

119
2ej.



UNIVERSIDAD NACIONAL
AUTÓNOMA DE MEXICO

UNIVERSIDAD NACIONAL AUTONOMA
DE MEXICO

FACULTAD DE ESTUDIOS SUPERIORES
CUAUTITLAN

**“TARJETA DE ADQUISICIÓN DE DATOS UTILIZANDO
EL MC-8051”**

T E S I S

QUE PARA OBTENER EL TITULO DE
INGENIERO MECANICO ELECTRICISTA

P R E S E N T A:

JUAN FRANCISCO TAPIA REYES

ASESOR: ING. JORGE BUENDIA GOMEZ

CUAUTITLAN IZCALLI, EDO. DE MEX. 1998

**TESIS CON
FALLA DE ORIGEN**

26 9550



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLAN
UNIDAD DE LA ADMINISTRACION ESCOLAR
DEPARTAMENTO DE EXAMENES PROFESIONALES

ASUNTO: VOTOS APROBATORIOS

DR. JUAN ANTONIO MONTARAZ CRESPO
DIRECTOR DE LA FES CUAUTITLAN
PRESENTE

AT'N: Q. Ma. del Carmen Garcia Mijares
Jefe del Departamento de Exámenes
Profesionales de la FES Cuautitlán

Con base en el art. 28 del Reglamento General de Exámenes, nos permitimos comunicar a usted que revisamos la TESIS:

"Tarjeta de adquisición de datos utilizando el MC-8051"

que presenta el pasante: Juan Francisco Tapia Reyes
con número de cuenta: 8817719-3 para obtener el TITULO de:
Ingeniero Mecánico Electricista

UNIVERSIDAD NACIONAL
AUTÓNOMA DE MÉXICO
FACULTAD DE ESTUDIOS SUPERIORES
CUAUTITLÁN

Considerando que dicha tesis reúne los requisitos necesarios para ser discutida en el EXAMEN PROFESIONAL correspondiente, otorgamos nuestro VOTO APROBATORIO

ATENTAMENTE.

"POR MI RAZA HABLARÁ EL ESPÍRITU"

Cuautitlán Izcalli, Edo. de Méx., a 26 de Mayo

de 1998

PRESIDENTE

Ing. José Luis Rivera López

VOCAL

Ing. Ubaldo Ramírez Urizar

SECRETARIO

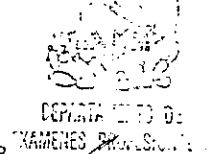
Ing. Jorge Buendía Gómez

PRIMER SUPLENTE

Ing. Blanca G. de la Peña Valencia

SEGUNDO SUPLENTE

Ing. Nicolás Calva Tapia



[Firmas manuscritas de los miembros del comité de exámenes]

De todas las criaturas que habían caminado sobre la Tierra, los hombres-mono fueron los primeros en mirar la Luna directa y sostenidamente. Aunque no podía recordarlo, en su temprana juventud Moon Watcher (El que Mira la Luna) intentó varias veces tocar el fantasmal rostro que surgía tras las colinas. Nunca lo logró, y ahora era lo suficientemente viejo como para comprender por qué.

La noche transcurrió, fría y clara, sin mayores signos de alarma, mientras la Luna recorría el cielo entre constelaciones ecuatoriales que ningún ojo humano vería. En las cuevas, entre una mezcla de ronquidos y espera temerosa, nacían las pesadillas de generaciones futuras.

Clarke, C. Arthur

"2001"

Este trabajo es el producto, como es costumbre de muchas personas, por ello quiero en este espacio, aprovechando que solo en esta ocasión mi voz tiene foro, dar gracias a todas aquellas con las cuales fuimos viajeros de la misma excursión, de esa excursión-transición de adolescentes en personas un poco más maduras, a los ingenieros Angel G., Alberto R., Oscar C., Hugo G., Oswaldo M. y a mi grandes amigas Lic. Yolanda A.(espero que pronto...) y Lic. Antonieta M..

Agradezco a la Universidad Nacional Autónoma de México, a la Facultad de Estudios Superiores Cuautitlán por brindarme ese espacio para convivir con mis amigos pues no solo fue escuela, sino punto de encuentro obligado para todos aquellos que tuvimos la suerte de conocernos allí y coincidir en muchos intereses personales.

Quiero agradecer especialmente a los profesores: Ingeniero Agapito Rodríguez Nava a quien más que maestro de una materia, recuerdo como el amigo que me enseñó a apreciar uno de los valores más importantes de esta vida: La lealtad. Al ingeniero José Antonio Sánchez por su eterna disposición a conversar y por ser un ejemplo a seguir para todo estudiante de ingeniería, al ingeniero Ubaldo Ramírez Urizar por haber creado en mí la orientación y el gusto por la electrónica, al ingeniero Buendía por que sin su ayuda este trabajo hoy no sería posible. Y quiero agradecer en general a todos aquellos profesores algunos de ellos ya no están en la facultad, que hicieron de mí la persona que soy, y que desinteresadamente nos dan un pedazo de su tiempo y con ello de su vida.

Quiero dedicar este trabajo: a mi madre señora Eulogia Reyes Sierra, por su apoyo moral, su preocupación, cariño y comprensión que siempre le ha manifestado a todos su hijos, a mi padre señor Rafael Tapia Palma por ser ese apoyo invaluable que me permitió estudiar y ser siempre ese amigo incondicional que tanto me ha ayudado, a mis hermanas a mi hermano, sin todos los cuales yo como individuo no existiría.

Allí donde el tiempo se detiene
entre el destello y el estruendo
Allí la esperanza está más cerca
de regresar el éxito a México

INTRODUCCIÓN Y OBJETIVOS	1
1.1 COMPONENTES BÁSICOS	3
Unidad aritmética lógica	
Unidad de memoria	
La unidad de entrada	
Unidad de salida	
Unidad de control	
La Unidad Central de Procesamiento	
1.2 ELEMENTOS BÁSICOS DE LA MICROCOMPUTADORA	6
Los buses de datos	
Bus de direccionamiento	
El microprocesador	
Los registros de la CPU	
Tamaño del conjunto de instrucciones	
Los registros de uso general	
Los registros de apuntadores	
Los registros de segmentos	
El registro apuntador de instrucciones	
El registro de banderas	
CAPITULO 2, ADQUISICIÓN DE DATOS	
2.1 ALGO DE HISTORIA	12
2.2 ACONDICIONAMIENTO DE LA SEÑAL	13
Amplificación	
Aislamiento	
Filtrado	
Linearización	
Excitación	
2.3 LA IMPORTANCIA DEL MEDIO AMBIENTE	14
2.4 FACTORES CLAVE	15
2.5 SISTEMAS DE UN SOLO CANAL	15
2.6 SISTEMAS MULTICANAL	17
CAPITULO 3, CARACTERÍSTICAS DEL MICROCONTROLADOR 8051	
3.1 CARACTERISTICAS DEL μ C-8051	20
3.2 DESCRIPCIÓN DE LINEAS DEL μ C-8051	20
3.3 CARACTERÍSTICAS ESPECÍFICAS DEL μ C-8051	22
Descripción de los espacios de memoria	
3.4 MEMORIA DEL PROGRAMA	23
Localidades asignadas a las interrupciones	
Memoria de programa interna y externa	

3.5 MEMORIA DE DATOS	25
3.6 LOCALIDADES DE LOS REGISTROS DE FUNCIONES ESPECIALES	26
3.7 REGISTROS DE PALABRA DEL ESTADO DEL PROGRAMA	26
3.8 REGISTRO DE CONTROL DE POTENCIA	27
3.9 REGISTRO DE INTERRUPCIONES	28
Registro Habilitador de Interrupciones (IE)	
Registro de Prioridad	

CAPITULO 4, MANEJO DE LOS TEMPORIZADORES Y CONTADORES

4.1 TIMER/CONTADOR	30
4.2 REGISTRO DE CONTROL DE PUERTO TIMER/CONTADOR	30
4.3 REGISTRO DE MODO 1 DE CONTROL DEL TIMER/CONTADOR	31
Modo 0 del Timer/Contador	
Modo 1 del Timer/Contador	
Modo 2 del Timer/Contador	
Modo 3 del Timer/Contador	
4.4 UTILIZACION DEL TIMER 1 COMO GENERADOR DEL "BAUD RATE" PARA LA TRANSMISIÓN SERIE	34
4.5 PUERTO SERIE	35
4.6 REGISTRO DE CONTROL DE PUERTO SERIE SCON	35
4.7 TRANSMISIÓN SERIE USANDO EL MODO DE CONTROL 1	37

CAPITULO 5 MODOS DE DIRECCIONAMIENTO

5.1 MODOS DE DIRECCIONAMIENTO	40
5.2 DIRECCIONAMIENTO DIRECTO	40
5.3 DIRECCIONAMIENTO INDIRECTO	40
5.4 DIRECCIONAMIENTO INMEDIATO	41
5.5 DIRECCIONAMIENTO POR INDEXADO	41
5.6 DIRECCIONAMIENTO POR REGISTRO	42
5.7 TRANSFERENCIA DE DATOS	42
RAM interna	
RAM externa	
Movimientos de tablas localizadas en la memoria del programa	
5.8 INSTRUCCIONES BOOLEANAS	43
5.9 INSTRUCCIONES DE SALTO	44
Saltos condicionados	
Saltos incondicionados	

CAPITULO 6, TARJETA DE ADQUISICIÓN DE DATOS EN BASE AL MC-8051

DISEÑO EXPERIMENTAL	47
RESULTADOS	56
CONCLUSIONES	57
APÉNDICE Convertidor A/D ADC0804	58
BIBLIOGRAFÍA	61

INTRODUCCIÓN

Muchas variables físicas son de naturaleza analógica y pueden tomar cualquier valor dentro de un rango continuo de medición. Ejemplos de variables de este tipo incluyen temperatura, presión, intensidad luminosa, señales de audio, posición, velocidad rotacional y de flujo. Los sistemas de almacenamiento electrónico de información y de procesamiento, sin embargo trabajan con información digital. Cualquier información que tenga que introducirse en un sistema de estos, primero debe ponerse en forma digital. De manera similar las salidas de un sistema de esta naturaleza siempre son digitales. Cuando un sistema tal como una computadora se va utilizar para vigilar un proceso físico, el diseñador se enfrenta con la diferencia entre la naturaleza digital de la computadora y la analógica de las variables del proceso. La figura a) ilustra esta situación. El diagrama muestra los tres elementos que participan cuando una computadora vigila una variable física cuya variable es analógica.

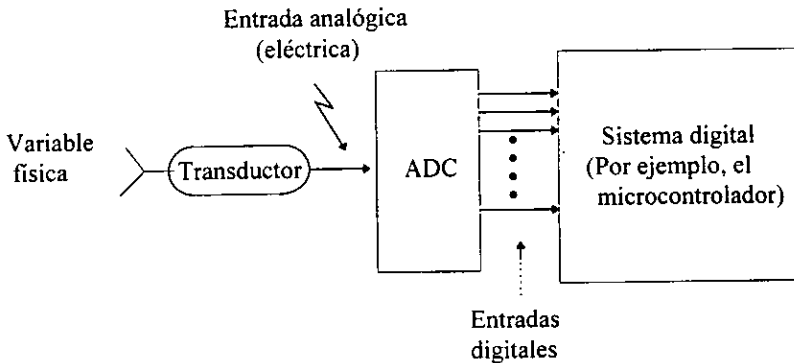


Fig. a) El convertidor analógico-digital (ADC) se utiliza para conectar al sistema digital con el mundo analógico de modo que este pueda vigilar una variable física.

Existen muchas aplicaciones en las cuales los datos analógicos tienen que ser digitalizados (convertidos a forma digital) y transferidos a la memoria de una computadora (o microcontrolador). Al proceso mediante el cual la computadora adquiere estos datos analógicos digitalizados se le llama adquisición de datos. La computadora puede ejecutar diferentes tareas con los datos, según la aplicación. En una aplicación de almacenamiento, tal como una grabación de audio o de un osciloscopio, la microcomputadora interna almacenará los datos y después puede procesarlos o transferirlos a un convertidor digital analógico, con lo cual después de un tiempo se producirá la señal analógica original. En una aplicación de control de procesos, la computadora puede examinar los datos o desarrollar cálculos sobre ellos para determinar que tipo de salidas de control generar.

Existen prácticas en el laboratorio de Termofluidos donde el registro de variables tales como la presión o la temperatura consumen gran parte del tiempo de las prácticas, si se

considera que el ciclo escolar de cada semestre tiene un número de prácticas que en la mayoría de veces no se puede cubrir en su totalidad por falta de tiempo, se nota que es injusto que el estudiante pase buena parte de su tiempo en el registro de estas variables que cualquier sistema comercial puede cubrir. Existen también fenómenos que debido a su rapidez de ocurrencia no se puede registrar de forma manual su acontecimiento, por ejemplo el monitoreo de un electrocardiograma, es por esto, por el ahorro de tiempo en nuestro caso, por el ahorro de dinero si vemos en un marco más general la necesidad de registrar datos para su posterior análisis (Sería un gasto absurdo mantener ocupada a una persona con la única labor de anotar la variación de una variable física durante un día o una semana para vigilar el desempeño de un fenómeno de la misma naturaleza), y por la gran popularidad de los sistemas de adquisición de datos, que se planteó la construcción de un sistema que cubriera las necesidades mínimas, además de que no existían tarjetas de adquisición de datos en el laboratorio de Termofluidos, y el comprar un sistema de "bajo costo" sería caro, tomando en cuenta el presupuesto que la universidad destina a tal laboratorio, considérese también de que no solo se requeriría de uno. En un inicio la solución más fácil era: el uso de una computadora, el puerto paralelo y un convertidor analógico a digital, pero aún el uso de una computadora para almacenar datos sacrificaba este recurso en el laboratorio que en ese tiempo solo poseía dos; se decidió por lo tanto usar un microcontrolador por a nuestro parecer cubrir las características de una computadora sin serlo en el sentido estricto, se podría haber usado un microprocesador pero los periféricos que requiere lo hacían más complejo. Después de decidir el uso del microcontrolador el dilema era cual usar, nuevamente se decidió emplear el 8031 de la familia de microcontroladores del 8051 de Intel por ser la versión más barata de esta familia y por extender el conocimiento en este tipo de dispositivos, pues en el laboratorio de electrónica ya se usaba el 68HC11 de Motorola, pero no se conocían las bondades del 8031, se seleccionó de 8 bits porque en el caso de monitorear presión o temperatura las variables más comunes en el laboratorio, la resolución de 8 bits era más que suficiente, además de que se tenía más familiaridad por las prácticas de electrónica digital con el uso de convertidores de 8 bits y de esta manera se podían interconectar directamente el bus de salida del convertidor a la entrada de datos del microcontrolador.

En el capítulo 1 se da una breve descripción de lo que es un microcontrolador y se explica brevemente una microcomputadora por ser un elemento más familiar y por tener una gran parecido operacional, en el capítulo 2 se habla de manera genérica de lo que es un sistema de adquisición de datos, en el capítulo 3 se da una descripción más detallada de lo que es un microcontrolador, en el capítulo 4 se habla del manejo de los temporizadores y contadores por la importancia que tienen en el programa en ensamblador de nuestro trabajo, en el capítulo 5 se tratan los modos de direccionamiento del microcontrolador y en el capítulo 6 se presenta la tarjeta de adquisición de datos propuesta.

OBJETIVOS. Uno de los objetivos de este trabajo es proveer al laboratorio de Termofluidos de un medio de adquisición de datos de bajo costo para sensar variables tales como: la temperatura, la presión, el nivel y el flujo. El otro objetivo es conocer el funcionamiento del microcontrolador MC-8051 de Intel para su posible implementación en las prácticas de los laboratorios de la carrera de Ingeniería Mecánica Eléctrica.

CAPITULO 1

Inicialmente los microcontroladores fueron pensados para ser simples implementaciones de las computadoras con un limitado conjunto de instrucciones y con capacidades entrada/salida (I/O) realizadas para que pudieran ser producidos en forma masiva a precios baratos. Con el desarrollo de microcontroladores más complejos como los de arquitecturas de 32-bit ,los microcontroladores se han hecho populares en aplicaciones tales como procesamiento digital de señales (DSP), control de periféricos de computadoras como las impresoras, detección de tiempo real, modulación y demodulación de señales de FAX, también como sistemas de control no lineal. Es por estas razones que se asume que el uso funcional de una computadora determina si se trata de una computadora una supercomputadora o un sistema de microcontrolador, es decir uno puede fijarse en parámetros tales como el número de instrucciones, cantidad de memoria virtual o física y la velocidad de procesamiento para diferenciar a los sistemas de microcontrolador de las supercomputadoras. En este capítulo se presenta de forma general los aspectos básicos que definen a una computadora para que en el capítulo dos se aborde con más confianza al sistema de microcontrolador, esto se hace así porque se considera de primera instancia, más cercano nuestro contacto con la terminología empleada para el microprocesador.

