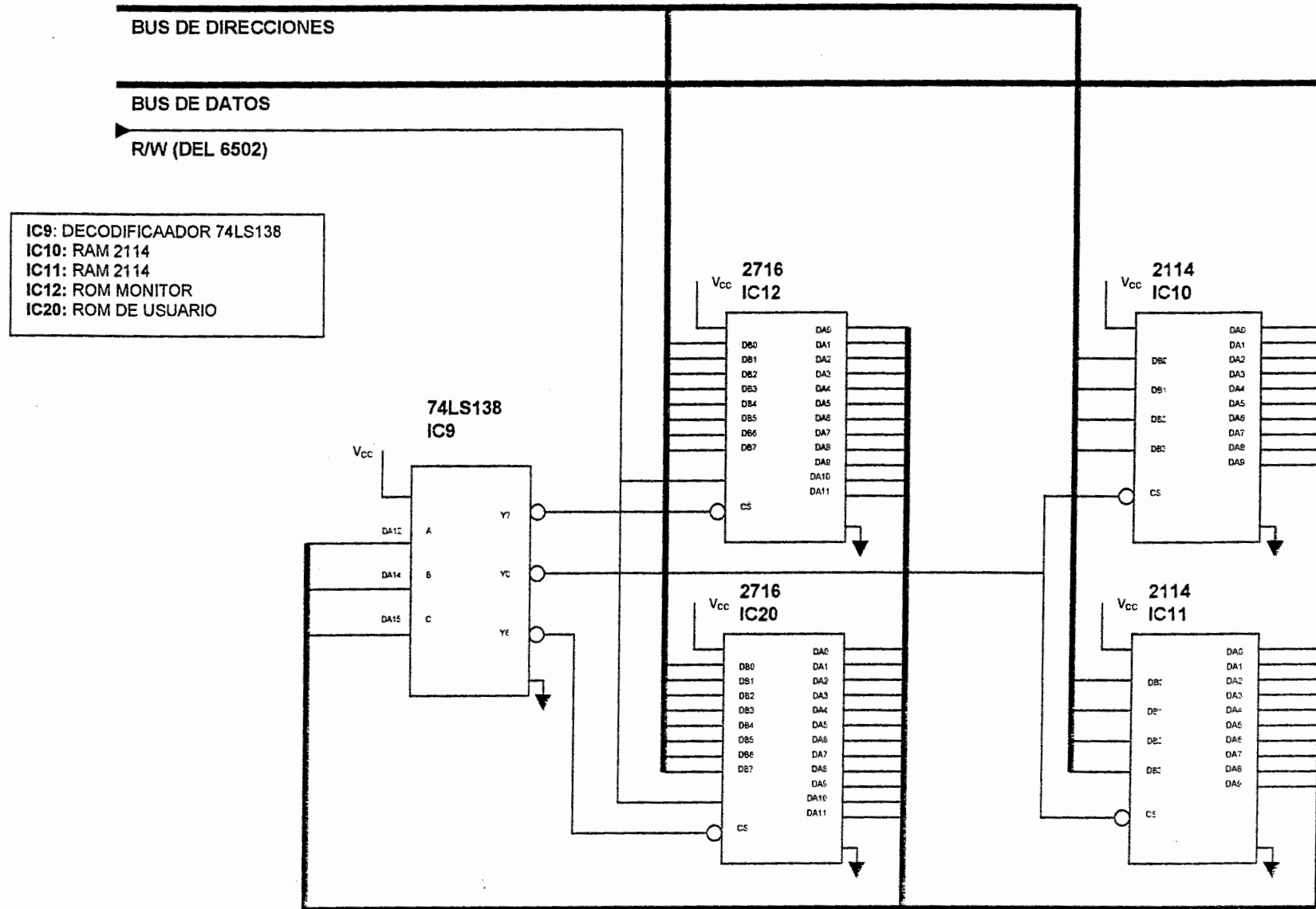
En la figura 4.2 se presenta el circuito correspondiente.

El ASSI utiliza un arreglo de memorias RAM (IC10 e IC11) denominado *banco A*, el cual ya ha sido explicado en la sección 3.5. La habilitación de éste se realiza a través de la línea Y0 del decodificador de direcciones 74LS138 (IC9).

Provenientes del mismo decodificador, se utilizan dos líneas para seleccionar las memorias ROM (Ver sección 3.5). La línea Y6 para la MROM (IC20) y la línea Y7 para la UROM (IC16).

El control de lectura/escritura en los diferentes dispositivos de memoria lo realiza directamente el μ P6502, con la línea R/W conectada a cada una de las memorias.

FIGURA 4.2 SELECCIÓN Y CONTROL DE MEMORIAS.



4.3 HABILITACION DE LOS PUERTOS DE ENTRADA/SALIDA DEL ENTRENADOR

El circuito correspondiente se presenta en la figura 4.3.

Las entradas/salidas del entrenador se encuentran en los puertos A y B de la PIA2 (IC3). Son en total 16 líneas que pueden programarse como entradas o como salidas dependiendo de los requerimientos del sistema que se debe controlar. La escritura y lectura de datos es controlada por el μ P6502 con la línea R/W. La habilitación de la PIA2 se realiza con un decodificador 74LS138 (IC9) con las direcciones 4000,4001,4002 y 4003 del mapa de memoria.

Los datos son transmitidos en sentido bidireccional entre la PIA2 y el microprocesador a través del bus de datos.

4.4 HABILITACION DE LA PIA1 DE COMUNICACIÓN CON EL VISUALIZADOR Y EL TECLADO.

El circuito correspondiente se presenta en la figura 4.3.

La recepción y generación de datos hacia el entrenador y desde el usuario, se encuentran en los puertos A y B de la PIA1 (IC2). La escritura y lectura de datos es controlada por el μ P6502 con la línea R/W. La habilitación de la PIA1 se realiza con un decodificador 74LS138 (IC13) con las direcciones 1000,1001,1002 y 1003 del mapa de memoria.

Los datos son transmitidos en sentido bidireccional entre la PIA 1 y el microprocesador a través del bus de datos.

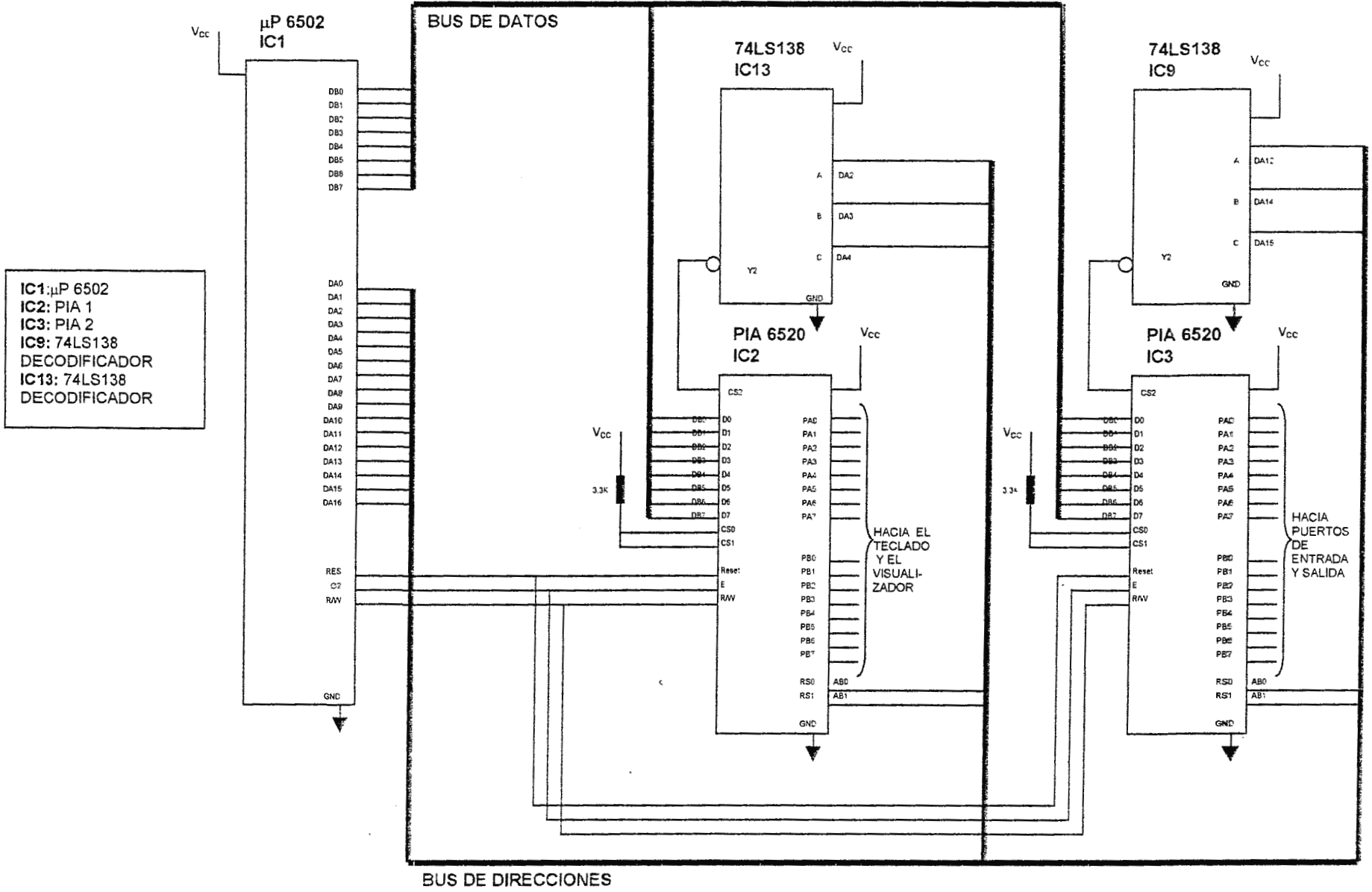


FIGURA 4.3 CONTROL DE PIA-1 Y PIA-2

4.5 CIRCUITOS DE TIEMPO

4.5.1 CIRCUITO DE GENERACIÓN DE LA SEÑAL DE RELOJ

En la figura 4.4 se presenta el circuito de generación de la señal de reloj del sistema, como un complemento a los diagramas presentados en esta sección, pero su funcionamiento se ha explicado en la sección 2 del capítulo 3.

4.5.2 EL CIRCUITO TEMPORIZADOR

Es el encargado de suministrar señales de reloj a dispositivos externos al entrenador ASSI. Estas señales pueden ser seleccionadas de forma independiente por medio de un programa, el cual envía los códigos necesarios para la elección de la salida.

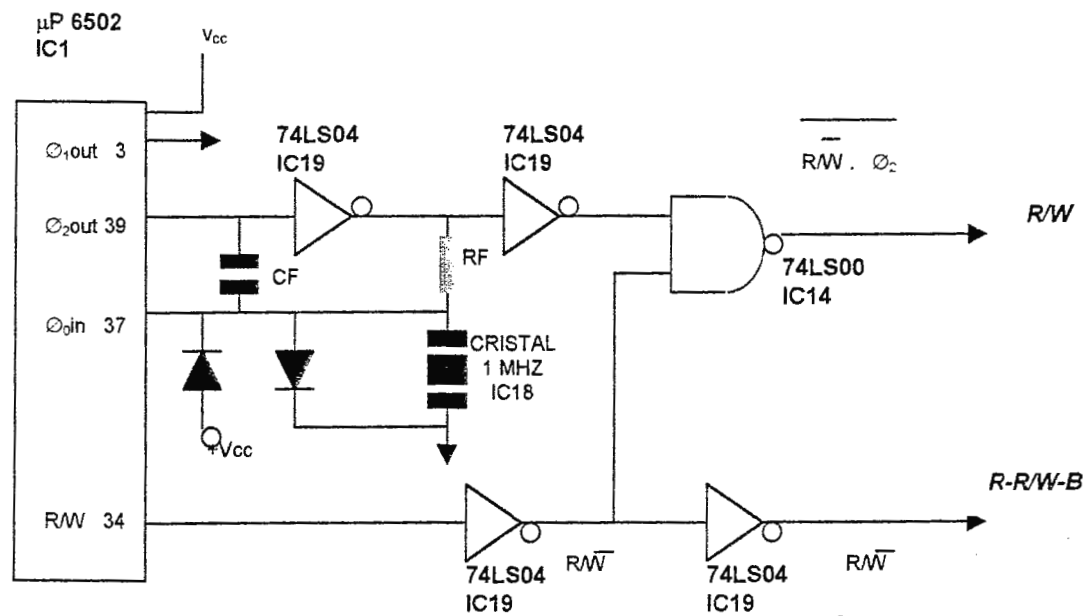
El objetivo es simular señales de reloj que normalmente se usan para el control de procesos industriales.

El circuito de la figura 4.5 muestra la técnica de cómo se seleccionan las distintas frecuencias de salida en el entrenador ASSI.

La frecuencia original o frecuencia primaria es generada con un circuito biestable, el cual ocupa un IC-555 configurado de forma astable. La frecuencia de salida (f_0) del 555 es aproximadamente de 10 KHz.

La frecuencia f_0 alimenta la entrada del reloj del contador binario 7493, la cual se activa con transiciones negativas. El 7493 es un contador binario de cuatro salidas que cuenta desde 0000 hasta 1111, es decir, de 0 a F en hexadecimal o de 0 a 15 en decimal.

Las salidas del 7493 se han enumerado de f_1 a f_4 , la figura 4.6 muestra el diagrama de tiempo de estas señales.



IC2: $\mu\text{P} 6502$
 IC18: CRISTAL DE 1M Hz.
 IC19: NOT 74LS04
 IC14: NAND 74LS00

FIGURA 4.4 RELOJ DEL SISTEMA

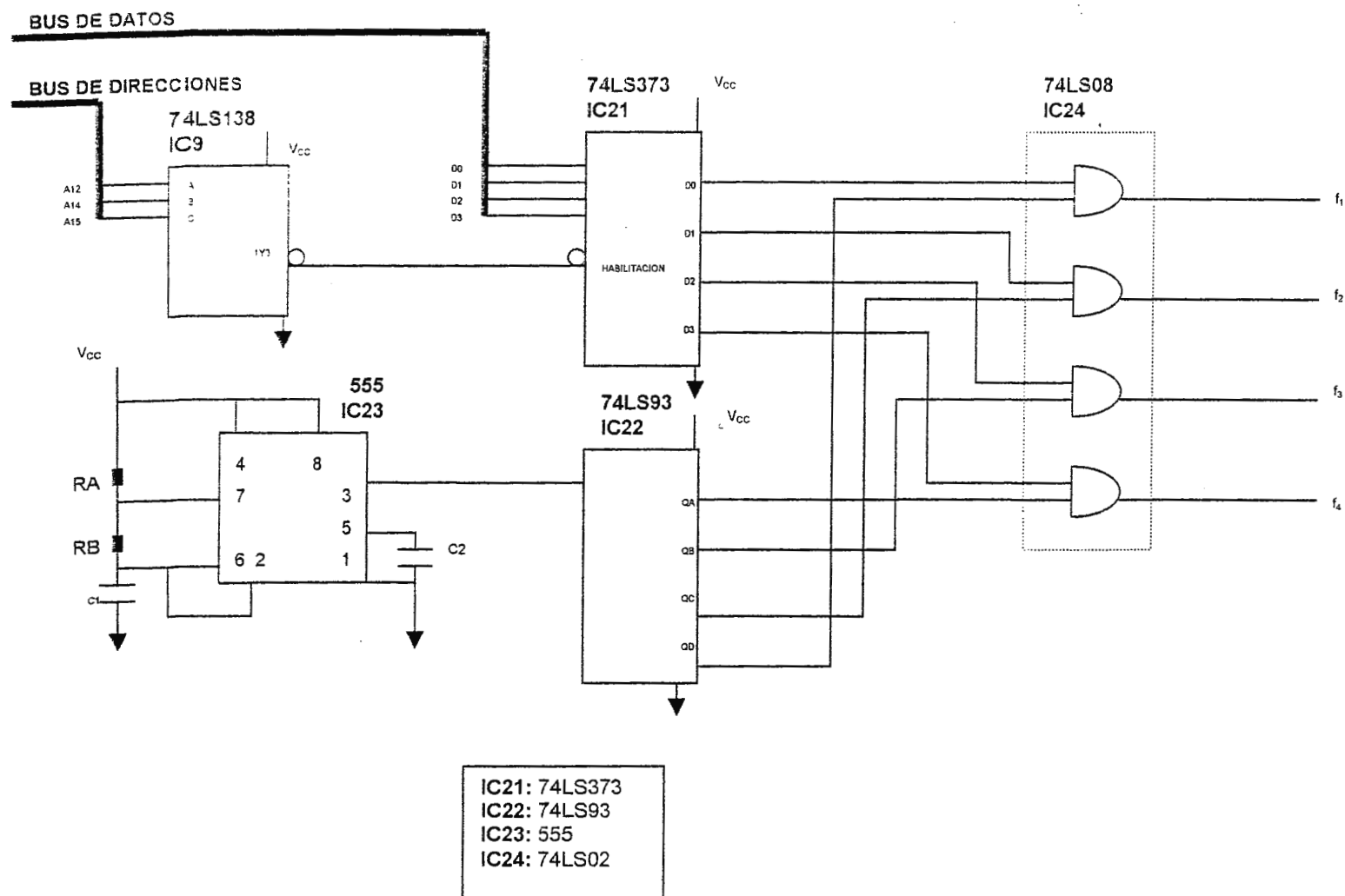


FIGURA 4.5 CIRCUITO TEMPORIZADOR

Como se puede observar cada señal es activada con el flanco negativo de la señal de entrada que lo alimenta, esto hace que la señal de entrada quede dividida por dos. De esto, si la señal f_0 es de 10 KHz, entonces:

$$f_1 = 5 \text{ KHz.}$$

$$f_2 = 2.5 \text{ KHz.}$$

$$f_3 = 1.25 \text{ KHz.}$$

$$f_4 = 0.625 \text{ KHz.}$$

Todas las salidas del 7493 pasan por una compuerta AND, la finalidad de esto es tener control sobre la salida de las señales de tiempo.

Las otras entradas de las AND'S son las salidas del latch 74373, cuya principal función es capturar los datos provenientes del programa para seleccionar de esta forma la frecuencia de salida deseada.

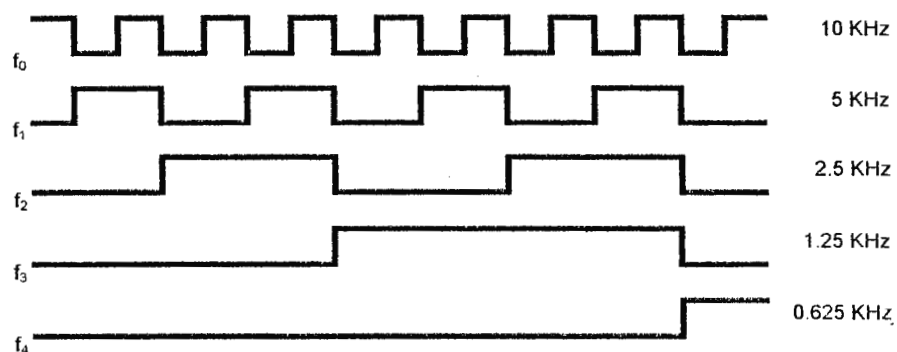


FIGURA 4.6. SEÑALES DEL TEMPORIZADOR

4.6 DECODIFICADORES

Los decodificadores, (ver sección 3.1), son los que habilitan cada sección o circuito integrado a través de una decodificación de líneas provenientes del bus de direcciones del μ P6502, y en base a las áreas de memoria especificadas en el mapa de memoria.

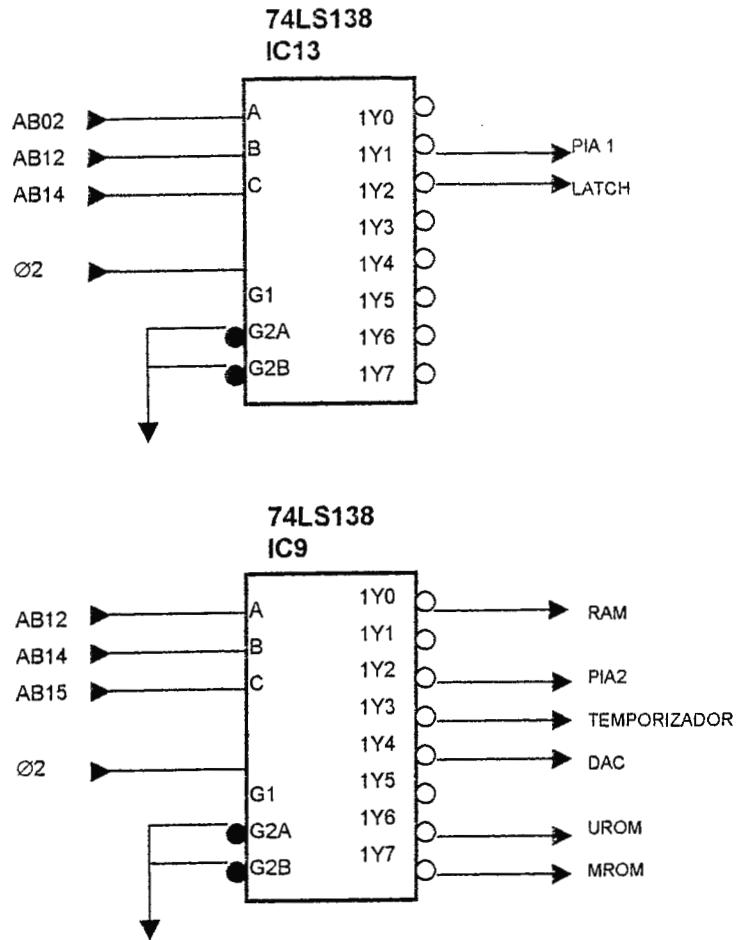
El circuito correspondiente se presenta en la figura 4.7

Los decodificadores utilizados, ya se han explicado ampliamente en secciones anteriores, por lo que ahora basta con recordar que se trata de dos 74LS138 de 3 a 8 líneas. Uno de ellos, IC9, utiliza AB12, AB14 y AB15 para seleccionar las siguientes áreas de memoria:

- RAM
- ROM
- PIA2
- Temporizador.
- DAC
- UROM
- MROM

Para las restantes áreas de memoria se utiliza otro 74LS138, IC13, que selecciona:

- PIA 1.
- Latch.



IC9:
DECODIFICAADOR
74LS138
IC13:
DECODIFICAADOR
74LS138

FIGURA 4.7. DECODIFICADORES DEL ASSI

4.7 EL CONVERTIDOR DIGITAL - ANALOGO

En la industria, el PLC debe ser capaz de controlar procesos análogos para esto es necesario que la información binaria que es procesada internamente se convierta a niveles analógicos.

En el entrenador ASSI, para la conversión de las salidas de datos digitales a valores análogos se utiliza un IC AD558; que es un convertidor DA de 8bits a niveles análogos entre 2.5 y 10 voltios.

El rango de salida depende de la configuración y de la alimentación que se le aplique al IC. En el entrenador ASSI se utiliza con una salida de 0 V - 2.56 V; debido a que la alimentación del sistema es de 5 V.

El circuito correspondiente es el mostrado en la figura 4.8. Los valores de la salida varían aproximadamente como se muestra en la tabla 4.1.

Binario	Hexa.	Análogo
0000 0000	00	0
0000 0001	01	0.010 V
0000 0010	02	0.020 V
0000 1111	0F	0.150 V
0001 0000	10	0.160 V
0111 1111	7F	1.270 V
1000 0000	80	1.280 V
1100 0000	C0	1.920 V
1111 1111	FF	2.550 V

TABLA 4.1

El DAC se encuentra ubicado en la memoria en las direcciones 8000 a la 8007 y se selecciona con la línea 1YA del 74LS138 (IC9). El registro correspondiente es el 15.

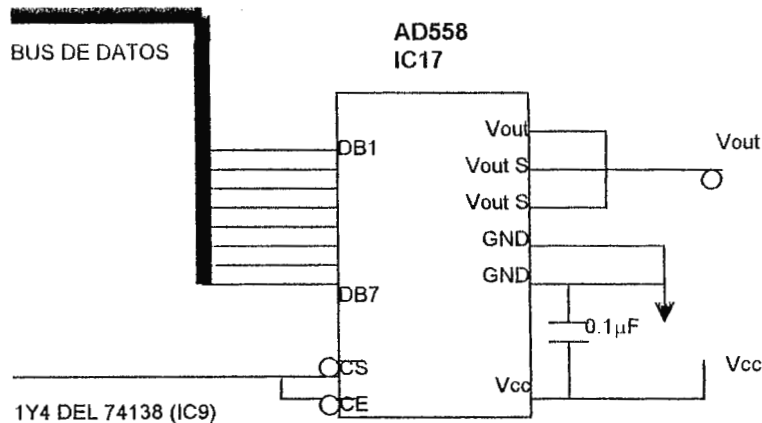


FIGURA 4.7. CONVERTIDOR DIGITAL-ANÁLOGO

4.8 LISTA DE CIRCUITOS INTEGRADOS DE ENTRENADOR ASSI.

Como complemento se presenta a continuación una lista de los integrados que se utilizaron para la implementación del entrenador ASSI.

IC1	µP 6502 – MICROPROCESADOR
IC2	6520 – PIA 1, CONTROL DE VISUALIZADOR Y TECLADO
IC3	6520 – PIA 2, CONTROL DE ENTRADAS / SALIDAS
IC4	HDSP2111 – VISUALIZADOR
IC5	74LS244 – BUFFER DE CONTROL DEL VISUALIZADOR
IC6	74LS244 – BUFFER DEL BUS DE DATOS DEL VISUALIZADOR
IC7	TECLADO HEXADECIMAL
IC8	74LS244 – BUFFER DEL TECLADO
IC9	74LS138 – DECODIFICADOR DE SELECCIÓN RAM, UROM, MROM, DAC, PIA 2

Y TEMPORIZADOR.

IC10	2114 - RAM DE DIRECCIONES BAJAS
IC11	2114 - RAM DE DIRECCIONES ALTAS
IC12	2716 – ROM DE PROGRAMA MONITOR
IC13	74LS138 – DECODIFICADOR DE SELECCIÓN DE PIA 1 Y LATCH
IC14	74LS373 – LATCH DE CONTROL DEL VISUALIZADOR Y EL TECLADO
IC15	74LS02 – NOR PARA HABILITACION DE IC14
IC16	74LS00 – NAND PARA CIRCUITO DE RELOJ
IC17	AD558 – CONVERTIDOR DIGITAL – ANALOGO
IC18	CRISTA DE 1 MHz.
IC19	74LS04 – NOT DEL CIRCUITO DE RELOJ
IC20	2716 – ROM DE USUARIO
IC21	74LS373
IC22	74LS93
IC23	555
IC24	NOR 74LS02

CAPITULO 5

PROGRAMA MONITOR

5.1 RUTINAS Y SUBRUTINAS DEL PROGRAMA MONITOR.

El funcionamiento del entrenador ASSI, al igual que el de cualquier sistema basado en microprocesador no depende solamente del correcto diseño y compatibilidad del hardware sino también del software que lo controla.

El software o programa que controla al sistema se denomina "Programa Monitor", y reside en una *Memoria de Lectura Solamente* (2716), denominada "ROM del Monitor" o "MROM", (IC12).

Es el programa monitor el que se encarga de iniciar el sistema, y el que posee todas las subrutinas necesarias para que éste funcione correctamente y cumpla los objetivos para los que ha sido construido.

El programa monitor puede dividirse en tres partes:

- Rutina Principal.
- Subrutinas.
- Tablas.

La Rutina Principal es la secuencia de instrucciones propias del microprocesador, necesarias para que las diferentes subrutinas interaccionen correctamente. Las *subrutinas* son programas que realizan tareas que se están utilizando todo el tiempo, y para el caso del ASSI, son subrutinas también las que realizan las funciones enfocadas al control de procesos industriales, que ofrece este entrenador. Se denomina *tablas* a las listas de

datos que están siendo consultadas una y otra vez, siguiendo un orden lógico; por diferentes subrutinas; su finalidad es ocupar menos espacio de memoria.

El programa monitor del Entrenador ASSI se ha estructurado de forma que el funcionamiento del mismo se base en cuatro funciones principales denominadas *procedimientos*, éstas se enuncian a continuación:

- **PROCEDIMIENTO INT.**
- **PROCEDIMIENTO EJE.**
- **PROCEDIMIENTO LER.**
- **PROCEDIMIENTO REG.**

Los procedimientos son conjuntos de subrutinas ordenadas de tal forma que ponen a disposición del usuario las cuatro funciones básicas que éste puede requerir del PLC:

- Introducir un programa.
- Ejecutar un programa.
- Leer un programa.
- Conocer los valores de las variables que serán o han sido manejadas.

Cada procedimiento es accedido por el programa monitor a través de códigos generados al seleccionar una tecla específica. Entendiendo por *código* el valor de la tecla que se convierte en el puntero que direcciona a la ubicación de memoria que realizará determinado procedimiento.

La figura 5.1 muestra el flujograma del programa principal o programa monitor.

En el flujograma pueden reconocerse cuatro etapas:

- La primera parte es para mostrar el rótulo de presentación de entrenador en el visualizador de la siguiente forma:

| - | - | A | S | S | I | - | - |

- La segunda parte revisa el teclado para detectar si existe una tecla activada. De no ser así, vuelve a ejecutar de nuevo el ciclo hasta que haya una tecla activada.
- La tercera parte selecciona el procedimiento correspondiente al código de la tecla presionada.
- Finalmente la cuarta parte ejecuta el procedimiento del programa en espera de otra orden.

