



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

**FACULTAD DE ESTUDIOS SUPERIORES
"CUAUTITLAN"**

**DISEÑO DE UN PROCESADOR
DIGITAL DIDACTICO**

T E S I S

QUE PARA OBTENER EL TITULO DE:

INGENIERO MECANICO ELECTRICISTA

P R E S E N T A :

MORALES ORTIZ ROBERTO

DIRECTOR DE LA TESIS:

M. EN C. JUAN ANTONIO NAVARRO MARTINEZ

CUAUTITLAN IZCALLI, EDO. DE MEXICO

1985



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

I N D I C E .

	Página:
Introducción.....	i
Capítulo I.	
Características del Procesador.	
I.1.- Objetivo Principal.....	I.1
I.2.- Descripción del Procesador.....	I.1
I.2.A.- Métodos de Direccinamiento.....	I.2
I.2.B.- Formato de las Instrucciones.....	I.2
I.2.C.- Grupos de Instrucciones.....	I.5
I.2.D.- Descripción de los Buses.....	I.6
I.2.E.- Operaciones de Entrada y Salida.....	I.9
I.2.F.- Registros del Procesador.....	I.10
I.2.G.- Dispositivos Empleados en la Cons- trucción del Procesador.....	I.11
Capítulo II.	
Instrucciones del Procesador.	
II.1.- Simbología Empleada en la Descripción de las Instrucciones.....	II.1
II.2.- Grupo de Instrucciones de Transferencia de Operandos.....	II.6
II.3.- Grupo de Instrucciones Aritméticas.....	II.7
II.4.- Grupo de Instrucciones de Corrimiento....	II.9
II.5.- Grupo de Instrucciones Lógicas.....	II.10
II.6.- Grupo de Instrucciones de Modificación de Programa.....	II.12
II.7.- Grupo de Instrucciones de Función Nula...	II.16
Capítulo III.	
Arquitectura del Procesador.	
III.1.- Descripción de los Bloques del Procesador.....	III.1
III.1.A.- La ALU.....	III.3
III.1.B.- El Registro A.....	III.7
III.1.C.- El Registro T.....	III.11
III.1.D.- El Registro F.....	III.13
III.1.E.- El Registro HL.....	III.17
III.1.F.- El Registro PC.....	III.20
III.1.G.- El Registro S.....	III.23
III.1.H.- CB1.....	III.26
III.1.I.- CB2.....	III.28
III.1.J.- CB3.....	III.30

	Página:
III.1.K.- CB4.....	III.32
III.1.L.- La Sección de Control.....	III.33

Capítulo IV.

Sección de Control del Procesador

IV.1.- Arquitectura del Controlador.....	IV.2
IV.1.A.- La Prom de Control.....	IV.2
IV.1.A.1.- Campo de Control de Proceso.....	IV.2
IV.1.A.2.- Campo de Control del Controlador.....	IV.6
IV.1.B.- El Registro SM.....	IV.16
IV.1.C.- La Prom de Correspondencia.....	IV.17
IV.1.D.- El Registro MPC.....	IV.18
IV.1.E.- El Multiplexor.....	IV.20
IV.1.F.- El Registro DR.....	IV.22
IV.1.G.- El Registro RI.....	IV.23
IV.1.H.- El Registro RS.....	IV.24
IV.1.I.- La Lógica de Espera.....	IV.26
IV.1.J.- La Lógica de Condición.....	IV.27
IV.2.- Descripción de las Microoperaciones.....	IV.29
IV.2.A.- Microoperaciones de Control de Proceso.....	IV.29
IV.2.B.- Microoperaciones de Control del Controlador.....	IV.42
IV.3.- Contenido de la Prom de Control.....	IV.57
IV.4.- Contenido de la Prom de Correspondencia...	IV.72

Capítulo V.

Manual de Funcionamiento del Procesador.

V.1.- Arquitectura del Sistema.....	V.1
V.1.A.- La Sección de Proceso.....	V.2
V.1.B.- La Sección de Control.....	V.3
V.1.C.- Líneas de Alimentación.....	V.6
V.2.- Ciclos de Máquina del Procesador.....	V.7
V.2.A.- Ciclo de Búsqueda de Instrucción.....	V.8
V.2.B.- Ciclo de Acceso para Lectura.....	V.10
V.2.C.- Ciclo de Acceso para Escritura.....	V.10
V.2.D.- Ciclo de Ejecución de una Instrucción ALTO.....	V.12
V.2.E.- Ciclo de Reconocimiento de Interrupción.....	V.13
V.2.F.- Ciclo de Concesión del Bus.....	V.15
V.3.- Repertorio de Instrucciones.....	V.16

Página:

V.3.A.- Formato y Modo de Direcciona- miento de las Instrucciones.....	V.17
V.3.B.- Lista de Instrucciones.....	V.18
Conclusiones.....	VI.1
Bibliografía.....	VI.6
Apéndice: Costos.....	VI.7

I N T R O D U C C I O N .

La aparición de los microprocesadores a principios de la década de los 70's, ocasionó un gran desarrollo en el área de las computadoras digitales; de tal forma que gran parte de los trabajos científicos y tecnológicos que anteriormente necesitaban meses y aún años de cálculos pudieron ser realizados de una forma bastante mas rápida jamás vista antes y al alcance de un gran número de personas.

Pero el desarrollo de la tecnología de las computadoras empezó algunas décadas antes; de hecho, la primera computadora de uso general fué desarrollada en 1946, estaba compuesta de alrededor de 18,000 bulbos y la fuente de poder para alimentarla era de casi un medio del tamaño de la computadora misma, el sistema completo ocupaba una sala de aproximadamente 9 X 15 metros.

Ya en los 60's computadoras mas pequeñas pero mas poderosas que la descrita anteriormente fueron desarrolladas usando cientos de compuertas y flip-flops y otros circuitos integrados similares llamados "de Pequeña Escala de Integración" (SSI), estas máquinas eran de aproximadamente el tamaño de una máquina de escribir.

Pronto y debido al desarrollo en la tecnología de los semiconductores fué posible construir circuitos integrados dentro de los cuales hubiera decenas de compuertas; estos circuitos fueron tales como los contadores, decodificadores, etc. llamados "de Mediana Escala de Integración" (MSI).

La tendencia a la miniaturización continuó y en 1971 el primer microprocesador fué desarrollado. El microprocesador contiene las secciones de control y proceso de una computadora en un solo circuito integrado llamado "Unidad Central de Proceso" (CPU). Este circuito integrado esta compuesto por miles de compuertas y es llamado un dispositivo "de Larga Es

cala de Integración" (LSI).

ii

Paralelamente al desarrollo de estos dispositivos fueron desarrollados elementos de memoria de larga escala de integración que hicieron posible el almacenamiento de millares - de bits de información digital en un solo circuito integrado. Actualmente, circuitos integrados como los descritos en las líneas anteriores tienen unas dimensiones de aproximadamente 1/4 de pulgada por lado, esto ha reducido de una manera drástica el costo y el tamaño de computadoras pequeñas.

Actualmente la aplicación de los microprocesadores es --- prácticamente ilimitada; lo mismo los podemos encontrar en - grandes fábricas donde controlan las líneas de producción y automatizan los procesos de producción; los tenemos en los - supermercados dentro de las cajas registradoras y las básculas; los podemos encontrar en el hogar, dentro de las lavadoras automáticas, las televisiones con control remoto, los -- hornos de microondas programables, las calculadoras y hasta en los juguetes. Y por supuesto que también están presentes - en los laboratorios de desarrollo científico y tecnológico - dentro de las computadoras y el equipo de prueba.

El desarrollo de la tecnología ha revolucionado de tal manera a la sociedad actual, que esta no sería concebible sin la existencia de las computadoras y los dispositivos computarizados; es por esto que el ingeniero moderno debe estar familiarizado con el uso y/o funcionamiento de las computadoras.

Una computadora moderna es capaz de almacenar una cantidad enorme de información y de procesarla a gran velocidad - haciéndolo de una manera muy confiable y exacta. Es por esto que la mayoría de la gente las considera grandes máquinas capaces inclusive de pensar; pero esto es algo aún lejano de - la realidad.

Tipicamente, una computadora esta compuesta por los siguientes bloques arquitectónicos:

-Unidad de Memoria; en ella es almacenada toda la información que esta en proceso.

-Dispositivos de Entrada y Salida; estos dispositivos le permiten a la computadora intercomunicarse con el mundo exterior.

-Unidad Central de Proceso; esta unidad es considerada el cerebro de la computadora, en ella es procesada toda la información proveniente ya sea de la unidad de memoria o de los dispositivos de entrada y salida.

Una Unidad Central de Proceso de una gran computadora es capaz de procesar en paralelo un gran número de bits (bit: unidad de información digital) y generalmente están construidas de decenas de circuitos integrados. Estas grandes unidades de proceso generalmente están fuera del alcance del diseñador promedio y el único contacto que este tiene con ellas es através del uso de la computadora; cuando corre programas y realiza cálculos.

Actualmente se ha generalizado el uso de pequeñas unidades centrales de proceso que son capaces de procesar, algunas de ellas, hasta 16 bits y son ampliamente conocidas con el nombre de microprocesadores debido tal vez en parte a su reducida capacidad de procesamiento de bits y en parte a su reducido tamaño, ya que generalmente están contenidas en un solo circuito integrado cuyo encapsulado es de apenas unos 5 X 2 cm.. Estos microprocesadores han sido ampliamente acogidos por los diseñadores de controles automáticos de todo tipo y por los diseñadores de pequeñas computadoras llamadas microcomputadoras. Su popularidad es debida en gran parte a lo económicos y flexibles que resultan estos dispositivos, de aquí que su estudio resulte ahora tan necesario en dife--

rentes áreas de la ingeniería.

Pero el estudio de los microprocesadores resulta un tanto abstracto ya que para la mayoría de los estudiantes resulta ser tan solo una caja negra de la que se dice estar compuesta por estos y aquellos elementos y procesar la información de esta o aquella manera, cosa que resulta didacticamente -- muy pobre e imposible de comprobar. Las consecuencias de esto son un vago entendimiento de los microprocesadores y un mal aprovechamiento de los mismos. Es por esto que observando la gran importancia actual de los microprocesadores surge la necesidad de diseñar un procesador cuya filosofía de diseño, estructuración y funcionamiento sea similar a la de los microprocesadores y cuyo tamaño resulte suficientemente grande como para que puedan ser observados tanto sus elementos constitutivos como las transformaciones que va sufriendo la información en su paso através de ellos; es presumible que un sistema con estas características ayudaría bastante a los estudiantes a tener un entendimiento mas claro de lo que son los microprocesadores, de lo que pueden hacer y de como lo hacen.

Es importante que antes de comenzar a trabajar en el diseño de algún sistema se realicen ciertas investigaciones que permitan evaluar las dificultades que se pueden presentar en el desarrollo del mismo. Primeramente se debe definir un objetivo primordial que señale la ruta que se debe seguir y -- las posibilidades de tener éxito en el diseño. Al diseñar un dispositivo digital es deseable que en la estructuración del plan de trabajo se sigan los pasos siguientes: especificación escrita, diseño arquitectural, elaboración de una lista basada en los recursos asequibles, elaboración de los diferentes diagramas de bloques, integración del sistema, revisión, rediseño y documentación.

Una revisión debe tener lugar en cada una de las etapas - antes de que la siguiente sea iniciada.

La especificación escrita debe contener cuando menos las secciones siguientes: propósitos de diseño, formatos, ins---trucciones requeridas, métodos de direccionamiento, operacio---nes de entrada y salida y memoria.

El propósito de la especificación es proveer la filosofía inherente a la máquina, una descripción funcional completa y las limitaciones propias de la máquina.

El diseño arquitectónico deberá proveer una descripción - de cada uno de los bloques controlados en la máquina, una -- descripción de los buses y una del sistema controlador. Cada uno de los bloques, cada dispositivo dentro del bloque, los buses o canales de comunicación y su control deberán ser descritos desde un punto de vista funcional.

No hay una fórmula que aplicada al diseño de sistemas digitales nos asegure un buen resultado final; solo la expe---riencia y la estructuración de un buen plan de trabajo pueden redituuar buenos resultados.

C A P I T U L O I.

CARACTERISTICAS DEL PROCESADOR.

Este capítulo tiene el propósito de hacer una descripción funcional del procesador que va a ser diseñado. Al término de este capítulo se deberá tener una idea de las características de funcionamiento del procesador.

I.1.-OBJETIVO PRINCIPAL. Como ya ha sido descrito en la introducción, el objetivo principal al realizar este trabajo es el de diseñar un procesador que tenga una filosofía de diseño, estructuración y funcionamiento similar a la de los microprocesadores, pero que resulte lo bastante grande como para que puedan ser observados tanto sus principales bloques arquitectónicos como las transformaciones que va sufriendo la información en su paso a través de ellos. Lo dicho anteriormente implica que el procesador será capaz de funcionar de una manera estática, o dicho mas claramente, a una velocidad lo mas baja posible. Esta debe ser una característica muy importante del sistema ya que permitirá hacer las observaciones necesarias - del funcionamiento del procesador. Debe quedar claro desde ahora que el procesador tendrá fines didácticos, por lo que la alta velocidad y la gran cantidad de registros internos que poseen los microprocesadores en la actualidad y que los hacen máquinas sumamente poderosas no serán características de este sistema. En suma, no se pretende diseñar una máquina poderosa ya que en caso de necesitarlo resultaría mas eficiente y económico un microprocesador de los existentes en el mercado.

I.2.-DESCRIPCION DEL PROCESADOR. Esta descripción debe ser tomada como punto de partida del diseño ya que contendrá de una manera explícita las características de funcionamiento -- del procesador que permitirán elegir la arquitectura que parezca mas apropiada para su funcionamiento.

En esta sección se describirán los tipos de direccionamiento, los grupos y formatos de las instrucciones, los buses del procesador y las operaciones de entrada y salida.

I.2.A.-METODOS DE DIRECCIONAMIENTO. Estos se refieren a la forma en la que el procesador obtiene los operandos que son necesarios para realizar cada instrucción. Los métodos de direccionamiento que serán utilizados por este procesador son los siguientes:

DIRECCIONAMIENTO DIRECTO. En este método, la instrucción posee la dirección de la localidad de memoria en la que se encuentra el operando.

DIRECCIONAMIENTO INDIRECTO. En el, la instrucción contiene la dirección de la localidad de memoria en la que se encuentra la dirección efectiva del operando.

DIRECCIONAMIENTO INMEDIATO. En este método, la instrucción posee al propio operando.

Los tres métodos de direccionamiento mencionados anteriormente están ilustrados de una manera representativa en la figura I.1.

DIRECCIONAMIENTO DE REGISTROS. Este es el cuarto y último de los métodos de direccionamiento que serán utilizados aquí. En este método, la instrucción posee el origen y el destino del operando; tanto el origen como el destino del operando serán registros internos del procesador.

I.2.B.-FORMATO DE LAS INSTRUCCIONES. Las instrucciones utilizadas por este procesador podrán ser de uno, dos o tres bytes. Siempre, en cualquiera de los tres formatos, el primer byte contendrá el código de operación de la instrucción, los bytes restantes contendrán ya sea a un operando o bien a la dirección de algún operando. En la figura I.2 se muestran instrucciones de uno, dos y tres bytes.

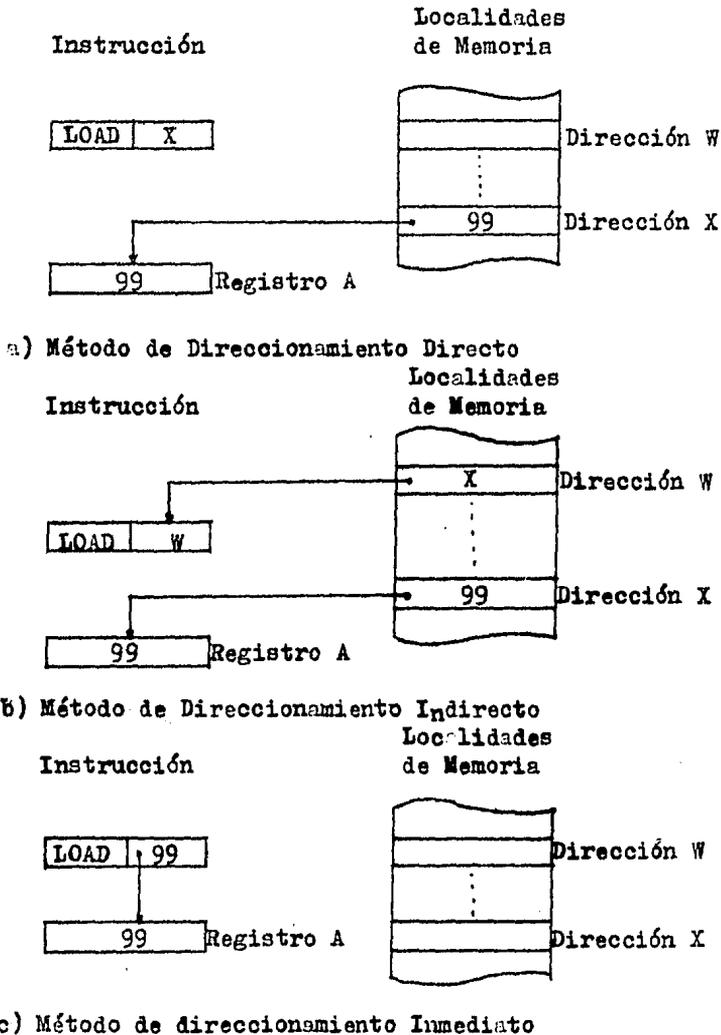


FIGURA I.1.- Tres instrucciones de carga que transfieren el número 99 al registro A del procesador - utilizando tres métodos de direccionamiento distintos: a) Directo. b) Indirecto. c) Inmediato.

1er. Byte

INC	A
-----	---

 Código de Operación

a). Instrucción de un byte.

1er. Byte

SUM	A
-----	---

 Código de Operación
2o. Byte

99

 Operando

b). Instrucción de dos bytes.

1er. Byte

SLM	A
-----	---

 Código de Operación
2o. Byte

00

 Byte Menos Significativo
3er. Byte

80

Byte Mas Significativo
Dirección del Operando

c). Instrucción de tres bytes.

FIGURA I.2.- Formato de las instrucciones.

- a) Instrucción de un byte
- b) Instrucción de dos bytes
- c) Instrucción de tres bytes

De la figura I.2 se puede intuir que el formato de las instrucciones dependerá del método de direccionamiento que se emplee en ellas; por ejemplo en la figura I.2a la instrucción -INC A (incrementa el contenido del registro A) utiliza el método de direccionamiento de registros; en la figura I.2b la -instrucción SUM A 99 (suma el contenido del registro A al operando 99) se utiliza el método de direccionamiento inmediato y en la figura I.2c la instrucción SLM A 8000 (suma al contenido del registro A el operando contenido en la localidad de memoria 8000) se utiliza el método de direccionamiento directo. No es este el momento apropiado para comenzar a definir -de una manera concisa a cada instrucción y es posible que los comentarios anteriores resulten vagos ya que aún no han sido diseñadas las instrucciones y las descritas anteriormente pueden o no estar dentro del repertorio de instrucciones de este procesador. La descripción y diseño de las instrucciones serán ampliamente tratadas en un capítulo posterior.

I.2.C.-GRUPOS DE INSTRUCCIONES. Existe una variedad de instrucciones que pueden ser agrupadas de acuerdo a la función -que desempeñan, por ejemplo existen instrucciones de transferencia, aritméticas, etc., a continuación en esta sección se mencionarán los grupos de instrucciones que estarán dentro --del repertorio de instrucciones del procesador:

TRANSFERENCIA DE OPERANDOS. Este grupo será el encargado -de realizar las siguientes funciones: Cargar un operando proveniente del contenido de una localidad de memoria en el contenido de un registro del procesador, cargar un operando contenido en un registro del procesador en el contenido de una -localidad de memoria y cargar un operando contenido en un registro del procesador en el contenido de otro registro del --procesador.

ARITMETICAS. Las instrucciones contenidas en este grupo realizarán las siguientes funciones: Suma con acarreo y suma -- sin acarreo, resta con acarreo y resta sin acarreo, incremento, decremento y comparación.

LOGICAS. Las instrucciones del grupo lógico realizarán las funciones AND, OR, NOT, EXOR y complemento a dos.

CORRIMIENTOS. Las funciones realizadas por las instrucciones de este grupo deberán comprender corrimientos a la derecha, corrimientos a la izquierda, rotaciones a la derecha y a la izquierda y desplazamientos a la derecha e izquierda.

MODIFICACION DEL PROGRAMA. Las instrucciones de este grupo serán las encargadas de realizar las funciones de salto, salto a subrutina y regreso de subrutina.

La descripción hecha en las secciones anteriores esta dada desde el punto de vista de la forma en que son manejados los operandos, ahora es necesario hacer una descripción que tenga un enfoque hacia la forma en que va a ser construido el procesador.

I.2.D.-DESCRIPCION DE LOS BUSES. Los buses son los canales de comunicación que utiliza el procesador para comunicarse -- con los dispositivos externos; los canales de comunicación de este procesador tendrán las siguientes características de --- construcción:

BUS DE DIRECCIONES. Este bus será de 16 bits, por lo que -- podrá direccionar hasta 65536 direcciones de memoria que es -- el equivalente a 64k's. La comunicación en este bus será en -- una sola dirección que es la de salida del procesador.

BUS DE DATOS. El bus de datos será de 8 bits, por el flui- -- rán los operandos, los códigos de operación y los bytes menos -- significativos y mas significativos de la parte de dirección -- de las instrucciones. Este bus deberá ser bidireccional ya --

que los datos podrán entrar o salir del procesador.

BUS DE CONTROL. Este bus contendrá las líneas necesarias para controlar al procesador (líneas de entrada) desde algún dispositivo externo y las líneas necesarias para que el procesador controle a los dispositivos externos (líneas de salida). Las líneas de entrada del bus de control pueden ser enumeradas de la siguiente manera:

DE ALIMENTACION. Estas son las que dan al procesador el suministro de energía necesario para que funcione. Son características las líneas de voltaje de alimentación (Vcc), tierra (Gnd) y la señal de reloj (Ck).

DE CONTROL PARA EL PROCESADOR. Estas líneas son necesarias para sincronizar al procesador con el funcionamiento de los dispositivos externos y también para que el procesador pueda dar atención a eventos que se suceden de una manera esporádica en los dispositivos externos.

Línea de Reset. El objetivo de esta línea es el de poner a funcionar al procesador bajo condiciones iniciales ya conocidas; al activarse esta línea se forzará al procesador a realizar un ciclo de búsqueda de instrucción en la localidad de memoria cero.

Línea de espera. Este procesador contará con una línea de espera que le permitirá recibir un aviso de los dispositivos externos para ponerse en sincronía con ellos. Al recibir esta señal el procesador esperará el tiempo necesario a los dispositivos externos y reanudará su trabajo hasta que la señal sea desactivada.

Línea de interrupción. También contará con una línea de interrupción para que puedan ser atendidos los dispositivos de entrada y salida. Al recibir una señal de interrupción, el procesador interrumpirá el trabajo que estaba haciendo y

atenderá a la rutina de servicio del dispositivo que interrumpió, al completar esta rutina el procesador continuará realizando el trabajo que hacía antes de ser interrumpido.

Línea de solicitud del bus. La última línea de entrada que falta de definir es la línea de solicitud del bus; esta línea indicará al procesador que algún dispositivo externo necesita hacer uso de los buses del procesador. Al censar esta línea, el procesador contestará activando la línea de salida de concesión del bus y poniendo en tercer estado los buses de direcciones y datos y todas las señales de salida del bus de control, con excepción de la línea de concesión del bus además el procesador interrumpirá el trabajo que estaba haciendo hasta que la línea de solicitud del bus sea desactivada.

Las líneas de salida del bus de control deberán ser las encargadas de realizar los ciclos de máquina del procesador; en estos ciclos de máquina, el bus de control del procesador generará las señales necesarias para realizar accesos ya sea para lectura o escritura en los dispositivos externos al procesador (los dispositivos externos son los dispositivos de entrada y salida, de memoria, etc.). Las líneas usadas con este fin serán las siguientes:

Línea de alto. Al activarse esta línea, el procesador indicará que se está realizando una instrucción ALTO.

Línea de acceso al exterior. Esta línea servirá para indicar que se está haciendo un acceso hacia los dispositivos externos al procesador, este acceso podrá ser de escritura o lectura.

Línea de escritura. Esta línea indicará que el acceso al exterior que se está haciendo tendrá como finalidad escribir en el dispositivo externo.

Línea de lectura. La finalidad de esta línea es indicar -

que el acceso al exterior se hace para leer el contenido del dispositivo externo. Las líneas de lectura y escritura son mutuamente excluyentes; esto quiere decir que no podrán ocurrir de manera simultánea.

Línea de búsqueda. Esta línea indicará que se está haciendo un ciclo de búsqueda de instrucción.

Línea de reconocimiento de interrupción. Esta línea servirá para indicar que se está realizando un ciclo de reconocimiento de interrupción.

Línea de concesión del bus. Al activarse esta línea el procesador indicará que puede hacerse uso de los buses de direcciones y datos y de las líneas de salida del bus de control con excepción de ésta.

Finalmente hace falta mencionar que todas las líneas del bus de control serán activas en un nivel de cero lógico.

I.2.E.- OPERACIONES DE ENTRADA Y SALIDA. Como ya se mencionó anteriormente, el procesador será capaz de direccionar hasta 64 K bytes de memoria. Los puertos de entrada y salida conectados al procesador deberán ser mapeados en memoria ya que no existirá diferencia entre los ciclos de máquina que acceden a la memoria y los ciclos de máquina que acceden a los puertos.

El procesador tendrá un solo nivel de interrupción por lo que en caso de necesitarse diferentes niveles de interrupción, estos deberán ser construidos en una lógica de selección de prioridad externa al procesador. En caso de ocurrir una interrupción, el dispositivo que interrumpe deberá presentar una identificación de 8 bits en el bus de datos que permitirá acceder a la rutina de servicio correspondiente; la rutina de servicio podrá estar localizada por lo tanto en cualquiera de las 256 primeras localidades de memoria.

Además el procesador deberá ser capaz de recibir una se--

Mal de aviso de los dispositivos externos (periféricos, memoria, etc.) que le permita sincronizarse a ellos y una señal que le indique al procesador que algún dispositivo externo necesita hacer uso de los buses.

I.2.F.- REGISTROS DEL PROCESADOR. El procesador poseerá varios registros internos, pero solo tres de ellos podrán ser usados y/o accedidos por las instrucciones del procesador. El más importante de ellos es el registro acumulador (REGISTRO A); este es un registro de trabajo, en el será guardado un operando y el resultado de las operaciones realizadas por la unidad lógico-aritmética; también en el se realizarán los corrimientos tanto a la derecha como a la izquierda. El registro acumulador será un registro de 8 bits.

Otro registro que podrá ser usado por las instrucciones es el registro de banderas (REGISTRO F). El registro F es un registro de almacenamiento de 8 bits, aunque no todos los bits tienen significado para el procesador. En este registro serán guardadas las banderas entregadas por la unidad lógico-aritmética como por ejemplo la bandera de cero (z) y la bandera de acarreo (a) estas banderas serán usadas y censadas por instrucciones por ejemplo aritméticas (suma con acarreo) o bien por instrucciones de salto condicional (salta a una dirección determinada si está puesta la bandera de z, o bien, la bandera de acarreo a).

El tercer registro es también un registro de almacenamiento de los llamados stack o pila (REGISTRO S); el funcionamiento de este tipo de registros se basa en la regla que dice que el último operando en entrar es el primero en salir de estos registros. Este es un registro de 16 bits; en el podrán ser almacenados los datos contenidos en los registros A y F, el contenido de estos registros será puesto en los 8 bits menos significativos de S. El acceso a este registro se

hará por medio de instrucciones que son conocidas como "PUSH" y "POP"; este tipo de instrucciones serán descritas en un capítulo posterior.

I.2.G.- DISPOSITIVOS EMPLEADOS EN LA CONSTRUCCION DEL PROCESADOR. Es importante señalar que el tipo de dispositivos que se pretende usar en la construcción del procesador son los de la familia TTL (hasta donde sea posible) ya que estos resultan ser los mas ampliamente conocidos por los estudiantes de las diferentes ramas que estan relacionadas - con la electrónica digital, además su adquisición resulta relativamente fácil y son muy resistentes a los malos tratos.

CAPITULO II.

INSTRUCCIONES DEL PROCESADOR.

Este capítulo tiene como finalidad hacer una descripción detallada de cada una de las instrucciones que serán incluidas en el repertorio de instrucciones del procesador.

Aquí serán descritas las instrucciones conforme a los grupos a que pertenecen -estos grupos ya fueron mencionados en la sección I.2.C.- y a cada grupo le serán aplicados los métodos de direccionamiento necesarios para que las instrucciones resulten mas eficientes.

II.1.- SIMBOLOGIA EMPLEADA EN LA DESCRIPCION DE LAS INSTRUCCIONES. Como auxiliar en la descripción de las instrucciones se empleará, siempre que sea posible, un lenguaje representativo de asignación de registros; este es un lenguaje muy simple y da una idea bastante clara de la función que realiza cada instrucción. Los símbolos empleados en este lenguaje serán los siguientes:

- A.- Esta letra servirá para designar al contenido del registro A del procesador.
- a.- Designa a la bandera de acarreo. Esta bandera será modificada de acuerdo al resultado de operaciones aritméticas y corrimientos; también será afectada por algunas operaciones lógicas (ANDR A,A, ORR A,A y EXR A,A).
- D.- Es un número binario de 7 bits -del bit cero que es el menos significativo hasta el bit seis-. El octavo bit (bit 7) indicará si se trata de un número positivo (bit 7=0) o de un número negativo representado en complemento a dos (bit 7=1).

La magnitud mas grande de un número positivo podrá ser -- por lo tanto;

$$127 \text{ base } 10 = 0111111 \text{ base } 2$$

y la magnitud mas grande de un número negativo será:

-128 base 10 = 10000000 complemento a dos

Las magnitudes mas pequeñas (excluyendo al cero) de un número positivo y uno negativo respectivamente serán:

1 base 10 = 00000001 base dos

-1 base 10 = 11111111 complemento a dos

por lo tanto se debe tener cuidado de indicar cuando se desea un desplazamiento positivo ($b_7=0$ seguido del desplazamiento deseado en base dos) y cuando uno negativo ($b_7=1$ seguido del desplazamiento deseado representado en complemento a dos). Esta representación solo será usada en las instrucciones de salto relativo.

F.- Designará al contenido del registro F.

M.- Designará la dirección de 16 bits de una localidad de memoria. (Ver FIGURA II.1).

(M).- Es el contenido de 8 bits de la localidad de memoria - cuya dirección es M.

Mi.- Es la dirección de 16 bits guardada en (M) y (Mmas1); - se empleará en el método de direccionamiento indirecto.

(Mi).- Es el contenido de 8 bits de Mi.

Mp.- Será la dirección de 16 bits de una localidad de memoria. Esta dirección -Mp- estará guardada en el contenido de las localidades de memoria (Mi) y (Mi mas 1).

N.- Designará a un número binario cualquiera que ocupe un campo de 8 bits.

PC.- Estas letras servirán para designar al contenido del registro contador de programa, este es un registro de propósito especial de 16 bits; en este registro se guarda la dirección de la localidad de memoria en la cual se encuentra la instrucción que va a ser ejecutada. Todas las instrucciones provocan que el PC sea incrementado a no ser que se indi

DIRECCION
(16 bits)

LOCALIDADES
DE MEMORIA

DESCRIPCION

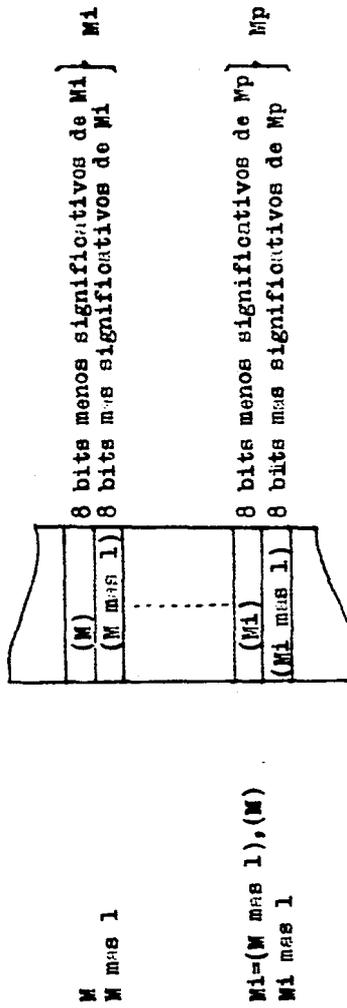


FIGURA II.1.a.- Regla General

FIGURA II.1.- Relación entre M, M_i y M_p.
a).- Regla General.
b).- Ejemplo Particular.

FIGURA II.1 (Continuación)

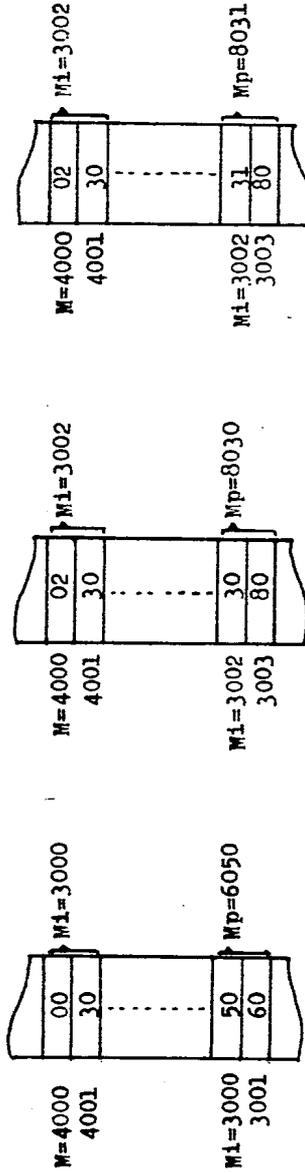


FIGURA II.1.b.- Ejemplo Particular. Notese que en la figura del centro ha sido cambiado el contenido de M respecto al contenido de M de la figura de la izquierda. Finalmente en la figura de la derecha ha sido cambiado el contenido de Mi respecto al que tenia en la figura del centro. Estos cambios son muy útiles en las instrucciones que emplean el direccionamiento indirecto y especialmente en las de salto programado.

FIGURA II.1.- Relación entre M, Mi y Mp.
 a).- Regla General.
 b).- Ejemplo particular.

que alguna otra cosa.

S.- Designará al contenido del registro S.

SL.- Es el contenido de los 8 bits menos significativos del registro S.

z.- Designa a la bandera de cero. Esta bandera será modificada por las instrucciones de los grupos aritmético, corrimientos y lógico de acuerdo al resultado de sus operaciones.

El símbolo \leftarrow denota transferencia; la transferencia se hará hacia el punto apuntado por la flecha.

Los símbolos \cdot , $+$, \oplus y $\bar{}$ denotarán las operaciones lógicas AND, OR, EXOR y NOT respectivamente.

La palabra mas denotará suma aritmética y el símbolo $-$ resta aritmética.

Al definir las instrucciones se usará el siguiente formato: Primeramente se dará el nombre o función de la instrucción y el método de direccionamiento empleado por ella; todo esto en lenguaje común. A partir del siguiente renglón se dará el nemotécnico en mayúsculas, el código de operación en hexadecimal y se indicará si se trata de una instrucción de uno, dos o tres bytes.

Finalmente se dará la descripción de la instrucción empleando la simbología descrita en las líneas anteriores.

II.2.- GRUPO DE INSTRUCCIONES DE TRANSFERENCIA DE OPERANDOS.

-Transferencia con el acumulador:

Carga el acumulador. Direccionamiento de Registros.

POP A 08 1 byte

A ← SL

Carga el acumulador. D. Inmediato.

CARN A,N 0A 2 bytes

A ← N

Carga el acumulador. D. Directo.

CARD A,M 0C 3 bytes

A ← (M)

Carga el acumulador. D. Indirecto.

CARI A,M 0E 3 bytes

A ← (Mi)

Descarga el acumulador. D. de Registros.

PUSH A 10 1 byte

SL ← A

Descarga el acumulador. D. Directo.

DESD A,M 12 3 bytes

(M) ← A

Descarga el acumulador. D. Indirecto.

DESI A,M 14 3 bytes

(Mi) ← SL

-Transferencia con el registro de banderas:

Carga el registro de banderas. D. de Registros.

POP F 16 1 byte

F ← SL

Descarga el registro de banderas. D. de Registros.

PUSH F 18 1 byte

SL ← F

II.3.- GRUPO DE INSTRUCCIONES ARITMETICAS.

-Comparación:

Compara al acumulador.	Direccionamiento Inmediato.	
CPN A,N	1A	2 bytes
	Evalúa A-N y a $A \leftarrow A$	
Compara al acumulador.	D. Directo.	
CPD A,M	1C	3 bytes
	Evalúa A-(M) y a $A \leftarrow A$	
Compara al acumulador.	D. Indirecto.	
CPI A,M	1E	3 bytes
	Evalúa A-(Mi) y a $A \leftarrow A$	

-Suma:

Suma al acumulador.	D. Inmediato.	
SUMN A,N	20	2 bytes
	$A \leftarrow A \text{ mas } N$	
Suma al acumulador.	D. Directo.	
SUMD A,M	22	3 bytes
	$A \leftarrow A \text{ mas } (M)$	
Suma al acumulador.	D. Indirecto.	
SUMI A,M	24	3 bytes
	$A \leftarrow A \text{ mas } (Mi)$	

-Suma con acarreo:

Suma al acumulador con acarreo.	D. Inmediato.	
SUMAN A,N	26	2 bytes
	$A \leftarrow A \text{ mas } N \text{ mas } a$	
Suma al acumulador con acarreo.	D. Directo.	
SUMAD A,M	28	3 bytes
	$A \leftarrow A \text{ mas } (M) \text{ mas } a$	
Suma al acumulador con acarreo.	D. Indirecto.	
SUMAI A,M	2A	3 bytes
	$A \leftarrow A \text{ mas } (Mi) \text{ mas } a$	

-Resta:

Resta al acumulador.	Direccionamiento Inmediato.	
RESN A,N	2C	2 bytes
	$A \leftarrow A-N$	
Resta al acumulador.	D. Directo.	
RESD A,M	2E	3 bytes
	$A \leftarrow A-(M)$	
Resta al acumulador.	D. Indirecto.	
RESI A,M	30	3 bytes
	$A \leftarrow A-(Mi)$	

-Resta con acarreo:

Resta al acumulador con acarreo.	D. Inmediato.	
RESAN A,N	32	2 bytes
	$A \leftarrow A-N-a$	
Resta al acumulador con acarreo.	D. Directo.	
RESAD A,M	34	3 bytes
	$A \leftarrow A-(M)-a$	
Resta al acumulador con acarreo.	D. Indirecto.	
RESAI A,M	36	3 bytes
	$A \leftarrow A-(Mi)-a$	

-Incremento

Incrementa al acumulador.	D. de Registros.	
INCRR A	38	1 byte
	$A \leftarrow A \text{ mas } 1$	
Incrementa a la memoria.	D. Directo.	
INCRD M	3A	3 bytes
	$(M) \leftarrow (M) \text{ mas } 1$	
Incrementa a la memoria.	D. Indirecto.	
INCRI M	3C	3 bytes
	$(Mi) \leftarrow (Mi) \text{ mas } 1$	

-Decremento:

Decrementa al acumulador. Direccionamiento de Registros.
 DECRR A 3E 1 byte
 $A \leftarrow A-1$

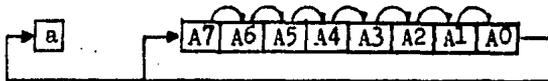
Decrementa a la memoria. D. Directo.
 DECRD M 40 3 bytes
 $(M) \leftarrow (M)-1$

Decrementa a la memoria. D. Indirecto.
 DECRI M 42 3 bytes
 $(Mi) \leftarrow (Mi)-1$

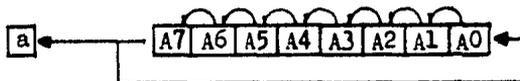
II.4.- GRUPO DE INSTRUCCIONES DE CORRIMIENTO.