1.1 COMPONENTES BÁSICOS

Podemos mencionar cinco elementos como los constituyentes esenciales de una computadora: la *unidad de entrada*, la *unidad de salida* (más conocidas como input/output (I/O)), la *unidad de memoria*, la *unidad de control* (CU), y la *unidad aritmético lógica* (ALU, por sus siglas en inglés), las dos últimas se les puede englobar en una sola más general la CPU. Esto se muestra en el diagrama de bloques de la figura 1.1. Las flechas en este diagrama indican la dirección en la cual fluyen los datos, información o señales de control. Se usan flechas de dos tamaños diferentes; las flechas mayores representan datos o información que consisten en un número relativamente grande de líneas paralelas, las flechas menores representan señales de control que, por lo general, son sólo una o unas cuantas líneas. Todos estos elementos son necesarios para conformar una computadora completa. Si no hay entrada, la computadora no puede responder a su entorno. Si no hay salida, la computadora no puede efectuar cambios en su entorno. Si no existe ALU, la computadora no puede desarrollar alteraciones en su entrada si no solo almacenarla o pasarla a la salida. Y si no tiene memoria, el sistema no es mas que una maquina de estado-finito. Mientras la mayoría de las aplicaciones de los microcontrolares se pueden representar por y funcionar como maquinas de estado-finito, la memoria de la computadora les permite realizar la función de una máquina de estas de la misma forma que procesar cualquier número procesable.

Unidad aritmética lógica

ALU (Arithmetic Logic Unit), realiza las operaciones aritméticas y lógicas requeridas por el conjunto de instrucciones que soporta. El tipo de operación que se realizará se determina por medio de la unidad de control (figura 1). Los datos que serán utilizados por la ALU pueden provenir de la unidad de memoria (flecha 2) o de la unidad de entrada (flecha 3). Los resultados de operaciones realizadas en la ALU pueden transferirse a la unidad de memoria para ser almacenados (flecha 4) o a la unidad de salida (flecha 5).

Unidad de memoria La memoria almacena grupos de dígitos (palabras) binarios que pueden representar instrucciones (programa) que la computadora ejecutará y los datos que serán operados en el programa. La memoria también sirve como almacenamiento de resultados intermedios y finales de operaciones aritméticas (flecha 4). La operación de la memoria es controlada por la unidad de control, que indica una operación de lectura o de escritura. Una localidad dada en la memoria es accesada por la

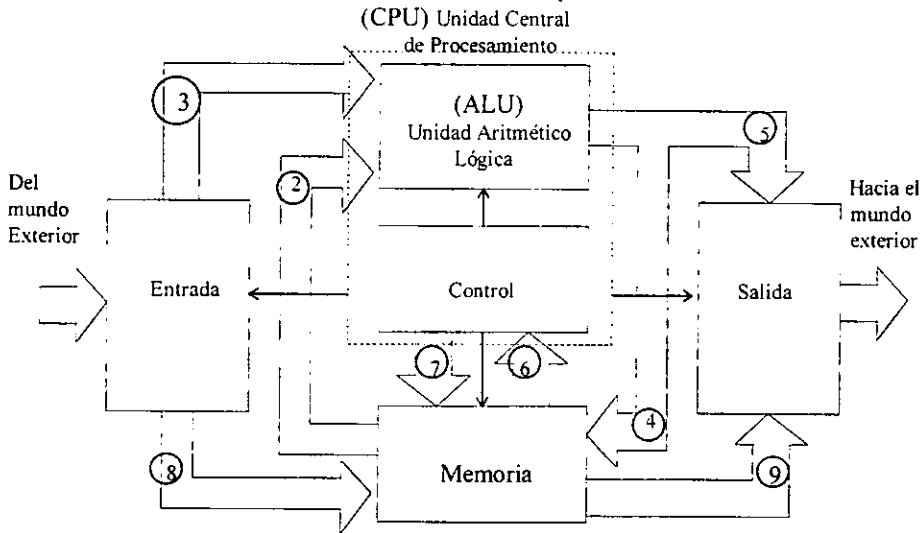


Figura 1.1 Organización básica de una computadora

unidad de control, la cual proporciona el código de dirección adecuado (flecha 7). Puede escribirse información en la memoria de la ALU o de la unidad de entrada (flecha 8), una vez más bajo el mando de la unidad de control. Puede leerse información de la memoria de la ALU (flecha 2) o de la unidad de salida (flecha 9).

La unidad de entrada Esta unidad consta de todos los dispositivos que se usan para tomar información y datos que son externos a la computadora y colocarlos en la unidad de memoria (flecha 8) o la ALU (flecha 3). La unidad de control determina hacia dónde se envía la información de entrada. La unidad de control se utiliza para meter el programa y los datos en la unidad de memoria antes de poner en marcha la computadora. Esta unidad se usa asimismo para introducir datos en la ALU desde un dispositivo externo durante la ejecución de un programa. Algunos de los dispositivos de entrada comunes son los teclados, interruptores, impresoras, lectoras de tarjetas perforadas, unidades de disco magnético, unidades de cinta magnética y convertidores de analógico a digital (ADC).

Unidad de salida Esta unidad consta de los dispositivos que se usan para transferir datos e información de la computadora al "mundo exterior". Los dispositivos de salida son dirigidos

CAPITULO 1

por la unidad de control y pueden recibir datos de la memoria (flecha 9) o de la ALU (flecha 5), los cuales después se colocan en forma adecuada para su uso externo. Algunos ejemplos de dispositivos de salida comunes son dispositivos de exhibición LED, impresoras, unidades de disco o cinta, monitores de vídeo y convertidores de digital a analógico (DAC).

Conforme la computadora ejecuta su programa, generalmente tiene resultados de señales de control que debe presentar al mundo exterior. Por ejemplo, un sistema de computación podría tener una impresora de matriz de punto como dispositivo de salida. Aquí la computadora envía señales para imprimir los resultados en papel.

Unidad de control

La función de la unidad de control ahora debe ser obvia. Dirige la operación de todas las otras unidades ofreciendo señales de temporización y control. En cierto sentido, la unidad de control es como el director de una orquesta, quien es el responsable de mantener a cada uno de los miembros de la orquesta en sincronización adecuada. Esta unidad contiene circuitos lógicos y de temporización que generan las señales adecuadas que se necesitan para ejecutar cada instrucción en un programa.

La unidad de control extrae una instrucción de la memoria, mediante el envío de una dirección (flecha 7) y un comando de lectura a la unidad de memoria. La palabra de instrucción almacenada en localidad de memoria se transfiere después a la unidad de control (flecha 6). Esta palabra de instrucción, la cual está en alguna forma de código binario es decodificada después por los circuitos lógicos de la unidad de control para determinar qué instrucción es solicitada, la unidad de control utiliza esta información para generar las señales necesarias para ejecutar la instrucción.

Esta secuencia de búsqueda y acarreo de un código de instrucción y luego la ejecución de la operación indicada, la repite una y otra vez la unidad de control. Esta secuencia repetitiva de búsqueda y ejecución continúa hasta que se apaga la computadora o hasta que la unidad de control recibe y carga una instrucción que le ordena se detenga.

Entonces como vemos la computadora continúa efectuando las mismas operaciones básicas una y otra vez: búsqueda, ejecución, bejecución, etcétera. Por supuesto, los diversos ciclos de ejecución serán distintos para cada tipo de instrucción, conforme la unidad de control envía señales diferentes a las otras unidades para ejecutar la instrucción particular.

La Unidad Central de Procesamiento En la figura 1-1 la ALU y la unidad de control se muestran combinadas en la llamada *unidad central de procesos* o (CPU) un conjunto de circuitos lógicos muy complejo, que ejecuta una serie de instrucciones alojadas en el mismo hardware. Su objetivo es buscar esas instrucciones, decodificarlas, ejecutarlas y finalmente almacenar el resultado de las operaciones realizadas. Para llevar acabo todo lo anterior, la CPU necesita acceder la información en algún lugar (Puede ser la memoria o un puerto) y decodificarla. El resultado de la decodificación se usa para especificar la operación que se realizará sobre algún operando y el sitio donde deberá almacenarse dicho resultado. Una vez que la instrucción ha sido decodificada y sus operandos leídos, la CPU ejecuta la lógica dictada por la instrucción auxiliándose de la ALU en el caso de que se tenga que efectuarse una operación sobre la instrucción.

1.2 ELEMENTOS BÁSICOS DE UNA MICROCOMPUTADORA

Es importante entender la diferencia entre la microcomputadora (μC) y el microprocesador (μP). Una microcomputadora contiene varios elementos, donde el microprocesador es el más importante de todos ellos. Generalmente el microprocesador es un solo CI que contiene toda la circuitería de las unidades de control y aritmética lógica, en otras palabras, la CPU de la microcomputadora.

Las muchas estructuras posibles de una microcomputadora son esencialmente iguales en principio. A fin de ofrecer un medio más claro la estructura que hemos elegido y de las cual se darán algunos detalles más adelante es la de microprocesador 8086.

La microcomputadora tiene tres buses que transportan toda la información y señales implicadas en la operación del sistema. Estos canales conectan el microprocesador (CPU) a cada uno de los elementos de memoria y E/S, de manera que los datos y la información puedan fluir entre la CPU y cualquiera de estos tres elementos. En otras palabras, la CPU continuamente está involucrada en el envío o recepción de información hacia o desde una localidad en la memoria, un dispositivo de entrada o un dispositivo de salida.

En la microcomputadora, todas las transferencias de información se dirigen a la CPU. Cuando está envía datos a otros elementos de la computadora, a ésta se le llama operación de escritura y la CPU escribe en el elemento seleccionado. Es muy importante entender que los términos "lectura" y "escritura" siempre se refieren a la operación realizada por la CPU.

Los buses implicados en todas las transferencias de datos tienen funciones que se describen como sigue:

Los buses de datos existen dos conjuntos distintos de buses que se pueden identificar basados en su localización y en si el diseñador de hardware tiene acceso a ellos. Los buses internos son parte de la organización de la computadora y están en gran medida relacionados con el diseñador del chip pero tienen muy poco que ver con el diseñador de las aplicaciones del microprocesador. Estos buses internos no tienen efecto en el conexionado externo del chip y solo afectan la velocidad operativa efectiva del chip.

La CPU necesita interactuar de alguna manera con su entorno (memoria y puertos). Esto lo realiza mediante un conjunto de líneas paralelas llamadas bus externo de datos. El tamaño (la cantidad de líneas) de dicho bus impone un límite a la cantidad de información que puede ser transmitida en una sola operación. Sólo se puede transmitir un bit por línea simultáneamente. A este proceso de transmisión se le conoce como un ciclo del bus.

Este bus tiene relación con la velocidad de transferencia dentro de la PC. Si para transferir 8 bits de información se requiere un ciclo de bus (tomando en cuenta que el tamaño del bus es de 8 bits), para transferir 16 o 32 tomará el doble o el triple de tiempo. Es en este punto donde se puede apreciar una de las formas que influyen sobre la velocidad total de operación en una PC (a mayor anchura de bus, mayor velocidad de transferencia). Los procesadores 386dx y 486 tienen una velocidad del doble en relación con el 8086 (debido a que en ambos la longitud del bus externo es de 32 bits, mientras que en el 8086 es de 16).

bus de direccionamiento El bus de datos se usa para permitir a la CPU interactuar con su mundo exterior (intercambiar datos con la memoria y los puertos). Sin embargo, para poder acceder dichas localidades de memoria es necesario generar una dirección. Dicha dirección es generada por la CPU y puesta en otro bus llamado bus de direcciones. El bus de direcciones determina el máximo de memoria accesible; el 8086 tiene un bus de direcciones de 20 bits. Por consiguiente, tiene la capacidad de acceder hasta 2^{20} (1 048 576 o bien 1 megabyte) diferentes localidades de memoria. Por su parte, el 80486 -con un bus de direcciones de 32 bits- tiene capacidad para acceder 2^{32} (4 294 967 296 o bien 4 gigabytes) diferentes localidades de memoria.

El microprocesador El microprocesador es el corazón de cualquier computadora. Realiza muchas funciones entre las cuales se incluyen:

1. Proporcionar las señales de control y temporización para todos los elementos de la microcomputadora.
2. Extraer las instrucciones y los datos de la memoria.
3. Transferir los datos a y desde la memoria y los dispositivos de entrada/salida.
4. Instrucciones para decodificar.
5. Ejecutar las operaciones aritméticas y lógicas invocadas por las instrucciones.
6. Responder a las señales de control generadas en entrada/salida tales como RESET e INTERRUPT

La lógica interna del microprocesador es muy compleja, pero puede considerarse como conformada por tres secciones básicas: la sección de control y temporización, la sección de registros y la ALU. Aunque existen interacciones bien definidas entre estas tres secciones, cada una tiene funciones específicas.

La función principal de la sección de control y temporización es traer de la memoria y decodificar (interpretar) las instrucciones que están en la memoria y que forman parte de un programa, para después generar las señales de control necesarias para ejecutar las instrucciones. Esta sección también genera las señales de control y temporización (por ejemplo, R/W, reloj) para la RAM, ROM y dispositivos E/S externos. Toda CPU es accionada por un reloj externo. La frecuencia del reloj, también conocida como velocidad de proceso (medida en millones de Hertz o MHz), se especifica en megahertz y determina qué tan rápido ejecutará instrucciones la CPU. Conforme aumenta la velocidad, la CPU es capaz de ejecutar más instrucciones en el mismo lapso.

Un ejemplo de tales instrucciones de alto nivel sería la multiplicación de dos operandos en una sola instrucción (MUL AX,BX). En el caso de la CPU tipo RISC, el programador carga con la tarea de implementar la instrucción anterior usando instrucciones de más bajo nivel proporcionadas por la misma CPU. Estas CPU permiten que la misma CPU sea optimizada vía hardware y le dan la capacidad de ejecutar las instrucciones de su repertorio en forma más eficiente y rápida.

CAPITULO 1

La familia 80X86 está formada por procesadores tipo CISC (Complex Instruction Set Computer: Computadora con un conjunto complejo de instrucciones), cuya orientación es más bien la de ejecutar instrucciones de alto nivel.

La sección de registros contiene varios registros (dentro del micro) cada uno tiene una función especial. El más importante de todos ellos es el **contador del programa (PC)**. Los demás registros se emplean para realizar funciones tales como guardar los códigos correspondientes a las instrucciones mientras éstas son decodificadas, retener los datos que constituyen los operandos con los que trabaja la ALU, almacenar direcciones donde se encuentran los datos en la memoria y varias funciones más de conteo y almacenamiento con propósitos generales.

Los registros de la CPU El 8086 tiene 14 registros de 16 bits que sirven para llevar el control de todo lo que sucede dentro de la PC. Dichos registros se dividen en diversas categorías de acuerdo con su actividad preponderante. Existen cuatro registros de uso general, dos registros de apuntadores, dos registros índice, cuatro registros de segmentos y el registro de banderas.

Tamaño de los registros en bits un registro está formado por un conjunto de flip-flops usados para manipular datos y procesar las instrucciones que están siendo ejecutadas. De la misma manera en que el bus de datos influye sobre la velocidad total de operación de la PC, también lo hace el tamaño del registro. Los registros más anchos (en bits) son más eficientes, pues permiten procesar operandos (datos sobre los cuales actuará una instrucción mnemónica) más grandes en un sólo paso.

Tamaño del conjunto de instrucciones Se le llama conjunto de instrucciones a la cantidad de instrucciones válidas que puede ejecutar la CPU. Generalmente se tienen dos categorías de instrucciones, dependiendo de la CPU diseñada. La primera se denomina RISC (Reduced Instruction Set Computer: computadora con un conjunto reducido de instrucciones). Este tipo de CPU está diseñado para ejecutar un conjunto de instrucciones muy pequeño, y deja la carga de lo que se considera instrucciones de alto nivel al programador.

Los registros de uso general Estos cuatro registros son duales. Es decir, pueden manipularse como si fueran de 8 o de 16 bits.

