

35

28j

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

"CAMPUS ARAGON"



FALLA DE ORIGEN

"MICROPROCESADORES DE 32 BITS Y
SU APLICACION AL CONTROL DIGITAL"

T E S I S

Que para obtener el Título de:
INGENIERO MECANICO ELECTRICISTA

P r e s e n t a :
ENRIQUE HERRERA HIDALGO

San Juan de Aragón Agosto de 1995



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.

TESIS CON FALLA DE ORIGEN

Dedicatoria

En memoria de:

Alejandrina Silvia y María del Pilar.

Mis padres:

Don Erasto Herrera M. y Doña Enriqueta Hidalgo M.
porque me han dado lo más precioso que tengo:
La vida y el entusiasmo de continuar
adelante.

Mis hermanos:

Ma. Isabel, Ma. Eulalia, Catalina, Claudio, Miguel
Angel, Hugo Erasto y Ma. Esmeralda, para que este
trabajo pueda servir de ejemplo a futuro.

FALLA DE ORIGEN EN SU TOTALIDAD

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

Desde la invención del circuito integrado CI, desarrollos continuos han dado lugar a dispositivos cada vez más complejos. Procesadores de computador, memorias, interfaces normalizadas e incluso sistemas de computador completos, están disponibles como circuitos integrados individuales. En consecuencia, se dispone de sistemas de computador muy pequeños y económicos y pueden incorporarse en muchos sistemas electrónicos. Puesto que hay muchas ventajas en este método, la instrumentación electrónica, los dispositivos de control y los sistemas de comunicación están haciendo uso del μ Computador. Una comprensión del funcionamiento y aplicación del μ Computador (μ C), es tan importante para un técnico actual, que cuando sólo se requiera un curso en sistemas digitales, debe incluirse en el mismo el tratamiento de los sistemas basados en el μ C.

En la actualidad los μ Procesadores (μ P), se emplean en una amplísima variedad de aplicaciones; se los encuentra en computadoras, juguetes electrónicos, procesadores de palabras, equipo electrónico de laboratorio, aparatos electrodomésticos, equipo médico, máquinas herramienta, robots industriales, alarmas, controles y automatismos de todo tipo, cajeros automáticos, básculas electrónicas, sintetizadores musicales, subsistemas de automoviles, etc.

El μ P es parte integral de la circuitería interna de una gran variedad de equipo, para dar una idea de su difusión e importancia basta decir que anualmente, se producen más de 25 millones de μ P y su mercado continúa en expansión.

Los circuitos integrados, han simplificado el diseño de los complejos circuitos analógicos y digitales. En la década pasada, numerosos fabricantes produjeron una extraordinaria cantidad de ellos. El Ingeniero o técnico, cuando afrontan la tarea de seleccionar los circuitos integrados y su diseño, deben consultar un gran número de catálogos de los fabricantes y un reducido número de notas de aplicaciones, a fin de ensayar y determinar la configuración óptima de los circuitos integrados y del circuito que se requiere.

1
FALLA DE ORIGEN

Las hojas de datos de los catálogos sirven para definir los parámetros de operación y del peor caso de un dispositivo en particular, pero no pueden utilizarse como una guía de selección puesto que los circuitos integrados no se evalúan a partir de comparaciones. Por lo demás, estos catálogos y notas de aplicación se limitan a los circuitos integrados de un fabricante y están organizados según el tipo de circuito integrado, no según la aplicación.

JUSTIFICACION

La industria mexicana, cada vez más consciente de la importancia del uso de las computadoras, los μP , los sistemas de control y de la Electrónica en general; cada día exige más preparación de los Ingenieros y por ende mayor calidad en las tareas realizadas, para lo cual establece criterios y parámetros que les permite analizar perfiles, y decidir cuál o cuáles personas son aptas para realizar con éxito planes y proyectos que eleven la imagen tanto de la Empresa como del País.

La razón de escoger como tema de tesis a los μP de 32 Bits y su Aplicación al Control Digital; radica en la importancia que estos dispositivos tienen en la actualidad, dentro de los sistemas automáticos más relevantes, dentro de la industria.

Es importante recalcar que en la medida que se conozcan estos dispositivos a detalle, se podrá estudiar, entender, modificar y optimizar a los sistemas construidos a partir de éstos.

Cada vez resulta imperioso, que los profesionales de la Ingeniería Electrónica, conozcan en gran medida los dispositivos y circuitos integrados, que se utilizan dentro de los procesos industriales, ya que con estos conocimientos se tendrá la oportunidad de conocer los sistemas y poderlos modificar y mejorar utilizando tecnología y mano de obra mexicana, sin necesidad de seguir siendo dependientes de la tecnología extranjera.

Por lo tanto, para que esto deje de ser una simple utopía y se convierta en realidad; los actuales Ingenieros del área Electrónica deberán conocer la Arquitectura, Programación y Aplicación de cada uno de los μP que interactúan dentro de un sistema digital.

ANTECEDENTES AL TRABAJO

Las computadoras han sido y serán una herramienta de trabajo imprescindible a nivel mundial. Aunque en 1975, ya se podían encontrar computadoras en casi todas las organizaciones medianas y grandes todavía eran demasiado costosas para grupos pequeños o en forma particular. No obstante, los avances en la Electrónica estaban a punto de introducir una categoría totalmente nueva de computadoras; máquinas de inventario con la capacidad de los antiguos gigantes y con precios al alcance de los individuos.

Actualmente, la gente está consciente de las ventajas que brindan las computadoras. Por otro lado, es impresionante el paso agigantado que la Electrónica va avanzando, por lo cual es casi imposible mantenerse al nivel de desarrollo tecnológico de países tan importantes como China, Japón, EUA, Alemania, etc., pero si podemos aprovechar la información que poseemos y actuar con ella en consecuencia y ya no delegar como por desgracia, hasta ahora se ha hecho.

En todo momento de la historia de las computadoras; se han tomado en cuenta puntos importantes como son: El saber y conocer las posibilidades y limitaciones de las computadoras (en cuanto al software, hardware y firmware), saber el manejo de las mismas, conocer su estructura interna, conocer la arquitectura básica de los elementos que la configuran y finalmente apreciar la gran utilidad y facilidad que brindan en el manejo de la información.

PLAN PROPUESTO

Antes de entrar de lleno al desarrollo del tema, es conveniente proponer un plan, en el que se exponga la metodología para llevar a cabo los objetivos buscados en la tesis, así como la utilidad de la misma.

En primera instancia, se requiere tener conocimientos sólidos de Electrónica Analógica y Digital, de Sistemas de Control tanto Analógico como Digital; como también haber leído y tener contacto físico con Circuitos Integrados, Memorias, Periféricos y con μP 's elementales. Una vez confirmado lo anterior, se recomienda la siguiente metodología:

1.- El trabajo está orientado en el Capítulo I, a estudiar los conceptos básicos de los μP de 32 Bits; es decir la estructura de este capítulo es definir correctamente todos los conceptos utilizados dentro del lenguaje propio de los sistemas basados en μP 's tales como Unidad Lógica-Aritmética, Memoria de Solo Lectura, Memoria de Acceso Aleatorio, Dispositivos de Entrada/Salida, Buses de Comunicaciones, Interfases de Comunicaciones y Protocolos de Comunicaciones.

2.- El Capítulo II, analiza detalladamente, la Arquitectura del μP 80386; siendo muy específico en cuanto al conexionado, la arquitectura interna y el manejo del repertorio de instrucciones propias de este μP . En este capítulo se desea establecer toda la información inherente al μP y conocerlo sólidamente.

3.- El Capítulo III, analiza también a detalle, la Arquitectura del μP 68406; en este capítulo, se realiza el estudio del μP exactamente igual que para el μP 80386.

4.- El Capítulo IV, analiza SOLO algunas de las aplicaciones que el Control Digital permite dentro de los sistemas digitales, y éstas son básicamente: La Inteligencia Artificial (AI) y los Sistemas Expertos (SE). Los Automatas Programables (AP) y la Robótica.

Todo lo anterior con la finalidad de dar información adicional para la aplicación de los Sistemas de Control Digital basados en los μP 's.

OBJETIVOS

OBJETIVO GENERAL:

Conocer y estudiar detalladamente, las características de la Arquitectura, Programación y Aplicación de los μP de 32 Bits, al igual que su Aplicación a Sistemas de Control Digital.

OBJETIVOS PARTICULARES:

1.- Conocer los fundamentos de los μP de 32 Bits, así como, las definiciones básicas que se involucran en los sistemas digitales basados en μP 's.

2.- Conocer detalladamente la Arquitectura interna del μP 80386, así como su conexionado y el repertorio de instrucciones para su programación.

3.- Conocer detalladamente la Arquitectura interna del μP 80486, así como su conexionado y el repertorio de instrucciones para su programación.

4.- Establecer algunas de las aplicaciones básicas del Control Digital, orientadas al uso de los sistemas digitales.

C A P I T U L O I

INTRODUCCION A LOS MICROPROCESADORES DE 32 BITS.

I.1.- MICROPROCESADOR (μ P).

Es difícil dar una definición exacta de lo que es un microprocesador, sobre todo porque en los últimos años, el desarrollo de la electrónica en esta área, ha dado lugar a la aparición de un sinnúmero de estos dispositivos, diseñados con enfoques hacia diversos campos de aplicación.

Sin embargo, todos los μ P's tienen ciertos atributos comunes que los distinguen de otros elementos electrónicos.

Las características principales de un μ P son su "universalidad" y su "programabilidad", que hacen de el un dispositivo tan versátil y poderoso, que puede utilizarse como elemento inteligente ó "cerebro", en aplicaciones que van desde la computadora personal hasta un rastreador de satélites ó un aparato detector de epilepsia.

Esencialmente, un μ P es un circuito de alta escala de integración (LSI), compuesto de muchos circuitos más simples como son flip-flop's, contadores, registros, decodificadores, comparadores, etc., todos ellos dentro de la misma pastilla de silicio, de modo que el μ P puede ser considerado un dispositivo lógico de propósito general ó universal.

La "programabilidad", se refiere a la capacidad que tiene un μ P para que su función sea definida a través de un programa. El "programa" consta de una serie de órdenes ó "instrucciones" relacionadas, ejecutadas secuencialmente (una a la vez) por el μ P y que pueden implicar operaciones lógicas ó aritméticas. Las instrucciones se especifican por medio de un código especial que constituye el lenguaje del μ P.

De los párrafos anteriores, se puede deducir la existencia de dos generos diferentes de elementos que se complementan y que juntos delimitan el concepto del μ P como un dispositivo útil. El primero de ellos lo forman el circuito mismo, sus componentes electrónicos, sus alambres de interconexión y, en general, todo aquello determinado por la configuración física del circuito integrado CI. A este genero se le conoce como el Hardware ó soporte físico del μ P, ya que por su misma naturaleza no admite modificaciones.

Por otro lado, se encuentran el código de instrucciones, los algoritmos y los programas que determinan el comportamiento del μP , los cuales pueden ser alterados cuantas veces sea necesario, con el fin de adaptar y optimizar el desempeño del dispositivo a algún campo de aplicación particular. Este segundo género recibe el nombre de software ó soporte lógico.

1.2.- UNIDAD LOGICA-ARITMETICA (ALU).

Una ALU es una función multi-operacional digital de lógica combinatorial. Esta puede realizar un conjunto de operaciones aritméticas básicas y un conjunto de operaciones lógicas. La ALU tiene un número de líneas de selección para habilitar una operación particular de la unidad.

Las líneas de selección se decodifican dentro del ALU de manera que las k variables de selección pueden especificar hasta 2 a la k operaciones diferentes.

Todos los μP contienen una ALU que con frecuencia se conoce simplemente como ALU. Esta es la parte del μP que lleva a cabo las operaciones aritméticas y lógicas en los datos binarios. Algunas de ellas se aplican sobre dos operandos, mientras que otras requieren solamente uno. La ALU generalmente es capaz de ejecutar las siguientes operaciones:

- 1.- Suma aritmética.
- 2.- Funciones lógicas AND, OR y XOR.
- 3.- Complemento.
- 4.- Rotación hacia la derecha o izquierda.

Además, la ALU contiene un conjunto de flip-flop's llamados "banderas", los cuales guardan información relacionada con el resultado de una operación aritmética o lógica. Por ejemplo, una de las banderas sirve para indicar si el resultado de la última operación fue cero.

1.3.- MEMORIA DE SOLO LECTURA (ROM).

Una ROM es esencialmente un dispositivo (o acumulador) de memoria en el cual se almacena un conjunto fijo de información binaria. La información binaria puede especificarse por el usuario y luego enclavarse en la unidad para formar el patrón de interconexión requerido. Las ROM vienen con enlaces internos especiales que pueden estar fusionados o abiertos. La interconexión deseada para una aplicación particular requiere que ciertos enlaces estén fusionados para formar los caminos necesarios del circuito.

Una vez que se establezca un patrón para una ROM, éste permanecerá fijo aunque se haga un corte de corriente y luego se restablezca.

Una ROM consiste de n líneas de entrada y m líneas de salida. Cada combinación de bits de las variables de entrada se llama una dirección y cada combinación de bits que sale por las líneas de salida se llama palabra. El número de bits por palabra es igual al número de líneas de salida m . Una dirección es esencialmente un número binario que denota uno de los términos mínimos de n variables. El número de direcciones diferentes posibles con n variables de entrada es 2 a la n . Una palabra de salida puede ser seleccionada por una dirección única y como hay 2 a la n direcciones diferentes en una ROM, hay 2 a la n palabras diferentes que se dice que están acumuladas en la unidad. La palabra disponible en las líneas de salida, en cualquier momento dado, depende del valor de la dirección aplicada a las líneas de entrada. Una ROM se caracteriza por el número de palabras 2 a la n y el número de bits por palabra m .

Considere una ROM de 32×8 . La unidad consiste en 32 palabras de 8 bits cada una. Esto significa que hay 3 líneas de salida y 32 palabras distintas almacenadas en la unidad, cada una de las cuales puede aplicarse a las líneas de salida. La palabra particular seleccionada que está presente en las líneas de salida se determina a partir de las 5 líneas de entrada. Hay solamente 5 entradas en una ROM de 32×8 , porque 2 a la 5 es igual a 32 ; y con 5 variables se puede especificar 32 direcciones o términos mínimos. Para cada dirección de entrada hay una palabra única seleccionada.

El número de palabras direccionadas en una ROM se determina del hecho de que se necesitan n líneas de entrada para especificar 2 a la n palabras. Una ROM se especifica algunas veces por el número total de bits que contiene, el cual será 2 a la n por m .

Internamente, la ROM es un circuito combinacional con compuertas AND conectadas como decodificador, y un número de compuertas OR igual al número de salidas de la unidad. La ROM es una configuración de dos niveles en forma de suma de términos mínimos. No tiene que ser una configuración AND-OR, pero puede ser cualquier otra posible configuración de términos mínimos de dos niveles. Las ROM tienen muchas aplicaciones importantes en el diseño de sistemas de computadores digitales. Su uso para la configuración de circuitos combinacionales complejos es justamente una de sus aplicaciones.

Los caminos necesarios en una ROM pueden ser programados de dos maneras diferentes. La primera se llama programación por máscara y la hace el fabricante durante el último proceso de fabricación de la unidad. El procedimiento para fabricar una ROM requiere que el cliente llene la tabla de verdad según lo que desee que haga la ROM. La tabla de verdad debe ser entregada en una forma especial suministrada por el fabricante. Se entrega a menudo en cinta de papel o tarjetas perforadas en el formato especificado en la hoja de datos de una ROM particular. El fabricante hace la máscara correspondiente para que los caminos produzcan "1" y "0" lógicos de acuerdo a la tabla de verdad del cliente. Este procedimiento es muy costoso ya que el vendedor, le carga al cliente una tarifa especial por hacerle una ROM con máscara.

Por esta razón, la programación con máscara es económica solamente si se van a fabricar grandes cantidades del mismo tipo de configuración de ROM.

Para pequeñas cantidades, es más económico usar un segundo tipo de ROM llamado memoria programable de sólo lectura o PROM. Cuando se ordenan las unidades PROM contienen "0" lógicos (ó "1" lógicos), en cada bit de las palabras almacenadas. Los enlaces en el PROM se rompen por medio de pulsos de corriente a través de los terminales de salida. Un enlace roto define un estado binario, y uno no roto representa el otro estado. Esto le permite al usuario programar la unidad en su propio laboratorio para lograr las relaciones deseadas entre las direcciones de entrada y las palabras almacenadas.

11 FALLA DE ORIGEN

Comercialmente se obtienen unidades especiales llamadas programadores de PROM para facilitar este procedimiento. De todas formas, todos los procedimientos para programar las ROM son procedimientos de los materiales (Hardware), aunque se use la palabra "programación". El procedimiento de los materiales para programar ROM y PROM es irreversible, y una vez programado el patrón dado, es permanente y no se puede alterar. Una vez que se ha establecido un patrón de bits se debe descartar la unidad si se quiere cambiar el patrón de bits. Un tercer tipo de unidad, es la llamada PROM borrable o EPROM.

Las EPROM pueden ser recuperables a su valor inicial (todos "1" o todos "0" lógicos), aunque se hayan cambiado previamente. Cuando una EPROM se coloca bajo una luz ultravioleta especial por un periodo dado de tiempo, la radiación de onda corta descarga los puentes internos que sirven de contactos. Una vez borrada la ROM regresa a su estado inicial para ser reprogramada. Ciertas ROM pueden ser borradas con señales eléctricas en vez de luz ultravioleta y se les llama algunas veces ROM eléctricamente alterable o EAPROM.

La función de una ROM puede interpretarse de dos maneras diferentes. La primera es la de una unidad que configura cualquier circuito combinatorial. Desde este punto de vista, cada terminal de salida se considera separadamente como una salida de una función de lógica expresada en suma de términos mínimos. La segunda interpretación considera la ROM como una unidad de almacenamiento que tiene un patrón fijo de cadenas de bits llamadas palabras. Visto de esta forma, las entradas especifican una dirección para una palabra específica almacenada que se aplica luego a las salidas.

Lectura se usa para implicar que el contenido de una palabra especificada por una dirección en una unidad de almacenamiento se localiza en los terminales de salida. Así, una ROM es una unidad de memoria con un patrón fijo de palabras que puede ser leído bajo la aplicación de una dirección dada. El patrón de bits en la ROM es permanente y no puede cambiarse durante la operación normal.

Las ROM se usan extensamente para ejecutar circuitos combinatoriales complejos directamente de sus tablas de verdad. Son muy útiles para convertir de un código binario a otro (tal como ASCII a EBCDIC), para funciones aritméticas como multiplicadores para mostrar caracteres en un IRL, y en cualquier otra aplicación que requiera un gran número de entradas y salidas.