-Rotaciones:

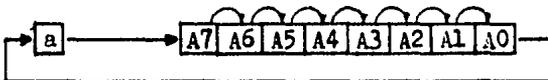
Rotación a la derecha. D. de Registros.
 RDRR 44 1 byte



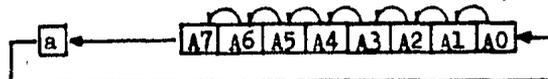
Rotación a la izquierda. D. de Registros.
 RIZR 46 1 byte



Rotación a la derecha con acarreo. D. de Registros.
 RDEAR 48 1 byte



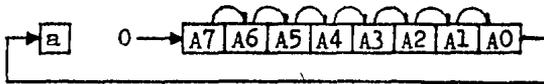
Rotación a la izquierda con acarreo. D. de Registros.
 RIZAR 4A 1 byte



-Corrimientos;

Corrimiento a la derecha. Direccionamiento de Registros.

CDER 4C 1 byte



Corrimiento a la izquierda. D. de Registros

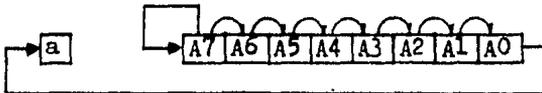
CIZR 4E 1 byte



-Desplazamientos;

Desplazamiento a la derecha. D. de Registros.

DDER 50 1 byte



Desplazamiento a la izquierda. D. de Registros.

DIZR 52 1 byte



II.5.- GRUPO DE INSTRUCCIONES LOGICAS.

-AND:

AND con el acumulador. D. de Registros.

ANDR A,A 54 1 byte

$$A \leftarrow A \cdot A \text{ además } a \leftarrow \bar{a} \text{ y } z \leftarrow \bar{z}$$

AND con el acumulador. D. Inmediato.

ANDN A,N 56 2 bytes

$$A \leftarrow A \cdot N$$

AND con el acumulador.	Direccionamiento Directo.	
ANDD A,M	58	3 bytes
	$A \leftarrow A \cdot (M)$	
AND con el acumulador.	D. Indirecto.	
ANDI A,M	5A	3 bytes
	$A \leftarrow A \cdot (M1)$	
-OR:		
OR con el acumulador.	D. de Registros.	
ORR A,A	5C	1 byte
	$A \leftarrow A+A$ además $a \leftarrow 0$ y $z \leftarrow 0$	
OR con el acumulador.	D. Inmediato.	
ORN A,N	5E	2 bytes
	$A \leftarrow A+N$	
OR con el acumulador.	D. Directo.	
ORD A,M	60	3 bytes
	$A \leftarrow A+(M)$	
OR con el acumulador.	D. Indirecto.	
ORI A,M	62	3 bytes
	$A \leftarrow A+(M1)$	
-OR EXCLUSIVA:		
EXOR con el acumulador.	D. de Registros.	
EXR A,A	64	1 byte
	$A \leftarrow A \oplus A$ además $a \leftarrow 0$ y $z \leftarrow 1$	
EXOR con el acumulador.	D. Inmediato.	
EXN A,N	66	2 bytes
	$A \leftarrow A \oplus N$	
EXOR con el acumulador.	D. Directo.	
EXD A,M	68	3 bytes
	$A \leftarrow A \oplus (M)$	
EXOR con el acumulador.	D. Indirecto.	
EXI A,M	6A	3 bytes

$$A \leftarrow A \oplus (M_i)$$

-Negación;

Niega al acumulador. Direcciónamiento de Registros.

NOTR A 6C 1 byte

$$A \leftarrow \bar{A}$$

-Complemento a dos;

Complementa al acumulador. D. de Registros.

CDOS A 6E 1 byte

$$A \leftarrow \bar{A} \text{ mas } 1$$

II.6.- GRUPO DE INSTRUCCIONES DE MODIFICACION DE PROGRAMA.

-Saltos;

Salto incondicional. D. Inmediato.

SALTON M 70 3 bytes

$$PC \leftarrow M$$

--Salto condicionado a la bandera de cero (z).

Salto relativo si no existe z. D. Inmediato.

SALRN NZ, \pm D 72 2 bytessi $z=0$ entonces $PC \leftarrow PC \pm D$ si $z \neq 0$ entonces $PC \leftarrow PC+1$

Salto inmediato si no existe z. D. Inmediato.

SALNN NZ, M 74 3 bytes

si $z=0$ entonces $PC \leftarrow M$ si $z \neq 0$ entonces $PC \leftarrow PC+1$

Salto programado si no existe z. D. Indirecto.

SALPI NZ, M 76 3 bytes

si $z=0$ entonces $PC \leftarrow M_p$ si $z \neq 0$ entonces $PC \leftarrow PC+1$

NOTA: Para las instrucciones del Grupo de Modificación de Programa el símbolo '+' denota suma aritmética y el símbolo '-' denota resta aritmética.

Salto relativo si existe z. Direccionamiento Inmediato.		
SALRN Z, [±] D	78	2 bytes
si z=1 entonces PC ← PC [±] D		
si z≠1 entonces PC ← PC+1		
Salto inmediato si existe z. D. Inmediato.		
SALNN Z,M	7A	3 bytes
si z=1 entonces PC ← M		
si z≠1 entonces PC ← PC+1		
Salto programado si existe z. D. Indirecto.		
SALPI Z,M	7C	3 bytes
si z=1 entonces PC ← Mp		
si z≠1 entonces PC ← PC+1		
--Salto condicionado a la bandera de acarreo (a).		
Salto relativo si no existe a. D. Inmediato.		
SALRN NA, [±] D	7E	2 bytes
si a=0 entonces PC ← PC [±] D		
si a≠0 entonces PC ← PC+1		
Salto inmediato si no existe a. D. Inmediato.		
SALNN NA,M	80	3 bytes
si a=0 entonces PC ← M		
si a≠0 entonces PC ← PC+1		
Salto programado si no existe a. D. Indirecto.		
SALPI NA,M	82	3 bytes
si a=0 entonces PC ← Mp		
si a≠0 entonces PC ← PC+1		
Salto relativo si existe a. D. Inmediato.		
SALRN A, [±] D	84	2 bytes
si a=1 entonces PC ← PC [±] D		
si a≠1 entonces PC ← PC+1		
Salto inmediato si existe a. D. Inmediato.		
SALNN A,M	86	3 bytes

si $a=1$ entonces $PC \leftarrow M$
 si $a \neq 1$ entonces $PC \leftarrow PC+1$
 Salto programado si existe a. Direccionamiento Indirecto.
 SALPI A,M 88 3 bytes
 si $a=1$ entonces $PC \leftarrow Mp$
 si $a \neq 1$ entonces $PC \leftarrow PC+1$
 -Saltos a subrutina ;
 Salto incondicional a subrutina. D. Inmediato.
 SUBRUTN M 8A 3 bytes
 $S \leftarrow PC$ además $PC \leftarrow M$
 --Salto a subrutina condicionado a la bandera de cero (z).
 Salto inmediato a subrutina si no existe z. D. Inmediato.
 SUBNN NZ,M 8C 3 bytes
 si $z=0$ entonces $S \leftarrow PC$ además $PC \leftarrow M$
 si $z \neq 0$ entonces $PC \leftarrow PC+1$
 Salto programado a subrutina si no existe z. D. Indirecto.
 SUBPI NZ,M 8E - 3 bytes
 si $z=0$ entonces $S \leftarrow PC$ además $PC \leftarrow Mp$
 si $z \neq 0$ entonces $PC \leftarrow PC+1$
 Salto inmediato a subrutina si existe z. D. Inmediato.
 SUBNN Z,M 90 3 bytes
 si $z=1$ entonces $S \leftarrow PC$ además $PC \leftarrow M$
 si $z \neq 1$ entonces $PC \leftarrow PC+1$
 Salto programado a subrutina si existe z. D. Indirecto.
 SUBPI Z,M 92 3 bytes
 si $z=1$ entonces $S \leftarrow PC$ además $PC \leftarrow Mp$
 si $z \neq 1$ entonces $PC \leftarrow PC+1$
 --Salto a subrutina condicionado a la bandera de acarreo (a).
 Salto inmediato a subrutina si no existe a. D. Inmediato.
 SUBNN NA,M 94 3 bytes
 si $a=0$ entonces $S \leftarrow PC$ además $PC \leftarrow M$

si $a \neq 0$ entonces $PC \leftarrow PC+1$

Regreso si existe a. Direccionamiento de Registros.
RETR A A4 1 byte.

si $a=1$ entonces $PC \leftarrow S$

si $a \neq 1$ entonces $PC \leftarrow PC+1$

Regreso de Interrupción. Direccionamiento de Registros.
RETIN A6 1 byte.

$PC \leftarrow S, F \leftarrow 3L, A \leftarrow SL$

II.7.- GRUPO DE INSTRUCCIONES DE FUNCION NULA.

Haz nada. Direccionamiento de Registros.
NADA A8 1 byte.

$PC \leftarrow PC+1$

Alto. Direccionamiento de Registros.
ALTO AB 1 byte.

$PC \leftarrow PC$

NOTA: Para las instrucciones del Grupo de Modificación de --
Programa el símbolo '+' denota suma aritmética y el -
símbolo '-' denota resta aritmética; la misma notación
es empleada en las instrucciones de la sección II.7.

CAPITULO III.

ARQUITECTURA DEL PROCESADOR.

El objetivo de este capítulo es el de presentar, a manera de bloques, la arquitectura del procesador.

Cada bloque arquitectónico del procesador será descrito de de tenidamente en este capítulo con excepción únicamente del -- bloque de control que, debido a su complejidad --mayor a la -- de los demás-- deberá ocupar por si solo todo un capítulo.

En la figura III.1 se muestra el diagrama de bloques que representa a la arquitectura del procesador, en ella se muestran los buses del procesador, los registros, la unidad lógica co-arithmética y la sección de control. En las secciones siguientes se dará una descripción funcional de cada uno de -- los bloques representados en la figura III.1 para que, a par tir de esta sea desarrollada la circuitería lógica correspon diente al funcionamiento de cada bloque.

III.1.- DESCRIPCION DE LOS BLOQUES DEL PROCESADOR. En esta sección se describen las características de cada bloque -- y se incluye a la descripción el diagrama lógico y la tabla de verdad correspondiente.

Para las tablas de verdad se ha de emplear la siguiente -- notación:

Subíndice i.- Denota el iésimo bit de un registro o bus; por ejemplo B_i es el iésimo bit de B.

Subíndice n.- Denota al nivel lógico existente antes de la -- última transición positiva de la señal de re-- loj.

Subíndice q.- Denota al nivel lógico existente antes de que las entradas indicadas fueran establecidas.

Z .- Denota un estado de alta impedancia; se debe -- tener cuidado de no confundirlo con el conteni

III.2

Bus de Direcciones (16 bits)

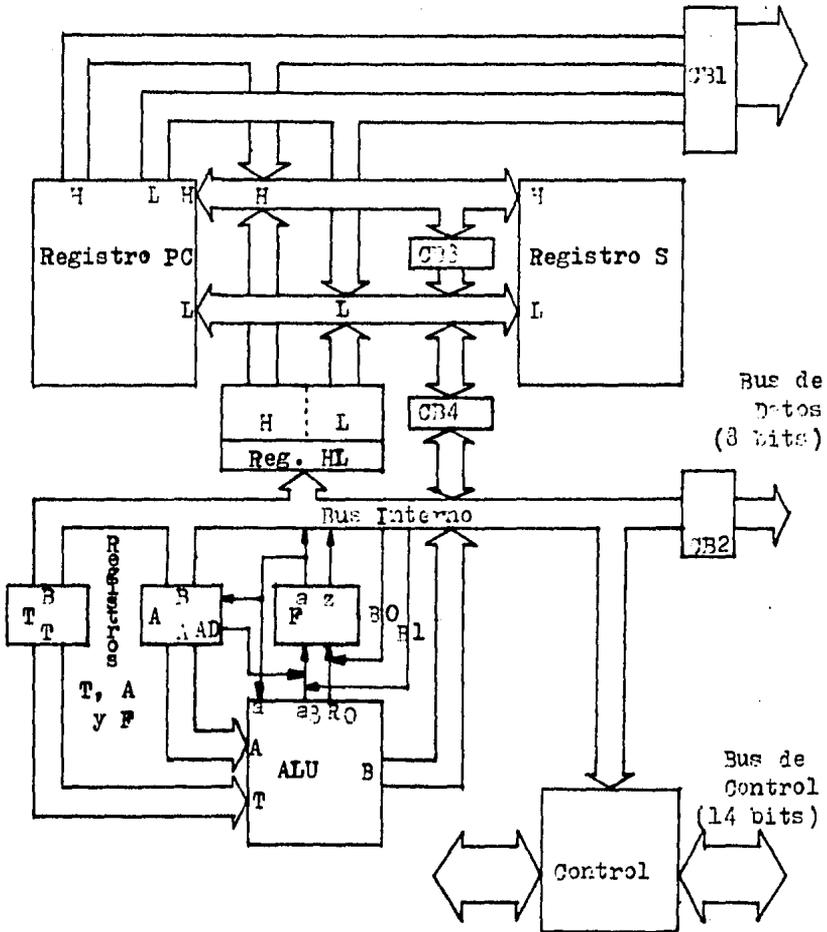


FIGURA III.1.- Arquitectura del Procesador.

do de la bandera de cero (z).

↑ .- Denota a la transición positiva de la señal de reloj. Es el momento en que ocurre un cambio de un nivel bajo a uno alto en la señal de reloj.

┌ .- Denota a cualquier nivel lógico a excepción de una transición positiva de la señal de reloj.

III.1.A.- LA ALU. La ALU es la unidad lógico-aritmética; como su nombre lo indica, en ella se realizarán operaciones aritméticas y lógicas de dos operandos de 8 bits cada uno, el resultado de la operación deberá ser también de 8 bits, además, la ALU deberá ser capaz de informar si en el resultado de la operación existe acarreo final o de si el resultado de la operación es cero. Las operaciones realizadas por la ALU serán las siguientes: Suma de dos operandos con acarreo de entrada y sin acarreo, resta de dos operandos con acarreo de entrada y sin acarreo, incremento y decremento de uno de los operandos, función AND de dos operandos, OR, EXOR, complemento a dos de uno de los operandos y negación (NOT). La ALU, debido a que compartirá el bus interno del procesador, deberá ser capaz de poner sus salidas en tercer estado.

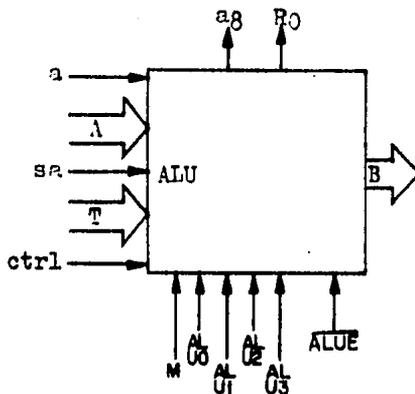


FIGURA III.1.A.- a) Bloque representativo de la ALU.

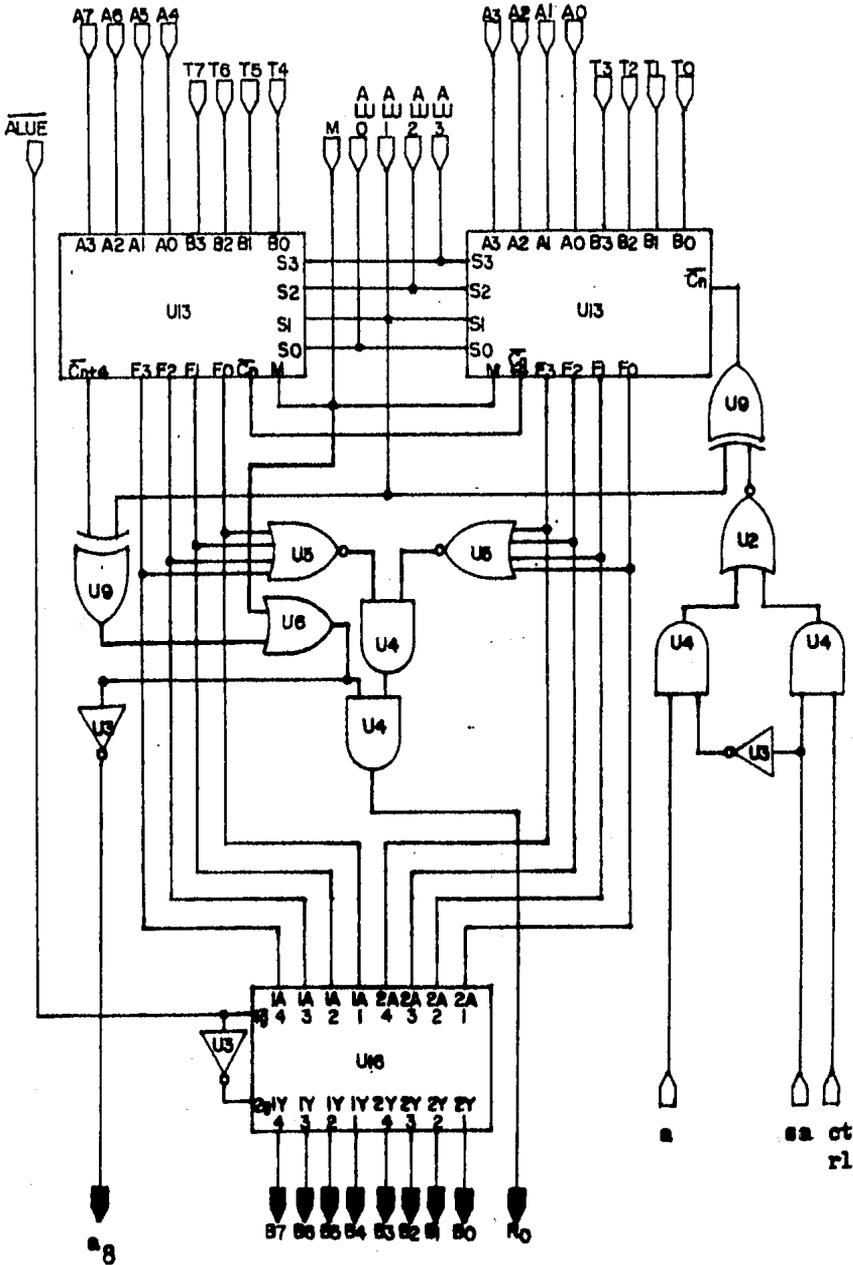


FIGURA III.1.A.- b) Diagrama lógico de la ALU.

ALU ₃ ALU ₂ ALU ₁ ALU ₀	ALUE = 0			ALUE = 1
	Lógicas M=1	Aritméticas M=0		
		sa=0, a=0 sa=1, ctrl=0	sa=0, a=1 sa=1, ctrl=1	
0 0 0 0	B=A	B=A	B=A+1	B=Z
0 0 1 1	B=0	B=0	B=-1	B=Z
0 1 1 0	B=A \bar{T}	B=A-T	B=A-T-1	B=Z
1 0 0 1	B=A \bar{T}	B=A+T	B=A+T+1	B=Z
1 0 1 1	B=A·T	B=A·T	B=A·T-1	B=Z
1 1 0 0	B=1	B=A+A	B=A+A+1	B=Z
1 1 1 0	B=A+T	B=(AorT)+A+1	B=(AorT)+A	B=Z
1 1 1 1	B=A	B=A	B=A-1	B=Z

TABLA III.1.A.- a) Tabla de verdad de la ALU.

Donde:

ALU₃, ..., ALU₀.- Líneas de selección de operación de la ALU.

M- Línea de selección del modo de operación de la ALU (M=1 para operaciones lógicas y M=0 para operaciones aritméticas).

ALUE ..- Línea de habilitación de la ALU hacia el bus interno del procesador.

sa- Línea de selección del origen del acarreo de entrada; sa=0 selecciona la bandera de acarreo 'a' y sa=1 selecciona un acarreo proveniente de una señal de control.

a- Línea de acarreo de entrada proveniente de la bandera de acarreo (Registro F).

ctrl ..- Línea de acarreo de entrada proveniente de la sección de control.

a_g ..- Línea de acarreo de salida de la ALU; esta línea se pone en uno lógico si existe acarreo final en el resultado de una operación de la ALU.

R₀ ..- Línea que indica si el resultado de la operación realizada en la ALU es cero; esta línea

III.6

se pone en un '1' lógico siempre y cuando el resultado de la operación sea cero y no exista acarreo final.

- A y T .- Operandos de la ALU de 8 bits cada uno guardados en el contenido del Registro Acumulador - y del Registro Temporal respectivamente.
- B .- Resultado de la operación realizada por la -- ALU; este resultado será habilitado o no por la señal ALUE en el bus interno del procesador para que de ahí sea mandado a su destino final.

Circuito Integrado (U)	Tipo	Cantidad Usada	Función
2	74LS02	1/4	4 Compuertas NOR de dos entradas
3	74LS04	3/6	6 Inversores (NOT)
4	74LS08	4/4	4 Compuertas AND de dos entradas
5	74LS25	2/2	2 Compuertas NOR de cuatro entradas
9	74LS86	2/4	4 Compuertas EXOR de -- entradas
13	74LS181	2/1	Unidad Lógico-Aritmética
16	74LS241	2/2	2 Buffers de tecer estado de 4 bits

TABLA III.L.A.- b) Dispositivos empleados en la construcción de la ALU.

III.1.B.- EL REGISTRO A. El Registro A es el registro acumulador del procesador, este es un registro de trabajo y en el estará almacenado uno de los operandos que intervienen en las operaciones de la ALU. Con excepción de la resta cuando es utilizada como método de comparación, todos los resultados de las operaciones lógico-aritméticas realizadas por el procesador serán finalmente guardadas en el acumulador. El acumulador podrá ser cargado en paralelo con un dato de 8 bits proveniente del bus interno del procesador, además, el acumulador deberá realizar los corrimientos, rotaciones y desplazamientos descritos en la sección II.4 por lo que deberá ser capaz de recibir carga en serie tanto por su extremo derecho como por el izquierdo proveniente de las siguientes fuentes, las cuales se podrán presentar solo una a la vez: - el registro de la bandera de acarreo, el bit menos significativo del acumulador, un cero lógico y el bit mas significativo del acumulador. Al realizar un corrimiento, desplazamiento o rotación, el acumulador deberá informar si el valor del bit desplazado al exterior era un uno lógico o un cero lógico.

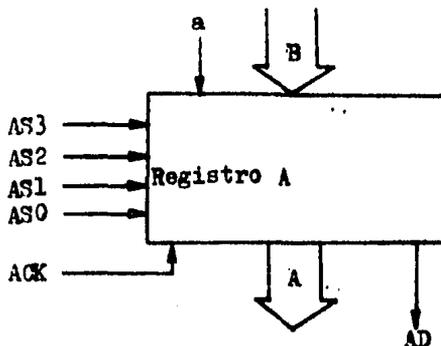


FIGURA III.1.B.- a) Bloque representativo del Registro A.

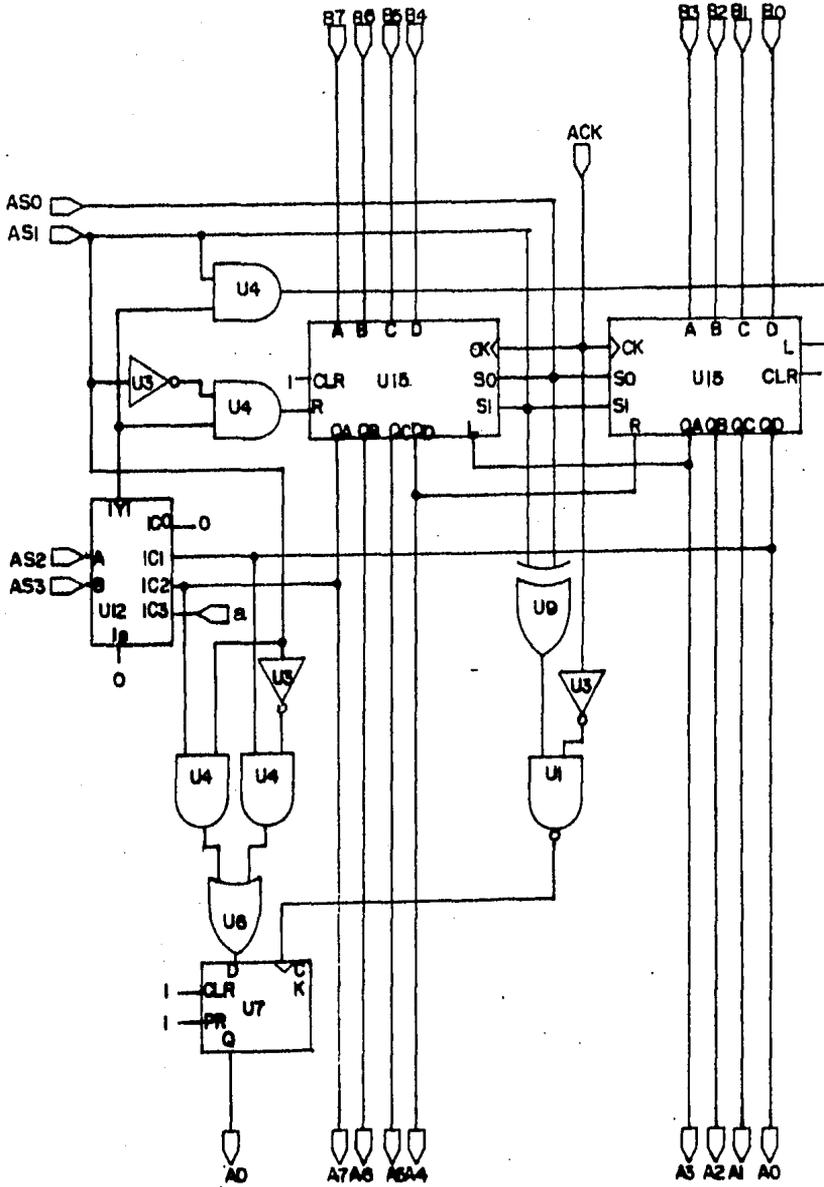


FIGURA III.1.B.- b) Diagrama lógico del Registro A.

AS ₃ AS ₂ AS ₁ AS ₀	ACK	Salidas	Bit Desplazado
		A7 A6 A5 A4 A3 A2 A1 A0	AD
X X 0 0	γ	A7 _Q A6 _Q A1 _Q A0 _Q	AD _Q
0 0 0 1	↑	0A7 _n A6 _n A2 _n A1 _n	A0 _n
0 1 0 1	↑	A0 _n A7 _n A6 _n A2 _n A1 _n	A0 _n
1 0 0 1	↑	A7 _n A7 _n A6 _n A2 _n A1 _n	A0 _n
1 1 0 1	↑	aA7 _n A6 _n A2 _n A1 _n	A0 _n
0 0 1 0	↑	A6 _n A5 _n A1 _n A0 _n 0	A7 _n
0 1 1 0	↑	A6 _n A5 _n A1 _n A0 _n A0 _n	A7 _n
1 0 1 0	↑	A6 _n A5 _n A1 _n A0 _n A7 _n	A7 _n
1 1 1 0	↑	A6 _n A5 _n A1 _n A0 _n a	A7 _n
γ γ 1 1	↑	B7 B6 B5 B4 B3 B2 B1 B0	AD _Q

TABLA III.1.3.- a) Tabla de Verdad del Registro A.

Donde:

- AS₃, ... AS₀ .- Líneas de selección del modo de control -- del Registro A.
- ACK .- Línea correspondiente a la señal de reloj - del Registro A.
- a .- Línea para cargar el dato serie 'a' (bande- ra de acarreo). Este registro también uti- liza un cero lógico y sus bits A0 y A7 co- mo datos de entrada en serie.
- AD .- Línea que indica el estado lógico del bit desplazado fuera del Registro A al efec- tuarse un corrimiento.
- A7A6....A1A0 .- Líneas de salida del Registro A
- B7B6....B1B0 .- Líneas de entrada para cargar en paralelo al Registro A.

NOTA: El modo de control del Registro A ($AS_3 \dots AS_0$) debe ser cambiado unicamente mientras la entrada del ACK esté en un nivel alto.

Circuito Integrado (U)	Tipo	Cantidad Usada	Función
1	74LS00	1/4	4 Compuertas NAND de 2 entradas
3	74LS04	3/6	6 Inversores (NOT)
4	74LS08	4/4	4 Compuertas AND de 2 - entradas
6	74LS32	1/4	4 Compuertas OR de 2 <u>en</u> <u>tradas</u>
7	74LS74	1/2	2 Flip-Flops tipo D
9	74LS86	1/4	4 Compuertas EXOR de 2 <u>entradas</u>
12	74LS153	1/2	2 Multiplexores de 4 a 1
15	74LS194	2/1	1 Registro de Corrimien <u>to</u> universal de 4 bits

TABLA III.1.B.- b) Lista de dispositivos empleados en la --- construcción del Registro A.

III.11

III.1.C.- EL REGISTRO T. El Registro T es un registro de almacenamiento temporal del procesador; este es un registro de 8 bits que podrá ser cargado del bus interno del procesador, en el será almacenado el otro de los operandos que intervienen en las operaciones realizadas por la ALU.

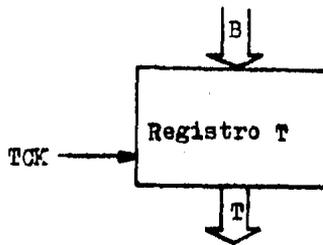


FIGURA III.1.C.- a) Bloque representativo del Registro T.

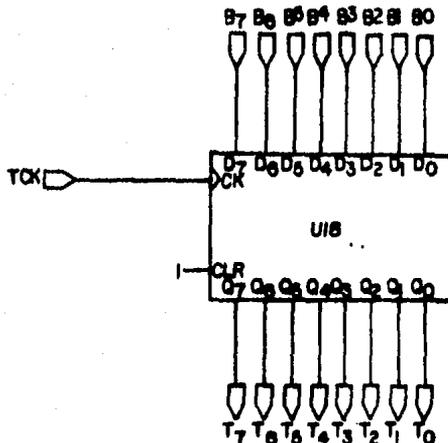


FIGURA III.1.C.- b) Diagrama lógico del Registro T.

Entradas B ₇ B ₆ B ₅ B ₄ B ₃ B ₂ B ₁ B ₀	TCK	Salidas T ₇ T ₆ T ₅ T ₄ T ₃ T ₂ T ₁ T ₀
B ₁	1	B ₁
Y		T ₁₀

TABLA III.1.C.- a) Tabla de verdad del Registro T.

Donde:

B₇B₆....B₁B₀ .- Líneas de entrada al Registro T provenientes del bus interno del procesador

TCK .- Línea correspondiente a la señal de reloj -- del Registro T.

T₇T₆....T₁T₀ .- Líneas de salida del Registro T.

Circuito In- tegrado (U)	Tipo	Cantidad Usada	Función
18	74LS273	1/1	8 Flip-flops tipo D

TABLA III.1.C.- b) Lista de dispositivos empleados en la --- construcción del Registro T.

III.1.D.- EL REGISTRO F. El Registro F es el registro de banderas del procesador; en él serán guardadas las banderas de cero (z) y de acarreo (a). El registro de banderas tiene tres posibles fuentes que son la ALU, el Registro A o el bus interno del procesador; estas fuentes deberán ser mutuamente excluyentes. El destino posible del Registro F podrá ser el bus interno del procesador, la ALU o el Registro A. Debido a que el Registro F compartirá sus líneas de salida con los otros elementos constituyentes del procesador, deberá tener la capacidad de poner en tercer estado las líneas que comparte en el bus interno. Además, el Registro F deberá ser capaz de realizar funciones de poner (SET) las banderas, limpiar (RESET) las banderas y complementar las banderas.

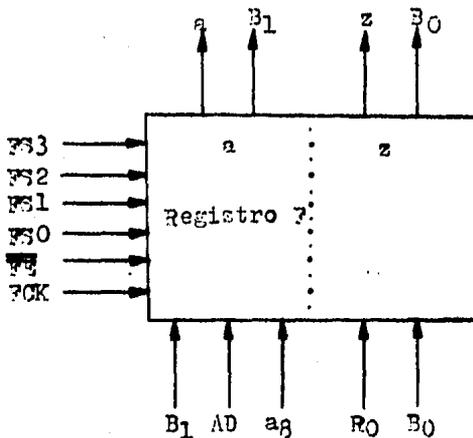


FIGURA III.1.D.- a) Bloque representativo del Registro F.

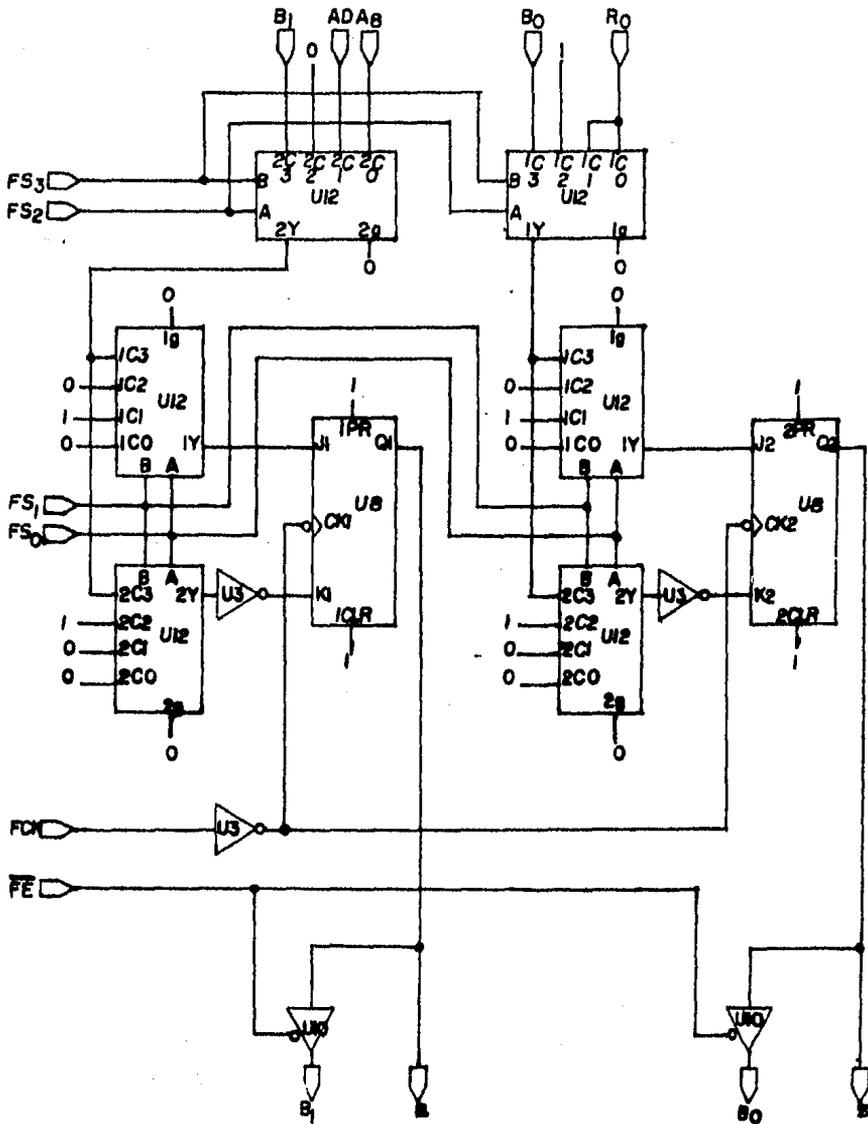


FIGURA III.1.D.- b) Diagrama lógico del Registro F.

FS ₃ FS ₂ FS ₁ FS ₀	FOK	a	z	FE=0		FE=1	
				B ₁	B ₀	B ₁	B ₀
0 0 1 1	↑	a ₈	R ₀	a ₈	R ₀	Z	Z
0 1 1 1	↑	AD	R ₀	AD	R ₀	Z	Z
1 0 1 1	↑	0	1	0	1	Z	Z
1 1 1 1	↑	B _{1n}	B _{0n}	B _{1n}	B _{0n}	Z	Z
X X 1 0	↑	a _n	z _n	a _n	z _n	Z	Z
X X 0 1	↑	a _n	z _n	a _n	z _n	Z	Z
X X 0 0	↑	0	0	0	0	Z	Z
X X X X	⊥	a _Q	z _Q	a _Q	z _Q	Z	Z

TABLA III.1.D.- a) Tabla de verdad del Registro F.

Donde:

FS₃..FS₀ .. Líneas de selección del modo de control del Registro F.

FOK .. Línea correspondiente a la señal de reloj del Registro F.

FE .. Línea que habilita el contenido del Registro F en el bus interno del procesador

B₁, AD, a₈.- Líneas a partir de las cuales puede ser alimentada la bandera de acarreo (a).

R₀, B₀.- Líneas a partir de las cuales puede ser alimentada la bandera de cero (z).

a, z .. Líneas que indican cual es el estado de las banderas de cero y acarreo respectivamente.

B₁, B₀.- Líneas que comunican al Registro F con con el bus interno del procesador. La bandera de acarreo estará comunicada con el bit B₁ y la de cero con el bit B₀ del bus interno del procesador.