El registro llamado acumulador, resulta indispensable para toda operación de entrada/salida y para algunas operaciones de cadenas. El registro base, se usa para calcular direcciones que accesan la memoria. El registro contador es usado para llevar la cuenta en operaciones repetitivas, o como contador en iteraciones que pueden involucrar o no operaciones sobre cadenas. También sirve como contador en operaciones relacionadas con bits individuales.

Finalmente se tiene el registro DX, también conocido como registro de datos, cuya finalidad es servir como depósito para las direcciones de los puertos. En combinación con el registro

CAPITULO 1

AX, se utiliza para designar cantidades de 32 bits (DX:AX). Otra aplicación de este registro son las operaciones de interrupciones, durante las cuales conserva el desplazamiento (offset) del bloque de información que será manejado; este bloque puede ser una cadena de caracteres.

Los registros de apuntadores Como su nombre lo indica, son registros que apuntan a alguna localidad de memoria en general. El BP (Base Pointer) o apuntador base es muy usado para manipular la información que se encuentra en la pila. Una de sus principales utilidades es proveer un mecanismo para poder pasar parámetros a rutinas. El SP (Stack Pointer) o apuntador de pila, en conjunto con el registro de segmento SS (Stack Segment) o segmento de pila, se usa para crear en memoria una estructura llamada pila (stack).

los registros de segmentos El procesador tiene cuatro registros de 16 bits llamados registros de segmentos, los cuales se detallan en seguida. El CS (code segment) o registro del segmento de código se encuentra asociado al código del programa; es decir, controla el código de los programas y tiene como socio al registro IP. La combinación CS:IP da como resultado la siguiente instrucción a ser ejecutada. El DS(data segment) o registro del segmento de datos es donde se guardan los datos del programa. El ES (extra segment) o registro de segmento extra generalmente tiene la función de servir como "colchón" para ampliar el segmento de datos. El SS (stack segment) o registro del segmento de pila tiene la función de controlar el área donde se creará la pila.

Los registros anteriores son primordiales para generar la dirección de 20 bits que se pondrá en el bus de direcciones de la CPU. Cualquier operación que accese la memoria usará forzosamente uno de estos registros. Cada registro selecciona un área contigua de memoria que puede ser hasta de 64 kb, conocida como segmento. Dicho segmento tiene en su interior desplazamientos que van desde 0 hasta 64 kb. Toda instrucción utiliza tanto los registros de segmentos como algún desplazamiento interno para acceder información.

El registro apuntador de instrucciones El IP (instruction pointer) o apuntador de instrucciones se usa en combinación con el registro CS para especificar el área de memoria donde se encuentra la siguiente instrucción por ejecutar. Este registro cambia de contenido conforme al flujo de ejecución del programa.

El registro de banderas Es un registro de 16 bits que reporta el status del resultado de alguna operación aritmética o lógica. También se le conoce como registro de estado (status register) o PSW (program status word: palabra de status del programa). Cada bit de este registro tiene dos condiciones: encendido o apagado. En otras palabras puede ser igual a 1 (cuando está encendido o puesto) o igual a 0 (cuando está apagado).

El bit de CF (carry flag) o bandera de acarreo indica si alguna operación causó el acarreo (carry) del bit más alto en el registro de operación. El bit de PF (parity flag) o bandera de paridad indica el resultado sobre la paridad de los 8 bits más bajos en una operación aritmética o lógica. El bit de AF (auxiliar carry flag) o bandera auxiliar de acarreo funciona como el CF, con la excepción de que sólo actúa sobre los 4 bits más bajos del resultado. El bit de ZF (zero

CAPITULO 1

flag) o bandera de cero indica si el resultado de alguna operación lógica o aritmética es cero. El bit de SF (sign flag) o bandera de signo siempre tendrá el valor del bit más alto en el resultado de una operación lógica o aritmética. El bit OF (overflow flag) o bandera de desbordamiento indica el resultado de operaciones lógicas o aritméticas sin signo; es decir, no se manejan cantidades negativas.

Ejemplo Para complementar las secciones anteriores, se presenta un pequeño ejemplo sobre la operación de la CPU al ejecutar un programa y efectuar el proceso de leer 1 byte de la memoria. Al ejecutar un programa, la CPU debe obtener las instrucciones de la memoria durante lo que se llama un ciclo de lectura de máquina. Este ciclo se comporta de la siguiente manera:

- ◆ La CPU genera la dirección de memoria donde se leerá el byte
- ◆ Pone la dirección en el bus de direcciones y manda una señal de lectura.
- ◆ Ciertos circuitos se encargan de seleccionar el chip de memoria donde se encuentra el byte por leer. El chip mencionado decodifica internamente la dirección y accesa el byte.

- ◆ El byte es puesto en el bus de datos por otros mecanismos, tras lo cual finalmente es leído y procesado por la CPU.

El bloque de registros de uso general representa todos aquellos registros programables a los que tiene acceso el usuario (en esta unidad).

Podríamos decir entonces que los Microcontroladores son computadoras digitales diseñadas particularmente para supervisar, monitorear, dirigir y controlar varios procesos en la industria, los negocios, la milicia y muchas otras áreas de aplicación. Con el advenimiento de la tecnología VLSI, los microcontroladores se convierten cada vez mas en microcomputadoras embebidas. Como un microcontrolador posee en esencia la arquitectura de una computadora, contiene por lo tanto una unidad central de procesamiento (CPU), memoria principal así como memoria cache, Interfaces de entrada salida (I/O), controladores de acceso directo a memoria (DMA), manejadores de interrupciones, temporizadores y otros subsistemas necesarios para implementar de forma eficiente un microcontrolador.

De hecho los microcontroladores, son un caso particular de las computadoras digitales, al igual que estas se tienen que programar para realizar su actividades asignadas en el control de procesos de relativa simplicidad tal como una intersección de tráfico sencilla donde un simple lenguaje de máquina es suficiente para llevar a cabo esta tarea.

ADQUISICIÓN DE DATOS

Veamos ahora una breve descripción de lo que es la adquisición de datos. Los datos analógicos son adquiridos a forma digital para cualquiera de los siguientes propósitos:

Almacenamiento	Procesamiento
Transmisión	Despliegue

Los datos en forma digital pueden almacenarse ya sea de forma directa o procesada; pueden ser retenidos por periodos cortos, medianos o largos. Pueden transmitirse a través de grandes distancias (por ejemplo hacia o desde el espacio exterior), o cortas distancias (de una parte a otra de un instrumento basado en microprocesador). Los datos pueden imprimirse en una impresora o plotter para una copia permanente, o pueden desplegarse sobre panel digital, como parte de una presentación de tubo de rayos catódicos, pueden desplegarse en forma de conversación u otros sonidos, o en cualquier otro modo que estimule los sentidos humanos.

El procesamiento puede ser tan simple como una comparación o bien ser un conjunto de manipulaciones matemáticas complicadas. Uno puede usarlo para propósitos tales como recolectar información, convertir datos a una forma útil, usar los datos para controlar un proceso físico, desarrollar cálculos repetidos con el fin de descubrir señales encubiertas en ruido, generación de información para displays, simplificación de trabajo para los empleados de almacén, control del color de pintura, el ancho de una envoltura, maniobras en un juego, la velocidad de un tren subterráneo.

Pero todo esto empieza con el traslado de los datos a forma digital, tan rápido, tan preciso, tan frecuente, tan completo, y tan barato como sea necesario. El instrumental básico para llevar acabo esta tarea es el convertidor analógico a digital (ADC). Este puede ser un chip de circuito integrado, un digitalizador de flecha, un DPM con salidas digitales, o un dispositivo sofisticado de alta resolución y de alta velocidad; físicamente, este puede tener la forma de una caja, una tarjeta, o un circuito integrado. Este puede estar separado o funcionalmente integrado a otros elementos como es el caso del MC68HC11.

Para acomodar el voltaje de entrada a la relación de conversión específica, alguna forma de escalamiento y de offseting (acondicionamiento de la señal) puede ser necesaria, realizada con un amplificador/atenuador. Para convertir información analógica desde más de una fuente, pueden ser necesarios o convertidores adicionales o bien un multiplexor. Para incrementar la tasa a la cual la información puede ser convertida de forma precisa, será deseable un sample-hold. Para comprimir un rango dinámico analógico extra ancho, es muy útil un amplificador o una relación logarítmica, más adelante hablaremos de la etapa de acondicionamiento de señal.

Las propiedades en las cuales un sistema de adquisición debe depender, contemplan tanto las propiedades de los datos analógicos mismos y lo que se tiene que hacer con estos datos. Este capítulo trata con los aspectos del flujo de la señal, desde los sensores hasta la conversión.

CAPITULO 2

En este capítulo, introduciremos algunas de las arquitecturas funcionales que han probado ser útiles y discutiremos algunas de las consideraciones involucradas en la selección de la configuración, de los componentes y de otros elementos de el sistema.

2.1 ALGO DE HISTORIA

Hace un cuarto de siglo, convertidores analógico-digitales capaces de desarrollar una conversión con una precisión del 0.05% y 50,000 muestras por segundo costaban cerca de \$8000 dólares, consumían cerca de 500 watts, y ocupaban cerca de un cuarto de metro cubico.

Hoy día, un dispositivo de Analog Devices AD573 completamente monolítico, que requiere menos de 20 microsegundos para 10 bits de conversión y que tiene la misma precisión del anterior, que esta empaquetado en plástico de 20 pines DIP, y que esta diseñado para interfaceamiento fácil con un moderno microprocesador cuesta algo menos de el 0.2% del precio del anterior.

El espacio ocupado por solo el convertidor de antaño, ahora puede encerrar (para una inversión en dólares del mismo orden) una MACSYM 150: un sistema completo de multitarea de adquisición de datos basado en una minicomputadora, que incluye, computadora, teclado, exhibidor, unidad de disco, convertidor, y un conjunto de tarjetas de entrada salida.

En los 25 años anteriores, como los ejemplos de arriba lo muestran, la capacidad de procesamiento y la complejidad de la adquisición de datos así como el hardware de la computadora se han incrementado radicalmente, gracias en primera instancia a la revolución de los semiconductores. Software sofisticado y terminales interactivas, las cuales hacen las técnicas informáticas accesibles hasta a los niños pequeños, permiten a las interconexiones, conmutación, y "knob twisting" ser llevados a cabo bajo control remoto o programable. Todo esto (disponible a precios que nunca se soñaron) a sentado la idea de que la forma digital, más que la analógica, para el manejo de información es una cuestión de rutina, más que de necesidad.

Lo que no ha cambiado sin embargo, son los problemas fundamentales del sistema a los que se enfrenta cada diseñador de sistemas digitales de datos. De acuerdo, ayuda mucho tener componentes de bajo consumo de corriente, pequeños, de bajo costo, silenciosos. Pero el diseñador se encuentra aún en contra de las leyes de la madre Naturaleza, quien frecuentemente prefiere guardar sus verdades secretamente oscurecidas por el ruido, por los lazos cerrados de tierra, por el recogimiento de líneas de alimentación, y por los transciendes inducidos en las líneas de señal debidas a la maquinaria. Separando las señales de estos oscuros efectos, se convierte en un reto a la ingenuidad y la imaginación, acopladas con mucha experiencia y persistencia. El diseño no es meramente un materia de comprar convertidores rápidos de alta resolución si no que tenerlos disponibles a un precio real es un incentivo para darles trabajos útiles.

haremos ahora un pequeño recuento de los tópicos que indirecta o directamente involucran a cualquier sistema de adquisición de datos antes de pasar a la descripción de los sistemas de simple y multicanal, a lo que se conoce como acondicionamiento de la señal

2.2 ACONDICIONAMIENTO DE LA SEÑAL Las salidas de los transductores con frecuencia deben de acondicionarse para proveer señales apropiadas para los sistemas de conversión A/D. Los accesorios de acondicionamiento de señal amplifican las señales de bajo nivel, las aíslan y las filtran para producir señales de alto nivel para los sistema de adquisición de señales.

Amplificación El tipo más común de acondicionamiento de una señal es la amplificación. Por ejemplo las señales de bajo nivel de los termopares, deben de amplificarse para incrementar la resolución y reducir el ruido. Para la precisión más alta posible, la señal debe ser amplificada, debido a eso el rango de voltaje máximo de la señal acondicionada debe ser igual a rango máximo del convertidor A/D. Una resolución muy alta reduce la necesidad de las altas amplificaciones y provee un rango dinámico amplio.

En nuestro trabajo no se incluyo la etapa de acondicionamiento de señal porque se tenia contemplada la prueba con un generador de señales, el cual en cuanto a amplificación ya da la salida con la señal acondicionada.

Aislamiento El aislamiento es otra aplicación común para el acondicionamiento de la señal, esta señal proveniente de los transductores debe de aislarse cuando se envía a la computadora por condiciones de seguridad. El sistema que esta siendo monitoreado puede contener transientes de alto voltaje que pueden dañar a la computadora, o bien si el sistema que esta siendo monitoreado es el ser humano mismo como es el caso del registro de un ECG, el aislamiento inadecuado puede incluso producir la muerte del paciente sujeto a registro. Una razón adicional para la necesidad de aislamiento es tener la seguridad de que las lecturas de nuestro dispositivo no son afectadas por diferencias en potenciales de tierra o voltajes a modo común. Cuando la señal de entrada de la tarjeta o el sistema de adquisición de datos están cada una referenciadas a "tierra", ocurren problemas si hay una mínima diferencia de potencial entre ambas tierras. Estas diferencias nos pueden llevar a lo que se conoce como un lazo de tierra, el cual puede causar imprecisión en la representación de la señal adquirida, o si es muy grande puede dañar el sistema de medición. Usar módulos de acondicionamiento de aislamiento de señal elimina los lazos de tierra y asegura que las señales son adquiridas con precisión. En nuestro trabajo no se usa esta etapa de aislamiento porque el generador de señales un Phillips PM-5131 contiene una señal aislada.

Filtrado El filtraje remueve el ruido indeseado de la señal que se esta tratando de medir. Un filtro de ruido se debe usar en señales de clase de DC tales como la temperatura para atenuar las señales de alta frecuencia que pueden reducir la precisión de nuestro sistema. Mucho de este ruido indeseado se puede eliminar con filtros de pasa bajas que van desde frecuencias de 4Hz hasta 10 Khz el cual se utiliza antes de que la señal sea digitalizada. Nuestro sistema tampoco incluye una etapa de filtraje.

Las señales de tipo AC tales como la vibración frecuentemente requieren de otro tipo diferente de filtro conocido como filtro antialias. Como el filtro para el ruido, el filtro antialias es un filtro de

paso bajo; sin embargo este tiene una pendiente de corte muy estrecha, con lo cual remueve caso por completo todos los componentes de frecuencia más grandes que el ancho de banda de la entrada. Si las señales no son removidas, aparecen erróneamente como señales de entrada en el ancho de banda de estudio.

Linearización Otra función del acondicionamiento de la señal es la liberalización, muchos transductores tales como los termopares, tienen una respuesta no lineal a los cambios en el fenómeno que esta siendo medido, algunos software de aplicación como el NI-DAQ de National Instruments incluyen rutinas de linearización para termopares, galgas extensiométricas y RTDs (Resistencias Detectoras de Temperatura).

Excitación El acondicionamiento de la señal en algunos casos incluye la generación de excitación para algunos transductores. Tales como las galgas extensiométricas, los termistores, y los RTDs, los cuales requieren de un voltaje o corriente de excitación externos, algunos equipos comerciales por ejemplo incluyen en el caso de las galgas extensiométricas la configuración del puente de Wheatstone.

2.3 LA IMPORTANCIA DEL MEDIO AMBIENTE

Si bien hay muchas formas de empezar a pensar acerca de los sistemas de adquisición de datos, una aproximación muy relevante tiene que ver con el medio ambiente. Algunos sistemas están diseñados para operar con modesta precisión en medios hostiles (fabricas, vehículos, entornos militares, e instalaciones remotas); otros son más apropiados para hacer medidas de alta precisión en condiciones favorables (probablemente ficticias) tales como laboratorios eléctricamente silenciosos a temperatura ambiente; y otros son más bien una mezcla, que adquieren datos analógicos generados en medios, ruidosos de altas temperaturas, con el procesamiento hecho en la seguridad de un silencioso cuarto de control, y transmitiendo los datos digitales resultantes a través de un mundo común en interferencia.