DESCRIPCIÓN DE LOS PROCEDIMIENTOS

PROCEDIMIENTO INT: Sirve para introducir el programa en la memoria RAM.

PROCEDIMIENTO EJE: Sirve para ejecutar el programa que se encuentra almacenado en la memoria RAM del entrenador ASSI.

PROCEDIMIENTO LER: Sirve para verificar el programa que se va a ejecutar.

PROCEDIMIENTO REG: Lee el contenido de cualquier registro y lo presenta al visualizador.

Flujograma de Programa principal.

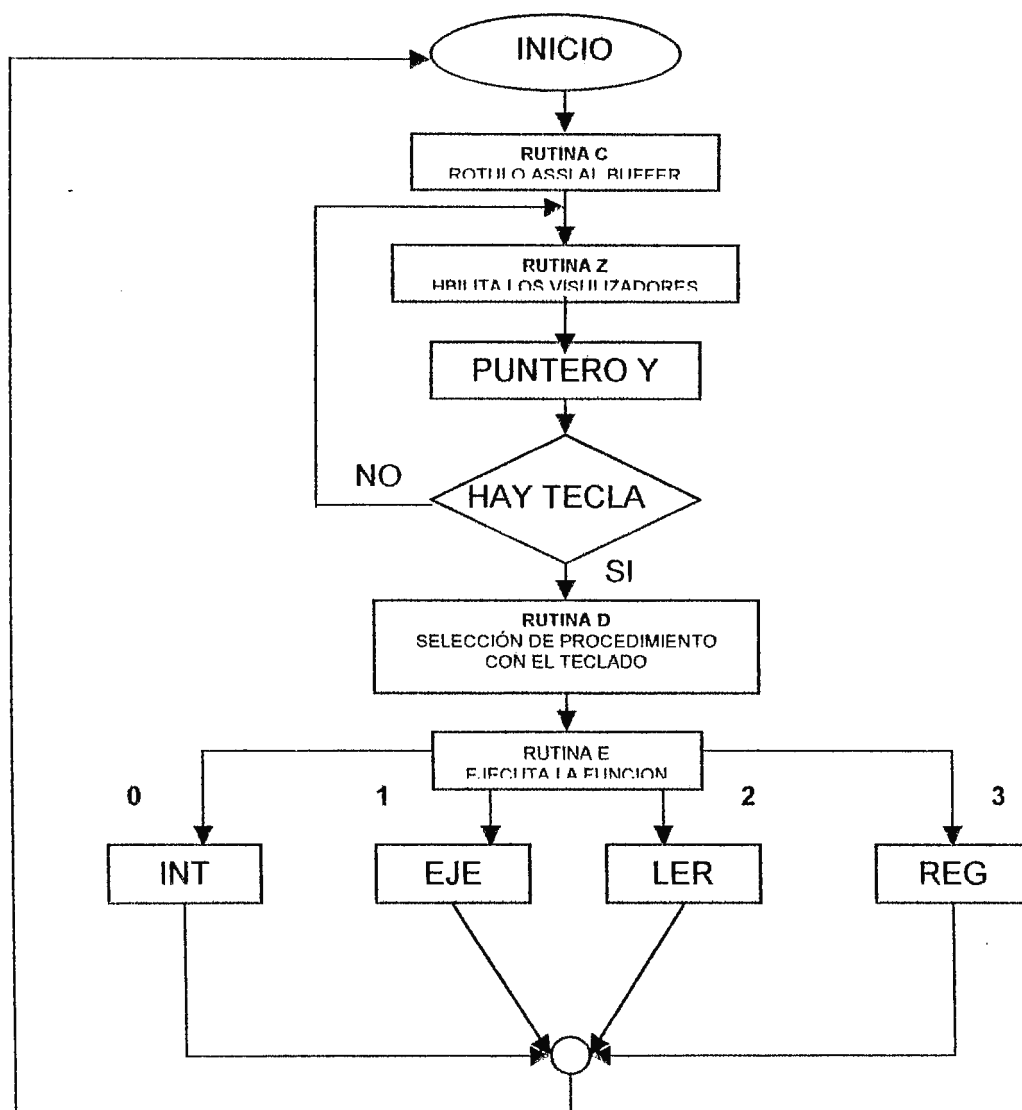


FIGURA 5.1. SUBRRUTINA DEL PROGRAMA MONITOR

5.1.1 PROCEDIMIENTO INT

Su función es editar el programa que se va a ejecutar, guardándolo en el buffer de memoria destinado para el almacenamiento del programa.

SELECCION:

Al iniciar el entrenador aparece el rótulo ASSI en el visualizador; para entrar al procedimiento INT debe presionarse la tecla "0", luego el programa monitor responde mostrando en el visualizador el rótulo:



Esto indica que se puede comenzar a editar el programa. Para introducir los datos se debe de tomar en cuenta el orden en que estos serán reconocidos por el programa monitor; el primer código será el OP-CODE y el segundo y el tercero los operandos.

Ej:



Debe tomarse en cuenta que INT solo reconocerá códigos que estén en modo decimal.

El flujograma del procedimiento INT se muestra en la figura 5.2.

El flujograma puede dividirse en dos etapas:

La primera etapa:

Se inicializan todas las constantes ocupadas por las subrutinas que son accesadas por este procedimiento.

La rutina F: es la encargada de mostrar el rótulo de presentación del procedimiento INT:

```
> > _ _ _ _ _
```

Además se encarga de grabar los datos de entrada en una región de memoria temporal llamada **Buffer1**. En **Buffer1** se almacena toda la información que se quiere que sea mostrada al visualizador, así el usuario puede ir observando los datos, que va introduciendo.

La segunda etapa:

Es la encargada de elegir la subfunción que se va a utilizar.

INT posee tres subfunciones:

1. Función Guardar: Se ejecuta con la **rutina M**, que es la encargada de almacenar los datos y ordenarlos en el buffer 3, para que puedan ser comprendidos por la subrutina que va a ejecutar el programa.

La rutina M es seleccionada presionando la tecla **B**.

2. Función Limpiar: Se ejecuta con la **rutina N**. Sirve para limpiar el visualizador, esta función se puede utilizar si en un dado caso hay un error de edición del programa que se está eligiendo. La rutina N es seleccionada presionando la tecla **A**.

3. Función Salir: Sirve para regresar a la rutina principal.

Esta función es accesada con la tecla **C**.

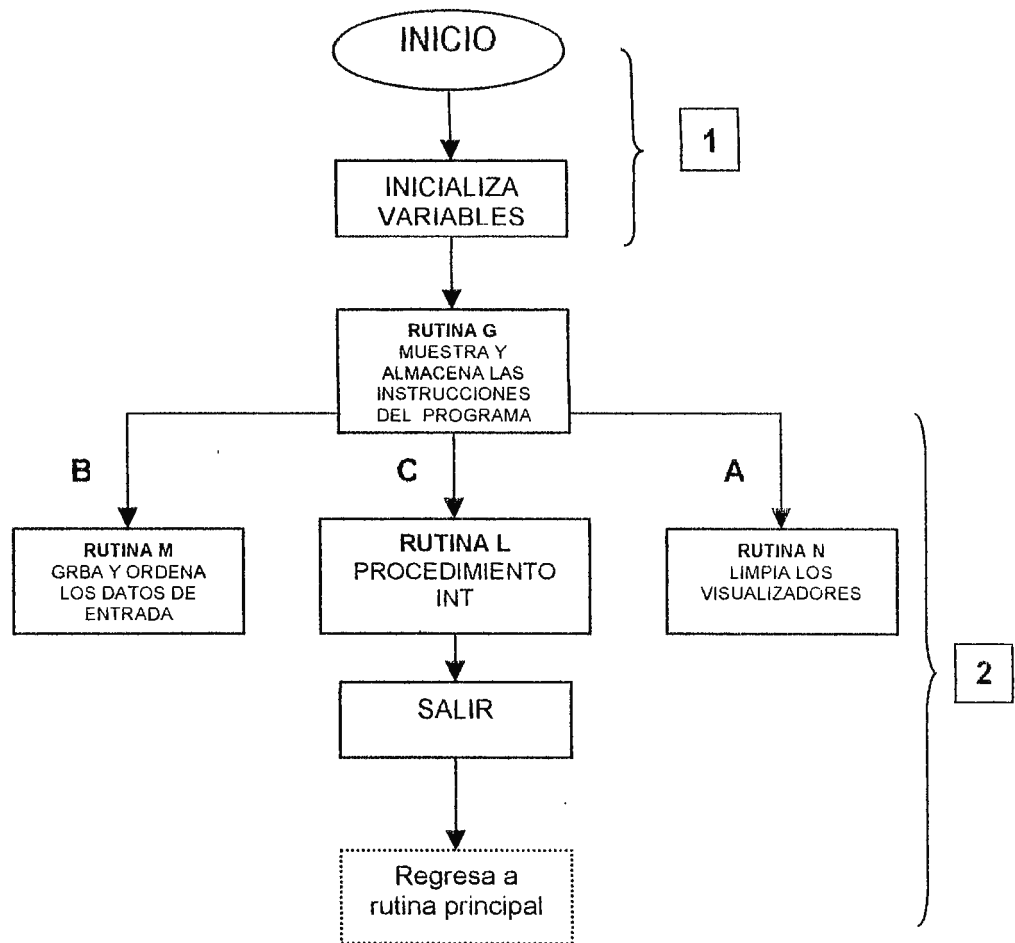


FIGURA 5.2. SUBROUTINA INT

5.1.2 PROCEDIMIENTO EJE

Su función es ejecutar el programa que se encuentra almacenado en la memoria. Este procedimiento se accesa con la tecla "1". En el visualizador aparece el siguiente mensaje:

P R O G R A M A

El procedimiento EJE decodifica todos los OP-CODE de cada instrucción que se encuentra en el buffer 3 uno por uno. Buffer 3 es región de memoria en que se encuentra el programa que se va a ejecutar. La ejecución del programa se detiene cuando llega a la instrucción **SAL**.

El flujograma se encuentra en la figura 5.3, y se divide en cuatro partes.

- La primera es donde se inicializan todas las constantes que van ha ser utilizadas por las subrutinas.
- La segunda parte es para mostrar el rótulo de identificación del procedimiento.
- La tercera parte monitorea el teclado para saber si la tecla **C** es presionada, de ser así, se ejecutará el programa.
- La tercera parte cuando la instrucción **SAL** es ejecutada el procedimiento **EJE** termina y se regresa al programa principal.

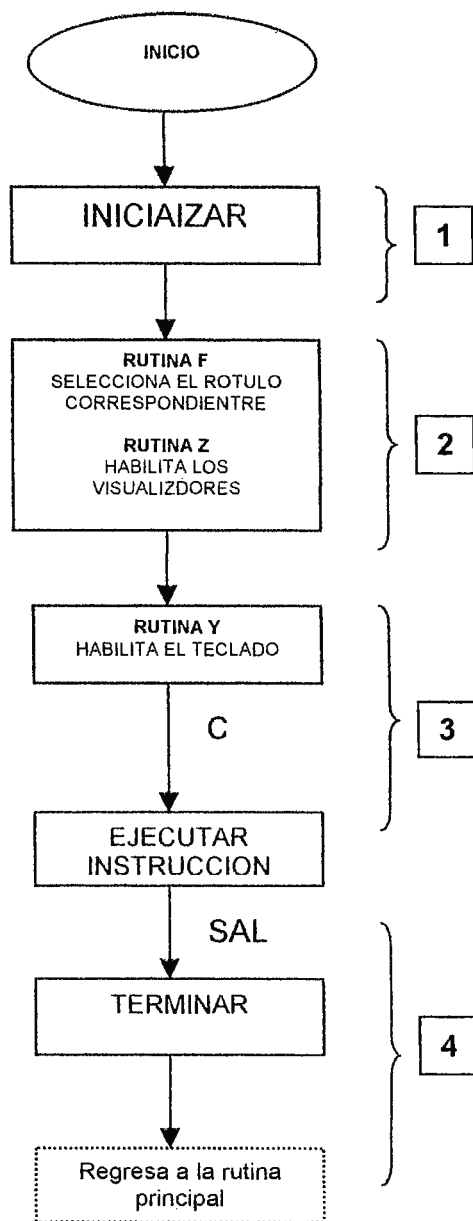


FIGURA 5.3 PROCEDIMIENTO EJE.

5.1.3 PROCEDIMIENTO LER

El procedimiento LER sirve para leer las líneas del programa que ya se encuentran grabadas en la memoria. También tiene la capacidad de corregir los errores ya que da acceso a "reeditar" los datos. El flujograma correspondiente se presenta en la figura 5.4.

El procedimiento LER es accesado desde el menú principal que se identifica con el rótulo **ASSI** con la **tecla 2**. Al presionar dicha tecla aparecerá en el visualizador el rotulo:

```

- - L E E R - -

```

La primera parte del programa es inicio de las constantes ocupadas por las subrutinas accesadas por LER.

En la segunda parte la rutina Q lee una línea del programa grabado y la rutina Z se encarga de mostrar en el display la línea accesada por la rutina Q.

Tercera parte: La rutina Y sensa el teclado por si hay alguna tecla presionada de los cuales solo reconocerá **A, B, C, #**.

- **TECLA A:** Le indica a la rutina Q que lea la siguiente línea del programa. Sirve como incremento para ir observando una por una las líneas.
- **TECLA B:** Le dice a la rutina Q que lea la línea anterior de programa, sirve como decremento.
- **TECLA C:** Es para editar una línea de programa. Llama al procedimiento INT para realizar esta tarea, una vez finalizada regresa hacia la rutina Q, para mostrar los cambios efectuados.
- **TECLA #,** sirve para salirse del procedimiento LER.

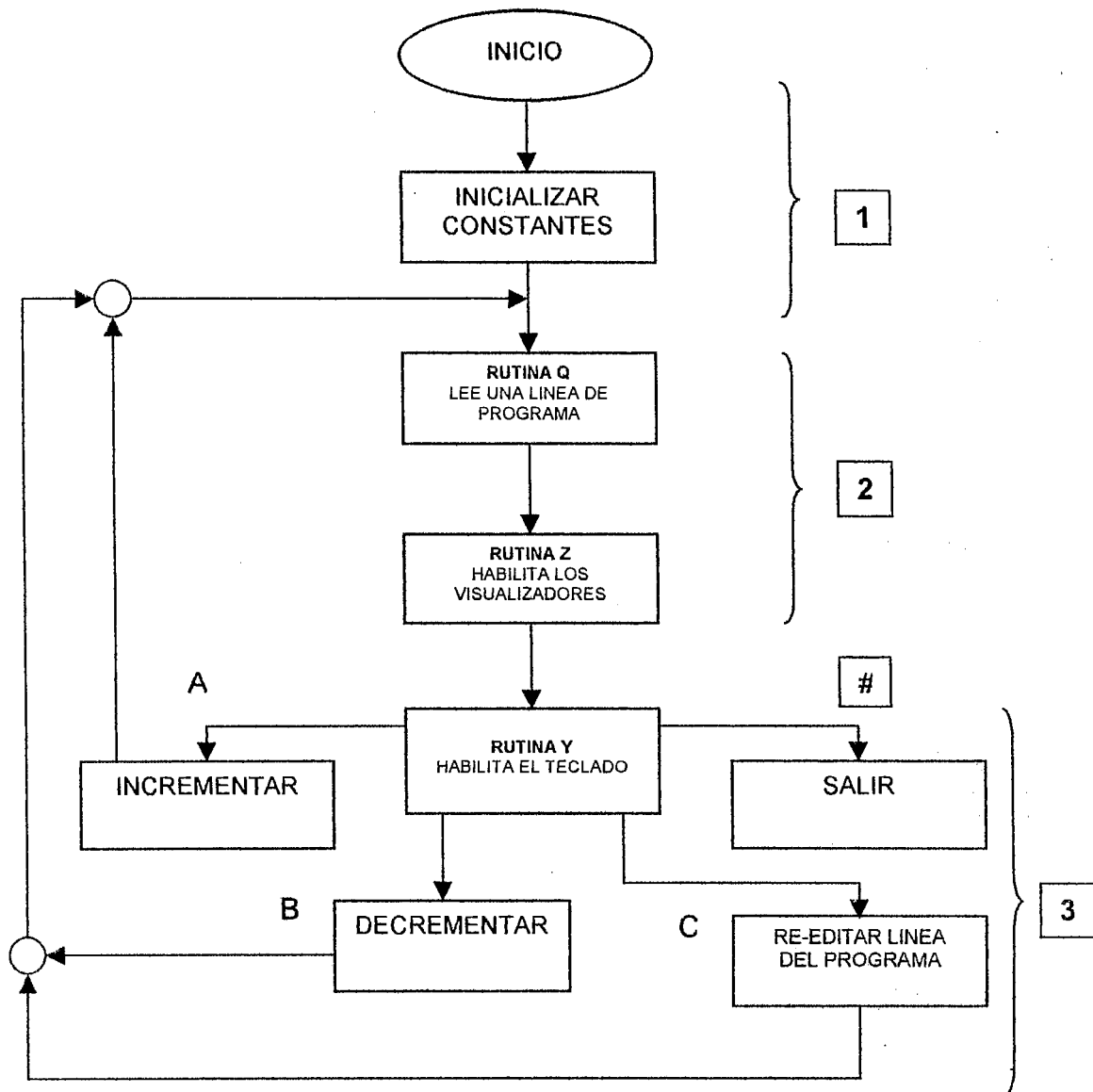


FIGURA 5.4 PROCEDIMIENTO LER

5.1.4 PROCEDIMIENTO REG

El Procedimiento REG se encarga de mostrar el contenido de un registro en el Visualizador. La rutina REG es accesada desde el menú principal por la tecla "3". El rótulo de presentación correspondiente es el siguiente:

```
N U R E G : _ _
```

En los dos últimos caracteres del visualizador se edita el registro cuyo contenido se desea conocer.

El flujograma correspondiente se muestra en la figura 5.5.

El flujograma se divide en tres partes:

- Primera Parte: Se inicializan las constantes.
- Segunda Parte: Se muestra el rótulo de presentación "NUREG__".
- Tercera Parte: Selecciona la función que se desea realizar. Las opciones son las siguientes:
 - **Tecla A:** Limpia los dos últimos caracteres de visualizador. Se aplica cuando el usuario comete errores al digitar el valor del registro a acceder.
 - **Tecla B:** Sirve para mostrar el contenido del registro. La tecla B llama a la rutina "Y", la cual se encarga de toda la operación.
 - **Tecla C:** Sirve para regresar a la rutina principal.

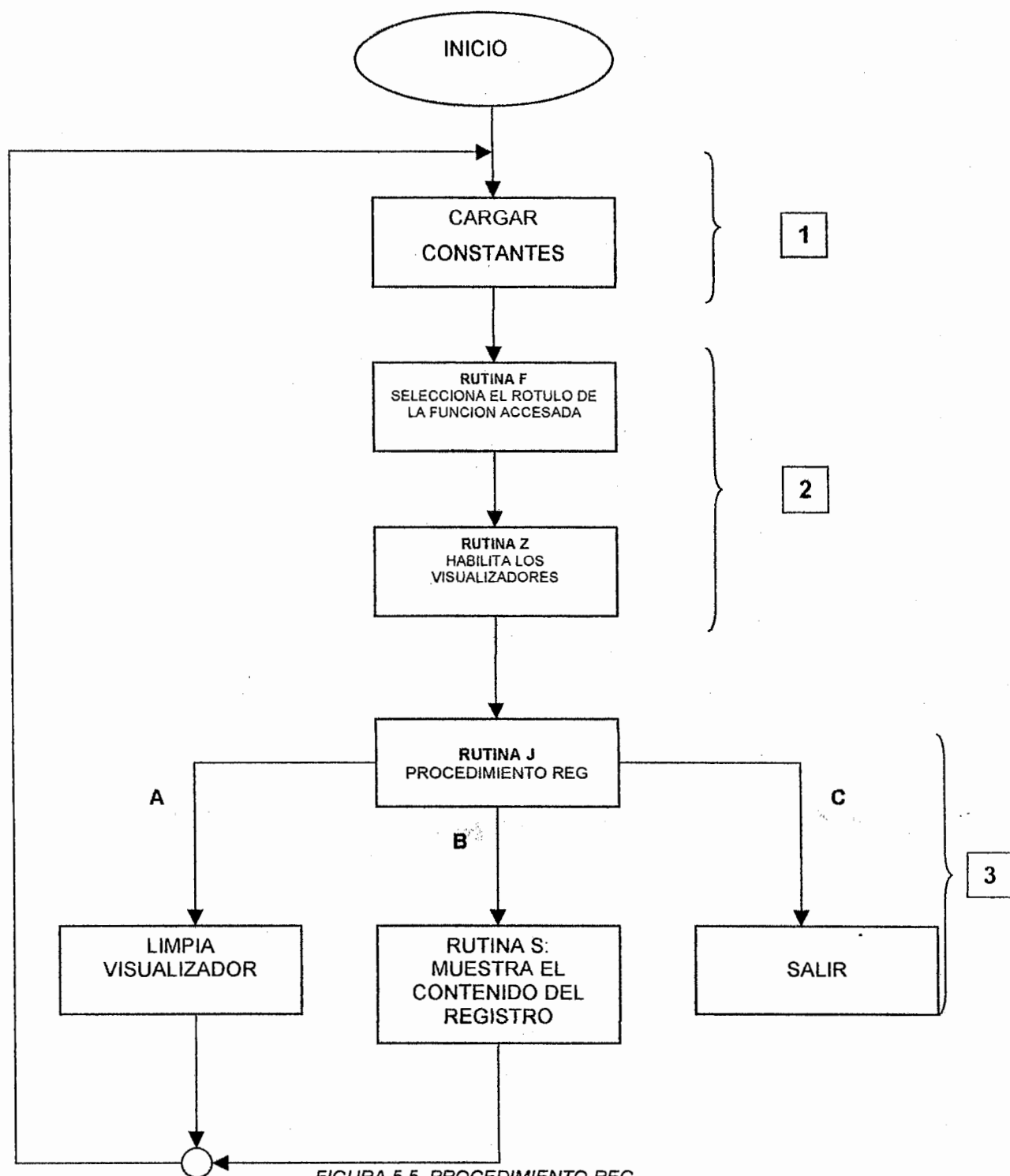


FIGURA 5.5. PROCEDIMIENTO REG.

5.2 EL LENGUAJE DE PROGRAMACION "ELL"

El programa monitor del entrenador ASSI es capaz de compilar el lenguaje de programación ELL (Ensamblado Línea por Línea). El ELL es propio del entrenador ASSI y fue creado bajo el ambiente del μ P6502.

El compilador del lenguaje ELL es una rutina que interpreta las instrucciones ELL y las ejecuta en código de máquina del μ P6502.

El lenguaje ELL posee 8 instrucciones que son capaces de realizar las funciones básicas de un PLC; éstas se presentan en la tabla 5.1.

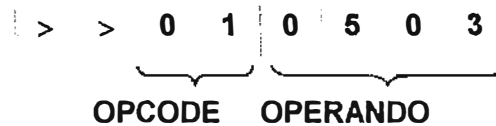
OP-CODE	INSTRUCCION	DESCRIPCION
00	SAL	TERMINA EL PROGRAMA SIEMPRE EN LA UTIMA INSTRUCCIÓN.
01	AND	OPERACIÓN AND ENTRE REGISTROS.
02	MOV	MUEVE EL CONTENIDO DE UN REGISTRO A OTRO.
03	LDR	CARGA UNA CONSTANTE A UN REGISTRO.
04	OR	OPERACIÓN OR ENTRE REGISTROS.
05	EXOR	OPERACIÓN EXOR ENTRE REGISTROS.
06	SUM	SUMA EL CONTENIDO DE UN REGISTRO CON OTRO.
07	RES	RESTA EL CONTENIDO DE UN REGISTRO CON OTRO.

TABLA 5.1 INSTRUCCIONES DEL ELL

Se incluyen 8 instrucciones, debido a que uno de los objetivos didácticos que se persigue es permitir al estudiante crear nuevas instrucciones, con el fin de demostrar

como son interpretadas las instrucciones por el compilador a partir del lenguaje de máquina de $\mu P6502$.

El formato de las instrucciones en ELL es el siguiente:



El OPCODE representa el código de la instrucción que se va a ejecutar. Este valor necesariamente debe estar en decimal.

El OPERANDO esta dividido en dos partes; el registro fuente y el registro destino. En algunas instrucciones el operando fuente será un valor predefinido como lo es una constante.

El registro fuente es soportado por las instrucciones en dos formas; como registro, o como constante predefinida.

➤ Como Registro:

La instrucción MOV puede mover el contenido de un registro a otro.

```
MOV (00)(02) 02 (00) (02)
SAL          00
```

El programa anterior se lee: "mueva el contenido del registro 02 al registro 00 y termine".

El registro fuente es copiado al contenido del registro destino. El contenido del registro fuente no es alterado por la operación.

➤ Como Constante:

Si se quiere cargar una constante a cualquier registro se utiliza la instrucción LDR.

EJ: LDR (03) (05) = 03 (03) (05)
 SAL 00

El programa anterior se lee: *cargue el registro 03 la constante predefinida 05, y termine.*

La interpretación lógica de la instrucción vista por el compilador para el ejemplo es la siguiente:

> > 0 1 0 5 0 3
 └───┬───┬───┬───┬───┬───┘
 AND REG.#5 REG.#3

Para este caso el resultado se guardará en el registro #5:

Si: - REGISTRO #5 = 22
 - REGISTRO #3 = 0F

Entonces:

REG#5 = (REG#5) AND (REG#3)

REGISTRO #5 = 02

En formato ELL el programa completo será:

1ª LINEA	01 03 05
2ª LINEA	00

Es necesario colocar la instrucción **SAL** al final de cada programa esto indicará al compilador que el programa ha terminado.

5.3 SIGNIFICADO DE BUFFERS Y REGISTROS

5.3.1 BUFFERS

Un buffer es una región de la memoria donde se pueden almacenar datos de forma temporal y pueden ser accedidos por cualquier subrutina que lo solicite. El programa monitor del entrenador ASSI maneja a los buffers con esa lógica. Los buffers pueden dividirse en:

- Buffers de propósito general: Son aquellos que pueden estar actuando indirectamente entre el programa y el usuario. Como ejemplo pueden mencionarse:
 1. Buffer1: Almacena los datos que serán mostrados en el visualizador.
 2. Buffer3: Almacena en forma ordenada los datos que serán introducidos por el usuario; o sea que almacena el programa que se va a ejecutar.
- Buffers de propósitos internos: Son ocupados por las subrutinas y los datos que en ellos se almacenan son el resultado de otras operaciones, que solo son reconocidas por el programa monitor.

5.3.2 REGISTROS

Un registro es un lugar dentro del entrenador ASSI donde se pueden escribir o leer datos. Un registro puede ser un puntero de entrada salida en esa región de memoria; o cualquier dispositivo que tenga asignada una dirección física dentro del mapa de memoria.

El programa monitor ha sido diseñado de forma que cualquier dispositivo conectado al sistema es tratado como un registro.

El programa monitor reconoce 16 registros enumerados de 0 al 15 correlativamente, en modo decimal. El registro 0, es la dirección física 0000; el registro 1 es la dirección 0001 y así sucesivamente hasta el registro número 9 cuya dirección es 0009. Sin embargo los seis registros restantes poseen otras direcciones.

El registro 10 posee la dirección física 5000 que es donde se encuentra el circuito temporizador, esto significa que si se escribe algo al registro se está escribiendo al temporizador.

La **Tabla 7 del programa monitor** presenta la dirección correcta para cada registro ocupado en el sistema. Esta **Tabla** se presenta en la tabla 5.2.

Los registros pueden clasificarse según la función que realizan:

- Registros de lectura y escritura.
- Registros de solo escritura.

La tabla 5.2, muestra los registros separados según sus características de escritura-lectura.

TABLA 7 DEL PROGRAMA MONITOR

	UBICACION		REGISTRO			FUNCION
F674	00	00	00	Memoria	L-E	
F676	01	00	01	Memoria	L-E	
F678	02	00	02	Memoria	L-E	
F67A	03	00	03	Memoria	L-E	
F67C	04	00	04	Memoria	L-E	
F67F	05	00	05	Memoria	L-E	
F680	06	00	06	Memoria	L-E	
F682	07	00	07	Memoria	L-E	
F684	08	00	08	Memoria	L-E	
F686	09	00	09	Memoria	L-E	
F688	00	20	10	ORA (PIA1)	L-E	
F68A	01	20	11	CRA (PIA1)	E	
F68C	02	20	12	ORB (PIA1)	L-E	
F68F	03	20	13	CRB (PIA1)	E	
F690	04	40	14	Temporizador	E	
F692	05	30	15	DAC	E	

TABLA 5.2. TABLA 7 DEL PROGRAMA MONITOR

5.4 PROGRAMA MONITOR

Para completar la información del entrenador ASSI se presenta a continuación el programa monitor completo. Tener acceso a éste permitirá a los usuarios estudiar la lógica con la que se ha diseñado e PLC ASSI, y agregar así las funciones que sean necesarias.

RUTINA A: Visualiza el contenido del buffer de visualizador.