Se emplean también en el diseño de unidades de control de los sistemas digitales. Como tales, se usan para almacenar patrones fijos de bits que representen una secuencia de variables de control necesarios para habilitar las diferentes operaciones en el sistema. Una unidad de control que utiliza una ROM para almacenar información de control binario, se llama una unidad de control #programada.

Un circuito combinacional puede tener ocasionalmente condiciones de no importa. Cuando se configura con una ROM una condición de no importa se convierte en una dirección de entrada que nunca ocurre. Las palabras en las direcciones de no importa no necesitan ser programadas, y pueden dejarse en su estado original (todos "1" o todos "0" lógicos).

El resultado es que no todos los patrones de bits disponibles en la ROM se usan, lo cual se considera como un desperdicio de equipo disponible. Para aquellos casos en los cuales el número de condiciones de no importa es excesivo, es más económico usar un segundo tipo de componente LSI, llamado Arreglo Lógico Programado (PLA).

I.4.- MEMORIA DE ACCESO ALEATORIO (RAM).

Estas memorias tienen como finalidad el almacenamiento de datos y resultados intermedios ó también se utilizan como memoria de programa para zonas del mismo que deben ser modificadas en curso de ejecución, para la fase de puesta a punto de un sistema, ó para sistemas en los que el programa es cargado al inicializar el sistema a partir de un medio auxiliar de memoria.

Las memorias RAM son volátiles, es decir, la información almacenada se pierde si falla la alimentación del sistema. Si es preciso salvar la información en caso de falla de tensión se deberá disponer de un sistema de baterías para alimentación de la RAM. En este caso las RAM de tecnología CMOS son especialmente útiles por su bajo consumo. Existen dos tipos de memoria RAM de acceso aleatorio y son:

a). RAM estáticas.- La información se almacena en un circuito biestable, de forma que el contenido se mantiene mientras haya alimentación.

b). RAM dinámicas.- La información se mantiene en forma de carga en la capacidad puerta-sustrato de un transistor MOS. Esta carga se disipa con el tiempo (del orden de milisegundos), por lo que es necesario un refresco periódico consistente en un ciclo de lectura/escritura en cada bit de memoria, realizado de forma paralela con todos los bits de una misma columna a la vez.

La principal desventaja de las memorias dinámicas es la necesidad de disponer de un circuito de refresco externo, en general complejo. Esta desventaja va desapareciendo al irse desarrollando CÍ's de memoria dinámica con circuitos de control de refresco incorporado. Además, algunos μP 's, incorporan en sus salidas de control señales para facilitar la circuitería de control de memorias dinámicas.

Las ventajas de la memoria dinámica es que permiten una mayor densidad de integración (actualmente del orden de 64Kbits frente a 16 de las estáticas), al precisar menos transistores por celda elemental de memoria, por lo que son más baratas que las RAM estáticas de la misma capacidad y consumen menos potencia.

Las memorias dinámicas acostumbran a organizarse en módulos de $N \times 1$ bits, lo cual implica que se debe utilizar un mínimo de 8 circuitos integrados para disponer de una memoria RAM para un μP de 8 bits de longitud de palabra. Esto es un inconveniente para sistemas reducidos pero no lo es para sistemas de gran capacidad de memoria, ya que la organización de $N \times 1$ es la más económica, para la misma capacidad.

I.5.- DISPOSITIVOS DE ENTRADA/SALIDA.

La solución lógica al programa de obtener máxima concurrencia en el procesamiento de la E/S, es disponer de un sistema de E/S inteligente que aisle a la UPC de los periféricos. La UPC sería entonces libre de proseguir a toda velocidad con su tarea primordial, de procesar el programa y manipular los datos internamente. El subsistema de E/S inteligente se obtiene mediante un procesador de E/S (PES).

Básicamente, un procesador de E/S es capaz de ejecutar un pequeño repertorio de órdenes para atender la petición de E/S. Este procesador está asociado directamente al bus del sistema, y es responsable de la selección y extracción de las órdenes de E/S individuales contenidas en la memoria principal. El PES contiene generalmente un procesador específicamente diseñado para el proceso de la E/S, y una serie de canales de E/S.

Los canales proporcionan una vía de comunicación entre el procesador de E/S y los controladores de dispositivos. También pueden existir canales de E/S sin PES.

En su forma más simple, y cuando está sólo, un canal puede ser un pequeño procesador que realiza operaciones de Acceso Directo a Memoria (DMA), para un reducido conjunto de dispositivos. Si el canal va incorporado dentro de un PES, se trata esencialmente de un componente pasivo sin capacidad por sí mismo de procesamiento lógico. Cuando el canal posee capacidad de proceso, se suele utilizar como PES.

I.6.- BUSES DE COMUNICACIONES.

Los distintos periféricos estarán enlazados con la UPC siguiendo una de las estructuras siguientes:

1.- Línea compartida (party line).- Los diversos módulos de entrada/salida comparten en el tiempo unas líneas únicas. En general, estas líneas son un subconjunto de los buses externos, por lo que el sistema periférico comparte el bus de direcciones, datos y control con el sistema de memoria. Esta estructura es la más utilizada, ya que reduce significativamente el número de líneas que precisa la UPC para la gestión de E/S y posibilita a la vez una concepción modular del sistema.

2.- Estructura radial (star).- En este caso cada periférico se conecta a la UPC a través de líneas dedicadas.

Este sistema está muy limitado por la cantidad de terminales que se precisan en la UPC. Se utiliza en sistemas en los que la UPC posee algunas líneas de E/S y en el caso de computadores integradas en una sola pastilla, donde la incorporación de memoria ROM y RAM en el mismo circuito integrado que el resto de la UPC, permite disponer de terminales para funciones de E/S.

3.- Estructura de cadena (daisy-chain).- En este caso las señales se propagan de un periférico a otro. Cuando un periférico concreto es seleccionado, bloquea la transmisión de las señales al resto del sistema. Este método se utiliza en algunos computadores para disponer de un sistema jerárquico de interrupciones, estableciéndose los niveles según la posición en la cadena.

Las necesidades de comunicación en un sistema computador aparecen a tres niveles distintos:

1.- Comunicación entre los elementos constituyentes de un circuito integrado.- Este sistema de comunicación constituye los buses internos, de estructura más o menos distinta para cada UPC considerada.

2.- Comunicación entre los distintos subsistemas del computador.- Unidad Central de Proceso, y entrada y salida.

El conjunto de las vías para este tipo de comunicación forman los buses externos, en general de carácter síncrono.

3.- Comunicación con periféricos.- Esta comunicación es a nivel de sistema E/S con periférico. El conjunto de líneas de transmisión que forman los buses periféricos, es en general de carácter asíncrono.

La información desde ó hacia la UPC y otros subsistemas puede ser de tres tipos: Direcciones, datos y control; por lo que funcionalmente se encontrarán tres tipos de buses en el computador, aunque no estén físicamente siempre separados, ya que pueden estar multiplexados en el tiempo sobre líneas comunes. La interconexión de la UPC con el sistema de memoria y E/S se realiza mediante estos buses.

1.- Bus de direcciones.- Son líneas de la UPC a memoria y sistema de E/S, que sirven para enviar la dirección del elemento seleccionado para una transferencia. Estas líneas estarán siempre gobernadas por la UPC, salvo en el caso de transferencia por Acceso Directo a Memoria. Pueden compartir ó parcialmente las líneas físicas con el bus de datos.

2.- Bus de datos.- Son líneas bidireccionales (ó dos buses unidireccionales independientes) para el envío de información (instrucciones y datos) entre subsistemas. Su anchura, en general, es la longitud de palabra del μP .

3.- Bus de control.- Son líneas de entrada ó salida de la UPC de órdenes ó de información de estado, que permiten coordinar la operación de todo el sistema. El bus de control presenta variaciones notables de un μP a otro. Existen dos tendencias en cuanto a su realización física:

a). Dar más señales de control elementales, que pueden estar codificadas, y que deben ser interpretadas temporalmente respecto al estado en que se produce en cada ciclo de máquina. Esta estructura se observa principalmente en μP 's de las primeras generaciones.

b). Dar un conjunto de señales completo y de multiplexado a las de memorias y periféricos, y para que respondan de una forma pasiva, por lo que son precisos circuitos especializados para la decodificación de las señales de control.

1.7.- INTERFASES DE COMUNICACION (SERIE Y PARALELO).

Varias familias de computadores, disponen de circuitos integrados LSI compatibles con el resto de los elementos del sistema, a fin de evaluar el sistema de entrada/salida. Las características fundamentales de estos subsistemas son:

1.- Facilidad de interconexión a los buses de comunicación.- Por ser elementos compatibles con el resto del sistema, en cuanto a señales y temporizaciones, es muy fácil conectarlos a los buses, requiriendo en general, solamente un sistema de decodificación de direcciones.

2.- Posibilidad de programación.- De esta forma, se logra disponer de un cierto número de opciones de funcionamiento. En general, la programación consiste en cargar de una forma determinada un registro interno de modo operacional, lo cual da, además, la posibilidad de una reprogramación dinámica, es decir, en fase de operación. Esta posibilidad no siempre es utilizada, pero una programación inicial que permita adaptar el sistema a las necesidades particulares de cada aplicación, amplía los campos de utilización de un determinado circuito, dando como resultado una modularidad y reducción en costo en comparación con el desarrollo de circuitos especiales para cada configuración.

3.- Inteligencia distribuida.- La posibilidad de que estos circuitos posean una cierta capacidad de decisión para desarrollar funciones de control, permite descargar al μP de estas tareas, simplificando de esta forma el programa y/o posibilitando determinadas prestaciones del sistema que serían imposibles de conseguir, si la unidad central tuviera que hacerse cargo de toda la gestión del sistema E/S.

Por la función que realizan, se pueden también dividir en tres categorías:

1.- Interfases dedicadas.- Son circuitos especializados para una función específica, aunque en general permiten un cierto número de operaciones. En esta categoría, se incluyen temporizadores, controladores de sistema (interrupciones y Acceso Directo a Memoria), y controladores de periféricos (discos flexibles, pantallas, teclados, protocolos, etc.).

2.- Interfases generales.- Son circuitos de tipo más universal que los anteriores, y tienen como misión posibilitar la transferencia y diálogo entre μP y los periféricos. Funcionalmente existen dos tipos distintos:

a).- Interfases para transmisión en paralelo.- Permiten disponer de un número elevado de líneas de entrada y salida, que permitirán el intercambio de información, así como el envío de órdenes y recepción de estado de una forma sencilla, flexible y rápida.

b).- Interfases para transmisión en serie.- Permiten utilizar canales de transferencia serie de un modo transparente al μP , es decir, se descarga a la unidad central de la tareas de serialización y sincronización.

Pueden ser sincronicas y/o asincronicas segun permitan un modo u otro de transmisión.

3.- μ Computadores especializados de entrada y salida.- En principio, cualquier μP puede considerarse a modo de "esclavo" de otro, realizando tareas bajo control del "director". Estas tareas pueden ser cualesquiera y por lo tanto la gestión de entradas y salidas es una de ellas.

1.8.- PROTOCOLOS DE COMUNICACION.

Una de las características claramente observable en la evolución de la tecnología de los computadores es la tendencia a la modularidad. Los elementos estructurales de los computadores se conciben cada vez más como unidades dotadas de cierta autonomía que cooperan entre sí. Esta tendencia se fundamenta no solo en la búsqueda de diseños más rápidos y eficientes sino también en principio de la división de funciones que facilita la concepción, diseño y mantenimiento de los diversos elementos que forman el sistema.

También es fácil constatar la atención que se ha prestado en la última década al estudio de los sistemas informáticos distribuidos que brindan la posibilidad de compartir recursos informáticos y aumentan la fiabilidad y disponibilidad de los sistemas a un precio justificable. Los multiprocesadores y las redes de computadores son ejemplos de este tipo de sistemas. La aparición y extensión de los μP 's incrementan el interés en los sistemas distribuidos al posibilitar la concepción de redes de μP 's y de nuevas arquitecturas multimicroprocesador formadas por un número elevado de μP 's que se comuniquen entre sí de forma bien definida, sin necesidad de un control centralizado.

Estas consideraciones sugieren la posibilidad de contemplar todo sistema informático como un conjunto de unidades más o menos autónomas con la función bien definida de colaborar entre sí para la consecución de tareas determinadas. La naturaleza y los objetivos perseguidos con esta cooperación difieren radicalmente de un sistema a otro, existiendo, sin embargo, ciertos aspectos comunes a todos ellos. Desde la óptica en que nos hemos situado puede señalarse con aspecto común de todos los sistemas la necesidad de intercambiar información entre los elementos que lo integran. Estos pueden ser circuitos, modems, concentradores, terminales, computadoras, procesos, personas, etc.

El intercambio de información entre los componentes de un sistema se denomina comunicación. La forma en que se realiza la comunicación depende de múltiples factores pero, en cualquier caso, es indispensable establecer claramente las reglas que han de seguirse en el intercambio de información. Se denomina protocolo de comunicación al conjunto de reglas que siguen la comunicación entre los elementos de un sistema. La materialización (Hardware ó Software) de estas reglas se recibe a sí mismo la denominación de protocolo.

Obsérvese que en estas definiciones no se ha impuesto ninguna restricción al tipo de información que intercambian los elementos del sistema. El concepto de comunicación se extiende desde el intercambio de datos, que presumiblemente involucrará procesos complejos de Software, hasta la simple señal que notifica a un elemento el estado de otro.

Obsérvese que un protocolo determinado controla el flujo de información entre las dos partes físicamente separadas de un mismo nivel. Ahora bien, igualmente necesario es establecer las reglas de comunicación entre los distintos niveles. Para distinguir los protocolos "horizontales" (entre partes de un mismo nivel) de los "verticales" (entre niveles adyacentes), se denomina a estos últimos interfases de comunicación. La utilización del término interfase es coherente con el hecho de que generalmente los equipos que realizan los protocolos de distinto nivel son diferentes, constituyendo las reglas de comunicación una verdadera interfase. Contrariamente, la comunicación dentro de un nivel suele realizarse entre entidades idénticas o cuando menos similares.

Las funciones básicas de los protocolos e interfases son:

1.- **Direccionamiento.** - La especificación del origen y destino de la información.

2.- **Control de error.** - La detección y recuperación de errores en la transmisión sobre el canal virtual de comunicaciones que se utiliza. En general, todo protocolo ha de garantizar la transmisión correcta de una única copia de cada mensaje.

3.- **Control de flujo.** - Se engloban en esta función las operaciones destinadas al mantenimiento del flujo de la información como son la selección de la ruta que han de seguir los mensajes, la reserva de espacio (de memoria, en la estación destino para información que ha de enviarse), etc.

4.- **Sincronización.** - Todo protocolo ha de ser capaz de mantener en sincronismo las partes de un mismo nivel en el curso de la comunicación.

Otras funciones características de los protocolos son la conexión y desconexión de circuitos físicos en ciertas aplicaciones, el mantenimiento de una secuencia ordenada de los mensajes, la segmentación de mensajes en unidades de transmisión más pequeñas (paquetes) y su ulterior reconstrucción, el tratamiento de mensajes con prioridades diversas, etc.

CAPITULO II.

ARQUITECTURA DEL MICROPROCESADOR μP 80386.

II.1.- INTRODUCCION AL μP 80386.

En octubre de 1985, INTEL organizó en Madrid, España; la presentación mundial de su nuevo μP de 32 Bits, el modelo 80386 en unión con un conjunto de componentes y tarjetas y un equipo lógico complementario, que configuran la familia 386.

El μP 80386 consta de más de 275,000 transistores y está fabricado en tecnología CMOS. La familia 80386 es compatible con todo el sistema lógico desarrollado para el μP 8086 y sus sucesores. Su costo inicial de venta será de \$299USA en EUA.

Dos aspectos destacables en el μP 80386 son su velocidad de funcionamiento, que oscila entre 4 y 5 MIPS (millones de instrucciones por segundo) y la capacidad de direccionamiento de memoria virtual, que supera los 64 trillones de bytes. Soporta una configuración multi-ejecución, que permite realizar distintos programas escritos en entornos correspondientes a diversos sistemas operativos, como UNIX, MS-DOS e iRMX.

La Unidad de Manejo de Memoria (MMU), está integrada en el μP 80386 y soporta los dos métodos de gestión de memoria más empleados: La segmentación y la paginación.

El μP tiene una memoria tampón o cache de 32 posiciones, en las que se almacena las direcciones físicas y sus correspondientes direcciones físicas, que más se utilizan en el programa de ejecución durante el periodo de tiempo anterior. Dichas direcciones actúan de puentes sobre una memoria tampón exterior, de cualquier tamaño, mediante la que se accede al código ó los datos necesarios. El hecho de que la memoria tampón interna contenga las tablas de traducción más usadas en el programa, minimiza el tiempo empleado en la gestión de la memoria.

Otra característica avanzada del μP 80386 la constituye la propia autocompulsión de más de las 747 partes de las funciones internas, que se produce en el momento de aplicar la tensión de alimentación. Otra novedad de este μP es la de permitir generar interrupciones cuando se accede a la memoria de datos.

Dentro de los cuatro campos en los que se prevee el mayor consumo de μP 's de 32 Bits: Automatización industrial, aplicaciones científicas, automatización de gestión y telecomunicaciones, existen unos sectores concretos en los que se implantará rápidamente el $\mu PB0386$ y son:

- Robótica y visión artificial.
- CAD/CAM/CIM.
- Tratamiento de textos y mensajes electrónicos.
- Sustitución de minicomputadores por μ computadores en configuraciones multi-usuario.
- Reconocimiento del lenguaje natural.
- Centrales telefónicas avanzadas.

(Fig. 11.1).

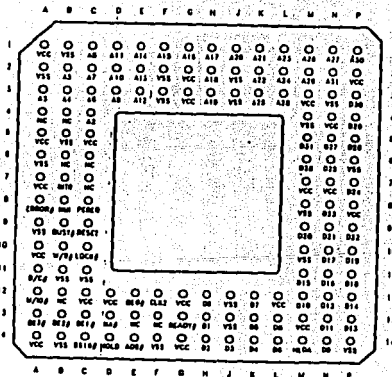


Fig.- 11.1.- Vista del $\mu PB0386$ desde el lado de las terminales. Las terminales asignadas por NC, no se conectan.

Existen varias terminales V_{CC} que deben conectarse al positivo de la alimentación, o sea, + 5V +/- 5% y varias GND que se conectan a tierra.

El μ P80386 está diseñado para funcionar con una temperatura comprendida entre 0 y 85 °C, medida en el centro de la pastilla, por el lado donde no hay terminales.