Circuito In- tegrado (U)	Tipo	Cantidad Usada	Función
3	74LS04	3/6	6 Inversores (NOT)
8	74LS76	2/2	2 Flip-flops J-K
10	74LS125	2/4	4 Buffers de tercer es- tado de 1 bit
12	74LS153	6/2	2 Multiplexores de 4 a 1

TABLA III.1.D.- b) Lista de dispositivos empleados en la ---
construcción del Registro F.

III.1.E.- EL REGISTRO HL. El Registro HL será un registro de 16 bits que servirá como almacenamiento temporal a las direcciones de memoria; funcionalmente, el Registro HL será -- construido a partir de dos registros de 8 bits cada uno; el H y el L. En la parte L del registro serán guardados los 8 bits menos significativos de la dirección de memoria; los 8 bits más significativos serán guardados en la parte H.

Tanto la parte H como la L serán cargadas a partir del -- bus interno del procesador cargándose una sola a la vez. Las salidas del Registro HL podrán ser puestas en tercer estado debido a que este registro compartirá con otros elementos -- las líneas del bus interno del procesador.

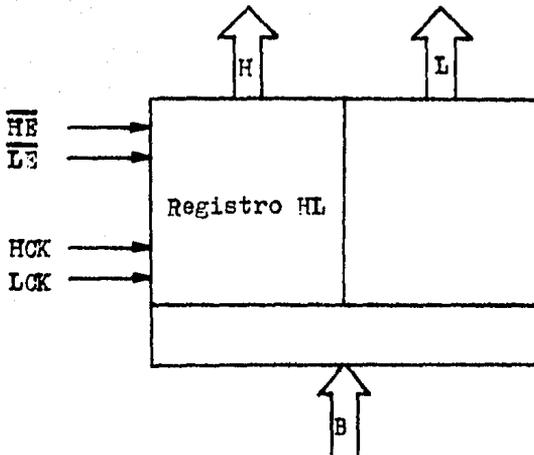


FIGURA III.1.E.- a) Bloque representativo del Registro HL.

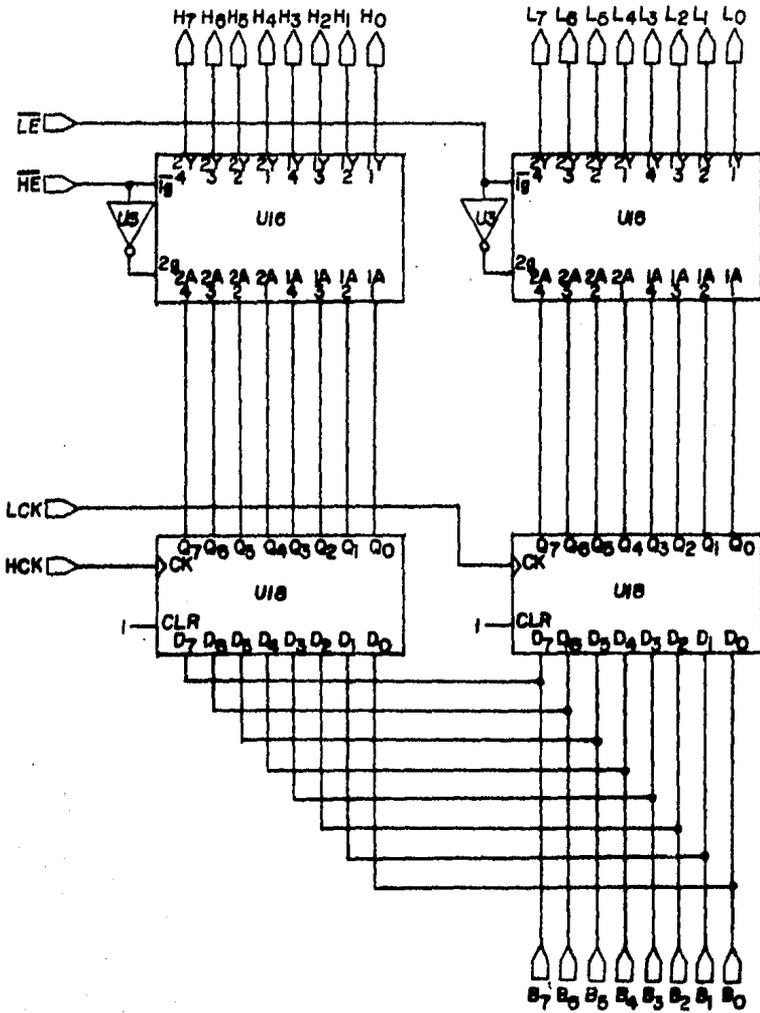


FIGURA III.1.8.- b) Diagrama lógico del Registro HL.

Entradas								Salidas														
B ₇	B ₆	B ₅	B ₄	B ₃	B ₂	B ₁	B ₀	HCK	LCK	\overline{HE}	\overline{LE}	H ₇	H ₆	...	H ₁	H ₀	L ₇	L ₆	...	L ₁	L ₀	
	B _i									0	0	H _i Q						B _i				
	B _i									0	0	B _i							L _i Q			
	X									0	0	H _i Q							L _i Q			
	X									0	1	H _i Q							Z			
	X									1	0	Z							L _i Q			
	X									1	1	Z							Z			

TABLA III.1.E.- a) Tabla de verdad del Registro HL.

Donde:

- \overline{HE} .- Línea que habilita o pone en tercer estado las salidas de la parte H del Registro HL.
- \overline{LE} .- Línea que habilita o pone en tercer estado las salidas de la parte L del Registro HL.
- HCK .- Línea correspondiente a la señal de reloj de la parte H del Registro HL.
- LCK .- Línea correspondiente a la señal de reloj de la parte L del Registro HL.
- B .- Líneas de entrada al Registro HL provenientes del bus interno del procesador
- H y L .- Líneas de salida del registro HL.

Circuito Integrado (U)	Tipo	Cantidad Usada	Función
3	74LS04	2/6	6 Inversores (NOT)
16	74LS241	4/2	2 Buffers de tercer estado de 4 bits
18	74LS273	2/1	8 Flip-flops tipo D

TABLA III.1.E.- b) Lista de dispositivos empleados en la construcción del Registro HL

III.1.F.- EL REGISTRO PC. El registro PC es el registro - contador de programa del procesador; este es un registro de 16 bits con la capacidad de cargarse a partir del bus interno del procesador (vía los Registros HL y S) y de incrementarse. Las salidas del Registro PC deberán tener la capacidad de ponerse en tercer estado para poder compartir el bus interno del procesador. Además el Registro PC deberá tener la capacidad de ser limpiado.

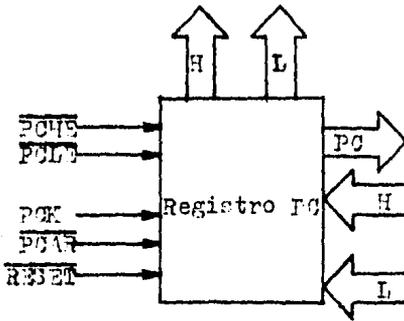


FIGURA III.1.F.- a) Bloque representativo del Registro PC.

Entradas		PCAR	RESET	RI ₀ +PCK	PCHE	PCLE	Salidas		
H ₇ ...H ₀	L ₇ ...L ₀						PC	H	L
X	X	1	1	↑	0	0	PC+1	HL + 1	
X	X	1	1	1	0	0	PC _{iQ}	H _{iQ}	L _{iQ}
H _i	L _i	0	1	X	0	0	HL	H _i	L _i
X	X	1	1	1	0	1	PC _{iQ}	H _{iQ}	Z
X	X	1	1	1	1	0	PC _{iQ}	Z	L _{iQ}
X	X	1	1	1	1	1	PC _{iQ}	Z	Z
X	X	X	0	X	X	X	0	X	X

TABLA III.1.F.- a) Tabla de verdad del Registro PC.

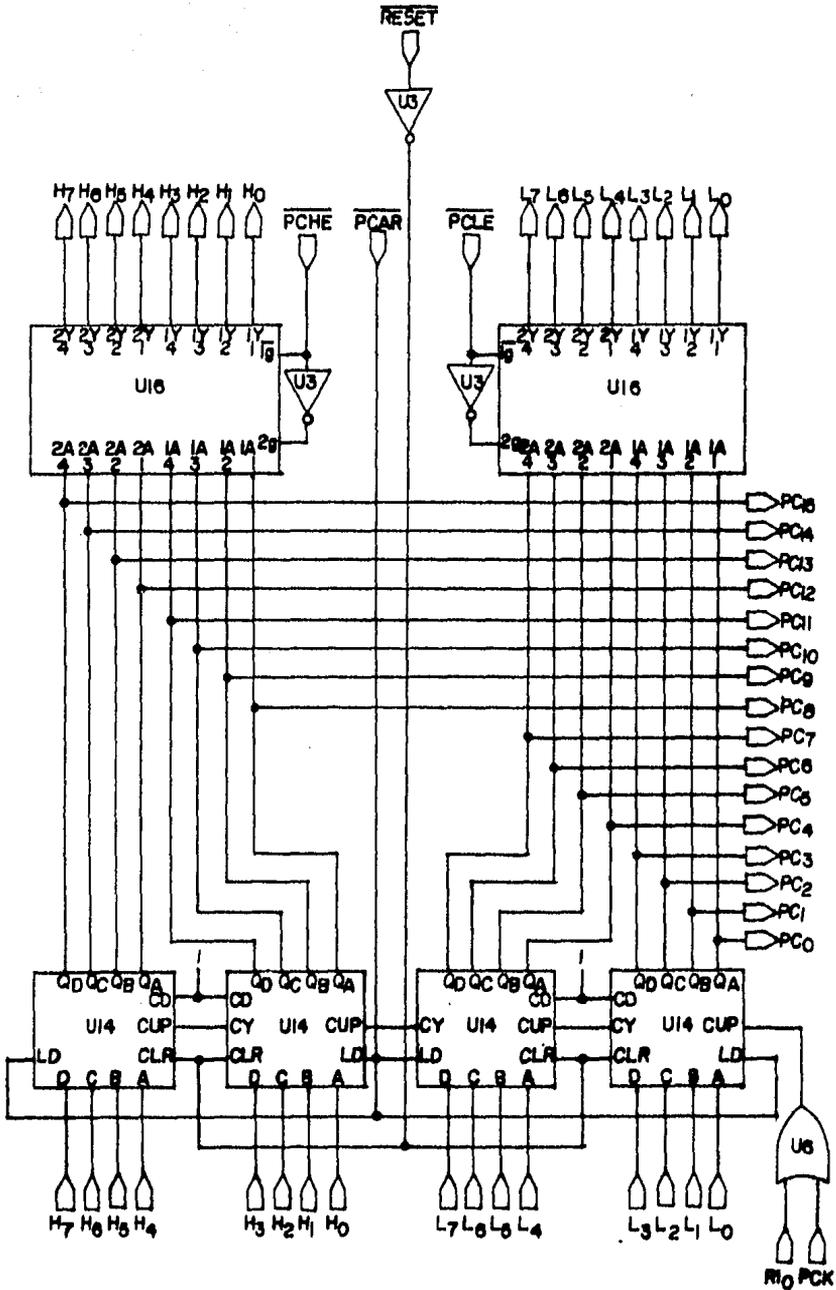


FIGURA III.1.F.- b) Diagrama lógico del Registro PC.

Donde:

- PCHE .- Línea que habilita a los 8 bits mas significativos del contenido del Registro PC en la parte H del bus interno del procesador.
- PCLE .- Línea que habilita a los 8 bits menos significativos del contenido del Registro PC en la parte L del bus interno del procesador.
- PCAR .- Línea de carga del Registro PC; cuando esta línea se activa, el contenido de las partes H y L del bus interno del procesador es cargado en el Registro PC.
- POK .- Línea correspondiente a la señal de reloj del - Registro PC.
- RI₀ .- Línea proveniente del Registro de Instrucción; sirve para habilitar o deshabilitar a la señal de reloj del Registro PC.
- RESET .- Línea para limpiar el contenido del Registro PC.
- H y L .- Líneas que intercomunican al Registro PC con el bus interno del procesador.
- PC₁₅...PC₀ .- Líneas que indican cual es el contenido del Registro PC al controlador del bus de direcciones CBI.

Circuito In- tegrado (U)	Tipo	Cantidad Usada	Función
3	74LS04	3/6	6 Inversores (NOT)
6	74LS32	1/4	4 Compuertas OR de 2 en <u>tr</u> adas
14	74LS193	4/1	Contador Universal de 4 bits
16	74LS241	4/2	2 Buffers de tercer es <u>t</u> ado de 4 bits.

TABLA III.1.F.- b) Lista de dispositivos empleados en la ---
construcción del Registro PC.

III.1.G.- EL REGISTRO S. El Registro S es un stack o apilamiento de 16 bits del tipo LIFO (último en entrar primero en salir). Funcionalmente, este registro estará constituido por un contador ascendente-descendente y por una RAM. Los 8 bits menos significativos de este registro podrán ser cargados directamente del bus interno del procesador -en este caso, los 8 bits mas significativos tendrán un estado intrasendente-; para cargar los 16 bits se usará el bus interno vía el Registro PC; la descarga de este registro se hará de la misma manera. Para evitar desperdicio en el espacio del apilamiento, el contador deberá ser cíclico para que de esta manera el apilamiento pueda comenzar en cualquier punto sin que esto afecte a la capacidad de apilamiento del registro.

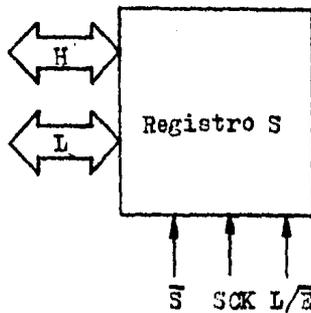


FIGURA III.1.G.- a) Bloque representativo del Registro S.

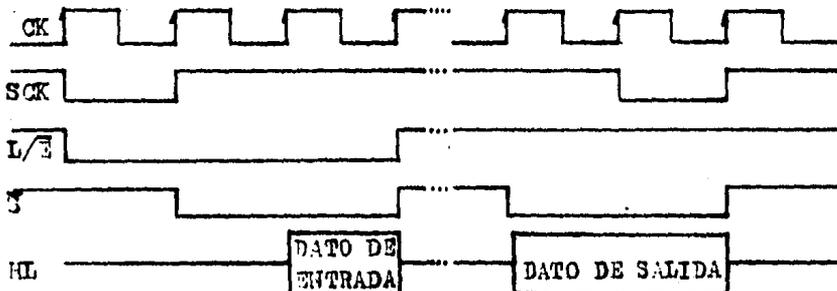


FIGURA III.1.G.- b) Diagrama de tiempo del Registro S.

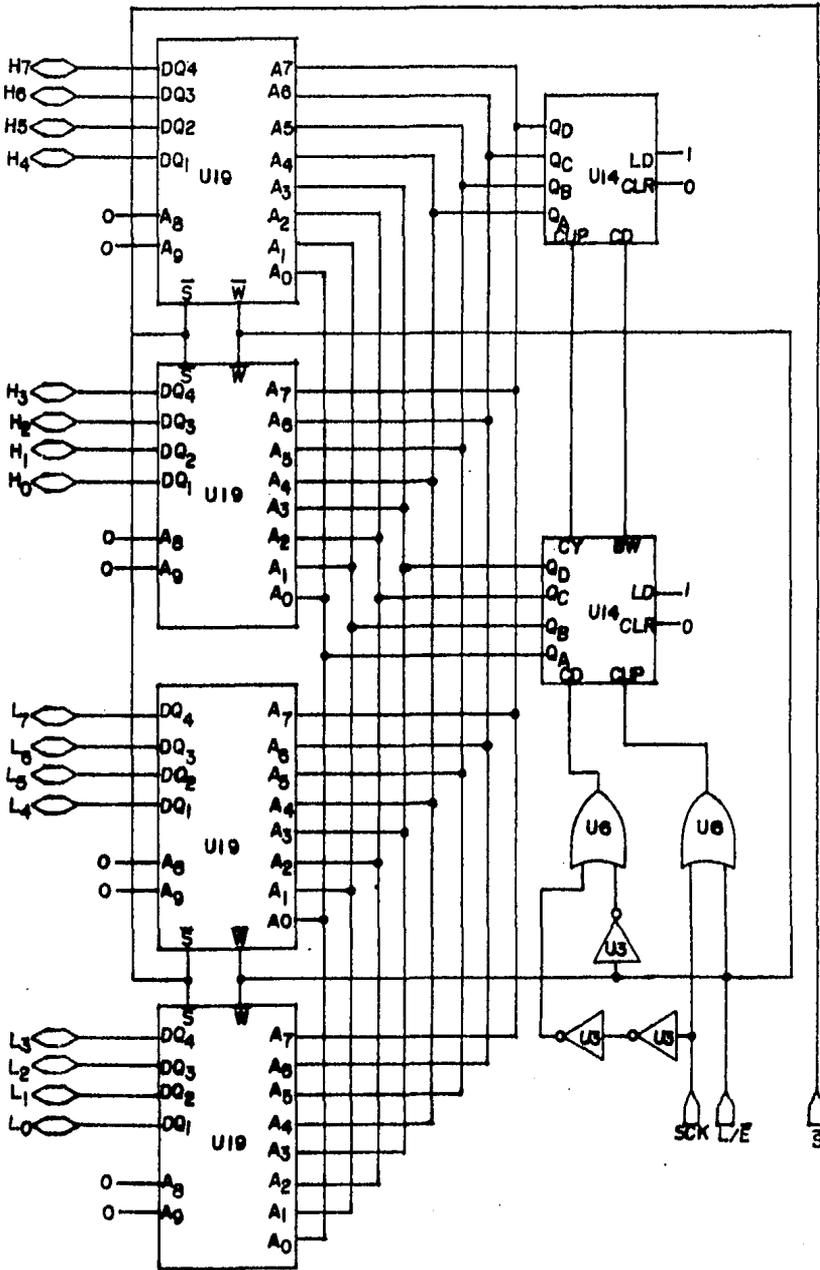


FIGURA III.1.G.- c) Diagrama lógico del Registro S.

Donde:

- CK .- Es la señal de reloj del procesador, en este --
diagrama es usada como referencia.
- 3CK .- Línea que sirve como señal de reloj al contador
interno del Registro S.
- L/\bar{S} .- Línea de lectura / escritura del Registro S.
- \bar{S} .- Línea de habilitación del Registro S.
- H y L .- Líneas de entrada-salida al Registro S prove---
nientes del bus interno del procesador.

Circuito In- tegrado (U)	Tipo	Cantidad Usada	Función
3	74LS04	3/6	6 Inversores (NOT)
6	74LS32	2/4	4 Compuertas OR de dos entradas
14	74LS193	2/1	Contador Universal de 4 bits
19	MCM2114	4/1	RAM estática 4bits X 1K

TABLA III.1.G.- a) Lista de dispositivos empleados en la ---
construcción del Registro S.

III.1.H.- CBI (Controlador del Bus 1). CBI es el buffer o restablecedor de 16 bits del bus de direcciones. Este buffer deberá tener la capacidad de poner al bus de direcciones en sentido de salida (hacia afuera del procesador) o bien en — tercer estado (alta impedancia).

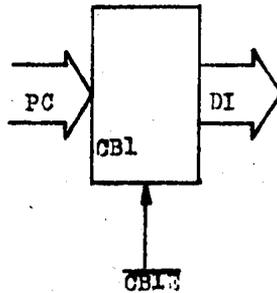


FIGURA III.1.H.- a) Bloque representativo del controlador de bus CBI

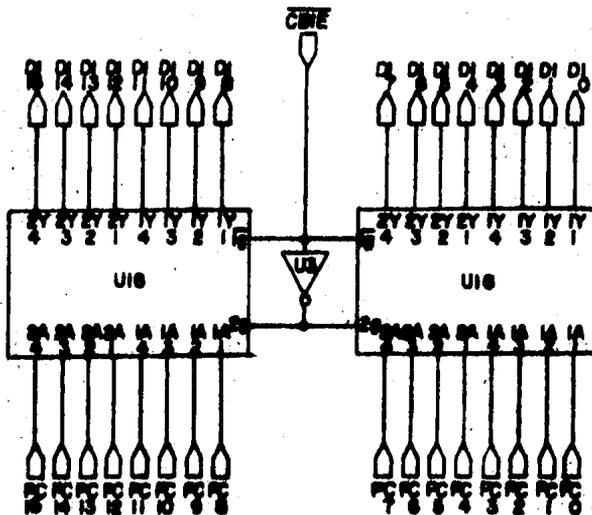


FIGURA III.1.H.- b) Diagrama lógico del controlador CBI.

Entrada	$\overline{\text{CBL}}$	Salida
PC		DI
PC_i	0	PC_i
X	1	Z

TABLA III.1.H.- a) Tabla de verdad del controlador CBL.

Donde:

- $\overline{\text{CBL}}$.- Línea que habilita o pone en tercer estado al controlador CBL
- PC .- Líneas de entrada al controlador CBL provenientes del Registro PC
- DI .- Líneas de salida del controlador CBL. Estas líneas corresponden a las del bus de direcciones del procesador.

Circuito Integrado (U)	Tipo	Cantidad Usada	Función
3	74IS04	1/6	6 Inversores (NOT)
16	74IS241	4/2	2 Buffers de tercer estado de 4 bits

TABLA III.1.H.- a) Lista de dispositivos empleados en la construcción del controlador CBL.

III.1.1.- CB2 (Controlador del Bus 2). CB2 es el buffer del bus de datos, este buffer será de 8 bits y podrá poner al bus de datos en los sentidos de entrada, salida o tercer estado.

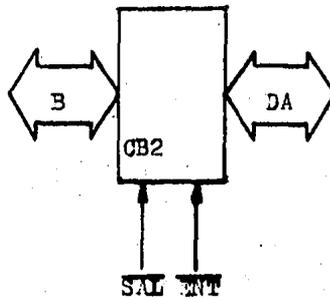


FIGURA III.1.1.- a) Bloque representativo del controlador CB2.

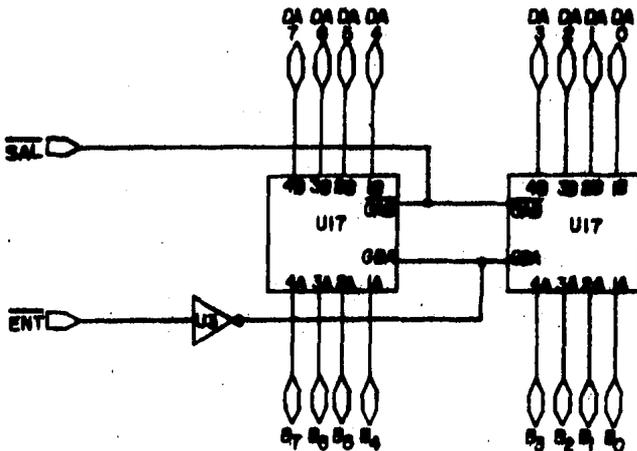


FIGURA III.1.1.- b) Diagrama lógico del controlador CB2.

ENT	SAL	Flujo de Datos	
		DE	HACIA
0	0	Estado no Permitido	
0	1	DA ₁	B ₁
1	0	B ₁	DA ₁
1	1	Tercer Estado	

TABLA III.1.I.- a) Tabla de Verdad del Controlador CB2.

Donde:

ENT y SAL .- Líneas que habilitan al controlador de bus - CB2 ya sea en dirección de entrada, salida - o tercer estado.

B₁ .- Líneas que intercomunican al controlador CB2 con el bus interno del procesador.

DA₁ .- Líneas de entrada y salida al controlador -- CB2; estas líneas corresponden a las líneas del bus de datos del procesador.

Circuito In-	Cantidad		Función
tegrado (U)	Tipo	Usada	
3	74LS04	1/6	6 Inversores (NOT)
17	74LS243	2/1	1 Buffer bidireccional - de 4 bits

TABLA III.1.I.- b) Lista de dispositivos empleados en la --- construcción del controlador CB2.

III.1.J.- CB3 (Controlador del Bus 3). Este es un buffer unidireccional de 8 bits con salidas de tercer estado que permite que la parte H del bus interno pueda ser comunicada con la parte L. La comunicación solo podrá realizarse de H hacia L y en caso de no ser necesaria esta comunicación, --CB3 permanecerá en tercer estado.

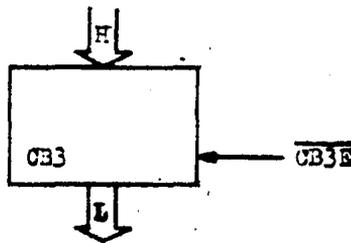


FIGURA III.1.J.- a) Bloque representativo del controlador - CB3

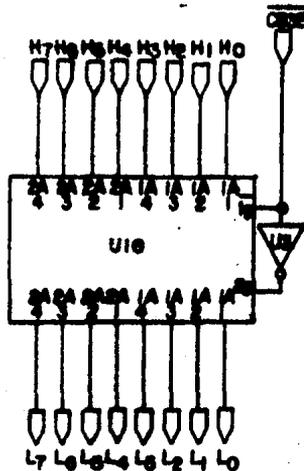


FIGURA III.1.J.- b) Diagrama lógico del controlador CB3

Entrada	$\overline{CB3E}$	Salida
H		L
H ₁	0	L ₁
X	1	Z

TABLA III.1.J.- a) Tabla de verdad del controlador CB3.

Donde:

- $\overline{CB3E}$.- Línea que habilita o pone en tercer estado al controlador CB3.
- H .- Líneas de entrada al controlador CB3 provenientes de la parte H del bus interno del procesador.
- L .- Líneas de salida del controlador CB3. Estas líneas corresponden a la parte L del bus interno del procesador.

Circuito Integrado (U)	Tipo	Cantidad Usada	Función
3	74LS04	1/6	6 Inversores (NOT)
16	74LS241	2/2	2 Buffers de tercer estado de 4 bits

TABLA III.1.J.- b) Lista de dispositivos empleados en la construcción del controlador CB3.

III.1.K.- CB4 (Controlador del Bus 4).- Este es un buffer bidireccional de 8 bits con salidas de tercer estado que permitirá llevar o traer datos a la parte L del bus interno.

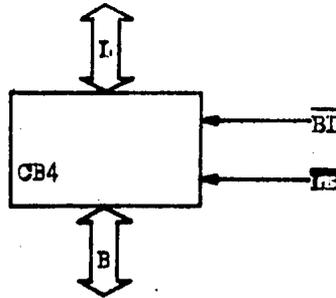


FIGURA III.1.K.- a) Bloque representativo del controlador CB4.

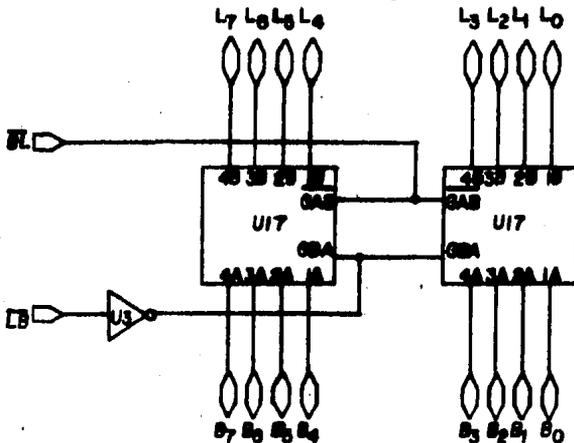


FIGURA III.1.K.- b) Diagrama lógico del controlador CB4.

\overline{LB}	\overline{BJ}	Flujo de Datos	
		DE	HACIA
0	0	Estado no Permitido	
0	1	L_1	B_1
1	0	B_1	L_1
1	1	Tercer Estado	

TABLA III.1.K.- a) Tabla de verdad del controlador CB4

Donde;

\overline{LB} y \overline{BL} .- Líneas que habilitan al controlador en dirección de entrada, salida o tercer estado.

L .- Líneas que intercomunican al controlador CB4 con la parte L del bus interno del procesador.

B .- Líneas que intercomunican al controlador CB4 con el bus interno del procesador.

Circuito Integrado (U)	Tipo	Cantidad Usada	Función
3	74LS04	1/6	6 Inversores (NOT)
17	74LS243	2/1	Buffer bidireccional de 4 bits

TABLA III.1.K.- b) Lista de dispositivos empleados en la construcción del controlador CB4.

III.1.L.- LA SECCION DE CONTROL. La sección de control deberá generar las señales que gobiernan el funcionamiento de cada bloque del procesador y las señales que emplea el procesador para comunicarse con otros dispositivos por medio del bus de control del procesador. Esta sección será ampliamente tratada en el siguiente capítulo.

Circuito In- tegrado	Tipo	Función
U1	74LS00	4 Compuertas NAND de 2 entradas.
U2	74LS02	4 Compuertas NOR de 2 entradas.
U3	74LS04	6 Inversores (NOT).
U4	74LS08	4 Compuertas AND de 2 entradas.
U5	74LS25	2 Compuertas NOR de 4 entradas.
U6	74LS32	4 Compuertas OR de 2 entradas.
U7	74LS74	2 Flip-flops tipo D.
U8	74LS76	2 Flip-flops J-K.
U9	74LS86	4 Compuertas EXOR de 2 entradas.
U10	74LS125	4 Buffers de 3er. estado de 1 bit.
U12	74LS153	2 Multiplexores de 4 a 1.
U13	74LS181	1 Unidad Lógico-aritmética de 4 bits.
U14	74LS193	1 Contador Universal de 4 bits.
U15	74LS194	1 Registro de Corrimiento de 4 bits.
U16	74LS241	2 Buffers de 3er. estado de 4 bits.
U17	74LS243	1 Buffer bidireccional de 4 bits.
U18	74LS273	8 Flip-flops tipo D.
U19	MCM-2114	1 RAM estática de 4 bits X 1 K.

TABLA III.1.L.- Lista de dispositivos empleados en la construcción de los circuitos mostrados en el Capítulo III.

C A P I T U L O IV.

SECCION DE CONTROL DEL PROCESADOR.

La Sección de Control de un procesador es la encargada de habilitar las señales que intervienen y hacen posible que el procesador realice alguna operación sencilla como por ejemplo cargar el Registro Temporal (T) a partir del bus interno del procesador. Para realizar esta operación la Sección de Control deberá generar las señales de control necesarias en el orden adecuado y durante el tiempo preciso para poder completarla; así por ejemplo el controlador debería generar una señal que pusiera un dato proveniente de X fuente en el bus interno del procesador y mantenerlo estable mientras es generada la señal que se encarga de cargar al Registro T; una vez cargado el Registro T lo mas conveniente sería deshabilitar el dato que ya ha sido cargado y disponerse a realizar otra operación. Es seguro que si esta operación es realizada en la secuencia mencionada, la operación será realizada con éxito. ¿Pero que pasaría si en vez de seguir la secuencia adecuada fuera realizada otra?. Por ejemplo generar primero la señal que carga al Registro T y después poner el dato en el bus interno del procesador o bien poner el dato en el bus y permitir que este cambie aleatoriamente durante el tiempo que el Registro T emplea para ser cargado. Seguramente la operación no sería realizada con éxito ya que el Registro T sería cargado con un dato desconocido y generado aleatoriamente. Es el peor de los casos una secuencia inadecuada podría ordenar poner dos datos diferentes en el bus interno y provocar con esto un corto-circuito que seguramente dañaría en forma severa al procesador. Todas las operaciones que realiza un procesador, por complejas que estas parezcan no son mas que el resultado de una serie de ordenes mucho mas senci

IV.2

llas que en una secuencia adecuada han sido generadas por la Sección de Control del procesador.

En este capítulo se hará una descripción de la forma en la cual se habrá de construir un controlador que gobierne el comportamiento de los bloques arquitectónicos del procesador en forma tal que puedan ser realizadas las instrucciones y operaciones que ya han sido descritas en los capítulos anteriores.

IV.1.- ARQUITECTURA DEL CONTROLADOR. En la FIGURA IV.1 se puede observar la Arquitectura del Controlador. En ella se ven las líneas de entrada del bus de control ($\overline{\text{RBUS}}$, $\overline{\text{INT}}$, $\overline{\text{RESET}}$, $\overline{\text{ESP}}$ y $\overline{\text{CK}}$), el bus interno del procesador (a través del cual es cargado el Registro RI), las líneas provenientes de las banderas de acarreo (a) y cero (z) (entrando a la lógica de condición), los Registros del Controlador y las líneas de salida del controlador que serán, a final de cuentas, las encargadas de activar tanto a los bloques del procesador como a los bloques del mismo controlador y a las líneas de salida pertenecientes al Bus de Control (siete que corresponden a $\overline{\text{CBUS}}$, $\overline{\text{ACEX}}$, $\overline{\text{ESC}}$, $\overline{\text{LRC}}$, $\overline{\text{BINS}}$, $\overline{\text{RINT}}$ y $\overline{\text{ALTO}}$). Los bloques arquitectónicos que conforman al controlador y su funcionamiento serán descritos en las secciones siguientes.

IV.1.A.- LA PROM DE CONTROL. La Prom de Control será construida a partir de 4 EPROM's MCM 2716 con lo cual se contará con 32 bits que serán los encargados de generar las ordenes que requieren los bloques arquitectónicos para realizar todas las funciones que ya han sido descritas. Estos 32 bits estarán divididos en dos campos principales: El de Control de Proceso y el de Control del Controlador.

IV.1.A.1.- CAMPO DE CONTROL DE PROCESO. Este campo ocupará 16 bits y generará las señales necesarias para controlar

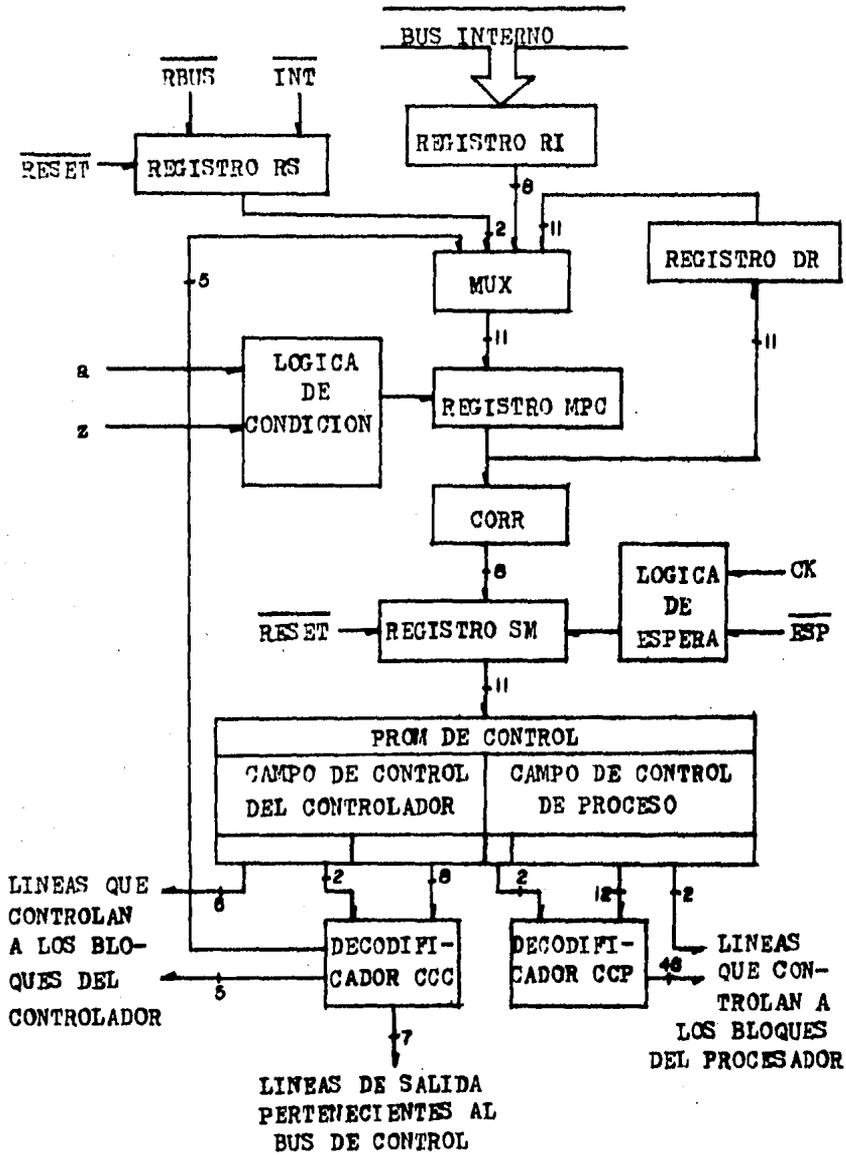


FIGURA IV.1.- Arquitectura del Controlador

IV.4

a los bloques arquitectónicos tales como la ALU, el Registro A, el Registro T, el Registro F, el Registro HL, el Registro PC, el Registro S, el Registro de Instrucción y los controladores de los buses CB1, CB2, CB3 y CB4. Debido a que sería muy costoso y poco práctico contar con un bit para cada una de las líneas de control con que cuenta cada bloque (para lo cual se necesitarían 38 bits en total para este campo) el Campo de Control de Proceso estará codificado de una manera sencilla y apropiada a los fines que persigue el diseño de este procesador. De esta forma, el Campo de Control de Proceso quedará dividido en dos subcampos; el de Selección y el de Ejecución como se muestra en la FIGURA IV.1.A.1.

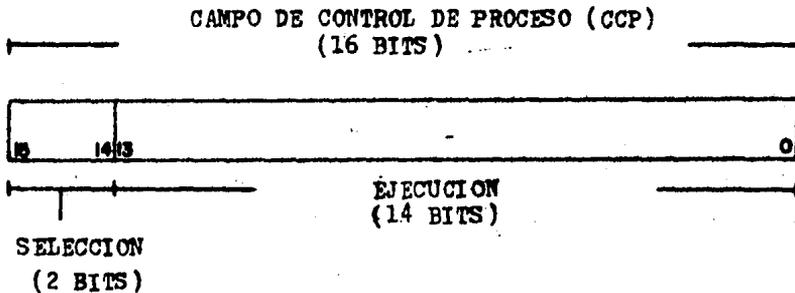


FIGURA IV.1.A.1.- División del Campo de Control de Proceso.

Con un subcampo de Selección de 2 bits será posible (mediante un decodificador en el subcampo de Ejecución) seleccionar cuatro grupos de bits de ejecución diferentes con lo que los bits del subcampo de Ejecución se multiplicarían y de esta manera sería posible generar todos los bits necesarios para controlar los bloques arquitectónicos mencionados con anterioridad de una manera sencilla. Los cuatro grupos de Ejecución y sus valores de Selección asociados se dan en la TABLA IV.1.A.1.