F000	A9	00		LDA	#00
F002	8D	01	08	STA	\$CRA
F005	8D	03	08	STA	\$CRB
F008	A9	FF		LDA	#FF
F00A	8D	00	08	STA	#DDRA
F00D	8D	02	08	STA	3DDRB
F010	A9	04		LDA	#04
F012	8D	01	08	STA	\$CRA
F015	8D	03	08	STA	\$CRB
F018	8A			TXA	
F019	48			PHA	
F01A	A2	08		LDX	#08
F01C	A9	02		LDA	#02
F01F	8D	02	08	STA	\$ORB
F021	A9	B0		LDA	#B0
F023	8D	00	08	STA	\$ORA
F026	BD	00	F5	A: LDA	# (TAB1 + X)
F029	8D	00	08	STA	\$ORB
F02C	B5	30		LDA	# (BUF1 + X)
F02F	8D	02	08	STA	\$ORB
F031	CA			DFX	
F032	D0		F3	BNF	A:
F034	48			PLA	
F035	AA			TAX	
F036	60			RTS	

RUTINA B: Sensor Teclado.

F037	98			TYA	
F038	48			PHA	
F039	A9	00		LDA	#00

F03B	8D	01	08	STA	\$CRA
F03F	A9	0F		LDA	#0F
F040	8D	00	08	STA	\$DDR
F043	A9	04		LDA	#04
F045	8D	01	08	STA	\$ORA
F048	A0	00		LDY	#00
F04A	A9	FF		LDA	#FF
F04C	4A			B3: LSR,A	
F04D	09	F0		ORA	#F0
F04F	48			PHA	
F050	8D	00	08	STA	\$ORA
F053	AD	00	08	LDA	\$ORA
F056	29	F0		AND	#F0
F058	49	F0		FOR	#F0
F05A	D0	0F		BNF	B1:
F05C	68			PLA	
F05D	C8			INY	
F05F	C8			INY	
F05F	C8			INY	
F060	C8			INY	
F061	C0	10		CPY	#10
F063	B0	03		BCS	B2:
F065	4C	4C	F0	JMP	B3:
F068	18			B2: CLC	
F069	60			RTS	
F06A	85	11		B1: STA,0	\$11
F06C	29	70		AND	#70
F06F	F0	0F		BFQ	B4:
F070	C8			INY	
F071	A5	11		LDA,0	#11
F073	29	B0		AND	#B0
F075	F0	08		BFQ	B4:
F077	C8			INY	
F078	A5	11		LDA,0	\$11
F07A	29	00		AND	\$D0
F07C	F0	01		BFQ	B4:
F07F	C8			INY	
F07F	B9	09	F5	B4: LDA,Y	\$TABLA2
F082	38			SFC	
F083	85	10		STA,0	\$TFCLA
F085	68			PLA	
F086	68			PA	
F087	A8			TAY	
F088	60			RTS	

RUTINA C. Transfiere el rotulo ASSI al buffer del display

F08B	98				TYA
F08C	48				PHA
F08D	A0	08			LDY #08
F08F	B9	0F	FG		D1 : LDA, Y \$ TABLA 2
F092	99	2F	00		STA, Y \$ BRFFER1
F095	88				DEY
F096	D0	F7			BNE D1:
F098	68				PLA
F099	A8				TAY
F09A	60				RTS

RUTINA D. Selecciona la dirección de inicio del procedimiento escogido con el teclado.

F09B	8A				TXA
F09C	48				PHA
F09D	A6	A0			LDX,0 #tecla
F09F	A9	F6			LDA #F6
F0A1	85	68			STA,0 \$ROT_ALT
F0A3	BD	18			LDA,X \$TABLA4
F0A6	85	67			STA,0 \$ROT_BAJ
F0A8	68				PA
F0A9	AA				TAX
F0AA	60				RTS

RUTINA E: Ejecuta la función solicitada.

F0AB	EA				PHA
F0AC	EA				TXA
F0AD	EA				PHA
F0AE	EA				TYA
F0AF	EA				PHA
F0B0	6C	67	00		(JMP) \$0067
F0B3	EA				PLA
F0B4	EA				TAY
F0B5	EA				PLA
F0B6	EA				TAX
F0B7	EA				PLA
F0BB	60				RTS

RUTINA F. Coloca los rótulos de la función accesada en el buffer de Display.

F0B9	48			PHA
F0BA	8A			TXA
F0BB	48			PHA
F0BC	98			TYA
F0BD	48			PHA
F0BE	A0	00		LDY #00
F0C0	A6	A0		LDX,0 \$10
F0C2	F0	0D		BEQ A:
F0C4	A0	08		LDY #08
F0C6	CA			DEX
F0C7	F0	08		BEQ A:
F0C9	98			TYA
F0CA	18			CLLC
F0CB	69	08		ADC #08
F0CD	A8			TAY
F0CE	4C	C6	F0	JMP B:
F0D1	A2	08		LDX #08
F0D3	B1	A2		(LDA), Y \$A2
F0D5	9D	2F	00	STA, Y \$BUFFER 1 - 1
F0D8	CA			DEX
F0D9	F0	04		BEQ C:
F0DB	C8			INY
F0DC	4C	D3	F0	JMP D:
F0DF	68			PLA
F0F0	68			TAY
F0E1	68			PLA
F0E2	AA			TXA
F0E3	68			PLA
F0E4	60			RTS

RUTINA Z. Habilita los visualizadores.

F0E5	A9	04		LDA #04
F0E7	8D	04	10	STA \$LATCH
F0EA	20	00	F0	JSR \$RUTINA A
F0ED	60			RTS

RUTINA Y. Habilita el teclado.

F0EE	A9	03		LDA #04
F0F0	8D	04	10	STA \$LATCH
F0F3	20	37	10	JSR \$RUTINA B
F0F6	60			RTS

RUTINA G. Muestra y almacena las instrucciones del programa a salvar.

F100	EA			NOP
F101	EA			NOP
F102	EA			NOP
F103	A2	00		LDA #01
F105	A9	01		STA,0 \$14
F107	85	A7		LDX #00
F109	20	B9	E0	JSR \$RUTINA F
F10C	20	E5	E0	G4: JSR \$RUTINA Z
F10F	20	EE	E0	G2: JSR \$RUTINA Y
F112	B0	0A		BCS G1:
F114	A9	00		LDA #00
F116	85	A7		STA \$14
F118	EA	EA	EA	JSR \$RUTINA X
F11B	AC	0F	E1	JMP G2
F11E	A5	A7		G1: LDA,0 \$14
F120	D0	ED		BNE G2
F122	A9	01		LDA #01
F124	85	A7		STA \$A7
F126	A5	A0		LDA,0 \$A0
F128	C9	0A		CMP,A #0A
F12A	B0	0E		BCS G3:
F12C	18			CLC
F12D	69	30		ADC,0 \$30
F12F	9D	32	00	STA (BUFFER1+4) =X
F132	E8			INX
F133	E0	06		CPX #06
F135	D0	D5		D0 64:
F137	0	E5	E0	JSR \$RUTINA Z
F13A	EA			G3: PLA
F13B	EA			TXA
F13C	EA			PLA
F13D	60			RTS

RUTINA H. Comprime el Buffer 1 y lo coloca en el Buffer 2.

F13E	48			PHA
F13F	8A			TXA
F140	48			PHA
F141	98			TYA
F142	48			PHA
F143	A2	00		LDX #00
F145	A0	00		LDY #00
F147	B9	32	00	HI: LDA,Y (\$BUFFER1+2)+Y
F14A	29	0F		AND #0F

F14C	0A			ASL
F14D	0A			ASL
F14E	0A			ASL
F14F	0A			ASL
F150	85	25		STA,0 \$25
F152	C8			INY
F153	B9	32	00	LDA,Y (\$BUFFER1+2)+Y
F156	29	0F		AND #0F
F158	65	25		ADC,0 \$25
F15A	95	60		STA,X (\$BUFFER2)+X
F15C	E8			INX
F15D	C8			INY
F15E	E0	03		CPX #03
F160	D0	E5		BNE HI:
F162	68			PLA
F163	A8			TAY
F164	68			PLA
F165	AA			TAY
F166	68			PLA
F167	60			RTS

RUTINA I. Ordena el programa.

F168	48			PHA
F169	8 ^a			TXA
F16A	48			PHA
F16B	98			TYA
F16C	48			PHA
F16D	A5	20		LDA,0 \$20
F16F	A2	02		LDX #02
F171	18			I2: CLC
F172	65	20		ADC,0 \$20
F174	90	02		BCC I1:
F176	E6	65		INC,0 \$65
F178	CA			I1: DEX
F179	B0	F6		BNE I2:
F17B	A8			TAY
F17C	A2	00		LDX #00
F17E	B5	60		I3: (LDA,X) \$BUFFER
F180	91	63		2 (STA,Y) \$1200
F182	C8			INY
F183	E8			INX
F184	E0	03		CPX #03
F186	D0	F6		BNE I3:
F188	68			PLA
F189	A8			TAY

F18A	68			PLA
F18B	AA			TXA
F18C	68			PLA
F18D	60			RTS

RUTINA K. Prepara la instrucción para ser ejecutada en las direcciones \$0070 - \$0073.

F18E	48			PHA
F18F	8A			TXA
F190	48			PHA
F191	88			TYA
F192	48			PHA
F193	A9	00		LDA #00
F195	85	2A		STA,0 \$2A
F197	BD	00	02	K2: LDA,X \$1200
F19A	0A			ASL
F19B	A8			TAY
F19C	A9	02		LDA #02
F19E	85	2B		STA,0 \$02
F1A0	B9	74	F6	K1: LDA,Y \$1350
F1A3	84	2C		STY,0 \$temp3
F1A5	A4	2C		LDY,0 \$2A
F1A7	99	70	00	STA,Y \$0070
F1AA	C8			INY
F1AB	84	2A		STY,0 \$2A
F1AD	A4	2C		LDY,0 \$2C
F1AF	C8			INY
F1B0	C6	2B		DEC,0 \$2B
F1B2	D0	EC		BNE K1:
F1B4	E8			INX
F1B5	A5	2A		LDA,0 \$2A
F1B7	C5	2D		CPM #04
F1B9	D0	DC		BNE K2:
F1BB	68			PLA
F1BC	A8			TAY
F1BD	68			PLA
F1BD	AA			TAX
F1BF	68			PLA
F1C0	60			RTS

RUTINA L. Procedimiento INT.

F1C1	A9	00		LDA #00
F1C3	85	20		STA,0 20
F1C5	20	46	F2	A4: JSR RUTINA O:
F1C8	EA			NOP

F1C9	20	00	F1	JSR RUTINA G:
F1CC	A5	A0		LDA,0 \$A0
F1CE	C9	0A		CMP #0A
F1D0	B0	08		BCS A1:
F1D2	20	EE	F0	A3: JSR RUTINA Y:
F1D5	B0	03		BCS A1:
F1D7	4C	D2	F1	JMP A3
F1DA	A5	A0		A1: LDA,0 \$A0
F1DC	C9	0A		CMP #0A
F1DE	D0	06		BNE A2:
F1E0	20	62	F2	JSR RUTINA N:
F1E3	4C	C9	F1	JMP A4:
F1E6	C9	0B		A2: CMP #0B
F1E8	D0	06		BNE A5
F1EA	20	59	F2	JSR RUTINA M:
F1ED	4C	C9	F1	JMP A4:
F1F0	C9	0C		A5: CMP #0C
F1F2	D0	06		BNE A6:
F1F4	68			PLA
F1F5	68			PLA
F1F6	EA			NOP
F1F7	4C	B3	F0	JMP \$02B3
F1FA	AC	D2	F1	A6: JMP A3:

RUTINA Q. Lee una línea del programa.

F200	48			PHA
F201	8A			TXA
F202	48			PHA
F203	98			TYA
F204	48			PHA
F205	A5	20		LDA,0 \$20
F207	A2	02		LDX #02
F209	18			Q2: CLC
F20A	65	20		ADC,0 \$20
F20C	90	02		BCC Q1:
F20E	EA	8A		INC,0 \$8 ^a
F210	CA			Q1: DEX
F211	D0	F6		BNE Q2:
F213	A8			TAY
F214	EA			NOP
F215	A2	00		LDX #00
F217	B1	89		Q4: (LDA,Y) \$89
F219	48			PHA
F21A	4A			LSR
F21B	4A			LSR
F21C	4A			LSR

F21D	4A		LSR
F21E	18		CLC
F21F	69	30	ADC #30
F221	95	32	STA,X \$32
F223	E8		INX
F224	68		PLA
F225	29	0F	AND #0F
F227	18		CLC
F228	69	30	ADC #30
F22A	95	32	STA,X \$32
F22C	C8		INX
F22D	D0	02	BNE Q3
F22F	E6	8A	INC,0 \$8A
F231	E8	E8	Q3: INX
F232	E0	06	CPX #06
F234	D0	E1	BNE Q4
F236	68		PLA
F237	A8		TYA
F238	68		PLA
F239	AA		TXA
F23A	68		PLA
F23B	60		RTS

RUTINA E1. Utilizada por E, dentro del procedimiento "LER".

F23F	A5	20	LDA,0 \$20
F241	F0	02	BEQ A:
F243	C6	20	DEC,0 \$20
F245	60		A: RTS

RUTINA O. Inicializa circuitos para "INT".

F246	A9	34	LDA #34
F248	85	A2	STA,0 \$C2
F24A	A9	E6	LDA #03
F24C	85	A3	STA,0 \$C3
F24E	A9	00	LDA #00
F250	EA	EA	
F252	85	63	STA,0 \$63
F254	A9	02	LDA #12
F256	85	64	STA,0 \$64
F258	60		RTS

RUTINA M. Graba y ordena los datos de entrada para INT.

F258	20	3E	F1	JSR \$RUTINA M:
F25C	20	68	F1	JSR \$RUTINA i:
F25F	E6	20		INC,0 \$20
F261	EA			RTS

RUTINA N. Limpia Display para INT.

F262	A9	00		LDA #00
F264	85	A0		STA,0 \$10
F266	60			RTS

RUTINA Ñ. Sale de procedimiento INT.

F267	A2	FF		LDX #FF
F269	9A			TXS
F26A	4C	84	F2	JMP INICIO

RUTINA P. RUTINA PRINCIPAL

F26D	20	88	F0	JSR RUT C
F270	20	E5	F0	JSR RUT Z
F273	20	EE	F0	A: JSR RUT Y
F276	90	FB		BCC A1:
F278	A5	A0		LDA ,0 \$A
F27A	C9	04		CMP #04
F27C	B0	F5		BCS A1:
F27E	20	9B	F0	JSR RUT D
F281	20	AB	F0	JSR RUT E
F284	4C	6D	F0	JSR INICIO

RUTINA X. Retrasa tiempo de lectura de la tecla.

F287	85	90		
F289	8A			
F28A	48			
F28B	A2	B0		
F28D	A5	90		
F28F	8D	00	10	
F292	AD	00	10	
F295	85	91		
F297	CA			
F298	D0	F3		

F29A	68	
F29B	AA	
F29C	A5	91
F29E	60	

RUTINA T. Procedimiento EJE

F29F	A2	00		LDX #00
F2A1	A9	E6		LDA #E6
F2A3	85	68		STA,0 \$68
F2A5	A9	34		LDA #34
F2A7	85	A2		STA,0 \$A2
F2A9	A9	E6		LDA #E6
F2AB	85	A3		STA,0 \$A3
F2AD	20	B9	F0	JSR \$RUTINA F:
F2B0	20	E5	F0	JSR \$RUTINA Z:
F2B3	20	EE	F0	JSR \$RUTINA Y:
F2B6	B0	FB		BCC A1
F2B8	A5	A0		LDA,0 \$A0
F2BA	C9	0C		CPA #0C
F2BC	D0	F5		BNE A1:
F2BE	BD	00	02	LDA,X \$1200
F2C1	A8			TAY
F2C2	B9	94	F6	LDA, Y \$1380
F2C5	85	67		STA,0 67
F2C7	20	AB	F0	JSR RUTINA E
F2CA	E8			INX
F2CB	E8			INX
F2CC	E8			INX
F2CD	4C	BE	F2	JMP PI:

RUTINA V. Llama al procedimiento INT desde LER.

F2D0	A5	67		LDA,0 \$67
F2D2	85	69		STA,0 \$69
F2D4	A5	68		LDA,0 \$68
F2D6	85	6A		STA,0 \$6A
F2D8	A9	3C		LDA #3C
F2DA	85	67		STA,0 #67
F2DC	A9	E2		LDA #13
F2DE	85	68		STA,0 \$68
F2E0	A9	00		LDA #00
F2F2	85	A0		STA,0 \$A0
F2E4	20	AB	F0	JSR RUTINA E:
F2E7	A5	69		LDA,0 \$69
F2E9	85	68		STA,0 \$67

F2EB	A5	6 ^a	LDA,0 \$6A
F2ED	B5	68	STA,0 \$68
F2EF	C6	20	DEC,0 \$20
F2F1	60		RTS

RUTINA U. Procedimiento LER.

F300	A9	00		LDA #00
F302	85	20		STA,0 \$20
F304	85	89		STA,0 \$89
F306	A9	02		LDA #12
F308	85	8 ^a		STA,0 \$8A
F30A	A9	34		LDA #58
F30C	85	A2		STA,0 \$A2
F30E	A9	E6		LDA #03
F310	85	A3		STA,0 \$A3
F312	20	B9	F0	JSR RUTINA F:
F315	20	E5	F0	JSR RUTINA Z:
F318	20	EE	F0	U0: JSR RUTINA Y:
F31B	B0	FB		BCC U0:
F31D	A5	A0		LDA,0 \$A0
F31F	C9	0C		CMP #A0
F321	D0	F5		BNE U0:
F323	20	00	F2	U3: JSR RUTINA Q:
F326	20	E5	F0	JSR RUTINA Z:
F329	20	EF	F0	U1: JSR RUTINA Y:
F32C	90	FB		BCC U1:
F32E	A5	A0		LDA,0 \$A0
F330	C9	0A		CMP #0A
F332	D0	06		BNE U2:
F334	E6	20	FA	INC,0 420; NOP
F337	4C	23	F3	JMP U3:
F33A	C9	0B		U2: CMP #0B
F33C	D0	06		BNE U4:
F33E	20	3F	F2	JSR RUTINA R:
F341	4C	23	F3	JMP U3:
F344	C9	0B		U4: CMP #0E
F346	D0	06		BNE U5
F348	20	D0	F2	JSR RUTINA V:
F34B	4C	23	F3	JMP U3:
F34E	C9	0F		U5: CMP 0F
F350	D0	05		BNE U6
F352	68			PLA
F353	68			PLA
F354	4C	B3	F0	JMP \$02B3
F357	4C	29	F3	U6: JMP U1:

RUTINA J. Procedimiento REG.

F35A	20	EB	F3	R4: LDA #03
F35D	EA			NOP
F35E	EA	EA		NOP,NOP
F360	EA	EA		NOP,NOP
F362	EA	EA		NOP,NOP
F364	A2	04		LDX #04
F366	20	05	F1	JSR RUTINA G+5
F369	A5	A0		LDA,0 A0
F36B	C9	0A		CMP #0A
F36D	B0	08		BCS R1
F36F	20	EE	F0	R3: JSR RUTINA Y
F372	B0	03		BCS R1:
F374	4C	6F	F3	JMP R3
F377	A5	A0		R1: LDA,0 \$A0
F379	C9	0A		CMP #0A
F37B	D0	03		BNE R2:
F37D	4C	5A	F3	JMP R4:
F380	C9	0B		R2: CMP #0B
F382	D0	06		BNE R5
F384	20	96	F3	JSR RUTINA JA
F387	4C	5A	F3	JMP R4:
F38A	C9	0C		R5: CMP #0C
F38C	D0	05		BNE R6:
F38E	68			PLA
F38F	68			PLA
F390	4C	B3	F0	JMP \$02B3
F393	4C	6F	F3	R6: JMP R3:
F396	A9	3A		LDA #3A
F398	85	89		STA,0 \$89
F39A	A9	00		LDA #00
F39C	85	8A		STA,0 \$8A
F39E	85	20		STA,0 \$20
F3A0	85	39		LDA,0 \$39
F3A2	A9	DE		LDA #DE
F3A4	85	3B		STA,0 \$3B
F3A6	20	3E	F1	JSR RUTINA H:
F3A9	A5	62		LDA 0 \$62
F3AB	85	38		STA,0 \$38
F3AD	85	3A	F2	STA,0 \$3A
F3AF	20	00		JSR RUTINA Q:
F3B2	A0	00		LDY #00
F3B4	B1	38		LDA,Y \$38
F3B6	48			PHA
F3B7	4A			LSR
F3B8	4A			LSR
F3B9	4A			LSR

F3BA	4A			LSR
F3BB	C9	0A		CMP #0A
F3BD	B0	06		BCS RB1:
F3BF	18			CLC
F3C0	69	30		ADC #30
F3C2	4C	C8	F3	JMP RB2
F3C5	18			RB1: CLC
F3C6	69	37		ADC #37
F3C8	85	36		RB2: STA,0 \$36
F3CA	68			PLA
F3CB	29	0F		AND #0F
F3CD	C9	0A		CMP #0A
F3CF	B0	06		BCS RB3
F3D1	18			CLC
F3D2	69	30		ADC #30
F3D4	4C	DA	F3	JMP RB4
F3D7	18			RB3: CLC
F3D8	69	37		ADC #37
F3DA	85	37		RB4: STA #37
F3DC	20	E5	F0	JSR RUTINA Z:
F3DF	20	EE	F0	RB3: JSR RUTINA Y:
F3E2	90	FB		BCC RB3:
F3E4	A5	A0		LLDA,0 \$A0
F3E6	C9	0E		CMP #0E
F3E8	D0	F5		BNE RB3:
F3EA	60			RTS
F3EB	A9	03		LDA #03
F3ED	85	A0		STA, \$A0
F3EF	A9	34		LDA #34
F3F1	85	A2		STA, 0 \$A2
F3F3	A9	E6		LDA F6
F3F5	85	A3		STA,0 \$A3
F3F7	60			60

- INSTRUCCIONES -

AND = 01

F500	8A			TXA
F501	48			PHA
F502	E8			INX
F503	A9	04		DA #04
F505	85	2D		STA,0 \$2D
F507	20	8E	F1	JSR RUTINA K
F50A	A0	00		LDY #00
F50C	B1	70		LDA,Y \$0070
F50E	31	72		AND,Y \$0072
F510	91	70		STA,Y \$0070
F512	68			PLA
F513	AA			TXA
F514	68			PLA
F515	68			PLA
F516	4C			JMP \$02B3

BRK = 00

F519	20	67	F2	JSR RUTINA Ñ
------	----	----	----	--------------

MOV = 02

F51C	8A			TXA
F51D	48			PLA
F51E	E8			INX
F51F	A9	04		LDA #04
F521	85	2D		STA,0 \$2B
F523	20	8E	F1	JSR RUTINA K
F526	A0	00		LDY #00
F528	B1	72		LDA,Y \$72
F52A	91	70		STA,Y \$70
F52C	68			PLA
F52D	AA			TAX
F52E	68			PLA
F52F	68			PLA
F530	4C			JMP \$02B3

LDR = 03

F533	8A			TXA
F534	48			PHA
F535	E8			INX
F536	A9	02		LDA #02
F538	85	2D		STA,0 \$2D
F53A	20	8E	F1	JSR RUTINA K
F53D	A0	00		LDY #00
F53F	E8			INX
F540	BD	00	02	LDA,X \$1200

F543	91	70		STA,Y \$0070
F545	68			PLA
F546	AA			TAX
F547	68			PLA
F548	68			PLA
F549	4C	B3	F0	JMP \$02B3

OR = 04

F54C	8A			TXA
F54D	48			PHA
F54E	E8			INX
F54F	A9	04		DA #04
F551	85	2D		STA,0 \$2D
F553	20	8E	F1	JSR RUTINA K
F556	A0	00		LDY #00
F558	B1	70		LDA,Y \$0070
F55A	11	72		ORA,Y \$0072
F55C	91	70		STA,Y \$0070
F55E	68			PLA
F55F	AA			TXA
F560	68			PLA
F561	68			PLA
F562	4C	B3	F0	JMP \$02B3

EOR = 05

F565	8A			TXA
F566	48			PHA
F567	E8			INX
F568	A9	04		DA #04
F56A	85	2D		STA,0 \$2D
F56C	20	8E	F1	JSR RUTINA K
F56F	A0	00		LDY #00
F571	B1	70		LDA,Y \$0070
F573	51	72		ORA,Y \$0072
F575	91	70		STA,Y \$0070
F577	68			PLA
F578	AA			TXA
F579	68			PLA
F57A	68			PLA
F57B	4C	B3	F0	JMP \$02B3

SUM = 06

F57E	8 ^a			TXA
F57F	48			PHA
F580	E8			INX
F581	A9	04		DA #04
F583	85	2D		STA,0 \$2D
F585	20	8E	F1	JSR RUTINA K

F588	A0	00		LDY #00
F58A	B1	70		LDA,Y \$0070
F58C	18			ORA,Y \$0072
F58D	71	72		STA,Y \$0070
F58F	91	70		PLA
F591	68			TXA
F592	AA			PLA
F593	68			PLA
F594	68			JMP \$02B3
F595	4C	B3	F2	
RES = 07				
F5 98	8A			TXA
F599	48			PHA
F59A	E8			INX
F59B	A9	04		DA #04
F59D	85	2D		STA,0 \$2D
F59F	20	8E	E1	JSR RUTINA K
F5A2	A0	00		LDY #00
F5A4	B1	70		LDA,Y \$0070
F5A6	38			ORA,Y \$0072
F5A7	F1	72		STA,Y \$0070
F5A9	91	70		PLA
F5AB	68			TXA
F5AC	AA			PLA
F5AD	68			PLA
F5AE	4C	B3	E2	JMP \$02B3

- TABLAS -

TABLA 1. Teclado

F600	0D	0E	00	0F
F604	0C	09	08	07
F608	0B	06	05	04
F60C	0A	03	02	01

TABLA 2

F610	2D	2D	41	53	53	49	2D	2D
------	----	----	----	----	----	----	----	----

TABLA 3

F618	1F	22	25	28	2B	2E	31
------	----	----	----	----	----	----	----

TABLA 4

F61F	20	C1	F1
------	----	----	----

F622	20	9F	F2
F625	20	00	F3
F628	20	5A	F4
F62B	20		
F62E	20		
F631	20		

TABLA 5

F634	5F	5F	5F	5F	5F	5F	3E	3E
F63C	41	4D	41	52	47	4F	52	50
F644	2D	2D	52	45	45	4C	2D	2D
F64C	5F	5F	3A	47	45	52	55	4E
F654								
F65C								
F664								

TABLA 6. Visualizadores

F66C	B8	B9	BA	BB	BC	BD	BE	BF
------	----	----	----	----	----	----	----	----

TABLA 7

F674	00	00
F676	01	00
F678	02	00
F67A	03	00
F67C	04	00
F67E	05	00
F680	06	00
F682	07	00
F684	08	00
F686	09	00
F688	10	00
F68A	11	00
F68C	12	00
F68E	13	00
F690	14	00
F692	15	00

TABLA 8: Punteros de las subrutinas de comandos

F694	A4
F695	A7
F696	AA
F697	AD
F698	B0
F699	B3
F69A	B6
F69B	B9
F69C	BC

F69D	BF
F69E	C2
F69F	C5
F6A0	C8
F6A1	CB
F6A2	CE
F6A3	D1

TABLA 9:

F6A4	20	19	F5
F6A7	20	00	F5
F6AA	20	1C	F5
F6AD	20	33	F5
F6B0	20	4C	F5
F6B3	20	65	F5
F6B6	20	7E	F5
F6B9	20	98	F5
F6BC			
F6BF			
F6C2			
F6C5			
F6C8			
F6CB			
F6CE			
F6D1			

Finalmente para facilitar el manejo del programa monitor, se presenta la tabla 5.3, que resume la función de cada subrutina.

LISTA DE SUBRRUTINAS

RUTINA	DESCRIPCIÓN
A	MUESTRA EL CONTENIDO DEL BUFFER DEL DISPLAY
B	SENSAR EL TECLADO
C	TRANSFIERE EL ROTULO ASSI AL BUFFER DEL DISPLAY
D	SELECCIONA LA DIRECCIÓN DE INICIO DEL PROCEDIMIENTO ESCOGIDO CON EL TECLADO
E	EJECUTA LA FUNCIÓN SOLICITADA
F	SELECCIONA RÓTULOS DE LA FUNCIÓN ACCESADA AL BUFFER DEL DISPLAY
G	MUESTRA Y ALMACENA LAS INSTRUCCIONES DEL PROGRAMA
H	COMPRIME BUFFER 1 Y LO COLOCA EN BUFFER 2
I	ORDENA EL PROGRAMA
J	PROCEDIMIENTO REG
K	PREPARA LA INSTRUCCIÓN PARA SER EFECTUADA EN LAS DIRECCIONES \$0070-\$0073
L	PROCEDIMIENTO INT
M	GRABA Y ORDENA LOS DATOS DE ENTRADA
N	LIMPIA LOS VISUALIZADORES
Ñ	SALE DEL PROGRAMA INT
O	INICIALIZA CONSTANTES PARA EL PROCEDIMIENTO INT
P	RUTINA PRINCIPAL
Q	LEE UNA LÍNEA DE PROGRAMA GRABADO EN BUFFER 2.
R	DECREMENTA EL PUNTERO DE LECTURA DE LAS LÍNEAS DEL PROGRAMA SALVADO
S	MUESTRA EL CONTENIDO DE UN REGISTRO
T	PROCEDIMIENTO EJE
U	ENCUENTRA EL PROCEDIMIENTO LER
V	LLAMA AL PROCEDIMIENTO INT DESDE LER
X	RETARDA EL TIEMPO DE LECTURA DE UNA TECLA
Y	HABILITA EL TECLADO
Z	HABILITA LOS VISUALIZADORES

TABLA 5.3 LISTA DE SUBRRUTINAS DEL PROGRAMA MONITOR

F69D	BF
F69E	C2
F69F	C5
F6A0	C8
F6A1	CB
F6A2	CE
F6A3	D1

TABLA 9:

F6A4	20	19	F5
F6A7	20	00	F5
F6AA	20	1C	F5
F6AD	20	33	F5
F6B0	20	4C	F5
F6B3	20	65	F5
F6B6	20	7E	F5
F6B9	20	98	F5
F6BC			
F6BF			
F6C2			
F6C5			
F6C8			
F6CB			
F6CE			
F6D1			

Finalmente para facilitar el manejo del programa monitor, se presenta la tabla 5.3, que resume la función de cada subrutina.