Otras características interesantes del μ P de 32 bits se exponen en la tabla de la figura II.2 que son válidas para las versiones de 12 y 16 MHz, con una alimentación de + 5V +/- 5%.

SÍMBOLO	PARÁMETRO	MÍNIMO	MÁXIMO
	TEMPERATURA DE ALMACENAMIENTO	-80°C	+130°C
V _{IL}	VOLTAJE DE ENTRADA BAJO	-0,3V	0,8V
V _{IH}	VOLTAJE DE ENTRADA ALTO	2V	V _{CC} +0,3V
V _{OL}	VOLTAJE DE SALIDA BAJO		0,45V
	I _{OL} * 4mA; A2-A31, D0-D31		
V _{OH}	VOLTAJE DE SALIDA ALTO		
	I _{OH} * 1mA; A2-A31, D0-D31	2,4V	
I _{CC}	CORRIENTE DE ALIMENTACION		400mA
C _{IN}	CAPACIDAD ENTRADA (f _c * 1MHz)		10pF
C _{OUT}	CAPACIDAD SALIDA (f _c * 1MHz)		12pF

Fig.-II.2.- Características relevantes comunes a las versiones 80386.

11.2.- ARQUITECTURA INTERNA DEL μP 80386.

El μP 80386 consta de tres grandes bloques:

- 1.- Unidad Central de Proceso (UCP).
- 2.- Unidad de Manejo de Memoria (MMU).
- 3.- Unidad de Interfase de Bus (BIU).

En la figura 11.3 se muestra la distribución de los tres bloques fundamentales del μP 80386, así como la relación que existe entre ellos.

La UCP de μP 80386 se compone de una Unidad para la Decodificación de las Instrucciones y otra de Ejecución. La Unidad de Ejecución de instrucciones dispone de una zona con 8 registros de propósito general de 32 bits, que se emplean en el cálculo del direccionamiento y en la manipulación de datos. Tiene también un Registro de Desplazamiento Rápido de 64 bits, dentro de la Unidad de Ejecución, muy eficaz en las operaciones de rotación, multiplicación, división y desplazamiento. La ALU completa la Unidad de Ejecución.

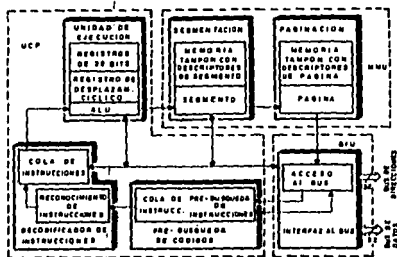


Fig. - 11.3.- Estructura general del 80386, formada, esencialmente, por los bloques UCP, MMU y BIU.

La Unidad de Manejo de Memoria, posee una zona dedicada al procedimiento de la paginación y otra para la segmentación. La memoria está organizada en uno o varios segmentos de longitud variable, con un máximo de 4 Gigabytes cada uno.

Cada segmento se divide en una ó mas paginas de 4 Kbytes cada una. Cada tarea en el $\mu P80586$ puede ocupar un máximo de 16,381 segmentos, lo que supone una memoria virtual de 64 Terabytes (64 trillones de bytes).

Cada región del espacio de memoria ó segmento se caracteriza por ciertos atributos, como posición, tamaño, tipo (código, dato, pila) y protección.

La unidad de segmentación proporciona 4 niveles de protección para aislar las aplicaciones o impedir la entrada de otros usuarios. El $\mu P80586$ puede trabajar de dos formas:

- 1.- Modo real (modo de direccionamiento real).
- 2.- Modo protegido (modo de direccionamiento virtual protegido).

En el modo real el $\mu P80586$ funciona como un $\mu P8086$ muy rápido y con una extensión de 32 Bits en el bus de direcciones.

En el modo protegido se puede acceder a formas de manejo de memoria sofisticadas, con capacidad para la paginación.

La BIU (Unidad de Interfase de Bus), del $\mu P80586$ facilita enormemente el diseño del equipo físico al copiar direccionado "pipelined", información dinámica del bus de datos y señales de permiso de byte para cada uno de los bytes de que se compone el bus de datos. La arquitectura "pipeline" admite las funciones de búsqueda de codificación, ejecución y manejo de memoria en paralelo.

11.3.- CONEXIONADO DEL μ P80386.

El bus de datos y el bus de direcciones del μ P80386 consta de 32 líneas cada uno, que son independientes entre sí, en oposición con el multiplexado que existe en los μ P's de 8 y 16 bits de INTEL.

El bus de datos es bidireccional y el de direcciones que tiene carácter unidireccional, utiliza dos de sus líneas para generar señales de habilitación para los cuatro bytes en que se puede descomponer un dato de 32 bits. El control de ambos buses se realiza mediante las señales asociadas que genera el propio μ P.

El ciclo de bus de μ P es el mecanismo básico para la transferencia de información entre el sistema y el μ P. El ciclo de bus del μ P80386 ejecuta una transferencia de datos en sólo dos períodos de reloj. Con un bus de datos de 32 Bits, a una frecuencia de 16 Mhz, el μ P80386 admite un nivel de transferencia que alcanza los 32 Mbits/s. En la fig. 11.4 se presenta un esquema de la arquitectura del μ P80386 y su interrelación con el mundo exterior a través de las señales de entrada y salida de los buses. En la fig. 11.5 se presenta un esquema de los 132 terminales de la pastilla, visto por su parte superior.

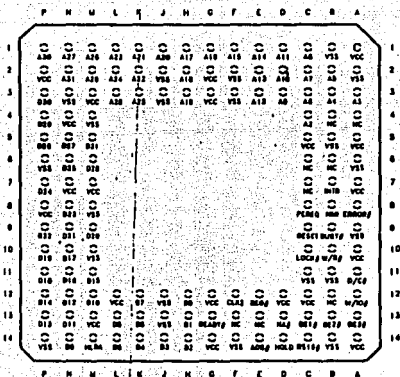


Fig. 11.5.- Distribución de las 132 terminales del μ P80386, visto desde la parte superior de la cápsula.

La Fig. 11.6 muestra las señales de interconexión del μ PB0386 con los demás elementos del sistema, agrupadas por funciones y son las siguientes:

1.- CLK2 (Señal de Reloj).- Por el terminal de entrada al μ PB0386, referenciada por CLK2, se introduce el control de tiempos necesario. CLK2 se divide por 2, para generar la señal de reloj interna que controla el funcionamiento del μ P en la ejecución de las instrucciones.

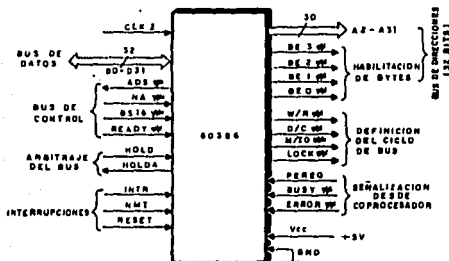


Fig. 11.6.- Señales de control y adaptación del μ PB0386 agrupadas en funciones comunes.

2.- D0-D31 (Bus de Datos).- Son 32 líneas que admiten el estado flotante y tienen carácter bidireccional. Una característica especial del bus de datos es que puede transferir información de 16 y de 32 Bits, usando como elemento de control la señal auxiliar de entrada DS16#.

3.- A2-A31 y BE0#BE3#. (Bus de direcciones).- Estas líneas de salida del μ PC80386 de tipo triestado, proporcionan la dirección física de una posición de memoria o de una Entrada/Salida. Con este lamazo de bus se pueden direccionar 4,000,000,000 de bytes (4 Gigabytes), comprendidos entre las direcciones físicas 0000 0000 y la FFFF FFFF. El bus de direcciones se encarga también de controlar 64 K Bytes dedicados a Entradas/Salidas y que ocupan las direcciones 0000 0000 hasta la 0000 FFFF.

Las cuatro señales auxiliares del bus de direcciones BE0#-BE3# de habilitación de byte, indica cuál de los cuatro bytes, que componen los 32 bits del bus de datos, se emplean en la transferencia en curso. Estas cuatro señales se obtienen por la combinación de las líneas del bus de direcciones A0 y A1 y el control de los bytes se efectúa de la siguiente forma:

BE0# controla el byte contenido en las líneas de datos D0-D7.

BE1# controla el byte contenido en las líneas de datos D8-D15.

BE2# controla el byte contenido en las líneas de datos D16-D23.

BE3# controla el byte contenido en las líneas de datos D24-D31.

Cuando se está realizando un ciclo de escritura y el operando se transfiere por las 16 líneas de más peso de bus de datos (D16-D31), dicha información se duplica en las 16 líneas de menos peso (D0-D15), para optimizar la escritura sobre buses de 16 líneas.

4.- Señales que definen el ciclo del bus.- Las señales triestado W/R#, D/C#, M/I0# y LOCK# sirven para definir el ciclo de bus que se ejecuta.

W/R#: Distingue entre los ciclos de escritura y los de lectura.

D/C#: Distingue entre los ciclos de datos y los de control.

M/I0#: Distingue entre los ciclos de memoria y los de Entrada/Salida.

LOCK#: Distingue entre los ciclos cerrados y abiertos.

5.- Señales de control del bus.- Las señales comprendidas en este grupo, indican el principio y el fin del ciclo del bus, el control de direccionamiento pipeline a otros sistemas a la anchura del bus de datos que se emplea.

ADS#. (Estado de dirección).- Es una salida triestado que indica la validez de un ciclo de bus y la conducción de una dirección por las terminales adecuadas del #PB0386 (A2-A31, BE0#-BE3#, W/R# y M/I-0#).

READY#. (Reconocimiento de transferencia).- Esta señal de entrada al procesador indica la terminación del ciclo del bus y la aceptación de los bytes activos determinados por BE0#-BE3# y por BS16#.

NA#. (Petición de la dirección siguiente).- Esta entrada señala que el sistema está preparado para recibir nuevos valores de BE0#-BE3#, A2-A31, W/R#, D/C# y M/I-0# desde el #PB0386, aunque el ciclo de bus actual aún no ha sido reconocido por READY#.

BS16#. (Tamaño del bus).- Esta señal permite al #PB0386 conectarse, de forma directa, con sistemas de buses de datos de 16 y 32 bits. La habilitación de esta señal restringe, el ciclo de bus en curso, a utilizar sólo la mitad menos significativa del bus de datos (D0-D15), correspondientes a BE0# y BE1#.

6.- Señales de arbitraje de bus.-Este grupo de líneas sirven al procesador para abordar el control de los buses locales cuando se produce una petición por otro master.

HOLD. (Petición del bus).- Esta señal de entrada al μ P80386 indica que otro dispositivo solicita el dominio del sistema de buses.

La señal HOLD permanece activa durante todo el tiempo que se cede el bus. Mientras permanezca activa la señal RESET, no se conoce a HOLD.

HLDA. (Reconocimiento de la cesión del bus).- La activación de esa línea de salida significa que el μ P80386 ha cedido el control de su bus local, en respuesta a una petición de HOLD.

La aceptación de HOLD deja en estado de alta impedancia las siguientes terminales del procesador: D0-D31, A2-A31, BE0#-BE3#, W/R#, D/CH#, M/I-0# LOCK# y ADS#.

7.- Señales de adaptación desde coprocesador.- Son tres terminales de entrada al μ P80386 que se encargan de implantar la interfase con el coprocesador numérico 80287 y 80387.

PEREQ. (Petición de coprocesador).- La activación de esta línea significa la petición del coprocesador para que sea transferido un dato a/desde la memoria por el μ P.

BUSY#. (Coprocesador ocupado).- La activación de esta señal, indica que el coprocesador está todavía ejecutando una instrucción y no puede aceptar otra nueva.

ERRDR#. (Error de coprocesador).- Esta señal de entrada, indica que la instrucción anterior del coprocesador, ha generado un tipo de error no enmascarable por el registro de control del coprocesador. Cuando el μ P80386 encuentra una instrucción para el coprocesador, explora la línea ERRDR#, y si está activa, genera la excepción del tipo 7, con objeto de acceder al sistema lógico de tratamiento de los errores.

B.- Señales de interrupción.- A continuación se presentan las tres señales características de este grupo y son:

1.- INTR. (Petición de interrupción enmascarable).- La activación de esta línea de entrada del μ P80C86, representa una petición de interrupción, que puede enmascararse con el bit IF del Registro de Señalizadores.

Cuando el μ P80C86 responde a una petición INTR, realiza dos ciclos de reconocimiento y, al final del segundo, carga un vector de interrupción de 8 bits sobre D0-D7, que identifica el origen de la interrupción. La señal INTR es activa por nivel y asíncrona con la señal CLK2. Para que se asegure el reconocimiento de una interrupción INTR, ésta ha de permanecer activa hasta el inicio del primer ciclo de reconocimiento de interrupción.

2.- NMI. (Interrupción no enmascarable).- Esta señal introduce al μ P80C86, una petición de servicio de interrupción, que no puede ser enmascarada con instrucciones en el programa. Siempre es atendida y la rutina a la que se accede, viene determinada por el puntero 2 de la Tabla de Interrupciones. Es cautiva por flanco ascendente y asíncrona con respecto a CLK2. Una vez iniciada la atención a una petición NMI, se ignoran otras peticiones NMI, hasta que no se ejecute la instrucción de retorno de interrupción IRET.

3.- RESET. (Reinicialización).- Al activarse esta señal se suspende cualquier operación en curso y se realiza una rutina que deja al μ P80C86 en un estado conocido. Para que el μ P acepte la señal de RESET, esta señal ha de permanecer activa durante 15 ó más periodos de CLK2. Cuando es aceptada la señal de RESET, se ignoran las restantes señales de entrada y se pasa a un estado de inactividad, en el que las señales del μ P80C86 toman los siguientes estados:

- ADS#	Alto
- D0-D31	Alta impedancia
- BE0#-BE3#	Bajo
- A2-A31	Alto
- W/R#	Alto
- D/C#	Alto
- M/I-0#	Bajo
- LOCK#	Alto
- HLDA	Bajo

II.4.- REPERTORIO DE INSTRUCCIONES DEL μ P80386.

Las instrucciones del μ P80386 soportan 0, 1, ó 3 operandos, que pueden residir: En un registro, en la memoria, ó en la propia instrucción. La mayoría de las instrucciones que carecen de operando ocupan un byte y las de un operando, generalmente, 2 bytes. El uso de dos operandos permite actuar sobre los siguientes elementos:

- Registro a registro.
- Memoria a registro.
- Inmediato a registro.
- Memoria a memoria.
- Registro a memoria.
- Inmediato a memoria.

Los operandos pueden constar de 8, 16 ó 32 bits. El repertorio de instrucciones del μ P80386 se clasifica en 9 grupos, según el tipo de operación que realizan:

- 1.- De transferencia de datos.
- 2.- Aritméticas.
- 3.- De desplazamiento y rotación.
- 4.- De manipulación de cadenas.
- 5.- De manipulación de bits.
- 6.- De transferencia de control.
- 7.- De soporte a los lenguajes de alto nivel.
- 8.- De soporte a sistemas operativos.
- 9.- De control del μ P.

1.- Instrucciones Para La Transferencia de Datos:

a). De proposito General.

- MOV: Mover operando.
- PUSH: Cargar operando en la Pila.
- POP: Sacar operando de la Pila.
- PUSHA: Cargar el contenido de todos los registros en la Pila.
- POPA: Sacar todos los registros de la Pila.
- XCHG: Intercambio de operandos.
- XLAT: De traducción.

b). De Conversión.

MOVZX: Mover byte, palabra ó doble palabra con extensión de ceros.

MOVSX: Mover byte, palabra ó doble palabra con extensión de signo.

CBW: Convertir byte a palabra ó palabra a doble palabra.

CDW: Convertir palabra a doble palabra.

CDQE: Convertir palabra a doble palabra extendida.

CDD: Convertir doble palabra a palabra cuádruple.

c). De Entrada y Salida.

IN: Entrada de un operando desde el espacio de Entrada/Salida.

OUT: Salida de un operando al espacio de Entrada/Salida.

d). De Actuación Sobre Dirección.

LEA: Cargar la dirección efectiva.

LSD: Cargar puntero en el registro D.

LES: Cargar puntero en el registro E.

LFS: Cargar puntero en el registro F.

LGS: Cargar puntero en el registro G.

LSS: Cargar puntero en el registro S.

e). De Manipulación de Señalizadores.

LAHF: Carga en el registro A los señalizadores (flags).

SAHF: Almacena el registro A en el registro de señalizadores.

PUSHF: Carga el registro de señalizadores en la Pila.

POPF: Saca el registro de señalizadores de la Pila.

PUSHFD: Carga el registro EFLAGS en la Pila.

POPF: Saca el registro de señalizadores de la Pila.

CLC: Borra el señalizador de acarreo.

CLD: Borra el señalizador de Dirección.

CMC: Complementa el acarreo.

STC: Pone a "1" lógico el acarreo.

STD: Pone a "1" lógico el señalizador de Dirección.

2.- Instrucciones Aritméticas:

a). Suma.

ADD: Suma operandos.
ADC: Suma con acarreo.
INC: Incrementa el operando una unidad.
AAA: Ajuste ASCII para la suma (no empaquetado).
DAA: Ajuste decimal para la suma (empaquetado).

b). Resta.

SUB: Resta operandos.
SDB: Resta con llevada.
DEC: Decrementa una unidad el operando.
NEG: Niega el operando.
CMP: Compara operandos.
AAS: Ajuste ASCII para la resta.

c). Multiplicación.

MUL: Multiplica con simple ó doble precisión.
IMUL: Multiplica enteros.
AAM: Ajuste ASCII para después de la multiplicación.

d). División.

DIV: Divide números sin signo.
IDIV: Divide números enteros con signo.
AAD: Ajuste ASCII para después de la división.

3.- Instrucciones Para el Manejo de Cadenas:

MOVS: Mueve una cadena (string) de bytes, palabras ó dobles palabras.

INS: Entrada de una cadena desde el espacio de Entrada/Salida.

OUTS: Salida de una cadena al espacio de Entrada/Salida.

CMPS: Compara bytes, palabras ó dobles palabras de una cadena.

SCAS: Busca Bytes, palabras ó dobles palabras de una cadena.

LODS: Carga una cadena de bytes, palabras ó dobles palabras.

STOS: Almacena una cadena de bytes, palabras ó dobles palabras.

REP: Repite una operación sobre una cadena, el valor de C.

REPE/REPZ: Repite según C, mientras sea igual a cero.

RENE/REPZ: Repite según C, mientras no sea igual ó no sea cero, a semejanza con la instrucción del $\mu P8086$ que tiene igual nemónico.

4.- Instrucciones Lógicas:

a). Booleanas.

NOT: Operación lógica de inversión.

AND: Operación AND.

OR: Operación OR.

XOR: Operación OR Exclusiva.