	Bits del CCP	Líneas que se han de controlar			
		Grupo 1	Grupo 2	Grupo 3	Grupo 4
Ejecución	0	ACK	ACK	ACK	ACK
	1	$\overline{\text{ALUE}}$	$\overline{\text{ALUE}}$	$\overline{\text{ALUE}}$	$\overline{\text{ALUE}}$
	2	$\overline{\text{WE}}$	SCK	$\overline{\text{CBLE}}$	ALU3
	3	$\overline{\text{LE}}$	L/E	PCK	ALU2
	4	HCK	$\overline{\text{S}}$	$\overline{\text{ENT}}$	ALU1
	5	LCK	$\overline{\text{PCHE}}$	SAL	ALU0
	6	$\overline{\text{PCAR}}$	$\overline{\text{PCLE}}$	TCK	M
	7	$\overline{\text{PCHE}}$	$\overline{\text{LB}}$	RICK	sa
	8	$\overline{\text{PCLE}}$	$\overline{\text{BL}}$	HCK/AS3	ctrl
	9	$\overline{\text{CB3E}}$	$\overline{\text{PE}}$	LCK/AS2	FS3
	10	$\overline{\text{LB}}$	PCK	AS1	FS2
	11	TCK	$\overline{\text{PCAR}}$	AS0	FS1
	12	X	X	FS3	FS0
13	X	X	PCK	PCK	
Selección	14	0	1	0	1
	15	0	0	1	1

TABLA IV.1.A.1.- Los cuatro grupos de líneas controladas por el subcampo de Ejecución y valores de Selección asociados.

En esta tabla se puede observar que cuando el subcampo de Selección sea 00, el significado de los bits del subcampo de Ejecución será el mostrado en la columna correspondiente al grupo número 1 y por lo tanto el decodificador deberá encargarse de que las líneas ahí indicadas sean las que entren en funcionamiento. Una discusión análoga puede hacerse para los casos en los cuales el subcampo de Selección adquiera los valores 01, 10 y 11.

Una vez hechas las consideraciones anteriores se puede proceder a construir un decodificador que tenga como entra--

das los bits CCP_2 a CCP_{15} y como salidas las líneas que se indican en las columnas correspondientes a los grupos 1, 2, 3 y 4 de la TABLA IV.1.A.1. El hecho de que solo se codifiquen los bits CCP_2 a CCP_{15} puede observarse en la TABLA IV.1.A.1 en la cual se observa que los bits CCP_0 y CCP_1 siempre tienen el mismo significado en cualquiera de los grupos de dicha tabla. De acuerdo a esto, el decodificador se puede -- construir en la manera que se indica en la FIGURA IV.1.A.2 -- donde se han de utilizar buffers de tercer estado de forma -- tal que cuando un grupo es seleccionado (habilitado), los de -- mas se deshabilitan; con esto se realiza la decodificación -- de una manera muy sencilla y se reduce considerablemente la cantidad de lógica empleada en la decodificación.

En el contenido de las EPROM's de control serán grabados (programados) los estados lógicos que deben adquirir las líneas de control para realizar alguna operación sencilla, que en este caso será llamada microoperación; así, una microoperación estará compuesta por una serie de estados lógicos que serán interpretados como ordenes de control por las líneas -- que poseen los diferentes bloques a los que se ha de controlar; la forma o regla mediante la cual las ordenes se han de ejecutar está dada por la tabla de verdad correspondiente a cada bloque. Las microoperaciones entonces se pueden clasifi -- car de dos formas: Microoperaciones de la Sección de Proceso y microoperaciones de la Sección de Control; las primeras se -- rán realizadas con la ayuda del Campo de Control de Proceso y las segundas con la ayuda del Campo de Control del Controlador que se define a continuación.

IV.1.A.2.- CAMPO DE CONTROL DEL CONTROLADOR. Al igual que el campo definido anteriormente, este es también de 16 bits -- y generará las señales o líneas necesarias para controlar --

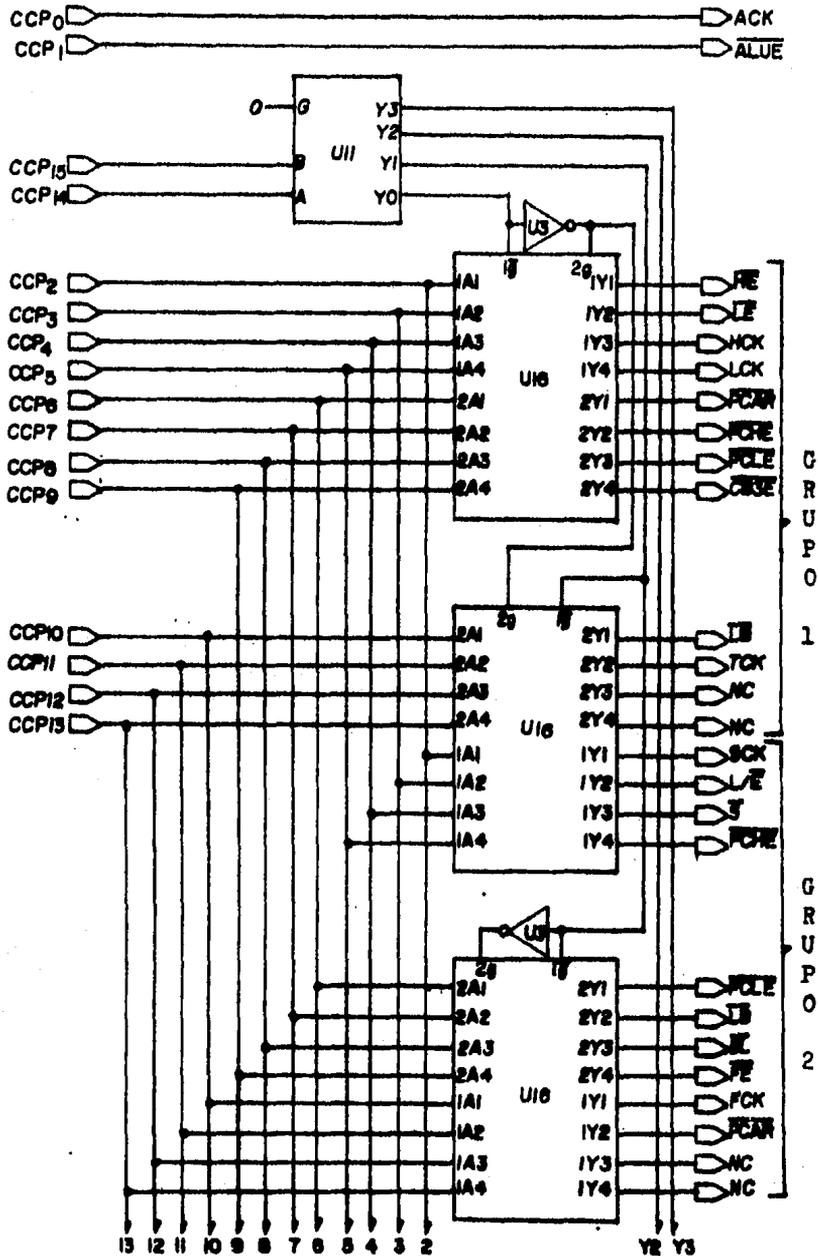
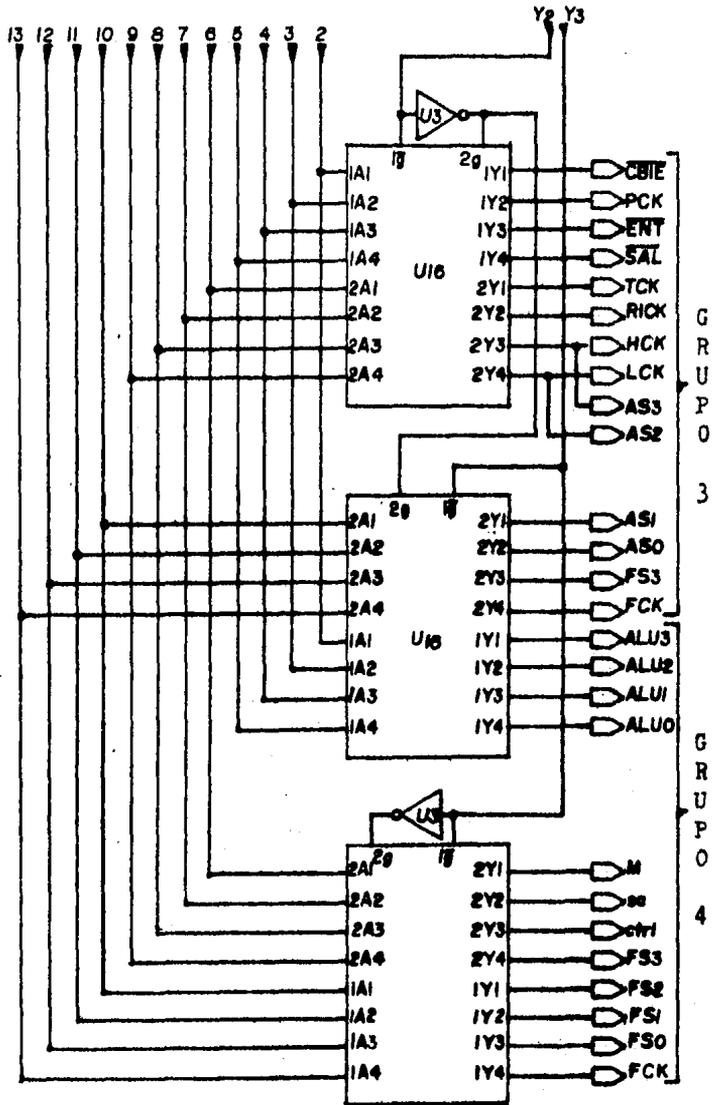


FIGURA IV.1.A.2.- Diagrama Lógico del Decodificador del Campo de Control de Proceso.



NC= No Conectar

FIGURA IV.1.A.2.- (continúa).

los bloques de la Sección de Control tales como el Registro de Rutinas de Servicio (RS), el Registro de Direcciones de Regreso (DR), el Registro Contador de Microprograma (MPC), el Registro Secuenciador de Microoperación (SM), la Lógica de Condición, la Lógica de Espera y las señales de salida — del Bus de Control $\overline{\text{CBUS}}$ (concesión del bus), $\overline{\text{ACEX}}$ (acceso al exterior), $\overline{\text{ESC}}$ (escritura), $\overline{\text{LEC}}$ (lectura), $\overline{\text{BINS}}$ (búsqueda de instrucción), $\overline{\text{RINT}}$ (reconocimiento de interrupción) y $\overline{\text{ALTO}}$ (ejecución de una instrucción ALTO). Este campo estará dividido en dos subcampos; el de Secuencia y el de Modificación como se muestra en la FIGURA IV.1.A.3.

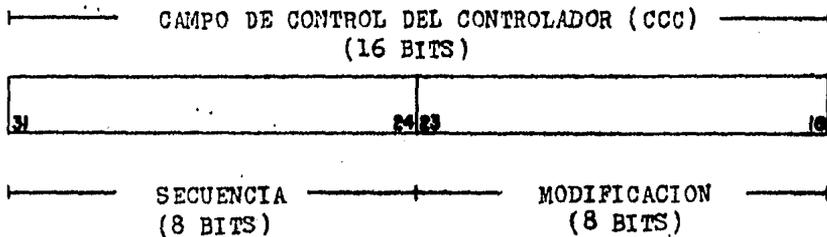


FIGURA IV.1.A.3.- División del Campo de Control del Controlador.

El subcampo de Secuencia será de 8 bits y no estará codificado; sus bits darán las ordenes para activar las líneas que se indican en la TABLA IV.1.A.2.

El subcampo de Modificación también será de 8 bits y estará codificado de acuerdo a los estados lógicos que se presenten en los bits 25 y 24 del subcampo de Secuencia; estos bits corresponden a las líneas $1/0$ y $\overline{\text{CBUS}}$ respectivamente. La forma en que deberán ser interpretados los bits del subcampo de Modificación y los valores asociados para $1/0$ y $\overline{\text{CBUS}}$ se dan

	Bits del CCC.	Líneas que se han de Controlar.
	Subcampo de Secuencia	31
30		$\overline{\text{DRC}}$
29		MS1
28		MS0
27		MPCK
26		$\overline{\text{SMC}}$
25		1/0
24		$\overline{\text{CBUS}}$

TABLA IV.1.A.2.- Líneas controladas por los bits del subcampo de Secuencia.

en la TABLA IV.1.A.3.

	Bits del CCC	Líneas que se han de Controlar			
	Subcampo de Modificación	23	$\overline{\text{ACEX}}$	SCL	$\overline{\text{ACEX}}$
22		$\overline{\text{ESC}}$	SCO	$\overline{\text{ESC}}$	SCO
21		$\overline{\text{LEC}}$	EX	$\overline{\text{LEC}}$	EX
20		$\overline{\text{BINS}}$	SD4	$\overline{\text{BINS}}$	SD4
19		$\overline{\text{RINT}}$	SD3	$\overline{\text{RINT}}$	SD3
18		$\overline{\text{ALTO}}$	SD2	$\overline{\text{ALTO}}$	SD2
17		$\overline{\text{ESM}}$	$\overline{\text{ESM}}$	$\overline{\text{ESM}}$	SD1
16		RSCK	RSCK	RSCK	SD0
	1/0	0	1	0	1
	$\overline{\text{CBUS}}$	0	0	1	1

TABLA IV.1.A.3.- Líneas controladas por los bits del subcampo de Modificación y valores para 1/0 y $\overline{\text{CBUS}}$ asociados.

A partir de esta tabla se puede construir el decodificador mostrado en la FIGURA IV.1.A.4 el cual tiene como entradas los bits CCC_{16} a CCC_{25} del campo de Control del Controlador y como salidas las líneas que se indican en la TABLA IV.1.A.3. En este diagrama también se muestran los bits CCC_{26} -

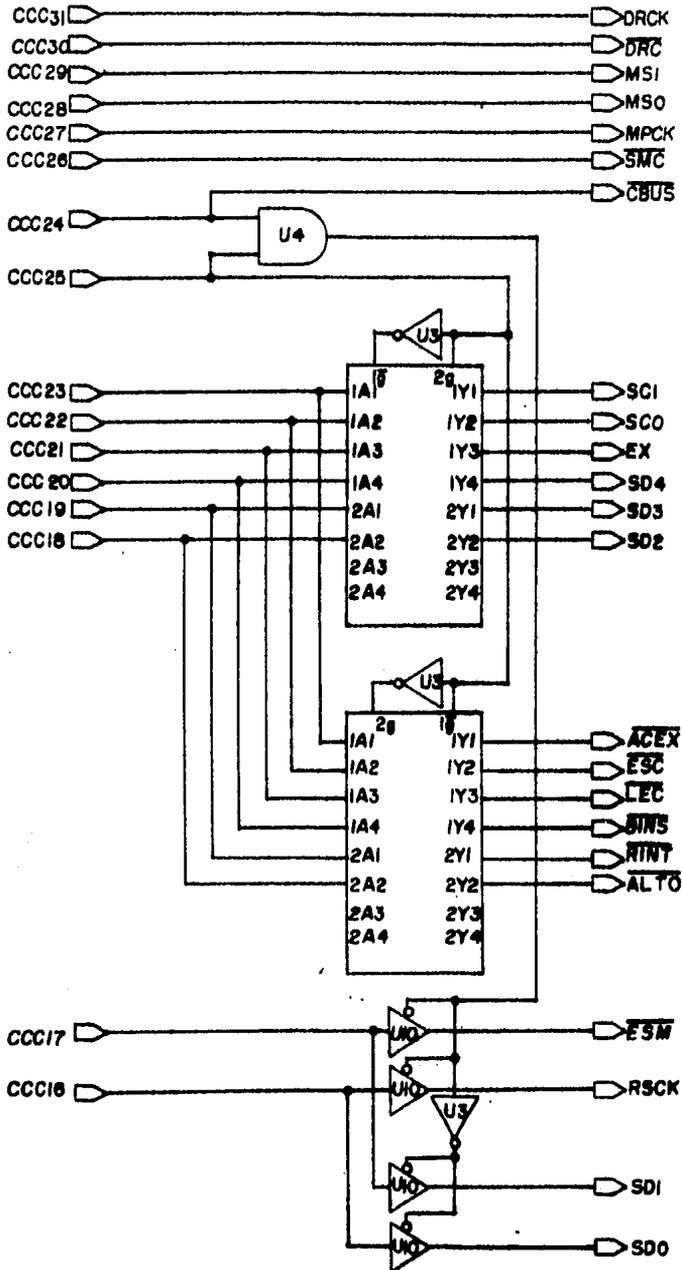


FIGURA IV.1.A.4.- Diagrama Lógico del Decodificador del Campo de Control del Controlador.

a CCG_{31} (que no entran al decodificador) con la indicación de las líneas a las que van a controlar. Hasta ahora solo se conoce el nombre de las líneas que han de ser controladas -- por el Campo de Control del Controlador y no se ha dado ninguna información acerca de la función que realizan o del bloque a que pertenecen (con excepción de las pertenecientes al Bus de Control); esto por el momento no ofrece mayor problema al desarrollo de esta descripción y será tratado a partir de la siguiente sección donde se comienza a hacer la descripción de los demás bloques que integran a la Sección de Control; en esta descripción se incluirá la tabla de verdad del bloque y las líneas de control que intervienen en su funcionamiento.

Con las consideraciones hechas hechas anteriormente se ha definido la forma en que quedarán ordenados los 32 bits de la PROM de Control y las funciones que estos realizarán. La forma en que serán realizadas las microoperaciones y las microoperaciones que serán realizadas en el procesador serán descritas posteriormente.

En la FIGURA IV.1.A.5 se muestran los 32 bits de la Prom de Control incluyendo los campos y subcampos en que fueron a grupados.

La forma en que será construida la Prom de Control se --- muestra en el diagrama lógico de la FIGURA IV.1.A.6 donde:

- C_0 a C_{10} .- Líneas correspondientes a las direcciones de la Prom de Control. Estas líneas de dirección provienen del Registro SM. Para referirse a ellas a lo largo de este texto se utilizará el término Micro-direcciones de Control.
- CCP_0 a CCP_{15} .- Líneas de salida de la Prom de Control

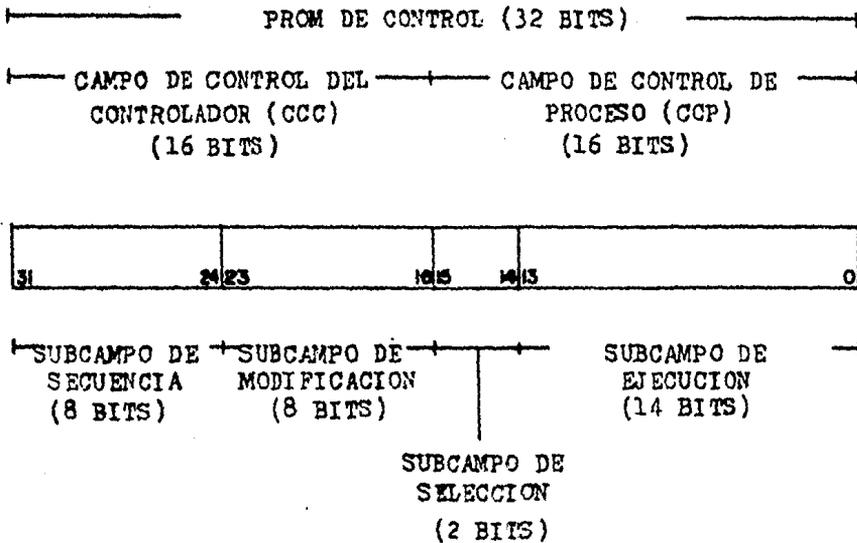


FIGURA IV.1.A.5.- Los 32 bits de la Prom de Control.

correspondientes a los 16 bits del Campo de Control de Proceso de la Prom de Control. Estos son los 16 bits menos significativos de la Prom de Control.

CCC_{16} a CCC_{31} .- Líneas de salida de la Prom de Control correspondientes a los 16 bits del Campo de Control del Controlador de la Prom de Control. Estos son los 16 bits más significativos de la Prom de Control.

NOTA. Los 16 bits menos significativos de la Prom de Control (CCP_0 a CCP_{15}) son entregados por la Prom de Control al decodificador del Campo de Control de Proceso y los 16 más significativos

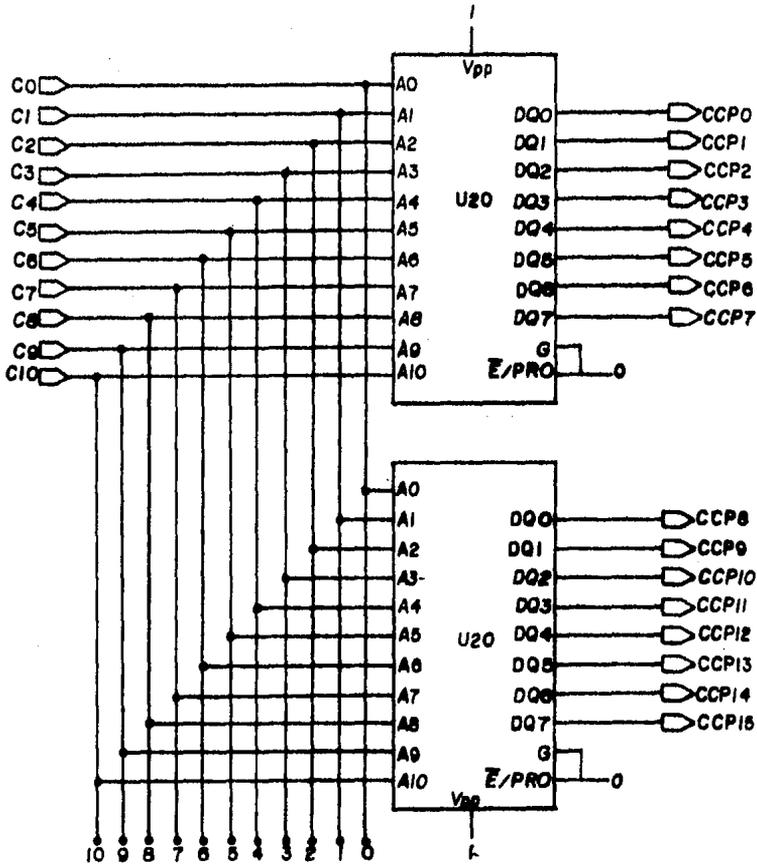


FIGURA IV.1.A.6.- Diagrama Lógico de la Prom de Control.

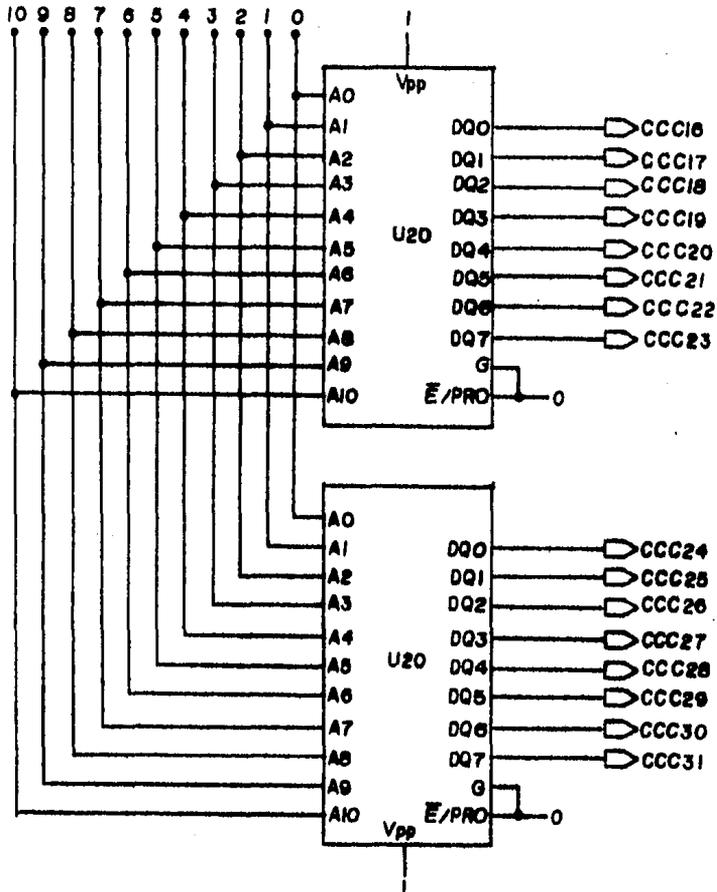


FIGURA IV.1.A.6.- (Continúa)

(CCC_{16} a CCC_{31}) al decodificador del Cam po de Control del Controlador.

IV.1.B.- EL REGISTRO SM (Secuenciador de Microoperación). Este registro almacena el código de la microoperación indica da por la Prom de Correspondencia (CORR) y lo transforma en las direcciones de la de la Prom de Control a través de las - cuales será realizada una microoperación. Funcionalmente es- tá construido a partir de tres contadores 74LS193 que pueden ser cargados o incrementados. El diagrama lógico y la tabla de verdad de este registro pueden ser observados en la FIGU- RA IV.1.B.1 y la TABLA IV.1.B.1. respectivamente.

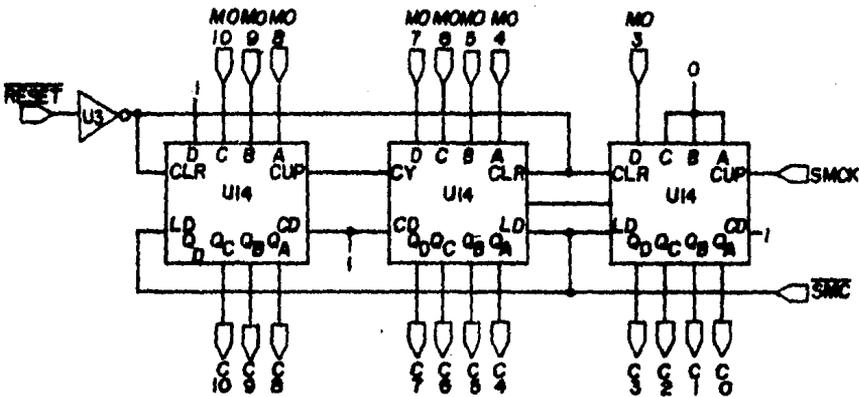


FIGURA IV.1.B.1.- Diagrama Lógico del Registro SM

ENTRADAS (MO)	$\overline{\text{RESET}}$	$\overline{\text{SMC}}$	SMCK	SALIDAS (C)
$MO_{10}, MO_9 \dots MO_4, MO_3$	1	0	1	$MO_{10}, MO_9 \dots MO_4, MO_3, 000$
X	1	1	↑	$C_Q + 1$
X	0	X	X	0

TABLA IV.1.B.1.- Tabla de Verdad del Registro SM.

Donde:

MO_{10}, \dots, MO_3 -- Líneas de entrada al Registro SM provenien- tes de la Prom de Correspondencia.

C_{10}, \dots, C_0 -- Líneas de salida del Registro SM. estas lí-

neas corresponden a las microdirecciones de la Prom de Control

SMCK .- Línea correspondiente a la señal de reloj del Registro SM. Proviene de la Lógica de Espera.

SMC

.- Línea de carga del Registro SM.

RESET

.- Línea de inicialización del procesador perteneciente al Bus de Control. Pone en ceros al Registro SM.

IV.1.C.- LA PROM DE CORRESPONDENCIA (CORR). Será construida con una memoria EPROM MCM 2716; su funcionamiento será parecido al que realiza un decodificador; por un lado, esta memoria recibirá una microdirección proveniente del Registro MPC y la transformará en un código correspondiente a una microoperación realizable con la ayuda de la Prom de Control. El contenido de la Prom de Correspondencia estará compuesto por series de códigos de microoperaciones ordenados de tal forma que al realizarse una serie de esas microoperaciones el procesador haya realizado una operación mas compleja; por ejemplo una instrucción o un reconocimiento de interrupción. Cada una de las series de códigos de microoperación grabados en la Prom de Correspondencia será llamado microrrutina. Generalmente, cada microrrutina tendrá el propósito de realizar una instrucción, aunque para esto se sirva de llamar a alguna otra microrrutina auxiliar; tambien, existen algunas otras microrrutinas que no realizan instrucción alguna, sino que sirven para dar servicio a las operaciones que realiza el procesador, como por ejemplo un reconocimiento de interrupción. A manera de conclusión puede decirse que cada microrrutina está compuesta por una serie de microoperaciones que ordenadas secuencial y coherentemente tienen como fi

nalidad realizar alguna instrucción u operación del procesador. Las microrrutinas contenidas en la Prom de Correspondencia serán descritas posteriormente. El diagrama lógico de la Prom de Correspondencia se da en la FIGURA IV.1.C.1.

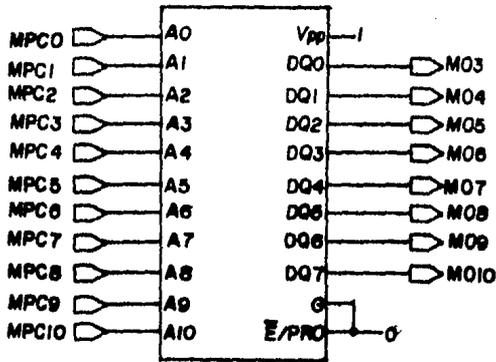


FIGURA IV.1.C.1.- Diagrama Lógico de la Prom de Correspondencia.

Donde:

$MPC_{10}, MPC_9, \dots, MPC_0$.- Líneas de dirección de la Prom de Correspondencia provenientes del Registro MPC, en este texto serán llamadas Microdirecciones de Correspondencia.

$MO_{10}, MO_9, \dots, MO_3$.- Líneas de salida de la Prom de Correspondencia; corresponden al código de la microoperación almacenado en la Microdirección de Correspondencia dada.

IV.1.D.- EL REGISTRO MPC (Contador de Microprograma). Este registro será construido a partir de tres contadores binarios 74LS193; deberá tener la capacidad de incrementarse y de ser cargado con la información proveniente del Multiplexor (MUX). La información proveniente del Multiplexor será interpretada ya sea como dirección inicial de una microrrutina

de las que se han de programar en la Prom de Correspondencia o bien como una dirección de salto o regreso de una microrrutina. El diagrama lógico del Registro MPC y su tabla de verdad se dan en la FIGURA IV.1.D.1. y en la TABLA IV.1.D.1 respectivamente.

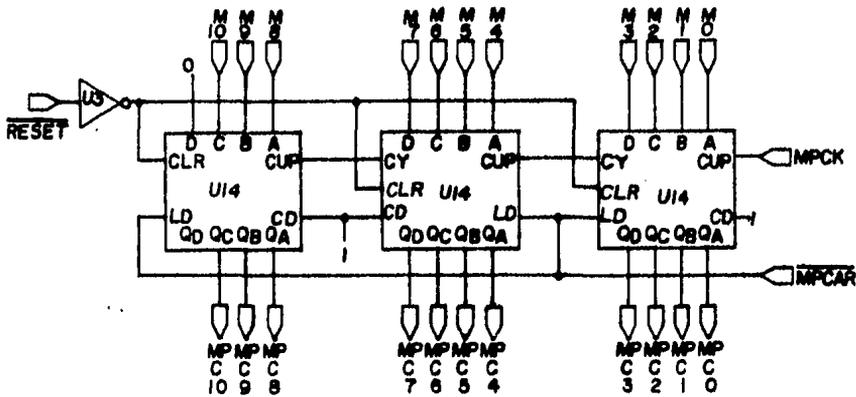


FIGURA IV.1.D.1.- Diagrama Lógico del Registro MPC.

ENTRADAS (M)	MPCK	$\overline{\text{MPCAR}}$	SALIDAS (MPC)
$M_{10}, M_9 \dots M_1, M_0$	1	0	$M_{10}, M_9 \dots M_1, M_0$
X	1	1	MPC_Q
X	↑	1	$\text{MPC}_Q + 1$

TABLA IV.1.D.1.- Tabla de verdad del Registro MPC.

Donde:

- $M_{10}, M_9 \dots M_1, M_0$.- Líneas de entrada al Registro MPC provenientes del Multiplexor.
- $\text{MPC}_{10}, \text{MPC}_9, \dots, \text{MPC}_0$.- Líneas de salida del Registro MPC, estas son las líneas pertenecientes a la Microdirección de Correspondencia.
- MPCK .- Línea correspondiente a la señal de reloj del Registro MPC.
- $\overline{\text{MPCAR}}$.- Línea de carga del Registro MPC; esta

línea proviene de la Lógica de Condición.

RESET

.- Línea de inicialización del procesador perteneciente al Bus de Control.

IV.1.E.- EL MULTIPLEXOR (MUX). El Multiplexor tiene como finalidad entregar al Registro MPC la microdirección de las microrrutinas de correspondencia que se han de realizar; estas microdirecciones pueden venir del Registro RS en cuyo caso las microrrutinas que se ejecutarán serán las de búsqueda de instrucción, concesión del bus o reconocimiento de interrupción; pueden venir del Registro RI y en este caso las microrrutinas que se han de ejecutar corresponderán a las que se encargan de realizar alguna de las instrucciones del procesador; pueden venir del Decodificador CCC y en este caso ocurrirá un salto a otra microrrutina y, finalmente, pueden venir del Registro DR en cuyo caso se tratará de una microdirección de regreso de una microrrutina. El diagrama lógico y la tabla de verdad del multiplexor se pueden observar en la FIGURA IV.1.E.1 y la TABLA IV.1.E.1 respectivamente.

MS ₁	MS ₀	SALIDAS										
		M ₁₀	M ₉	M ₈	M ₇	M ₆	M ₅	M ₄	M ₃	M ₂	M ₁	M ₀
0	0	DR ₁₀	DR ₉	DR ₈	DR ₇	DR ₆	DR ₅	DR ₄	DR ₃	DR ₂	DR ₁	DR ₀
0	1	0	0	0	0	0	0	RS ₁	RS ₀	0	0	0
1	0	1	1	0	SD ₄	SD ₃	SD ₂	SD ₁	SD ₀	0	0	0
1	1	RI ₇	RI ₆	RI ₅	RI ₄	RI ₃	RI ₂	RI ₁	RI ₀	0	0	0

TABLA IV.1.E.1.- Tabla de verdad del Multiplexor.

Donde:

M₁₀, M₉...M₁, M₀ .- Líneas de salida del Multiplexor entregadas al Registro MPC.

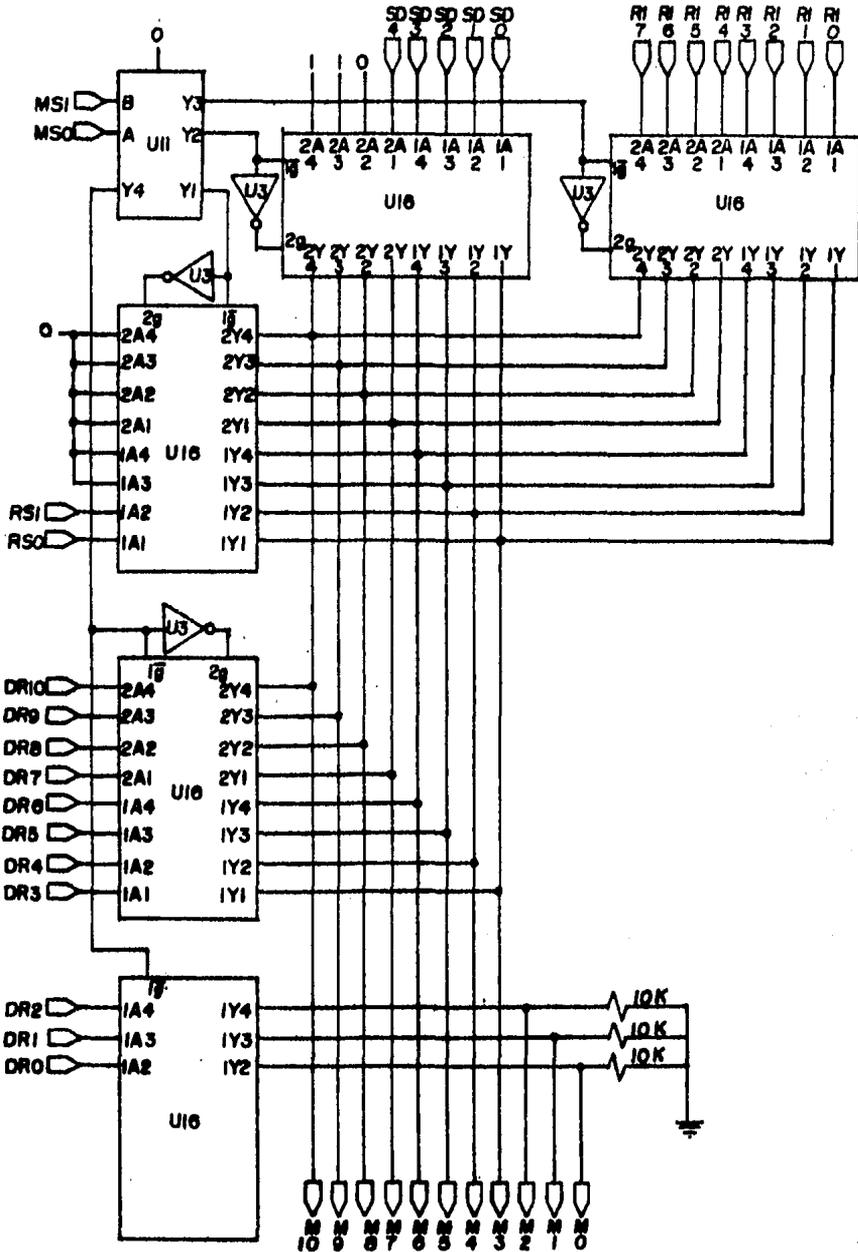


FIGURA IV.1.E.1.- Diagrama Lógico del Multiplexor.

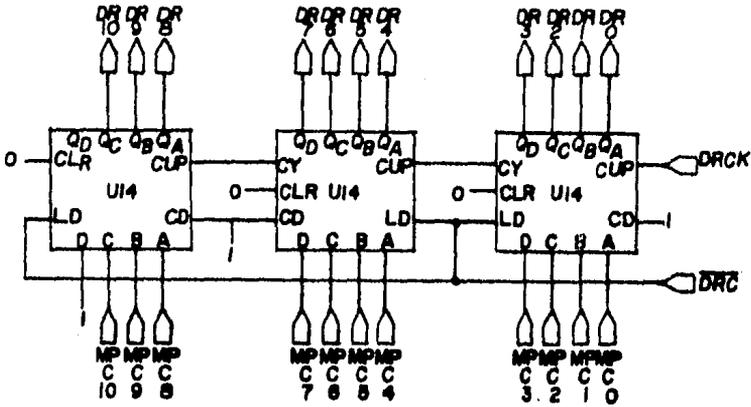


FIGURA IV.1.F.1.- Diagrama Lógico del Registro DR.

nientes del Registro MPC.