CAPITULO 6

MANUAL DE USUARIO

Para poder utilizar el Entrenador ASSI es necesario conocer sus características, lenguaje de programación y las variables que este es capaz de controlar. Además los menús y submenús a los que se accesa en los diferentes procedimientos.

Está sección reúne la información necesaria para poder utilizar el entrenador ASSI.

Se divide en cuatro partes:

- 6.1 **CARACTERÍSTICAS DEL ENTRENADOR.**
- 6.2 **COMO UTILIZAR EL ENTRENADOR ASSI.**
- 6.3 **COMO ANEXAR UNA INSTRUCCIÓN.**
- 6.4 **PUNTOS DE PRUEBA.**

6.1 CARACTERÍSTICAS DEL ENTRENADOR

6.1.1 CARACTERÍSTICAS DE HARDWARE:

- a. Teclado hexadecimal
- b. Display matricial de 8 caracteres con capacidad para aceptar códigos ASCII.
- c. 16 líneas bidireccionales de entrada-salida.
- d. 1K de memoria RAM, con capacidad de expansión hasta 4K.
- e. 2K de memoria ROM, con capacidad de expansión hasta 4K.
- f. Cuatro salidas de reloj:
 - 5 KHz.
 - 2.5 KHz.
 - 1.25 KHz.
 - 0.625 KHz.

6.1.2 CARACTERÍSTICAS DE SOFTWARE:

1. Lenguaje de programación ELL.
2. Programa monitor.
3. 8 instrucciones básicas con capacidad de expansión a 80 instrucciones.

6.2 COMO UTILIZAR EL ENTRENADOR ASSI

1. Función de cada Tecla

Cuando se enciende el entrenador ASSI el programa monitor muestra el rótulo de iniciación:

- - A S S I - -

Esto indica que el monitor está esperando que el usuario seleccione una de las cuatro teclas correspondientes a los cuatro procedimientos principales.

1 EJE	2 LER	3 REG	A
4	5	6	B
7	8	9	C
*	0 INT	#	D

2. Introduciendo el programa

TECLA 0: Selecciona el procedimiento INT. Una vez se ha entrado a él mostrará el rótulo:

> > | - - - - - |

el cual indica que se puede empezar a editar el programa.

Ahora que se esta dentro del "Procedimiento INT", las siguientes teclas son validas:

1	2	3	A LMP
4	5	6	B GRA
7	8	9	C SAL
*	0	#	D

donde:

LMP: Limpia el visualizador por si hay algún error en la edición.

GRA: Graba una línea del programa.

SAL: Regresa a la rutina principal.

Ejemplo:

- Encienda el entrenador ASSI.
- Cuando aparezca el rótulo de presentación presione la tecla "0", para acceder al Procedimiento INT.
- Cuando aparezca el rótulo de presentación del procedimiento le indicará que puede empezar a editar el programa, Presione "0" y después "1", en el visualizador aparecerá:

```
> > | 0 | 1 | _ | _ | _ | _ |
```

Cada vez que se presione una tecla, ésta aparecerá inmediatamente en el visualizador.

- Presione la tecla A; (LMP); observará que se limpió el visualizador y los datos se han perdido.
- Vuelva a introducir 01 está vez acompañados de 03 y 04. En el visualizador aparecerá:

```
> > | 0 | 1 | 0 | 3 | 0 | 4 |
```

- f) Presione la tecla B; (GRA); para grabar la línea del programa que se ha introducido; verá que los datos desaparecen. Esto se debe a que GRA a la vez que graba los datos, limpia el visualizador, para facilitar visualmente la introducción de la siguiente línea del programa.
- g) Digite 00 y grabe el dato con "B".
- h) Ahora presione la tecla "C"; (SAL); con esto se regresa a la rutina principal.




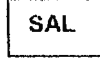
3. Leer un programa ya grabado en memoria

TECLA 2: Selecciona el procedimiento "LER", cuyo rótulo de presentación es:

- - L E E R - -

Cuando el rótulo aparece el procedimiento LER esta esperando la tecla "C" (continuar), que es la única que será reconocida a este nivel. Al presionar la tecla "C" quedan habilitadas todas las funciones del procedimiento LER, las siguientes teclas son válidas:

1	2	3	A →
4	5	6	B ←
7	8	9	C
* SAL	0	# EDI	D

	: Muestra la siguiente línea del programa.
	: Muestra la línea anterior del programa.
	: Edita una línea ya grabada.
	: Regresa a la rutina principal.

Ejemplo:

- a. Con el rótulo de la rutina principal presente, presione la tecla "2", se mostrará en el visualizador lo siguiente:

```
- - | L E E R - -
```

- b. Para entrar a las funciones del procedimiento presione la tecla "C". Inmediatamente observará en el visualizador la primera línea del programa; retomando el ejemplo anterior se verá:

```
> > | 0 | 1 | 0 | 3 | 0 | 4 |
```

- c) Con la tecla A (→), podrá observar la siguiente línea

```
> > | 0 | 0 | - | - | - | - |
```

efectivamente 00 nos indica el final del programa.

- d) La tecla B (←), regresa a línea anterior.

Al presionar la tecla E (EDI), se podrá editar toda la línea. Aquí la línea se edita de igual forma que en el procedimiento INT.

- e) Si se presiona la tecla * (SAL), regresamos a la rutina principal.

4. Observar el contenido de un registro

TECLA 3: Selecciona al procedimiento REG, el rótulo de presentación es:

N U R E G : _ _

En este caso el procedimiento REG está esperando que se introduzca el número de registro a observar, las funciones de estas subrutinas son:

1	2	3	A LIM
4	5	6	B VER
7	8	9	C SAL
*	0	# REG	D

LIM	: Limpia los dos últimos bytes del visualizador.
VER	: Muestra el contenido del registro seleccionado.
SAL	: Regresa a la rutina principal.
REG	: Habilita la selección de un nuevo registro.

Ejemplo:

a) Presione la tecla tres observe que aparece el rótulo:

N U R E G : _ _

Esto significa que el procedimiento REG esta esperando que se edite el número del registro.

- b) Escriba "03" y luego la tecla "A", (LIM); esto limpiará los dos bytes más a la derecha del visualizador y esperará para que se edite el nuevo valor.
- c) Escriba "05" y luego presione la tecla "B"; (VER); esto mostrará el contenido del registro 5 en el visualizador junto con el mensaje:

N U 0 5 = > 0 7

Esto indica que el registro número 5 contiene el dato 07. (El dato puede variar).

- d) Al presionar la tecla "#"; (REG); regresa a buscar el rótulo principal, para permitirle al usuario ver el contenido de otros registros.
- e) Si se presiona la tecla "C" (SAL); regresará a la rutina principal.

5. Ejecutar un programa

TECLA 1: Selecciona el procedimiento EJE, y su rótulo de presentación es:

P R O G R A M A

Esto indica que la rutina EJE está esperando el comando que le ordene ejecutar el programa.

La función que ordena ejecutar un programa es CONT y se muestra en el teclado con la tecla "C".

1	2	3	A
4	5	6	B
7	8	9	C CONT
*	0	#	D

Una vez presionada la tecla C, se ejecuta el programa y se vuelve a la rutina principal.

6.3 CÓMO ANEXAR UNA INSTRUCCIÓN AL LENGUAJE ELL

En la tabla 5.1 se muestran 9 instrucciones que son aceptadas por el lenguaje ELL. Se puede pensar el número de instrucciones es muy limitada para crear programas funcionales, eso es cierto, por eso ELL es un programa que permite anexar nuevas instrucciones y reconocerlas como parte del mismo.

ELL está construido bajo el ambiente del μ P 6502, así que cualquier instrucción que se quiera anexar deberá estar desarrollada en el código de maquina del μ P 6502.

En la figura 6.1 se presenta el flujograma que muestra el proceso de cómo una instrucción es interpretada y ejecutada.

El programa que se va a ejecutar se encuentra almacenado en el *buffer 2* el cual se encuentra en la dirección \$0200 hasta \$02FF, aquí se comienzan a tomar los opcode con sus respectivos operandos. Si el opcode es 00 ("SAL", ver tabla 5.1) el programa termina y regresa a la rutina principal, de no ser así, se compara ese número con el número correlativo de la *tabla 9 del programa monitor* y se ejecuta el comando.

En forma más detallada, en la *tabla 9 del programa monitor* se encuentran ordenadas las direcciones de inicio de las subrutinas de cada instrucción de ELL.

Cuando el opcode no es igual a 00, el programa adquiere la dirección de la subrutina de la instrucción que se va a ejecutar a partir del opcode encontrado. Cuando se termina de ejecutara subrutina, éste regresa a tomar el próximo opcode hasta que encuentre 00 y termine.

Un ejemplo:

Si en el buffer se encuentra la siguiente información;

LÍNEA 1	01 03 05	AND	03, 05
LÍNEA 2	00	SAL	

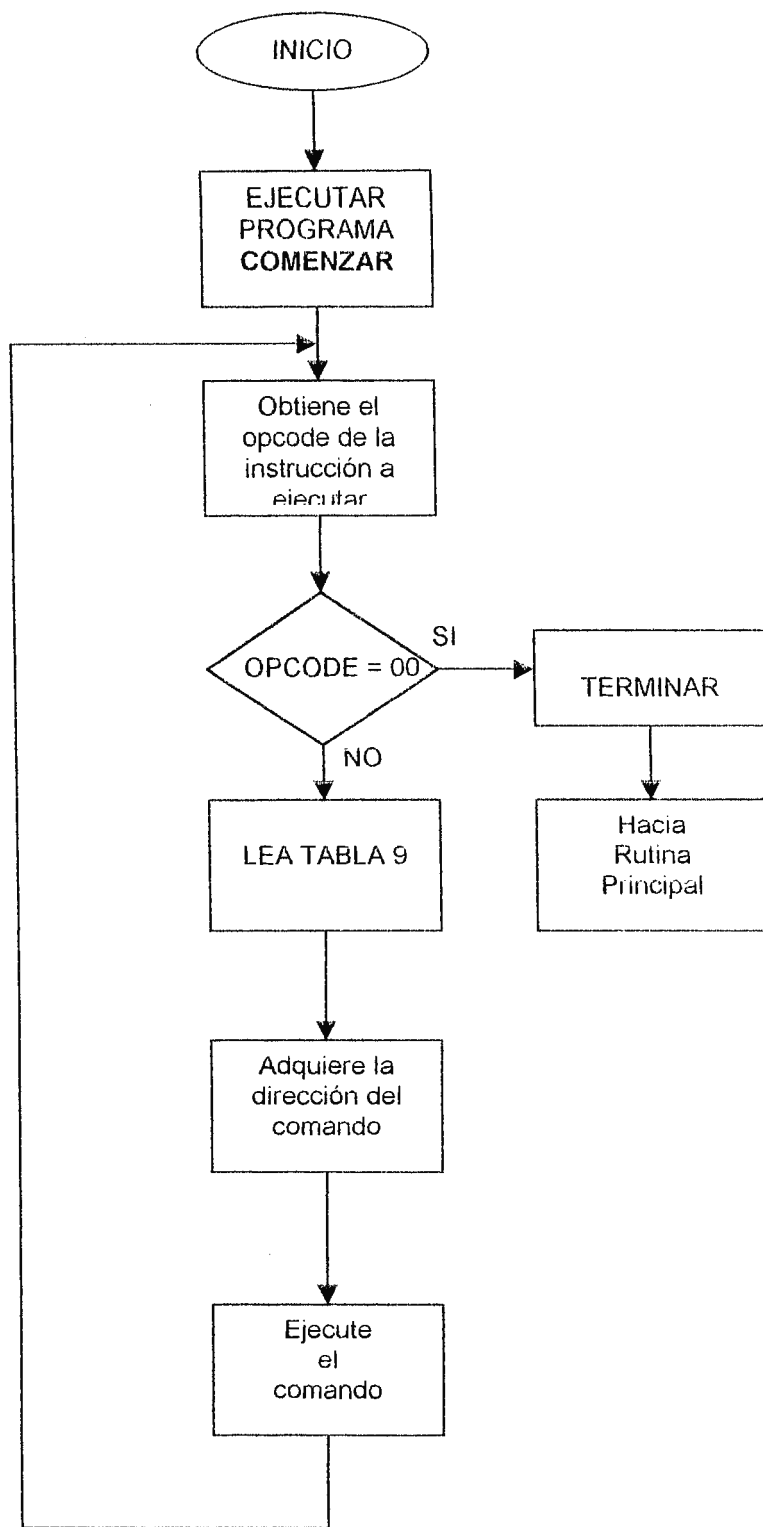


FIGURA 6.1 INTERPRETACION Y EJECUCION DE UNA INSTRUCCIÓN.

Como el primer opcode es 01 el programa busca la dirección de inicio de forma correlativa en la *tabla 9 del monitor* y ejecuta la subrutina que se encuentra en esa dirección.

En la dirección F519 se encuentra la subrutina correspondiente al opcode 01, es decir a la rutina de la función AND.

Si se quiere crear una nueva instrucción es obvio que el opcode debe ser 08, para que su posición en la tabla 6.1 sea la del número correlativo 8, la cuál esta libre.

En la tabla 6.1 se muestran las direcciones de cada subrutina (Instrucción de ELL).

# CORRELATIVO	OPCODE	INSTRUCCIÓN	DIRECCION
0	00	SAL	F500
1	01	AND	F519
2	02	MOV	F51C
3	03	LDR	F533
4	04	OR	F54C
5	05	EOR	F565
6	06	SUM	F57E
7	07	RES	F598

TABLA 6.1 DIRECCIONES DE LAS SUBRRUTINAS

Se ha visto que para anexar una instrucción:

- El opcode debe ser un número correlativo de la *tabla 9 del monitor*.
- La *tabla 9 del programa monitor* debe contener la dirección de inicio de la subrutina, la cual se tomará como instrucción.
- La subrutina debe ser escrita en código máquina del μ P 6502.

6.3.1 FORMATO DE LA SUBROUTINA

El formato divide en dos partes:

- a) Instrucciones predeterminadas. Éstas se encargan de que el programa monitor reconozca la instrucción.
- b) La operación que se desea ejecutar.

Por ejemplo, se presenta a continuación el caso de la función AND:

F500	8A	TXA	}	INSTRUCCIONES PREDETERMINADAS
F501	48	PHA		
F502	E8	INX		
F503	A9 04	LDA #04		
F505	85 2D	ST A,0 #2D		
F507	20 8E F1	JSR RUTINA K		
F50A	A0 00	LDY, #06	}	OPERACIÓN A EJECUTAR
F50C	B1 70	LDA, Y \$0070		
F50E	31 72	AND, Y \$0072		
F510	91 70	STA, Y \$0070	}	INSTRUCCIONES PREDETERMINADAS
F512	68	PLA		
F513	AA	PLA		
F514	68	PLA		
F515	68	PLA		
F516	4C	JMP \$02B3		

- a) **INSTRUCCIÓN PREDETERMINADA:** Las instrucciones predeterminadas son fijas y toda nueva instrucción ELL que se quiera anexar deberá tenerlas.

La única instrucción predeterminada que puede variar su valor es: A9 04, o sea "LDA #04", que en algunas ocasiones será: "A9 02", es decir "LDA #02"; pero esto dependerá de la operación que se ejecute.

Ej: (Ver sección 5.4).

La instrucción MOV ocupa "A9 04", o sea "LDA #04"

La instrucción LDR ocupa "A9 02", o sea "LDA #02"

- Si la operación que se va a utilizar incluye dos registros, un registro fuente y un destino, entonces se ocupan "A9 04", o sea "LDA #04".
- Si la operación utiliza una constante se utiliza "A9 02", o sea "LDA #02".

➤ Ej: MOV 03, 05 

Esto significa: *mueva el contenido del registro fuente "05" y depositelo en el registro destino "03".*

Como MOV ocupa dos registros, entonces se utiliza "A9 04", o sea "LDA 04".

➤ Ej: LDR 03, 05 

Esto significa: *mueva la constante "05" al registro "03" como LDR ocupa una constante y un registro, entonces se utiliza "A2 02", o sea "LDA #02".*

b) OPERACIÓN A EJECUTAR: En este espacio se coloca la operación real que se quiere que la instrucción ELL ejecute.

La rutina "K", es la encargada de preparar las direcciones de los registros que se van a ejecutar.

➤ Ej:  MOV 01, 03

Donde "01" registro destino y "03" registro fuente.

La Rutina K; que es donde se encuentra almacenado el programa; traslada del *buffer 2* las direcciones de los registros ocupados a las direcciones 0070 hasta 0073.

➤ Ej: AND 01 03; se almacena en el buffer 2

Como:

02	01	03
Opcode	Operando	

El registro "01" se encuentra en la dirección "\$0001" y el registro "03" se encuentra en la dirección "\$0003".

Entonces la rutina K ordena las direcciones de los registros y las coloca de forma ordenada como sigue:

0070	00	}	Registro fuente
0071	01		
0072	00	}	Registro destino
00 73	03		

Existen dos tipos diferentes de operaciones a ejecutar en el lenguaje ELL:

Primer Caso: Operación de registro a registro.

Para el caso de la función AND, las siguientes instrucciones ejecutan la operación deseada:

1)	B1	70	LDA,Y \$0070
2)	31	72	AND,Y \$0072
3)	91	70	STA,Y \$0070

Se utiliza el modo de direccionamiento indirecto y del μ P 6502 para obtener el resultado esperado.

1. La primera línea del programa se lee.

“Cargue el acumulador con el contenido de la dirección que se encuentra dentro de la dirección 0070 y 0071, que es “00” y “01” respectivamente”.

Es decir que, se carga al acumulador el contenido de la dirección \$0001 en forma directa, que para el caso es el registro 1 o registro destino.

2. La segunda línea:

Ejecuta la operación AND entre el acumulador y el contenido de la dirección \$0003, que para el caso es el registro 3 o registro fuente.

3. La tercera línea:

Almacena el acumulador en la dirección \$0001 o registro destino.

De esta forma es interpretada la instrucción AND del programa ELL dentro del programa monitor cuando la operación es de registro a registro.

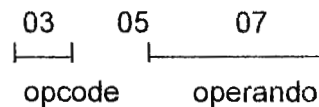
Segundo Caso: Operación de constante a registro:

➤ Ejemplo: Para la función LDR (ver sección 5.4):

Las líneas que ejecutan la operación son:

- 1) INX
- 2) LDA, X \$1200
- 3) STA, Y \$70

Si en el buffer se encuentra:



Esto significa cargue la constante "07" al registro "05".

Como "07" es una constante y no es un registro hay que buscar la forma de extraer "07" del *buffer 2*. Esto se logra mediante un registro especial denominado **Registro X**.

El **registro X** es un puntero que indica la posición de un dato en el *buffer 2*.

Pro ejemplo:

1200	1201	1200	DIRECCIÓN
03	05	07	DATOS
X = 0	X = 1	X = 2	VALOR DE X

Aquí, el valor de X al entrar a cualquier subrutina de instrucción del ELL tiene el valor cero. X es incrementado a 1 por la instrucción predeterminada para cada subrutina de instrucción del ELL.

Por ejemplo para el siguiente programa:

F533	8A	TXA
F534	48	PHA
F535	E8	INX
F536	A8 02	LDA,#02

F538	85 8D	STA,0 \$02	
F53A	20 8E F1	JSR \$RUTINA K	
F53D	A0 00	LDY #00	
F53F	E8	INX	↕ Línea 1
F540	BD 00 02	LDA,X \$1200	↕ Línea 2
F543	91 70	STA,Y \$0070	↕ Línea 3
F545	68	PLA	
F546	AA	TAX	
F547	68	PLA	
F548	68	TAX	
F549	4C B3 F0	JMP \$02B3	

Se tiene

1. INX; con esto obtenemos el valor de X=2.
2. LDA,X \$0070; carga el acumulador en el registro destino que es "05".
3. STA,Y \$0070; almacene el acumulador en el registro destino que es "00".

Entonces la constante "07" es almacenada en el registro "05".

6.3.2 EJEMPLO: CREANDO LA INSTRUCCIÓN NOT

FUNCION NOT: Invierte el contenido de un registro por medio de una mascara binaria.

NMOTECNICO: **INV Registro Fuente, Mascara**

EJ:

INV 03, 01

Significado: Invierte el bit 1 del Registro 3.

Para que esta instrucción, sea aceptada por ELL se siguen los siguientes pasos:

- 1) Asignarle un Opcode. Para nuestro ejemplo " 08 ".
- 2) Asignarle en la tabla de datos 9, la dirección de inicio de la subrutina y la instrucción JSR.

Para el caso de la función NOT, sería:

F6BC 20 B1 F5

Ya que la instrucción " RES " termina en la dirección F5B0, INV debe de comenzar en F5B1.

- 3) Construir la subrutina en código maquina del microprocesador 6502.
- 4) Determinar a acción de la subrutina.

Ej: INV 03, 01

Invierte el bit 01 del registro 03

➤ Ahora bien, ¿Cómo se invierte el bit o los bits de un registro?

La forma más fácil es:

- a) Cargar al acumulador con el dato que se quiere invertir, en este caso el contenido de algún registro.
- b) Aplicar la mascara deseada.
- c) Devolver el dato que se carga al acumulador a su lugar de origen.

Para nuestro ejemplo:

INV 03, 01 ➡ Nemotecnico.

08 03 01 ➡ Código Máquina.

Donde:

- Registro Fuente: 03
- Constante: 01

Entonces:

- 1) *Se carga el contenido del Registro Fuente al Acumulador.* **LDA,Y \$0070**
- 2) *Se le aplica por medio del modo de direccionamiento absoluto x la mascara al acumulador* **INX**
EOR,X \$0200
- 3) *Se devuelve el dato del acumulador al registro origen.* **STA,Y \$0070**

➤ ¿ Porqué se utiliza la instrucción EOR para hacer la mascara ?

Para comprender esto, se presenta el siguiente ejemplo:

Acumulador	=	0000 0111	07
Mascara	=	0000 0001	01
Aplicamos EOR	=	0000 0110	06

Entonces la operación binaria EXOR (EOR) dará un cero cada vez que dos bits sean iguales.

INV es una instrucción que opera un registro con una constante, por lo que se utilizará la siguiente instrucción predeterminada:

A9 02 ó "LDA #02 "

Reuniendo todo lo anterior se obtiene:

F5B1	8A			}	Instrucciones Predeterminadas	
F5B2	48					
F5B3	E8					
F5B4	A9	02				
F5B6	85	2D			}	Operación a Ejecutar
F5B8	20	8E	F1			
F5BB	A0	00				
F5BD	B1	70				
F5BF	E9					
F5C0	5D	00	02		}	Instrucciones Predeterminadas
F5C3	91	70				
F5C5	68					
F5C6	AA					
F5C7	68					
F5C8	68					
F5C9	4C	B3	F0			

Después de anexar estas líneas al programa monitor se obtiene una nueva instrucción con código operando " 08 " llamada INV.

Siguiendo los pasos anteriores pueden crearse muchas instrucciones, para optimizar el funcionamiento de este entrenador.

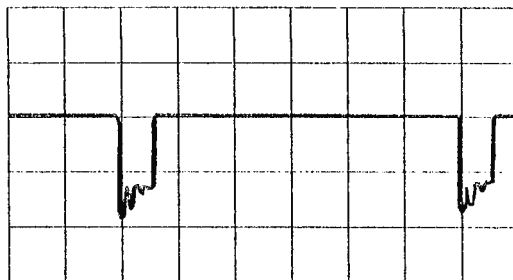
6.4 PUNTOS DE PRUEBA

Siendo el PLC ASSI un equipo didáctico, se ha incluido en el manual de usuario algunos puntos de prueba que ayudan a comprender el funcionamiento interno, de las señales de comunicación y de selección entre los diferentes dispositivos.

Estas señales se observan con ayuda de un osciloscopio cuando en el ASSI se está ejecutando la rutina principal.

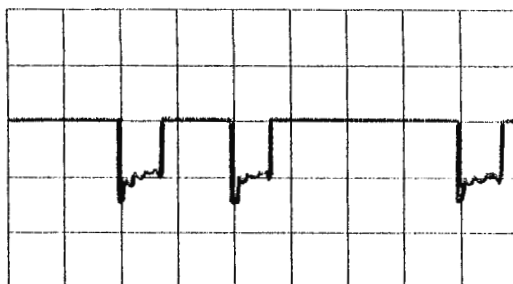
▪ PUNTO 1:

Señal R/W, que controla la Lectura y Escritura del microprocesador en los diferentes dispositivos del sistema.



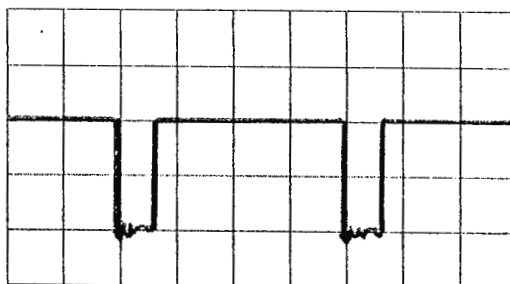
▪ **PUNTO 2:**

Se toma de la salida del 74LS138 utilizado para seleccionar la PIA 1. Éste es el pin 1Y1 del IC13.



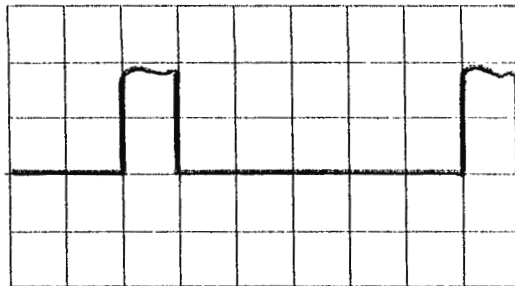
▪ **PUNTO 3:**

Se toma de la salida del 74LS138 utilizado para seleccionar la Memoria de Acceso Aleatorio. (RAM). Éste es el pin 1Y0 del IC9.



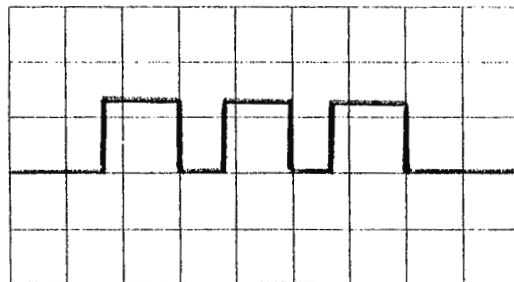
▪ PUNTO 4:

Se toma de la salida del 74LS138 utilizado para seleccionar la Memoria de Solo Lectura que contiene el programa monitor (MROM). Éste es el pin 1Y7 del IC9.



▪ PUNTO 5:

Señal de reloj del sistema; ϕ_2 ; éste es el pin 37 del μP 6502.

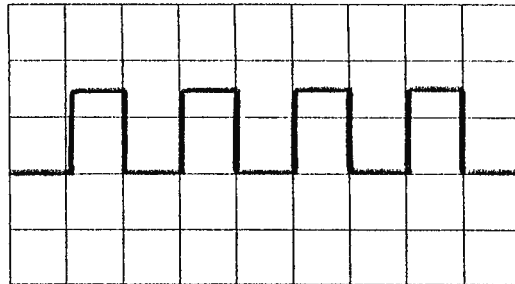


▪ PUNTO 6 Y 7:

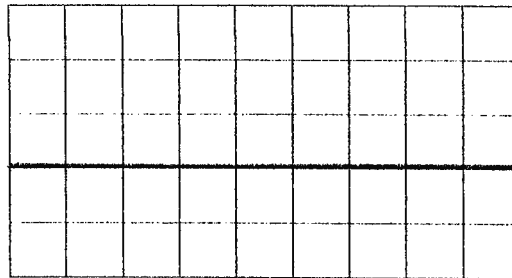
Señal de salida de temporizador al aplicar al registro de control de temporización el

dato 01:

PUNTO 6:



PUNTO 7:



CONCLUSIONES

Cuando se inicio este trabajo de graduación se plantearon una cantidad de objetivos, sin embargo no se tenia la certeza de poder cumplirlos. Pero al finalizarlo, ya no se valora el trabajo solo en base a haber cumplido dichos objetivos, sino más bien al camino recorrido para alcanzarlos. En nuestro caso, este camino nos deja una serie de experiencias que nos llevan a ciertas conclusiones, tanto generales como técnicas.