TEST: Operación AND sin resultado. Sólo afecta los señalizadores.

b). De Desplazamiento.

SHL/SHR: Desplazamiento lógico a la izquierda ó a la derecha.

SAL/SAR: Desplazamiento aritmético a la izquierda ó a la derecha.

SHLD/SHRD: Doble desplazamiento a la izquierda ó a la derecha.

c). De Rotación.

RDL/RDR: Rotación a la izquierda ó a la derecha.

RCL/RCR: Rotación a través del Acarreo a la izquierda ó a la derecha.

5.- Instrucciones para la Manipulación de Bits:

a). De simple bit.

BT: Prueba de un bit.

BTS: Prueba de un bit y puesta a "1" lógico.

BTR: Prueba de un bit y puesta a "0" lógico.

BTC: Prueba de un bit y complemento.

BSF: Búsqueda de un bit hacia adelante.

BSR: Búsqueda de un bit hacia atrás.

b). De cadena de bits.

IBTS: Inserta cadena de bits.

XBTS: Intercambia cadena de bits.

6.- Instrucciones Condicionales:

SETCC: Poner el byte igual al código de condición.

JAZ/JNBE: Salta si más alto/no más bajo o igual.

JAE/JNBE: Salta si más alto o igual/no menor.

JBE/JNAE: Salta si más bajo/no mayor o igual.

JC: Salta si el Acarreo es "1" lógico.

JE/JZ: Salta si igual/cero.

JG/JNLE: Salta si mayor/no menor o igual.

JL/JNGE: Salta si menor/no mayor o igual.

JLE/JNBE: Salta si menor o igual/no mayor.

JNE: Salta si no acarreo.

JNE/JNZ: Salta si no igual/no cero.

JNO: Salta si no Overflow.

JNF/JFO: Salta si no Paridad/paridad impar.

JNS: Salta si el señalizador de Signo es 0.

JO: Salta si hay Overflow.

JF/JPF: Salta si Paridad/paridad par.

JS: Salta si el señalizador de Signo es 1.

a). Transferencias incondicionales.

CALL: Llamada a procedimiento ó tarea.
RET: Retorno desde procedimiento ó tarea.
JMP: Salto incondicional.

b). De control de iteración.

LOOP: Bucle repetitivo según C.
LOOPE/LOOPZ: Bucle repetitivo si igual/cero.
LOOPNE/LOOPNZ: Bucle si no igual/no cero.
JCXZ: Salta si el registro CX = 0.

c). De interrupción.

INT: Interrupción por programa.
INTO: Interrupción si hay Overflow.
IRET: Retorno desde interrupción.
CLI: Pone a cero el señalizador de Interrupción.
SLI: Pone a uno el señalizador de Interrupción.

7.- De Soporte a Los Lenguajes de Alto Nivel:

BOUND: Comprueba los límites de un array ó una tabla.
ENTER: Inicializa parámetros para entrar en un procedimiento.
LEAVE: Dejar procedimiento.

8.- Para Modelo de Protección:

SGTD: Almacena Tabla de Descriptores Globales (GTD).
SIDT: Almacena Tabla de Descriptores de Interrupción
(IDT).
STR: Almacena Registro de Tarea (TR).
SLDI: Almacena Tabla de Descriptores Locales (LDT).
LGDT: Carga Tabla de Descriptores Globales.
LTR: Carga Registro de Tarea.
LIDT: Carga Tabla de Descriptores Locales.
LLDT: Carga Tabla de Descriptores Locales.
ARPL: Ajuste al nivel de privilegio solicitado.
LAR: Carga derechos de acceso.
LSL: Carga límite de segmento.
VERR/VERW: Verifica segmento para lectura ó escritura.
LWSW: Carga la "palabra de estado de la maquina", que
son los 16 Bits de menos peso de CR0.
SMSW: Almacena la palabra de estado.

9.- Instrucciones para el control del #P:

HLT: Alto.
WAIT: Espera hasta que BUSY# se niegue.
ESC: Escape.
LOCK: Bus LOCK.

CAPITULO III.

ARQUITECTURA DEL μ P80486.

III.1.- INTRODUCCION AL μ P80486.

La serie de μ P's 80486, es en gran parte una versión mejorada de la serie μ P80386; ya que cuenta fundamentalmente con 2 adiciones incluídas en el mismo integrado:

- 1.- Una memoria caché de 8 Kbytes.
- 2.- Una unidad de números reales (FPU, también conocida como un co-procesador matemático).

Los Ingenieros de Intel también rediseñaron la arquitectura de los circuitos lógicos del integrado 80486, para que pueda ejecutar más operaciones en menos ciclos de reloj.

Existen varias versiones del μ P80486, ya que se tienen:

- 1.- μ P80486DX.
- 2.- μ P80486SX.
- 3.- μ P80486DX2.

El diseño mejorado del μ P80486DX le da a las unidades 486, mejoras significativas en el rendimiento del μ P y la memoria. Una unidad 486 de 33MHz, produce un rendimiento en el μ P de un 85% como mejora, y un rendimiento en memoria que duplica el de una unidad 386 de 33MHz.

El coprocesador matemático interno en el μ P80486DX mejora el rendimiento del Software que lo utiliza; Analisis estadístico, CAD y otros. Este coprocesador matemático interno del μ P80486, maneja los cálculos de números reales al doble de la velocidad de la combinación de un μ P80386DX y un μ P80387DX que es un coprocesador, todo esto operando a la misma frecuencia de reloj.

Intel fabricó el μ P80486SX el cual es básicamente un μ P80486DX sin el FPU, para aquellas aplicaciones en las cuales no es necesario el coprocesador. El μ P80486SX sirve como un μ P80386DX más eficiente y tiene la capacidad de mejora que Intel incluyó en su línea 486; se encuentra en versiones de 16, 20 y 25 MHz.

El μ P80486SX está diseñado para ejecutar instrucciones al doble de la velocidad de un μ P80386 que opere a la misma velocidad.

El μ P80486DX2, es un μ P80486DX que ejecuta internamente al doble de velocidad del reloj conectado, pero externamente ejecuta al mismo tiempo que marca el reloj. Por ejemplo; un μ P80486DX2 de 50MHz, usa un reloj de cristal de 25 MHz y ejecuta internamente a 50 MHz pero externamente a 25 MHz. Esta tecnología, permite el uso de componentes en la tarjeta madre con frecuencias más bajas para crear un sistema más accesible. Una unidad 486 de 50 MHz, es casi 50 veces más veloz que una iCMXT original; como una indicación más práctica se dice que una 486DX de 50 MHz es casi 5 veces más rápida que una 386SX de 20 MHz.

El cache incluido en la 486 es más eficiente que un cache externo, porque tiene una vía de acceso de información directa de 120 Bits al circuito de procesamiento del integrado. Intel limitó el tamaño a 8 Kbytes, porque sus circuitos ocupan 1/3 del área del integrado.

Los caches externos que se deben de conectar al 486 mediante su ruta de datos de 32 Bits, pueden mejorar el rendimiento porque pueden ser tan grandes como se quiera. Se debe mencionar que un cache es un bloque pequeño de memoria estática (SRAM), rápido pero costoso que opera sin ciclos de espera que se interponen entre el μ P y la lenta memoria del sistema.

Un controlador de cache intenta anticipar las necesidades del μ P y llena el cache con el contenido de la memoria que tenga más probabilidad de acceso. El cache tiene un "hit" o acierto cuando la información necesaria está en el cache y el μ P no tiene que esperar para extraerla; el cache tiene un "mistake" o fracaso, cuando la información necesaria no está en el cache y el μ P debe esperar a que la información se extraiga de la memoria general. En general, mientras más grande sea el cache mayor será la probabilidad de éxito.

Por otra parte, ya existen μ P80486SL, los cuales representan un gran ahorro de energía, ya que presentan una alimentación de 3.5 Volts, y con un consumo de potencia mínimo.

Los μ P's SX, se utilizan para sistemas de escritorio a nivel captura; los de clase DX de alto rendimiento se emplean en instrucciones frecuentemente utilizadas en menos ciclos, aún a costa de utilizar más ciclos para las instrucciones utilizadas menos frecuentemente, lo que resulta de esto es una ganancia significativa en el rendimiento. Un resultado del perfil de Intel, es que se está acumulando una enorme base de datos de trazos de instrucciones de diversas aplicaciones comerciales.

El nuevo 486SL tiene un bus PI (Interfase Periférica) que es funcionalmente equivalente a un bus local, en una computadora de escritorio. El bus PI, permite que un controlador gráfico o una tarjeta relámpago cortocircuiten el bus de Entrada/Salida, y se enlace directamente a la UPC; el bus PI opera a la velocidad del reloj del UPC, comparado con el relativamente, lento ancho de banda del bus de Entrada/Salida que es de 8 MHz.

Algunas características de los 486 son:

- 1.- Efectúan el "pipeline" de cinco etapas.
- 2.- Además se emplea una tecnología de un μ rón, para empacar 1.2 millones de transistores.
- 3.- Estaciones y servidores de alta tecnología.

Intel fue el pionero en el concepto de una actualización de un sólo integrado con su μ P llamado "Overdrive"; el cual aumenta al doble la velocidad del reloj interno de un μ P e incrementa su rendimiento general en aproximadamente un 50%. Anteriormente, al integrado "Overdrive", la mayoría de las actualizaciones se llevaban a cabo fuera, mediante el reemplazo de la Tarja Principal o mediante la instalación de tarjetas extra o Tarjetas bujas.

Una de las características de estos nuevos μ P's 80486, es la perfilación de instrucciones (la cual se empleaba en los ambientes de minicomputadoras y mainframes). La perfilación de instrucciones es el análisis de la frecuencia relativa de las instrucciones utilizadas por las aplicaciones. Con esta información los ingenieros pueden afinar una arquitectura para que ejecute las estaciones y servidores de alta tecnología.

III.2.- ARQUITECTURA INTERNA DEL μ P00486.

Existen varias versiones del μ P00486, las cuales, presentan diversas características, las diferentes compañías presentan los siguientes CI's durante los primeros seis meses de 1993:

- 1.- Intel 486SX/33 el cual es simplemente una versión más rápida de la línea existente.
- 2.- IBM 486SCL es la versión de IBM del CI de Intel.
- 3.- CYRIX 33MHz 486SCL es una versión más rápida del SCL con un conjunto totalmente nuevo de características de manejo de energía.
- 4.- Intel 486SL proporciona bajo consumo de energía.
- 5.- CYRIX DRU², Un 486DCL que aumenta al doble el rendimiento de un sistema 386DX a 33MHz.
- 6.- CYRIX 486SC/50 es un μ P compatible con el 486SX con una memoria caché más pequeña que el equivalente de Intel. CYRIX asegura que haciendo la memoria caché de "retroescritura", se compensa con el tamaño más pequeño.

El μ P00486 es un μ P avanzado de 32 Bits, diseñado para aquellas aplicaciones donde sea requerida una alta perfección y optimización para sistemas que operen en multitarea. Los registros de 32 Bits y las direcciones de este procesador le permiten direccionar arriba de 4 Gigabytes de memoria física y 64 Terabytes de memoria virtual.

El manejador de memoria integrado y la arquitectura de protección incluye el traspaso de direcciones, registros y un mecanismo de protección para soportar sistemas operativos variados, además de la capacidad de realizar multitareas.

Este μ P, es capaz de realizar arriba de 5 MIPS (Millones de Instrucciones Por Segundo). Mantiene compatibilidad con los códigos objeto de todos los miembros de la familia 8086, de tal manera, que puede ser "conectado" a la mayoría de los μ P's del mundo que mantengan la misma base.

El μ P00486 consiste de una Unidad Central de Proceso; una Unidad Manejadora de Memoria y en Bus de Manejo de Interfase, además dependiendo de la versión podía contener ó no un coprocesador matemático integrado. La unidad de ejecución contiene registros de propósito general de 32 Bits, los cuales son empleados para el cálculo de direcciones, operación de datos y un manejador de 128 Bits empleado para rotar, multiplicar y dividir operaciones.

La unidad de instrucción decodifica los códigos de operación y los almacena dentro del decodificador de instrucciones para posteriormente emplearlos por la unidad de ejecución.

La unidad manejadora de memoria consiste de una unidad de segmentación y una unidad de página; la segmentación, es para el manejo de las direcciones un espacio lógico para suministrar un componente de direccionamiento extra, de tal manera, que un código y un dato son fácilmente relocables, el mecanismo de paginación opera de una manera transparente a la segmentación, cada segmento es dividido en uno ó más segmentos de página de 4 Kbytes.

Para implementar un sistema de memoria virtual, el μ FB0486 soporta reinicio total para todas las páginas y segmentos.

La memoria está organizada dentro de uno ó más segmentos de longitud variable, cada uno de ellos arriba de 4 Gigabytes. Una región dada por el espacio de las líneas de dirección, puede ser asociado con un segmento, donde se tendrán una serie de atributos, estos incluyen su localización, tamaño, tipo (si es código ó dato) y características de protección. Cada grupo del μ FB0486, puede tener un máximo de 16381 segmentos con 4 Gigabytes cada uno, lo cual suministra un total de 64 Terabytes (trillones de bytes) de memoria virtual. La unidad de segmentación suministra 4 niveles de protección. Maneja los modos de operación, de modo de direccionamiento real (modo real), y modo de direccionamiento virtual protegido (modo protegido).

El modo real es requerido para que después pueda instalarse el modo protegido, este proporciona el acceso al sofisticado manejador de memoria, a la paginación y capacidades privilegiadas del μ .

Para facilitar el alto desarrollo del diseño de "Hardware"; el bus de la interfase del μ FB0486 ofrece direccionamiento "Pipeline", bus de datos dinámico y habilitación de señales de byte directas para cada Byte del bus de datos.

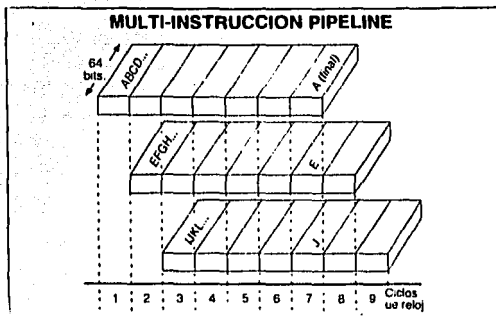


Fig. III.1.- El empalme de multi-instrucciones Pipeline del μ P80486.

MICROPROCESADOR 486SL DE INTEL

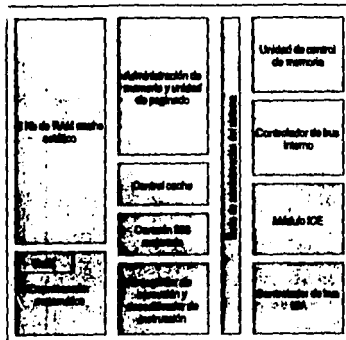


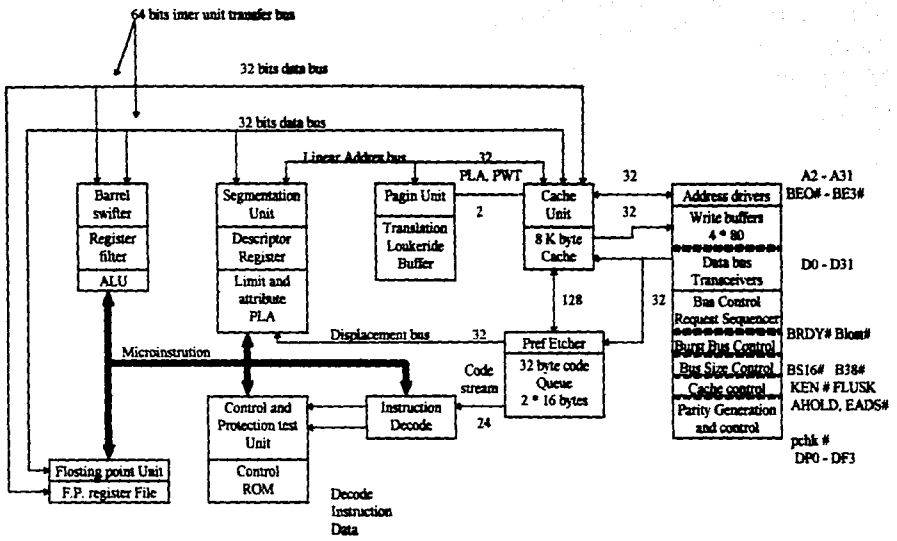
Fig. 111.2.- uP00486SL de Intel, para sistemas Notebook.

FALLA DE ORIGEN



Fig. III.3.- Diversificación del sistemas y UPC's, basados en el μ P80486.

FIG. 111.4.- Arquitectura del i486 de Intel.



486 MICROPROCESOR DIPLELINED 32 BIT MICROARCHITECTURE

MICROARQUITECTURA

III.3.- REGISTROS INTERNOS DEL μ P80486.

El μ P80486 tiene los registros agrupados en las siguientes categorías:

- 1.- Registros de propósito general.
- 2.- Registros de segmento.
- 3.- Instrucciones de puntero y banderas.
- 4.- Registros de control.
- 5.- Registros del sistema de direcciones.
- 6.- Registros depuradores.
- 7.- Registros de prueba.

Todos los registros anteriores son un "superconjunto" de los registros manejados anteriormente, por todos los μ P's precedentes de Intel.

La arquitectura base además, incluye seis segmentos que se pueden acceder directamente, cada uno de 4 Gigabytes, los segmentos son seleccionados colocando los valores adecuados en el registro de segmentos del μ P80486, varios valores seleccionados pueden ser cargados cuando un programa se está ejecutando si así se desea.

III.4.- JUEGO DE INSTRUCCIONES DEL μ P80486.

El juego de instrucciones está dividido en nueve categorías de operación:

- 1.- Transferencia de datos.
- 2.- Aritmética.
- 3.- Rotación.
- 4.- Manipulación de cadenas.
- 5.- Manipulación de bit.
- 6.- Transferencias de control.
- 7.- Soporte de lenguajes de alto nivel.
- 8.- Soporte de sistemas operativos.
- 9.- Control de procesos.

Todas las instrucciones operan con 0, 1, 2 ó 3 operandos, donde un operando reside en un registro, dentro de la misma instrucción o en memoria; la mayoría de las operaciones e instrucciones cero toman únicamente un byte, una instrucción de operando generalmente tiene una longitud de 2 bytes, el promedio de la longitud de las instrucciones es de 3,2 bytes.

El uso de dos operandos permite los siguientes tipos de instrucciones comunes:

- 1.- Registro a registro.
- 2.- Memoria a registro.
- 3.- Inmediato a registro.
- 4.- Registro a memoria.
- 5.- Inmediato a memoria.