$DR_{10}, DR_9 \dots DR_0$.- Líneas de salida del Registro DR entregadas al Multiplexor.

\overline{DRG} .- Línea de carga del Registro DR.

DRCK .- Línea correspondiente a la señal de reloj del Registro DR.

IV.1.G.- EL REGISTRO RI. En el Registro de Instrucción se rá almacenado el código de operación de la instrucción que será realizada en el procesador; este código de operación se rá cargado a partir del bus interno del procesador. Para realizar la instrucción; el contenido del Registro RI será transmitido al Registro MPC através del Multiplexor. A continuación se dan la tabla de verdad y el diagrama lógico del Registro RI, FIGURA IV.1.G.1 y TABLA IV.1.G.1 respectivamente.

Donde;

$B_7, B_6 \dots B_0$.- Líneas de entrada al Registro RI provenientes del bus interno del procesador.

- $RI_7, RI_6 \dots RI_0$.- Líneas de salida del Registro RI entregadas al Multiplexor.
- RICK .- Línea correspondiente a la señal de reloj del Registro RI.

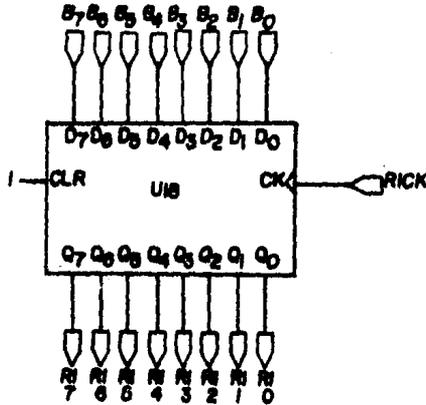


FIGURA IV.1.G.1.- Diagrama Lógico del Registro RI.

ENTRADAS (B)	RICK	SALIDAS (RI)
B ₇ , B ₆ ... B ₁ , B ₀	1	B ₇ , B ₆ ... B ₁ , B ₀
X	1	RI ₇ , RI ₆ ... RI ₀

TABLA IV.1.G.1.- Tabla de verdad del Registro RI.

IV.1.H.- EL REGISTRO RS. El Registro de Rutinas de Servicio tiene el propósito de censar el estado de las líneas de entrada del Bus de Control INT y RBUS y registrar si estas han sido activadas o no. Al final de cada una de las instrucciones realizadas por el procesador, el Registro RS indicará si debe darse servicio a una interrupción, a un requerimiento de bus o simplemente a la siguiente instrucción en turno. El diagrama lógico del Registro RS y su tabla de verdad se pueden observar en la FIGURA IV.1.H.1 y en la TABLA -

IV.1.H.1 respectivamente.

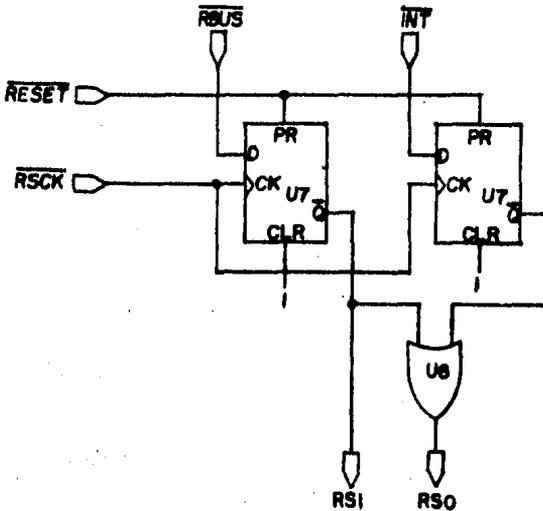


FIGURA IV.1.H.1.- Diagrama Lógico del Registro RS.

ENTRADAS		RESET	RSCK	SALIDAS	
RBUS	INT			RS ₁	RS ₀
X	X	0	X	0	0
1	0	1	1	0	1
0	1	1	1	1	1
0	0	1	1	1	1
1	1	1	1	0	0

TABLA IV.1.H.1.- Tabla de verdad del Registro RS.

Donde:

$\overline{\text{RBUS}}$ e $\overline{\text{INT}}$

.- Líneas de entrada al Registro RS pertenecientes al Bus de Control; $\overline{\text{RBUS}}$ servirá para indicar al procesador que se desea realizar un requerimiento de bus e $\overline{\text{INT}}$ servirá para indicar al procesador que se desea hacer una interrupción.

- RESET .- Línea de inicialización perteneciente al Bus de Control del procesador. Pone en ceros al Registro RS.
- RSCK .- Línea correspondiente a la señal de reloj del Registro RS.
- RS₁ y RS₀ .- Líneas de salida del Registro RS entregadas al Multiplexor.

IV.1.1.- LA LOGICA DE ESPERA. Esta lógica es una trampa que atrapa a la señal de reloj del procesador en el momento en que las líneas ESM y ESP coinciden en un cero lógico. El diagrama lógico y la tabla de verdad de esta lógica pueden ser observados en la FIGURA IV.1.1.1 y la TABLA IV.1.1.1 respectivamente.

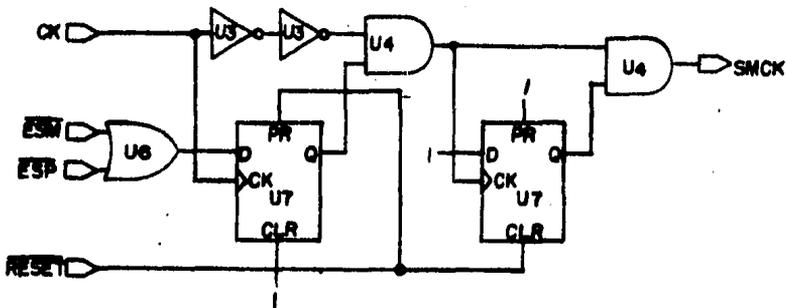


FIGURA IV.1.1.1.- Diagrama Lógico de la Lógica de Espera.

Donde:

- ESP .- Línea de espera perteneciente al Bus de Control.
- RESET .- Línea de inicialización del procesador - perteneciente al Bus de Control.
- ESM .- Línea que indica cuando sea censada la -

\overline{ESP}	\overline{RESET}	$\overline{E3M}$	CK	SMCK
X	0	X	X	0
1	1	1	1	1
1	1	0	1	1
0	1	1	1	1
0	1	0	1	0

TABLA IV.1.I.1.- Tabla de verdad de la Lógica de Espera.

\overline{ESP} . Esta línea proviene del Decodificador CCC.

CK .- Línea correspondiente a la señal de reloj general del procesador; esta es una de las líneas de alimentación del procesador.

SMCK .- Línea mediante la cual la Lógica de Espera entrega al Registro SM la señal de reloj.

IV.1.J.- LA LOGICA DE CONDICION. La Lógica de Condición será la encargada de indicar cuando el Registro MPC debe ser cargado; lo que podrá hacerse de acuerdo a las condiciones existentes en la banderas de acarreo y cero y tambien en forma incondicional. El diagrama lógico y la tabla de verdad de esta lógica se dan a continuación en la FIGURA IV.1.J.1 y la TABLA IV.1.J.1 respectivamente.

Donde:

SC_1 y SC_0 .- Líneas de selección de condición de la Lógica de Condición.

a y z .- Líneas que indican el estado de las banderas de acarreo y cero respectivamente; estas líneas provienen del Registro F.

EX .- Línea de existencia (o no existencia) de condición; cuando esta línea se pone en

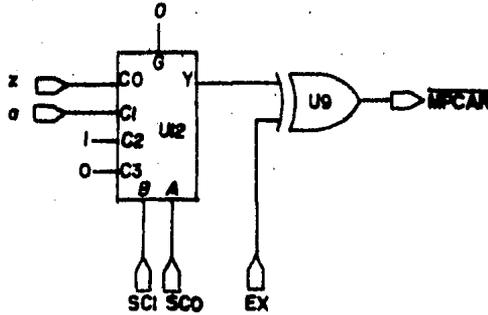


FIGURA IV.1.J.1.- Diagrama Lógico de la Lógica de Condición.

SC ₁	SC ₀	z	a	EX	MPCAR
0	0	1	X	0	1
0	0	1	X	1	0
0	0	0	X	0	0
0	0	0	X	1	1
0	1	X	1	0	1
0	1	X	1	1	0
0	1	X	0	0	0
0	1	X	0	1	1
1	0	X	X	1	0
1	1	X	X	1	1

TABLA IV.1.J.1.- Tabla de verdad de la Lógica de Condición lógico, la condición para cargar al Registro MPC será la existencia de bandera; si se pone en cero, la condición será la no existencia de bandera.

MPCAR .- Línea que la Lógica de Condición entrega al Registro MPC para indicar que este sea o no car

gado.

IV.2.- DESCRIPCIÓN DE LAS MICROOPERACIONES. En la sección IV.1 se determinó que la Prom de Control contaría con 32 --- bits divididos en dos campos principales de 16 bits cada uno; el de Control de Proceso y el de Control del Controlador, de donde se desprende que deberán existir dos tipos o clases de microoperaciones; uno por cada campo de control, las cuales, si resulta necesario, podrán realizarse simultáneamente. A continuación se describirán, en primer lugar las microoperaciones pertenecientes al campo de Control de Proceso.

Por comodidad, en esta sección se empleará el signo ":= " en vez de "←" para denotar transferencia.

IV.2.A.- MICROOPERACIONES DE CONTROL DE PROCESO. Estas se ran agrupadas en cuatro grupos de acuerdo a lo discutido en la sección IV.1.A y con referencia a la TABLA IV.1.A.1.

PRIMER GRUPO. Las microoperaciones del primer grupo po--- drán manipular las líneas de control \overline{ACK} , \overline{ALUE} , \overline{CBJE} , \overline{PCAR} , \overline{PCHE} , \overline{PCLE} , \overline{LB} , \overline{TCK} , \overline{HE} , \overline{LE} , \overline{HCK} y \overline{LCK} que son las suficientes para realizar transferencias entre los Registros PC, HL, T y A del procesador. Por lo tanto las microoperaciones realizables en este grupo, descritas en lenguaje de asignación, son las siguientes:

PC:=HL

T:=PCH

T:=PCL

L:=A

H:=A

las cuales serán realizadas en la forma descrita por el diagrama de tiempo de la FIGURA IV.2.A.1.

En donde es posible observar que estas microoperaciones -

serán realizadas en cinco ciclos de reloj;

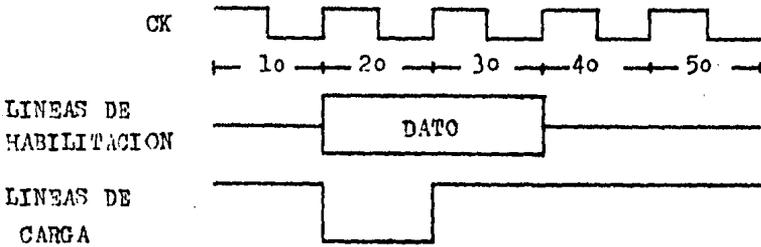


FIGURA IV.2.A.1.- Diagrama de tiempo de las microoperaciones del primer grupo de Control de Proceso.

- 10.- Durante el primer ciclo de reloj no es activada ninguna de las líneas de control y es el ciclo en el que el subcampo de selección elige al grupo 1 de control por medio del Decodificador CCP; para esto, el subcampo de Selección pone sus bits en 00.
- 20.- Durante el segundo ciclo de reloj son activadas las líneas que conducen al dato desde su destino inicial al final através del bus interno del procesador (\overline{ALU} , \overline{E} , $\overline{CB3E}$, \overline{PCHE} , \overline{PCLE} , \overline{LB} , \overline{HE} , \overline{LE}) y las líneas de carga o la línea de carga apropiada (\overline{ACK} , \overline{PCAR} , \overline{TCK} , \overline{HCK} , \overline{LCK}) es puesta en un nivel de cero lógico.
- 30.- Durante el tercer ciclo de reloj las líneas de carga son regresadas a un nivel de uno lógico provocando con esto que el dato sea atrapado en el registro indicado. Mientras tanto el dato continúa estable en el bus interno del procesador.
- 40.- Durante el cuarto ciclo de reloj son desactivadas todas las líneas de control.
- 50.- El procesador se prepara a realizar la siguiente microoperación.

Al describir una microoperación realizada en la forma an-

terior no se ha hecho mención del funcionamiento que ha tenido el controlador mientras tanto, pero es lógico suponer que este debió haber estado buscando la siguiente microoperación en turno al mismo tiempo que ordenaba que la microoperación descrita con anterioridad fuera realizada. El mismo comentario es aplicable a los otros tres grupos de microoperaciones de Control de Proceso.

SEGUNDO GRUPO. Las microoperaciones del segundo grupo podrán manipular las líneas de control \overline{ACK} , \overline{ALUE} , \overline{PCK} , \overline{PCAR} , \overline{PCHE} , \overline{PCLE} , \overline{LB} , \overline{SCK} , L/\overline{E} , \overline{S} , \overline{BE} y \overline{FE} . Con estas líneas será posible realizar transferencias entre el Registro S y los Registros A, F y PC del procesador. Las microoperaciones realizables en este grupo, descritas en lenguaje de asignación -- son las siguientes:

$S:=A$, $S:=F$, $S:=PC$, $A:=S$, $F:=S$ y $PC:=S$

Las tres primeras que transfieren del registro hacia el - stack serán realizadas en la forma descrita por el diagrama de tiempo de la FIGURA IV.2.A.2. a continuación.

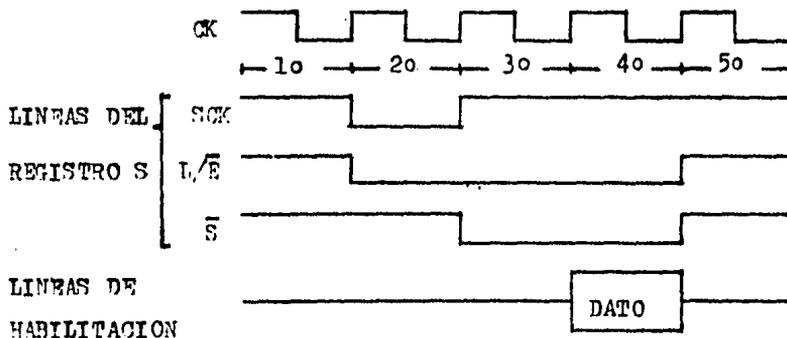


FIGURA IV.2.A.2.- Diagrama de tiempo de las microoperaciones que cargan al Registro S (Grupo 2).

En esta figura se puede observar que estas microoperacio-

nes serán realizadas en cinco ciclos de reloj;

- 1o.- Durante el primer ciclo de reloj el subcampo de Selección adquiere el valor 01 y no es activada ninguna línea de control.
- 2o y 3o.- Son activadas las señales de control del Registro S (\overline{SCK} , L/\overline{E} , \overline{S}) de tal forma que este esté listo para recibir un dato de entrada.
- 4o.- Durante el cuarto ciclo de reloj son activadas las líneas que conducen al dato desde su destino inicial al final a través del bus interno del procesador (\overline{PCH} , \overline{E} , \overline{PCLE} , \overline{BL} , \overline{ALUE} , \overline{FE}), este dato será cargado en el Registro S.
- 5o.- Son desactivadas todas las líneas de control el controlador se prepara a realizar otra microoperación.

La forma en que el Registro S transferirá su contenido a los registros mencionados se describe en el diagrama de tiempo de la FIGURA IV.2.A.3 a continuación.

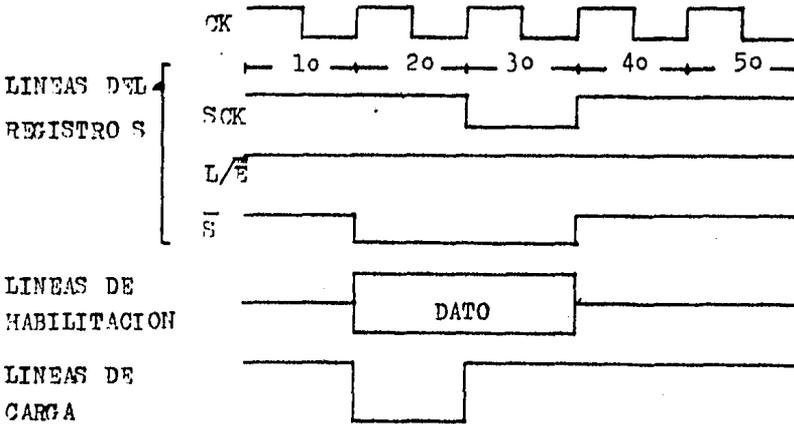


FIGURA IV.2.A.3.- Diagrama de tiempo de las microoperaciones que descargan al Registro S (Grupo 2)

Estas microoperaciones serán realizadas en cinco ciclos - de reloj:

- 10.- Durante este ciclo de reloj el subcampo de Selección adquiere el valor 01 y no es activada ninguna línea de control.
- 20.- Son activadas las líneas de control del Registro S - de tal forma que este muestre el dato con el cual es tá cargado; al mismo tiempo son activadas las líneas de habilitación (LB) que se encargan de llevar el da to a su destino final através del bus interno del -- procesador y se ponen en cero lógico las líneas de - carga del registro que ha de ser cargado (ACK, PCK, PCAR).
- 30.- Son regresadas a un uno lógico las líneas de carga, lo que provoca que el registro que las posee quede - cargado con el dato proveniente del Registro S que a ún permanece estable.
- 40.- Son desactivadas todas las líneas de control.
- 50.- El procesador se prepara a realizar otra microoperación.

TERCER GRUPO. Para realizar las microoperaciones de este grupo podrán ser activadas las líneas de control ACK, ALUE, PCK, ENT, SAL, PCK, PS3, TCK, CBLE, AS3/HCK, AS2/LCK, AS1, - AS0 y RICK. Activando estas líneas las microoperaciones que podrán realizarse serán las de transferencia entre la memo-- ria externa y los Registros A, T, RI y HL del procesador. -- Tambien será posible realizar las operaciones en el Acumula-- dor necesarias para completar las rotaciones, corrimientos y desplazamientos descritos en la sección II.4: Las microopera-- ciones de transferencia entre los registros del procesador y

La memoria son las siguientes:

A:=(MAPC), T:=(MAPC), RI:=(MAPC), L:=(MAPC),

H:=(MAPC), (MAPC):=A y L:=INT

donde se ha usado (MAPC) para denotar al contenido de la localidad de Memoria Apuntada por el Registro PC e INT para denotar a la identificación de 8 bits que debe presentar el dispositivo que realice alguna interrupción.

Las microoperaciones que serán realizadas en el Acumulador son las siguientes:

A:=ACOD .- Realiza el corrimiento descrito en la instrucción CDER.

A:=ACAOB .- Realiza la rotación descrita en la instrucción RDER.

A:=ACA7I .- Realiza la rotación descrita en la instrucción RIZR.

A:=ACAD .- Realiza la rotación descrita en la instrucción RDEAR.

A:=ACAI .- Realiza la rotación descrita en la instrucción RIZAR.

A:=ACOI .- Realiza el corrimiento descrito en la instrucción CIZR.

A:=ACA7D .- Realiza el desplazamiento descrito en la instrucción DDER.

A:=ACAOI .- Realiza el desplazamiento descrito en la instrucción DIZR.

El diagrama de tiempo de la FIGURA IV.2.A.4 muestra la forma en que serán realizadas las instrucciones que transfieren un dato desde la memoria externa hacia alguno de los Registros ya sea A, T, RI, L o H.

Las microoperaciones que ejecuten esta transferencia serán realizadas en 5 ciclos de reloj que son los siguientes:

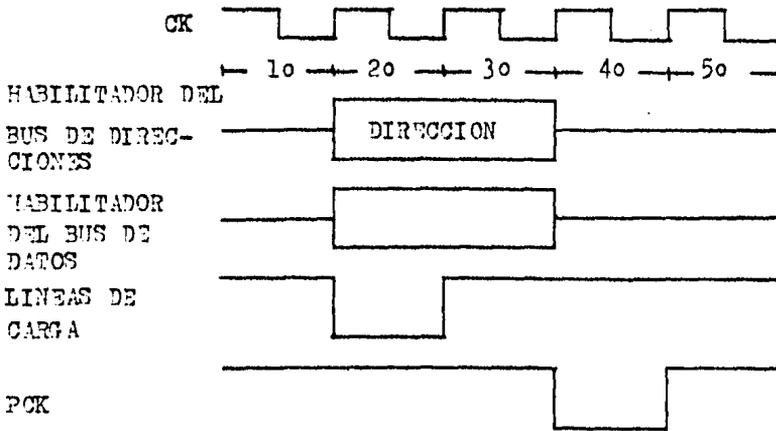


FIGURA IV.2.A.4.- Diagrama de tiempo de las microoperaciones que transfieren un dato desde la memoria externa hacia alguno de los registros del procesador.

- 10.- Durante el primer ciclo de reloj no es activada ninguna línea de control y el subcampo de Selección adquiere el valor 10.
- 20.- En el segundo ciclo de reloj se habilita el bus de direcciones (\overline{CBLE}) y es mostrada la dirección a partir de la cual el dato debe ser transferido; mientras tanto, el bus de datos es puesto en sentido de entrada (\overline{WNT}) y la línea de carga del registro apropiado es puesta en un cero lógico (ACK, TCK, HCK, LCK, RICK).
- 30.- Durante el tercer ciclo de reloj el estado de los Buses de Direcciones y Datos permanece igual que antes y la línea de carga es regresada a un uno lógico, provocando con esto que el dato que hubiere sido puesto en el Bus de Datos sea cargado en el registro elegido.

- 40.- Durante el cuarto ciclo de reloj son deshabilitados los Buses de Direcciones y Datos y es puesta en cero la señal que incrementa el Registro PC (PCK).
- 50.- Durante el quinto ciclo de reloj, son desactivadas todas las líneas de control y es regresada a un nivel de uno lógico la línea PCK provocando con esto que el contenido del Registro PC sea incrementado. - Mientras tanto el procesador se prepara a realizar otra microoperación.

En forma simultánea a las microoperaciones realizadas en la manera descrita con anterioridad será necesario que el procesador realice, con la ayuda de su Bus de Control, un Ciclo de Lectura o bien un Ciclo de Búsqueda de Instrucción para el caso particular en que la transferencia sea hecha de la memoria al Registro de Instrucción (RI). Si el Registro RI fuera cargado con una instrucción ALTO esto provocaría que el Registro PC no fuera incrementado y que esta misma instrucción fuera buscada un número indefinido de veces.

La transferencia del Registro A hacia el contenido de la localidad de memoria apuntada por el Registro PC será realizada en la forma que lo describe el diagrama de tiempo de la FIGURA IV.2.A.5 y en forma simultánea deberá ser realizado un Ciclo de Escritura en el Bus de Control del procesador.

Los cinco ciclos de reloj en los que se realiza esta microoperación son los siguientes:

- 10.- Durante el primer ciclo de reloj no es activada ninguna línea de control y el subcampo de Selección adquiere el valor 10
- 20.- En este ciclo de reloj es habilitado el Bus de Direcciones (\overline{CBLE}) y es mostrada la dirección a la cual el dato debe ser transferido; al mismo tiempo, el --

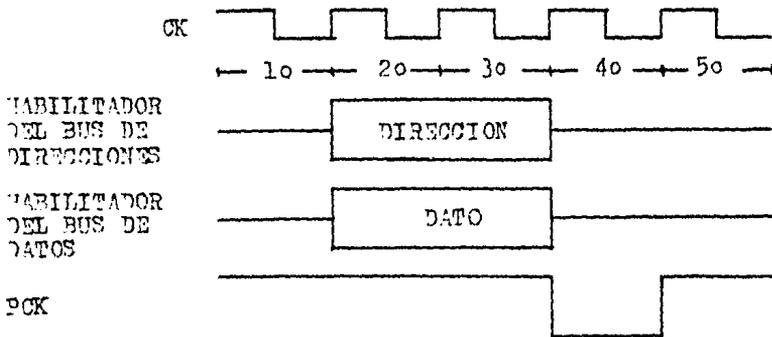


FIGURA IV.2.A.5.- Diagrama de tiempo de la microoperación -- que transfiere un dato desde el contenido del Registro A hacia la memoria externa.

contenido del Registro A es puesto en el bus interno (ALUE) y el Bus de Datos es puesto en sentido de salida (SAL).

- 30.- Durante el tercer ciclo de reloj son conservados los mismos estados que fueron establecidos durante el segundo.
- 40.- Durante el cuarto ciclo son deshabilitados los Buses de Datos y Direcciones y el bus interno del procesador; al mismo tiempo es puesto en un cero lógico la línea que incrementa al Registro PC (PCK).
- 50.- Durante el quinto ciclo de reloj son desactivadas todas las líneas de control y el procesador se prepara a realizar otra microoperación. Al ser regresada la línea PCK a un uno lógico, se provocará que el Regis

tro PC sea incrementado.

La microoperación denominada $L:=INT$ será realizada en la forma que indica el diagrama de tiempo de la FIGURA IV.2.A.6.

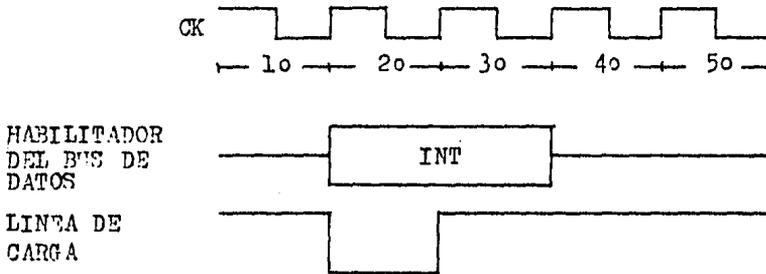


FIGURA IV.2.A.6.- Diagrama de tiempo de la microoperación -- $L:=INT$.

Esta microoperación será realizada en cinco ciclos de reloj:

- 10.- Durante el primer ciclo de reloj no es activada ninguna línea de control y el subcampo de Selección adquiere el valor 10.
- 20.- Durante este ciclo de reloj el Bus de Datos es habilitado en sentido de entrada (\overline{ENT}) y la línea de carga del Registro L (LCK) es puesta en un cero lógico.
- 30.- Durante el tercer ciclo de reloj el Bus de Datos permanece en el mismo estado adquirido un ciclo de reloj antes y es regresada a un uno lógico la línea de carga del Registro L provocando con esto que la identificación INT puesta en el Bus de Datos sea cargada en el Registro L. Debe recordarse que la identificación INT es presentada al Bus de Datos por el dispositivo que realiza una interrupción.

4o.- Durante este ciclo de reloj son desactivadas todas las líneas de control

5o.- Permanecen inactivas las líneas de control y el procesador se prepara a realizar otra microoperación.

Paralelamente al desarrollo de esta microoperación debe ser realizado un ciclo de Reconocimiento de Interrupción.

El diagrama de tiempo de la figura IV.2.A.7 muestra la forma en que serán completadas las microoperaciones realizadas en el Acumulador.

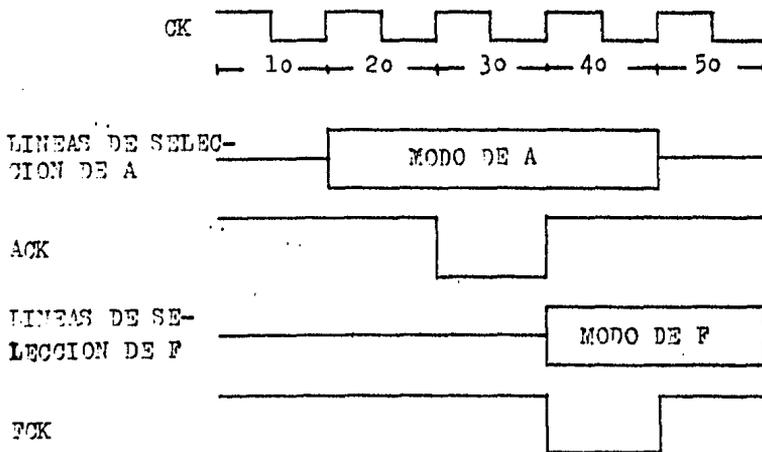


FIGURA IV.2.A.7.- Diagrama de tiempo de las microoperaciones realizadas en el acumulador.

Los cinco ciclos de reloj empleados para realizar las microoperaciones del Acumulador son las siguientes:

1o.- Durante el primer ciclo de reloj el subcampo de Selección adquiere el valor 10.y no es activada ninguna línea de control.

2o.- Durante el segundo ciclo de reloj es seleccionado el

modo de operación del Registro A (AS3, AS2, AS1, AS0) mientras la señal de reloj ACK permanece en un uno lógico.

- 30.- ES puesta en cero lógico la señal ACK y la selección del modo de operación del Registro A permanece estable.
- 40.- ACK regresa al nivel de un uno lógico y es realizada la operación indicada por las líneas de selección -- del modo de operación que aún permanecen estables. - Al mismo tiempo, las líneas de selección del modo de control del Registro F (FS3) son activadas y la línea correspondiente a FCK es puesta en cero lógico.
- 50.- Durante el quinto ciclo de reloj las líneas de selección del modo de operación de A son desactivadas al igual que FCK que al regresar a un nivel de uno lógico completa la operación que debe realizar el Registro F; mientras tanto las líneas de selección del modo de F permanecen estables y el procesador se prepara a realizar otra microoperación.

Debido a que AS3 y AS2 son comunes a HCK y LCK respectivamente, el hecho de regresar AS3 (ó AS2) de un nivel de cero a un nivel de uno lógico provocará que en HCK (ó LCK) exista una transición positiva de la señal de reloj y esto ocasionará que el Registro H (ó el L) sea cargado con el dato presente en el bus interno del procesador. Por lo tanto se debe tener cuidado de no cambiar involuntariamente el contenido del Registro HL.

CUARTO GRUPO. Para realizar las microoperaciones del cuarto grupo podrán ser activadas las líneas de control ACK, $\overline{\text{ALU}}$, FCK, M, ALU3, ALU2, ALU1, ALU0, sa, ctrl, FS3, FS2, FS1, y FSO. Activando estas líneas; las microoperaciones que po--

drán realizarse son las operaciones de la ALU y el Registro F. Estas microoperaciones son las siguientes:

A-T, A:=A-T, A:=A-T-a, A:=A+T, A:=A+T, A:=A+T+a
 A:=A+l, A:= \bar{A} , A:=A-l, A:=AorT, A:=AT, A:=OOH,
 A:=FFH, F(a,z):=(0,1), F(a,z):=(\bar{a}, \bar{z}), F(a,z):=(0,0)

donde se ha usado el signo "+y. ("-") para denotar suma (resta) aritmética.

La forma en que serán realizadas las microoperaciones del cuarto grupo se muestra en el diagrama de tiempo de la FIGURA IV.2.A.8. En esta figura es posible observar que las instrucciones del cuarto grupo serán realizadas en cinco ciclos de reloj:

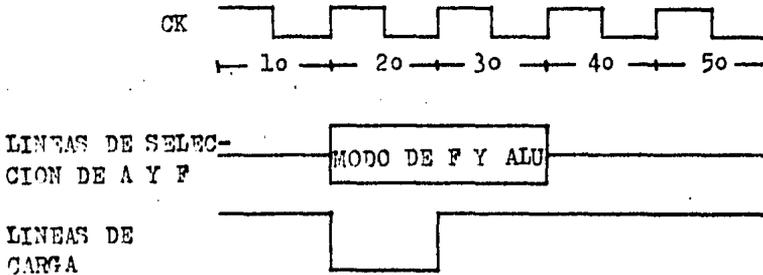


FIGURA IV.2.A.8.- Diagrama de tiempo de las microoperaciones realizadas en la ALU y el Registro F.

10.- Durante el primer ciclo de reloj no es activada ninguna de las líneas de control; es el ciclo en el que el subcampo de Selección pone sus bits en 11 y selecciona al cuarto grupo de Control de Proceso.

20.- Durante el segundo ciclo de reloj son activadas las

líneas de selección de modo y operación de la ALU y del Registro F (ALU3, ALU2, ALU1, ALU0, M, sa; ctrl, ALUE, FS3, FS2, FS1, FS0) y las líneas de carga (ACK, PCK) son puestas en un nivel de cero lógico.

30.- Las líneas de carga son regresadas a un nivel de uno lógico y el Acumulador y el Registro F son cargados con los resultados de la operación realizada en la ALU. Mientras tanto las líneas de selección de modo y operación permanecen en estado estable.

40.- Durante el cuarto ciclo de reloj son desactivadas todas las líneas de control.

50.- Durante el quinto ciclo de reloj el procesador se prepara a realizar la siguiente microoperación.

Paralelamente al desarrollo de estas microoperaciones podrá ser realizada otra microoperación de las realizables bajo el mando del Campo de Control del Controlador, este tipo de microoperaciones se describen a continuación.

IV.2.B.- MICROOPERACIONES DE CONTROL DEL CONTROLADOR. Estas microoperaciones son definidas con referencia a las TABLAS IV.1.A.2 y IV.1.A.3 donde se indican las líneas de control activadas por el subcampo de Secuencia y el subcampo de Modificación respectivamente. Estas microoperaciones son las siguientes:

Microoperación MPC:=MPC+1

El diagrama de tiempo de esta microoperación se muestra a continuación en la FIGURA IV.2.B.1.

Los cinco ciclos de reloj en los que es realizada esta microoperación son los siguientes:

10.- Durante el primer ciclo de reloj no es activada ninguna línea de control y los bits correspondientes a I/O y $\overline{\text{CBUS}}$ adquieren los valores 11.

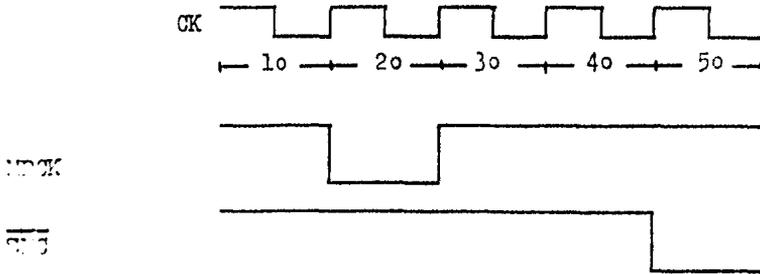


FIGURA IV.2.3.1.- Diagrama de tiempo de la microoperación --
 $MPO = MPO + 1$.

- 20.- Durante el segundo ciclo de reloj es puesta en cero lógico la línea MPOCK.
- 30.- Durante el tercer ciclo de reloj la línea MPOCK es regresada a un uno lógico provocando con esto que el Registro MPO sea incrementado con lo que la Microdirección de Correspondencia también será incrementada.
- 40.- Al ser incrementada la Microdirección de Correspondencia, la Prom CORR contestará buscando el código de la nueva microoperación que se ha de realizar y poniéndolo a la entrada del Registro SM. Para cuando el cuarto ciclo de reloj esté por terminar la Prom CORR ha tenido suficiente tiempo para estabilizar los estados lógicos del código en sus salidas.
- 50.- Durante el quinto ciclo de reloj la línea SMC es puesta en un cero lógico permitiendo con esto que el código entregado por la Prom CORR pase a través del Registro SM y llegue directamente a las líneas de dirección de la Prom de Control. Antes de terminar el quinto ciclo de reloj la Prom de Control ya deberá estar realizando la siguiente microoperación en su primer ciclo de reloj y esto provocará que todas las

líneas de control sean desactivadas; por lo tanto, - al regresar la línea \overline{SMC} a un uno lógico el código de la microoperación entregado por la Prom de Correspondencia (CORR) quedará atrapado en el Registro SM. Microoperaciones:

MPC:=RS, MPC:=RI y MPC:=DR

El diagrama de tiempo de estas microoperaciones se muestra a continuación en la FIGURA IV.2.B.2.

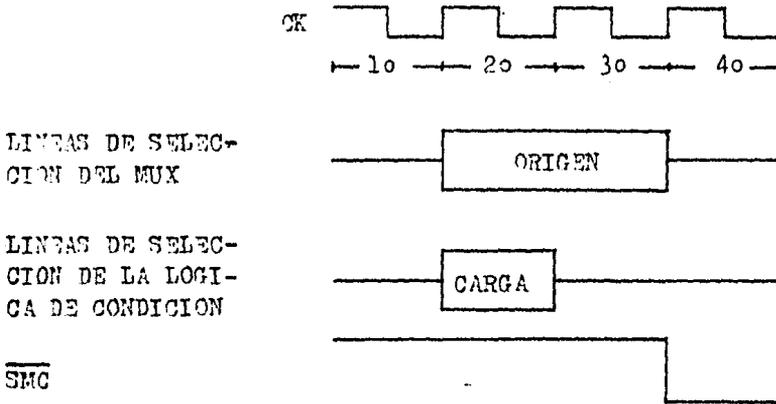


FIGURA IV.2.B.2.- Diagrama de tiempo de las microoperaciones MPC:=RS, MPC:=RI y MPC:=DR.

Los cuatro ciclos de reloj en los que serán realizadas estas microoperaciones son los siguientes:

- 10.- No es activada ninguna línea de control y los bits correspondientes a I/O y \overline{CBUS} son puestos en 11.
- 20.- Por medio de las líneas de selección del Multiplexor (MS1 y MS0) es seleccionado el origen del dato que ha de ser cargado en el Registro MPC; este dato podrá venir de los Registros RS, DR ó bien del Registro RI y será puesto a la entrada del Registro MPC. Al mismo tiempo en las líneas de selección de la Lógica de Condición (SC1, SC0, EX) se indica que el Re

gistro MPC debe realizar una carga incondicional.

30.- Durante el tercer ciclo de reloj las líneas de selección de la Lógica de Condición son desactivadas y el Registro MPC es cargado; mientras tanto el dato entregado por el Multiplexor permanece en estado estable. Al ser cargado el Registro MPC con una nueva Microdirección de Correspondencia, la Prom CORR modifica sus salidas y al finalizar el tercer ciclo de reloj estas ya permanecen estables a la entrada del Registro SM.

40.- Durante el cuarto ciclo de reloj todas las líneas de control son desactivadas y la línea \overline{SMC} es puesta en un cero lógico permitiendo con esto que el código entregado por la Prom CORR pase a través del Registro SM y llegue directamente a las líneas de dirección de la Prom de Control. Antes de terminar el cuarto ciclo de reloj la Prom de Control ya deberá estar realizando la siguiente microoperación en su primer ciclo de reloj y esto provocará que todas las líneas de control sean desactivadas; por lo tanto, al regresar la línea \overline{SMC} a un uno lógico el código de la microoperación entregado por la Prom CORR quedará atrapado en el Registro SM.

Microoperaciones:

MPC:=SD(Microdirección)

Estas microoperaciones realizan la función de una llamada incondicional de microrrutina. Su diagrama de tiempo se muestra en la FIGURA IV.2.B.3.