1. De forma general puede decirse que básicamente, los controladores lógicos programables (PLC) son diseñados para automatizar sistemas industriales. Éstos son programados para controlar variables específicas de un proceso, tales como: temperatura, luminosidad, presión velocidad, etc. Los tipos de PLCs que se encuentran disponibles en el mercado, están destinados al control de sistemas o maquinaria sofisticada, lo que los vuelve robustos y caros; y su programación, la cual debe cumplir con todos los requisitos que posee la planta, es muy complicada. Por otro lado, para sistemas pequeños; como el control del sistema eléctrico de una casa habitacional; sería inapropiado utilizar un PLC como los descritos anteriormente, pues se estaría subutilizando.

Por ejemplo, para automatizar sistemas pequeños, como por ejemplo, el control de un horno industrial en una panadería, se puede diseñar un pequeño PLC que cumpla con los requisitos y especificaciones del usuario.

En los laboratorios de la UDB, se cuenta con PLCs diseñados para la automatización de sistemas industriales grandes; pero, dichos laboratorios no cuentan con un equipo didáctico, enfocado especialmente al estudio interno del PLC, por lo que todos los cursos impartidos ahí se limitan a la programación de éstos, obviando el diseño de Autómatas Programables según las necesidades del usuario.

Ya que éste es un problema generalizado en las instituciones educativas de nuestro país; se concluye lo siguiente:

- En el país desperdicia un gran potencial en cuanto a capacidad de diseño electrónico, el cual vendría a bajar los costos de la automatización industrial.
- Explotar este potencial de diseño permitiría ampliar el campo de aplicación de la automatización, mejorando así la calidad de producción de una gran cantidad de industrias que no poseen actualmente la capacidad económica de utilizar el costoso equipo extranjero.
- La explotación de la capacidad de diseño electrónico industrial generaría una nueva área de desarrollo económico.

Entonces, el Entrenador ASSI, es uno de los primeros pasos para solventar esta necesidad de ampliar el área didáctica de la automatización industrial; ya que éste, está orientado al conocimiento interno de las partes principales que constituyen el hardware de un PLC, así como el programa monitor que lo gobierna. Por lo tanto el ASSI es una propuesta de solución para la Universidad Don Bosco; llenado el vacío de un equipo puramente didáctico destinado a orientar al estudiante en el área de diseño de PLCs que ofrezcan una opción de solución a requerimientos de automatización a pequeña escala.

2. Analizando desde un punto de vista técnico, todos los PLCs necesitan un elemento principal el cual se encargue de administrar todas las operaciones que se van a realizar. Este elemento debe ser capaz de adquirir, interpretar, procesar y codificar las instrucciones que recibe a través de un programa.

Los microprocesadores cumplen con lo antes mencionado, por lo tanto:

- El entrenador ASSI utiliza el microprocesador 6502, pero se pudo haber utilizado cualquier otro microprocesador. Se debe hacer notar que el hardware desarrollado en el diseño no es el más óptimo, pero si es funcional. Esto, significa que pueden hacerse cambios para lograr una reducción en el número de elementos utilizados.

El diseño del entrenador ASSI es una guía para el desarrollo de futuros proyectos, ya sea que se mejore la versión ya existente de dicho

entrenador, o que se apliquen los criterios de diseño de software y hardware presentados, pero utilizando diferentes tecnologías.

- La estructura del programa monitor, es decir, la lógica de funcionamiento puede aplicarse a otros microprocesadores.
- En cuanto al hardware, todos los elementos, excepto el teclado hexadecimal fueron extraídos de computadoras en calidad de chatarra, promoviendo así el reciclaje de los componentes, con lo que los costos de implementación bajan considerablemente.

BIBLIOGRAFIA

- [1] Jeannette A. Fines and staff of Quintech, Lab-Volt, PROGRAMABLE LOGIC CONTROL, 1988.
- [2] Ronald J. Tocci, SISTEMAS DIGITALES, PRINCIPIOS Y APLICACIONES, 1988.
- [3] Buck Engineering Co. Inc., Lab-Volt, INTERFAZ DE ENTRADA/SALIDA DE MICROPROCESADORES, 1985.
- [4] Buck Engineering Co. Inc., Lab-Volt, MEMORIA DEL MICROPROCESADOR, CONCEPTOS Y APLICACIONES, 1984.
- [5] Frederick F. Driscoll. MICROPROCESSOR-MICROCOMPUTER TECHNOLOGY, 1983.
- [6] D.E. Pippenger and E.J. Tobaben, LINEAR AND INTERFACE CIRCUITS APPLICATIONS. (II Ed.), TEXAS INSTRUMENTS ELECTRONICS SERIES, 1988.
- [7] Douglas M. Considine, PROCESS/INDUSTRIAL INSTRUMENTS AND CONTROLS HANDBOOK. (II Ed). 1993.
- [8] E. Mandado, E. Tassis, DISEÑO DE SISTEMAS DIGITALES CON MICROCESADORES, 1983.
- [9] C.J. Savant, Jr., DISEÑO ELECTRONICO, CIRCUITOS Y SISTEMAS. (II Ed.), 1992.
- [10] J. Horacio Egüizabal, DISEÑO DE SISTEMA CONTROLADOR PROGRAMABLE DE PROCESOS INDUSTRIALES, 1994.
- [11] Rudolf F. Graf, MODERN DICTIONARY OF ELECTRONICS, 1991.
- [12] Donald G. Fink, EQUIPOS ELECTRONICOS DE MEDICION Y CONTROL. (II Ed), 1992.

APENDICE A

EL AMPLIFICADOR OPERACIONAL

EL AMPLIFICADOR OPERACIONAL. (AMP – OP)

El Amplificador Operacional Ideal.

Los amplificadores operacionales son ampliamente utilizados en sistemas electrónicos lineales. Idealmente el Amp-Op consiste en un amplificador de alta ganancia, directamente acoplado, que en general se alimenta con fuentes positivas y negativas, permitiendo así que la salida tenga excursiones tanto por arriba como por debajo de tierra.

En la figura A-1 se presenta el símbolo del Amp-Op y en la figura A-2 se muestra su circuito equivalente.

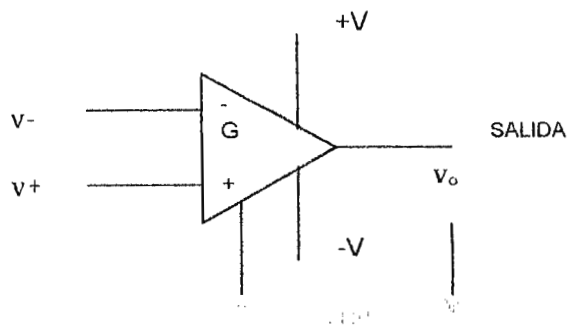


FIGURA A-1. SIMBOLO PARA EL AMP-OP IDEAL

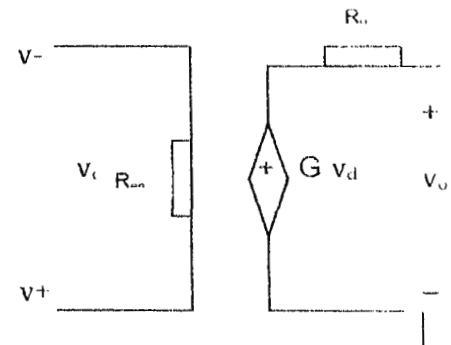


FIGURA A-2. CIRCUITO EQUIVALENTE PARA EL AMP OP IDEAL.

Las terminales de entrada del Amp-Op se conocen como “Inversora” y “No Inversora”, las cuales se alimentan con tensiones v_- y v_+ , respectivamente.

El circuito equivalente contiene una fuente de voltaje que depende del voltaje de entrada. La salida es independiente, y se representa con una resistencia R_o .

La salida del amplificador no depende de las magnitudes de las tensiones de entrada, sino de la diferencia entre ellas, llamada "Tensión diferencial de entrada v_d ". Entonces la tensión de entrada derivada de la diferencia es:

$$v_d = v_+ - v_-$$

La impedancia de entrada se representa con una resistencia " R_{en} " en el circuito equivalente. La tensión de salida se relaciona proporcionalmente con la de entrada, mediante la ganancia de lazo abierto " G ", según la siguiente ecuación:

$$v_o = G (v_+ - v_-)$$

Las características del Amp-Op ideal son:

1. Resistencia de entrada $R_{en} \rightarrow \infty$.
2. Resistencia de salida $R_o = 0$.
3. Ganancia de tensión de lazo abierto, $G \rightarrow \infty$.
4. Ancho de banda $\rightarrow \infty$.
5. $v_o = 0$, cuando $v_+ = v_-$.

El hecho de que la ganancia de lazo abierto sea infinita significa que:

$$\text{Si } v_+ - v_- = v_o / G ;$$

haciendo que G se aproxime a infinito se obtiene:

$$v_+ - v_- = 0 .$$

Y como la tensión entre las dos terminales es cero;

$$v_+ = v_-$$

Entonces, como la resistencia de entrada es infinita, la corriente en cada entrada, *inversora* y *no inversora* es cero.

Análisis del Amp-Op Ideal.

Para el análisis se utilizan dos de las propiedades del amplificador operacional ideal:

1. La tensión entre v^- y v^+ es cero, o $v^- = v^+$.
2. La tensión en v^- y v^+ es cero.

Entonces; recordando que las corrientes en v^- y v^+ son cero; los pasos para el análisis son:

1. Se escribe la ecuación de nodos de Kirchhoff en la terminal no inversora, v^+ .
2. Se escribe la ecuación de nodos de Kirchhoff en la terminal inversora, v^- .
3. Se hace $v^- = v^+$ y se resuelven las ganancias de lazo cerrado que se desean.

Amp-Op en configuración de lazo cerrado con retroalimentación.

La configuración mostrada en la figura A-1 se llama *de lazo abierto* porque no existe retroalimentación de la salida a la entrada, pero la configuración más utilizada en aplicaciones lineales es la de *lazo cerrado con retroalimentación*, la cual se muestra en la figura A-3. Los elementos externos sirven para retroalimentar una porción de la señal de salida a la entrada. Si estos elementos se colocan entre la entrada inversora y la salida, disminuye la ganancia de lazo cerrado, pues una parte de la salida se resta de la entrada.

La retroalimentación además de reducir la ganancia total, también la vuelve menos sensible al valor de G .

La ganancia de lazo cerrado depende solo del valor de los elementos de retroalimentación, y no del valor de G .

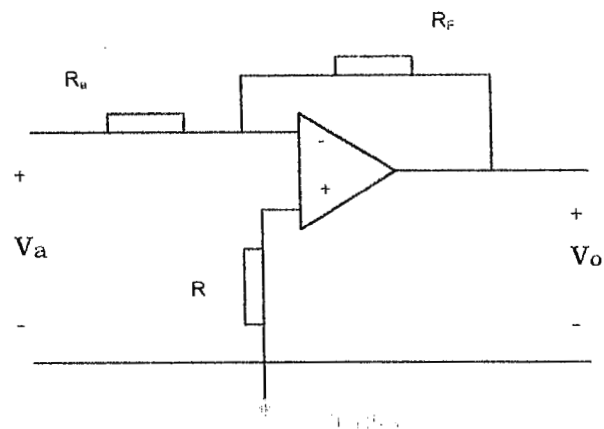


FIGURA A-3. CIRCUITO DE AMP-OP CON RETROALIMENTACION EN CONFIGURACION INVERSORA.

En un circuito simple de Amp-Op con retroalimentación, como el que se muestra en la figura A-3, se utiliza un solo resistor, R_F , para conectar la tensión de salida, v_o a la entrada inversora, v_- . Se utiliza otro resistor, R_a , para conectar la tensión de entrada a la entrada inversora.

Los Amp-Op se pueden configurar para realizar muchas operaciones, utilizando además resistores y capacitores. Algunas de estas operaciones son sumar, restar, integrar, filtrar, comparar, y amplificar; pero estas se basan en alguna de las siguientes configuraciones:

- Amplificador Inversor.
- Amplificador no inversor.

El Amplificador Inversor

En la figura A-3 se muestra un amplificador inversor con retroalimentación. En este circuito la ganancia de lazo cerrado, v_o/v_a , depende de la relación de los dos resistores, R_F/R_A , y es independiente de la ganancia de lazo abierto, G .

Con la retroalimentación desde la salida a la entrada, a través de R_F se lleva la tensión diferencial, $v_+ - v_-$, a cero. Y como V_+ se encuentra conectada a tierra, v_- se lleva también a cero; se dice entonces que existe una *tierra virtual* en v_- . Por lo tanto

$$v_+ = v_- = 0$$

Tierra virtual significa que v_- es cero pero no fluye ninguna corriente real a tierra.

La ecuación que relaciona el voltaje de entrada v_a con el voltaje de salida, v_o , es:

$$V_o = -V_a (R_F / R_a)$$

El Amplificador no Inversor

El amplificador no inversor se muestra en la figura A-4.

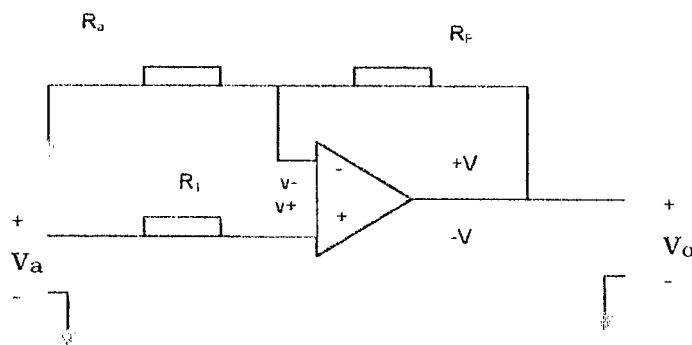


FIGURA A-4. AMPLIFICADOR NO INVERSOR.

Esta configuración permite usar al Amp-Op como un amplificador no inversor con alta impedancia de entrada. La ganancia de voltaje del circuito se puede establecer con los resistores R_a y R_F .

En este circuito la impedancia de entrada es alta porque el único paso a tierra para la corriente de entrada es a través de la alta impedancia de entrada del amplificador.

Como la corriente necesaria para el amplificador es muy baja se colocan R_a y R_F como divisor de voltaje de poca carga, entonces la corriente en ambos resistores es igual y la relación de voltajes es:

$$V_o = V_a (R_F + R_a) / R_a$$

APENDICE B

SET DE INSTRUCCIONES DEL MICROPROCESADOR 6502

SET DE INSTRUCCIONES DEL MICROPROCESADOR 6502

INSTRUCCIÓN	MNEMOTECNICO	BANDERAS AFECTADAS					
		N	Z	C	I	D	V
Sumar memoria al acumulador con acarreo.	ADC	*	*	*			*
"AND" memoria y acumulador.	AND	*	*				*
Desplazar a la izquierda 1 bit (M o ACC).	ASL	*	*	*			
Bifurcar si C=0.	BCC						
Bifurcar si C=1.	BCS						
Bifurcar en resultado cero.	BEQ						
Prueba de bits memoria y acumulador.	BIT	M ₇					M ₁
Bifurcar en resultado negativo.	BMI						
Bifurcar en resultado no cero.	BNE						
Bifurcar en resultado positivo.	BPL						
Forzar Ruptura.	BRK				1		
Bifurcar en sobreflujo cero.	BVC						
Bifurcar en sobreflujo 1.	BVS						
Limpiar acarreo.	CLC			0			
Limpiar modo decimal.	CLD					0	
Limpiar bit de habilitación de interrupción.	CLI				0		
Limpiar Sobreflujo.	CLV						0
Comparar Memoria y Acumulador.	CMP	*	*	*			
Comparar Memoria e índice X.	CPX	*	*	*			
Comparar memoria e índice Y.	CPY	*	*	*			
Decrementar memoria.	DEC	*	*				
Decrementar índice X.	DEX	*	*				
Decrementar índice Y.	DEY	*	*				
"OR- Exclusiva" Memoria con acumulador.	EOR	*	*				
Incrementar Memoria.	INC	*	*				
Incrementar índice X.	INX	*	*				
Incrementar índice Y.	INY	*	*				
Salto a nueva localidad.	JMP						
Salto a subrutina.	JSR						
Cargar acumulador con memoria.	LDA	*	*				
Cargar índice X con memoria.	LDX	*	*				
Cargar índice Y con memoria.	LDY	*	*				
Desplazar a la derecha un bit (M o ACC)	LSR	0	*	*			
No operación	NOP						
"OR" Memoria con Acumulador.	ORA	*	*				
Empujar Acumulador a la pila.	PHA						
Empujar PS a la pila.	PHP						
Halar acumulador de la pila.	PLA	*	*				
Halar PS de la pila.	PLP						De la Pila.
Rotar un bit a la izquierda (M o ACC).	ROL	*	*	*			
Rotar un bit a la derecha (M o ACC)	ROR	*	*	*			
Retorno de interrupciones.	RTI						De la pila
Retorno de subrutinas.	RTS						
Restar memoria de acumulador con prestamo.	SBC	*	*	*			*
Fijar bandera de acarreo.	SEC			1			
Fijar modo decimal.	SED					1	
Fijar bandera de deshabilitación de interrupción.	SEI				1		
Almacenar acumulador en memoria.	STA						
Almacenar índice X en memoria.	STX						
Almacenar índice Y en memoria.	STY						
Transferir acumulador al índice X.	TAX	*	*				
Transferir acumulador al índice Y.	TAY	*	*				
Transferir SP al índice X.	TSX	*	*				
Transferir índice X al acumulador.	TXA	*	*				
Transferir índice Y al SP.	TXS	*	*				
Transferir índice Y al ACC.	TYA						

TABLA B 1 SECUENCIA ALFABÉTICA DE LAS INSTRUCCIONES DEL 6502.

SET DE INSTRUCCIONES DEL MICROPROCESADOR 6502

INSTRUCCION	MODO DE DIRECCIONAMIENTO										
	IMPLICITO	INMEDIATO	PAG. CERO	PAG. CERO X	ABSOLUTO	ABSOLUTO X	ABSOLUTO Y	INDIRECTO X	INDIRECTO Y	INDIRECTO	RELATIVO
ADC		69	65	75	6D	7D	79	61	71		
AND		29	25	35	2D	3D	39	21	31		
ASL	0A		06	16	0E	1E					
BCC											90
BCS											B0
BEQ											F0
BIT			24		2C						
BMI											30
BNE											D0
BPL											10
BRK	00										
BVC											50
BVS											70
CLC	18										
CLD	D8										
CLI	58										
CLV	B8										
CMP		C9	C5	D5	CD	DD	D9	C1	D1		
CPX		E0	E4		EC						
CPY		C0	C4	CC							
DEC			C6	D6	CE	DE					
DEX	CA										
DEY	88										
EOR		49	45	55	4D	5D	59	41	51		
INC			E6	F6	EE	FE					
INX	E8										
INY	C8										
JMP					4C					6C	
JSR					20						
LDA		A9	A5	B5	AD	BD	B9	A1	B1		
LDX		A2	A6	B6	AE		BE				
LDY		A0	A4	BA	AC	BC					
LSR	4A		46	56	4E	5E					
NOP	EA										
ORA		09	05	15	0D	1D	19	01	11		
PHA		48									
PHP	08										
PLA	68										
PLP	28										
ROL	2A		26	36	2E	3E					
ROR	6A		66	76	6E	7E					
RTI	40										
RTS	60		E5	F5	ED	FD	F9	E1	F1		
SBC	E9										
SEC	38										
SED	F8										
SEI	78										
STA			85	95	8D	9D	99	81	91		
STX			86	96	8E						
STY			84	94	8C						
TAX	AA										
TAY	A8										
TSX	98										
TXA	BA										
TXS	8A										
TYA	9A										

TABLA B.2 CODIGOS DE OPERACIÓN DEL 6502.

APENDICE C

**DECODIFICADOR
SN74LS138**

SN54LS138, SN54S138, SN74LS138, SN74S138A 3-LINE TO 8-LINE DECODERS/DEMULTIPLEXERS

SDLS014

DECEMBER 1972 - REVISED MARCH 1988

- Designed Specifically for High-Speed: Memory Decoders Data Transmission Systems
- 3 Enable Inputs to Simplify Cascading and/or Data Reception
- Schottky-Clamped for High Performance

description

These Schottky-clamped TTL MSI circuits are designed to be used in high-performance memory decoding or data-routing applications requiring very short propagation delay times. In high-performance memory systems, these decoders can be used to minimize the effects of system decoding. When employed with high-speed memories utilizing a fast enable circuit, the delay times of these decoders and the enable time of the memory are usually less than the typical access time of the memory. This means that the effective system delay introduced by the Schottky-clamped system decoder is negligible.

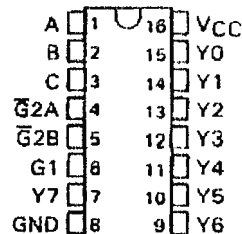
The 'LS138, SN54S138, and SN74S138A decode one of eight lines dependent on the conditions at the three binary select inputs and the three enable inputs. Two active-low and one active-high enable inputs reduce the need for external gates or inverters when expanding. A 24-line decoder can be implemented without external inverters and a 32-line decoder requires only one inverter. An enable input can be used as a data input for demultiplexing applications.

All of these decoder/demultiplexers feature fully buffered inputs, each of which represents only one normalized load to its driving circuit. All inputs are clamped with high-performance Schottky diodes to suppress line-ringing and to simplify system design.

The SN54LS138 and SN54S138 are characterized for operation over the full military temperature range of -55°C to 125°C . The SN74LS138 and SN74S138A are characterized for operation from 0°C to 70°C .

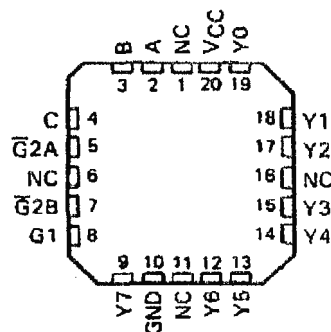
SN54LS138, SN54S138 . . . J OR W PACKAGE
SN74LS138, SN74S138A . . . D OR N PACKAGE

(TOP VIEW)



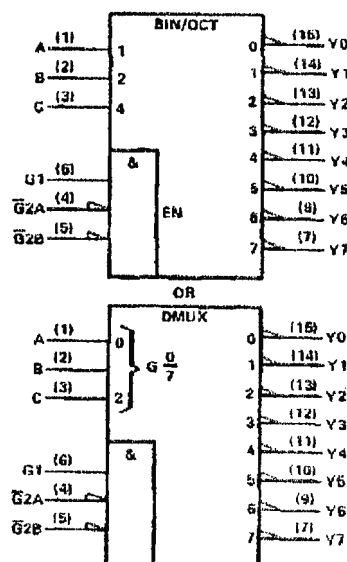
SN54LS138, SN54S138 . . . FK PACKAGE

(TOP VIEW)



NC—No internal connection

logic symbols†



†These symbols are in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.

Pin numbers shown are for D, J, N, and W packages.

PRODUCTION DATA documents contain information current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

TEXAS
INSTRUMENTS

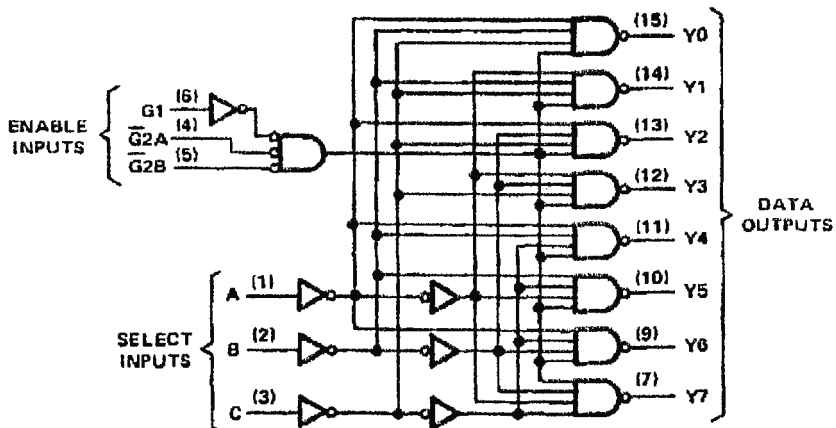
POST OFFICE BOX 655012 • DALLAS, TEXAS 75265

Copyright © 1972, Texas Instruments Incorporated

SN54LS138, SN54S138, SN74LS138, SN74S138A
3-LINE-TO 8-LINE DECODERS/DEMULTIPLEXERS

logic diagram and function table

'LS138, SN54S138, SN74S138A



Pin numbers shown are for D, J, N, and W packages.

'LS138, SN54S138, SN74S138A
FUNCTION TABLE

INPUTS					OUTPUTS							
ENABLE		SELECT										
G1	G2*	C	B	A	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
X	H	X	X	X	H	H	H	H	H	H	H	H
L	X	X	X	X	H	H	H	H	H	H	H	H
H	L	L	L	L	L	H	H	H	H	H	H	H
H	L	L	L	H	H	L	H	H	H	H	H	H
H	L	L	H	L	H	H	L	H	H	H	H	H
H	L	L	H	H	H	H	L	H	H	H	H	H
H	L	H	L	L	H	H	H	L	H	H	H	H
H	L	H	L	H	H	H	H	L	H	L	H	H
H	L	H	H	L	H	H	H	H	H	L	H	H
H	L	H	H	H	H	H	H	H	H	H	L	L

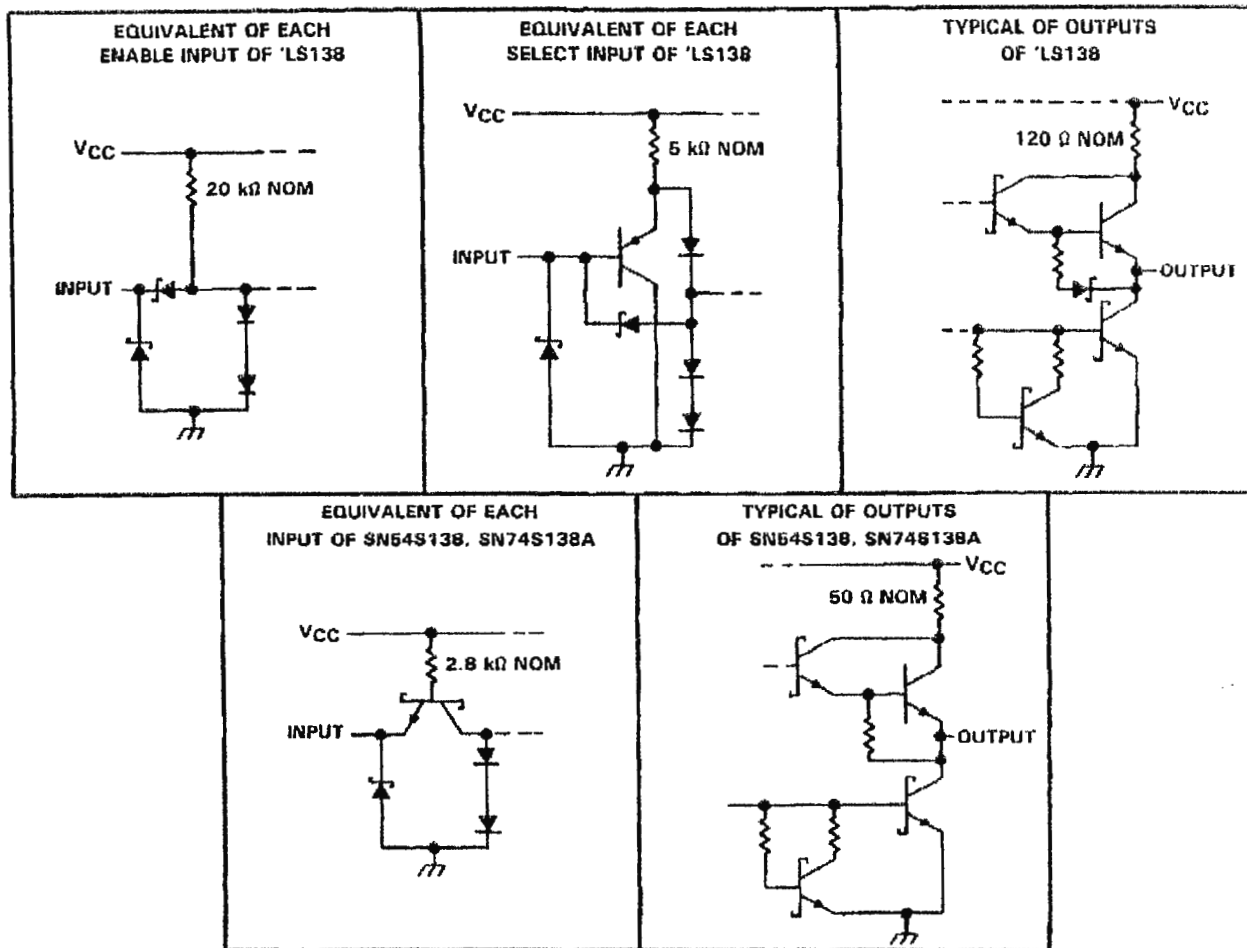
* $\overline{G2} = \overline{G2A} + \overline{G2B}$

H = high level, L = low level, X = irrelevant



SN54LS138, SN54S138, SN74LS138, SN74S138A
3-LINE TO 8-LINE DECODERS/DEMULTIPLEXERS

schematics of inputs and outputs



absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

Supply voltage, VCC (see Note 1)	7 V
Input voltage	7 V
Operating free-air temperature range: SN54LS138, SN54S138	-55°C to 125°C
SN74LS138, SN74S138A	0°C to 70°C
Storage temperature range	-65°C to 150°C

NOTE 1: Voltage values are with respect to network ground terminal.