Los operandos pueden ser de 8, 16 o 32 bits de largo, como regla general, cuando se ejecuta un código de escritura (código de 32 bits), los operandos son de 8 ó 32 bits.

A continuación se presenta el juego completo de instrucciones del μ P80486 en función de los siguientes parámetros y funciones:

- 1.- Transferencia de Datos.
 - a). De propósito general.
 - b). De conversión.
 - c). De Entrada/Salida.
 - d). De direccionamiento.
 - e). De manipulación de banderas ó interrupciones.

2.- De Instrucciones Aritméticas.

- a). De operación suma.
- b). De operación resta.
- c). De operación multiplicación.
- d). De operación división.

3.- De Instrucciones de Comparación.

4.- De operaciones Lógicas.

- a). De operaciones lógicas.
- b). De cambio.
- c). De rotación.

5.- Instrucciones de Manipulación de Bits.

- a). Instrucciones de bit simple.
- b). Instrucciones de comparación de bit.

6.- Instrucciones de Control de Programa.

- a). Transferencias condicionales.
- b). Transferencias no condicionales.
- c). Control de iteraciones.
- d). Interrupciones.

7.- Instrucciones de Lenguajes de Alto Nivel.

8.- Instrucciones de Modo de Protección.

9.- Instrucciones de Control del μP .

También a continuación, se presenta el juego (SET) de instrucciones del coprocesador matemático 80487; el cual le confiere al μP , la ventaja de encargarse de las operaciones matemáticas, lo cual permite que el tiempo de ejecución global sea menor. Trayendo una eficiencia superior en su desempeño; pero se debe de tener presente, que existen versiones del $\mu P80486$ que ya vienen con el coprocesador matemático incluido en el mismo encapsulado del $\mu P80486$, este modelo del 486 es el $\mu P80486DX$. Mas lo anterior no quiere decir que el 80487 no tenga su propio juego de instrucciones, por lo que se mostrará despues de analizar el del propio $\mu P80486$.

Table 2-2a. Data Transfer	
GENERAL PURPOSE	
MOV	Move operand
PUSH	Push operand onto stack
POP	Pop operand off stack
PUSHA	Push all registers on stack
POPA	Pop all registers off stack
XCHG	Exchange Operand, Register
XLAT	Translate
CONVERSION	
MOVZX	Move byte or Word, Dword, with zero extension
MOVBX	Move byte or Word, Dword, sign extended
CBW	Convert byte to Word, or Word to Dword
CWD	Convert Word to DWORD
CQDE	Convert Word to DWORD extended
CDQ	Convert DWORD to QWORD
INPUT/OUTPUT	
IN	Input operand from I/O space
OUT	Output operand to I/O space
ADDRESS OBJECT	
LEA	Load effective address
LDS	Load pointer into D segment register
LES	Load pointer into E segment register
LFS	Load pointer into F segment register
LGS	Load pointer into G segment register
LSS	Load pointer into S (Stack) segment register
FLAG MANIPULATION	
LAHF	Load A register from Flags
SAHF	Store A register in Flags
PUSHF	Push flags onto stack
POPF	Pop flags off stack
PUSHFD	Push EFlags onto stack
POPFD	Pop EFlags off stack
CLC	Clear Carry Flag
CLD	Clear Direction Flag
CMC	Complement Carry Flag
STC	Set Carry Flag
STD	Set Direction Flag

Table 2-2b. Arithmetic Instructions	
ADDITION	
ADD	Add operands
ADC	Add with carry
INC	Increment operand by 1
AAA	ASCII adjust for addition
DAA	Decimal adjust for addition
SUBTRACTION	
SUB	Subtract operands
SBB	Subtract with borrow
DEC	Decrement operand by 1
NEG	Negate operand
CMP	Compare operands
DAS	Decimal adjust for subtraction
AAS	ASCII Adjust for subtraction
MULTIPLICATION	
MUL	Multiply Double/Single Precision
IMUL	Integer multiply
AAM	ASCII adjust after multiply
DIVISION	
DIV	Divide unsigned
IDIV	Integer Divide
AAD	ASCII adjust before division

Table 2-2c. String Instructions	
MOVB	Move byte or Word, Dword string
INS	Input string from I/O space
OUTS	Output string to I/O space
CMPS	Compare byte or Word, Dword string
SCAS	Scan Byte or Word, Dword string
LODS	Load byte or Word, Dword string
STOS	Store byte or Word, Dword string
REP	Repeat
REPE/ REPZ	Repeat while equal/zero
REPNE/ REPNZ	Repeat while not equal/not zero

Table 2-2d. Logical Instructions	
LOGICALS	
NOT	"NOT" operands
AND	"AND" operands
OR	"Inclusive OR" operands
XOR	"Exclusive OR" operands
TEST	"Test" operands

Fig. 111.5.- Juego de instrucciones del i486.

Table 2-2d. Logical Instructions (Continued)

SHIFTS	
SHL/SHR	Shift logical left or right
SAL/SAR	Shift arithmetic left or right
SHLD/SHRD	Double shift left or right
ROTATES	
ROL/ROR	Rotate left/right
RCL/RCR	Rotate through carry left/right

Table 2-2e. Bit Manipulation Instructions

SINGLE BIT INSTRUCTIONS	
BT	Bit Test
BTS	Bit Test and Set
BTR	Bit Test and Reset
BTC	Bit Test and Complement
BSF	Bit Scan Forward
BSR	Bit Scan Reverse
BIT STRING INSTRUCTIONS	
IBTS	Insert Bit String
KBTS	Extract Bit String

Table 2-2f. Program Control Instructions

CONDITIONAL TRANSFERS	
BETCC	Set byte equal to condition code
JA/JNBE	Jump if above/not below nor equal
JAE/JNB	Jump if above or equal/not below
JB/JNAE	Jump if below/not above nor equal
JBE/JNA	Jump if below or equal/not above
JC	Jump if carry
JE/JZ	Jump if equal/zero
JG/JNLE	Jump if greater/not less nor equal
JGE/JNL	Jump if greater or equal/not less
JL/JNGE	Jump if less/not greater nor equal
JLE/JNG	Jump if less or equal/not greater
JNC	Jump if not carry
JNE/JNZ	Jump if not equal/not zero
JNO	Jump if not overflow
JNP/JPO	Jump if not parity/parity odd
JNS	Jump if not sign
JO	Jump if overflow
JP/JPE	Jump if parity/parity even
JS	Jump if sign

Table 2-2g. Program Control Instructions (Continued)

UNCONDITIONAL TRANSFERS	
CALL	Call procedure/task
RET	Return from procedure
JMP	Jump
ITERATION CONTROLS	
LOOP	Loop
LOOPE/LOOPZ	Loop if equal/zero
LOOPNE/LOOPNZ	Loop if not equal/not zero
JCXZ	JUMP if register CX = 0
INTERRUPTS	
INT	Interrupt
INTO	Interrupt if overflow
IRET	Return from Interrupt/Task
CLI	Clear Interrupt Enable
STI	Set Interrupt Enable

Table 2-2h. High Level Language Instructions

BOUND	Check Array Bounds
ENTER	Setup Parameter Block for Entering Procedure
LEAVE	Leave Procedure

Table 2-2i. Protection Model

BGDT	Store Global Descriptor Table
SIDT	Store Interrupt Descriptor Table
STR	Store Task Register
SLDT	Store Local Descriptor Table
LGDT	Load Global Descriptor Table
LIDT	Load Interrupt Descriptor Table
LTR	Load Task Register
LLDT	Load Local Descriptor Table
ARPL	Adjust Requested Privilege Level
LAR	Load Access Rights
LSL	Load Segment Limit
VERR/VERW	Verify Segment for Reading or Writing
LMSW	Load Machine Status Word (lower 16 bits of CR0)
SMSW	Store Machine Status Word

Table 2-2j. Processor Control Instructions

HLT	Halt
WAIT	Wait until BUSY# negated
ESC	Escape
LOCK	Lock Bus

Fig. III.6.- Juena de instrucciones del #PB0486 (Continuación).

Instruction	Encoding			Clock Count Range			
	Byte 0	Byte 1	Optional Bytes 2-3	22-28 Real	22-31 Integer	32-38 Real	32-39 Integer
DATA MOVEMENT							
LD = Load							
Integer/real memory to BT(0)	ESC MF 1	MOD 000 R/M	SB/DSP	20	45-52	25	51-56
Long Integer memory to BT(0)	ESC 111	MOD 101 R/M	SB/DSP		56-67		
Extended real memory to BT(0)	ESC 011	MOD 101 R/M	SB/DSP		44		
B/C/D memory to BT(0)	ESC 111	MOD 100 R/M	SB/DSP		206-276		
BT(0) to BT(0)	ESC 001	11000 BT(0)			14		
PUT = Store							
BT(0) to Integer/real memory	ESC MF 1	MOD 010 R/M	SB/DSP	44	79-93	48	82-96
BT(0) to BT(0)	ESC 101	11010 BT(0)			11		
PSTP = Store and Pop							
BT(0) to Integer/real memory	ESC MF 1	MOD 011 R/M	SB/DSP	44	79-93	48	82-96
BT(0) to long Integer memory	ESC 111	MOD 111 R/M	SB/DSP		80-97		
BT(0) to extended real	ESC 011	MOD 111 R/M	SB/DSP		83		
BT(0) to B/C/D memory	ESC 111	MOD 110 R/M	SB/DSP		812-834		
BT(0) to BT(0)	ESC 101	11001 BT(0)			12		
PXCH = Exchange							
BT(0) and BT(0)	ESC 001	11001 BT(0)			18		
COMPARISON							
FCOM = Compare							
Integer/real memory to BT(0)	ESC MF 0	MOD 010 R/M	SB/DSP	26	86-93	21	71-78
BT(0) to BT(0)	ESC 000	11010 BT(0)			84		
FCOMP = Compare and pop							
Integer/real memory to BT	ESC MF 0	MOD 011 R/M	SB/DSP	26	86-93	21	71-78
BT(0) to BT(0)	ESC 000	11011 BT(0)			26		
PCOMP = Compare and pop twice							
BT(1) to BT(0)	ESC 110	1101 1001			28		
BT(1) to BT(0)	ESC 001	1110 0100			28		
FCOUP = Unordered compare and pop							
BT(1) to BT(0)	ESC 101	11101 BT(0)			28		
FCOUP = Unordered compare and pop twice							
BT(1) to BT(0)	ESC 010	1110 1001			28		
FXAM = Examine BT(0)							
	ESC 001	11100101			20-28		
CONSTANTS							
PLDZ = Load + 0.0 into BT(0)	ESC 001	1110 1110			20		
PLD1 = Load + 1.0 into BT(0)	ESC 001	1110 1000			24		
PLDP1 = Load pi into BT(0)	ESC 001	1110 1011			40		
PLDLET = Load log ₂ (10) into BT(0)	ESC 001	1110 1001			40		

Fig. 111.7. - Juego de instrucciones para el Coprocesador 00487.

Instruction	Encoding			Clock Count Range			
	Byte 0	Byte 1	Optional Bytes 2-3	23-Bit Real	23-Bit Integer	34-Bit Real	10-Bit Integer
CONSTANTS (Continued)							
PLBLSE = Load log _e (e) into ST(0)	ESC 001	1110 1010			40		
PLBLSE = Load log _e (2) into ST(0)	ESC 001	1110 1100			41		
PLBLSE = Load log _e (7) into ST(0)	ESC 001	1110 1101			41		
ARITHMETIC							
FADD = Add							
Integer/real memory with ST(0)	ESC MF 0	MOD 000 R/M	S8/D8P	24-32	27-72	28-37	71-88
ST(0) and ST(0)	ESC d P 0	11000 ST(0)			23-31 ^a		
FSUB = Subtract							
Integer/real memory with ST(0)	ESC MF 0	MOD 10 R R/M	S8/D8P	24-32	27-52	28-36	71-82 ^c
ST(0) and ST(0)	ESC d P 0	1110 R R/M			28-34 ^d		
FMUL = Multiply							
Integer/real memory with ST(0)	ESC MF 0	MOD 001 R/M	S8/D8P	27-36	31-52	32-37	76-87
ST(0) and ST(0)	ESC d P 0	1100 1 R/M			28-37 ^e		
FDIV = Divide							
Integer/real memory with ST(0)	ESC MF 0	MOD 11 R R/M	S8/D8P	30	120-127 ^f	34	100-1000
ST(0) and ST(0)	ESC d P 0	1111 R R/M			30 ^g		
FRDINT = Round real	ESC 001	1111 1010			122-126		
FRSCALE = Scale ST(0) by ST(1)	ESC 001	1111 1101			37-38		
FRREM = Partial remainder	ESC 001	1111 1000			74-166		
ROUND = Round ST(0) to integer							
ESC 001	1111 1101				65-100		
EXTRACT = Extract components of ST(0)							
ESC 001	1111 0100				70-78		
FABS = Absolute value of ST(0)							
ESC 001	1110 0001				22		
FNCH = Change sign of ST(0)							
ESC 001	1110 0000				24-25		

NOTES:

- Add 3 clocks to the range when $d = 1$.
- Add 1 clock to each range when $R = 1$.
- Add 3 clocks to the range when $d = 0$.
- typical = 52 (When $d = 0$, 46-54, typical = 48).
- Add 1 clock to the range when $R = 1$.
- 135-141 when $R = 1$.
- Add 3 clocks to the range when $d = 1$.
- $0 \leq ST(0) \leq +\infty$.

Fig. III.0.- Juego de instrucciones del Coprocesador 60487 (Continuación).

Instruction	Encoding			Clock Count Range
	Byte 0	Byte 1	Optional Bytes 2-3	
TRANSCENDENTAL				
POOSP = Cosine of ST(0)	ESC 001	1111 1111		183-278
PPATAN = Partial tangent of ST(0)	ESC 001	1111 0010		191-487
PPATAN = Partial arctangent	ESC 001	1111 0011		314-487
PPATH = sine of ST(0) (1)	ESC 001	1111 1110		123-771
PPATHOP = sine and cosine of ST(0)	ESC 001	1111 1011		194-788
PZEXP = $2^{ST(0)} - 1$	ESC 001	1111 0000		211-476
PYLSX = $ST(1) * \log_2(ST(0))$	ESC 001	1111 0001		120-523
PYLSXP = $ST(1) * \log_2(ST(0) + 1.0)$	ESC 001	1111 1001		257-547
PROCESSOR CONTROL				
PRIT = Initialize MPX	ESC 011	1110 0011		33
PSTRWAZ = Store status word	ESC 111	1110 0000		19
PLDCW = Load control word	ESC 001	MOD 101 R/M	SB/DSP	18
PSTCW = Store control word	ESC 101	MOD 111 R/M	SB/DSP	19
PSTRW = Store status word	ESC 101	MOD 111 R/M	SB/DSP	15
PCLEX = Clear exception	ESC 011	1110 0010		11
PSTENV = Store environment	ESC 001	MOD 110 R/M	SB/DSP	103-104
PLDENV = Load environment	ESC 001	MOD 100 R/M	SB/DSP	71
PSAVE = Save state	ESC 101	MOD 110 R/M	SB/DSP	275-376
PRSTOR = Restore state	ESC 101	MOD 100 R/M	SB/DSP	308
PIWCSTP = Increment stack pointer	ESC 001	1111 0111		21
PDCWSTP = Decrement stack pointer	ESC 001	1111 0110		21
PPFREE = Free ST(0)	ESC 101	1100 0101		18
FNOP = No operations	ESC 001	1101 0000		12

NOTES:

1 These timings hold for operands in the range $|x| < w/4$. For operands not in this range, up to 76 additional clocks may be needed to reduce the operand

1. $0 \leq |ST(0)| < 2^5$

1. $-0.5 \leq ST(0) \leq 0.5$

1. $m \leq ST(0) < \infty$, $-\infty < ST(1) < +\infty$

1. $n \leq |ST(0)| < (2 - \text{SORT}(2))/2$, $-\infty < ST(1) < +\infty$

Fig. III.9.- Juego de instrucciones del Coprocesador 80487 (Continuación).