El medio ambiente debe de dar origen a consideraciones tales como :

- La transmisión de señales Analógicas vs. digitales
- Precisión de la señal vs. Recuperación de la señal
- Aislamiento vs. alambrado directo
- Tamaño macroscópico vs. miniaturización
- Arquitecturas simples vs. complejas
- aproximaciones integradas vs. distribuidas
- procesamiento local vs remoto
- Selección de las fuentes de suministro y el hardware fisico

Los entornos hostiles se manifiestan así mismos en combinación de retos físicos, químicos y eléctricos. Los medios hostiles físicamente pueden exhibir condiciones extremas de temperatura, presión, aceleración, humedad, y radiación (ambas ionizante y no ionizante). Los medios hostiles

CAPITULO 2

químicamente pueden involucrar alrededores tan corrosivos como rociamiento de sal, fluidos biológicos, atmósferas nocivas, suciedad, y gases y fluidos químicamente activos. Los medios hostiles Eléctricamente pueden incluir altos voltajes destructivos y campos magnéticos, así como interferencia por transiente sobre la totalidad del espectro.

Además de estos medios activamente hostiles hay medios ambientes hostiles "pasivos" que deben ser perturbados tanto poco como sea posible físicamente, químicamente y eléctricamente con la introducción de equipo de adquisición de datos. Algunos entornos son hostiles tanto activa como pasivamente.

Los sistemas que existen en ambientes hostiles pueden requerir de dispositivos electrónicos capaces de operar en un rango ancho de temperaturas, con blindaje excelente, esfuerzos considerables de diseño se hacen con la intención de eliminar los errores de modo común y preservar la resolución, la conversión temprana y la transmisión digital de datos (tal vez vía fibra óptica), las trayectorias redundantes para las medidas críticas, y en algunos casos el procesamiento de los datos digitales para extraer el máximo de información.

Las medidas en el Laboratorio , con rangos de temperatura más estrechos y fuentes más pequeñas de interferencia ambiental eléctrica, pueden ser más fácil para hacer y comunicar, pero las altas precisiones (o resoluciones) pueden requerir dispositivos mas sensibles, además de un grado considerable de esfuerzo para preservar tazas apropiadas de señal a ruido.

2.4 FACTORES CLAVE

Los factores ambientales aparte, de la selección de la configuración de los bloques de construcción del circuito en la adquisición de datos dependen de un gran número de consideraciones críticas, entre las cuales están :

Resolución y precisión

El número de canales analógicos que serán monitoreados

Taza de muestreo por canal.

Requerimientos de acondicionamiento de la señal

El costo del funcionamiento

Más que la selección de los niveles apropiados del desempeño de los componentes, el análisis cuidadoso de los factores de arriba es necesario para obtener la configuración del circuito de más bajo costo para obtener el desempeño global deseado.

Los sistemas comerciales de adquisición de datos disponibles van desde los convertidores A/D básicos hasta los convertidores de multicanal en chips monolíticos hasta sistemas de tarjetas completamente integrados o en cajas, y que frecuentemente también incluyen convertidores que son funcionalmente inseparables del procesador digital.

2.5 SISTEMAS DE CONVERSIÓN DE UN SOLO CANAL

tesis profesional "Tarjeta de adquisición de datos utilizando el microcontrolador 8051"

CAPITULO 2

La figura 2.1 representa el sistema de digitalización más simple, un solo convertidor A/D llevando a cabo conversiones repetitivas en corrida libre a una tasa determinada por el tiempo que tarda una conversión completa. Este tiene una entrada de señal analógica así como entradas de alimentación. Sus salidas son en palabras de código digital las cuales pueden incluir una indicación de sobre rango en paralelo, en byte serial, o en forma serial; información de polaridad (si la entrada analógica es bipolar); y el "status" de salida que indica cuando los dígitos de salida se han vuelto validos.

Tal vez el convertidor más conocido de este tipo es el circuito básico de medidor de panel digital, el cual consiste de un convertidor a/d básico y un display numérico. Para muchas aplicaciones, el solo propósito de digitalizar es obtener el despliegue de los dígitos, sin embargo, no es necesariamente la mejor forma de digitalizar un solo canal. Sus mayores desventajas son: su lentitud y su código BCD digital debe cambiarse a forma binaria, si su salida debe ser procesada por equipo binario. Cuando esta en corrida libre con un sistema, su salida es activada en seguimiento a una Interrupción cuando los datos se convierten en válidos, más que de un comando del sistema interfasado.

Los convertidores diseñados para aplicación de sistemas pueden generalmente recibir comandos externos para convertir o para retener esa conversión. Para señales de baja frecuencia o de DC, el convertidor es generalmente de tipo de pendiente dual, el cual tiene la ventaja de que es inherentemente un filtro pasabajas, capaz de dejar fuera el ruido de alta

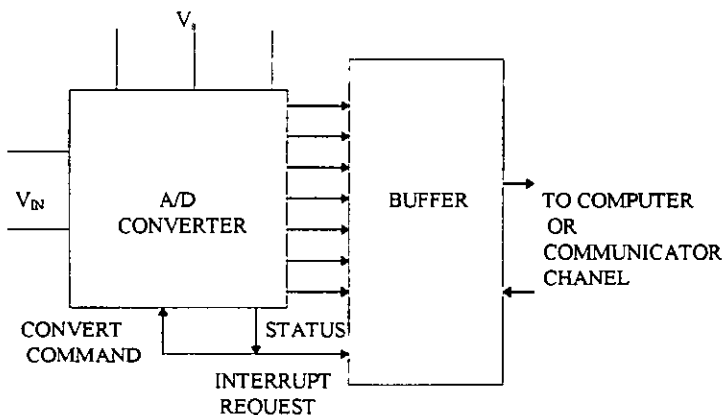


Fig. 2.1 Sistema de digitalización simple

frecuencia y nulificar frecuencias armónicamente relacionadas a su periodo de integración. (Por esta razón , el periodo de integración se hace generalmente igual al periodo de la frecuencia de la línea, debido a que la mayor parte de la interferencia de un sistema así como sus armónicos ocurre generalmente a esta frecuencia.

CAPITULO 2

El valor "actual" de entrada que se convierte por un convertidor de tipo integrativo es representado por un promedio sobre un intervalo de integración de la señal. Debido a que ese intervalo es una fracción (cerca de un tercio, pero no necesariamente constante) del tiempo total requerido por el ciclo de conversión, lo que uno puede decir acerca del valor de la señal y cuando ocurre es que la salida digital representa el valor más probable durante una porción significativa de el período de conversión.

Para voltajes de dc o señales conocidas para tener valores constantes durante el periodo de conversión, aun si los cambios ocurren rápidamente (tan pronto como estén listos antes de que inicie la conversión), cualquier nuevo valor será convertido a una resolución y precisión específica dentro del intervalo de conversión.

Para las conversiones respectivas de las entradas que cambian continuamente, sin embargo la tasa máxima a la cual la señal de entrada puede variar y permitir al convertidor la resolución de 1 bit menos significativo (LSB) de salida binaria, respectivamente a la señal de onda es:

$$dV/dt = 2^{-n}V_{FS}/T$$

donde V es la entrada, n es el numero de bits de la resolución binaria, V_{FS} es la expansión de escala completa, y T es el tiempo entre conversiones. La tasa máxima de cambio es entonces 1 LSB por periodo de conversión.

2.6 SISTEMAS MULTICANAL

En los sistemas de multicanal, los elementos de la cadena de adquisición pueden compartirse por dos o más fuentes de entrada. Esta compartición puede ocurrir en muchas formas, dependiendo de las propiedades deseadas de los sistemas de multiplexaje. Los grandes sistemas pueden combinar muchos tipos diferentes de multiplexaje, así como hileras en cascada de el mismo tipo.

El multiplexaje de las salidas de los convertidores de canal simple.

Si bien la forma convencional de digitalizar los datos provenientes de muchos sistemas analógicos es introducir un proceso de tiempo compartido a la porción analógica del sistema por medio del multiplexaje de la entrada de un solo convertidor A/D entre varios fuentes analógicas, en secuencia, un proceso alternativo de conversión paralela se convierte cada vez más en algo muy práctico. El costo de los convertidores A/D ha caído vertiginosamente en los años recientes, y es ahora posible ensamblar un sistema de conversión multicanal, con la aparente extravagancia de usar un convertidor para cada fuente analógica, con lo que se mejora el desempeño considerablemente a un costo razonable.

Este acercamiento por la conversión paralela tiene sus ventajas. Primero, una tasa global de conversión digital puede tenerse con convertidores lentos; alternativamente, un convertidor por canal puede correr a una velocidad tope, proveyendo un flujo de datos más grande en la interface digital. Para una tasa de datos moderada, sin embargo, con más canales (y menores conversiones por canal), puede ser posible eliminar los procesos de sampling-hold, salvando costos. Menores

CAPITULO 2

conversiones pueden también significar que un convertidor más lento pueda ser usado, generalmente resultando en futuros ahorros de dinero, especialmente debido a que algunos canales pueden no requerir alta resolución.

Las estructuras de bus usadas por sistemas que emplean microprocesadores alientan el uso del multiplexaje digital, con todos los dispositivos conectados al bus vía switches de tres estados, que permiten la selección por medio de la señal del pin "chip select" proveniente de los decodificadores y las señales de control read/write (write para iniciar la conversión, y read para obtener los resultados). La línea de estado del convertidor puede proveer señales de interrupción que indican "la conversión esta completa los datos están listos".

La conversión paralela provee una ventaja posterior cuando se aplican a sistemas de adquisición de datos industriales, donde las galgas extensiométricas, los termopares, los termistores, etc., se encuentran extendidas en una gran área geográfica. En esencia digitalizando las señales analógicas justo en sus fuentes y transmitiéndolas de en forma de datos seriales, más que las señales analógicas de bajo nivel, se consigue una inmunidad considerable al ruido de la frecuencia de la línea (50-60-400Hz) y las interferencias debidas a los lazos de tierra. Entre otros factores, las señales digitales pueden acoplarse ópticamente o bien transmitirse vía líneas de fibra óptica para aislamiento eléctrico completo e indiferencia total a la interferencia eléctrica.

Un arreglo multicanal de convertidores de voltaje a frecuencia (v/f) es un medio interesante de transmitir los datos generados por señales que varían lentamente, con rangos dinámicos de arriba de 10^6 y que requieren de precisiones de cerca 0.01%. Las salidas son trenes de pulsos TTL, las cuales pueden ser aisladas ópticamente. La salida de cada VFC, en turno, es contada y leída. En este caso la computadora controla al multiplexor y actúa como base de tiempo para el contador.

No al menos, dentro de los sutiles beneficios de digitalizar las señales de los sensores en su fuente, esta la habilidad de desarrollar operaciones lógicas en los datos digitalizados antes de ser alimentados en la computadora. Más específicamente, el procesamiento remoto hace posible, por ejemplo, el acceso a datos provenientes de la lenta variación de un termopar de menos frecuencia, mientras se leen datos a los cambios rápidos de la fuentes críticas a velocidades mejoradas. De hecho la versatilidad de un subsistema digital debe de ser explotada para hacer la propia decisión así como también cuando un canal particular de datos debe de atraerse de la atención de la computadora por medio de una interrupción. Si ciertas señales permanecen constantes o dentro de un rango muy estrecho de para largos periodos de tiempo, y después cambia rápidamente en el proceso, es posible ignorar estos datos hasta que los cambios ocurren. (Una computadora local puede almacenar los valores estacionarios y tomar decisiones).

En resumen, un gran reto de flexibilidad y versatilidad se gana cambiando la interface del proceso de multiplexaje analógico a multiplexaje digital. Los circuitos de decisiones lógicas o los microprocesadores locales pueden ejercer juicios aun cuando los datos a alimentarse a la computadora y, cuando en general, puede dar la interface global a mucha más grande medida de la autonomía que puede ser posible más que con un completo sistema analógico de conversión

CAPITULO 2

Finalmente se debe notar, que si , por ejemplo, los datos están siendo transmitidos desde un vehiculo lunar hacia la tierra, el canal esta totalmente atisbado de gente, y el tópico de redundancy-reduction data compresión descritos arriba es absolutamente posible)

En el siguiente capítulo pasaremos a una descripción más detallada de los elementos que harán posible nuestro sistema de adquisición de datos el cual es de un solo canal y no toma en cuenta el transductor todavía por tratarse de un sistema general en el cual consideramos sería muy tedioso considerar todas las configuraciones posibles así como las formas de acondicionamiento de señal necesarias que se pueden requerir en un laboratorio como lo es del de termofluidos.

3.1 CARACTERÍSTICAS DEL 8051

La familia de μ c-8051 es variada, y se encuentra en diversas presentaciones, la selección de uno o de otro tipo de microcontrolador dependerá principalmente de las necesidades a satisfacer. En este trabajo hablaremos de manera particular del 8051 el cual se presenta en 3 versiones, con ROM interna (8051) la cual es programada directamente por el fabricante, con EPROM interna (8751) que puede ser programada por el usuario y sin PROM ni EPROM (8031) en el cual, el programa se selecciona de manera externa.

El μ c-8051 esta basado en los microprocesadores de 8 bits, contiene internamente un CPU de 8 bits, 3 puertos de entrada y salida paralelo, un puerto de control, el cual a su vez contiene; un puerto serie, 2 entradas para Timer/contador de 16 bits, 2 entradas para interrupciones externas, las señales de RD y WR para la toma o almacenamiento de datos externos en RAM, la señal de PSEN para la lectura de instrucciones almacenadas en EPROM externo. Gracias a estas 3 señales el μ c-8051 puede direccionar 64K de programa y 64K de datos separadamente, es decir un total de 128 Kb. Además cuenta con 128 bytes de memoria RAM interna.

El μ c-8051 puede generar la frecuencia (Baud Rate) de transmisión/Recepción de datos por el puerto serie, de manera automática partiendo de la frecuencia del oscilador general, por medio de la programación del Timer 1. Dicha frecuencia de transmisión puede ser cambiada en cualquier momento con sólo cambiar el valor almacenado en el contador, o también se puede duplicar o dividir la frecuencia con solo escribir directamente sobre el bit 7 (SMOD) del registro de control PCON.

A continuación comenzaremos a ver con mayor detalle todo lo referente a sus conexiones como características especiales del μ c-8051.

3.2 DESCRIPCIÓN DE LAS LÍNEAS (PINS) DEL 8051.

El elemento más básico de la familia 8051 es el 8031, que carece de EPROM o PROM, el cual es direccionado externamente.

El 8031 es fundamentalmente un chip de 40 líneas, Fig. 3.1

DESCRIPCIÓN DE CONEXIONES.

VSS Tierra 0V referencia.

P0.0 - P0.7 Es un puerto bidireccional con salidas en e_n colector abierto. Cuando el puerto esta en alto, las salidas están flotadas y pueden servir como entradas de alta impedancia. El puerto 0 es también multiplexado para obtener el DATO y la parte baja de la dirección.

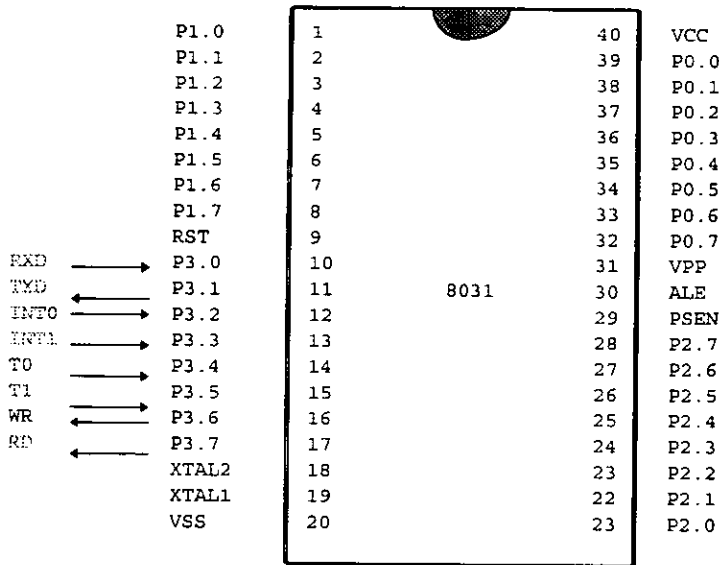


Figura 3.1 MICROCONTROLADOR 8051

P1.0 - P1.7 Es un puerto quasi-bidireccional, cuando esta en alto, el puerto puede ser utilizado como entrada

P2.0 - P2.7 Es un puerto quasi-bidireccional con fijadores de nivel internos (PULL-UP). Cuando se escriben 1's sobre el puerto, las líneas pueden ser utilizadas como entradas o como salidas. Como entradas las líneas que son externamente colocadas en la posición baja proporcionarán una corriente hacia el exterior. El puerto 2 es utilizado además para direccionar memoria externa. Este puerto, emite el byte más alto de la dirección durante la búsqueda de datos en la memoria del programa externo y durante el acceso a memorias de datos de 16 bits. Durante el acceso a una memoria de datos externa que usa direcciones de 8 bits, el puerto 2 emite el contenido del registro correspondiente a este puerto, que se encuentra en el espacio de funciones especiales.