SN54LS138, SN74LS138 3-LINE TO 8-LINE DECODERS/DEMULTIPLEXERS

recommended operating conditions

		SN54LS138			SN74LS138			UNIT
		MIN	NOM	MAX	MIN	NOM	MAX	
V _{CC}	Supply voltage	4.6	5	6.5	4.75	5	6.25	V
V _{IH}	High-level input voltage	2			2			V
V _{IL}	Low-level input voltage			0.7			0.8	V
I _{OH}	High-level output current			-0.4			-0.4	mA
I _{OL}	Low-level output current			4			8	mA
T _A	Operating free-air temperature	-55		125	0		70	°C

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER	TEST CONDITIONS†	SN54LS138			SN74LS138			UNIT	
		MIN	TYP‡	MAX	MIN	TYP‡	MAX		
V _{IK}	V _{CC} = MIN, I _I = -18 mA			-1.5			-1.5	V	
V _{OH}	V _{CC} = MIN, V _{IH} = 2 V, V _{IL} = MAX, I _{OH} = -0.4 mA	2.5	3.4		2.7	3.4		V	
V _{OL}	V _{CC} = MIN, V _{IH} = 2 V, V _{IL} = MAX			0.25	0.4		0.25	0.4	V
I _I	V _{CC} = MAX, V _I = 7 V			0.1			0.1	mA	
I _{IH}	V _{CC} = MAX, V _I = 2.7 V			20			20	μA	
I _{IL}	V _{CC} = MAX, V _I = 0.4 V			-0.4			-0.4	mA	
				-0.2			-0.2	mA	
I _{OS} §	V _{CC} = MAX	-20		100	-20		-100	mA	
I _{CC}	V _{CC} = MAX, Outputs enabled and open		6.3	10		6.3	10	mA	

†For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.

‡All typical values are at V_{CC} = 5 V, T_A = 25°C.

§Not more than one output should be shorted at a time, and duration of the short-circuit test should not exceed one second.

switching characteristics, V_{CC} = 5 V, T_A = 25°C

PARAMETER†	FROM (INPUT)	TO (OUTPUT)	LEVELS OF DELAY	TEST CONDITIONS	SN54LS138 SN74LS138			UNIT
					MIN	TYP	MAX	
t _{PLH}	Binary Select	Any	2	R _L = 2 kΩ, C _L = 15 pF, See Note 2		11	20	ns
t _{PHL}						18	41	ns
t _{PLH}			3			21	27	ns
t _{PHL}						20	39	ns
t _{PLH}	Enable	Any	2			12	18	ns
t _{PHL}						20	32	ns
t _{PLH}			3			14	26	ns
t _{PHL}						13	38	ns

†t_{PLH} = propagation delay time, low-to-high-level output

t_{PHL} = propagation delay time, high-to-low-level output

NOTE 2: Load circuits and voltage waveforms are shown in Section 1.

SN54S138, SN74S138A 3-LINE TO 8-LINE DECODERS/DEMULTIPLEXERS

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

Supply voltage, V_{CC} (see Note 1)	7 V
Input voltage	5.5 V
Operating free-air temperature range: SN54S138	-55°C to 125°C
SN74S138A	0°C to 70°C
Storage temperature range	65°C to 160°C

NOTE 1: Voltage values are with respect to network ground terminal.

recommended operating conditions

		SN54S138			SN74S138A			UNIT
		MIN	NOM	MAX	MIN	NOM	MAX	
V_{CC}	Supply voltage	4.5	5	5.5	4.75	5	6.25	V
V_{IH}	High-level input voltage	2			2			V
V_{IL}	Low-level input voltage				0.8			V
I_{OH}	High-level output current				-1			mA
I_{OL}	Low-level output current				20			mA
T_A	Operating free-air temperature	-65			125			°C

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER	TEST CONDITIONS†	SN54S138 SN74S138A		UNIT		
		MIN	TYP‡		MAX	
		V_{IK}	$V_{CC} = \text{MIN.}$, $I_I = -18 \text{ mA}$		-1.2	V
V_{OH}	$V_{CC} = \text{MIN.}$, $V_{IH} = 2 \text{ V.}$ $V_{IL} = 0.8 \text{ V.}$ $I_{OH} = -1 \text{ mA}$	SN54S'	2.5	3.4	V	
		SN74S'	2.7	3.4		
V_{OL}	$V_{CC} = \text{MIN.}$, $V_{IH} = 2 \text{ V.}$ $V_{IL} = 0.8 \text{ V.}$ $I_{OL} = 20 \text{ mA}$			0.6	V	
I_I	$V_{CC} = \text{MAX.}$, $V_I = 5.5 \text{ V}$			1	mA	
I_{IH}	$V_{CC} = \text{MAX.}$, $V_I = 2.7 \text{ V}$			50	µA	
I_{IL}	$V_{CC} = \text{MAX.}$, $V_I = 0.5 \text{ V}$			-2	mA	
$I_{OS}^§$	$V_{CC} = \text{MAX.}$			-40	-100	mA
I_{CC}	$V_{CC} = \text{MAX.}$, Outputs enabled and open			49	74	mA

† For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.

‡ All typical values are at $V_{CC} = 5 \text{ V.}$ $T_A = 25^\circ\text{C.}$

§ Not more than one output should be shorted at a time, and duration of the short circuit test should not exceed one second.


**TEXAS
INSTRUMENTS**

POST OFFICE BOX 655012 • DALLAS, TEXAS 75265

SN54S138, SN74S138A 3-LINE TO 8-LINE DECODERS/DEMULTIPLEXERS

switching characteristics, $V_{CC} = 5\text{ V}$, $T_A = 25^\circ\text{C}$

PARAMETER [†]	FROM (INPUT)	TO (OUTPUT)	LEVELS OF DELAY	TEST CONDITIONS	SN54S138 SN74S138A			UNIT
					MIN	TYP	MAX	
t _{PLH}	Binary Select	Any	2	R _L = 280 Ω, C _L = 15 pF. See Note 2		4.5	7	ns
t _{PHL}						7	10.5	ns
t _{PLH}			3			7.5	12	ns
t _{PHL}						8	12	ns
t _{PLH}	Enable	Any	2			5	8	ns
t _{PHL}						7	11	ns
t _{PLH}			3			7	11	ns
t _{PHL}						7	11	ns

[†]t_{PLH} = propagation delay time, low-to-high-level output

t_{PHL} = propagation delay time, high-to-low-level output

NOTE 2: Load circuits and voltage waveforms are shown in Section 1.

TEXAS
INSTRUMENTS

POST OFFICE BOX 655012 • DALLAS, TEXAS 75205

APR 1988

**BUFFER
SN74LS244**

**SN54LS240, SN54LS241, SN54LS244, SN54S240, SN54S241, SN54S244,
SN74LS240, SN74LS241, SN74LS244, SN74S240, SN74S241, SN74S244**
OCTAL BUFFERS AND LINE DRIVERS WITH 3-STATE OUTPUTS

SDLS144

APRIL 1985 - REVISED MARCH 1983

- 3-State Outputs Drive Bus Lines or Buffer Memory Address Registers
- PNP Inputs Reduce D-C Loading
- Hysteresis at Inputs Improves Noise Margins

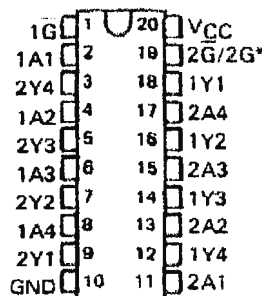
description

These octal buffers and line drivers are designed specifically to improve both the performance and density of three-state memory address drivers, clock drivers, and bus-oriented receivers and transmitters. The designer has a choice of selected combinations of inverting and noninverting outputs, symmetrical \bar{G} (active-low output control) inputs, and complementary G and \bar{G} inputs. These devices feature high fan-out, improved fan-in, and 400-mV noise-margin. The SN74LS' and SN74S' can be used to drive terminated lines down to 133 ohms.

The SN54' family is characterized for operation over the full military temperature range of -55°C to 125°C . The SN74' family is characterized for operation from 0°C to 70°C .

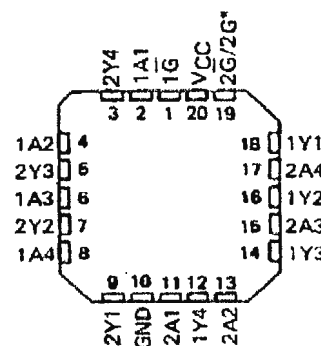
SN54LS', SN64S' ... J OR W PACKAGE
SN74LS', SN74S' ... DW OR N PACKAGE

(TOP VIEW)



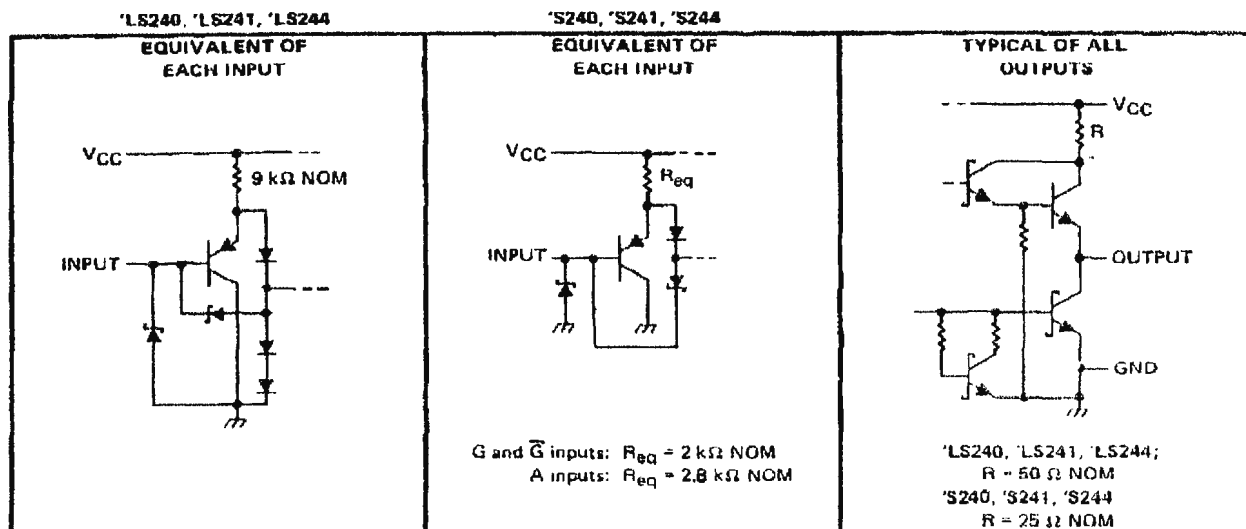
SN54LS', SN64S' ... FK PACKAGE

(TOP VIEW)



*2G for 'LS241 and 'S241 or 2G for all other drivers.

schematics of inputs and outputs



PRODUCTION DATA documents contain information current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.



POST OFFICE BOX 655012 • DALLAS, TEXAS 75265

**SN54S240, SN54S241, SN54S244, SN74S240, SN74S241, SN74S244,
OCTAL BUFFERS AND LINE DRIVERS WITH 3-STATE OUTPUTS**

switching characteristics, $V_{CC} = 5\text{ V}$, $T_A = 25^\circ\text{C}$

PARAMETER	TEST CONDITIONS	'S240			'S241, 'S244			UNIT
		MIN	TYP	MAX	MIN	TYP	MAX	
t_{PLH}	$R_L = 90\ \Omega$, See Note 4	$C_L = 50\text{ pF}$	4.5	7	6	9	ns	
t_{PHL}			4.5	7	6	9	ns	
t_{PZL}			10	16	10	15	ns	
t_{PZH}	$R_L = 90\ \Omega$, See Note 4	$C_L = 5\text{ pF}$	6.5	10	8	12	ns	
t_{PLZ}			10	15	10	15	ns	
t_{PHZ}			6	9	6	9	ns	

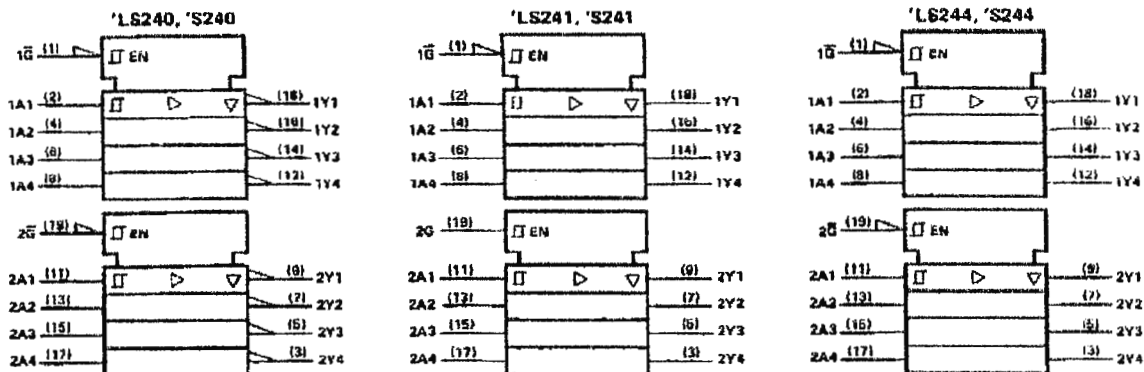
NOTE 4: Load circuits and voltage waveforms are shown in Section 1.


**TEXAS
INSTRUMENTS**

POST OFFICE BOX 655012 • DALLAS, TEXAS 75265

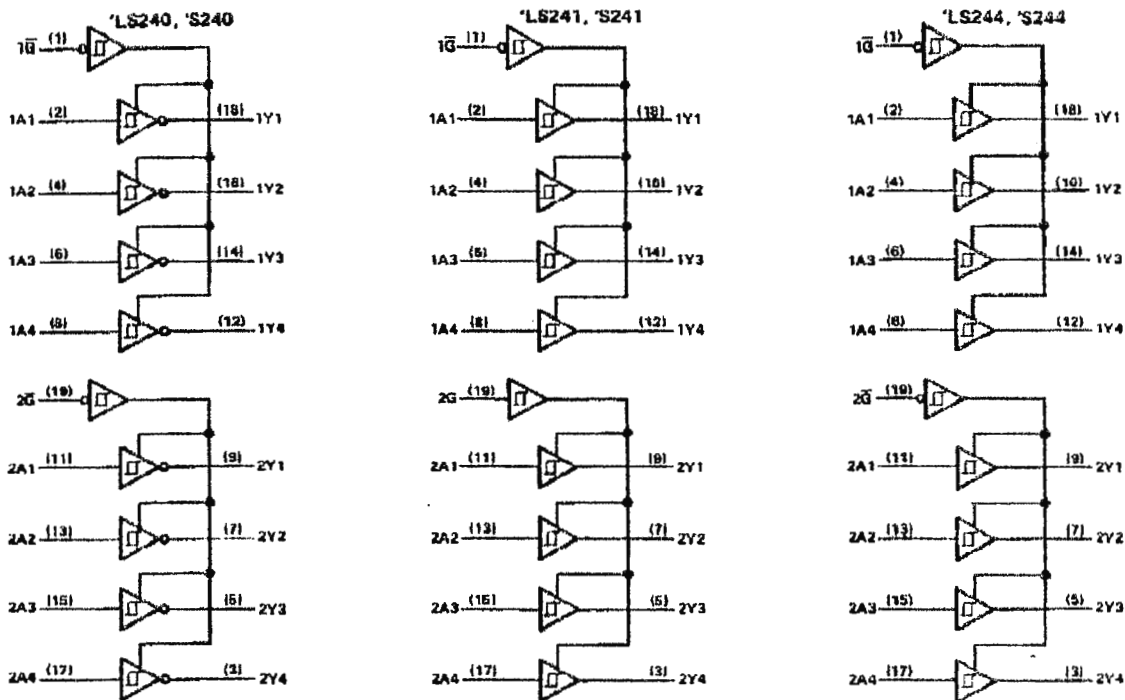
**SN54LS240, SN54LS241, SN54LS244, SN54S240, SN54S241, SN54S244,
SN74LS240, SN74LS241, SN74LS244, SN74S240, SN74S241, SN74S244**
OCTAL BUFFERS AND LINE DRIVERS WITH 3-STATE OUTPUTS

logic symbols†



†These symbols are in accordance with ANSI/IEEE Std. 91-1984 and IEC Publication 817-12.

logic diagrams (positive logic)



Pin numbers shown are for DW, J, N, and W packages.

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

Supply voltage, VCC (see Note 1)	7 V
Input voltage: 'LS Circuits	7 V
'S Circuits	5.5 V
Off-state output voltage	5.5 V
Operating free-air temperature range: SN54LS', SN54S' Circuits	-55° C to 125° C
SN74LS', SN74S' Circuits	0° C to 70° C
Storage temperature range	-65° C to 150° C

NOTE 1: Voltage values are with respect to network ground terminal.

SN54LS240, SN54LS241, SN54LS244, SN74LS240, SN74LS241, SN74LS244 OCTAL BUFFERS AND LINE DRIVERS WITH 3-STATE OUTPUTS

recommended operating conditions

PARAMETER	SN54LS*			SN74LS*			UNITY
	MIN	NOM	MAX	MIN	NOM	MAX	
V _{CC} Supply voltage (see Note 1)	4.5	5	5.5	4.75	5	5.25	V
V _{IH} High-level input voltage	2			2			V
V _{IL} Low-level input voltage			0.7			0.8	V
I _{OH} High-level output current			-12			-15	mA
I _{OL} Low-level output current			12			24	mA
T _A Operating free-air temperature	-65		125	0		70	°C

NOTE 1: Voltage values are with respect to network ground terminal.

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER	TEST CONDITIONS†	SN54LS*		SN74LS*		UNITY		
		MIN	TYP‡	MAX	MIN		TYP‡	MAX
V _{IK}	V _{CC} = MIN, I _I = -18 mA			-1.5		-1.5	V	
Hysteresis (V _{T+} - V _{T-})	V _{CC} = MIN	0.2	0.4		0.2	0.4	V	
V _{OH}	V _{CC} = MIN, V _{IH} = 2 V, V _{IL} = MAX, I _{OH} = -3 mA	2.4	3.4		2.4	3.4	V	
	V _{CC} = MIN, V _{IH} = 2 V, V _{IL} = 0.5 V, I _{OH} = MAX	2			2			
V _{OL}	V _{CC} = MIN, V _{IH} = 2 V, V _{IL} = MAX, I _{OL} = 12 mA			0.4		0.4	V	
	V _{CC} = MIN, V _{IH} = 2 V, V _{IL} = MAX, I _{OL} = 24 mA					0.5		
I _{OZH}	V _{CC} = MAX, V _{IH} = 2 V, V _{IL} = MAX			20		20	μA	
I _{OZL}	V _{IL} = MAX, V _O = 0.4 V			-20		-20	μA	
I _I	V _{CC} = MAX, V _I = 7 V			0.1		0.1	mA	
I _{IH}	V _{CC} = MAX, V _I = 2.7 V			20		20	μA	
I _{IL}	V _{CC} = MAX, V _{IL} = 0.4 V			-0.2		-0.2	mA	
I _{OS} §	V _{CC} = MAX	-40		-225	-40	-225	mA	
I _{CC}	V _{CC} = MAX, Output open	Outputs high	All	17	27	17	27	mA
		Outputs low	'LS240	26	44	26	44	
			'LS241, 'LS244	27	46	27	46	
		All outputs disabled	'LS240	29	50	29	50	
				32	54	32	54	

† For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.

‡ All typical values are at V_{CC} = 5 V, T_A = 25°C.

§ Not more than one output should be shorted at a time, and duration of the short-circuit should not exceed one second.

switching characteristics, V_{CC} = 5 V, T_A = 25°C

PARAMETER	TEST CONDITIONS	'LS240			'LS241, 'LS244			UNITY
		MIN	TYP	MAX	MIN	TYP	MAX	
t _{PLH}	R _L = 667 Ω, C _L = 45 pF, See Note 2		9	14		12	18	ns
t _{PHL}			12	18		12	18	ns
t _{PZL}			20	30		20	30	ns
t _{PZH}			15	23		15	23	ns
t _{PLZ}	R _L = 667 Ω, C _L = 5 pF, See Note 2		10	20		10	20	ns
t _{PHZ}			15	25		15	25	ns

NOTE 2: Load circuits and voltage waveforms are shown in Section 1.


TEXAS
INSTRUMENTS

POST OFFICE BOX 655012 • DALLAS, TEXAS 75265

SN54S240, SN54S241, SN54S244, SN74S240, SN74S241, SN74S244, OCTAL BUFFERS AND LINE DRIVERS WITH 3-STATE OUTPUTS

recommended operating conditions

PARAMETER	SN54S'			SN74S'			UNIT
	MIN	NOM	MAX	MIN	NOM	MAX	
V _{CC} Supply voltage, (see Note 1)	4.5	5	5.5	4.75	5	5.25	V
V _{IH} High-level input voltage	2			2			V
V _{IL} Low-level input voltage			0.8			0.8	V
I _{OH} High-level output current			-12			-15	mA
I _{OL} Low-level output current			48			64	mA
External resistance between any input and V _{CC} or ground			40			40	k Ω
T _A Operating free-air temperature (see Note 3)	-55		125	0		70	$^{\circ}$ C

NOTES: 1. Voltage values are with respect to network ground terminal.

3. An SN54S241J operating at free-air temperature above 116 $^{\circ}$ C requires a heat sink that provides a thermal resistance from case to free-air R_{θCA} of not more than 40 $^{\circ}$ C/W.

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER	TEST CONDITIONS†		SN54S'		SN74S'		UNIT	
			MIN	TYP‡	MAX	MIN		TYP‡
V _{IK}	V _{CC} = MIN, I _I = -18 mA				-1.2	-1.2	V	
Hysteresis (V _{T+} - V _{T-})	V _{CC} = MIN		0.2	0.4		0.2 0.4	V	
V _{OH}	V _{CC} = MIN, V _{IH} = 2 V, V _{IL} = 0.8 V, I _{OH} = -1 mA					2.7	V	
	V _{CC} = MIN, V _{IH} = 2 V, V _{IL} = 0.8 V, I _{OH} = -3 mA		2.4	3.4		2.4 3.4		
	V _{CC} = MIN, V _{IH} = 2 V, V _{IL} = 0.5 V, I _{OH} = MAX		2			2		
V _{OL}	V _{CC} = MIN, V _{IH} = 2 V, V _{IL} = 0.8 V, I _{OL} = MAX				0.55	0.55	V	
I _{OZH}	V _{CC} = MAX, V _{IH} = 2 V, V _O = 2.4 V				50	50	μ A	
I _{OZL}	V _{IL} = 0.8 V, V _O = 0.5 V				-50	-50		
I _I	V _{CC} = MAX, V _I = 5.5 V				1	1	mA	
I _{IH}	V _{CC} = MAX, V _I = 2.7 V				50	50	μ A	
I _{IL}	Any A	V _{CC} = MAX, V _I = 0.5 V			-400	-400	μ A	
	Any G				-2	-2		
I _{OS} §	V _{CC} = MAX		-50		-225	-50 -225	μ A	
I _{CC}	Outputs high	V _{CC} = MAX, Outputs open	'S240	80	123	80	135	mA
			'S241, 'S244	95	147	95	160	
	Outputs low		'S240	100	145	100	150	
			'S241, 'S244	120	170	120	180	
	Outputs disabled		'S240	100	145	100	150	
			'S241, 'S244	120	170	120	180	

† For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.

‡ All typical values are at V_{CC} = 5 V, T_A = 25 $^{\circ}$ C.

§ Not more than one output should be shorted at a time, and duration of the short-circuit should not exceed one second.

TEXAS
INSTRUMENTS

POST OFFICE BOX 655012 • DALLAS, TEXAS 75265

APENDICE E

**DISPLAY
HDSP-2111**

Eight Character 5 mm and 7 mm Smart Alphanumeric Displays

Technical Data

HDSP-210X Series
HDSP-211X Series
HDSP-250X Series

Features

- X Stackable (HDSP-21XX)
- XY Stackable (HDSP-250X)
- 128 Character ASCII Decoder
- Programmable Functions
- 16 User Definable Characters
- Multi-Level Dimming and Blanking
- TTL Compatible CMOS IC
- Wave Solderable

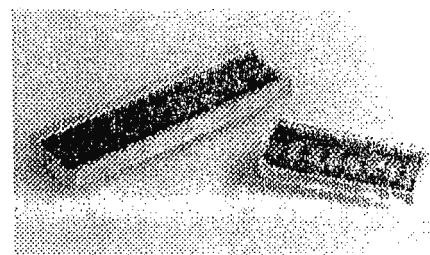
Applications

- Computer Peripherals
- Industrial Instrumentation
- Medical Equipment
- Portable Data Entry Devices
- Cellular Phones

- Telecommunications Equipment
- Test Equipment

Description

The HDSP-210X/-211X/-250X series of products is ideal for applications where displaying eight or more characters of dot matrix information in an aesthetically pleasing manner is required. These devices are 8-digit, 5 x 7 dot matrix, alphanumeric displays and are all packaged in a standard 15.24 mm (0.6 inch) 28 pin DIP. The on-board CMOS IC has the ability to decode 128 ASCII characters which are permanently stored in ROM. In addition, 16 programmable symbols may be stored in on-board ROM, allowing consider-

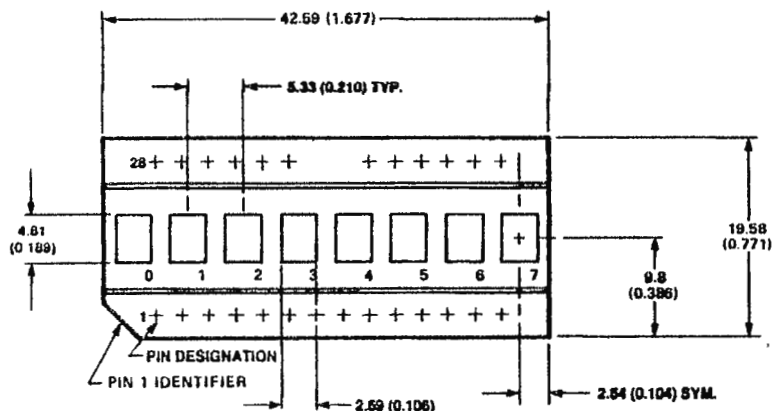


able flexibility for displaying additional symbols and icons. Seven brightness levels provide versatility in adjusting the display intensity and power consumption. The HDSP-210X/-211X/-250X products are designed for standard microprocessor interface techniques. The display and special features are accessed through a bidirectional 8-bit data bus.

Device Selection Guide

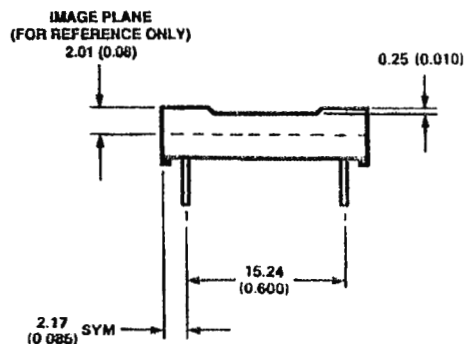
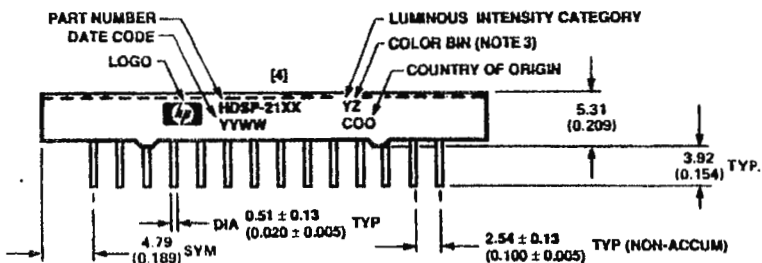
Font Height	AlGaAs Red	High Efficiency Red	Orange	Yellow	Green
0.2 inches	HDSP-2107	HDSP-2112	HDSP-2110	HDSP-2111	HDSP-2113
0.27 inches	-	HDSP-2502	HDSP-2500	HDSP-2501	HDSP-2503

Package Dimensions



Pin Function Assignment Table

Pin No.	Function	Pin No.	Function
1	RST	15	GND(SUPPLY)
2	FL	16	GND (LOWIC)
3	A ₀	17	CE
4	A ₁	18	RD
5	A ₂	19	D ₀
6	A ₃	20	D ₁
7	DO NOT CONNECT	21	NO PIN
8	DO NOT CONNECT	22	NO PIN
9	DO NOT CONNECT	23	D ₂
10	A ₄	24	D ₃
11	CLB	25	D ₄
12	CLK	26	D ₅
13	WR	27	D ₆
14	V _{DD}	28	D ₇



- NOTES:
 1. DIMENSIONS ARE IN mm (INCHES).
 2. UNLESS OTHERWISE SPECIFIED, TOLERANCE ON ALL DIMENSIONS IS ± 0.25 mm (0.010 INCH).
 3. FOR YELLOW AND GREEN DEVICES ONLY.