COMPUTADORAS DE ESCRITORIO 486DX A 33-MHz

Considere la actualización después de chequear el soporte en los procesadores, el número de ranuras de expansión y el máximo en RAM

COMPANIA MODELO	Logic Research Inc. Rev. 224 CT Model 34200W	Altec ATive D133	Asus Ace 486/33	Bi Link B1 433	Compaq Compaq 486DX/33	Compaq Desligné/33 Model 210
CHIPS						
Procesadores soportados (procesador del chip)	486SX/25, 486DX/33 y 50, 486DX/266	486DX/33, 486DX/266	486DX/33, 486DX/50, 486DX/266	486DX/33, 486DX/266	486DX/33	386DX/33, 486SX/25, 486DX/33
Uso de cache
Posibilidad de actualizar el CPU
Soporta a procesador externo
Marca de las chips en la tarjeta madre	Western 4187	Western 4187	Western 4187	Western 4187	Western 4187	Ninguno
Marca del BIOS	Phonix	UMC AMI	UMC AMI	UMC AMI	Microcom Phoenix	Compaq
MEMORIA						
RAM estándar (Mb)	8	4	4	4	4	4
RAM máximo (Mb)	64	32	32	32	64	32
Número de bancos	Ninguno	Ninguno	Ninguno	Ninguno	Ninguno	Ninguno
Cache estándar (Kb)	0	64	64	64	256	64
Cache máximo (Kb)	0	256	256	256	256	64
SISTEMAS DE ALMACENAMIENTO						
Configuración estándar de disco flexible						
Disco y sus tipos	1.44-Mb 3 1/2"	1.2-Mb 5 1/4", 1.44-Mb 3 1/2"	1.2-Mb 5 1/4", 1.44-Mb 3 1/2"	1.2-Mb 5 1/4", 1.44-Mb 3 1/2"	1.2-Mb 5 1/4", 1.44-Mb 3 1/2"	1.44-Mb 3 1/2"
CONFIGURACIÓN DEL DISCO DURO						
Capacidad estándar (Mb)	Variable	213	130	130	130	210
Marca y modelo	Variable	Maxtor XT7213AT	Maxtor 7120AT	Compaq CP30104H	Lanstar 320 Mb	Compaq 210
Interfase	IDE	IDE o SCSI	IDE (SCSI OPT.)	IDE (SCSI opt.)	SCSI	IDE
Incluye controlador de cache
Número de drive bay internas y externas	12	6	6	6	6	3
Plano de la capacidad de disco duro disponible	120Mb a 535Mb	213 Mb a 2 Gb	80 Mb a 1 Gb	120Mb a 530Mb	120Mb a 320Mb	60Mb a 210Mb
MONITOR						
Marca	ALR	CTX 5468N	Viewsonic 6c	TVM 4A	AAMazing C	Compaq C
Color (C Monocromático M)	C	C	C	C	C	C
Resolución máxima	1024x768	1024x768 (sin interfase)	1024x768	1024x768 (sin interfase)	1024x768	1024x768
Resolución de voltaje dual (115V/60 Hz-220V/50 Hz)	1024x768	1024x768 (sin interfase)	1024x768	1024x768 (sin interfase)	1024x768	1024x768
RANURAS DE EXPANSIÓN						
Número de ranuras ISA	10	8	8	8	8	3
Número de ranuras EISA	Ninguno	Ninguno	Ninguno	Ninguno	Ninguno	Ninguno
Ranuras propietarias	2 de 32 bits	Uno de 32 bits video local del bus	Uno de 32 bits video local del bus	Uno de 32 bits video local del bus	Ninguno	Ninguno
INTERFASES ESTÁNDAR						
Número de puertos seriales	2	2	2	2	2	1
Número de puertos paralelos para impresora	1	1	1	1	1	1
Resolución del adaptador de video	SVGA/1024x768	SVGA/1280x1024	SVGA/1280x1024	SVGA/1280x1024	SVGA/1024x768	SVGA/1024x768
VRAM estándar (Mb)	1	1	1	1	1	0.5
Marca del adaptador de video	Western Digital	Ochid Fahrenheit	Diamond Syath	ALR	Trend	Olivion 10241
Mouse incluido
Mouse serial (S) Puerto de Mouse (MP)	MP	S	S	S	S	MP
Puerto para joystick incluido
ADMINISTRADOR DE ENERGIA						
Watts	300	230	250	300	250	145
Voltaje dual (115/60 Hz-220/50 Hz)
COMPATIBILIDAD						
MS-DOS 5.0
Windows 3.1
OS/2
NetWare 386
Novell certificado
ISO Unix
MPX
Una interfaz
Otros sistemas operativos	Microport Unix	Ninguno	AT&T Unix, Xenix, pc/MOS	Ninguno	DOS concurrente	Ninguno
SOFTWARE INCLUIDO						
MS-DOS 5.0
Windows 3.1
Otros	Ninguno	Ninguno	Ninguno	Ninguno	Ninguno	Utilities
GARANTÍA						
	1 año en partes y mano de obra	1 año en partes y mano de obra	30 días de devolución de dinero, 13 meses en partes y mano de obra	1 año en partes y mano de obra	2 años en partes y mano de obra 1 año de servicio	1 año en partes y mano de obra 1 año de servicio
Soporte técnico
LISTA DE PRECIOS						
Configuración estándar	3,195 dls	2,850 dls	2,080 dls	1,685 dls	3,295 dls	2,540 dls
La unidad revisada en este artículo	2,990 dls	2,850 dls	2,648 dls	2,596 dls	4,595 dls	3,597 dls
Opciones para la unidad revisada	512 Kb VRAM, controlador de cache IDE	Ninguno	Controlador de cache, monitor SVGA Diamond, expansión a 8 Mb en RAM, 256-Kb de cache	Expansión a 8 Mb en RAM, 210 Mb en disco duro, adaptador SVGA, mouse y cable Windows 3.1, gabinete tipo torre	4 Mb RAM, 330 Mb en disco duro, controlador de cache SCSI, monitor a color SVGA, 256-Kb de cache, adaptador SVGA	4 Mb RAM, 210 Mb en disco duro, monitor Compaq

Fig. 11.16.- Estudio comparativo entre 486DX a 33MHz de diversos fabricantes.

COMPañA MODELO	Gateway 2000 486X/33	Inlight Distribución Ledsar Computer Insight 486-33 Cache	Ledsar Computer 486 LB Data Master	tec. Technologies PowerMac 486(33)	Northgate Computer	Packard Bell PB 486X/33
CHIPS						
Procesadores soportados (procesador del chip)	486DX/33	486DX/33 486DX/25	486SX/20 y 25, 486DX/25 31 y 50, 486DX/33 y 66	486DX/33 486DX/20 y 66	486SX y 25, 486DX/25 31 y 33, 486DX/33 y 66	486DX/33, 486DX/25
Uso de cache	"	"	"	"	"	"
Posibilidad de actualizar el CPU	"	"	"	"	" (2F socket)	"
Soporte a coprocesador en float	Webb 4187	Webb 4187	Webb 4187	Ninguno	Webb 4187	Ninguno
Manejo de los chips de la tarjeta madre	Intel	CPTI	UMAC	Ninguno	CPTI	Intel
Manejo del BIOS	Phoenix	AMI	AMI	Phoenix	Northgate	Phoenix
MEMORIA						
RAM estándar (Mb)	4	4	4	4	4	4
RAM máximo (Mb)	32	32	32	32	32	30
Número de bancos	Ninguno	Ninguno	Ninguno	Ninguno	Ninguno	Ninguno
Cache estándar (Mb)	Ninguno	64	64	64	64	256
Cache máximo (Mb)	Ninguno	256	256	128	256	256
SISTEMAS DE ALMACENAMIENTO						
Configuración estándar de disco flexible	1.2 Mb 5 1/4"	1.2 Mb 5 1/4"	1.2 Mb 5 1/4"	1.2 Mb 5 1/4"	1.2 Mb 5 1/4"	1.2 Mb 5 1/4"
Disquete y sus tipos	1.44 Mb 3 1/2"	1.44 Mb 3 1/2"	1.44 Mb 3 1/2"	1.44 Mb 3 1/2"	1.44 Mb 3 1/2"	1.44 Mb 3 1/2"
CONFIGURACIÓN DEL DISCO DURO						
Capacidad estándar (Mb)	200	213	120	120	240	130
Marca y modelo	Western Digital Caviar	Western Digital	Maxtor XT7120A	Quantum Prodrive	Seagate IDE	Seagate ST3144A
Interfaz	IDE (SCSI op)	IDE	IDE (SCSI op)	IDE	IDE	IDE
Soporta controlador de cache	"	"	"	"	"	"
Número de chips de puertos internos y externos	2	8	4	4	5	4
Manejo de la capacidad de disco duro disponible	80 Mb a 1.2 Gb	40 Mb a 4 Gb	42 Mb a 1.2 Gb	120 Mb y 240 Mb	40 Mb a 15 Gb	130 Mb a 400 Mb
CONTROL						
Manejo	CrysalScan	Viewsonic o TVM	StarView 2000	Ninguno	Panasonic	Packard Bell
Color (C) Monocromático (M)	C	C	C	Ninguno	C	C
Resolución máxima	1024x768	1024x768	1024x768	Ninguno	1024x768	1024x768
Posibilidad de voltaje dual (115/200VHz)	"	No, Viewsonic, Si, TVM	"	"	"	"
RAMURAS DE EXPANSIÓN						
Número de ranuras ISA	8	8	6	4	7	4
Número de ranuras EISA	Ninguno	Ninguno	Ninguno	Ninguno	Ninguno	Ninguno
Ranuras propietarias	Ninguno	Ninguno	Dot ranuras de bus de 32 bits	Ninguno	Ninguno	Ninguno
INTERFASES ESTÁNDAR						
Número de puertos seriales	2	2	2	1	2	1
Número de puertos paralelos para impresora	1	1	1	1	1	1
Resolución del adaptador de vídeo	SVGA/1024x768	SVGA/1280x1024	SVGA/1280x1024	SVGA/1280x1024	SVGA/1024x768	SVGA/1024x768
VFAM estándar (Mb)	1	1	1	1	1	0.5
Marca del adaptador de vídeo	ATI Graphics Ultra	Diamond Speedstar	Leadtek local bus VGA	Tsang Labs ET-4000	STB Ergo	Oak Tech, OTI-077
Mouse incluido	"	"	"	"	"	"
Mouse serial (S) Puerto de Mouse (MP)	"	"	"	MP	"	"
Puerto para juegos incluidos	"	"	"	"	"	"
SUBMINISTRO DE ENERGÍA						
Watts	200	250	230	110	220	150
Voltaje dual (115/200VHz)	"	"	"	"	"	"
COMPATIBILIDAD						
MS-DOS 3.0	"	"	"	"	"	"
Windows 3.1	"	"	"	"	"	"
OS/2	"	"	"	"	"	"
NetWare 386	"	"	"	"	"	"
Mouse certificado	"	"	"	"	"	"
SCC Unix	"	"	"	"	"	"
MPX	"	"	"	"	"	"
Linux instructivo	"	"	"	"	"	"
Otros sistemas operativos	Ninguno	Ninguno	Ninguno	Ninguno	DR DOS 6.0	Ninguno
SOFTWARE INCLUIDO						
MS-DOS 3.0	"	"	"	"	"	"
Windows 3.1	"	"	"	"	"	"
Otros	Una de 7 aplicaciones a escoger	Ninguno	Ninguno	Ninguno	Diagnostic Utilities	Utilerías, tutoriales, Lotus 1-2-3 for Windows
GARANTÍA						
Soporte técnico	"	"	"	"	"	"
LISTA DE PRECIOS						
Configuración estándar	2,395 ds	1,999 ds	1,979 ds	2,099 ds	1,999	precio de DEM
La unidad revisada en este artículo	2,662 ds	3,099 ds	2,599 ds	3,099 ds	2,919	Aprox. 2,200 ds
Opciones para la unidad revisada	4 Mb RAM	4 Mb RAM memoria a color SVGA, 213 Mb en disco duro, controlador de cache en disco duro	256 Kb cache, 4 Mb RAM, 210 Mb en disco duro, memoria a color SVGA, teclado programable, 2 Mb de cache en RAM	240 Mb en disco duro, 4 Mb RAM	4 Mb RAM 240 Mb en disco duro, memoria a color SVGA teclado Omnivox	4 Mb RAM

Fig. 114.12.- Catálogo comparativo del *486/33* a 33 MHz de diversos fabricantes.

CAPITULO IV

APLICACIONES AL CONTROL DIGITAL.

IV.1.- Introducción.

El manejo automático de equipos para el hogar, la escuela y la industria es particularmente útil para tareas peligrosas, repetitivas, tediosas ó simples. En la industria se emplean máquinas que cargan, descargan, soldan, cortan ó moldean con el fin de conseguir precisión, seguridad, economía y productividad. El empleo de computadores integrados a máquinas que realizan tareas como lo hace un ser humano, fue provisto por diversos autores.

Los robots son computadores programables integrados en máquinas. Con frecuencia sustituyen la labor humana en tareas repetitivas específicas. Algunos dispositivos tienen incluso mecanismos antropomórfos, incluyendo algunos que reconoceríamos como brazos mecánicos, muñecas y manos. Un robot se define como un manipulador reprogramable y multifuncional diseñado para mover materiales peligrosos, partes, herramientas ó dispositivos especiales a través de movimientos variables programados para realizar diversas tareas. La capacidad de un robot es que sea programable, lo que permite utilizarlo en nuevas tareas.

Algunas aplicaciones de los robots, tienen requisitos para los cuales un computador digital es el método más apropiado de control de la célula de trabajo. Al hacer referencia a la utilización de un computador dedicado (generalmente un microcomputador ó un computador) en lugar de la computadora que se utiliza como la unidad de control del robot. En casos donde un computador es el controlador de la célula de trabajo, se debería utilizar ó en serie con un automatá programable ó como un sustituto de éste. La computadora podría efectuar otras funciones en la planta y así se prepararía para controlar a la célula del robot en un modo de operación de tiempo compartido. También la computadora probablemente formaría parte como un componente de una red de computadoras jerárquicas en la fábrica, conectadas en su parte final a los automatás programables y/o controladores de robots en la célula, y conectadas hacia arriba al siguiente nivel jerárquico en la planta.

Los autómatas programables son dispositivos especializados que se diseñan para comunicarse con procesos industriales. Se proporcionan con puertos de entrada/salida que pueden ser cableados directamente a los elementos de la planta. Esto es una ventaja sobre el computador digital, ya que se deben realizar disposiciones especiales para comunicar el computador a los equipos industriales en la célula. Sin embargo, el AP tiene ciertas limitaciones en procesamiento de datos y lenguaje de programación que dan al computador una ventaja de aplicaciones que necesitan estas capacidades. Algunos ejemplos de las clases de características de aplicaciones de robot que podrían tender a favorecer el uso de computadoras para el control de célula de trabajo incluirían las siguientes:

1.- Casos en los que existe alguna célula cuyas operaciones se deben coordinar y significantes cantidades de datos se deben comunicar entre ellos.

2.- Células en las cuales el problema de detección y recuperación de error constituye una parte importante de la codificación que se debe programar en la operación de la célula de trabajo.

3.- Cuando algunos productos diferentes se hacen sobre la misma línea de producción automatizada, las operaciones en las diferentes estaciones se tienen que coordinar y secuenciar adecuadamente. Los computadores estarían bien adecuados a las tareas de procesamiento de datos que se necesitan en este tipo de aplicación. En casos donde las líneas de producción se utilizan para operaciones de ensamblaje, los diversos tamaños y estilos de las piezas componentes se deben clasificar y adaptar al modelo particular que está siendo ensamblado en cada estación de trabajo respectiva, a lo largo de la línea.

4.- Situaciones en las cuales se requiere un alto nivel de planificación de la producción en control de inventarios en la operación de la célula. Otra vez, este tipo de función de procesamiento de datos podría requerir la utilización de una computadora además de o como un sustituto de un autómata programable.

Las diferencias entre los computadores digitales y los autómatas programables, son principalmente, diferencia en aplicación, más que diferencia en tecnología básica. El AP puede, de hecho, considerarse como una forma especializada de computador digital con características dedicadas para el control de entrada/salida de elementos industriales. Las tecnologías de los dos tipos de control son bastante similares.

IV.2.- Inteligencia Artificial y Sistemas Expertos.

La Inteligencia Artificial es la solución de problemas complejos con el apoyo del computador digital, mediante la aplicación de procesos que son análogos al proceso de razonamiento humano.

Sólo unos pocos de los acontecimientos que han ocurrido ó ocurrirán en el último cuarto de este siglo tendrán un efecto tan profundo y duradero sobre la vida humana, como lo es la creación de máquinas inteligentes. La introducción de computadoras y robots inteligentes y autónomos provocará un cambio fundamental en nuestra sociedad. Para entender su importancia es conveniente que se lleguen a entender dos puntos fundamentales. Lo primero es que prácticamente todos los usos y aplicaciones de las computadoras y de la automatización, en general a la industria, están estrechamente relacionados con los principios básicos en que se fundó la Revolución Industrial: Mas específicamente, el uso de las computadoras y de la automatización ha reemplazado a aquellos trabajadores que desempeñaban labores poco cualificadas y repetitivas. El segundo punto a tener en cuenta es que la introducción de una automatización inteligente, provocará una segunda revolución industrial. No obstante, en esta revolución, los trabajadores que serán reemplazados por las máquinas pertenecerán a un nivel medio, donde entrarían todos aquellos trabajos que exigen la toma de decisiones meditadas (pero, quizá, ninguna inventiva).

Es difícil especificar una fecha exacta del comienzo de lo que normalmente se denomina Inteligencia Artificial (IA).

Las primeras computadoras eran en realidad, máquinas que tenían que ser literalmente reconfiguradas en su totalidad, para resolver problemas diferentes. El almacenamiento de programas permitía a la computadora cambiarse rápida y fácilmente con sólo encargar un nuevo programa. Esta capacidad implica que una computadora podría ser capaz de cambiar su propia función, es decir, aprender o pensar!

El campo de la IA, se compone de varias áreas de estudio. De ellas se listan las mas importantes y son:

- 1.- Búsqueda (de soluciones).
- 2.- Sistemas Expertos.
- 3.- Procesamiento en lenguaje natural.
- 4.- Reconocimiento de modelos.
- 5.- Robótica.
- 6.- Aprendizaje de las máquinas.
- 7.- Lógica.
- 8.- Incertidumbre y "lógica difusa".

Algunas de las áreas representan aplicaciones finales, tales como los sistemas expertos, otras como el procesamiento del lenguaje natural y la búsqueda de soluciones, son bloques de la IA que se añaden a otros programas para llevar a cabo su realización.

Cuando se hace referencia a la IA, el término búsqueda se refiere a la búsqueda de soluciones a un problema. (No significa encontrar una información específica dentro de una base de datos). Los Sistemas Expertos, son el primer producto de la IA viable comercialmente. Un SE tiene dos atributos especiales y principales. Primero, le permite introducir información sobre un tema en una computadora. A esta información se le suele llamar base de conocimiento. En segundo lugar, le permite interrogar a esta base de conocimiento y luego actuar como si fuese un experto en la materia, que es en definitiva la razón de su nombre.

Para muchos investigadores en IA, el procesamiento del lenguaje natural (conocido como PLN), es uno de los fines principales que la IA debe alcanzar porque permite a la computadora la entrada del lenguaje humano de forma directa.

El mayor obstáculo para alcanzar esta meta es el tamaño y la complejidad de los lenguajes humanos. Además, tenemos el problema de que la computadora sea consciente de la información contextual que pueda aparecer en cualquier situación que no sea de las más simples.

El reconocimiento y relación de modelos es importante para varias aplicaciones, incluidas la robótica y el procesamiento de imágenes. Por ejemplo, cuando se da una imagen de TV digitalizada, ¿cómo puede determinar la computadora dónde termina un objeto y empieza otro, ¿ si un objeto está sobre otro ?

Al igual que el procesamiento del lenguaje natural, el reconocimiento y relación de modelos es necesario para que una computadora se interrelacione con el mundo humano.

Aplicado a la robótica, la IA ayuda a que una computadora controle el movimiento usando un razonamiento espacial. Para los robots industriales como los utilizados en el ensamblaje de automóviles, los problemas para la IA aparecen al tratar de suministrarles un movimiento natural ó preciso dentro de un conjunto de posiciones concretas. Los robots autónomos tienen mayores problemas para desenvolverse en un mundo humano, con sus obstáculos, sucesos inesperados y cambio de ambiente.

Una de las áreas más importantes de la IA es la del aprendizaje mecánico. Esta área trata de hacer que los programas aprendan de sus propios errores, en base a la observación y a la autoevaluación. El aprendizaje mecánico significa simplemente hacer que la computadora sea capaz de beneficiarse de su propia experiencia. De los muchos productos de la IA de importancia práctica, están aquellos que pueden usarse para estudiar la corrección lógica de un argumento aplicando unas reglas lógicas generales. En este contexto, la palabra argumento hace mención a las distintas afirmaciones correctadas de manera lógica para alcanzar un fin. Esto incluye pruebas matemáticas, lógica formal y lógica silogística o filosófica. La mayoría de las decisiones que se toman, están basadas en un conocimiento incorrecto. Por ejemplo; cuando se compra una casa, no se sabe que todas las cañerías funcionan correctamente, etc. La decisión de comprar se basa en la suposición de que hay una cierta probabilidad o posibilidad de que todo se encuentre en perfectas condiciones. El que una computadora pueda pensar de la misma manera implica el uso de la lógica incierta (es decir, la toma de decisiones basadas en una información incompleta o probable).