P3.0 - P3.7 Es un puerto quasi-bidireccional con fijadores de nivel internos (PULL-UP). Cuando se escriben 1's sobre el puerto, las líneas que son externamente colocadas en la posición baja proporcionarán una corriente hacia el exterior. El puerto 3 es utilizado además para producir señales de control de dispositivos externos como son las siguientes:

RXD(P3.0) : Puerto serie de entrada

TXD(P3.1) : Puerto serie de salida

INTO(P3.2): Interrupción externa
INT1(P3.3): Interrupción externa
T0 (P3.4): Entrada Externa timer 0
T1 (P3.5): Entrada Externa timer 1
WR (P3.6): Habilitador de escritura para memoria externa de datos.
RD (P3.7): Habilitador de lectura para memoria externa de datos.

RST Reset. Una entrada alta en esta línea durante dos ciclos de máquina, mientras el oscilador esta corriendo, detiene el dispositivo. Un resistor interno conectado a Vss permite un alto en la fuente usando solamente un capacitor externo a Vcc.

ALE Address Latch enable. Un pulso positivo de salida permite fijar el byte bajo de la dirección durante el acceso a una memoria externa. En operación normal, ALE es emitido en un rango constante de 1/6 de la frecuencia del oscilador, y puede ser usada para cronometrar. Note que un pulso del ALE es omitido durante cada acceso a la memoria de datos externos.

PSEN Program Store Enable. Habilitador de lectura para memoria de programas externos. Cuando el 8031B/8051 esta ejecutando un código de una memoria de programas externos, PSEN es activada dos veces cada ciclo de máquina, excepto cuando se accesa la memoria de datos externos que omiten las dos activaciones del PSEN externos. PSEN tampoco es activada cuando se usa la memoria de programas internos.

EA External Access Enable debe mantenerse externamente en posición baja para habilitar el mecanismo que elige el código de las localizaciones de la memoria de programas externos, 0000H y 0FFFH. Si EA se mantiene en posición alta, el dispositivo ejecuta los programas que se encuentran en la memoria interna ROM, a menos que el contador del programa contenga una dirección mayor a 0FFFH.

XTAL1 Crystal 1 Es la entrada del cristal para el circuito oscilador (generador del reloj interno) que amplifica e invierte la entrada.

XTAL2 Crystal 2 Es la salida del amplificador oscilador inversor.

3.3 CARACTERÍSTICAS ESPECIFICAS DEL 8051

El 8051 contiene las siguientes características:

- 1 CPU de 8 bits como parte central
- 32 líneas bidireccionales de entrada y salida (4 puertos)
- 128 bytes de memoria RAM
- 2 Contadores / Timers de 16 bits
- 1 UART completo
- 5 estructuras de interrupción con dos niveles de prioridad

- 1 circuito de reloj
- 64 Kbytes de espacio para programa
- 64 Kbytes de espacio para datos

Descripción de los espacios de memoria La memoria del sistema del 8051 se clasifica en tres partes fundamentales : La primera, llamada memoria de programa, en donde se encuentran todas las instrucciones que van a ser ejecutadas por el μ c-8051, es decir, el programa de trabajo. Algunas versiones del 8051 cuentan con memoria de programa interna (de 2 a 4 Kb). Cuando se requiere trabajar con una localidad arriba de ésta, la memoria del programa (externa) es seleccionada mediante la activación de la señal PSEN (estado bajo). El máximo espacio de memoria de programa que se puede acceder es de 64KB.

El segundo espacio de memoria denominado, memoria de datos es accesado mediante la activación de las señales RD y WR, durante la lectura o escritura de datos respectivamente. En este espacio el μ c toma todos los valores que se encuentran en memoria como DATOS, es decir, el μ c no puede ejecutar ninguna instrucción que se encuentre aquí almacenada. El 8051 puede direccionar también 64KB de memoria de datos.

El tercer espacio de memoria es el denominado memoria RAM interna, el cual se divide en 128 bytes de memoria bajos y en 128 bytes de memoria altos. En los primeros 128 bytes, se encuentran 4 bancos de 7 registros cada uno. Estos registros son de gran ayuda para la simplificación de los programas, debido a que cada uno de ellos nos permite almacenar datos momentáneamente y realizar un basto número de instrucciones del 8051. También dentro de este espacio, se encuentran 16 bytes (del 20H al 2FH) que pueden ser direccionados directamente por bit.

En la parte alta de la memoria RAM interna, se encuentra el contenido de los Registros de Funciones Especiales, formado por Puertos, Registros de Control, Acumuladores, Registros de interrupción, etc. Todos estos registros los veremos detalladamente posteriormente.

3.4 MEMORIA DEL PROGRAMA.

Localidades asignadas a las interrupciones A continuación veremos de manera más detallada el espacio destinado al programa de trabajo.

La tabla 3.1, muestra las localidades que has sido asignadas por el fabricante, para dar servicio a las rutinas de interrupción.

FUENTE DE INTERRUPCIÓN	VECTOR DE DIRECCIONES
IE0 (interrupción 0 externa)	0003H
TF0 (interrupción del timer 0)	000BH
IE1 (interrupción 1 externa)	0013H
TF1 (interrupción del timer 1)	001BH
RI y T1 (interrupción serie)	0023H
TF2 y EXF2 (sólo para el 8052)	002BH

TABLA 3.1 INTERRUPCIONES

CAPITULO 3

Por el contrario, si la línea EA = 0, el CPU seleccionará de forma externa el ROM, desde la dirección 0000H hasta FFFFH.

En el caso del 8031 ésta línea se conecta siempre a 0 volts (Vss). La línea PSEN (Program Store Enable), que sirve para leer el ROM externo, es activado en todas las búsquedas (Fetches) del programa. PSEN NO SE ACTIVA en búsquedas (fetches) del ROM interno. La fig. 3.2, muestra un conexionado a una EPROM externo.

3.5 MEMORIA DE DATOS (DATA MEMORY)

El espacio de memoria RAM interno está dividido en tres espacios, el primer bloque es referido como parte baja de 128 bytes, el segundo (se tiene sólo en algunas versiones del 8051 v.gr. 8052), la parte alta de 128 bytes y el tercero, llamado espacio SFR (Registros de Funciones Especiales).

Las direcciones de la Memoria Interna de Datos siempre son de un byte (de 00H a FFH). Sin embargo los modos de direccionamiento para la memoria interna pueden acomodar hasta 384 bytes, como se ve en la versión 8052, lo cual es posible debido a que el modo de direccionamiento directo accede un espacio de memoria diferente físicamente al permitido por el modo de direccionamiento indirecto.

Los primeros 128 bytes, son presentados en todos los dispositivos de la familia MCS-51, que están mapeados. Como se puede apreciar en la figura 3.3, los 128 bytes más bajos son divididos en 4 bloques de 8 registros cada uno, que contienen los valores de los registros R0 a R7, los bloques pueden ser seleccionados mediante escritura de los bits 3 y 4 del registro PSW (Palabra del Estado del Programa), el cual veremos más adelante. La utilización de registros permiten un uso más eficiente del espacio de códigos debido a que sus direccionamientos son de 8 bits únicamente.

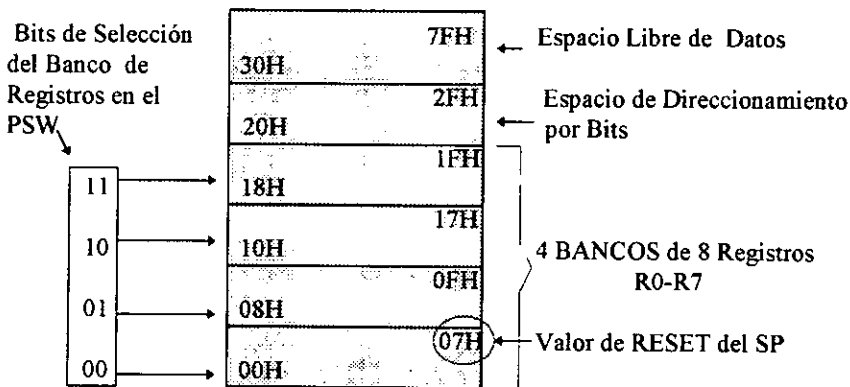


Figura 3.3 Distribución de los 128 bytes más bajos de la memoria RAM interna

Como ya habíamos mencionado anteriormente, existen algunas versiones del 8051, como el 8052, que contienen 128 bytes de memoria interna que puede ser direccionada indirectamente. Por otro lado,

CAPITULO 3

todas las versiones del 8051 contienen un espacio de 128 bytes en la parte alta de la memoria que son direccionados directamente, en este espacio se localizan los Registros de Funciones Especiales (SFR). Estos registros especiales, tienen sus localidades bien establecidas, y son utilizados por el microcontrolador para realizar las distintas operaciones internas que ejecuta el microcontrolador, así como también para el control y acceso de los diferentes puertos de entrada y salida.

3.6 LOCALIDADES DE LOS REGISTROS DE FUNCIONES ESPECIALES

En la sección anterior habíamos visto que los SFR se encuentran en la parte alta (128 bytes) de la memoria RAM interna del 8051. Las direcciones de los SFR se muestran en la tabla siguiente.

En la Tabla 3.2 se muestra el mapa de memoria correspondiente a los registros de funciones especiales.

A continuación veremos con mayor detalle cada uno de los principales registros de control del 8051.

3.7 REGISTRO DE PALABRA DEL ESTADO DEL PROGRAMA (PROGRAM STATUS WORD)

El registro de palabra del estado del programa contiene algunos bits que reflejan el estado del CPU en ese instante. El PSW se muestra en la figura 3.4. Este registro como ya se vió, reside en

SIMBOLO	NOMBRE	DIRECCION
ACC	Acumulador	0E0H
B	Registro B	0F0H
PSW	Program Estatus Word (Palabra de estado del programa)	0D0H
SP	Stack Pointer (apuntador de pilamiento)	81H
DPTR	Data Pointer (apuntador de datos) 16 bits	
DPL	Data Pointer low byte	82H
DPH	Data Pointer high byte	83H
P0	Puerto 0	80H
P1	Puerto 1	90H
P2	Puerto 2	0A0H
P3	Puerto 3	0B0H
IP	Control de Prioridad de Interrupciones	0B8H
IE	Control de validación de la Interrupción	0A8H
TMOD	Modo de Control Timer/Contador	89H
TCON	Control del Timer/Contador	88H
TH0	Byte alto del T/C 0	8CH
TL0	Byte bajo del T/C 0	8AH
TH1	Byte alto del T/C 1	8DH
TL1	Byte bajo del T/C 1	8BH
SCON	Control de la transmisión Serie	98H
SBUF	Buffer de datos serie	99H
PCON	Control de Potencia	87H

Tabla 3.2 Espacio de los registros de funciones especiales.

CAPITULO 3

el espacio de SFR. El registro contiene; el bit de Carry, el bit auxiliar (para operaciones BCD), los dos bits de selección del banco de registros, la bandera de overflow, el bit de paridad y dos banderas sin definir.

REGISTRO PSW

CY	AC	F0	RS1	RS0	OV	-	P
----	----	----	-----	-----	----	---	---

Figura 3.4 Registro PSW (Palabra de Estado del Programa)

CY	PSW.7	BANDERA DEL CARRY
AC	PSW.6	BANDERA DEL CARRY AUXILIAR (operaciones en BCD)
F0	PSW.5	BANDERA 0 PARA USOS GENERALES
RS1	PSW.4	BIT 1 SELECTOR DEL BANCO DE REGISTROS
RS0	PSW.3	BIT 0 SELECTOR DEL BANCO DE REGISTROS
OV	PSW.2	BANDERA DE OVERFLOW
-	PSW.1	BANDERA SIN DEFINIR
P	PSW.0	BANDERA DE PARIDAD SE ESTABLECE Y LIMPIA POR HARDWARE, INDICA SI EL ACUMULADOR ES PAR O IMPAR.

F8								FF
F0	B							F7
E8								EF
E0	ACC							E7
D8								DF
D0	PSW							D7
C8	T2CON				TL2	TH2		CF
C0								C7
B8	IP							BF
B0	P3							B7
A8	IE							AF
A0	P2							A7
98	SCON							9F
90	P1							97
88	TCON	TMOD	TLO	TL1	THO	TH1		8F
80	P0	SP	DPL	DPH			PCON	87

Tabla 3.3 Mapa de memoria de los SFR

El bit de paridad refleja el número de 1s, en el acumulador: P=1, si el Acumulador contiene un número par de 1s, es decir el número de 1s, en el acumulador más P es siempre par.

3.8 REGISTRO DE CONTROL DE POTENCIA (CONSUMO DE ENERGIA)

CAPITULO 3

En la figura 3.5 tenemos el registro PCON, el cual a excepción de la bandera SMOD, sirve para controlar, principalmente el consumo de energía, el cual es utilizado solo para los dispositivos fabricados con la tecnología CHMOS que permite disminuir dicho consumo de energía, en estados de espera. La bandera PCON.7 (SMOD) sirve para dividir la frecuencia de transmisión o de recepción por el puerto serie, proporcionada ya sea por, por la fase 2 de los estados. (1/2 de la frecuencia del oscilador en la transmisión serie en modo 2), o bien por el timer 1 en los modos 1 y 3.

SMOD	-	-	-	GF1	GF0	PD	IDL
------	---	---	---	-----	-----	----	-----

Figura 3.5 PCON (Registro del Control de Potencia)

SMOD	Dobla el "BAUD RATE" para el puerto serie cuando se utiliza el timer para generar el BAUD RATE.	
GF1	Propósitos generales	
GF0	Propósitos generales	
PD	Bajo Consumo de energía	80C51BH CHMOS
IDL	Bajo Consumo de energía	

3.9 REGISTRO DE INTERRUPCIONES

Las interrupciones son controladas mediante la escritura en los registros IE (interrupción Enable) e IP (Interruption Priority) los cuales son físicamente representados en la fig. 3.6

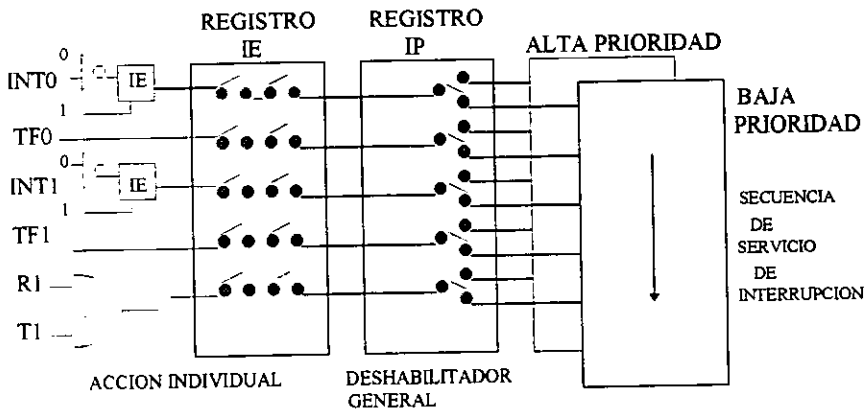


Figura 3.6 Registro de Interrupciones

Registro Habilitador de interrupciones (IE) La desactivación general de las interrupciones es efectuada mediante la escritura de un 0 lógico, en la bandera EA (IE.7) del

CAPITULO 3

registro habilitador de interrupciones. Con la bandera EA=1, el 8051 está en condiciones de aceptar interrupciones, aunque la verdadera aceptación se realiza cuando se escribe un 1 lógico, en la bandera de la interrupción correspondiente al registro de interrupciones, IE

EA	-	ET2	ES	ET1	EX1	ET0	EX0
----	---	-----	----	-----	-----	-----	-----

Figura 3.7 IE (Registro Habilitador de Interrupciones)

EA	IE.7	Desactiva todas las INTERRUPCIONES cuando EA=0
ET2	IE.5	Activa la interrupción causada por el timer 2 (ET2=1)
ES	IE.4	Activa la interrupción causada por el puerto serie
ET1	IE.3	Activa la interrupción de sobreflujo causada por el timer 1
EX1	IE.2	Activa la interrupción causada externamente por INT 1
ET0	IE.1	Activa la interrupción de sobreflujo causada por el timer 0
EX0	IE.0	Activa la interrupción causada externamente en INT 0

Registro de Prioridad El 8051 tiene dos planos de prioridad para trabajar las interrupciones, llamados alto y bajo, respectivamente. En la inicialización, todas las interrupciones trabajan en el plano de baja prioridad. Para pasar del plano de baja prioridad al de alta, es necesario escribir un 1 lógico en las banderas correspondientes a las interrupciones que se desean aumentar de prioridad ubicadas dentro del registro IP

-	-	PT2	PS	PT1	PX1	PT0	PX0
---	---	-----	----	-----	-----	-----	-----

Figura 3.8 Registro de prioridad de interrupciones

PT2	IP.5	Timer 2 PT2=1 mayor prioridad
PS	IP.4	Define el nivel de prioridad de la interrupción del puerto serie
PT1	IP.3	Define el nivel de prioridad de la interrupción del Timer 1
PX1	IP.2	Define el nivel de prioridad de la interrupción 1 externa
PT0	IP.1	Define el nivel de prioridad de la interrupción del Timer 0
PX0	IP.0	Define el nivel de prioridad de la interrupción 0 externa

Aunque los registros de control de los puertos del Timer/Contador y Serie pertenecen a los registros de funciones especiales les dedicaremos un capítulo completo a cada uno de ellos debido a la importancia que representan para el desarrollo e interconexión con sistemas periféricos.