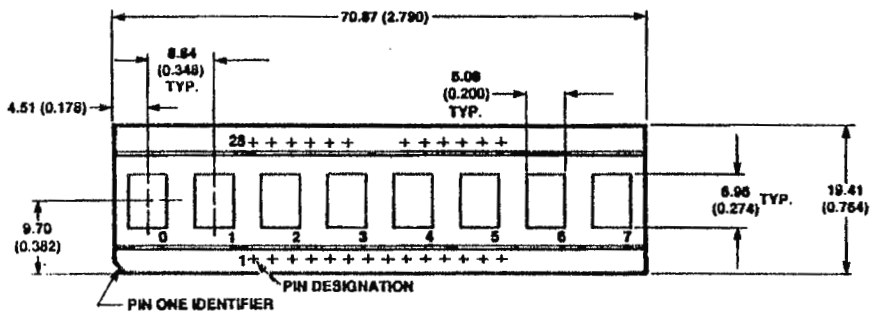
Absolute Maximum Ratings

- Supply Voltage, V_{DD} to Ground^[1] -0.3 to 7.0 V
- Operating Voltage, V_{DD} to Ground^[2] 5.5 V
- Input Voltage, Any Pin to Ground -0.3 to V_{DD} + 0.3 V
- Free Air Operating Temperature Range, T_A^[3] -45°C to +85°C
- Storage Temperature Range, T_S -55°C to +100°C
- Relative Humidity (non-condensing) 85%
- Maximum Solder Temperature
 (Below Seating Plane), t < 5 sec 260°C
- ESD Protection @ 1.5 kΩ, 100 pF V_Z = 4 kV (each pin)

- Notes:
 1. Maximum Voltage is with no LEDs illuminated.
 2. 20 dots ON in all locations at full brightness.
 3. Maximum supply voltage is 5.25 V for operation above 70°C.

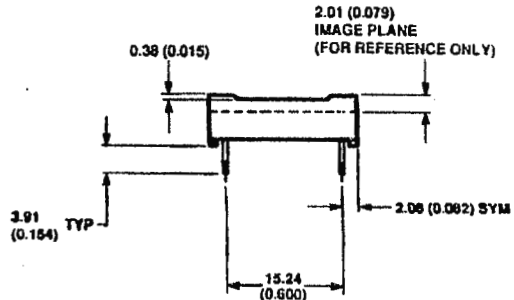
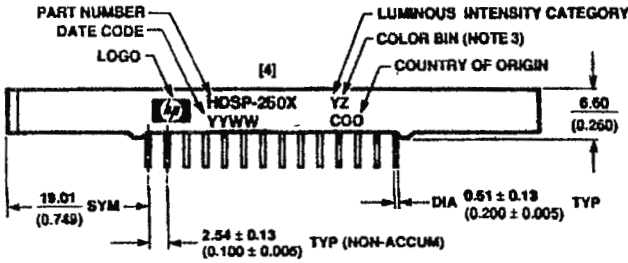
ESD WARNING: STANDARD CMOS HANDLING PRECAUTIONS SHOULD BE OBSERVED TO AVOID STATIC DISCHARGE.

Package Dimensions



Pin Function Assignment Table

Pin No.	Function	Pin No.	Function
1	RST	15	GND(SUPPLY)
2	FL	16	GND (LOGIC)
3	A ₀	17	CE
4	A ₁	18	RD
5	A ₂	19	D ₀
6	A ₃	20	D ₁
7	DO NOT CONNECT	21	NO PIN
8	DO NOT CONNECT	22	NO PIN
9	DO NOT CONNECT	23	D ₂
10	A ₄	24	D ₃
11	CLS	25	D ₄
12	CLK	26	D ₅
13	WR	27	D ₆
14	V _{DD}	28	D ₇



- NOTES:**
1. DIMENSIONS ARE IN mm (INCHES).
 2. UNLESS OTHERWISE SPECIFIED, TOLERANCE ON ALL DIMENSIONS IS ± 0.25 mm (0.010 INCH).
 3. FOR YELLOW AND GREEN DEVICES ONLY.

Electrical Characteristics over Operating Temperature Range (-45°C to +85°C)

4.5 V < V_{DD} < 5.5 V, unless otherwise specified

Parameter	Symbol	T _A = 25°C V _{DD} = 5.0 V		-45°C < T _A < +85°C 4.5 V < V _{DD} < 5.5 V		Units	Test Conditions
		Typ.	Max.	Min.	Max.		
Input Leakage (Input without pullup)	I _{IH} I _{IL}				1.0 -1.0	μA	V _{IN} = 0 to V _{DD} , pins CLK, D ₀ -D ₇ , A ₀ -A ₄
Input Current (Input with pullup)	I _{IPL}	-11	-18		-30	μA	V _{IN} = 0 to V _{DD} , pins <u>CLS</u> , <u>RST</u> , <u>WR</u> , <u>RD</u> , <u>CE</u> , <u>FL</u>
I _{DD} Blank	I _{DD} (BLK)	0.5	3.0		4.0	mA	V _{IN} = V _{DD}
I _{DD} 8 digits 12 dots/character ^[1,2]	I _{DD} (V)	200	255		330	mA	"V" on in all 8 locations
I _{DD} 8 digits 20 dots/character ^[1,2,3,4]	I _{DD} (#)	300	370		430	mA	"#" on in all locations
Input Voltage High	V _{IH}			2.0	V _{DD} +0.3	V	
Input Voltage Low	V _{IL}			GND -0.3 V	0.8	V	
Output Voltage High	V _{OH}			2.4		V	V _{DD} = 4.5 V, I _{OH} = -40 μA
Output Voltage Low D ₀ -D ₇	V _{OL}				0.4	V	V _{DD} = 4.5 V, I _{OL} = 1.6 mA
Output Voltage Low CLK	V _{OL}				0.4	V	V _{DD} = 4.5 V, I _{OL} = 40 μA
High Level Output Current	I _{OH}				-60	mA	V _{DD} = 5.0 V
Low Level Output Current	I _{OL}				50	mA	V _{DD} = 5.0 V
Thermal Resistance IC Junction-to-Case	Rθ _{J-C}	15				°C/W	

Notes:

1. Average I_{DD} measured at full brightness. See Table 2 in Control Word Section for I_{DD} at lower brightness levels. Peak I_{DD} = 28/15 x I_{DD} (#).
2. Maximum I_{DD} occurs at -55°C.
3. Maximum I_{DD}(#) = 355 mA at V_{DD} = 5.25 V and IC T_J = 150°C.
4. Maximum I_{DD}(#) = 375 mA at V_{DD} = 5.5 V and IC T_J = 150°C.

Optical Characteristics at 25°C^[1]

$V_{DD} = 5.0$ V at Full Brightness

Description	Part Number	Luminous Intensity Character Average (#) I_v (mcd)		Peak Wavelength λ_{Peak} (nm)	Dominant Wavelength λ_d (nm)
		Min.	Typ.		
AlGaAs	HDSP-2107	5.0	15.0	645	637
HER	HDSP-2112 -2502	2.5	7.5	635	626
Orange	HDSP-2110 -2500	2.5	7.5	600	602
Yellow	HDSP-2111 -2501	2.5	7.5	583	585
High Performance Green	HDSP-2113 -2503	2.5	7.5	568	574

Note: 1. Refers to the initial case temperature of the device immediately prior to measurement.

AC Timing Characteristics over Temperature Range (-45°C to +85°C)

4.5 V < V_{DD} < 5.5 V, unless otherwise specified

Reference Number	Symbol	Description	Min. ^[1]	Units
1	t_{ACC}	Display Access Time Write Read	210 230	ns
2	t_{ACS}	Address Setup Time to Chip Enable	10	ns
3	t_{CE}	Chip Enable Active Time ^[2,3] Write Read	140 160	ns
4	t_{ACH}	Address Hold Time to Chip Enable	20	ns
5	t_{CER}	Chip Enable Recovery Time	60	ns
6	t_{CES}	Chip Enable Active Prior to Rising Edge of ^[2,3] Write Read	140 160	ns
7	t_{CEH}	Chip Enable Hold Time to Rising Edge of Read/Write Signal ^[2,3]	0	ns
8	t_W	Write Active Time	100	ns
9	t_{WSU}	Data Write Setup Time	50	ns
10	t_{WH}	Data Write Hold Time	20	ns
11	t_R	Chip Enable Active Prior to Valid Data	160	ns
12	t_{RD}	Read Active Prior to Valid Data	75	ns
13	t_{DF}	Read Data Float Delay	10	ns
	t_{RC}	Reset Active Time ^[4]	300	ns

Notes:

- Worst case values occur at an IC junction temperature of 150°C.
- For designers who do not need to read from the display, the Read line can be tied to V_{DD} and the Write and Chip Enable lines can be tied together.
- Changing the logic levels of the Address lines when $\overline{CE} = "0"$ may cause erroneous data to be entered into the Character RAM, regardless of the logic levels of the \overline{WR} and \overline{RD} lines.
- The display must not be accessed until after 3 clock pulses (110 μ s min. using the internal refresh clock) after the rising edge of the reset line.

AC Timing Characteristics over Temperature Range (-45°C to +85°C)

4.5 V < V_{DD} < 5.5 V, unless otherwise specified

Symbol	Description	25°C Typ.	Min. ^[1]	Units
F _{OSC}	Oscillator Frequency	57	28	kHz
F _{RF} ^[2]	Display Refresh Rate	256	128	Hz
F _{FL} ^[3]	Character Flash Rate	2	1	Hz
t _{ST} ^[4]	Self Test Cycle Time	4.6	9.2	sec

Notes:

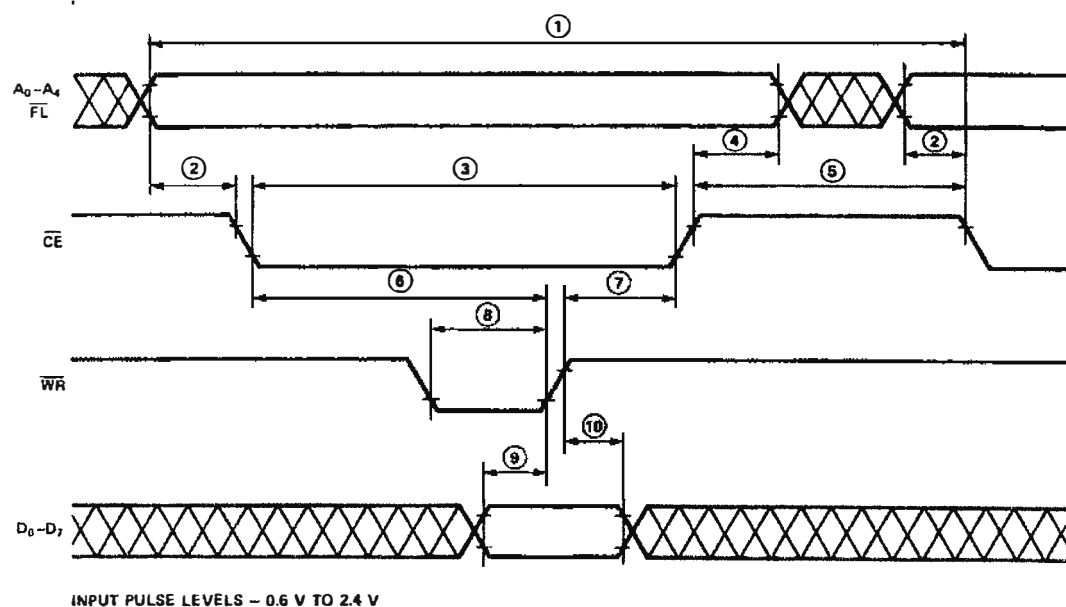
1. Worst case values occur at an IC junction temperature of 150°C.

2. F_{RF} = F_{OSC}/224

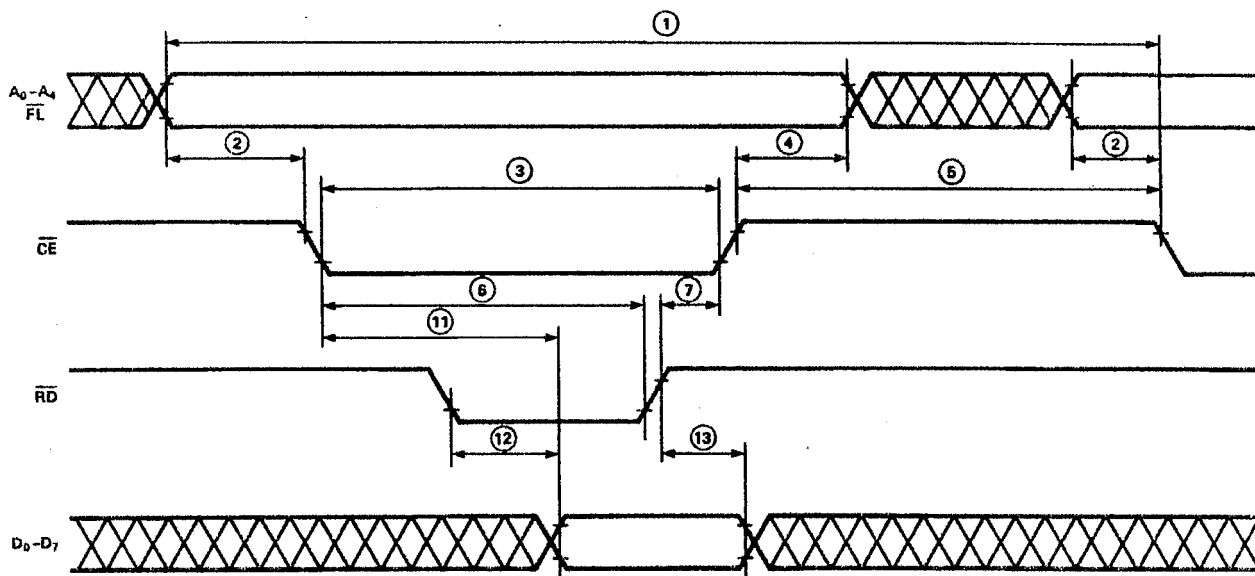
3. F_{FL} = F_{OSC}/28,672

4. t_{ST} = 262,144/F_{OSC}

Write Cycle Timing Diagram

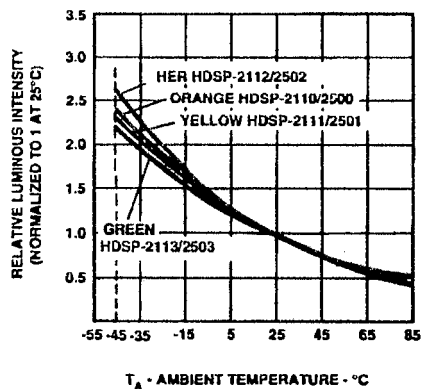


Read Cycle Timing Diagram



INPUT PULSE LEVELS: 0.6 V TO 2.4 V
 OUTPUT REFERENCE LEVELS: 0.8 V TO 2.2 V
 OUTPUT LOADING = 1 TTL LOAD AND 100pFd

Relative Luminous Intensity vs. Temperature



Electrical Description

Pin Function

RESET ($\overline{\text{RST}}$, pin 1)

FLASH ($\overline{\text{FL}}$, pin 2)

ADDRESS INPUTS
(A_0 - A_4 , pins 3-6, 10)

Description

Initializes the display.

$\overline{\text{FL}}$ low indicates an access to the Flash RAM and is unaffected by the state of address lines A_3 - A_4 .

Each location in memory has a distinct address. Address inputs (A_0 - A_2) select a specific location in the Character RAM, the Flash RAM or a particular row in the UDC (User-Defined Character) RAM. A_3 - A_4 are used to select which section of memory is accessed. Table 1 shows the logic levels needed to access each section of memory.

Table 1. Logic Levels to Access Memory

Section of Memory	$\overline{\text{FL}}$	A_4	A_3	A_2 A_1 A_0
Flash RAM	0	X	X	Char. Address
UDC Address Register	1	0	0	Don't Care
UDC RAM	1	0	1	Row Address
Control Word Register	1	1	0	Don't Care
Character RAM	1	1	1	Character Address

CLOCK SELECT
(CLS, pin 11)

Used to select either an internal (CLS = 1) or external (CLS = 0) clock source.

CLOCK INPUT/OUTPUT
(CLK, pin 12)

Outputs the master clock (CLS = 1) or inputs a clock (CLS = 0) for slave displays.

WRITE ($\overline{\text{WR}}$, pin 13)

Data is written into the display when the $\overline{\text{WR}}$ input is low and the $\overline{\text{CE}}$ input is low.

CHIP ENABLE ($\overline{\text{CE}}$, pin 17)

Must be at a logic low to read or write data to the display and must go high between each read and write cycle.

READ ($\overline{\text{RD}}$, pin 18)

Data is read from the display when the $\overline{\text{RD}}$ input is low and the $\overline{\text{CE}}$ input is low.

DATA Bus (D_0 - D_7 ,
pins 19, 20, 23-28)

Used to read from or write to the display.

GND (SUPPLY) (pin 15)

Analog ground for the LED drivers.

GND (LOGIC) (pin 16)

Digital ground for internal logic.

V_{DD} (POWER) (pin 14)

Positive power supply input.

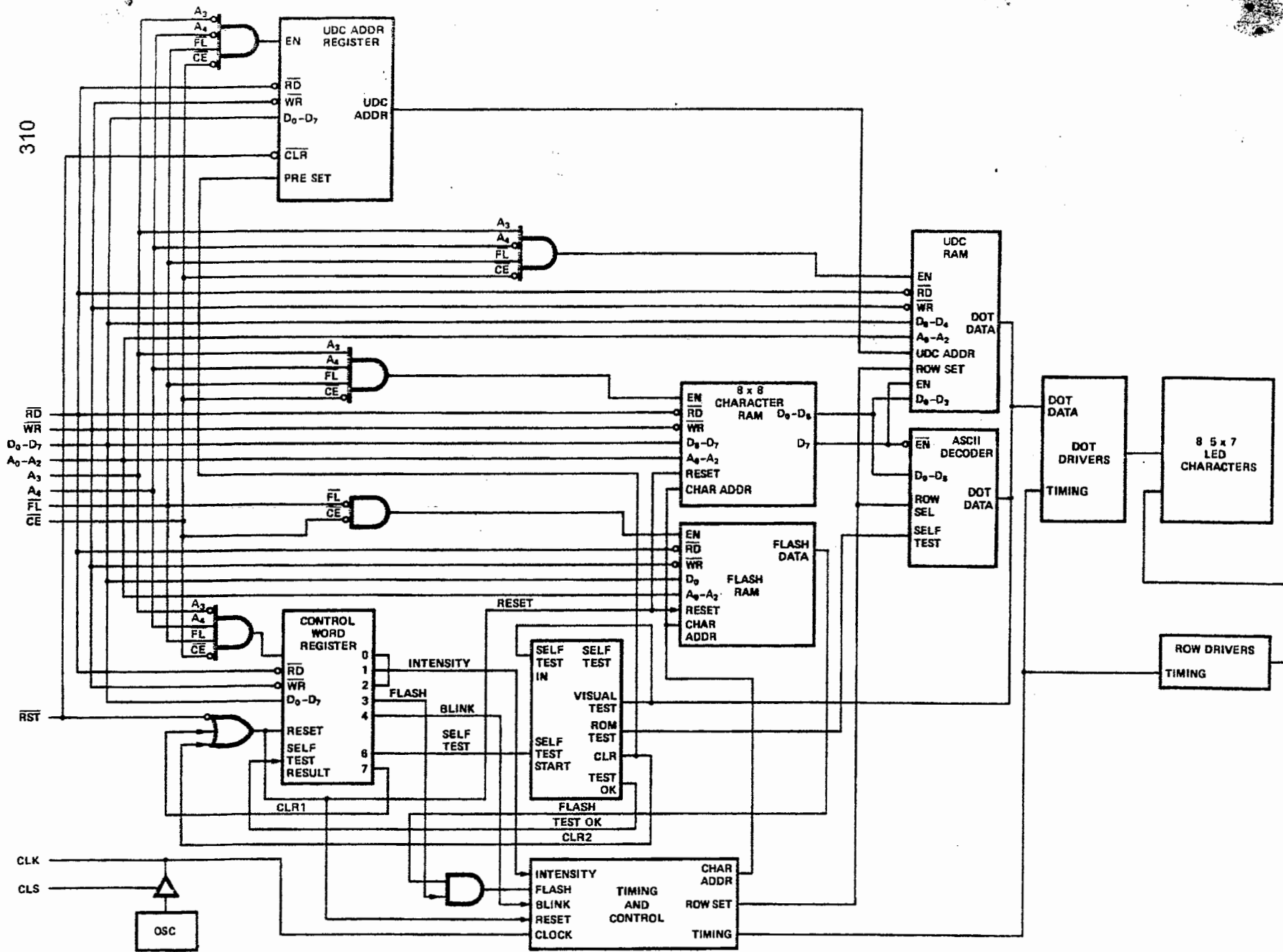


Figure 1. HDSP-210X/-211X/-212X/-250X Internal Block Diagram.

Display Internal Block Diagram

Figure 1 shows the internal block diagram of the HDSP-210X/-211X/-250X displays. The CMOS IC consists of an 8 byte Character

RAM, an 8 bit Flash RAM, a 128 character ASCII decoder, a 16 character UDC RAM, a UDC Address Register, a Control Word Register, and refresh circuitry necessary to synchronize the

decoding and driving of eight 5 x 7 dot matrix characters. The major user-accessible portions of the display are listed below:

Character RAM	This RAM stores either ASCII character data or a UDC RAM address.
Flash RAM	This is a 1 x 8 RAM which stores Flash data.
User-Defined Character RAM (UDC RAM)	This RAM stores the dot pattern for custom characters.
User-Defined Character Address Register (UDC Address Register)	This register is used to provide the address to the UDC RAM when the user is writing or reading a custom character.
Control Word Register	This register allows the user to adjust the display brightness, flash individual characters, blink, self test, or clear the display.

Character Ram

Figure 2 shows the logic levels needed to access the HDSP-210X/-211X/-250X Character RAM. During a normal access, the $\overline{CE} = "0"$ and either $\overline{RD} = "0"$ or $\overline{WR} = "0"$. However, erroneous data may be written into the Character RAM if the address lines are unstable when $\overline{CE} = "0"$ regardless of the logic levels of the \overline{RD} or \overline{WR} lines. Address lines A_0-A_2 are used to select the location in the Character RAM. Two types of data can be stored in each Character RAM location: an ASCII code or a UDC RAM address. Data bit D_7 is used to differentiate between the ASCII character and a UDC RAM address. $D_7 = 0$ enables the ASCII decoder and $D_7 = 1$ enables the UDC RAM. D_0-D_6 are used to input ASCII data and D_0-D_3 are used to input a UDC address.

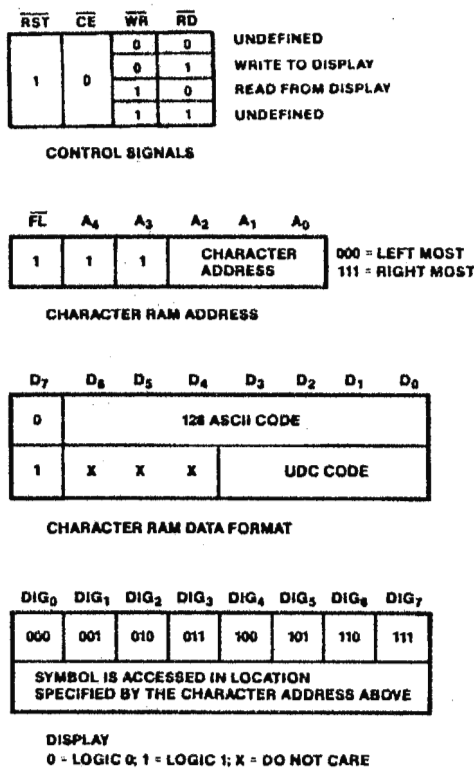


Figure 2. Logic Levels to Access the Character RAM.

UDC RAM and UDC Address Register

Figure 3 shows the logic levels needed to access the UDC RAM and the UDC Address Register. The UDC Address Register is eight bits wide. The lower four bits (D₀-D₃) are used to select one of the 16 UDC locations. The upper four bits (D₄-D₇) are not used. Once the UDC address has been stored in the UDC Address Register, the UDC RAM can be accessed.

To completely specify a 5 x 7 character, eight write cycles are required. One cycle is used to store the UDC RAM address in the UDC Address Register and seven cycles are used to store dot data in the UDC RAM. Data is entered by rows and one cycle is needed to access each row. Figure 4 shows the organization of a UDC character assuming the symbol to be stored is an "F." A₀-A₂ are used to select the row to be accessed and D₀-D₄ are used to transmit the row dot data. The upper three bits (D₅-D₇) are ignored. D₀ (least significant bit) corresponds to the right most column of the 5 x 7 matrix and D₄ (most significant bit) corresponds to the left most column of the 5 x 7 matrix.

Flash RAM

Figure 5 shows the logic levels needed to access the Flash RAM. The Flash RAM has one bit associated with each location of the Character RAM. The Flash input is used to select the Flash RAM while address lines A₃-A₄ are ignored. Address lines A₀-A₂ are used to select the location in the Flash RAM to store the attribute. D₀ is used to store or remove the flash attribute. D₀ = "1" stores the attribute and D₀ = "0" removes the attribute.

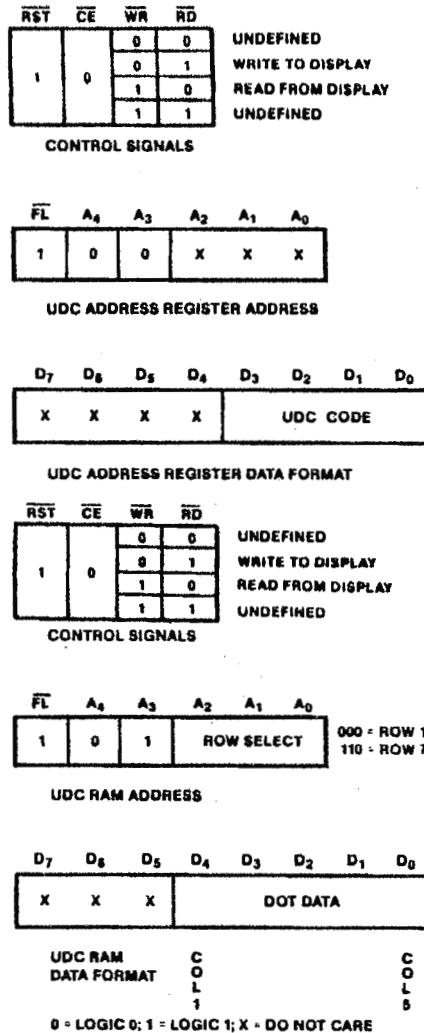


Figure 3. Logic Levels to Access a UDC Character.

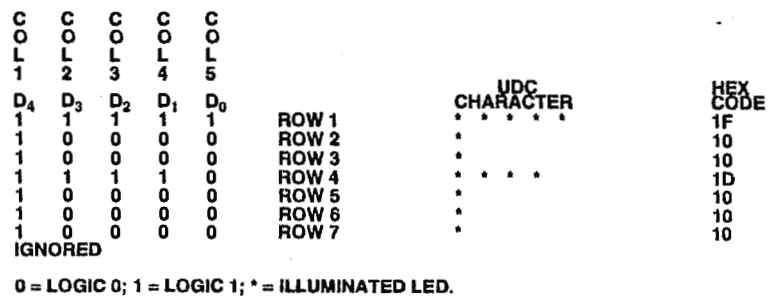


Figure 4. Data to Load "F" into the UDC RAM.

When the attribute is enabled through bit 3 of the Control Word and a "1" is stored in the Flash RAM, the corresponding character will flash at approxi-

mately 2 Hz. The actual rate is dependent on the clock frequency. For an external clock the flash rate can be calculated by dividing the clock frequency by 28,672.

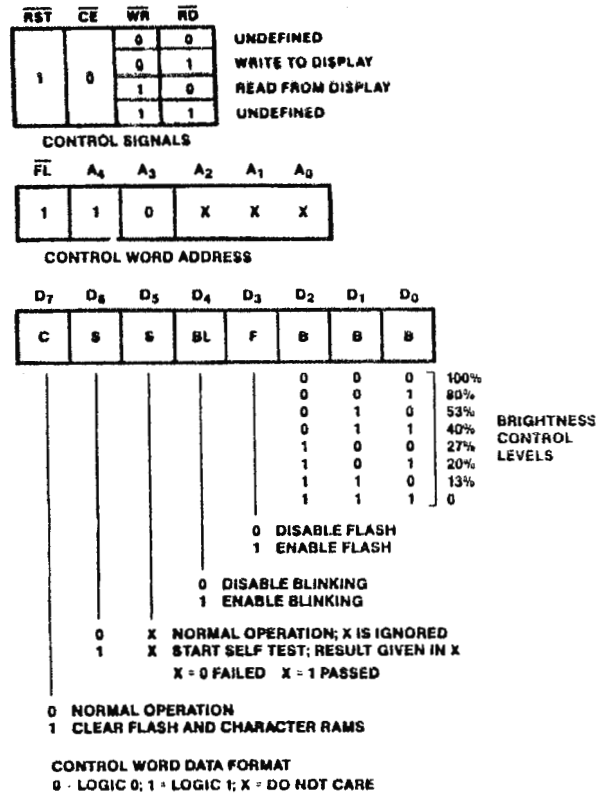
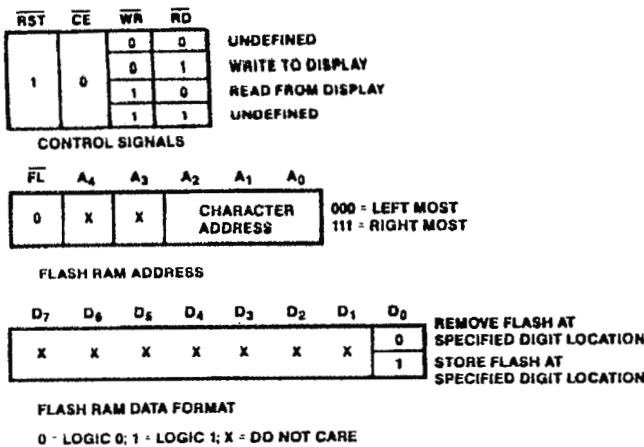


Figure 5. Logic Levels to Access the Flash RAM.