Los seis ciclos de reloj en que serán realizadas estas microoperaciones son los siguientes:

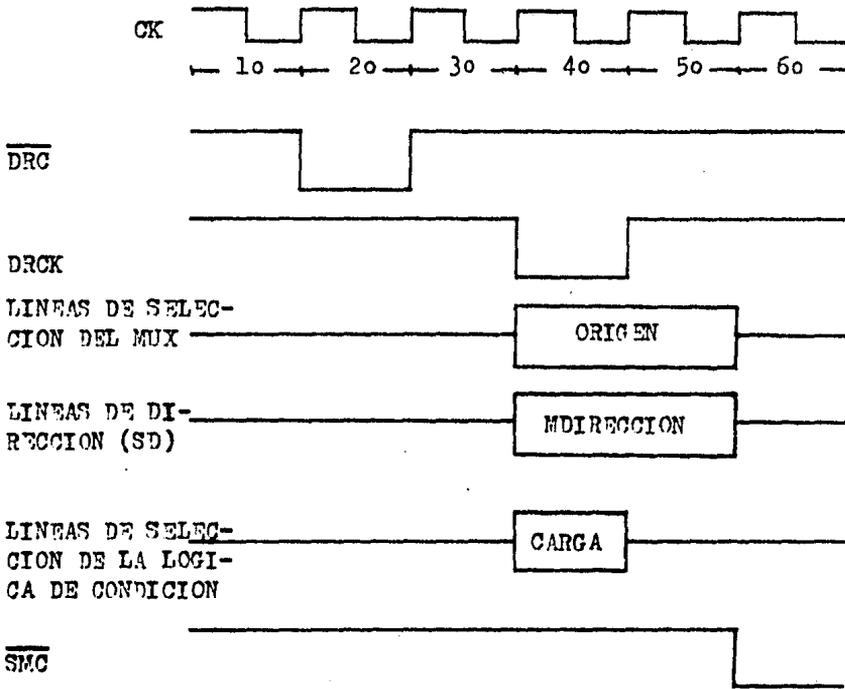


FIGURA IV.2.B.3.- Diagrama de tiempo de las microoperaciones MPC:=SD(MDIRECCION).

- 10.- Durante el primer ciclo de reloj no es activada ninguna línea de control y los bits 1/0 y $\overline{\text{CBUS}}$ adquieren los valores 11.
- 20.- Durante el segundo ciclo de reloj la línea de carga del Registro DR ($\overline{\text{DRC}}$) es puesta en un cero lógico -- permitiendo con esto que el contenido del Registro MPC pase através del Registro DR.
- 30.- Durante el tercer ciclo de reloj la línea de carga del Registro DR es regresada a un uno lógico y con esto el contenido del Registro MPC es atrapado en el

Registro DR.

- 40.- Por medio de las líneas de selección del Multiplexor (MS1, MS0) es seleccionado el origen del dato que ha de ser cargado en el Registro MPC, en esta ocasión - el origen deberá corresponder a las líneas provenientes del Decodificador CCC (SD4, SD3, SD2, SD1, SD0); al mismo tiempo, las líneas que indican la siguiente dirección (SD4, SD3, SD2, SD1, SD0,) son activadas y las líneas de selección de la Lógica de Condición (S Cl, SC0, EX) indican que el Registro MPC debe realizar una carga incondicional; también durante este -- cuarto ciclo de reloj la línea DRCK del Registro DR es puesta en un cero lógico.
- 50.- Durante el quinto ciclo de reloj es regresada a un -- uno lógico la línea DRCK y esto provoca que el Registro DR sea incrementado; con esto queda almacenada -- la dirección de regreso en el Registro DR. Mientras tanto las líneas de selección de la Lógica de Condición son desactivadas y esto ocasiona que el Registro MPC sea cargado con la microdirección que aún -- permanece estable a la salida del Multiplexor y en -- las líneas SD4, SD3, SD2, SD1 y SD0. Al finalizar el quinto ciclo de reloj la Prom de Correspondencia ya ha puesto en sus salidas el nuevo código de microoperación correspondiente a la nueva microdirección al macenada en el Registro MPC.
- 60.- Durante el sexto ciclo de reloj todas las líneas de control son desactivadas y la línea SMC es puesta en en un cero lógico permitiendo con esto que el código entregado por la Prom de Correspondencia pase através del Registro SM y llegue directamente a las líneas -

de dirección de la Prom de Control. Antes de terminar el sexto ciclo de reloj la Prom de Control ya deberá estar realizando la siguiente microoperación en su primer ciclo de reloj y esto provocará que todas las líneas de control sean desactivadas; por lo tanto, al regresar la línea \overline{SMC} a un uno lógico el código de la microoperación entregado por la Prom de Correspondencia quedará atrapado en el Registro SM.

Microoperaciones:

$MPC:=SD,CC(\text{Microdirección})$

Estas microoperaciones realizan la función de un salto -- condicionado; la condición CC puede ser Z (si está puesta la bandera de cero), NZ (si no está puesta), A (si está puesta la bandera de acarreo) y NA (si no está puesta). Su diagrama de tiempo se muestra en la FIGURA IV.2.B.4.

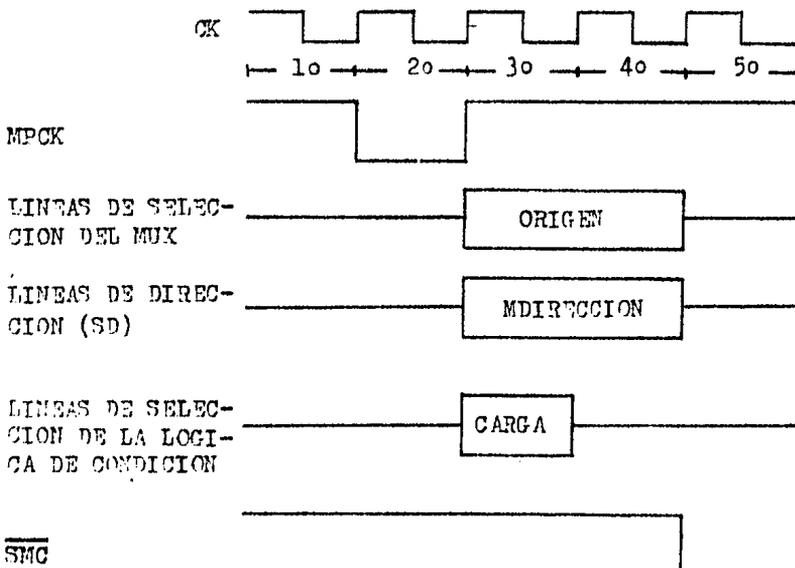


FIGURA IV.2.B.4.- Diagrama de tiempo de las microoperaciones $MPC:=SD,CC(MDIRECCION)$.

Los cinco ciclos de reloj en que serán realizadas estas - microoperaciones son los siguientes:

- 10.- Durante el primer ciclo de reloj no es activada ninguna línea de control y los bits I/O y CBUS adquieren los valores 11.
- 20.- Durante el segundo ciclo de reloj la línea MPCK es puesta en un nivel de cero lógico.
- 30.- Durante el tercer ciclo de reloj las líneas de selección del Multiplexor (MS1, MS0) seleccionan el origen del dato que ha de ser cargado en el Registro MPC; el origen corresponde al Decodificador CCC que al mismo tiempo activa las líneas que muestran la dirección que ha de ser cargada (SD4, SD3, SD2, SD1, SD0) en el Registro MPC. Mientras tanto la línea MPCK es regresada a un uno lógico y las líneas de selección de la Lógica de Condición (SC1, SC0, EX) indican que el Registro MPC sea cargado en forma condicional. En este momento podrán ocurrir dos cosas:

Si no se cumple la condición de carga, el Registro MPC ya habrá sido incrementado y permanecerá en el mismo estado hasta el final de la microoperación.

Si se cumple la condición de carga el dato seleccionado por el Multiplexor pasará a través del Registro MPC desplazando a cualquier otro dato que pudiera haber existido anteriormente.
- 40.- Son desactivadas las líneas de selección de la Lógica de Condición y esto provocará, en caso de que hubiese cumplida la condición, que el Registro MPC atrape al dato entregado por el Multiplexor, mientras tanto, el dato aún permanece estable a la salida del Multiplexor; si la condición no hubiese sido cumpli-

da, el hecho de desactivar las líneas de selección de la Lógica de Condición resultaría intrascendente para el Registro MPC debido a que este ya ha sido incrementado desde el ciclo anterior. Al final del --- cuarto ciclo, la Prom de Correspondencia ya ha puesto en sus salidas el código de la microoperación que se ha de realizar enseguida.

- 50.- Durante el quinto ciclo de reloj son desactivadas todas las líneas de control y la línea $\overline{\text{SMC}}$ es puesta en un cero lógico, permitiendo con esto que el código entregado por la Prom de Correspondencia pase através del Registro SM y llegue directamente a las líneas de dirección de la Prom de Control. Antes de -- terminar el quinto ciclo de reloj la Prom de Control ya deberá estar realizando la siguiente microoperación en su primer ciclo de reloj y esto provocará -- que todas las líneas de control sean desactivadas, -- por lo tanto al regresar la línea $\overline{\text{SMC}}$ a un uno lógico el código de la microoperación entregado por la -- Prom de Correspondencia quedará atrapado en el Registro SM.

Microoperaciones;

LEC, BINS, ESC, RINT y ALTO

Estas microoperaciones tienen la función de realizar los Ciclos de Máquina de Lectura (LEC), Escritura (ESC), Búsqueda de Instrucción (BINS), Reconocimiento de Interrupción (RINT) y Ejecución de una Instrucción ALTO (ALTO). El diagrama de tiempo que indica la forma en que serán realizadas estas microoperaciones es mostrado en la FIGURA IV.2.B.5.

Los cinco ciclos de reloj en que serán realizadas estas -- microoperaciones son los siguientes:

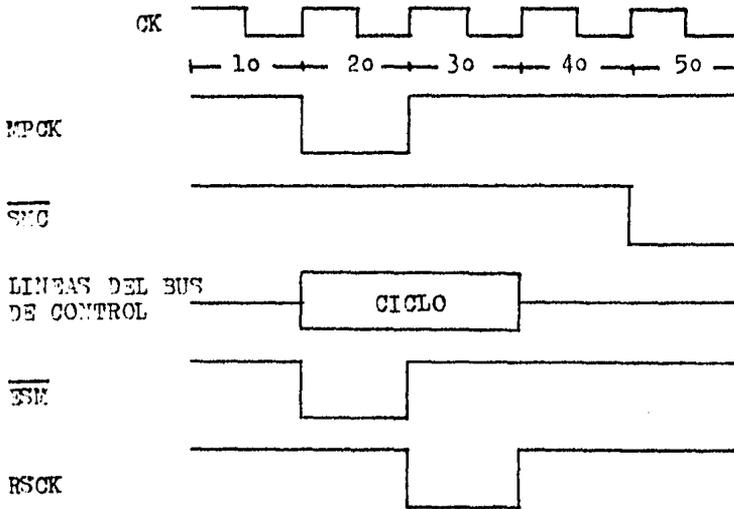


FIGURA IV.2.B.5.- Diagrama de tiempo de las microoperaciones LEO, BINS, ESC, RINT y ALTO.

- 1o.- Durante el primer ciclo de reloj no es activada ninguna línea de control y los bits I/O y $\overline{\text{CBUS}}$ adquieren los valores de 0 y 1 respectivamente.
- 2o.- Durante el segundo ciclo de reloj la línea MPCK es puesta en un cero lógico y las líneas de salida correspondientes al Bus de Control ($\overline{\text{ACTX}}$, $\overline{\text{ESC}}$, $\overline{\text{LEO}}$, $\overline{\text{BI}}$, $\overline{\text{NS}}$, $\overline{\text{ALTO}}$) adquieren los estados lógicos que indican se está realizando un ciclo de máquina; la señal $\overline{\text{FSM}}$ es puesta también en un cero lógico, si es activada la línea de espera $\overline{\text{ESP}}$, el procesador permanecerá en este ciclo de reloj hasta que esta sea desactivada.
- 3o.- Durante el tercer ciclo de reloj son regresadas a un uno lógico las líneas MPCK y $\overline{\text{FSM}}$, esto provoca que el Registro MPC sea incrementado; al mismo tiempo, es puesta en cero lógico la línea (RSCK) encargada -

de censar si han sido activadas las líneas de entrada del Bus de Control $\overline{\text{RBUS}}$ e $\overline{\text{INT}}$. Mientras tanto las líneas de salida del Bus de Control aún continúan indicando que se realiza un ciclo de máquina.

40.- Durante el cuarto ciclo de reloj las líneas de salida del Bus de Control son desactivadas y la línea $\overline{\text{RS}}$ es regresada a un nivel de uno lógico; entonces el Registro $\overline{\text{RS}}$ registra si fueron activadas o no las líneas de $\overline{\text{RBUS}}$ e $\overline{\text{INT}}$ del Bus de Control. Al finalizar el cuarto ciclo de reloj, la Prom de Correspondencia ha tenido tiempo suficiente para poner en sus salidas el código de la siguiente microoperación que se ha de realizar.

50.- Durante el quinto ciclo de reloj son desactivadas todas las líneas de control y la línea $\overline{\text{SMC}}$ es puesta en un cero lógico; entonces el controlador funciona en la forma que ya ha sido descrita en las microoperaciones anteriores.

Microoperación:

CBUS

Esta microoperación tiene la función de realizar un Ciclo de Máquina de Concesión del Bus. El diagrama de tiempo que describe la forma en que será realizada esta microoperación es mostrado en la FIGURA IV.2.B.6.

Los cinco ciclos de reloj en que esta microoperación será realizada son los siguientes:

10- Durante el primer ciclo de reloj no es activada ninguna línea de control y las líneas I/O y $\overline{\text{CBUS}}$ adquieren los valores 1 y 0 respectivamente.

20.- Durante el segundo ciclo de reloj la línea $\overline{\text{MPOCK}}$ es puesta en un cero lógico.

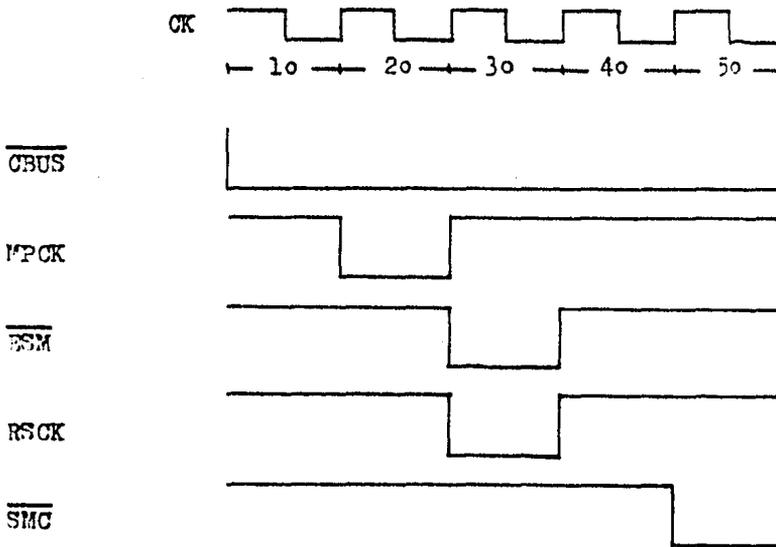


FIGURA IV.2.B.4.- Diagrama de tiempo de la microoperación CBUS.

- 30.- Durante el tercer ciclo de reloj las líneas \overline{ESM} y $RSCK$ encargadas de censar si son activadas las líneas de entrada del Bus de Control \overline{ESP} , \overline{RBUS} e \overline{INT} son -- puestas en un nivel de cero lógico y la línea $MPCK$ es regresada a un nivel de uno lógico, esto provoca que el Registro MPC sea incrementado. Si la línea de espera \overline{ESP} del Bus de Control ha sido activada a lo largo de este ciclo de reloj, el procesador permanecerá en este ciclo hasta que esta línea (\overline{ESP}) sea -- desactivada.
- 40.- Durante el cuarto ciclo de reloj son regresadas a un uno lógico las líneas \overline{ESM} y $RSCK$; al volver $RSCK$ a un estado lógico de uno, el Registro RS registrará -- si son o no activadas las líneas \overline{RBUS} e \overline{INT} del Bus

de Control. Al finalizar el cuarto ciclo de reloj, - la Prom de Correspondencia ha tenido el tiempo suficiente para poner en sus salidas el código de la siguiente microoperación que se ha de realizar.

- 50.- Durante el quinto ciclo de reloj son desactivadas todas las líneas de control y la línea SMC es puesta en un cero lógico; entonces el controlador funciona en la forma que ya ha sido descrita en otras microoperaciones.

Todas las microoperaciones descritas en las secciones anteriores, tanto las de Control de Proceso como las de Controlador y la forma en que estas están relacionadas se muestran en la TABLA IV.2.B.1. En la primera columna de esta tabla se muestra la microoperación de Control de Proceso y en la segunda la microoperación de Control del Controlador que deberá ser realizada en paralelo a esta microoperación. **Aquí existe un gran número de microoperaciones que son realizadas en paralelo, las llamadas dobles;** sin embargo existen algunas microoperaciones de Control del Controlador que son realizadas en forma individual y que no están relacionadas con ninguna de las microoperaciones de Control de Proceso; cuando alguna de estas microoperaciones sea realizada el Campo de Control de Proceso de la Prom de Control permanecerá inactivo; es decir con todos sus bits en un estado de uno lógico (NOP). Finalmente, en la tercera columna aparece lo que es llamado el nombre genérico de la microoperación; con este nombre será designado el par de microoperaciones de la primera y segunda columnas que son realizadas en paralelo; así por ejemplo, al hacer referencia a la microoperación cuyo nombre genérico es PC:=HL en realidad se estará haciendo referencia a dos microoperaciones que son realizadas en parale

lo una de Control de Proceso (PC:=HL) y otra de Control del Controlador (MPC:=MPC+1).

MICROOPERACIONES

CONTROL DE PROCESO	CONTROL DEL CONTROLADOR	NOMBRE GENERICO
PG:=HL	MPC:=MPC+1	PC:=HL
T:=PCH	MPC:=MPC+1	T:=PCH
T:=PCL	MPC:=MPC+1	T:=PCL
L:=A	MPC:=MPC+1	L:=A
H:=A	MPC:=MPC+1	H:=A
S:=A	MPC:=MPC+1	S:=A
S:=F	MPC:=MPC+1	S:=F
S:=PC	MPC:=MPC+1	S:=PC
A:=S	MPC:=MPC+1	A:=S
F:=S	MPC:=MPC+1	F:=S
PC:=S	MPC:=MPC+1	PC:=S
A:=(MAPC)	LEC	A:=(MAPC)
T:=(MAPC)	LEC	T:=(MAPC)
L:=(MAPC)	LEC	L:=(MAPC)
H:=(MAPC)	LEC	H:=(MAPC)
RI:=(MAPC)	BINS	RI:=(MAPC)
(MAPC):=A	ESC	(MAPC):=A
L:=INT	RINT	L:=INT
A:=ACOD	MPC:=MPC+1	A:=ACOD
A:=ACAOD	MPC:=MPC+1	A:=ACAOD
A:=ACA7D	MPC:=MPC+1	A:=ACA7D
A:=ACAD	MPC:=MPC+1	A:=ACAD
A:=ACOI	MPC:=MPC+1	A:=ACOI
A:=ACAOI	MPC:=MPC+1	A:=ACAOI
A:=ACA7I	MPC:=MPC+1	A:=ACA7I
A:=ACAI	MPC:=MPC+1	A:=ACAI

TABLA IV.2.B.1.- Microoperaciones de Control de Proceso y de Control del Controlador

A-T	MPC:=MPC+1	A-T
A:=A-T	MPC:=MPC+1	A:=A-T
A:=A-T-a	MPC:=MPC+1	A:=A-T-a
A:=A \bar{m} T	MPC:=MPC+1	A:=A \bar{m} T
A:=A+T	MPC:=MPC+1	A:=A+T
A:=A+T+a	MPC:=MPC+1	A:=A+T+a
F(a,z):=(0,1)	MPC:=MPC+1	F(a,z):=(0,1)
A:=A+1	MPC:=MPC+1	A:=A+1
A:= \bar{A}	MPC:=MPC+1	A:= \bar{A}
F(a,z):=(\bar{a} , \bar{z})	MPC:=MPC+1	F(a,z):=(\bar{a} , \bar{z})
A:=A-1	MPC:=MPC+1	A:=A-1
A:=AorT	MPC:=MPC+1	A:=AorT
A:=AT	MPC:=MPC+1	A:=AT
A:=OOH	MPC:=MPC+1	A:=OOH
F(a,z):=(0,0)	MPC:=MPC+1	F(a,z):=(0,0)
NOP	MPC:=DR	MPC:=DR
NOP	MPC:=RS	MPC:=RS
NOP	MPC:=RI	MPC:=RI
NOP	MPC:=SD(AD)	MPC:=SD(AD)
NOP	MPC:=SD(TD)	MPC:=SD(TD)
NOP	MPC:=SD(AI)	MPC:=SD(AI)
NOP	MPC:=SD(TI)	MPC:=SD(TI)
NOP	MPC:=SD(DA)	MPC:=SD(DA)
NOP	MPC:=SD(IA)	MPC:=SD(IA)
NOP	MPC:=SD,Z(RN)	MPC:=SD,Z(RN)
NOP	MPC:=SD,Z(NN)	MPC:=SD,Z(NN)
NOP	MPC:=SD,Z(PI)	MPC:=SD,Z(PI)
NOP	MPC:=SD,Z(SN)	MPC:=SD,Z(SN)
NOP	MPC:=SD,Z(SP)	MPC:=SD,Z(SP)
NOP	MPC:=SD,Z(RT)	MPC:=SD,Z(RT)
NOP	MPC:=SD,NZ(RN)	MPC:=SD,NZ(RN)
NOP	MPC:=SD,NZ(NN)	MPC:=SD,NZ(NN)
NOP	MPC:=SD,NZ(PI)	MPC:=SD,NZ(PI)
NOP	MPC:=SD,NZ(SN)	MPC:=SD,NZ(SN)
NOP	MPC:=SD,NZ(SP)	MPC:=SD,NZ(SP)
NOP	MPC:=SD,NZ(RT)	MPC:=SD,NZ(RT)

NOP	MPC:=SD,A(RN)	MPC:=SD,A(RN)
NOP	MPC:=SD,A(NN)	MPC:=SD,A(NN)
NOP	MPC:=SD,A(PI)	MPC:=SD,A(PI)
NOP	MPC:=SD,A(SN)	MPC:=SD,A(SN)
NOP	MPC:=SD,A(SP)	MPC:=SD,A(SP)
NOP	MPC:=SD,A(RT)	MPC:=SD,A(RT)
NOP	MPC:=SD,NA(RN)	MPC:=SD,NA(RN)
NOP	MPC:=SD,NA(NN)	MPC:=SD,NA(NN)
NOP	MPC:=SD,NA(PI)	MPC:=SD,NA(PI)
NOP	MPC:=SD,NA(SN)	MPC:=SD,NA(SN)
NOP	MPC:=SD,NA(SP)	MPC:=SD,NA(SP)
NOP	MPC:=SD,NA(RT)	MPC:=SD,NA(RT)
NOP	ALTO	ALTO
NOP	RBUS	RBUS

TABLA IV.2.B.1.- (Conclusión).

IV.3.- CONTENIDO DE LA PROM DE CONTROL. En la Prom de Control estarán grabadas todas las microoperaciones descritas - en la sección IV.2. Para definir el contenido de la Prom de Control se usará el código Hexadecimal que resulta ser de fácil traducción al Binario; de esta forma los 32 bits de la Prom de Control podrán ser escritos con solo 8 dígitos hexadecimales. El formato para describir al contenido de Prom de Control consta de tres partes; en primer lugar es dado el nombre genérico de la microoperación en lenguaje de asignación, enseguida de este nombre es dada la Microdirección de Control correspondiente a la primera microinstrucción realizada por esta microoperación y finalmente es dada la microinstrucción, todo esto en código Hexadecimal. Todas las microoperaciones que serán dadas enseguida están referidas a los diagramas de tiempo de la sección IV.2 y cada microinstrucción es realizada en un ciclo de reloj del diagrama de tiempo. La forma en que están relacionados los códigos binario y hexadecimal se da en la TABLA IV.3.1.

Código Binario	Código Hexadecimal
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	A
1011	B
1100	C
1101	D
1110	E
1111	F

TABLA IV.3.1.- Conversión de Códigos Binario a Hexadecimal.

El contenido de la Prom de Control se da a continuación; en donde los dígitos Hexadecimales menos significativos son los que se encuentran a la derecha de las columnas y los mas significativos a la izquierda.

RI:=(MAPC):	000	FD	FF	BF	FF
	001	F5	CD	BF	6B
	002	FD	CE	BF	EB
	003	FD	FF	BF	F7
	004	F9	FF	BF	FF
	005	FF	FF	FF	FF
	:	:	:	:	:
	00F	FF	FF	FF	FF
ALTO:	010	FD	FF	FF	FF
	011	F5	F9	FF	FF
	012	FD	FA	FF	FF
	013	FD	FF	FF	FF
	014	F9	FF	FF	FF
	015	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
	01F	FF	FF	FF	FF

CBUS:	020	FE	FF	FF	FF
	021	F6	FF	FF	FF
	022	FE	FC	FF	FF
	023	FE	FF	FF	FF
	024	FA	FF	FF	FF
	025	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
	03F	FF	FF	FF	FF
PC:=HL:	040	FF	FF	3F	FF
	041	B7	FF	3F	B3
	042	FF	FF	3F	F3
	043	FF	FF	3F	FF
	044	FB	FF	3F	FF
	045	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
T:=PCH:	050	FF	FF	3F	FF
	051	F7	FF	31	7F
	052	FF	FF	39	7F
	053	FF	FF	3F	FF
	054	FB	FF	3F	FF
	055	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
T:=PCL:	060	FF	FF	3F	FF
	061	F7	FF	32	FF
	062	FF	FF	3A	FF
	063	FF	FF	3F	FF
	064	FB	FF	3F	FF
	065	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
L:=A:	070	FF	FF	3F	FF
	071	F7	FF	3F	DD
	072	FF	FF	3F	FD
	073	FF	FF	3F	FF
	074	FB	FF	3F	FF
	075	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
H:=A:	080	FF	FF	3F	FF
	081	F7	FF	3F	ED
	082	FF	FF	3F	FD
	083	FF	FF	3F	FF
	084	FB	FF	3F	FF
	085	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:

S:=A:	090	FF	FF	7F	FF
	091	F7	FF	7F	F3
	092	FF	FF	7F	E7
	093	FF	FF	7E	E5
	094	FB	FF	7F	FF
	095	FF	FF	FF	FF
	:	:	:	:	:
S:=F:	0A0	FF	FF	7F	FF
	0A1	F7	FF	7F	F3
	0A2	FF	FF	7F	E7
	0A3	FF	FF	7C	E7
	0A4	FB	FF	7F	FF
	0A5	FF	FF	FF	FF
	:	:	:	:	:
S:=PC:	0B0	FF	FF	7F	FF
	0B1	F7	FF	7F	F3
	0B2	FF	FF	7F	E7
	0B3	FF	FF	7F	87
	0B4	FB	FF	7F	FF
	0B5	FF	FF	FF	FF
	:	:	:	:	:
A:=S:	0C0	FF	FF	7F	FF
	0C1	F7	FF	7F	6E
	0C2	FF	FF	7F	6B
	0C3	FF	FF	7F	FF
	0C4	FB	FF	7F	FF
	0C5	FF	FF	FF	FF
	:	:	:	:	:
F:=S:	0D0	FF	FF	7F	FF
	0D1	F7	FF	7B	6F
	0D2	FF	FF	7F	6B
	0D3	FF	FF	7F	FF
	0D4	FB	FF	7F	FF
	0D5	FF	FF	FF	FF
	:	:	:	:	:
PC:=S:	0E0	FF	FF	7F	FF
	0E1	F7	FF	77	EF
	0E2	FF	FF	7F	EB
	0E3	FF	FF	7F	FF
	0E4	FB	FF	7F	FF
	0E5	FF	FF	FF	FF
	:	:	:	:	:

A:=(MAPC):	OF0	FF	FF	BF	FF
	OF1	F5	5D	BF	EA
	OF2	FD	5E	BF	EB
	OF3	FD	FF	BF	FF
	OF4	F9	FF	BF	FF
	OF5	FF	FF	FF	FF
	:	:	:	:	:
T:=(MAPC):	100	FF	FF	BF	FF
	101	F5	5D	BF	AD
	102	FD	5E	BF	EB
	103	FD	FF	BF	F7
	104	F9	FF	BF	FF
	105	FF	FF	FF	FF
	:	:	:	:	:
L:=(MAPC):	110	FF	FF	BF	FF
	111	F5	5D	BD	EB
	112	FD	5E	BF	EB
	113	FD	FF	BF	F7
	114	F9	FF	BF	FF
	115	FF	FF	FF	FF
	:	:	:	:	:
M:=(MAPC):	120	FF	FF	BF	FF
	121	F5	5D	BE	EB
	122	FD	5E	BF	EB
	123	FD	FF	BF	F7
	124	F9	FF	BF	FF
	125	FF	FF	FF	FF
	:	:	:	:	:
PC:=PC+1	130	FF	FF	BF	FF
	131	F7	FF	BF	FF
	132	FF	FF	BF	FF
	133	FF	FF	BF	F7
	134	FB	FF	BF	FF
	135	FF	FF	FF	FF
	:	:	:	:	:
(MAPC):=A:	140	FD	FF	BF	FF
	141	F5	7D	BF	D9
	142	FD	3E	BF	D9
	143	FD	FF	BF	F7
	144	F9	FF	BF	FF
	145	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:

L:=INT	150	FD	FF	BF	FF
	151	F5	D5	BD	EF
	152	FD	D6	BF	EF
	153	FD	FF	BF	FF
	154	F9	FF	BF	FF
	155	FF	FF	FF	FF
	:	:	:	:	:
A:=ACOD:	160	FF	FF	BF	FF
	161	F7	FF	D8	FF
	162	FF	FF	B8	FE
	163	FF	FF	88	FF
	164	FB	FF	AF	FF
	165	FF	FF	FF	FF
	:	:	:	:	:
A:=ACAOD:	170	FF	FF	BF	FF
	171	F7	FF	BA	FF
	172	FF	FF	BA	FE
	173	FF	FF	8A	FF
	174	FB	FF	AF	FF
	175	FF	FF	FF	FF
	:	:	:	:	:
A:=ACA7D	180	FF	FF	BF	FF
	181	F7	FF	B9	FF
	182	FF	FF	B9	FE
	183	FF	FF	89	FF
	184	FB	FF	AF	FF
	185	FF	FF	FF	FF
	:	:	:	:	:
A:=ACAD:	190	FF	FF	BF	FF
	191	F7	FF	BB	FF
	192	FF	FF	BB	FE
	193	FF	FF	8B	FF
	194	FB	FF	AF	FF
	195	FF	FF	FF	FF
	:	:	:	:	:
A:=ACOI:	1A0	FF	FF	BF	FF
	1A1	F7	FF	B4	FF
	1A2	FF	FF	B4	FE
	1A3	FF	FF	84	FF
	1A4	FB	FF	AF	FF
	1A5	FF	FF	FF	FF
	:	:	:	:	:

A:=ACAOI:	1B0	FF	FF	BF	FF
	1B1	F7	FF	B6	FF
	1B3	FF	FF	86	FF
	1B4	FB	FF	AF	FF
	1B5	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
A:=ACA7I:	1C0	FF	FF	BF	FF
	1C1	F7	FF	B5	FF
	1C2	FF	FF	B5	FF
	1C3	FF	FF	85	FF
	1C4	FB	FF	AF	FF
	1C5	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
A:=ACAI:	1D0	FF	FF	BF	FF
	1D1	F7	FF	B7	FF
	1D2	FF	FF	B7	FF
	1D3	FF	FF	87	FF
	1D4	FB	FF	AF	FF
	1D5	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
A-T:	1F0	FF	FF	FF	FF
	1F1	F7	FF	D8	99
	1F2	FF	FF	F8	99
	1F3	FF	FF	FF	FF
	1F4	FB	FF	FF	FF
	1F5	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
A:=A-T:	200	FF	FF	FF	FF
	201	F7	FF	D8	98
	202	FF	FF	F8	99
	203	FF	FF	FF	FF
	204	FB	FF	FF	FF
	205	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
A:=A-T-r:	210	FF	FF	FF	FF
	211	F7	FF	D9	18
	212	FF	FF	F9	19
	213	FF	FF	FF	FF
	214	FB	FF	FF	FF
	215	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:

A:=A+T:	220	FF	FF	FF	FF
	221	F7	FF	D9	D8
	222	FF	FF	F9	D9
	223	FF	FF	FF	FF
	224	FB	FF	FF	FF
	225	FF	FF	FF	FF
	:	:	:	:	:
A:=A+T:	230	FF	FF	FF	FF
	231	F7	FF	D8	A4
	232	FF	FF	F8	A5
	233	FF	FF	FF	FF
	234	FB	FF	FF	FF
	235	FF	FF	FF	FF
	:	:	:	:	:
A:=A+T+a:	240	FF	FF	FF	FF
	241	F7	FF	D9	24
	242	FF	FF	F9	25
	243	FF	FF	FF	FF
	244	FB	FF	FF	FF
	245	FF	FF	FF	FF
	:	:	:	:	:
F(a,z):=(0,1):	250	FF	FF	FF	FF
	251	F7	FF	DB	FF
	252	FF	FF	FB	FF
	253	FF	FF	FF	FF
	254	FB	FF	FF	FF
	255	FF	FF	FF	FF
	:	:	:	:	:
A:=A+1:	260	FF	FF	FF	FF
	261	F7	FF	D9	80
	262	FF	FF	F9	81
	263	FF	FF	FF	FF
	264	FB	FF	FF	FF
	265	FF	FF	FF	FF
	:	:	:	:	:
A:=\bar{A}	270	FF	FF	FF	FF
	271	F7	FF	D9	C0
	272	FF	FF	F9	C1
	273	FF	FF	FF	FF
	274	FB	FF	FF	FF
	275	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:

$F(a, z) := (\bar{a}, \bar{z}) :$	280	FF	FF	FF	FF
	281	F7	FF	D7	FF
	282	FF	FF	F7	FF
	283	FF	FF	FF	FF
	284	FB	FF	FF	FF
	285	FF	FF	FF	FF
	:	:	:	:	:
$A := A-1 :$	290	FF	FF	FF	FF
	291	F7	FF	D9	BC
	292	FF	FF	F9	BD
	293	FF	FF	FF	FF
	294	FB	FF	FF	FF
	295	FF	FF	FF	FF
	:	:	:	:	:
$A := A \text{ or } T :$	2A0	FF	FF	FF	FF
	2A1	F7	FF	D9	DC
	2A2	FF	FF	F9	DD
	2A3	FF	FF	FF	FF
	2A4	FB	FF	FF	FF
	2A5	FF	FF	FF	FF
	:	:	:	:	:
$A := AT :$	2B0	FF	FF	FF	FF
	2B1	F7	FF	D9	F4
	2B2	FF	FF	F9	F5
	2B3	FF	FF	FF	FF
	2B4	FB	FF	FF	FF
	2B5	FF	FF	FF	FF
	:	:	:	:	:
$A := 00H :$	2C0	FF	FF	FF	FF
	2C1	F7	FF	D9	F0
	2C2	FF	FF	F9	F1
	2C3	FF	FF	FF	FF
	2C4	FB	FF	FF	FF
	2C5	FF	FF	FF	FF
	:	:	:	:	:
$F(a, z) := (0, 0) :$	2D0	FF	FF	FF	FF
	2D1	F7	FF	C7	FF
	2D2	FF	FF	E7	FF
	2D3	FF	FF	FF	FF
	2D4	FB	FF	FF	FF
	2D5	FF	FF	FF	FF
	:	:	:	:	:

MPC:=DR:	2E0	FF	FF	FF	FF
	2E1	CF	BF	FF	FF
	2E2	CF	FF	FF	FF
	2E3	FB	FF	FF	FF
	2E4	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
MPC:=RS:	2F0	FF	FF	FF	FF
	2F1	DF	BF	FF	FF
	2F2	DF	FF	FF	FF
	2F3	FB	FF	FF	FF
	2F4	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
MPC:=RI:	300	FF	FF	FF	FF
	301	FF	BF	FF	FF
	302	FF	FF	FF	FF
	303	FB	FF	FF	FF
	304	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
MPC:=SD,(AD):	310	FF	FF	FF	FF
	311	BF	FF	FF	FF
	312	FF	FF	FF	FF
	313	6F	A0	FF	FF
	314	EF	E0	FF	FF
	315	FB	FF	FF	FF
	316	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
MPC:=SD,(TD):	320	FF	FF	FF	FF
	321	BF	FF	FF	FF
	322	FF	FF	FF	FF
	323	6F	A2	FF	FF
	324	EF	E2	FF	FF
	325	FB	FF	FF	FF
	326	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
MPC:=SD,(AI):	330	FF	FF	FF	FF
	331	BF	FF	FF	FF
	332	FF	FF	FF	FF
	333	6F	A4	FF	FF
	334	EF	E4	FF	FF
	335	FB	FF	FF	FF
	336	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:

MPC:=SD,(TI):	340	FF	FF	FF	FF
	341	BF	FF	FF	FF
	342	FF	FF	FF	FF
	343	6F	A6	FF	FF
	344	EF	E6	FF	FF
	345	FB	FF	FF	FF
	346	FF	FF	FF	FF
	:	:	:	:	:
MPC:=SD,(DA)	350	FF	FF	FF	FF
	351	BF	FF	FF	FF
	352	FF	FF	FF	FF
	353	6F	A8	FF	FF
	354	EF	E8	FF	FF
	355	FB	FF	FF	FF
	356	FF	FF	FF	FF
	:	:	:	:	:
MPC:=SD,(IA)	360	FF	FF	FF	FF
	361	BF	FF	FF	FF
	362	FF	FF	FF	FF
	363	6F	AA	FF	FF
	364	EF	EA	FF	FF
	365	FB	FF	FF	FF
	366	FF	FF	FF	FF
	:	:	:	:	:
MPC:=SD,Z(RN):	370	FF	FF	FF	FF
	371	F7	FF	FF	FF
	372	EF	36	FF	FF
	373	EF	F6	FF	FF
	374	FB	FF	FF	FF
	375	FF	FF	FF	FF
	:	:	:	:	:
MPC:=SD,Z(NN):	380	FF	FF	FF	FF
	381	F7	FF	FF	FF
	382	EF	2E	FF	FF
	383	EF	E6	FF	FF
	384	FB	FF	FF	FF
	385	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
	:	:	:	:	:
	:	:	:	:	:

MPC:=SD, Z(PI):	390	FF	FF	FF	FF
	391	E7	FF	FF	FF
	392	EF	30	FF	FF
	393	EF	F0	FF	FF
	394	FB	FF	FF	FF
	395	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
MPC:=SD, Z(SN):	3A0	FF	FF	FF	FF
	3A1	F7	FF	FF	FF
	3A2	EF	32	FF	FF
	3A3	EF	F2	FF	FF
	3A4	FB	FF	FF	FF
	3A5	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
MPC:=SD, Z(SP):	3B0	FF	FF	FF	FF
	3B1	F7	FF	FF	FF
	3B2	EF	34	FF	FF
	3B3	EF	F4	FF	FF
	3B4	FB	FF	FF	FF
	3B5	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
MPC:=SD, Z(RT):	3C0	FF	FF	FF	FF
	3C1	F7	FF	FF	FF
	3C2	EF	2C	FF	FF
	3C3	EF	EC	FF	FF
	3C4	FB	FF	FF	FF
	3C5	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
MPC:=SD, NZ(RN):	3D0	FF	FF	FF	FF
	3D1	F7	FF	FF	FF
	3D2	EF	16	FF	FF
	3D3	EF	F6	FF	FF
	3D4	FB	FF	FF	FF
	3D5	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
MPC:=SD, NZ(NN):	3E0	FF	FF	FF	FF
	3E1	F7	FF	FF	FF
	3E2	EF	0E	FF	FF
	3E3	EF	EE	FF	FF
	3E4	FB	FF	FF	FF
	3E5	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:

MPC:=SD,NZ(PI):	3F0	FF	FF	FF	FF
	3F1	F7	FF	FF	FF
	3F2	EF	10	FF	FF
	3F3	EF	F0	FF	FF
	3F4	FB	FF	FF	FF
	3F5	FF	FF	FF	FF
	:	:	:	:	:
MPC:=SD,NZ(SN):	400	FF	FF	FF	FF
	401	E7	FF	FF	FF
	402	EF	12	FF	FF
	403	EF	F2	FF	FF
	404	FB	FF	FF	FF
	405	FF	FF	FF	FF
	:	:	:	:	:
MPC:=SD,NZ(SP):	410	FF	FF	FF	FF
	411	F7	FF	FF	FF
	412	EF	14	FF	FF
	413	EF	F4	FF	FF
	414	FB	FF	FF	FF
	415	FF	FF	FF	FF
	:	:	:	:	:
MPC:=SD,NZ(RT):	420	FF	FF	FF	FF
	421	F7	FF	FF	FF
	422	EF	0C	FF	FF
	423	EF	BC	FF	FF
	424	FB	FF	FF	FF
	425	FF	FF	FF	FF
	:	:	:	:	:
MPC:=SD,A(RN):	430	FF	FF	FF	FF
	431	F7	FF	FF	FF
	432	EF	76	FF	FF
	433	EF	F6	FF	FF
	434	FB	FF	FF	FF
	435	FF	FF	FF	FF
	:	:	:	:	:
MPC:=SD,A(NN):	440	FF	FF	FF	FF
	441	F7	FF	FF	FF
	442	EF	6E	FF	FF
	443	EF	BE	FF	FF
	444	FB	FF	FF	FF
	445	FF	FF	FF	FF
	:	:	:	:	:

MPC:=SD,A(PI):	450	FF	FF	FF	FF
	451	F7	FF	FF	FF
	452	EF	70	FF	FF
	453	EF	F0	FF	FF
	454	FB	FF	FF	FF
	455	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
MPC:=SD,A(SN):	460	FF	FF	FF	FF
	461	F7	FF	FF	FF
	462	EF	72	FF	FF
	463	EF	F2	FF	FF
	464	FB	FF	FF	FF
	465	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
MPC:=SD,A(SP):	470	FF	FF	FF	FF
	471	F7	FF	FF	FF
	472	EF	74	FF	FF
	473	EF	F4	FF	FF
	474	FB	FF	FF	FF
	475	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
MPC:=SD,A(RT):	480	FF	FF	FF	FF
	481	F7	FF	FF	FF
	482	EF	6C	FF	FF
	483	EF	BC	FF	FF
	484	FB	FF	FF	FF
	485	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
MPC:=SD,NA(RN):	490	FF	FF	FF	FF
	491	F7	FF	FF	FF
	492	EF	56	FF	FF
	493	EF	F6	FF	FF
	494	FB	FF	FF	FF
	495	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
MPC:=SD,NA(NN):	4A0	FF	FF	FF	FF
	4A1	F7	FF	FF	FF
	4A2	EF	4E	FF	FF
	4A3	EF	BE	FF	FF
	4A4	FB	FF	FF	FF
	4A5	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:

MPC:=SD,NA(PI):	4B0	FF	FF	FF	FF
	4B1	F7	FF	FF	FF
	4B2	EF	50	FF	FF
	4B3	EF	F0	FF	FF
	4B4	FB	FF	FF	FF
	4B5	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
MPC:=SD,NA(SN):	4C0	FF	FF	FF	FF
	4C1	F7	FF	FF	FF
	4C2	EF	52	FF	FF
	4C3	EF	F2	FF	FF
	4C4	FB	FF	FF	FF
	4C5	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
MPC:=SD,NA(SP):	4D0	FF	FF	FF	FF
	4D1	F7	FF	FF	FF
	4D2	EF	54	FF	FF
	4D3	EF	F4	FF	FF
	4D4	FB	FF	FF	FF
	4D5	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:
MPC:=SD,NA(RT):	4E0	FF	FF	FF	FF
	4E1	F7	FF	FF	FF
	4E2	EF	4C	FF	FF
	4E3	EF	BC	FF	FF
	4E4	FB	FF	FF	FF
	4E5	FF	FF	FF	FF
	:	:	:	:	:
	:	:	:	:	:

IV.4.- CONTENIDO DE LA PROM DE CORRESPONDENCIA. Para describir al contenido de la Prom de Correspondencia se empleará el siguiente formato; Primeramente será dada la parte del nemotecnico de la instrucción u operación que describe el funcionamiento de la misma y en caso necesario la condición; solo se indicarán las variables en casos especiales para evitar confusión. Inmediatamente despues será dada la serie de microoperaciones que integran la microrrutina mediante la cual la instrucción u operación será realizada, estas microoperaciones serán dadas en lenguaje de asignación; en la siguiente columna, en Hexadecimal, serán dadas las microdirecciones de la Prom de Correspondencia en las cuales se ha de programar el código de la microoperación que se ha de realizar; este código será dado finalmente en la cuarta columna.

BINS:	RI:=(MAPC)	000	00
	MPC:=RI-	001	60
	NOP	002	FF
	:	:	:
	:	:	:
RINT:	L:=INT	008	2A
	S:=A	009	12
	S:=F	00A	14
	S:=PC	00B	16
	A:=00H	00C	58
	H:=A	00D	10
	PC:=HL	00E	08
	MPC:=RS	00F	5E
	NOP	010	FF
	:	:	:
	:	:	:
RBUS:	CBUS	018	04
	MPC:=RS	019	5E
	NOP	01A	FF
	:	:	:
	:	:	:
POP A:	A:=S	040	18
	MPC:=RS	041	5E
	NOP	042	FF
	:	:	:

GARN:	A:=(MAPC)	050	1E
	MPC:=RS	051	5E
	NOP	052	FF
	:	:	:
CARD:	MPC:=SD, (AD)	060	62
	MPC:=RS	061	5E
	NOP	062	FF
	:	:	:
CARI:	MPC:=SD, (AI)	070	66
	MPC:=RS	071	5E
	NOP	072	FF
	:	:	:
PUSH A:	S:=A	080	12
	MPC:=RS	081	5E
	NOP	082	FF
	:	:	:
DESD:	MPC:=SD, (DA)	090	6A
	MPC:=RS	091	5E
	NOP	092	FF
	:	:	:
DESI:	MPC:=SD, (IA)	0A0	6C
	MPC:=RS	0A1	5E
	NOP	0A2	FF
	:	:	:
POP F:	F:=S	0B0	1A
	MPC:=RS	0B1	5E
	NOP	0B2	FF
	:	:	:
PUSH F:	S:=F	0C0	14
	MPC:=RS	0C1	5E
	NOP	0C2	FF
	:	:	:
OPN:	T:=(MAPC)	0D0	20
	A-T	0D1	3E
	MPC:=RS	0D2	5E
	NOP	0D3	FF
	:	:	:
GPD:	MPC:=SD, (TD)	0E0	64
	A-T	0E1	3E
	MPC:=RS	0E2	5E
	NOP	0E3	FF
	:	:	:

CPI:	MPC:=SD, (TI)	OF0	68
	A-T	OF1	3E
	MPC:=RS	OF2	5E
	NOP	OF3	FF
	:	:	:
SUMN:	T:=(MAPC)	100	20
	A:=A+T	101	46
	MPC:=RS	102	5E
	NOP	103	FF
	:	:	:
SUMD:	MPC:=SD, (TD)	110	64
	A:=A+T	111	46
	MPC:=RS	112	5E
	NOP	113	FF
	:	:	:
SUMI:	MPC:=SD, (TI)	120	68
	A:=A+T	121	46
	MPC:=RS	121	5E
	NOP	123	FF
	:	:	:
SUMAN:	T:=(MAPC)	130	20
	A:=A+T+a	131	48
	MPC:=RS	132	5E
	NOP	133	FF
	:	:	:
SUMAD:	MPC:=SD, (TD)	140	64
	A:=A+T+a	141	48
	MPC:=RS	142	5E
	NOP	143	FF
	:	:	:
SUMAI:	MPC:=SD, (TI)	150	68
	A:=A+T+a	151	48
	MPC:=RS	152	5E
	NOP	153	FF
	:	:	:
RESN:	T:=(MAPC)	160	20
	A:=A-T	161	40
	MPC:=RS	162	5E
	NOP	163	FF
	:	:	:
RESD:	MPC:=SD, (TD)	170	64

	A:=A-T	171	40
	MPC:=RS	172	5E
	NOP	173	FF
	:	:	:
RESI:	MPC:=SD, (TI)	180	68
	A:=A-T	181	40
	MPC:=RS	182	5E
	NOP	183	FF
	:	:	:
RESAN:	T:=(MAPC)	190	20
	A:=A-T-a	191	42
	MPC:=RS	192	5E
	NOP	193	FF
	:	:	:
RESAD:	MPC:=SD, (TD)	1A0	64
	A:=A-T-a	1A1	42
	MPC:=RS	1A2	5E
	NOP	1A3	FF
	:	:	:
RESAI:	MPC:=SD, (TI)	1B0	68
	A:=A-T-a	1B1	42
	MPC:=RS	1B2	5E
	NOP	1B3	FF
	:	:	:
INCR:	A:=A+1	1C0	4C
	MPC:=RS	1C1	5E
	NOP	1C2	FF
	:	:	:
INCRD:	L:=(MAPC)	1D0	22
	H:=(MAPC)	1D1	24
	S:=PC	1D2	16
	PG:=HL	1D3	08
	A:=(MAPC)	1D4	1E
	A:=A+1	1D5	4C
	PG:=HL	1D6	08
	(MAPC):=A	1D7	28
	PG:=S	1D8	1C
	MPC:=RS	1D9	5E
	NOP	1DA	FF
	:	:	:
INCRI:	L:=(MAPC)	1E0	22
	H:=(MAPC)	1E1	24

	S:=PC	1E2	16
	PC:=HL	1E3	08
	L:=(MAPC)	1E4	22
	H:=(MAPC)	1E5	24
	PC:=HL	1E6	08
	A:=(MAPC)	1E7	1E
	A:=A+1	1E8	4C
	PC:=HL	1E9	08
	(MAPC):=A	1EA	28
	PC:=S	1EB	1C
	MPC:=RS	1EC	5E
	NOP	1ED	FF
	:	:	:
	:	:	:
DECRR:	A:=A-1	1F0	52
	MPC:=RS	1F1	5E
	NOP	1F2	FF
	:	:	:
	:	:	:
DECRD:	L:=(MAPC)	200	22
	H:=(MAPC)	201	24
	S:=PC	202	16
	PC:=HL	203	08
	A:=(MAPC)	204	1E
	A:=A-1	205	52
	PC:=HL	206	08
	(MAPC):=A	207	28
	PC:=S	208	1C
	MPC:=RS	209	5E
	NOP	20A	FF
	:	:	:
	:	:	:
DECRI:	L:=(MAPC)	210	22
	H:=(MAPC)	211	24
	S:=PC	212	16
	PC:=HL	213	08
	L:=(MAPC)	214	22
	H:=(MAPC)	215	24
	PC:=HL	216	08
	A:=(MAPC)	217	1E
	A:=A-1	218	52
	PC:=HL	219	08
	(MAPC):=A	21A	28
	PC:=S	21B	1C
	MPC:=RS	21C	5E
	NOP	21D	FF
	:	:	:
	:	:	:

RDER:	A:=ACAOD	220	2E
	MPC:=RS	221	5E
	NOP	222	FF
	:	:	:
RIZR:	A:=ACA7I	230	38
	MPC:=RS	231	5E
	NOP	232	FF
	:	:	:
RDEAR:	A:=ACAD	240	32
	MPC:=RS	241	5E
	NOP	242	FF
	:	:	:
RIZAR:	A:=ACAI	250	3A
	MPC:=RS	251	5E
	NOP	252	FF
	:	:	:
CDER:	A:=ACOD	260	2C
	MPC:=RS	261	5E
	NOP	262	FF
	:	:	:
CIZR:	A:=ACOI	270	34
	MPC:=RS	271	5E
	NOP	272	FF
	:	:	:
DDER:	A:=ACA7D	280	30
	MPC:=RS	281	5E
	NOP	282	FF
	:	:	:
DIZR:	A:=ACAOI	290	36
	MPC:=RS	291	5E
	NOP	292	FF
	:	:	:
ANDR:	$F(a, z) := (\bar{a}, \bar{z})$	2A0	50
	MPC:=RS	2A1	5E
	NOP	2A2	FF
	:	:	:
ANDN:	T:=(MAPC)	2B0	20
	A:=AT	2B1	56
	MPC:=RS	2B2	5E
	NOP	2B3	FF
	:	:	:

ANDD:	MPC:=SD, (TD)	2C0	64
	A:=AT	2C1	56
	MPC:=RS	2C2	5E
	NOP	2C3	FF
	:	:	:
ANDI:	MPC:=SD, (TI)	2D0	68
	A:=AT	2D1	56
	MPC:=RS	2D2	5E
	NOP	2D3	FF
	:	:	:
ORR:	$F(a, z) := (0, 0)$	2E0	5A
	MPC:=RS	2E1	5E
	NOP	2E2	FF
	:	:	:
	:	:	:
ORN:	T:=(MAPC)	2F0	20
	A:=AorT	2F1	54
	MPC:=RS	2F2	5E
	NOP	2F3	FF
	:	:	:
ORD:	MPC:=SD, (TD)	300	64
	A:=AorT	301	54
	MPC:=RS	302	5E
	NOP	303	FF
	:	:	:
ORI:	MPC:=SD, (TI)	310	68
	A:=AorT	311	54
	MPC:=RS	312	5E
	NOP	313	FF
	:	:	:
EXR:	A:=00H	320	58
	$F(a, z) := (0, 1)$	321	4A
	MPC:=RS	322	5E
	NOP	323	FF
	:	:	:
EXN:	T:=(MAPC)	330	20
	A:=AorT	331	44
	MPC:=RS	332	5E
	NOP	333	FF
	:	:	:
EXD:	MPC:=SD, (TD)	340	64
	A:=AorT	341	44
	MPC:=RS	342	5E

	NOP	343	FF
	:	:	:
EXI:	MPC:=SD, (TI)	350	68
	A:=A+T	351	44
	MPC:=RS	352	5E
	NOP	353	FF
	:	:	:
NOTR:	A:= \bar{A}	360	4E
	MPC:=RS	361	5E
	NOP	362	FF
	:	:	:
GDOS:	A:= \bar{A}	370	4E
	A:=A+1	371	4C
	MPC:=RS	372	5E
	NOP	373	FF
	:	:	:
SALTON:	L:=(MAPC)	380	22
	H:=(MAPC)	381	24
	PC:=HL	382	08
	MPC:=RS	383	5E
	NOP	384	FF
	:	:	:
SALRN, NZ:	MPC:=SD, NZ(RN)	390	7A
	PC:=PC+1	391	3C
	MPC:=RS	392	5E
	NOP	393	FF
	:	:	:
SALNN, NZ:	MPC:=SD, NZ(NN)	3A0	7C
	PC:=PC+1	3A1	3C
	PC:=PC+1	3A2	3C
	MPC:=RS	3A3	5E
	NOP	3A4	FF
	:	:	:
SALPI, NZ:	MPC:=SD, NZ(PI)	3B0	7E
	PC:=PC+1	3B1	3C
	PC:=PC+1	3B2	3C
	MPC:=RS	3B3	5E
	NOP	3B4	FF
	:	:	:
SALRN, Z:	MPC:=SD, Z(RN)	3C0	6E
	PC:=PC+1	3C1	3C
	MPC:=RS	3C2	5E

	NOP	3C3	FF
	NOP	3C4	FF
	:	:	:
SALNN, Z:	MPC:=SD, Z(NN)	3D0	70
	PC:=PC+1	3D1	3C
	PC:=PC+1	3D2	3C
	MPC:=RS	3D3	5E
	NOP	3D4	FF
	:	:	:
SALPI, Z:	MPC:=SD, Z(PI)	3E0	72
	PC:=PC+1	3E1	3C
	PC:=PC+1	3E2	3C
	MPC:=RS	3E3	5E
	NOP	3E4	FF
	:	:	:
SALRN, NA:	MPC:=SD, NA(RN)	3F0	92
	PC:=PC+1	3F1	3C
	MPC:=RS	3F2	5E
	NOP	3F3	FF
	:	:	:
SALNN, NA:	MPC:=SD, NA(NN)	400	94
	PC:=PC+1	401	3C
	PC:=PC+1	402	3C
	MPC:=RS	403	5E
	NOP	404	FF
	:	:	:
SALPI, NA:	MPC:=SD, NA(PI)	410	96
	PC:=PC+1	411	3C
	PC:=PC+1	412	3C
	MPC:=RS	413	5E
	NOP	414	FF
	:	:	:
SALRN, A:	MPC:=SD, A(RN)	420	86
	PC:=PC+1	421	3C
	MPC:=RS	422	5E
	NOP	423	FF
	:	:	:
SALNN, A:	MPC:=SD, A(NN)	430	88
	PC:=PC+1	431	3C
	PC:=PC+1	432	3C
	MPC:=RS	433	5E
	NOP	434	FF
	:	:	:

SALPI, A:	MPC:=SD, A(PI)	440	8A
	PC:=PC+1	441	3C
	PC:=PC+1	442	3C
	MPC:=RS	443	5E
	NOP	444	FF
	:	:	:
SUBRUTN:	L:=(MAPC)	450	22
	H:=(MAPC)	451	24
	S:=PC	452	16
	PC:=HL	453	08
	MPC:=RS	454	5E
	NOP	455	FF
	:	:	:
SUBNN, NZ:	MPC:=SD, NZ(SN)	460	80
	PC:=PC+1	461	3C
	PC:=PC+1	462	3C
	MPC:=RS	463	5E
	NOP	464	FF
	:	:	:
SUBPI, NZ:	MPC:=SD, NZ(SP)	470	82
	PC:=PC+1	471	3C
	PC:=PC+1	472	3C
	MPC:=RS	473	5E
	NOP	474	FF
	:	:	:
SUBNN, Z:	MPC:=SD, Z(SN)	480	74
	PC:=PC+1	481	3C
	PC:=PC+1	482	3C
	MPC:=RS	483	5E
	NOP	484	FF
	:	:	:
SUBPI, Z:	MPC:=SD, Z(SP)	490	76
	PC:=PC+1	491	3C
	PC:=PC+1	492	3C
	MPC:=RS	493	5E
	NOP	494	FF
	:	:	:
SUBNN, NA:	MPC:=SD, NA(SN)	4A0	98
	PC:=PC+1	4A1	3C
	PC:=PC+1	4A2	3C
	MPC:=RS	4A3	5E
	NOP	4A4	FF
	:	:	:

SUBPI, NA:	MPC:=SD, NA(SP)	4B0	9A
	PC:=PC+1	4B1	3C
	PC:=PC+1	4B2	3C
	MPC:=RS	4B3	5E
	NOP	4B4	FF
:	:	:	:
SUBNN, A:	MPC:=SD, A(SN)	4C0	8C
	PC:=PC+1	4C1	3C
	PC:=PC+1	4C2	3C
	MPC:=RS	4C3	5E
	NOP	4C4	FF
:	:	:	:
SUBPI, A:	MPC:=SD, A(SP)	4D0	8E
	PC:=PC+1	4D1	3C
	PC:=PC+1	4D2	3C
	MPC:=RS	4D3	5E
	NOP	4D4	FF
:	:	:	:
RETOR:	PC:=S	4E0	1C
	MPC:=RS	4E1	5E
	NOP	4E2	FF
	:	:	:
RETR, NZ:	MPC:=SD, NZ(RT)	4F0	84
	MPC:=RS	4F1	5E
	NOP	4F2	FF
	:	:	:
RETR, Z:	MPC:=SD, Z(RT)	500	78
	MPC:=RS	501	5E
	NOP	502	FF
	:	:	:
RETR, NA:	MPC:=SD, NA(RT)	510	9C
	MPC:=RS	511	5E
	NOP	512	FF
	:	:	:
RETR, A:	MPC:=SD, A(RT)	520	90
	MPC:=RS	521	5E
	NOP	522	FF
	:	:	:
RETIN:	PC:=S	530	1C
	F:=S	531	1A
	A:=S	532	18
	MPC:=RS	533	5E

	NOP	534	FF
	:	:	:
NADA:	MPC:=RS	540	5E
	NOP	541	FF
	:	:	:
ALTO:	ALTO	558	02
	MPC:=RS	559	5E
	NOP	55A	FF
	:	:	:
AD:	L:=(MAPC)	600	22
	H:=(MAPC)	601	24
	S:=PC	602	16
	PC:=HL	603	08
	A:=(MAPC)	604	1E
	PC:=S	605	1C
	MPC:=DR	606	5C
	NOP	607	FF
	:	:	:
TD:	L:=(MAPC)	610	22
	H:=(MAPC)	611	24
	S:=PC	612	16
	PC:=HL	613	08
	T:=(MAPC)	614	20
	PC:=S	615	1C
	MPC:=DR	616	5C
	NOP	617	FF
	:	:	:
AI:	L:=(MAPC)	620	22
	H:=(MAPC)	621	24
	S:=PC	622	16
	PC:=HL	623	08
	L:=(MAPC)	624	22
	H:=(MAPC)	625	24
	PC:=HL	626	08
	A:=(MAPC)	627	1E
	PC:=S	628	1C
	MPC:=DR	629	5C
	NOP	62A	FF
	:	:	:
TI:	L:=(MAPC)	630	22
	H:=(MAPC)	631	24
	S:=PC	632	16
	PC:=HL	633	08

	L:=(MAPC)	634	22
	H:=(MAPC)	635	24
	PC:=HL	636	08
	T:=(MAPC)	637	20
	PC:=S	638	1C
	MPC:=DR	639	5C
	NOP	63A	FF
	:	:	:
DA:	L:=(MAPC)	640	22
	H:=(MAPC)	641	24
	S:=PC	642	16
	PC:=HL	643	08
	(MAPC):=A	644	28
	PC:=S	645	1C
	MPC:=DR	646	5C
	NOP	647	FF
	:	:	:
	:	:	:
IA:	L:=(MAPC)	650	22
	H:=(MAPC)	651	24
	S:=PC	652	16
	PC:=HL	653	08
	L:=(MAPC)	654	22
	H:=(MAPC)	655	24
	PC:=HL	656	08
	(MAPC):=A	657	28
	PC:=S	658	1C
	MPC:=DR	659	5C
	NOP	65A	FF
	:	:	:
	:	:	:
RT:	PC:=S	660	1C
	MPC:=RS	661	5E
	NOP	662	FF
	:	:	:
	:	:	:
NN:	L:=(MAPC)	670	22
	H:=(MAPC)	671	24
	PC:=HL	672	08
	MPC:=RS	673	5E
	NOP	674	FF
	:	:	:
	:	:	:
PI:	L:=(MAPC)	680	22
	H:=(MAPC)	681	24
	PC:=HL	682	08
	L:=(MAPC)	683	22

	H:=(MAPC)	684	24
	PC:=HL	685	08
	L:=(MAPC)	686	22
	H:=(MAPC)	687	24
	PC:=HL	688	08
	MPC:=RS	689	5E
	NOF	68A	FF
	:	:	:
SN:	L:=(MAPC)	690	22
	H:=(MAPC)	691	24
	S:=PC	692	16
	PC:=HL	693	08
	MPC:=RS	694	5E
	NOF	695	FF
	:	:	:
SP:	L:=(MAPC)	6A0	22
	H:=(MAPC)	6A1	24
	S:=PC	6A2	16
	PC:=HL	6A3	08
	L:=(MAPC)	6A4	22
	H:=(MAPC)	6A5	24
	PC:=HL	6A6	08
	L:=(MAPC)	6A7	22
	H:=(MAPC)	6A8	24
	PC:=HL	6A9	08
	MPC:=RS	6AA	5E
	NOF	6AB	FF
	:	:	:
RN:	T:=(MAPC)	6B0	20
	S:=F	6B1	14
	$F(a, z) := (0, 1)$	6B2	4A
	$F(a, z) := (\bar{a}, \bar{z})$	6B3	50
	A:=OOH	6B4	58
	A:=ACAD	6B5	32
	A:=AT	6B6	56
	A:=ACA7D	6B7	30
	A:=ACA7D	6B8	30
	A:=ACA7D	6B9	30
	A:=ACA7D	6BA	30
	A:=ACA7D	6BB	30
	A:=ACA7D	6BC	30
	A:=ACA7D	6BD	30
	S:=A	6BE	12

A:=OOH	6BF	58
A:=A+T	6CO	46
T:=PCL	6C1	0C
A:=A+T+a	6C2	48
L:=A	6C3	0E
A:=S	6C4	18
T:=PCH	6C5	0A
A:=A+T+a	6C6	48
H:=A	6C7	10
PC:=HL	6C8	08
F:=S	6C9	1A
MPC:=RS	6CA	5E
NOP	6CB	FF
:	:	:
:	:	:
NOP	FFF	FF

Circuito Integrado (U)	Tipo	Cantidad Usada	Descripción
3	74LS04	15/6	6 Inversores (NOT).
4	74LS08	3/4	4 Compuertas AND de dos <u>en</u> <u>tradas</u> .
6	74LS32	2/4	4 Compuertas OR de dos <u>en</u> <u>tradas</u> .
7	74LS74	4/2	2 Flip-flops tipo D.
9	74LS86	1/4	4 Compuertas EXOR de dos <u>en</u> <u>tradas</u> .
10	74LS125	4/4	4 Buffers de tercer estado de un bit.
11	74LS139	2/2	2 Demultiplexores de 1 a 4.
12	74LS153	1/2	2 Multiplexores de 4 a 1.
14	74LS193	9/1	1 Contador Universal de 4 bits.
16	74LS241	25/2	2 Buffers de tercer estado de 4 bits.
18	74LS273	1/1	8 Flip-flops tipo D.
20	MCM-2716	5/1	1 EPROM de 8 bits X 2 K.

TABLA IV.4.- Lista de dispositivos empleados en la construcción de la Sección de Control del procesador.

CAPITULO V.

MANUAL DE FUNCIONAMIENTO DEL PROCESADOR.

En los capítulos anteriores se hizo una descripción funcional de cada uno de los elementos que integran al procesador. El objetivo de este capítulo es dar una descripción del funcionamiento del procesador visto como un sistema ya completamente integrado.

V.1.- ARQUITECTURA DEL SISTEMA. En la FIGURA V.1 se puede observar la arquitectura del Sistema Procesador; esta consta de dos partes principales: La Sección de Proceso y la Sección de Control; además de los Buses de Direcciones, Datos y Control.

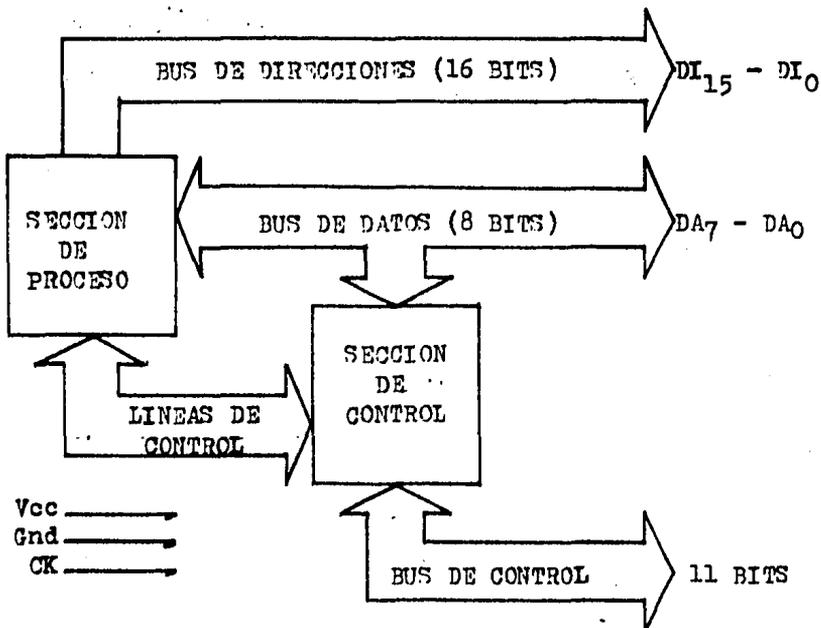


FIGURA V.1.- Arquitectura del Sistema Procesador.

V.1.A.- LA SECCION DE PROCESO. La Sección de Proceso es el lugar donde son realizadas las operaciones y cálculos indicados por las instrucciones del procesador; es aquí también donde son generadas las direcciones de donde han de ser tomados los códigos de operación y datos que requieren estas instrucciones para ser realizadas; para esto, la Sección de Proceso cuenta con una Unidad Lógico-Aritmética (ALU) de 8 bits y con una serie de registros de los cuales algunos no son accesibles al programador por medio del Repertorio de Instrucciones del procesador pero que son utilizados por la Sección de Proceso en su funcionamiento. Los registros que no son transparentes y que el programador podrá acceder por medio del Repertorio de Instrucciones del procesador son los siguientes:

Registro A.- Este es el Registro Acumulador del procesador; en el será almacenado un operando de 8 bits y el resultado de todas las operaciones realizadas por la ALU. Este registro podrá ser cargado con un dato de 8 bits proveniente de la memoria y también su contenido podrá ser transferido a la memoria.

Registro F.- Es el Registro de Banderas del procesador; en el serán almacenados los estados de las banderas de acarreo (a) y cero (z). El contenido del Registro F será afectado por las instrucciones lógicas, aritméticas y de corrimientos.

Registro S.- Es un "stack" o apilamiento del tipo LIFO (último en entrar primero en salir) con una profundidad de 256 palabras; en este registro podrá ser salvado (almacenado) el conte

nido de los Registros A y F, también en este Registro será salvada la dirección de regreso de subrutina.

Registro PC.- Es el Registro Contador de Programa. En este Registro es almacenada la dirección de memoria en donde se localiza la instrucción que habrá de ser ejecutada. Este es un Registro de 16 bits que no podrá ser accedido por el programador.

Además de los registros mencionados anteriormente, la Sección de Proceso cuenta con otros dos registros que, al igual que la ALU, no podrán ser accedidos por el programador.

BUS DE DIRECCIONES (DI₁₅ a DI₀). Salida de tercer estado de 16 bits mediante la cual el procesador proporciona la dirección de la localidad a partir de la cual podrán ser leídos o escritos los datos que necesita el procesador para realizar un programa. Con 16 líneas de direcciones el procesador podrá acceder hasta 65,536 localidades de memoria diferentes. El bit menos significativo del Bus de Direcciones es DI₀ y el mas significativo es DI₁₅; todos los bits del Bus de Direcciones son activos en un nivel de uno lógico.

BUS DE DATOS (DA₇ a DA₀). Entrada y salida de tercer estado de 8 bits activos en un nivel de uno lógico cuyo bit mas significativo es DA₇ y el menos significativo DA₀. Este bus es utilizado para realizar intercambio de datos entre el procesador y los dispositivos externos (dispositivos de entrada y salida, memoria, etc.).

V.L.B.- SECCION DE CONTROL. La Sección de Control es la encargada de tomar las decisiones e indicar a la Sección de Proceso las funciones que esta debe realizar; para esto, la Sección de Control cuenta con una serie de registros, lógi-

cas y almacenamientos que no son accesibles al programador.

BUS DE CONTROL. Este bus cuenta con 11 líneas de control que pueden ser organizadas en los siguientes tres grupos:

CONTROL INTERNO. Las cuatro líneas de este grupo sirven para que cualquier dispositivo externo ejerza algún control sobre el procesador. Las líneas de este grupo son $\overline{\text{RESET}}$, $\overline{\text{ESP}}$, y $\overline{\text{ALTO}}$.

CONTROL EXTERNO. Por medio de las cinco líneas de este grupo el procesador podrá ejercer su control a los dispositivos externos; las líneas de este grupo son $\overline{\text{BINS}}$, $\overline{\text{ACEX}}$, $\overline{\text{LEG}}$, $\overline{\text{ESC}}$, y $\overline{\text{RINT}}$.

CONTROL DE LOS BUSES. Por medio de las líneas de este grupo será posible controlar las líneas de tercer estado de los Buses de Direcciones, Datos y Control. Las líneas de este grupo son $\overline{\text{RBUS}}$ y $\overline{\text{CBUS}}$.

A continuación se da una lista y la explicación correspondiente de las funciones que realizan las líneas del Bus de Control.

$\overline{\text{RESET}}$. Entrada activa a un nivel de cero lógico. Cuando esta línea es activada el procesador es forzado a realizar un Ciclo de Búsqueda de Instrucción en la localidad de memoria 0000_H.

$\overline{\text{ESP}}$. Entrada activa a un nivel de cero lógico. La activación de esta línea indicará al procesador que -- los dispositivos externos (memoria, dispositivos de entrada y salida, etc.) que han sido seleccionados para realizar un intercambio de datos no están listos para realizarlo. El procesador esperará el tiempo que sea necesario a que los dispositivos externos se sincronicen con el y proseguirá su trabajo hasta que la línea $\overline{\text{ESP}}$ sea desactivada.

- INT. Entrada activa a un nivel de cero lógico. La activación de esta línea indicará al procesador que algún dispositivo externo requiere hacer una interrupción, este requerimiento será registrado al final de cada ciclo de máquina y el procesador contestará al requerimiento antes de que el código de operación de la siguiente instrucción sea buscado.
- ALTO. Salida de tercer estado activa a un nivel de cero lógico. Al activarse esta línea el procesador indicará a los dispositivos externos que está realizando una instrucción ALTO y está listo para recibir un requerimiento de interrupción o de bus.
- BINS. Salida de tercer estado activa a un nivel de cero lógico. Al activarse esta línea el procesador indicará que está realizando un ciclo de máquina que tiene como objetivo leer el código de operación de la instrucción que está por realizarse.
- ACEX. Salida de tercer estado activa a un nivel de cero lógico. Al activar esta línea el procesador indicará que está realizando un ciclo de máquina que tiene como objetivo acceder a los dispositivos externos; este acceso podrá ser de lectura o escritura.
- RINT. Salida de tercer estado activa a un nivel de cero lógico. Con la activación de esta línea el procesador indicará que está realizando un ciclo de máquina que tiene como objetivo leer la identificación de 8 bits que debe presentar el dispositivo externo que realizó un requerimiento de interrupción; esta identificación deberá ser presentada al Bus de Datos.
- LEC. Salida de tercer estado activa a un nivel de cero

lógico. Esta línea será activada por el procesador cuando se realice un acceso a los dispositivos externos que tenga como finalidad leer un dato de ellos.

ESC. Salida de tercer estado activa a un nivel de cero lógico. Esta línea será activada por el procesador cuando se realice un acceso a los dispositivos externos que tenga como finalidad escribir un dato en ellos.

RBUS. Entrada activa a un nivel de cero lógico. Al activar esta línea un dispositivo externo estará indicando al procesador que se requiere hacer uso de los buses. Este requerimiento será registrado al final de cada ciclo de máquina y el procesador contestará al requerimiento antes de que el código de operación de la siguiente instrucción sea buscado. En caso de que la activación de esta línea coincidiera con la activación de la línea INT (descrita anteriormente), el procesador solo registrará la activación de RBUS.

CBUS. Salida activa a un nivel de cero lógico. Con la activación de esta línea el procesador indicará que puede hacerse uso de los buses y además pondrá en tercer estado a sus Buses de Direcciones y Datos y a las líneas de salida del Bus de Control que sean de tercer estado.

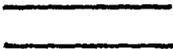
V.1.C.- LINEAS DE ALIMENTACION. El procesador requiere para su funcionamiento de tres señales de alimentación:

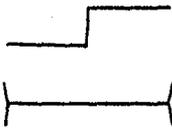
Vcc Vcc es una señal de corriente directa de 5 volts - de acuerdo a lo estandarizado en la familia de los dispositivos TTL.

Gnd Gnd es la señal o nivel de tierra (0 Volts).

Ck Ck es la señal de reloj del procesador; esta es -- también de acuerdo a los estándares de la familia TTL (5 Volts de amplitud típica y de forma de onda cuadrada) y su principal característica es que debe ser de una frecuencia baja de preferencia producida manualmente.

V.2.- CICLOS DE MAQUINA DEL PROCESADOR. Todas las operaciones que realiza el procesador son el resultado de una secuencia de ciclos de ejecución y ciclos de máquina. Durante los ciclos de ejecución, la Sección de Proceso del procesador ejecuta los cálculos y operaciones y durante este tiempo no es accesado ninguno de los dispositivos de entrada y salida o memoria. Por el contrario, el objetivo de los ciclos de máquina es establecer una comunicación con los dispositivos externos ya sea para leer o escribir datos o bien para contestar a alguno de los requerimientos que le son hechos al procesador. Típicamente un ciclo de máquina está compuesto por dos o más ciclos de reloj T. Las siguientes secciones -- tienen el propósito de ilustrar los diferentes ciclos de máquina que son realizados por el procesador; para esto se empleará la siguiente notación de las formas de onda de las señales presentes en las líneas de los buses del procesador.