4.1 TIMER / CONTADOR.

En contraste al concepto de Motorola de un solo timer de 16 bits con múltiples registros de comparación y captura, el 8051 tiene 2 timer/contadores de 16 bits cada uno, llamados Timer 0 y Timer 1 respectivamente que pueden ser operados en varios modos. Ambos pueden ser configurados para operar como temporizadores (timers) o como contadores (counters).

Cuando se trabaja como contador, el registro interno del contador, es incrementado cada vez que existe una transición negativa (de 1 a 0) por la línea de entrada correspondiente a T0 ó T1. En cambio, cuando funciona como temporizador "Timer", el registro es incrementado cada 12 periodos de oscilación es decir su frecuencia de conteo es 1/12 de la frecuencia del oscilador.

En el momento que los bits del registro del contador pasan de todos 1's a todos 0's, se activa la línea de interrupción interna correspondiente a TF0 ó TF1, generándose, (si ha sido permitida) una interrupción.

4.2 REGISTRO DE CONTROL DEL PUERTO TIMER/CONTADOR (TCON).

El registro de control del Timer/Contador de la fig. 4.1 es direccionable por Bit, para activar o desactivar cada una de sus banderas.

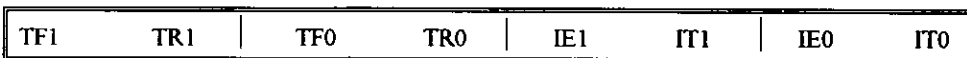


Fig 4.1 Registro de Control del Timer/Contador

- TF1** Bandera de sobreflujo (overflow) del registro del Timer 1. Activada por hardware cuando el registro que guarda la cuenta del timer/cotador 1, incrementa su contenido pasando todos sus bits de 1's a 0's. Limpiado por hardware cuando existe el procesamiento de los vectores del servicio de las rutinas de interrupción.
- TR1** Bit de control de activación del timer 1, Habilitado/Deshabilitado por software para colocar el Timer/contador en Encendido/Apagado.
- TF0** Bandera de sobreflujo (overflow) del registro del timer 0. Activada por hardware cuando el registro que guarda la cuenta del timer/contador 0, incrementa su contenido pasando todos sus bits de 1's a 0's. Limpiado por hardware cuando existe el precesamiento de los vectores del servicio de las rutinas de interrupción.
- TR0** Bit de control de activación del timer 0, Habilitado/Deshabilitado por software para colocar el Timer/contador en Encendido/Apagado.

CAPITULO 4

- IE1** Bandera de transición de la interrupción externa 1. Activada por hardware cuando una transición (de 1 a 0) en la línea de interrupción externa 1, es detectada. Limpiada por hardware cuando la interrupción es procesada. (solamente se acciona si se programó la aceptación de la interrupción por transiente, IT1=1).

- IT1** Bit de control del interrup 1. Activado/limpiado por software para especificar el tipo de interrupción, por nivel bajo (IT1 = 0) o por transiente negativo (IT1=1).

- IE0** Bandera de transición de la interrupción externa 0, Activada por hardware cuando una transición (de 1 a 0) en la línea de interrupción externa 0, es detectada. Limpiada por hardware cuando la interrupción es procesada. (solamente se acciona si se programó la aceptación de la interrupción por transiente,IT0=1).

- IT0** Bit de control del Interrup 0. Activado/Limpiado por software para especificar el tipo de interrupción, por nivel bajo (IT0 =0) o por transiente negativo (IT0=0).

4.3 REGISTRO DE MODO DE CONTROL DEL TIMER/CONTADOR (TMOD).

Este registro permite especificar si se van a trabajar como Temporizadores (Timers) o como Contadores (counters), los puertos denominados Timer 0 y Timer 1.

Existen 4 modos de trabajo para estos puertos, los cuales son definidos por la escritura en los bits M1 y M0 de TMOD fig. 4.2.

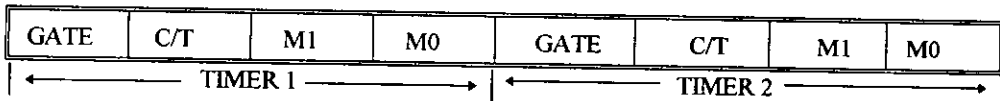


Fig. 4.2 TMOD: Registro del Modo de control del Timer/Counter

- GATE** Cuando TRx (en TCON) esta activada y GATE = 1, TIMER/Counterx correrá. Solamente si la línea INTx esta en posición alta (Control por hardware). Cuando GATE=0, TIMER/COUNTERx correrá solamente si TRx=1 (control por software)
- C/T** Selector de Timer o de contador. Se limpia para la operación del Timer (entrada del reloj del sistema interno). Se activa para la operación del contador (entrada de la línea Tx).

- M1** Bit selector del modo.

- M0** Bit selector del modo.

Modo 0, Del TIMER/CONTADOR En este modo, cualquiera de los 2 Timers, 0 ó 1, trabajan como un contador de 8 bits, al cual le antecede un predivisor de la frecuencia de conteo. En la fig 4.3 se muestra el modo de operación 0 para el timer 1.

M1	M2	MODO	ESPECIFICACION
0	0	0	Timer/contador de 13 bits
0	1	1	Timer/contador de 16 bits
1	0	2	Timer/contador de 8 bits recargables
1	2	3	Timer 0, TL0 Timer/contador de 8 bits

Tabla 4.1 Configuración del modo de trabajo del registro del Timer / Contador

El registro del Timer 1 está configurado como un registro de 13 bits, que consisten de los 8 bits de TH1 y los 5 bits menos significativos de TL1. Los 3 bits más significativos de TL no se utilizan en este modo.

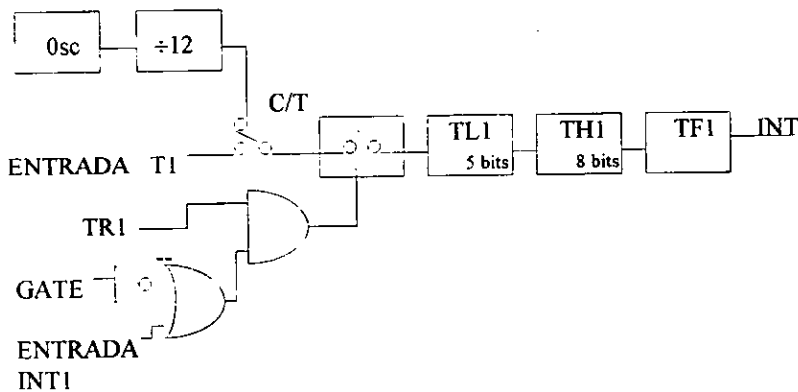


Fig. 4.3 Modo 0, contador de 13 bits con el Timer1

El valor de la cuenta se puede apreciar en el registro TH1, dado que TL1 solamente actúa como divisor de frecuencia, TH1 se puede cargar con cualquier valor, de 1 a 256 cuentas, pudiendo obtener de esta forma varios retardos, solamente detectando la bandera de sobreflujo (overflow TF1).

Modo 1 Del TIMER/CONTADOR Este modo es utilizado por cualquiera de los 2 Timers, se caracteriza principalmente por ser un Timer/contador de 16 bits cuyos valores se encuentran cargados en los registros TH y TL de cada uno de los Timers.

En la fig. 4.4 se puede apreciar la disposición de estos registros que se asemejan al modo 0, sólo que en el modo 1, actúan en cascada.

Modo 2 Del TIMER/CONTADOR Este modo puede ser utilizado tanto por el timer 0 como por el 1, tiene un registro de conteo de 8 bits (TLx).

En la figura 4.5 se presenta el manejo del Timer 1. en el modo 2. El registro TL1 es cargado automáticamente con el contenido de TH1, cuando se produce el sobreflujo en TL1, el cual además establece la bandera de TF1.

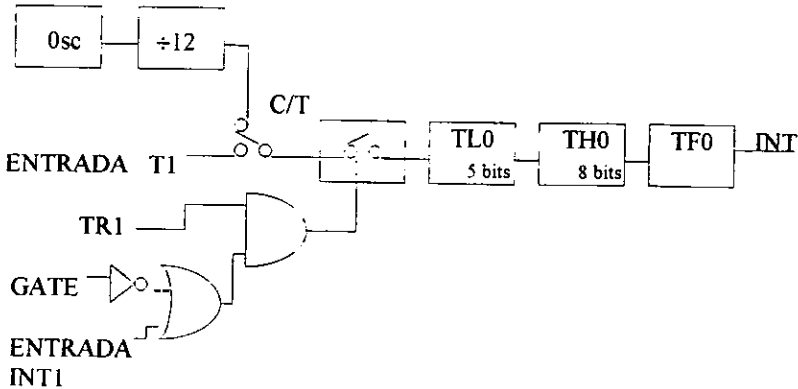


Fig 4.4 Modo 1 Contador de 16 bits con Timer/Contador 1

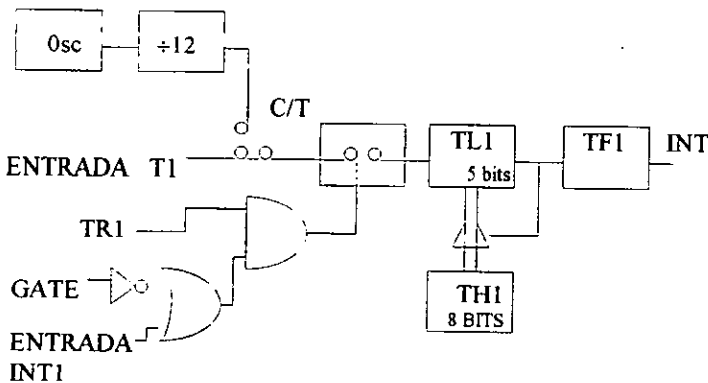


Fig. 4.5 Modo 2, registro de 8 bits autorecargable

Modo 3 Del TIMER/CONTADOR El Timer 1, en el modo 3 mantiene su cuenta, es decir, tiene el mismo efecto que cuando se establece la bandera TR1=0.

El Timer 0, en este modo, establece TL0 y TH0 como dos contadores separados. La figura 4.6 muestra la lógica para el modo 3. TL0 0 utiliza los bits de control (C/T, GATE, TR0, INTO) del Timer 0. TH0 es bloqueado como temporizador "timer", el cual emplea las señales de control del Timer 1, TR1 y TF1.

El timer 1 puede ser activado o desactivado con sólo salir o entrar al modo 3 respectivamente o puede permanecer siendo utilizado por el puerto serie cuando está generando la frecuencia de oscilación "Baud rate", o en efecto en cualquier aplicación que no se requiere una interrupción.

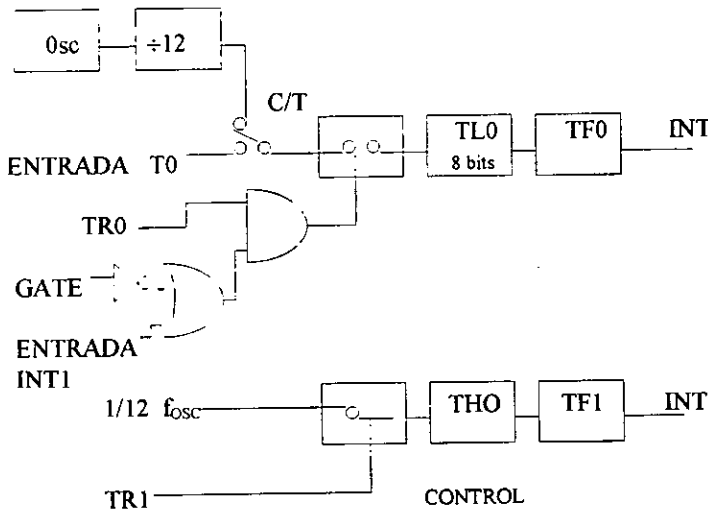


Fig. 4.6 Modo 2, 2 contadores de 8 bits

4.4 UTILIZACION DEL TIMER 1 COMO GENERADOR DEL "BAUD RATE" PARA LA TRANSMISION SERIE.

El timer 1 es usado para generar la frecuencia de transmisión o recepción de datos serie, cuando el puerto es programado para trabajar en el modo 1 ó 3. La frecuencia de transmisión es obtenida a partir del valor almacenado en TH1 y el valor de SMOD mediante la ecuación siguiente:

$$BAUD\ RATE = \frac{2^{SMOD}}{32} \times \frac{frec.\ del\ oscilador}{12 \times [256 - (TH1)]}$$

CAPITULO 4

El valor $2^{SMOD} / 32$, es debido a los circuitos divisores de frecuencia que se encuentran en la etapa de control del puerto serie, los cuales dividen entre 16 ó 32 dependiendo del bit 7 (SMOD) del registro de control PCON y la frecuencia que nos proporciona la salida del Timer 1 (overflow).

El valor 12 que divide a la frecuencia del oscilador proviene del divisor, que se encuentra en la etapa de control del Timer 1, cuando éste es utilizado como temporizador.

NOTA: Cabe recalcar que el valor que se almacena en TH1 es el valor negativo de la cuenta que se desea, debido a que, el contador se incrementa cada vez que un pulso es detectado, de ahí, que en la ecuación se representa como $256 - (TH1)$.

La interrupción del Timer 1 en este caso no tendría mucha aplicación por lo que se podría deshabilitar. El timer 1 actúa en modo 2, es decir en modo recargable, el valor de conteo se encuentra fijo en el registro TH1, el cual se recarga cada vez que existe overflow.

La Tabla 4.2 muestra un rango de valores de TH1, para generar el Baud Rate, tomando en cuenta la frecuencia del oscilador.

Baut Rate	fosc	SMOD	TIMER 1		
			C/T	MODE	Valor de TH1
19.2 kHz	11.059 Mhz	1	0	2	FDH
9.6 KHz	11.059 Mhz	0	0	2	FDH
4.8 KHz	11.059 Mhz	0	0	2	FAH
2.4 KHz	11.059 Mhz	0	0	2	F4H
1.2 KHz	11.059 Mhz	0	0	2	E8H
137.5 KHz	11.059 Mhz	0	0	2	1DH
110 Hz	6.000 MHz	0	0	2	72H

Tabla 4.2 Valores para generar el Baud Rate

4.5 PUERTO SERIE

El puerto serie es un puerto "FULL DUPLEX", lo cual significa que puede transmitir y recibir datos simultáneamente. El receptor contiene un almacén "Buffer", que le permite comenzar a recibir un segundo dato sin necesidad de que el primero haya sido completamente leído del registro del Buffer. Sin embargo si el primer byte permanece sin ser leído hasta el final de la recepción del segundo dato, éste se perderá.

El dato de la Recepción y de la Transmisión se encuentra en el registro SBUF del SFR (espacio de funciones especiales).