Control Word Register

Figure 6 shows how to access the Control Word Register. This 8-bit register performs five functions: Brightness control, Flash RAM control, Blinking, Self Test, and Clear. Each function is independent of the others; however, all bits are updated during each Control Word write cycle.

Brightness (Bits 0-2)

Bits 0-2 of the Control Word adjust the brightness of the display. Bits 0-2 are interpreted as a three bit binary code with code (000) corresponding to maximum brightness and code (111) corresponding to a blanked display. In addition to varying the display brightness, bits 0-2 also vary the average value of I_{DD} . I_{DD} can be calculated at any brightness level by multiplying the percent brightness level by the value of I_{DD} at the 100% brightness level. These values of I_{DD} are shown in Table 2.

Flash Function (Bit 3)

Bit 3 determines whether the flashing character attribute is on or off. When bit 3 is a "1," the output of the Flash RAM is checked. If the content of a location in the Flash RAM is a "1," the associated digit will flash at

Table 2. Current Requirements at Different Brightness Levels $V_{DD} = 5.0 V$

Symbol	D ₂	D ₁	D ₀	% Brightness	Current at 25°C Typ.	Units
I_{DD} (V)	0	0	0	100	200	mA
	0	0	1	80	160	mA
	0	1	0	53	106	mA
	0	1	1	40	80	mA
	1	0	0	27	54	mA
	1	0	1	20	40	mA
	1	1	0	13	26	mA

approximately 2 Hz. For an external clock, the blink rate can be calculated by driving the clock frequency by 28,672. If the flash enable bit of the Control Word is a "0," the content of the Flash RAM is ignored. To use this function with multiple display systems, see the Display Reset section.

Blink Function (Bit 4)

Bit 4 of the Control Word is used to synchronize blinking of all

eight digits of the display. When this bit is a "1" all eight digits of the display will blink at approximately 2 Hz. The actual rate is dependent on the clock frequency. For an external clock, the blink rate can be calculated by dividing the clock frequency by 28,672. This function will override the Flash function when it is active. To use this function with multiple display systems, see the Display Reset section.

Self Test Function (Bits 5, 6)

Bit 6 of the Control Word Register is used to initiate the self test function. Results of the internal self test are stored in bit 5 of the Control Word. Bit 5 is a read only bit where bit 5 = "1" indicates a passed self test and bit 5 = "0" indicates a failed self test.

Setting bit 6 to a logic 1 will start the self test function. The built-in self test function of the IC consists of two internal routines which exercise major portions of the IC and illuminate all of the LEDs. The first routine cycles the ASCII decoder ROM through all states and performs a checksum on the output. If the checksum agrees with the correct value, bit 5 is set to "1." The second routine provides a visual test of the LEDs using the drive circuitry. This is accomplished by writing checkered and inverse checkered patterns to the display. Each pattern is displayed for approximately 2 seconds.

During the self test function the display must not be accessed. The time needed to execute the self test function is calculated by multiplying the clock period by 262,144. For example, assume a clock frequency of 58 KHz, then the time to execute the self test function frequency is equal to $(262,144/58,000) = 4.5$ second duration.

At the end of the self test function, the Character RAM is loaded with blanks, the Control Word Register is set to zeros except for bit 5, the Flash RAM is cleared, and the UDC Address Register is set to all ones.

Clear Function (Bit 7)

Bit 7 of the Control Word will clear the Character RAM and the Flash RAM. Setting bit 7 to a "1" will start the clear function. Three clock cycles (110 ms minimum using the internal refresh clock) are required to complete the clear function. The display must not be accessed while the display is being cleared. When the clear function has been completed, bit 7 will be reset to a "0." The ASCII character code for a space (20H) will be loaded into the Character RAM to blank the display and the Flash RAM will be loaded with "0"s. The UDC RAM, UDC Address Register, and the remainder of the Control Word are unaffected.

Display Reset

Figure 7 shows the logic levels needed to Reset the display. The display should be Reset on Power-up. The external Reset clears the Character RAM, Flash RAM, Control Word and resets the internal counters. After the rising edge of the Reset signal, three clock cycles (110 μ s minimum using the internal refresh clock) are required to complete the reset sequence. The display must not be accessed while the display is being reset. The ASCII Character code for a

RST	CE	WR	RD	FL	A ₇ -A ₀	D ₇ -D ₀
0	1	x	x	x	x	x

0 = LOGIC 0; 1 = LOGIC 1; x = DO NOT CARE
NOTE:
IF RST, CE AND WR ARE LOW, UNKNOWN
DATA MAY BE WRITTEN INTO THE DISPLAY.

Figure 7. Logic Levels to Reset the Display.

space (20H) will be loaded into the Character RAM to blank the display. The Flash RAM and Control Word Register are loaded with all "0"s. The UDC RAM and UDC Address Register are unaffected. All displays which operate with the same clock source must be simultaneously reset to synchronize the Flashing and Blinking functions.

Mechanical and Electrical Considerations

The HDSP-210X/-211X/-250X are 28 pin dual-in-line packages with 26 external pins. The devices can be stacked horizontally and vertically to create arrays of any size. The HDSP-210X/-211X/-250X are designed to operate continuously from -45°C to +85°C with a maximum of 20 dots on per character at 5.25 V. Illuminating all thirty-five dots at full brightness is not recommended.

The HDSP-210X/-211X/-250X are assembled by die attaching and wire bonding 280 LED chips and a CMOS IC to a thermally conductive printed circuit board. A polycarbonate lens is placed over the PC board creating an air gap over the LED wire bonds. A protective cap creates an air gap over the CMOS IC. Backfill epoxy environmentally seals the display package. This package construction makes the display highly tolerant to temperature cycling and allows wave soldering.

The inputs to the IC are protected against static discharge and input current latchup. However, for best results standard CMOS handling precautions should be

used. Prior to use, the HDSP-210X/-211X/-250X should be stored in antistatic tubes or in conductive material. During assembly, a grounded conductive work area should be used, and assembly personnel should wear conductive wrist straps. Lab coats made of synthetic material should be avoided since they are prone to static buildup. Input current latchup is caused when the CMOS inputs are subjected to either a voltage below ground ($V_{IN} < \text{ground}$) or to a voltage higher than V_{DD} ($V_{IN} > V_{DD}$) and when a high current is forced into the input. To prevent input current latchup and ESD damage, unused inputs should be connected either to ground or to V_{DD} . Voltages should not be applied to the inputs until V_{DD} has been applied to the display.

Thermal Considerations

The HDSP-210X/-211X/250X have been designed to provide a low thermal resistance path for the CMOS IC to the 26 package pins. Heat is typically conducted through the traces of the printed circuit board to free air. For most applications no additional heatsinking is required.

Measurements were made on a 32 character display string to determine the thermal resistance of the display assembly. Several display boards were constructed using 0.062 in. thick printed circuit material, and one ounce copper 0.020 in. traces. Some of the device pins were connected to a heatsink formed by etching a copper area on the printed circuit board surrounding the display. A maximally metallized printed circuit board was also evaluated.

The junction temperature was measured for displays soldered directly to these PC boards, displays installed in sockets, and finally displays installed in sockets with a filter over the display to restrict airflow. The results of these thermal resistance measurements, $R\theta_{J-A}$ are shown in Table 3 and include the effects of $R\theta_{J-C}$.

Ground Connections

Two ground pins are provided to keep the internal IC logic ground clean. The designer can, when necessary, route the analog ground for the LED drivers separately from the logic ground until an appropriate ground plane is available. On long interconnections between the display and the host system, the designer can keep voltage drops on the analog ground from affecting the display logic levels by isolating the two grounds.

The logic ground should be connected to the same ground potential as the logic interface circuitry. The analog ground and the logic ground should be connected at a common ground

which can withstand the current introduced by the switching LED drivers. When separate ground connections are used, the analog ground can vary from -0.3 V to +0.3 V with respect to the logic ground. Voltage below -0.3 V can cause all dots to be on. Voltage above +0.3 V can cause dimming and dot mismatch.

Soldering and Post Solder Cleaning Instructions for the HDSP-210X/-211X/-250X

The HDSP-210X/-211X/-250X may be hand soldered or wave soldered with SN63 solder. When hand soldering, it is recommended that an electronically temperature controlled and securely grounded soldering iron be used. For best results, the iron tip temperature should be set at 315°C (600°F). For wave soldering, a rosin-based RMA flux can be used. The solder wave temperature should be set at 245°C \pm 5°C (473°F \pm 9°F), and the dwell in the wave should be set between 1 1/2 to 3 seconds for optimum soldering. The preheat temperature should not exceed 105°C (221°F) as measured on the solder side of the PC board.

Table 3. Thermal Resistance, θ_{JA} , Using Various Amounts of Heatsinking Material

Heatsinking Metal per Device sq. in.	W/Sockets W/O Filter (Avg.)	W/O Sockets W/O Filter (Avg.)	W/Sockets W/Filter (Avg.)	Units
0	31	30	35	°C/W
1	31	28	33	°C/W
3	30	26	33	°C/W
Max. Metal	29	25	32	°C/W
4 Board Avg	30	27	33	°C/W

For additional information on soldering and post solder cleaning, see Application Note 1027, *Soldering LED Components*.

Contrast Enhancement

The objective of contrast enhancement is to provide good readability in a variety of ambient lighting conditions. For information on contrast enhancement see Application Note 1015, *Contrast Enhancement Techniques for LED Displays*.

APPENDICE F

TRANSPARENT LATCHES AND EDGE-TRIGGERED FLIP-FLOPSSN74LS373

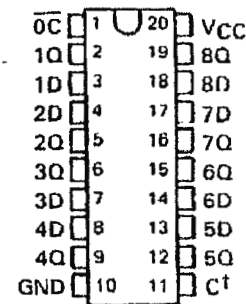
SDLS165

OCTAL D-TYPE TRANSPARENT LATCHES AND EDGE-TRIGGERED FLIP-FLOPS

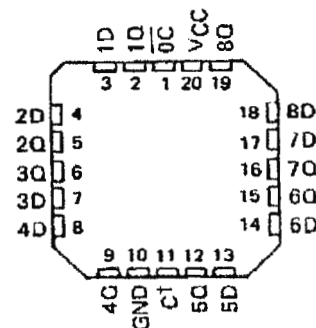
OCTOBER 1975—REVISED MARCH 1988

- Choice of 8 Latches or 8 D-Type Flip-Flops In a Single Package
- 3-State Bus-Driving Outputs
- Full Parallel-Access for Loading
- Buffered Control Inputs
- Clock/Enable Input Has Hysteresis to Improve Noise Rejection ('S373 and 'S374)
- P-N-P Inputs Reduce D-C Loading on Data Lines ('S373 and 'S374)

SN54LS373, SN54LS374, SN54S373, SN54S374 . . . J OR W PACKAGE
 SN74LS373, SN74LS374, SN74S373, SN74S374 . . . DW OR N PACKAGE
 (TOP VIEW)



SN54LS373, SN54LS374, SN54S373, SN54S374 . . . FK PACKAGE
 (TOP VIEW)



'LS373, 'S373
 FUNCTION TABLE

OUTPUT ENABLE	ENABLE LATCH	D	OUTPUT
L	H	H	H
L	H	L	L
L	L	X	Q ₀
H	X	X	Z

'LS374, 'S374
 FUNCTION TABLE

OUTPUT ENABLE	CLOCK	D	OUTPUT
L	↑	H	H
L	↑	L	L
L	L	X	Q ₀
H	X	X	Z

description

These 8-bit registers feature three-state outputs designed specifically for driving highly-capacitive or relatively low-impedance loads. The high-impedance third state and increased high-logic-level drive provide these registers with the capability of being connected directly to and driving the bus lines in a bus-organized system without need for interface or pull-up components. They are particularly attractive for implementing buffer registers, I/O ports, bidirectional bus drivers, and working registers.

The eight latches of the 'LS373 and 'S373 are transparent D-type latches meaning that while the enable (C) is high the Q outputs will follow the data (D) inputs. When the enable is taken low the output will be latched at the level of the data that was set up.

C for 'LS373 and 'S373; CLK for 'LS374 and 'S374

PRODUCTION DATA documents contain information current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.



POST OFFICE BOX 855012 • DALLAS, TEXAS 75265

SN54LS373, SN54LS374, SN74LS373, SN74LS374

OCTAL D-TYPE TRANSPARENT LATCHES AND EDGE-TRIGGERED FLIP-FLOPS

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

Supply voltage, V_{CC} (see Note 1)	7 V
Input voltage	7 V
Off-state output voltage	5.5 V
Operating free-air temperature range: SN54LS'	-55°C to 125°C
SN74LS'	0°C to 70°C
Storage temperature range	-65°C to 150°C

NOTE 1: Voltage values are with respect to network ground terminal.

recommended operating conditions

		SN54LS'			SN74LS'			UNIT
		MIN	NOM	MAX	MIN	NOM	MAX	
V_{CC}	Supply voltage	4.5	5	5.5	4.75	5	5.25	V
V_{OH}	High-level output voltage			5.5			5.5	V
I_{OH}	High-level output current			-1			-2.6	mA
I_{OL}	Low-level output current			12			24	mA
t_w	Pulse duration	CLK high		15			15	ns
		CLK low		15			15	ns
t_{su}	Data setup time	'LS373		5†			5†	ns
		'LS374		20†			20†	ns
t_h	Data hold time	'LS373		20†			20†	ns
		'LS374		5†			0†	ns
T_A	Operating free-air temperature	-55		125	0		70	°C

† The t_h specification applies only for data frequency below 10 MHz. Designs above 10 MHz should use a minimum of 5 ns. (Commercial only)

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER	TEST CONDITIONS†	SN54LS'			SN74LS'			UNIT	
		MIN	TYP‡	MAX	MIN	TYP‡	MAX		
V_{IH}	High-level input voltage	2			2			V	
V_{IL}	Low-level input voltage			0.7			0.8	V	
V_{IK}	Input clamp voltage	$V_{CC} = \text{MIN}, I_I = -18 \text{ mA}$					-1.5	V	
V_{OH}	High-level output voltage	$V_{CC} = \text{MIN}, V_{OH} = 2 \text{ V}, V_{IL} = V_{ILmax}, I_{OH} = \text{MAX}$			2.4	3.4	2.4	3.1	V
V_{OL}	Low-level output voltage	$V_{CC} = \text{MIN}, V_{IH} = 2 \text{ V}, V_{IL} = V_{ILmax}$					0.25	0.4	V
							0.35	0.5	V
I_{OZH}	Off-state output current, high-level voltage applied	$V_{CC} = \text{MAX}, V_{IH} = 2 \text{ V}, V_O = 2.7 \text{ V}$					20	20	µA
I_{OZL}	Off-state output current, low-level voltage applied	$V_{CC} = \text{MAX}, V_{IH} = 2 \text{ V}, V_O = 0.4 \text{ V}$					-20	-20	µA
I_I	Input current at maximum input voltage	$V_{CC} = \text{MAX}, V_I = 7 \text{ V}$					0.1	0.1	mA
I_{IH}	High-level input current	$V_{CC} = \text{MAX}, V_I = 2.7 \text{ V}$					20	20	µA
I_{IL}	Low-level input current	$V_{CC} = \text{MAX}, V_I = 0.4 \text{ V}$					-0.4	-0.4	mA
I_{OS}	Short-circuit output current§	$V_{CC} = \text{MAX}$			-30	-130	-30	-130	mA
I_{CC}	Supply current	$V_{CC} = \text{MAX},$							mA
		Output control at 4.5 V			'LS373	24	40	24	40
					'LS374	27	40	27	40

† For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.

‡ All typical values are at $V_{CC} = 5 \text{ V}, T_A = 25^\circ \text{C}$.

§ Not more than one output should be shorted at a time and duration of the short circuit should not exceed one second.

TEXAS
INSTRUMENTS

POST OFFICE BOX 655012 • DALLAS, TEXAS 75265

SN54LS373, SN54LS374, SN74LS373, SN74LS374
OCTAL D-TYPE TRANSPARENT LATCHES AND
EDGE-TRIGGERED FLIP-FLOPS

switching characteristics. $V_{CC} = 5\text{ V}$, $T_A = 25^\circ\text{C}$

PARAMETER	FROM (INPUT)	TO (OUTPUT)	TEST CONDITIONS	'LS373			'LS374			UNIT
				MIN	TYP	MAX	MIN	TYP	MAX	
f_{max}			$C_L = 45\text{ pF}$, $R_L = 667\ \Omega$ See Notes 2 and 3				35	50		MHz
t_{PLH}	Data	Any Q		12	18					ns
t_{PHL}				12	18					
t_{PLH}	Clock or enable	Any Q		20	30	15	28			ns
t_{PHL}				18	30	19	28			
t_{PZH}	Output Control	Any Q		15	28	20	26			ns
t_{PZL}	Control		25	36	21	28				
t_{PHZ}	Output Control	Any Q	$C_L = 5\text{ pF}$, $R_L = 667\ \Omega$ See Note 3	15	25	15	28			ns
t_{PLZ}	Output Control	Any Q		12	20	12	20			ns

NOTES: 2. Maximum clock frequency is tested with all outputs loaded.

3. Load circuits and voltage waveforms are shown in Section 1.

f_{max} = maximum clock frequency

t_{PLH} = propagation delay time, low-to-high-level output

t_{PHL} = propagation delay time, high-to-low-level output

t_{PZH} = output enable time to high level

t_{PZL} = output enable time to low level

t_{PHZ} = output disable time from high level

t_{PLZ} = output disable time from low level


TEXAS
INSTRUMENTS

POST OFFICE BOX 655012 • DALLAS, TEXAS 75265

SN54S373, SN54S374, SN74S373, SN74S374 OCTAL D-TYPE TRANSPARENT LATCHES AND EDGE-TRIGGERED FLIP-FLOPS

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER	TEST CONDITIONS [†]		MIN	TYP [‡]	MAX	UNIT
V _{IH}			2			V
V _{IL}					0.8	V
V _{IK}	V _{CC} = MIN, I _I = -18 mA				-1.2	V
V _{OH}	SN54S'	V _{CC} = MIN, V _{IH} = 2 V, V _{IL} = 0.8 V, I _{OH} = MAX	2.4	3.4		V
	SN74S'		2.4	3.1		
V _{OL}	V _{CC} = MIN, V _{IH} = 2 V, V _{IL} = 0.8 V, I _{OL} = 20 mA				0.5	V
I _{OZH}	V _{CC} = MAX, V _{IH} = 2 V, V _O = 2.4 V				50	μA
I _{OZL}	V _{CC} = MAX, V _{IH} = 2 V, V _O = 0.5 V				-60	μA
I _I	V _{CC} = MAX, V _I = 5.5 V				1	mA
I _{IH}	V _{CC} = MAX, V _I = 2.7 V				50	μA
I _{IL}	V _{CC} = MAX, V _I = 0.5 V				-250	μA
I _{OS} [§]	V _{CC} = MAX		-40		-100	mA
I _{CC}	V _{CC} = MAX	'S373	outputs high		160	mA
			outputs low		160	
			outputs disabled		190	
		'S374	outputs high		110	
			outputs low		140	
			outputs disabled		160	
		CLK and OC at 4 V, D inputs at 0 V		160		

[†]For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.

[‡]All typical values are at V_{CC} = 5 V, T_A = 25°C.

[§]Not more than one output should be shorted at a time and duration of the short-circuit should not exceed one second.

switching characteristics, V_{CC} = 5 V, T_A = 25°C

PARAMETER	FROM (INPUT)	TO (OUTPUT)	TEST CONDITIONS	'S373			'S374			UNIT
				MIN	TYP	MAX	MIN	TYP	MAX	
f _{max}			C _L = 15 pF, R _L = 280 Ω, See Notes 2 and 4				75	100	MHz	
t _{PLH}	Data	Any Q		7	12				ns	
t _{PHL}					7	12				
t _{PLH}	Clock or enable	Any Q		7	14	8	15		ns	
t _{PHL}					12	18	11	17		
t _{PZH}	Output Control	Any Q		8	15	8	15		ns	
t _{PZL}				11	18	11	18			
t _{PHZ}	Output Control	Any Q	C _L = 5 pF, R _L = 280 Ω, See Note 3		6	9	5	9	ns	
t _{PLZ}				8	12	7	12			

NOTES: 2. Maximum clock frequency is tested with all outputs loaded.

4. Load circuits and voltage waveforms are shown in Section 1.

f_{max} = maximum clock frequency

t_{PLH} = propagation delay time, low-to-high-level output

t_{PHL} = propagation delay time, high-to-low-level output

t_{PZH} = output enable time to high level

t_{PZL} = output enable time to low level

t_{PHZ} = output disable time from high level

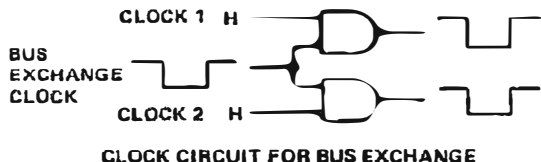
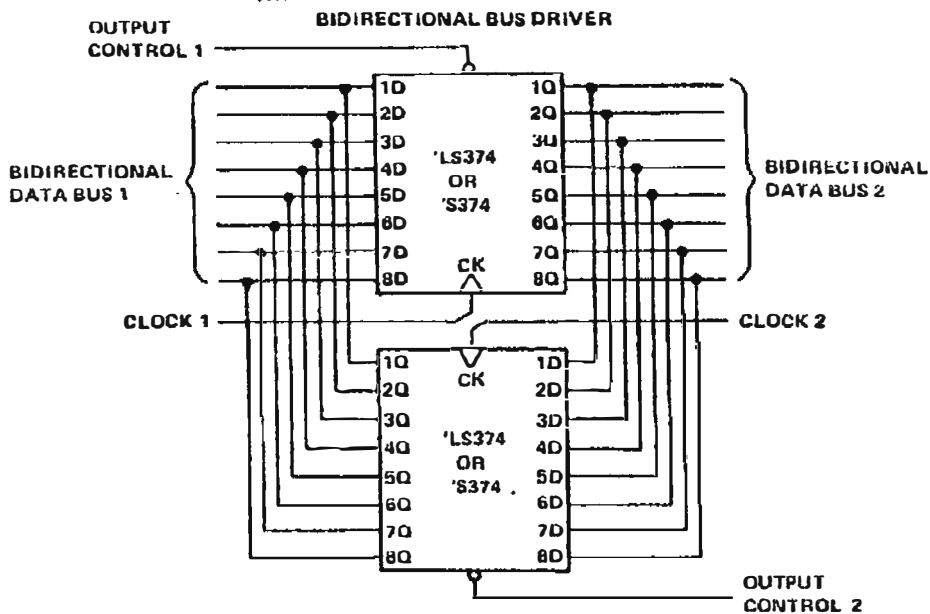
t_{PLZ} = output disable time from low level

TEXAS
INSTRUMENTS

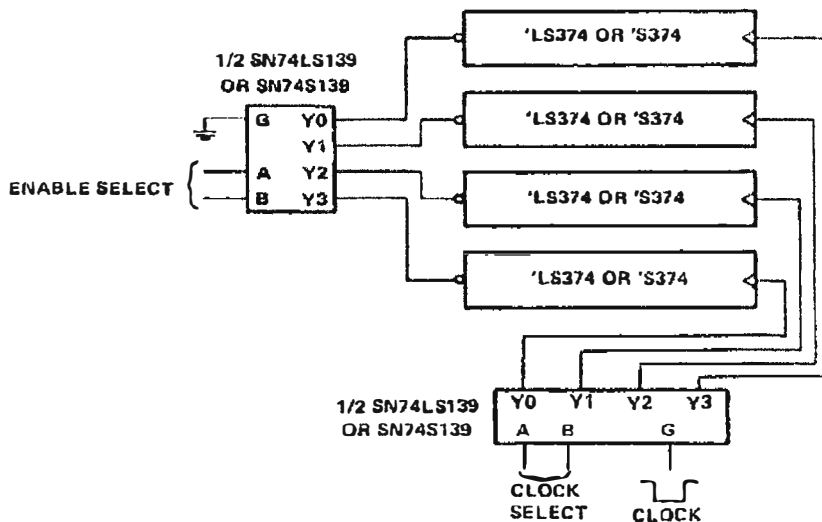
POST OFFICE BOX 655012 • DALLAS, TEXAS 75265

**SN54LS373, SN54LS374, SN54S373, SN54S374,
SN74LS373, SN74LS374, SN74S373, SN74S374
OCTAL D-TYPE TRANSPARENT LATCHES AND EDGE-TRIGGERED FLIP-FLOPS**

TYPICAL APPLICATION DATA



EXPANDABLE 4-WORD-BY-8-BIT GENERAL REGISTER FILE



**SN54LS373, SN54LS374, SN54S373, SN54S374,
SN74LS373, SN74LS374, SN74S373, SN74S374**
OCTAL D-TYPE TRANSPARENT LATCHES AND EDGE-TRIGGERED FLIP-FLOPS

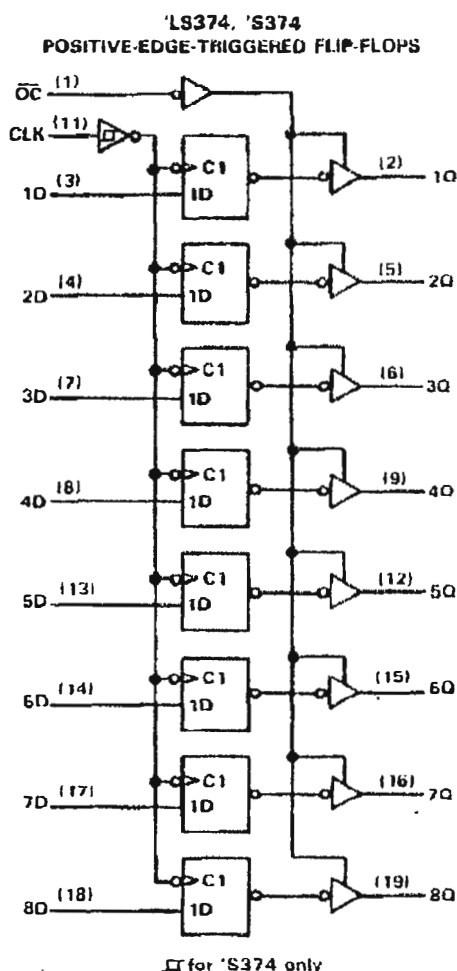
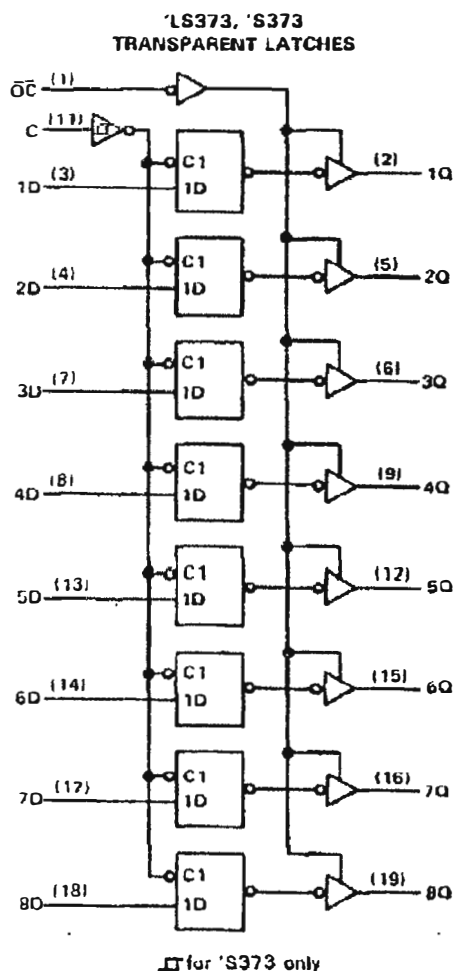
description (continued)

The eight flip-flops of the 'LS374 and 'S374 are edge-triggered D-type flip-flops. On the positive transition of the clock, the Q outputs will be set to the logic states that were setup at the D inputs.

Schmitt-trigger buffered inputs at the enable/clock lines of the 'S373 and 'S374 devices, simplify system design as ac and dc noise rejection is improved by typically 400 mV due to the input hysteresis. A buffered output control input can be used to place the eight outputs in either a normal logic state (high or low logic levels) or a high-impedance state. In the high-impedance state the outputs neither load nor drive the bus lines significantly.

The output control does not affect the internal operation of the latches or flip-flops. That is, the old data can be retained or new data can be entered even while the outputs are off.

logic diagrams (positive logic)



Pin numbers shown are for DW, J, N, and W packages.

TEXAS
INSTRUMENTS

POST OFFICE BOX 655012 • DALLAS, TEXAS 75265