A continuación se enlistan los principios fundamentales en que se basa la IA, y estos son:

Principio 1.0.- Las técnicas de la IA intentan en forma explícita, trasladar el proceso de razonamiento hacia el programa.

Principio 1.1.- Un Sistema Experto (SE), se dedica a un problema de un área específica. No se intenta enfocar las capacidades humanas en todas las áreas.

Principio 1.2.- Habitualmente se espera de una persona un desempeño aceptable, pero no se le exige una solución óptima en todos los casos.

Principio 1.3.- Un SE busca una solución satisfactoria, tal que sea lo suficientemente buena para hacer el trabajo, aunque no sea la óptima.

Principio 1.4.- El nivel de exactitud y precisión que exige una solución satisfactoria se dictamina por el dominio del problema.

En lo referente a los Sistemas Expertos, se puede mencionar lo siguiente:

Los Sistemas Expertos (SE), se emplean para ejecutar una variedad muy complicada de tareas, que en el pasado solamente podrían llevarse a cabo por un número limitado de personas expertas intensamente entrenadas.

Un Sistema Experto (SE), es una aplicación informática que soluciona problemas complicados que de otra manera exigirían ampliamente la pericia humana. Para lograr esto, se simula el proceso de razonamiento humano mediante la aplicación específica de conocimientos e inferencias.

Internamente, un SE ideal se puede caracterizar como un sistema que comprende:

- Amplio conocimiento específico a partir del campo de interés.
- Aplicación de técnicas de búsqueda.
- Soporte para análisis heurístico.
- Habilidad para inferir nuevos conocimientos a partir de conocimientos ya existentes.
- Procesamiento de símbolos.
- Capacidad para explicar su propio razonamiento.

Los principios básicos en los que se basan los SE se enumeran a continuación:

Principio 1.0.- La potencia de un experto se debe más al conocimiento amplio del área específica que a la comprensión del desempeño genérico de un experto.

Principio 1.1.- La selección del esquema de representación del conocimiento es una de las decisiones más críticas en el diseño de un SE.

Principio 1.2.- El proceso de buscar los conocimientos apropiados y a partir de estos deducir nuevos conocimientos, constituye un elemento clave del procesamiento de un sistema experto.

Principio 1.3.- La selección del paradigma de inferencia considerando la explosión combinatoria, influye fuertemente en el desempeño global de un SE.

Principio 1.4.- En un SE ideal, el motor de inferencia nunca debería necesitar de modificaciones.

Principio 1.5.- La credibilidad que se le concede a un SE depende de la habilidad del SE para explicar su propio proceso de razonamiento.

IV.3.- Automatas Programables.

Se entiende por Controlador Lógico Programable (PLC) o Automata Programable; a toda máquina electrónica, diseñada para controlar en tiempo real y en medio industrial procesos secuenciales. Su manejo y programación puede ser realizada por personal eléctrico o electrónico sin conocimientos informáticos. Realiza funciones lógicas: Series, paralelos, temporizaciones, contajes y otras mas potentes como cálculos, regulaciones, etc.

El PLC, por sus especiales características de diseño, tiene un campo de aplicación muy extenso. La constante evolución del Hardware y Software, amplia continuamente este campo para poder satisfacer las necesidades que se detectan en el espectro de sus posibilidades reales.

Su utilización se da fundamentalmente en aquellas instalaciones en donde es necesario realizar procesos de maniobra, control, señalización, etc., por lo tanto, su aplicación abarca desde procesos de fabricación industrial de cualquier tipo de transformaciones industriales, control de instalaciones, etc.

Sus reducidas dimensiones, la extremada facilidad de su montaje, la posibilidad de almacenar los programas para su posterior y rápida utilización, la modificación o alteración de los mismos, etc., hace que su eficacia se aprecie fundamentalmente en procesos en que se producen necesidades tales como:

- Espacio reducido.
- Procesos de producción periódicamente cambiantes.
- Procesos secuenciales.
- Maquinaria de procesos variables.
- Instalaciones de procesos complejos y amplios.
- Verificación de programación centralizada de las partes del proceso.

A continuación de enumeran ejemplos de aplicación general, que pueden ser los siguientes:

1.- De maniobra de máquinas:

- a). Maquinaria industrial del mueble y madera.
- b). Maquinaria en procesos de grava, arena y cemento.
- c). Maquinaria en la industria del plástico.
- d). Máquinas-herramienta complejas.
- e). Maquinaria en procesos textiles y de confección.
- f). Maquinaria de ensamblaje.
- g). Máquinas de transferencia.

2.- Maniobra de instalaciones:

- a). Instalaciones de aire acondicionado, calefacción, etc.
- b). Instalaciones de seguridad.
- c). Instalaciones de frío industrial.
- d). Instalaciones de almacenamiento y trasvase de cereales.
- e). Instalaciones en plantas embotelladoras.
- f). Instalaciones en la industria de automoción.
- g). Instalaciones de tratamientos térmicos.
- h). Instalaciones de plantas depuradoras de residuos.
- i). Instalaciones de cerámica.

3.- Señalizaciones y control:

- a). Verificación de programas.
- b). Señalización del estado de procesos.

A continuación, se analizarán las ventajas e inconvenientes del PLC, y son:

Las condiciones favorables que presenta un PLC son las siguientes:

1.- Menor tiempo empleado en la elaboración de proyectos debido a que:

- a). No es necesario dibujar el esquema de contactos.
- b). No es necesario simplificar las ecuaciones lógicas, ya que, por lo general, la capacidad de almacenamiento del módulo de memoria es lo suficientemente grande.

c). La lista de materiales queda sensiblemente reducida, y al elaborar el presupuesto correspondiente eliminaremos parte del problema que supone el contar con diferentes proveedores, distintos plazos de entrega, etc.

2.- Posibilidad de introducir modificaciones sin cambiar el cableado, ni añadir aparatos.

3.- Mínimo espacio de ocupación.

4.- Menor coste de mano de obra de la instalación.

5.- Economía de mantenimiento. Además de aumentar la fiabilidad del sistema, al eliminar contactos móviles, los mismos Automatas Programables, pueden detectar e indicar averías.

6.- Posibilidad de gobernar varias máquinas con un mismo PLC.

7.- Menor tiempo para la puesta en funcionamiento del proceso al quedar reducido el tiempo de cableado.

8.- Si por alguna razón la máquina queda fuera de servicio, el PLC sigue siendo útil para otra máquina ó sistema de producción.

Los inconvenientes del PLC, se puede mencionar, en primer lugar, de que hace falta un programador, lo que obliga a adiestrar a uno de los técnicos en ese sentido.

Fero hay otro factor importante, como el costo inicial, que puede ser ó no un inconveniente, según las características del automatismo en cuestión. Dado que el PLC cubre ventajosamente un amplio espacio entre la lógica cableada y el μP , es preciso que el proyectista lo conozca tanto en su amplitud como en sus limitaciones. Por lo tanto, aunque el costo inicial debe ser tenido en cuenta a la hora de decidir por uno u otro sistema, conviene analizar todos los demás factores para asegurar una decisión acertada.

IV.3.1.- Estructura Externa.

La estructura externa o configuración externa de un Automata Programable (PLC); se refiere al aspecto físico exterior del mismo, bloques o elementos en que esta dividido, etc. Desde su nacimiento y hasta nuestros días han sido varias las estructuras y configuraciones que han salido al mercado condicionadas no solo por el fabricante del mismo, sino por la tendencia existente en el área al que perteneciese: Americana o Europea. Actualmente, son dos las estructuras más significativas que existen en el mercado:

a). Estructura compacta.- Este tipo de PLC, se distingue por presentar en un solo bloque todos sus elementos, esto es, fuente de alimentación, CPU, memorias, Entradas/Salidas, etc.

En cuanto a su unidad de programación, existen tres versiones: Unidad fija o enchufable directamente en el PLC; enchufable mediante cable y conector, o la posibilidad de ambas conexiones. Si la unidad de programación es sustituida por una PC, se encuentra en la posibilidad de que la conexión del mismo será mediante cable y conector. El montaje del PLC al armario que ha de contenerlo se realiza por cualquiera de los sistemas conocidos: Carril DIN, placa perforada, etc.

b). Estructura modular.- La estructura de este tipo de PLC se divide en módulos o partes del mismo que realizan funciones específicas. Aquí cabe hacer dos divisiones para distinguir entre las que se denominan americana y europea:

- Estructura americana.- Se caracteriza por separar las E/S del resto del PLC, de tal forma que en un bloque compacto están reunidas las CPU, memoria de usuario o de programa y fuente de alimentación, y separadamente las unidades de E/S en los bloques o tarjetas necesarios.

- Estructura europea.- Su característica principal es la de que existe un módulo para cada función: Fuente de poder, CPU, Entradas/Salidas, etc. La unidad de programación se une mediante cable y conector. La sujeción de los mismos se hace bien sobre carril DIN o placa perforada, bien sobre RACK, en donde va alojado el BUS externo de unión de los distintos módulos que lo componen.

IV.3.2.- Estructura ó Arquitectura Interna.

Los PLC se componen esencialmente de tres bloques, tal como lo presenta la fig. IV.1.

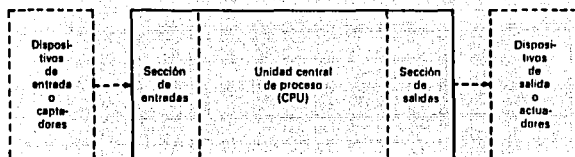


Fig. IV.1.- Autómata Programable básico (PLC).

Dentro del bloque de analisis anterior, se deben especificar los siguientes elementos que lo configuran:

a). La Sección de Entradas.- Mediante una interfase, adapta y codifica de forma comprensible por la UPC las señales procedentes de los dispositivos de entrada ó captadores, esto es, pulsadores, finales de carrera, sensores, etc.; también tiene una misión de protección de los circuitos electrónicos internos del PLC, realizando una separación eléctrica entre estos y los captadores.

b). La Unidad Central de Proceso (CPU).- Es la unidad de inteligencia del sistema, ya que mediante la interpretación de las instrucciones del programa de usuario y en función de los valores de las entradas, activa las salidas deseadas.

c). La Sección de Salidas.- Mediante la interfase trabaja de forma inversa a la de entradas, es decir, decodifica las señales procedentes de la UPC, las amplifica y manda con ellas los dispositivos de salida ó actuadores, como lámparas, relés, contactores, arrancadores, electroválvulas, etc., aquí también existen unas interfaces de adaptación a las salidas y de protección de circuitos internos.

Con las partes descritas, se puede decir que se tiene un PLC; pero para que sea operativo son necesarios otros elementos tales como:

- La unidad de alimentación.
- La unidad ó consola de programación.
- Los dispositivos periféricos.
- Interfases.

En la fig. IV.2, se han incluido de manera explícita todos estos elementos.

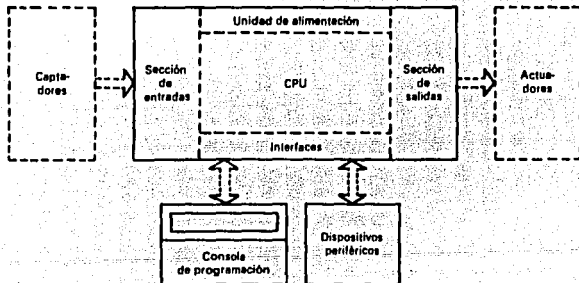


Fig. IV.2.- Autómata Programable (PLC), con sus periféricos y unidad de alimentación.

d). La Unidad de Alimentación.- Adapta la tensión de red de 127 V y 60 Hz (en América) ó de 220 V y 50 Hz (en Europa); a la de funcionamiento de los circuitos electrónicos internos del PLC, así como a los dispositivos de entrada: 24 V por ejemplo.

cc

e). La Unidad de Programación.- Se ha dicho que la UPC elabora las salidas en función de los estados de las entradas y de las instrucciones del programa de usuario, pero, ¿cómo accede el usuario al interior de la UPC para cargar en memoria su programa? La respuesta es mediante la unidad de programación. En los PLC más sencillos es un teclado con un "display" similar a una calculadora que cuando se quiere cargar un programa en la UPC se acopla a ésta mediante un cable y un conector, o bien mediante un enchufe directo a la UPC.

f). Periféricos ó Equipos Periféricos.- Son aquellos elementos auxiliares, físicamente independientes del PLC, que se unen al mismo para realizar su función específica y que amplían su campo de aplicación ó facilitan su uso. Como tales no intervienen directamente ni en la elaboración, ni en la ejecución del programa.

g). Interfases.- Son aquellos circuitos ó dispositivos electrónicos que permiten la conexión a la UPC de los elementos periféricos descritos.

**ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA**

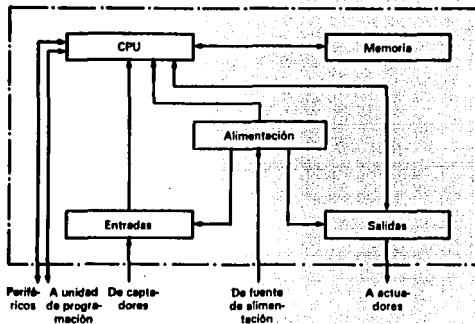
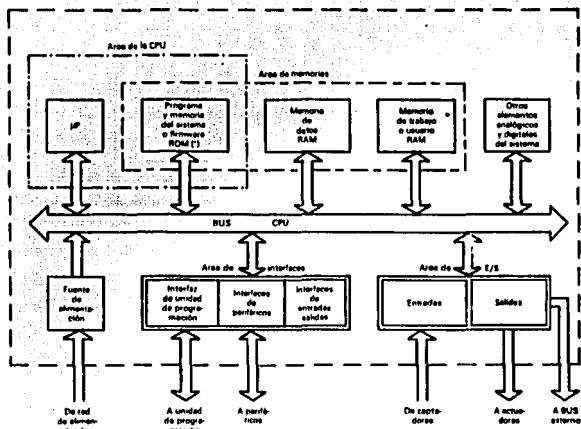


Fig. IV.3.- Esquema de bloques simplificado de interconexión de un PLC.



(*) El programa se encuentra en ROM y la memoria en RAM. Algunos fabricantes utilizan una única EPROM.

Fig. IV.4.- Arquitectura de un PLC.

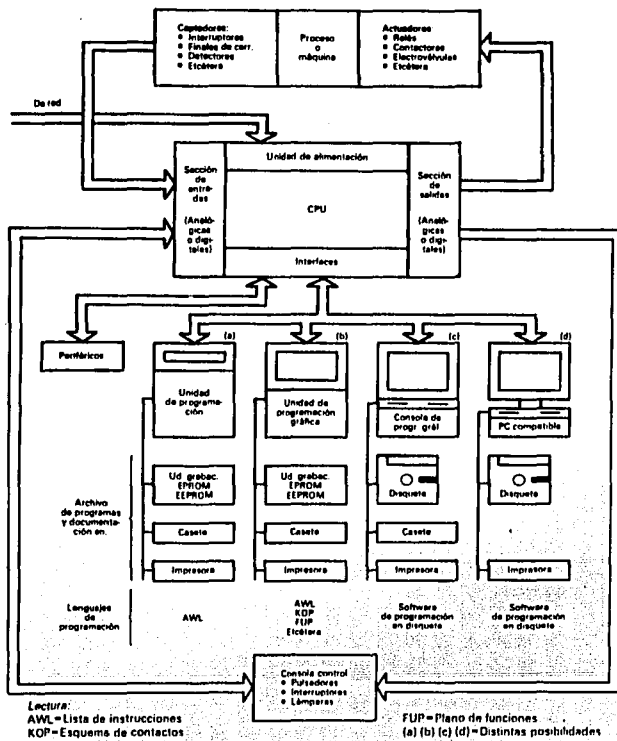


Fig. IV.5.- Estructura completa de un PLC y su entorno.

IV.4.- Robótica.

La robótica es una ciencia aplicada que ha sido considerada como una combinación de tecnología de las máquinas-herramienta y de la informática. Comprende campos tan aparentemente diferentes como son diseño de máquinas, teoría de control, mecatrónica, programación de computadoras, inteligencia artificial, factores humanos y teoría de la producción. El sector de investigación y desarrollo está procediendo en todas estas áreas para mejorar la forma en que los robots trabajan y "piensan". Es probable que los esfuerzos de investigación den lugar a futuros robots que hagan que las máquinas actuales parezcan bastante primitivas. Los avances en tecnología ampliarán la gama de las aplicaciones industriales de los robots.

Los campos técnicos anteriormente citados son muy interdependientes en la manera en que se utilizan en robótica. Para poder apreciar la tecnología de la robótica y su programación debe conocerse la forma en que los robots se aplican en la industria. Para comprender el empleo de sensores en robótica hay que estar familiarizado con la forma en que se programan los robots. Para comprender el uso de un efector final debe conocerse que una función fundamental de un robot es manipular piezas y herramientas.

Para describir la tecnología de un robot, se tiene que definir una diversidad de características técnicas relativas a la forma en que está construido el robot, y a la manera en que opera. Los robots trabajan con sensores, herramientas y pinzas, y deberán definirse esos términos. La programación del robot, se realiza de varias formas. Los robots se utilizan para ejecutar trabajos en la industria, de diversa índole y aplicación.

La anatomía del robot se refiere a la construcción física del cuerpo, brazo y muñeca de la máquina. La mayoría de los robots utilizados en las fábricas actuales están montados sobre una base que está sujeta al suelo. El cuerpo está unido a la base y el conjunto del brazo está unido al cuerpo. Al final del brazo está la muñeca. La muñeca está constituida por varios componentes que le permiten orientarse en una diversidad de posiciones. Los movimientos relativos entre los diversos componentes del cuerpo, brazo y muñeca son proporcionados por una serie de articulaciones.

Estos movimientos de las articulaciones suelen implicar deslizamientos o giros. El cuerpo, el brazo y el conjunto de la muñeca se denomina, a veces, el manipulador.

Unida a la muñeca del robot va una mano. El nombre técnico aplicado a la mano es "efector final". El efector final, no se considera como parte de la anatomía del robot.

Las articulaciones del cuerpo y del brazo del manipulador se emplean para situar el efector final y las articulaciones de la muñeca del manipulador se utilizan para orientar dicho efector final.