4.6 REGISTRO DE CONTROL DEL PUERTO SERIE SCON.

Como se había mencionado, el chip tiene implementada una interfase serial asincrona "Full Duplex". El receptor tiene un buffer doble por lo que puede recibir un segundo dato mientras que el primero se

retiene en un registro intermedio hasta que el software lo solicita. Este buffer doble reduce críticamente el tiempo de servicio de la interface serial. Al igual que con el timer el puerto serie puede ser operado en 4 modos diferentes, que son especificados mediante la escritura en los bits SM0 y SM1 del Registro de Control del Puerto Serie (fig. 4.7)

SM0	SM1	SM2	REN	TB8	RB8	TI	RI
-----	-----	-----	-----	-----	-----	----	----

Fig. 4.7 Registro de control del puerto serie. Bit direccionable

- SM0** Especifica el modo de control del puerto serie.
- SM1** Especifica el modo de control del puerto serie.
- SM2** Habilita la comunicación del tipo "multiprocesador" utilizado en los modos 2 y 3. En estos modos, si SM2 =1, RI no es activado si el noveno dato recibido RB8 es 0. En modo 1, RI no es activado si no se recibe un bit stop. El modo 0, SM2 será 0.
- REN** Establece la recepción serie, cuando REN=0 se desactiva la recepción (por software).
- TB8** Almacena el noveno bit que será transmitido en los modos 2 y 3.
- RB8** Es el noveno bit que fue recibido en los modos 2 y 3. En el modo 1, si SM2 = 0, RB8 es el bit de stop recibido. En modo 0 RB8 no es usado.
- TI** Bandera de interrupción de la transmisión. Activada por hardware al final del octavo bit en el modo 0, o al principio del bit de stop en otros modos. Debe ser limpiado por software

SM0	SM1	MOD0	ESPECIFICACION	BAUD RATE
0	0	0	Reg. de corrimient	f osc./12
0	1	1	UART 8 bits	variable
1	0	2	UART 9 bits	f.osc./32 o /64
1	1	3	UART 9 bits	variable.

Tabla 3.3 Modos de operación del puerto serie

RI Bandera de interrupción de la recepción. Activada por medio del tiempo de transmisión del bit de stop en los otros modos. Debe limpiarse por software.

MOD0 0: Los datos de recepción o transmisión son enviados mediante 8 corrimientos con una frecuencia de 1/12 de la frecuencia de oscilación.

MOD0 1: 10 bits son los que se transmiten por la línea (TXD) o se reciben por la línea (RXD). Un bit de inicio (start bit de nivel 0 lógico), 8 bits de datos y un bit de fin (Stop Bit de nivel 1 lógico). En recepción el bit de Fin (Stop), se almacena en RB8 de SCON, si SM2 = 0. El Baud Rate (frecuencia de transmisión o recepción) es variable.

CAPITULO 4

MODO 2: 11 bits son transmitidos (TXD) o recibidos (RXD), un bit de inicio (Sart Bit de nivel 0 lógico), 8 bits de datos, un noveno bit de datos programable, en la transmisión es TB8, en la recepción es RB8, y un bit de Fin (Stop bit). Su Baud Rate es de 1/32 o 1/64 de la frecuencia de oscilación.

MODO 3: 11 bits son transmitidos (TXD) o recibidos (RXD), en la misma forma que en el modo 2, sólo que aquí la frecuencia de transmisión/recepción (Baud Rate) es variable. Se utiliza el timer 1 para generar el Baud Rate.

En los 4 modos, la transmisión es inicializada cuando SBUF es utilizado como registro de destino. La recepción en el modo 0 comienza cuando RI = 0 y REN = 1, en los otros modos cuando se detecta el bit de inicio (START) si REN = 1.

4.7 TRANSMISIÓN SERIE UTILIZANDO EL MODO DE CONTROL 1.

Veamos con mayor detalle el modo 1 de control para transmisión/recepción de datos por el puerto Serie.

En este modo 10 bits son transmitidos (por TXD) o recibidos (por RXD). Un bit de inicio "START" en nivel 0, 8 bits de datos (primero el LSB), y un bit de paro "stop" en nivel 1. En la recepción, el bit de STOP viene en RB8 de SCON. El Baud Rate es determinado por la frecuencia de los "overflows" del Timer 1. La figura 4.8 muestra un diagrama del Puerto Serie en MODO 1.

La Transmisión es iniciada por cualquier instrucción que escriba en el registro SBUF. La señal de escritura en SBUF también carga un 1 dentro de la novena posición de registro de corrimiento de transmisión y establece en la unidad de control TX una demanda de transmisión. La transmisión comienza en S1P1 del ciclo de máquina siguiente, en el primer pulso proporcionado por el circuito divisor ($\div 16$ de la frecuencia dada por el Timer 1). Es decir los datos de entrada están sincronizados por el circuito divisor $\div 16$, y no por la escritura en SBUF.

La transmisión comienza con la activación de la señal SEND, la cual introduce el bit de START en TxD. Un período (1/16 de la frecuencia dada por el Timer 1) más tarde la señal DATA es activada, la cual permite la salida de los bits que se transmitirán por TxD. El primer pulso de corrimiento ocurre un período después de esto. La recepción es iniciada por una transición de 1 a 0 en RxD. La línea RxD es muestreada a una frecuencia de 16 veces la frecuencia de "Baud Rate" que ha sido establecida. Cuando un transiente es detectado el divisor entre 16 es inmediatamente restablecido y 1FFH es escrito en la entrada del registro de corrimiento.

Cada tiempo que dura el bit recibido, es dividido en 16 períodos. Durante los períodos séptimo, octavo y noveno, el valor del bit es muestreado, el valor aceptado, es el que se obtuvo en las últimas dos muestras. Esto se hace para eliminar ruido. Si el valor aceptado del primer bit es 1, el circuito receptor es restablecido y la unidad receptora espera otro transiente de inicio. Si el bit de comienzo es correcto, es decir 0, la recepción continúa.

CAPITULO 4

Cuando RI=0 SM2=0 ó el bit de stop es =1, el bit de stop se introducirá en RB8, los 8 bits de datos en SBUF y RI se activa. A partir de éste momento la unidad receptora espera otra transición negativa en la línea RxD.

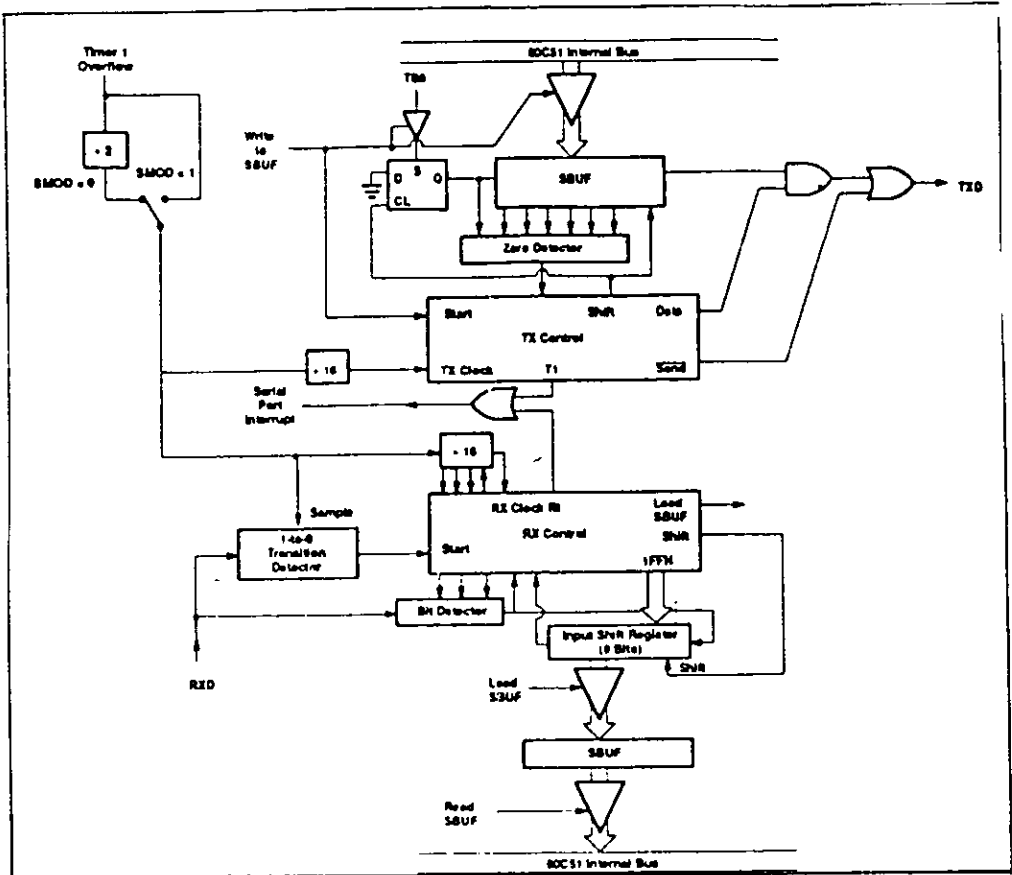


Figura 4.8 Transmisión serie MODO 1

Justo como en MC68HC11 de motorola, el MCS-51 implementa ambos un timer vigia y un detector de falla del oscilador que resetearan la maquina en el momento en el que se detecte un malfuncionamiento. La memoria de programa y la RAM puede extenderse al costo de uno o más puertos de propósito general. Hay dos tipos de interrupción internas y externas implementadas en el tesis profesional "Tarjeta de adquisición de datos utilizando el microcontrolador 8051"

CAPITULO 4

MCS-51. Dos son externas y el resto internas. Todas las interrupciones son mascarables, cada una individualmente y todas de forma simultanea bajo el control de un simple bit. Las interrupciones internas se generan por el timer, por la interfase serial y por el convertidor A/D. A cada interrupción puede asignarse una o dos prioridades: alta o baja. Las interrupciones de alta prioridad interrumpiran a las de baja prioridad, pero entre el nivel de prioridad simple, el orden del servicio se determina por una secuencia fija de escrutinio.

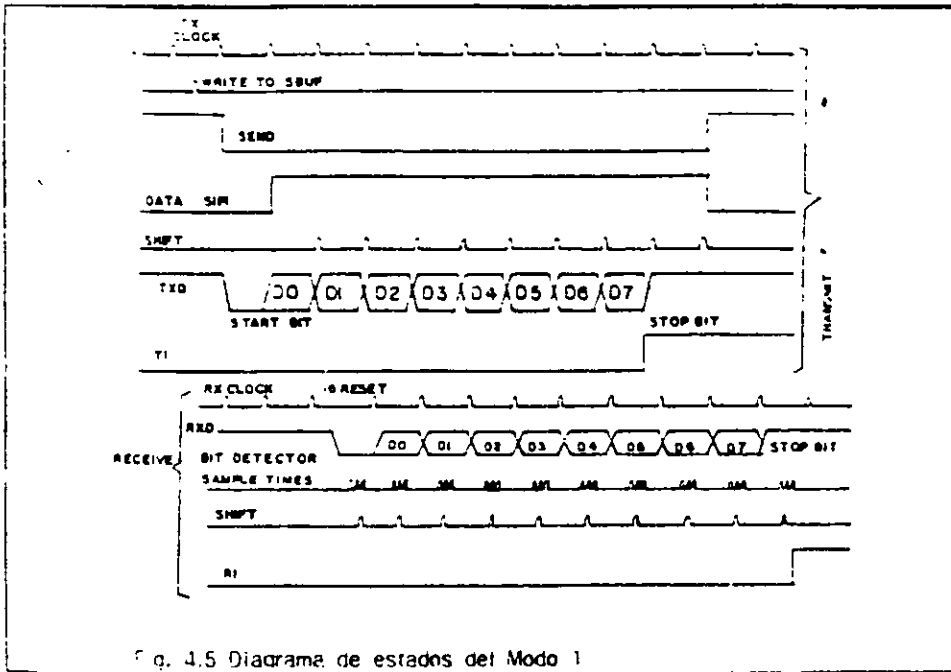


Fig. 4.5 Diagrama de estados del Modo 1

Fig. 4.9 Diagramas de estados del Modo 1

5.1 MODOS DE DIRECCIONAMIENTO

Para entender un conjunto de instrucciones es fundamental conocer la variedad de modos de direccionamiento disponibles. Mientras que los siguientes modos de direccionamiento nos son familiares, estan seguidos de un comentario para su interpretaci3n en el ambiente del MCS-51. Los modos de direccionamiento que la familia del MCS-51 trabaja son los siguientes:

1. Directo: Nos permite el acceso solo a la RAM interna y a los SFRs.
2. Indirecto: Accesa a toda la RAM desde R₀ hasta el registro R₁ de el banco de registros seleccionado o del stack pointer, el direccionamiento de 16 bits debe usar el registro DPTR.
3. Inmediato: Los datos inmediatamente siguen a la instrucci3n.
4. Indexado: Solo la memoria del programa puede accesarse de este forma y es de solo lectura.
5. Por registro: Un c3digo de 3 bits selecciona uno de los ocho registros del banco de registros especificado en el PSW.

Se proporciona una lista completa de instrucciones en la referencia 10 de la bibliografia. Es interesante notar que todo lo que esta en la RAM y los SFRs pueden moverse hacia o del stack sin tener que pasar por el Acumulador. Todos los modos de direccionamiento tambi3n estan disponibles para mover datos directamente de una localidad de memoria a otra sin tener que pasar a trav3s del acumulador, a continuaci3n se detallan estos modos de direccionamiento.

5.2 DIRECCIONAMIENTO DIRECTO

En este forma de direccionamiento el operando se especifica por una direcci3n de 8 bits en la instrucci3n. Solamente los datos de la RAM interna y los del campo del SFR pueden ser direccionados directamente.

EJEMPLO:

ADD A, 7FH ; Se suma al acumulador el dato que se encuentra en la direcci3n 7FH de la RAM interna y el resultado ser3 almacenado en el Acumulador.

MOV A, 2EH ; Se carga al acumulador con el dato que se encuentra en la direcci3n 2EH de la memoria RAM interna.

MOV 3DH, 4EH; El contenido de la direcci3n 3DH se carga con el dato que se encuentra en la direcci3n 4EH.

5.3 DIRECCIONAMIENTO INDIRECTO.

En este direccionamiento se utiliza un registro en el cual se encuentra la direcci3n del operando. Toda la memoria RAM interna y externa puede ser direccionada indirectamente.

Los registros de direcciones de 8 bits, pueden ser los registros R0 y R1 del banco de registros, o el SP.

El registro de direcci3n de 16 bits puede ser solamente el registro DPTR.

EJEMPLO :

ADD A, @R0 ; Se suma al acumulador el contenido de la dirección que esta apuntando R0.

MOV A, @R0 : El Acumulador se carga con el dato que se encuentra en la dirección apuntada por R0.

MOV @DPTR, A; El contenido del acumulador se guarda en la dirección apuntada el DPTR.

5.4 DIRECCIONAMIENTO INMEDIATO.

El valor de una constante sigue al código de operación en el programa.

EJEMPLO :

MOV A, #64H ; El acumulador se carga con el dato 64H inmediatamente.

ADD A, #120 ; El acumulador se suma al número decimal 120 y el resultado se almacena en el acumulador.

MOV DPTR, #1245H ; El DPTR se carga con el dato 1245H en forma inmediata.

5.5 DIRECCIONAMIENTO INDEXADO.

Solamente la memoria del programa puede ser accesada mediante este modo de direccionamiento y solo en lecturas.

Este modo de direccionamiento es utilizado en las lecturas de tablas de la memoria del programa o datos que se encuentran como constantes.

Un registro de 16 bits (el DPTR o el PC), apunta la base de la tabla y mediante el Acumulador se establece el número de la entrada de la tabla. La dirección de la entrada de la tabla en la memoria del programa está formada por la suma del Acumulador y el Apuntador de Base (DPTR o PC).

Otro tipo de Direccionamiento indexado, es usando la instrucción "Salto de casillero". En este caso la dirección del destino del salto es calculada como la suma del apuntador de base más el Acumulador.

EJEMPLO :

MOVC A, @A+DPTR ; Mueve una constante que se encuentra en la memoria del programa. El acumulador se carga con el dato apuntado por la dirección formada por la suma del Acumulador A y el Contador del programa (PC).

MOVC A, @A+PC ; El Acumulador se carga con el dato que se encuentra en la dirección formada por la suma del mismo Acumulador A y el Contador del programa (PC).