FORMA DE ONDA	DESCRIPCION
	Denota a un estado irrelevante (cualquier estado no importa)
	Denota a un estado válido en estado estable.
	Denota un cambio de estado de un 1 a un 0 lógico.



Denota un cambio de estado de un 0 a un 1 lógico.

Denota un estado de alta impedancia (tercer estado).

V.2.A.- CICLO DE BUSQUEDA DE INSTRUCCION (BINS).

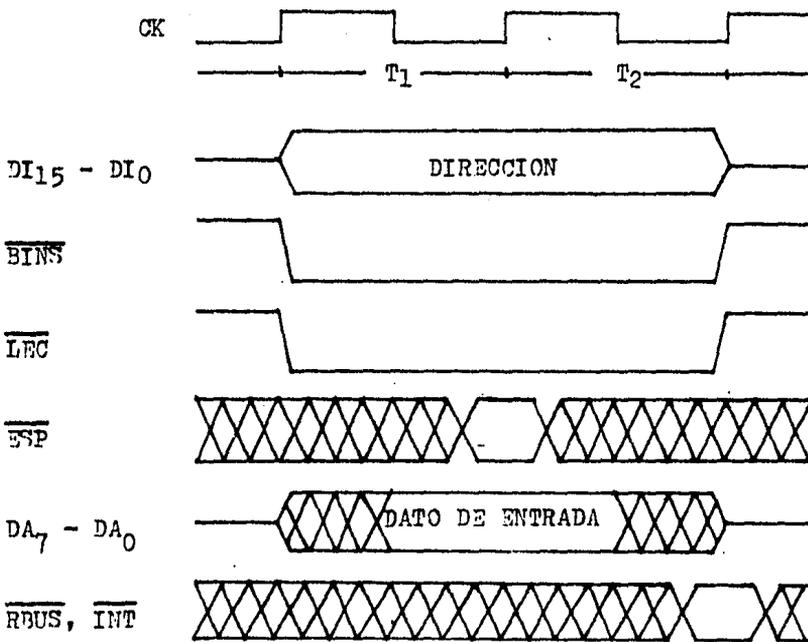


FIGURA V.2.A.1.- Diagrama de tiempo del Ciclo BINS.

El objetivo del Ciclo BINS es leer el código de operación de la instrucción en turno; para esto el procesador empleará típicamente dos ciclos de reloj (T_1 y T_2) como se muestra en la FIGURA V.2.A.1. Al principio del primer ciclo de reloj el Bus de Direcciones muestra la dirección de memoria a partir de la cual el código de operación deberá ser leído; al mismo tiempo el Bus de Datos es puesto en sentido de entrada por lo que el código puede ser puesto desde el principio del ciclo.

ciclo T_1 . También al principio del ciclo T_1 son activadas las líneas $\overline{\text{BINS}}$ y $\overline{\text{LEC}}$ del Bus de Control.

Al final del ciclo T_1 es censado el estado de la línea de entrada $\overline{\text{ESP}}$ del Bus de Control; si esta línea permanece inactiva en este momento, el procesador procederá a ejecutar el ciclo de reloj T_2 ; por el contrario, si la línea $\overline{\text{ESP}}$ hubiere permanecido activada en ese instante, el procesador procederá a ejecutar un ciclo de reloj de espera T_e durante el cual las condiciones establecidas durante el ciclo T_1 permanecen inalterables y al final del cual volverá a ser censado el estado de la línea de entrada $\overline{\text{ESP}}$ repitiéndose así los eventos ocurridos al final del ciclo de reloj T_1 .

Al principio del ciclo de reloj T_2 es realizada la lectura del dato.

Al final del segundo ciclo de reloj son desactivadas las líneas de $\overline{\text{BINS}}$ y $\overline{\text{LEC}}$ y los Buses de Direcciones y Datos son puestos en tercer estado; al mismo tiempo es censado y registrado el estado de las líneas $\overline{\text{INT}}$ y $\overline{\text{RBUS}}$ del Bus de Control.

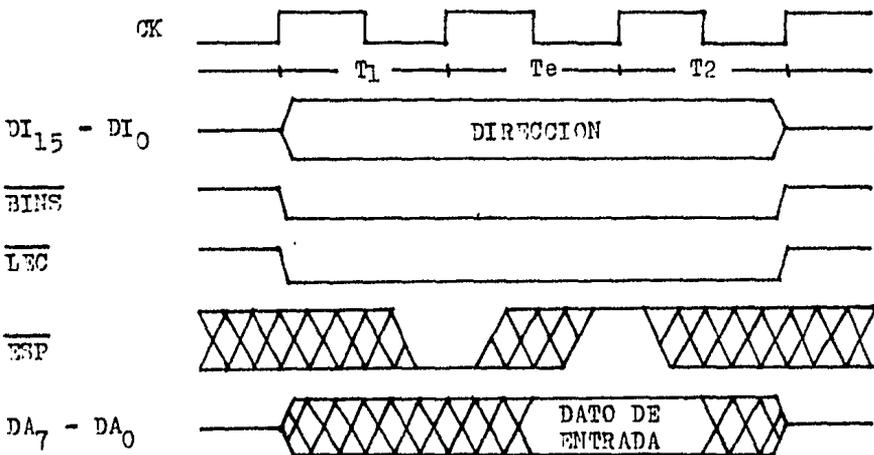


FIGURA V.2.A.2.- Diagrama de tiempo de un Ciclo BINS en el cual se ha solicitado un ciclo de espera (T_e).

V.2.B.- CICLO DE ACCESO PARA LECTURA (LEC).

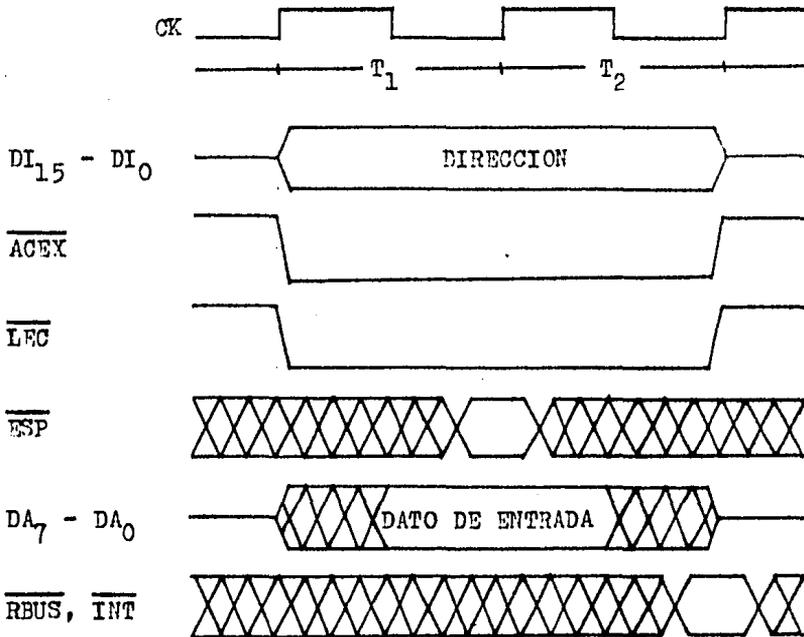


FIGURA V.2.B.1.- Diagrama de tiempo del Ciclo LEC.

El objetivo del ciclo LEC es leer un dato cualquiera proveniente de un dispositivo externo. Típicamente, este ciclo de máquina será realizado en dos ciclos de reloj (T_1 y T_2) y tiene la misma sincronía que el ciclo BINS solo que en este ciclo de lectura es activada la línea del Bus de Control \overline{ACEX} en vez de \overline{BINS} .

V.2.C.- CICLO DE ACCESO PARA ESCRITURA (ESC).

La finalidad del Ciclo de Acceso para Escritura es que el procesador escriba un dato en alguno de los dispositivos externos. Típicamente, el ciclo ESC será realizado en dos ciclos de reloj (T_1 y T_2) tal como lo muestra la FIGURA V.2C1 en la que es posible observar el momento en que serán activa

das las líneas que intervienen en este ciclo.

La línea $\overline{\text{ESP}}$ será censada al final del ciclo T_1 y antes de que el ciclo T_2 sea generado.

El estado de las líneas $\overline{\text{INT}}$ y $\overline{\text{RBUS}}$ será censado y registrado al final de T_2 .

En la FIGURA V.2.C.2 se puede observar un Ciclo ESC en el que ha sido solicitado un ciclo de espera T_e .

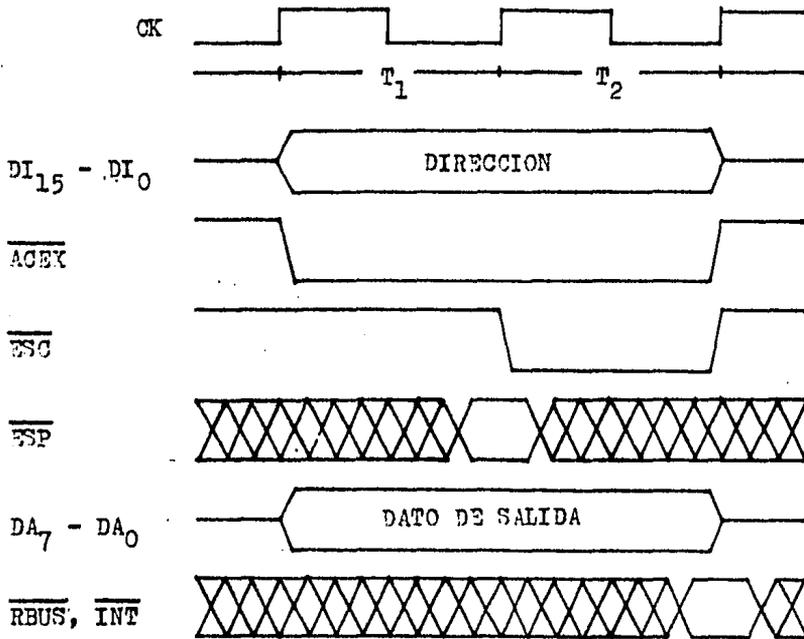


FIGURA V.2.C.1.- Diagrama de tiempo del ciclo ESC.

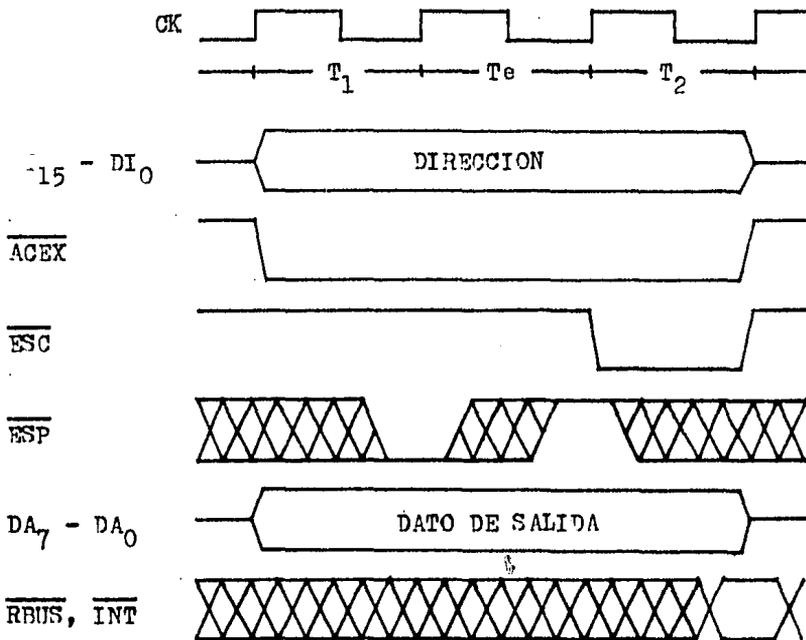


FIGURA V.2.C.2.- Diagrama de tiempo de un ciclo ESC en el cual se ha solicitado un ciclo de espera T_e .

V.2.D.- CICLO DE EJECUCION DE UNA INSTRUCCION ALTO (ALTO).

Al realizar este ciclo de máquina el procesador indicará que está ejecutando una instrucción ALTO en espera de un requerimiento de interrupción (\overline{INT}) o de bus (\overline{RBUS}). Este ciclo típicamente tendrá una duración de dos ciclos de reloj (T_1 y T_2) y el estado de las líneas \overline{INT} y \overline{RBUS} del Bus de Control será registrado al final del ciclo T_2 tal como lo muestra la FIGURA V.2.D.1.

Si al final de este ciclo de máquina es censado que no han sido activadas las líneas \overline{RBUS} o \overline{INT} , el procesador realizará un Ciclo de Búsqueda de Instrucción en el que será buscada la misma instrucción ALTO y posteriormente ejecutada.

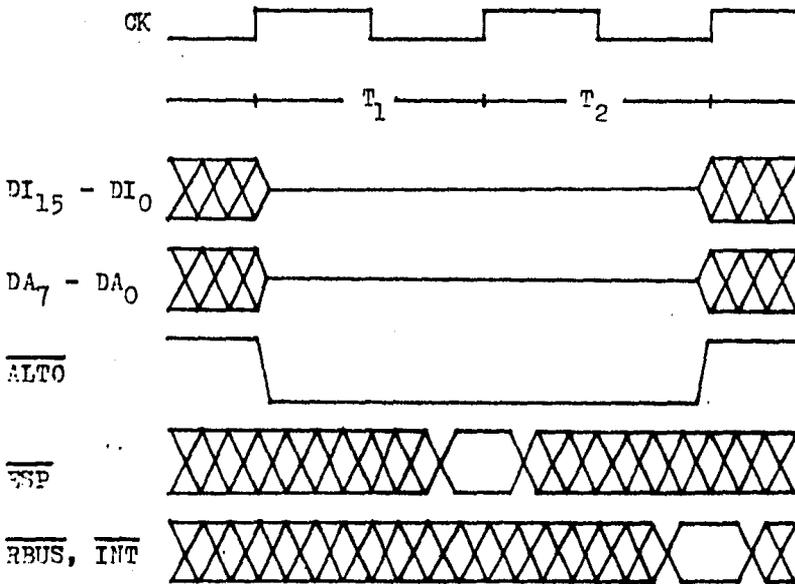


FIGURA V.2.D.1.- Diagrama de tiempo del ciclo ALTO.

V.2.E.- CICLO DE RECONOCIMIENTO DE INTERRUPCION (RINT).

Este ciclo es la respuesta que da el procesador a un requerimiento de interrupción; la forma en como serán activadas las líneas que intervienen en este ciclo se muestra en la FIGURA V.2.E.1.

El objetivo de este ciclo de máquina es leer la identificación de 8 bits que debe presentar en el Bus de Datos el dispositivo que hubiere realizado la interrupción. Los requerimientos de interrupción serán registrados por el procesador al final de el último ciclo de reloj de los ciclos de máquina BINS, LEC, ESC, ALTO y RINT y la respuesta al requerimiento (Ciclo RINT) será dada por el procesador antes de que sea realizado el siguiente Ciclo de Búsqueda de Instrucción.

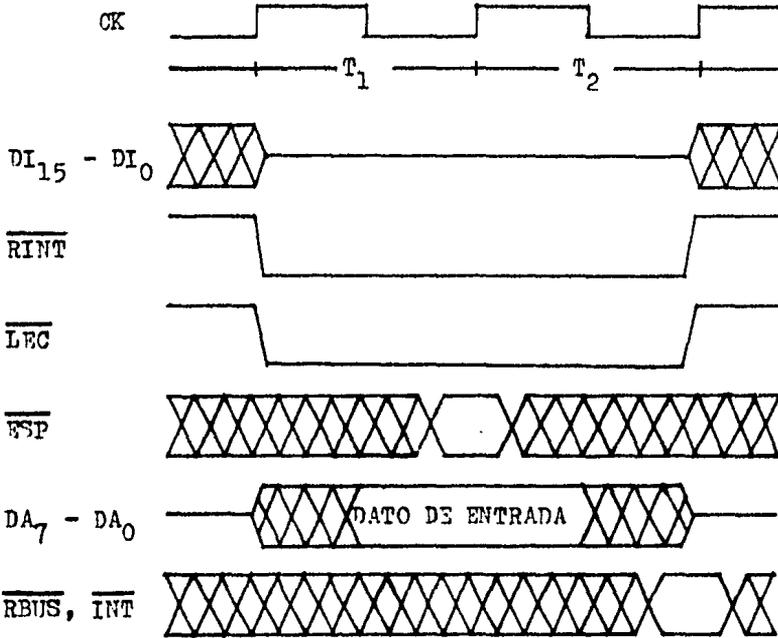


FIGURA V.2.E.1.- Diagrama de tiempo del Ciclo RINT.

La identificación de 8 bits será interpretada por el procesador como la dirección de inicio de la rutina que da servicio a esa interrupción; de esta forma, la identificación -corresponderá a los 8 bits menos significativos de la dirección de la rutina de servicio y el procesador asumirá que -- los 8 bits mas significativos son todos ceros; por lo consiguiente, la rutina de servicio podrá estar localizada en --- cualquiera de las 256 primeras localidades de memoria. Antes de realizar un salto a la rutina de servicio de la interrupción el procesador realizará la siguiente secuencia de operaciones;

- 1o. $S:=A$ En primer lugar salvará en el stack al contenido del Registro A.

- 20. S:=F Salvará al contenido del Registro F.
- 30. S:=PC Salvará al contenido del Registro PC.
- 40. A:=00H Limpiará el contenido del Acumulador.
- 50. Y finalmente realizará el salto a la rutina de servicio.

V.2.F.- CICLO DE CONCESION DEL BUS (CBUS).

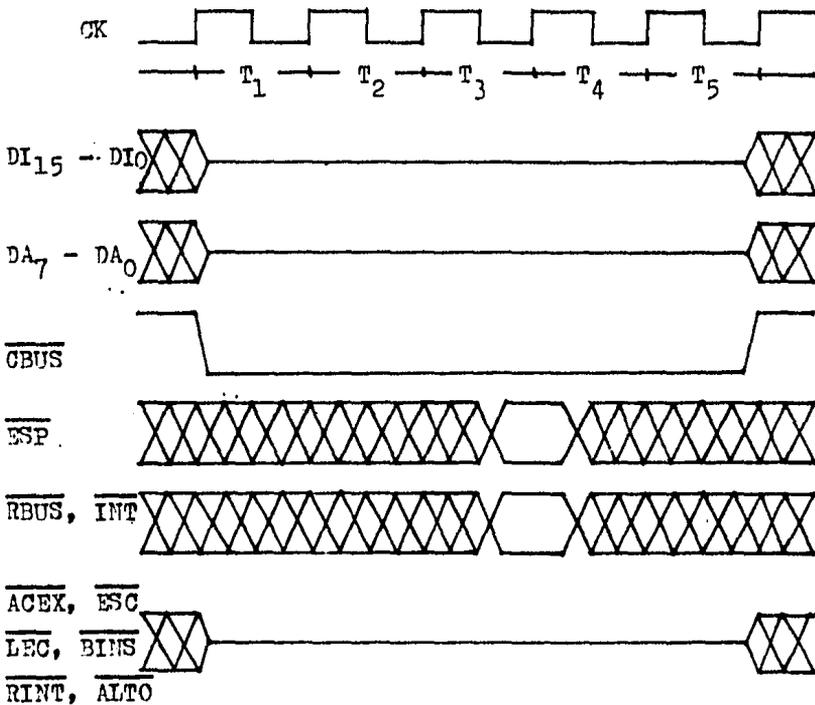


FIGURA V.2.F.1.- Diagrama de tiempo del Ciclo CBUS.

Este ciclo es la respuesta que da el procesador a un requerimiento de bus; la forma en como serán activadas las líneas que intervienen en este ciclo se muestra en la FIGURA - V.2.F.1.

Los requerimientos de bus serán registrados por el procesador al final de el último ciclo de reloj de los ciclos de máquina BINS, LEC, ESC, ALTO y RINT; los requerimientos de -

bus tendrán prioridad sobre los requerimientos de interrupción en caso de suceder en forma simultánea. La respuesta a un requerimiento de bus (Ciclo CBUS) será dada por el procesador antes de que sea realizado el siguiente Ciclo de Búsqueda de Instrucción y consistirá en poner en tercer estado los Buses de Direcciones, Datos y las salidas de tercer estado del Bus de Control.

V.3.- REPERTORIO DE INSTRUCCIONES. El procesador puede ejecutar 82 instrucciones que están agrupadas dentro de los siguientes seis grupos:

1. **Transferencia de operandos.** Las instrucciones de este grupo son las encargadas de realizar el intercambio de datos entre el Registro A y la memoria, entre el Registro A y el Registro S y entre los Registros S y F.
2. **Aritméticas.** Las instrucciones de este grupo realizan las operaciones de suma, resta, incremento y decremento sobre un dato de 8 bits contenido en el Registro A; el otro dato, también de 8 bits, que interviene en la suma o resta deberá provenir de la memoria externa. Las instrucciones aritméticas afectarán al contenido de las banderas de acarreo y cero de acuerdo al resultado de la operación indicada.
3. **Corrimientos.** Estas instrucciones realizan corrimientos, rotaciones y desplazamientos sobre un dato de 8 bits contenido en el Registro A. Las instrucciones de este grupo afectarán al contenido de las banderas de acarreo y cero.
4. **Lógicas.** Las operaciones realizadas por las instrucciones de este grupo son las funciones AND, OR, EX-

OR y Complementación de un dato de 8 bits contenido en el Registro A; el otro dato, también de 8 bits - que interviene en las funciones AND, OR y EXOR debe venir de la memoria. Las instrucciones de este grupo afectarán a las banderas de acarreo y cero.

5. **Modificación de programa.** Las instrucciones de este grupo realizan las funciones de salto incondicional, saltos Relativo, Inmediato y Programado condicionados a la bandera de acarreo o cero, salto a subrutina incondicional, saltos Inmediato y Programado a subrutina condicionados a la bandera de acarreo o cero, regreso incondicional de subrutina, regreso de subrutina condicionado a la bandera de acarreo o cero y regreso de interrupción.
6. **Función nula.** Las dos instrucciones de este grupo no realizan ninguna operación o modificación sobre los datos que maneja el procesador y pueden resultar buenas auxiliares en la depuración de programas y en los servicios que presta el procesador a los dispositivos externos.

V.3.A.- FORMATO Y MODO DE DIRECCIONAMIENTO DE LAS INSTRUCCIONES. Dentro del repertorio de instrucciones del procesador hay instrucciones de uno, dos y tres bytes; esto es debido principalmente a que cada instrucción necesita, además de su código de operación, indicar cual es la dirección del dato o el dato mismo sobre el cual realizará la operación. El código de operación de cada instrucción le indica al procesador que tipo de operación realizará esa instrucción, también le indica el número de bytes de la instrucción y el modo de direccionamiento que emplea la instrucción. Entonces, sea cual fuere el número de bytes de una instrucción (1, 2 o 3)

siempre el primero corresponderá al código de operación y -- los restantes podrán ser o un dato o la dirección de un dato cuyo byte menos significativo será el segundo de la instrucción y el mas significativo el tercero.

El procesador empleará cuatro modos de direccionamiento - en sus instrucciones; estos modos se refieren a la forma en la que el procesador obtiene los datos que son necesarios para realizar cada instrucción, los modos empleados son Direccionamiento Directo, Direccionamiento Indirecto, Direccionamiento Inmediato y Direccionamiento de Registros.

V.3.B.- LISTA DE INSTRUCCIONES. La lista que se da a continuación es una forma resumida del Capítulo II (en realidad todo este capítulo es un resumen de los anteriores) y para - obtener mayor información acerca de las instrucciones se deberá consultar, principalmente, ese capítulo.

Transferencia de operandos (Sección II.2):

NEMOTECNICO	CODIGO DE OPERACION (Hex)	NUMERO DE BYTES	AFECTA A LAS BANDERAS	
			(a)	(z)
POP A	08	1	n	n
CARN A,N	0A	2	n	n
CARD A,M	0C	3	n	n
CARI A,M	0E	3	n	n
PUSH A	10	1	n	n
DESD A,M	12	3	n	n
DESI A,M	14	3	n	n
POP F	16	1	n	n
PUSH F	18	1	s	s

Aritméticas (Sección II.3):

NEMOTECNICO	CODIGO DE OPERACION (Hex)	NUMERO DE BYTES	AFECTA A LAS BANDERAS	
			(a)	(z)
CPN A,N	1A	2	S	S
CPD A,M	1C	3	S	S
CPI A,M	1E	3	S	S
SUMN A,N	20	2	S	S
SUMD A,M	22	3	S	S
SUMI A,M	24	3	S	S
SUMAN A,N	26	2	S	S
SUMAD A,M	28	3	S	S
SUMAI A,M	2A	3	S	S
RESN A,N	2C	2	S	S
RESD A,M	2E	3	S	S
RESI A,M	30	3	S	S
RESAN A,N	32	2	S	S
RESAD A,M	34	3	S	S
RESAI A,M	36	3	S	S
INCRR A	38	1	S	S
INCRD M	3A	3	S	S
INCRI M	3C	3	S	S
DECRR A	3E	1	S	S
DECRD M	40	3	S	S
DECRI M	42	3	S	S

Corrimientos (Sección II.4)

RDER	44	1	S	S
RIZR	46	1	S	S
RDEAR	48	1	S	S
RIZAR	4A	1	S	S

NEMOTECNICO	CODIGO DE OPERACION (Hex)	NUMERO DE BYTES	AFECTA A LAS BANDERAS	
			(a)	(z)
CDER	4C	1	s	s
CIZR	4E	1	s	s
DDER	50	1	s	s
DIZR	52	1	s	s

Lógicas (Sección II.5):

ANDR A, A	54	1	\bar{a}	\bar{z}
ANDN A, N	56	2	0	s
ANDD A, M	58	3	0	s
ANDI A, M	5A	3	0	s
ORR A, A	5C	1	0	0
ORN A, N	5E	2	0	s
ORD A, M	60	3	0	s
ORI A, M	62	3	0	s
EXR A, A	64	1	0	1
EXN A, N	66	2	0	s
EXD A, M	68	3	0	s
EXI A, M	6A	3	0	s
NOTR A, A	6C	1	0	s
CDCS A, A	6E	1	s	s

Modificación de programa (Sección II.6):

SALTON M	70	3	n	n
SALRN NZ, \pm D	72	2	n	n
SALNN NZ, M	74	3	n	n
SALPI NZ, M	76	3	n	n
SALRN Z, \pm D	78	2	n	n

NEMOTECNICO	CODIGO DE OPERACION (Hex)	NUMERO DE BYTES	AFECTA A LAS BANDERAS	
			(n)	(z)
SALNI Z,M	7A	3	n	n
SALPI Z,M	7C	3	n	n
SALRN NA,±D	7E	2	n	n
SALNN NA,M	80	3	n	n
SALPI NA,M	82	3	n	n
SALRN A,±D	84	2	n	n
SALNN A,M	86	3	n	n
SALPI A,M	88	3	n	n
SUBRUTN M	8A	3	n	n
SUBNN NZ,M	8C	3	n	n
SUBPI NZ,M	8E	3	n	n
SUBNN Z,M	90	3	n	n
SUBPI Z,M	92	3	n	n
SUBNN NA,M	94	3	n	n
SUBPI NA,M	96	3	n	n
SUBNN A,M	98	3	n	n
SUBPI A,M	9A	3	n	n
RTROR	9C	1	n	n
RTRR NZ	9E	1	n	n
RTRR Z	A0	1	n	n
RTRR NA	A2	1	n	n
RTRR A	A4	1	n	n
RTRNR	A6	1	n	n

Función nula (Sección II.7):

NADA	A8	1	n	n
ALTO	AB	1	n	n

NOTA: En la columna que dice "AFECTA A LAS BANDERAS" se ha -

empleado la siguiente notación:

n.- No las afecta de ninguna forma.

s.- Las afecta de acuerdo al resultado de la operación.

Como ejemplo de la forma en que son afectadas las banderas de acuerdo al resultado de una operación considerense -- las siguientes operaciones:

0000 0000	
<u>+ 0000 0001</u>	
0000 0001	entonces a=0 y z=0
0000 0000	
<u>- 0000 0001</u>	
1111 1111	entonces a=1 y z=0
1111 1111	
<u>+ 0000 0001</u>	
0000 0000	entonces a=1 y z=0
0000 0001	
<u>- 0000 0001</u>	
0000 0000	entonces a=0 y z=1
1010 1010	
<u>OR 0101 0101</u>	
1111 1111	entonces a=0 y z=0
1010 1010	
<u>AND 0101 0101</u>	
0000 0000	entonces a=0 y z=1

En los corrimientos, rotaciones y desplazamientos la bandera de cero se pondrá en z=1 si el Acumulador queda con todos sus bits iguales a cero; la bandera de acarreo será modificada de acuerdo al estado del bit del Acumulador que sea desplazado hacia el bit de la bandera de acarreo.

En la última letra de la primera palabra del nemotécnico se ha indicado el modo de direccionamiento de la instrucción; las letras empleadas son las siguientes:

R.- Direccionamiento de Registros.

N.- Direccionamiento Inmediato.

D.- Direccionamiento Directo.

I.- Direccionamiento Indirecto.

CONCLUSIONES.

El diseño del procesador digital ha resultado una tarea - bastante laboriosa que comprende no solo al diseño mismo de los bloques arquitectónicos y de las microrutinas y microoperaciones que los hacen funcionar, de hecho, la labor comenzó antes de pensar siquiera en los bloques arquitectónicos que constituirían al procesador; en un principio, la tarea consistió en realizar un trabajo de investigación que permitiera adquirir un panorama de la filosofía de funcionamiento y estructuración de algunos microprocesadores, gracias a esto fué posible delimitar los objetivos de diseño y plantear una arquitectura tentativa del procesador que parecía ser muy modesta. Una vez planteada la arquitectura la tarea consistió en investigar el tipo y funciones de los dispositivos que se emplearían en su implementación, aunque ya se había pensado desde un principio en los de la familia TTL, fué necesario - investigar cuales de los dispositivos elegidos existían en - el mercado nacional y posteriormente cuales estaban disponibles en el laboratorio de la escuela, afortunadamente todos los dispositivos mencionados en este trabajo son posibles de encontrar ahí.

Tal vez la parte mas didáctica de este trabajo es la correspondiente a la arquitectura del procesador ya que en ella se da una descripción clara de los bloques arquitectónicos y se incluye además el diagrama lógico y la tabla de verdad de cada bloque; además, es esta parte la correspondiente a la Sección de Proceso del procesador que resulta ser la - mas interesante y la que mejor debe conocer cualquier persona interesada en el aspecto de programación y funcionamiento del procesador desde el punto de vista del usuario.

Al terminar de diseñar los bloques arquitectónicos y las

tablas de verdad de la Sección de Proceso parecía ser que el trabajo estaba por terminar, pero aún faltaba la Sección de Control.

La Sección de Control de cualquier procesador es su parte mas complicada. Si bien la Sección de Proceso es accesible a cualquier persona con conocimientos generales de circuitos lógicos; la Sección de Control requiere de conocimientos mas especializados y no es recomendable su estudio para una persona que se interese en el procesador solo desde el punto de vista usuario; en forma práctica basta con saber que la Sección de Control cuenta con un Registro de Instrucción y que es la encargada de dar las ordenes que activan a todos los bloques arquitectónicos del procesador y al Bus de Control.

Si se compararan los bloques arquitectónicos que componen a las Secciones de Proceso y Control de este procesador en particular, se podría aseverar que los de la Sección de Proceso son mas complicados que los de la Sección de Control; entonces ¿ De donde parte lo complicado de la Sección de Control ? Es obvio que su complejidad reside principalmente en su funcionamiento; en la comparativamente gran cantidad de funciones que realiza y que no son posibles de observar en su estructura sino que están almacenadas en solo cinco circuitos integrados que son todos los que componen a la Prom de Correspondencia y a la Prom de Control. Para tener una idea de esto será suficiente mencionar que, por ejemplo, la Prom de Control ejecutará 76 microoperaciones distintas las cuales estarán compuestas por un promedio de cinco ordenes de control cada una y que en la Prom de Correspondencia se ejecutarán alrededor de 93 microrutinas, cada una formada por cuatro microoperaciones en promedio. En esto es principalmente en lo que reside la complejidad de la Sección de --

control y en consecuencia es esto mismo lo que ocasionó que que una tarea que parecía ser modesta se volviera complicada.

Ha resultado ser un gran mérito el elegir un sistema o método de trabajo sugerido por algunas personas que poseen una mayor experiencia en el diseño de circuitos lógicos, esto ha permitido que una tarea de proporciones razonablemente grandes pudiera ser fraccionada y abordada de una manera más simple, aunque cabe señalar que, debido principalmente a razones económicas, no ha sido posible el integrar y poner en funcionamiento conjuntamente a todos los bloques arquitectónicos del procesador. Sin embargo; se ha probado el funcionamiento de cada bloque armandolo en tabletas para prototipos (Protoboard) y comprobando su tabla de verdad; en el caso de la Prom de Control y la Prom de Correspondencia se ha revisado que los estados lógicos que han de ser almacenados en ellas produzcan los resultados esperados en los diferentes bloques a los que han de poner en funcionamiento.

Es un hecho que este procesador resulte muy caro y poco eficiente si es usado con fines diferentes a los meramente didácticos que son los fines para los que ha sido diseñado y en cuyo caso el precio y la eficiencia como máquina son disculpables; también es un hecho que la arquitectura del procesador es muy simple y que los dispositivos empleados en el diseño también son muy simples y que seguramente estas características son muy convenientes en un sistema con propósitos didácticos, entonces es recomendable que al armar el procesador se haga de una manera simple; lo más recomendable es que cada bloque arquitectónico de la Sección de Proceso del procesador fuera armado en una sola tableta de circuito impreso, de forma tal que pudiera enzambarse o desenzambarse del resto del procesador; esto permitiría analizar el funciona--

miento individual de cada bloque arquitectónico y comprobar su tabla de verdad, también permitiría observar el funcionamiento de todos los bloques arquitectónicos en conjunto realizando una labor en común, por ejemplo ejecutando algunas instrucciones de las del repertorio del procesador. En lo tocante a la Sección de Control, debido a que pueden resultar de poco interés los detalles de su funcionamiento además de innecesarios, lo más recomendable sería armarla en una sola pieza; es decir, considerarla como otro bloque arquitectónico más y dejar los detalles de su funcionamiento solo para las personas que tengan algún interés particular en ellos.

En el diseño de este procesador se han tratado de incluir algunas de las características más representativas que pueden ser encontradas en los diferentes microprocesadores que existen en el mercado; esto puede ser observado tanto en el tipo de registros que este procesador posee como en las operaciones e instrucciones que es capaz de ejecutar; esto también puede resultar una ayuda complementaria a las características didácticas que el diseño de este procesador pudiera tener.

Personalmente, el diseño de este procesador ha resultado ser una experiencia muy grande y ha permitido adquirir una visión más real y completa de lo que son los microprocesadores, de lo que pueden hacer y de como lo hacen; además ha dado la oportunidad de tener una relación más estrecha con el diseño de circuitos lógicos y con algunos de los dispositivos TTL más usuales.

Finalmente, por medio de estas líneas quisiera agradecer la valiosa colaboración de mi asesor el Maestro en Ciencias Juan Antonio Navarro Martínez y de los ingenieros José Luis Rivera López y Luis García Gutiérrez que de una u otra mane-

ra han hecho posible la realización de este trabajo.

B i b l i o g r a f í a.

Bit-Slice Microprocessor Design.

John Mick & James Brick

McGraw Hill, 1980

Circuitos Digitales y Microprocesadores.

Herbert Taub

McGraw Hill, 1983

Computer Architecture and Organization.

John P. Hayes

McGraw Hill, 1979

Motorola Memory Data Manual.

Motorola Inc., 1979

The TTL Data Book for Design Engineers.

Texas Instruments Inc., 1976

A P E N D I C E.

Costos.

Resulta muy difícil hacer una evaluación económica confiable del costo que pudiera tener la construcción del Procesador Digital Didáctico ya que debido a muchos factores, el -- precio de los componentes varía constantemente; sin embargo, es importante que en todo proyecto sean considerados los factores económicos.

Los precios de los componentes que se incluyen en la TABLA VI.1 han sido tomados del catálogo publicado en febrero de 1985 de una negociación importante en el ramo nacional de la electrónica. Estos precios son los aplicados en la venta al menudeo y no incluyen el impuesto al valor agregado.

Además de los costos que pudieran tener los componentes -- es muy importante considerar el montaje de estos; esto resulta aún mas difícil debido a que el procesador digital no fué completamente armado y el diseño de los circuitos impresos -- para su montaje puede resultar una tarea mucho muy laboriosa.

Un fabricante de circuitos impresos requiere del diseño del circuito impreso para poder hacer un presupuesto ya que, según el, el precio del circuito impreso depende de factores como son: el número de perforaciones, la densidad de componentes, las dimensiones y "otras cosas". Sin embargo este -- mismo fabricante indica que en promedio; una tarjeta de circuito impreso tiene un costo aproximado de treinta pesos por centimetro cuadrado y que en una tarjeta de 25 X 12.5 centímetros caben "comodamente" alrededor de 25 componentes (circuitos integrados). En base a estas consideraciones y observando un total de 84 circuitos integrados en la TABLA VI.1 -- es posible suponer que serían necesarias cuatro tarjetas de 25 X 12.5 cm. para el montaje del Procesador Digital Didácti

co.

Esto tendría un costo aproximado de
4 tabletas de 25 X 12.5 cm² y a \$30.00 el cm² = \$ 37,500
que aunado al costo de los componentes

$$37,500 + 41,654 = \$ 79,154$$

da un total de 79,154 pesos; esto sin contar el costo que pu
diera tener el armado del procesador.

Circuito Integrado	Cantidad Usada Total	Total de Unidades	Precio Unitario	Precio Total
74LS00	1/4	1	141.00	141.00
74LS02	1/4	1	141.00	141.00
74LS04	37/6	7	141.00	987.00
74LS08	11/4	3	141.00	423.00
74LS25	2/2	1	141.00	141.00
74LS32	7/4	2	141.00	282.00
74LS74	5/2	3	180.00	540.00
74LS76	2/2	1	188.00	188.00
74LS86	4/4	1	180.00	180.00
74LS125	6/4	2	188.00	376.00
74LS139	2/2	1	258.00	258.00
74LS153	3/2	4	211.00	844.00
74LS181	2/1	2	936.00	1872.00
74LS193	15/1	15	336.00	5040.00
74LS194	2/1	2	281.00	562.00
74LS241	41/2	21	507.00	10,647.00
74LS243	4/1	4	468.00	1872.00
74LS273	4/1	4	507.00	2028.00
MCM-2114	4/1	4	858.00	3432.00
MCM-2716	5/1	5	2340.00	11,700.00
Suma Total:		84		41,654.00

TABLA VI.1.- Costo de los circuitos integrados usados en el
diseño del procesador.