Los robots industriales están diseñados para realizar un trabajo productivo. El trabajo se realiza permitiendo que el robot desplace su cuerpo, brazo y muñeca mediante una serie de movimientos y posiciones. Unido a la muñeca está el efector final, que se utiliza por el robot para realizar una tarea específica. Los movimientos del robot pueden dividirse en dos categorías generales: movimientos de brazo y cuerpo, y movimientos de la muñeca. Los movimientos de articulaciones individuales asociados con estas dos categorías se denominan, a veces, por el término "grado de libertad", y un robot típico industrial, está dotado de cuatro a seis grados de libertad.

Los movimientos del robot se realizan por medio de articulaciones accionadas. Tres articulaciones suelen estar asociadas con la acción del brazo y del cuerpo, y dos a tres articulaciones se suelen emplear para accionar la muñeca.

Para la conexión de las diversas articulaciones del manipulador se emplean unos elementos rígidos denominados uniones. En cualquier cadena de uniones articuladas, se llama unión de entrada a aquella que está más próxima a la base en la cadena, la unión de salida es la que se encuentra con respecto a la entrada, las articulaciones intermedias en el diseño de robots industriales, suelen emplearse un movimiento relativo de los miembros contiguos, movimientos que son lineal o rotacional. Las articulaciones lineales implican un movimiento deslizante o de translación de los miembros de conexión. Este movimiento puede conseguirse de varias formas, y por ejemplo, mediante un pistón, un mecanismo de levas, o por el movimiento relativo a lo largo de un carril (ver lineal 1).

Las articulaciones del brazo y del cuerpo a las diseñadas para permitir al robot desplazar el efector final a una posición deseada dentro de los límites del tamaño del robot, y de los movimientos de las articulaciones. Para robots de configuración polar, típicos a de brazo articulado, los tres grados de libertad asociados con los movimientos del brazo y del cuerpo son:

1.- Transversal vertical.- Es la capacidad para desplazar la muñeca hacia arriba ó abajo para proporcionar la postura vertical deseada.

2.- Transversal radial.- Implica la extensión ó retracción (movimiento hacia adentro ó afuera) del brazo desde el centro vertical del robot.

3.- Transversal rotacional.- Es la rotación del brazo alrededor del eje vertical.

IV.4.1.- Cuatro Tipos de Controles de Robot.

Los robots industriales disponibles en el mercado pueden clasificarse en cuatro categorías, según sus sistemas de control. Las cuatro categorías son:

- 1.- Robot de secuencia limitada.
- 2.- Robots de reproducción con control punto a punto.
- 3.- Robot de reproducción con control recorrido continuo.
- 4.- Robots inteligentes.

De las cuatro categorías, los robots de secuencia limitada representan el control de nivel más bajo, y los robots inteligentes el más sofisticado.

Los robots de secuencia limitada no utilizan servocontrol para indicar las posiciones relativas de las articulaciones. En cambio, se controlan por el posicionamiento de los interruptores de fin de carrera y/o toques mecánicos para establecer los puntos finales de desplazamiento para cada una de sus articulaciones. El establecimiento de las posiciones y las secuencias de estos toques implica una puesta a punto mecánica del manipulador en lugar de una programación del robot en el sentido habitual del término. Con este método de control, las articulaciones individuales sólo pueden desplazarse a sus límites de desplazamientos extremos. Esto hace el costo de hacer severamente el número de puntos de puntos que puedan especificarse en un programa para estos robots.

La secuencia en la que se reproduce el ciclo de movimiento se define mediante un conmutador paso a paso, una placa de clavijas, "pegboard", u otro dispositivo de secuenciamento. Este dispositivo que constituye el controlador del robot, señala cada uno de los actuadores particulares para que operen en la sucesión adecuada. No suele existir ninguna realimentación asociada con un robot de secuencia limitada para indicar que se alcanza la posición deseada. Cualquiera de estos tres sistemas de impulsión puede utilizarse con este tipo de sistema de control; sin embargo, la impulsión neumática parece ser el tipo utilizado con mayor frecuencia. Los actuadores para este tipo de robot suelen implicar movimientos simples, tales como operaciones de "coger y tirar".

Los robots de reproducción utilizan una unidad de control más sofisticada, en la que una serie de posiciones ó movimientos son " enseñados " al robot, registrados en memoria y luego repetidos por el robot bajo su propio control. El término " reproducción " es descriptivo de este modo operativo general. El procedimiento de enseñar y registrar en memoria, se conoce como la programación del robot. Los robots de reproducción suelen tener alguna forma de servocontrol para asegurar que las posiciones conseguidas por el robot son las posiciones que se le " enseñaron ".

Los robots de reproducción pueden clasificarse en dos categorías: Robot punto a punto (PTP) y robot de trayectoria continua (CP). Los robots punto a punto son capaces de realizar ciclos de movimiento que consisten en una serie de localizaciones de puntos deseados y acciones afines. Al robot se le enseña cada punto, y estos puntos se registran en la unidad de control del robot. Durante la reproducción, el robot se controla para desplazarse desde un punto a otro en la secuencia adecuada. Los robots punto a punto no controlan la trayectoria tomada por el robot para pasar de un punto al siguiente. Si el programador quiere ejercer una cantidad limitada de control sobre la trayectoria seguida, debe realizarlo mediante la programación de una serie de puntos a lo largo de la trayectoria deseada. El control de la secuencia de posiciones es bastante apropiado para muchas clases de aplicaciones, incluyendo las máquinas de carga y descarga, y la soldadura por puntos.

Los robots de trayectoria continua son capaces de realizar ciclos de movimiento, en los que se controla la trayectoria seguida por el robot. Esto suele realizarse efectuando el desplazamiento del robot a través de una serie de puntos próximos, que describen la trayectoria deseada.

Los puntos individuales se definen por la unidad de control y no por el programador. El movimiento en línea recta es una forma común de control de trayectoria continua para los robots industriales. El programador especifica el punto inicial y el punto final de la trayectoria, y la unidad de control calcula la secuencia de puntos individuales que permiten al robot seguir una trayectoria de línea recta. Algunos robots tienen capacidad para seguir una trayectoria curva suave, definida por un programador que desplaza manualmente el brazo a través del ciclo de movimiento deseado.

Para conseguir un control de trayectoria continua más allá de una extensión limitada se exige que la unidad de control sea capaz de almacenar un gran número de posiciones de puntos individuales que definan la trayectoria curva compuesta. Actualmente, esto implica el empleo de una computadora digital (se suele utilizar un μP como unidad central de proceso para la computadora) como unidad de control del robot. El control C^1 se requiere para algunos tipos de aplicaciones industriales, tales como revestimiento por pulverización y soldadura por arco.

Los robots inteligentes constituyen una clase cada vez más numerosa de los robots industriales, y capacidad no sólo para reproducir un ciclo de movimiento programado, sino para interaccionar con su entorno de una manera que parece inteligente. Invariablemente, el controlador consiste en una computadora digital o dispositivo similar. Los robots inteligentes pueden modificar su ciclo programado en respuesta a las condiciones particulares que se produzcan en el lugar de trabajo, pueden tomar decisiones lógicas basadas en los datos del sensor recibidos desde la operación. Los robots de esta clase tienen capacidad para comunicarse, durante el ciclo de trabajo, con los operadores humanos o con sistemas basados en computadora. Los robots inteligentes se suelen programar utilizando un lenguaje similar al inglés, y un lenguaje simbólico no muy diferente a un lenguaje de programación de computadora. En realidad, las clases de aplicaciones que se realizan por robots inteligentes se basan en el empleo de un lenguaje de alto nivel para realizar las actividades complejas y no estructuradas que pueden ser ejecutadas por estos robots, operaciones típicas de los robots inteligentes son las tareas de montaje y las operaciones de soldadura por arco.

IV.4.2.- Control Coordinado de Fuerza y Posición.

Una característica del robot que está relacionada con esta exposición, es el control coordinado de fuerza y posición. Dicho control del manipulador del robot se refiere al desplazamiento del extremo de la muñeca en respuesta a una fuerza ó torsión que se ejerza sobre él. Un valor alto de esta característica significa que la muñeca se desplaza en una gran magnitud como respuesta a una fuerza relativamente pequeña. A veces se utiliza el término "elástico" para describir un robot con un alto valor de esta característica. Si tiene un valor bajo significa que el manipulador es relativamente rígido y no se desplaza en una magnitud significativa.

El control coordinado de fuerza y posición del manipulador de un robot es una característica direccional. Es decir, será mayor en determinadas direcciones que en otras, debido a la construcción mecánica del brazo.

Se trata de una característica importante puesto que reduce la precisión de movimiento del robot bajo carga. Si el robot está manipulando una carga pesada, el peso de la carga hará que se desvie el brazo del robot. Si el robot está presionando una herramienta contra una pieza de trabajo, la fuerza de reacción de la pieza puede producir una desviación del manipulador. Si la programación del robot para la situación final en su efector ha sido hecha en condición de descarga, si la exactitud de la posición es importante para la aplicación, cuando trabaje en condición de carga puede ver degradado su rendimiento debido precisamente a esa característica.

CAPITULO V.

CONCLUSIONES

A manera de conclusiones al presente trabajo de tesis se pueden mencionar las siguientes:

1.- Se han definido de forma específica las características que tienen los μP de 32 Bits; estas características corresponden a su arquitectura, modo de direccionamiento, teoría de operación, configuración interna, juego básico de instrucciones y su aplicación a algunos tipos característicos de computadores. Este análisis está enfocado a los μP 's 80386 y 80486, teniendo en consideración que son de los más utilizados y actualizados que existen en el mercado de μP 's.

Mención aparte correspondería al μP PENTIUM, que por tener algunos errores de origen (en especial el error de dividir), hace que en aplicaciones muy especiales, pudiera tener errores inherentes que afectan al sistema en el cual está instalado.

Otro μP importante es el P_0 , el nuevo elemento de Intel que está por entrar al mercado, y que pretende sustituir al PENTIUM con sus errores de origen. De este procesador se tiene poca información (estamos hablando de junio de 1995), por lo que no se incluye en este trabajo de investigación y aplicación de los μP de 32 Bits.

2.- La necesidad cada vez más prestante de aumentar la productividad y conseguir productos acabados de una calidad uniforme, está haciendo que la industria que cada vez hacia una automatización basada en computador. En el momento actual, la mayoría de las tareas de fabricación automatizada, se realizan mediante máquinas de uso especial diseñadas para realizar funciones predefinidas en un proceso de manufacturación. La inflexibilidad y generalmente el alto costo de estas máquinas, a menudo llamadas "sistemas de automatización duros", han llevado a un interés creciente en el uso de robots capaces de ejecutar una variedad de funciones de fabricación en un entorno de trabajo más flexible y a un menor costo de producción, al utilizar los modernos métodos del control digital, los robots.

FALLA DE ORIGEN

De lo anterior podemos concluir que, un robot industrial es un manipulador de uso general controlado por computadora que consiste en algunos elementos rígidos conectados en serie mediante articulaciones prismáticas ó de revolución. El final de la cadena está fijo a una base soporte, mientras el otro extremo está libre y equipado con una herramienta para manipular objetos ó realizar tareas de montaje. El movimiento de las articulaciones resulta en, ó produce, un movimiento relativo de los distintos elementos.

Mecanicamente, un robot se compone de un brazo y una muñeca más una herramienta. Se diseña para alcanzar una pieza de trabajo localizada dentro de su volumen de trabajo. El volumen de trabajo es la esfera de influencia de un robot cuyo brazo puede colocar el submontaje de la muñeca en cualquier punto dentro de la esfera. El brazo generalmente se puede mover con tres grados de libertad. La combinación de los movimientos posiciona a la muñeca sobre la pieza de trabajo. La muñeca normalmente consta de tres movimientos giratorios. La combinación de estos movimientos orienta a la pieza de acuerdo a la configuración del objeto para facilitar su recolección. Estos tres últimos movimientos se denominan elevación (pitch), desviación (yaw), y giro (roll). Por tanto, para un robot con seis articulaciones, el brazo es el mecanismo de posicionamiento, mientras que la muñeca es el mecanismo de orientación.

Muchos de los robots industriales, que están disponibles comercialmente, se utilizan ampliamente en tareas de fabricación y de ensamblaje, tales como manejo de material, soldaduras por arco y punto, montajes de piezas, pintura al spray, carga y descarga de máquinas controladas numéricamente, exploraciones espaciales y submarinas, investigación de brazos protésicos y en el manejo de materiales peligrosos.

3.- Las áreas de aplicación de los sistemas expertos incluyen la diagnosis y prescripción médica; automatización del conocimiento médico; interpretación de datos químicos; síntesis biológicas y químicas; descubrimiento de minerales y petróleo; planificación y programación; ayudas de estrategia militar; localizaciones tácticas de blancos; defensa nacional; control de tráfico aéreo; análisis de circuitos; diseños VLSI; ayudas para determinar los daños de una estructura; selección para la configuración de una computadora; entendimiento del habla; enseñanza asistida por computadora; acceso a conocimiento base y administración; planificación de procesos de programación y fabricación de sistemas expertos.

Existen unas pequeñas limitaciones para el uso de sistemas expertos en última instancia. Sin embargo, la naturaleza de su diseño y construcción es cambiante. Se ponen de manifiesto algunas de las limitaciones de los sistemas basados en reglas: No todos los conocimientos se pueden estructurar como relaciones empíricas. Tales asociaciones tienden a ocultar las relaciones causales y tampoco adecuadas para estructuras y funciones muy complicadas. Los Sistemas Expertos (SE) más recientes contienen conocimientos sobre causalidad y la estructura. Estos sistemas prometen ser mucho más sólidos que los sistemas actuales, y pueden dar respuestas correctas lo suficientemente buenas para que se considere su uso en sistemas autónomos y no sólo como ayudantes inteligentes.

Otra modificación es la tendencia creciente hacia los sistemas que no están basados en reglas. Tales sistemas, que emplean redes semánticas, matrices y otras estructuras de representación de conocimiento, se adaptan generalmente mejor a un modelo causal. Bando al problema concreto una representación más adecuada del conocimiento, se tiende a simplificar el razonamiento requerido. Algunos sistemas expertos emplean la solución "blackboard", que combina partes basadas en reglas y no basadas en reglas que trabajan juntas para construir soluciones de manera incremental, de tal manera que cada segmento de programa contribuye con su propia experiencia particular.

BIBLIOGRAFIA

- " SISTEMAS MODERNOS DE CONTROL ".
Richard C. Dorf.
Edit. Adisson-Wesley Iberoamericana. 2ª Edic.
- " DIGITAL CONTROL OF DYNAMIC SYSTEMS ".
Gene F. Franklin.
Edit. Adisson-Wesley. 2ª Edic.
- " ROBOTICA INDUSTRIAL. TECNOLOGIA. PROGRAMACION Y APLICACIONES ".
Mikell P. Groover.
Edit. Mc Graw-Hill. 1ª Edic.
- " ROBOTICA. CONTROL, DETECCION, VISION E INTELIGENCIA ".
R.C. González.
Edit. Mc Graw-Hill. 1ª Edic.
- " MANUAL DEL MICROPROCESADOR #P80386 ".
INTEL México, 1985.
- " MANUAL DEL MICROPROCESADOR #P80486 ".
INTEL México, 1992.
- " ROBOTICA ".
Marvin Minsky.
Edit. Planeta. 1ª Edic.
- " INTERCONEXION DE PERIFERICOS A MICROPROCESADORES ".
Mompín.
Edit. Marcombo. 2ª Edic.
- " AUTOMATAS PROGRAMABLES ".
Alejandro Forras Criado.
Edit. Mc Graw-Hill. 1ª Edic.

" PRINCIPIOS DE INTELIGENCIA ARTIFICIAL Y SISTEMAS
EXPERTOS ".

David W. Roalston.

Edit. Mc Graw-Hill.

1ª Edic.

" UTILIZACION DE C EN INTELIGENCIA ARTIFICIAL ".

Herbert Schildt.

Edit. Mc Graw-Hill.

1ª Edic.

" FUNDAMENTOS DE LOS MICROPROCESADORES ".

Roger Tothheim.

Edit. Mc Graw-Hill.

2ª Edic.

" MICROPROCESADORES, PROGRAMACION E INTERCONEXION ".

José María Uruñuela M.

Edit. Mc Graw-Hill.

2ª Edic.

I N D I C E

INTRODUCCION	1
JUSTIFICACION	3
ANTECEDENTES AL TRABAJO	4
PLAN PROPUESTO	5
OBJETIVO GENERAL	6
OBJETIVOS PARTICULARES	6
CAPITULO I.- INTRODUCCION A LOS MICROPROCESADORES DE 32 BITS	7
I.1.1. MICROPROCESADOR (μ P)	7
I.1.2. UNIDAD LÓGICA ARITMÉTICA (ALU)	9
I.1.3. MEMORIA DE SOLO LECTURA (ROM)	10
I.1.4. MEMORIA DE ACCESO ALEATORIO (RAM)	14
I.1.5. DISPOSITIVOS DE ENTRADA/SALIDA (E/S)	16
I.1.6. BUSES DE COMUNICACIONES	17
I.1.7. INTERFASES DE COMUNICACION	19
I.1.8. PROTOCOLOS DE COMUNICACION	21
CAPITULO II.- ARQUITECTURA DEL MICROPROCESADOR μ P80386 DE INTEL	24
II.1. INTRODUCCION AL MICROPROCESADOR μ P80386	24
II.2. ARQUITECTURA INTERNA DEL MICROPROCESADOR μ P80386 DE INTEL	27
II.3. CONEXIONADO DEL MICROPROCESADOR μ P80386	29
II.4. REPERTORIO DE INSTRUCCIONES DEL MICROPROCESADOR μ P80386 DE INTEL	37

CAPITULO III.- ARQUITECTURA DEL MICROPROCESADOR μ P80486
DE INTEL44

III.1. INTRODUCCION AL MICROPROCESADOR μ P80486	44
III.2. ARQUITECTURA INTERNA DEL MICROPROCESADOR μ P80486 DE INTEL	47
III.3. REGISTROS DEL MICROPROCESADOR μ P80486	53
III.4. REPERTORIO DE INSTRUCCIONES DEL MICROPROCESADOR μ P80486 DE INTEL	54

CAPITULO IV.- APLICACIONES AL CONTROL DIGITAL66

IV.1. INTRODUCCION	66
IV.2. INTELIGENCIA DELICUAL Y SISTEMAS EXPERTOS	68
IV.3. AUTOMATAS PROGRAMABLES	73
IV.3.1. ESTRUCTURA EXTERNA	76
IV.3.2. ESTRUCTURA O ARQUITECTURA INTERNA	77
IV.4. ROBOTICA	83
IV.4.1. CUATRO TIPOS DE CONTROLES DE ROBOT	86
IV.4.2. CONTROL COORDINADO DE FUERZA Y POSICION	89

CAPITULO V.- CONCLUSIONES90

BIBLIOGRAFIA

INDICE