5.6 DIRECCIONAMIENTO POR REGISTRO

Los 8 registros pueden ser accedidos mediante ciertas instrucciones que simplifican sus códigos de operación (opcode) y en la mayoría de los casos son más rápidas.

Existen 4 bancos de registros, cada banco contiene a su vez 8 registros (R0 a R7), éstos bancos pueden ser accedidos mediante los bits 3 y 4 del PSW.

EJEMPLO :

ADD A, R7 ;Se carga al acumulador con el resultado de suma del Acumulador y el contenido del Registro R7.
DEC R0 ; Decrementa el registro R0.

5.7 TRANSFERENCIA DE DATOS.

RAM interna Para poder mover datos de la memoria interna RAM o SFR, existen 8 instrucciones principales:

La instrucción **MOV<dest>,<fuente>**; permite transferir datos de la memoria interna RAM y del SFR sin pasar a través del Acumulador, mientras que las instrucciones **MOV A,<fuente>** y **MOV <destino>,A** utilizan al Acumulador para el movimiento de datos dentro de la memoria interna RAM.

MOV A,<fuente> ; A= <fuente>
MOV <destino>,A ; <destino>=A
MOV<destino>,<fuente> ; <destino>=<fuente>
MOV DPTR,#dato 16 bits ; DPTR = constante de 16 bits inmed.
PUSH<fuente> ;INC SP →<@SP> ← (fuente)
POP <destino> ;(destino) ← <@SP> ; DEC SP
XCH A,<byte> ; ACC y <byte> intercambian sus datos.
XCHD A,@Ri ; ACC y @Ri intercambian el nibble bajo.

La instrucción **PUSH**, primeramente incrementa el Stack Pointer (SP), y después guarda el dato dentro de la localidad de memoria apuntada por el Stack.

La instrucción **POP**, primero toma el dato de la memoria y después decreuenta el SP. Cabe señalar que si el SP apunta en la localidad 7FH en los dispositivos que no tienen implementados los 128 bytes altos de memoria interna RAM, NO SE PODRAN SEGUIR ALMACENANDO DATOS, DEBIDO A QUE ESTOS SE PERDERIAN. Arriba de la dirección 80H los datos almacenados con la instrucción **PUSH** se pierden y los datos tomados con **POP**, son indeterminados.

CAPITULO 5

La instrucción **XCH A, <dirección>** permite al acumulador y al dato apuntado por la dirección de intercambiarse entre sí. La instrucción **XCHD A,@R0** intercambia los 4 bits menos significativos (*low nibble*), de Acumulador con los 4 bits menos significativos del dato apuntado por el registro R0.

RAM EXTERNA Las instrucciones para la transferencia de datos en memoria RAM externa, son 4 básicamente:

```
MOVX A,@Ri      ; A ← <@Ri>
MOVX @Ri, A     ; <@Ri> ← A
MOVX A, @DPTR   ; A ← <@DPTR>
MOVX @DPTR,A    ; <@DPTR> ← A
```

Las habilitaciones de lectura y escritura en RAM externa son activadas solamente durante la activación de una instrucción MOVX.

Movimiento de tablas localizadas en la memoria del programa La mayoría de las veces dentro de nuestro programa principal tenemos algunas constantes, como son palabras de control, contadores, banderas iniciales, tablas de datos, etc. Por tal motivo, el 8051 utiliza las siguientes instrucciones solamente en lectura (ya que en escritura no tendría caso por ser una memoria EPROM o ROM).

MOVC A, @A+DPTR ; A ← <@A+DPTR> direc. en memoria de programa

MOVC A, @A+PC ; A ← <@A+PC> direcc. en memoria de programa

5.8 INSTRUCCIONES BOOLEANAS

EL 8051 contiene un completo procesador Booleano (por bits), el cual permite ejecutar instrucciones de movimiento, limpieza, establecimiento, complementación de un sólo bit, y operaciones lógicas como AND y OR entre bits. A continuación se muestran las instrucciones Booleanas soportadas por el MC-8051:

```
ANL C, bit      ; C ← C AND bit
ANL C,/bit      ; C ← C AND NOT bit
ORL C, bit       ; C ← C OR bit
ORL C,/bit      ; C ← C OR NOT bit
MOV C, bit       ; C ← bit
MOV bit, C       ; bit ← C
CLR C            ; C ← 0
CLR bit         ; bit ← 0
SETB C          ; C ← 1
SETB bit        ; bit ← 1
CPL C           ; C ← NOT C
CPL bit         ; bit ← NOT bit
JC rel          ; Brinca si C=1
JNC rel         ; Brinca si C=0
```

JB bit, rel ;Brinca si bit = 1
 JNB bit, rel ;Brinca si bit= 0
 JNB bit, rel ;Brinca si bit =1; limpia bit

Ejemplo :

```
CONTROL DATA 20H
BANDERA BIT CONTROL,7
ORG 100H
MOV C,BANDERA
MOV P1.0,C
```

La etiqueta BANDERA representa, un bit direccionable de los 128 bits que se encuentran en la memoria baja RAM interna (desde la dirección 20H a la 2FH), o de los 128 bits de algunos registros del espacio SFR. En este caso particular, pertenece al bit 7 de la palabra CONTROL que se encuentra en la dirección 20H de la RAM interna, la línea 0 del Puerto 1 es establecida o limpiada dependiendo del valor de BANDERA.

Otro ejemplo, sería la implementación de la función Booleana OR-Exclusivo por programación. Como sabemos si el bit 2 es igual a 0 el bit 1 tiene el mismo valor de C, y si el bit 2 es igual a 1, el bit 1 tiene el valor negado de C, de tal forma que su implementación sería:

```
MOV C, bit 1 ; El carry C= bit 1
JNB bit 2, cont ; Es bit 2 = 0?, sino salta
CPL C ; Se complementa C=bit 1
cont : RET ; Regresa al programa
```

5.9 INSTRUCCIONES DE SALTO.

Salto Condicionados El juego de instrucciones con saltos que estan condicionados a la activación o desactivación de algunas de las banderas del PSW son las siguientes:

```
JZ rel ; salta si A=0
JNZ rel ; salta si A≠ 0
DJNZ <byte>, rel ; decrementa y salta si no es igual a 0
CJNE A, <byte>, rel ; Salta si A ≠ <byte>
CJNE <byte>, #dato, rel ; Salta si <byte> ≠ #dato
```

Como se vió anteriormente, en el PSW no se tiene la bandera Z, que prueba el CERO, sin embargo, aqui se muestran las instrucciones JZ y JNZ, las cuales prueban todo el dato del acumulador para esta condición.

La instrucción DJNZ (decrementa y salta sino es igual a cero), ejecuta un lazo N veces, hasta que un "contador" es igual a cero.

EJEMPLO:

```

MOV     CONTADOR,#10 ; N = 10
LAZO : (comienzo del conjunto de instrucciones)
DJNZ   CONTADOR,LAZO
        (continua)

```

De la misma forma podemos apreciar la instrucción **CJNE**, la cual compara y salta si el Byte (registro, Acumulador o dirección) es igual al dato comparado. El salto es relativo.

En los saltos relativos el rango del desplazamiento es de -128 a +127 bytes con respecto al primer byte siguiente a la instrucción.

Salto incondicionado Las instrucciones que permiten hacer los saltos incondicionados son las siguientes:

```

JMP dirección ; Salta a la dirección
JMP @A+DPTR; Salta a la dirección A+DPTR
CALL dirección ; Llama a la subrutina "dirección"
RET           ; Regreso de la subrutina
RETI          ; Regreso de la interrupción
NOP           ; Sin operación

```

Arriba se muestra una sola instrucción de salto **JMPdirec** pero en realidad existen 3 tipos de saltos; **SJMP**, **LJMP** y **AJMP** los cuales difieren en el formato de la dirección de destino,

JMP es un nemónico genérico el cual puede ser usado (en algunos compiladores) durante la programación sin tener en cuenta de que forma el salto sera codificado.

La instrucción **SMJP**, codifica la dirección como un desplazamiento relativo. La instrucción es de 2 bytes, es decir el **OPCODE** y el salto relativo. Su límite es de -128 a +127 bytes a partir de instrucción siguiente al **SJMP**.

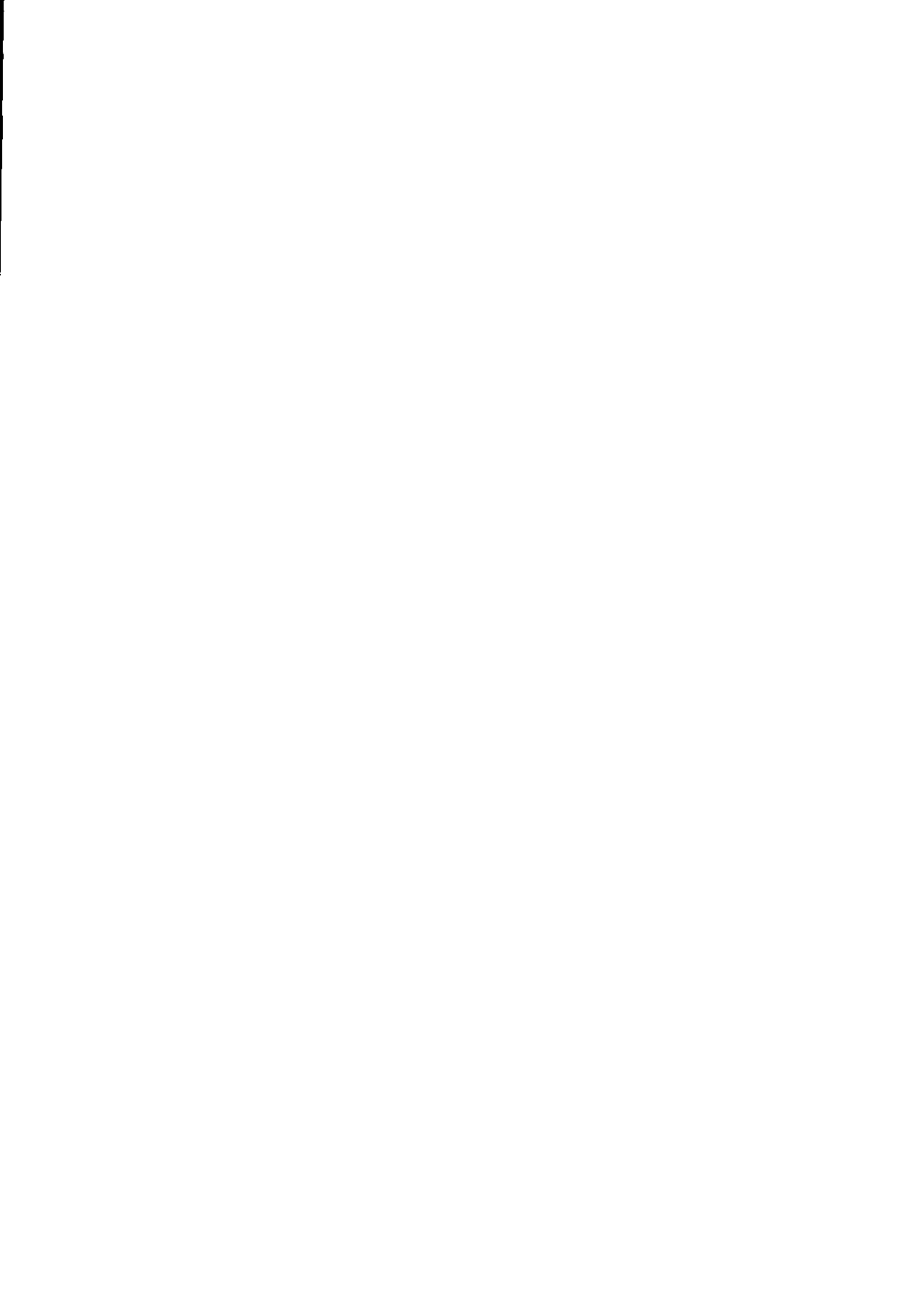
La instrucción **LJMP**, codifica la dirección como una constante de 16 bits. La instrucción es de 3 bytes de largo, el **OPCODE** y la dirección en 2 bytes. Este salto puede desplazarse en todos los 64K de memoria.

La instrucción **AJPM** codifica la dirección como una constante de 11 bits. La instrucción es de 2 bytes de largo, el **OPCODE** el cual contiene 3 bits de los 11 bits de direcciones y el otro byte de 8 bits de direcciones. Cuando la instrucción es ejecutada esos 11 bits son simplemente sustituidos por los 11 bits más bajos en el PC. Los 5 bits más altos permanecen sin alterarse. Estos saltos pueden ser hasta de 2K.

CAPITULO 5

En cualquiera de los casos el programador especifica la dirección al ensamblador de la misma manera; como una etiqueta o constante de 16 bits. El ensamblador deberá poner la dirección dentro del formato correcto de forma automática.

La instrucción **RETI** es usada para el regreso de una rutina de servicio de interrupción. La diferencia con la instrucción **RET** es que **RETI** llama al sistema de control de interrupción mientras que la interrupción está en proceso. Si no existe ninguna interrupción en proceso, entonces la instrucción **RETI** es igual a **RET**.



DISEÑO EXPERIMENTAL La figura 6.1 muestra como se conecta una microcomputadora a un convertidor ADC de rampa-digital para la adquisición de datos como propósito. La computadora genera los pulsos de INICIO para comenzar cada nueva conversión A/D. La señal de EOC (end-of-conversion) proveniente del ADC es alimentada a la computadora. La computadora examina la EOC para hallar cuando la conversión A/D actual se ha completado; entonces se transfiere la información digital de la salida del ADC hacia su memoria. El proceso comienza cuando la computadora genera el pulso de inicio (START) para comenzar el ciclo de conversión A/D. La conversión se completa en un tiempo t_1 , entonces la línea EOC del convertidor genera un pulso de bajada. Esta transición negativa NGT (Negative Going Transition) de la EOC indica a la computadora que el CAD tiene una salida digital que ahora representa el valor de un voltaje al cual le llamaremos V_A , y la computadora cargará estos datos en la memoria.

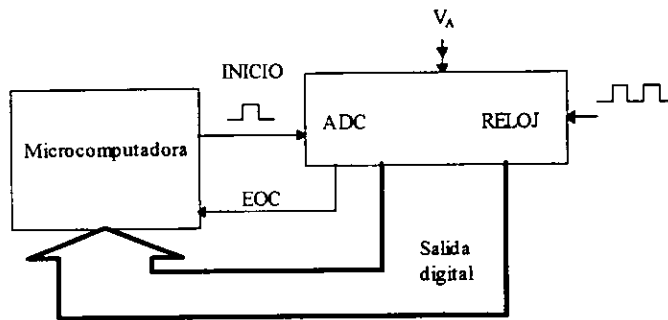


Figura 6.1 interconexión de una microcomputadora con un convertidor analógico digital

La computadora genera un nuevo pulso de inicio START después del tiempo t_1 para iniciar un segundo ciclo de conversión. Note que la señal de EOC se manda a estado alto porque el pulso de inicio START reinicia el contador del ADC. La segunda conversión termina en un tiempo t_2 cuando se detecta en el ADC el nuevo el valor de V_A la computadora carga entonces la información digital en el lugar correspondiente en la memoria. Estos pasos se repiten sucesivamente.

El proceso mediante el cual la computadora genera un pulso de inicio, examina el EOC, y carga información del CAD hacia la memoria se hace bajo el control de un programa que la computadora esta ejecutando. Este programa de adquisición de datos determinará cuantos datos de la señal analógica serán almacenados en la memoria de la computadora. En nuestro caso hemos sustituido a la microcomputadora de la figura 6.1 con un microcontrolador de la familia 8051 de Intel.

Un parámetro importantísimo, que debe ser tomado en cuenta, al inicio de todos los diseños, es la capacidad de almacenamiento de información de los dispositivos principalmente en los sistemas de adquisición de datos, donde, el tratamiento, análisis y cálculo de resultados es indispensable,

El almacenar datos en memoria RAM, es relativamente sencillo y fácil de realizar, la rapidez de acceso de éstos circuitos, permite procesar datos en tiempo real, más sin embargo, su capacidad de almacenado, es relativamente limitada. Por otro lado, aunque de menor velocidad de acceso, los discos flexibles así como los discos duros, ofrecen una capacidad mucho mayor de almacenado.

Por tal motivo, en este trabajo se plantea la transmisión de datos, de una memoria RAM hacia una computadora del tipo IBM-PC compatible, haciendo uso del puerto paralelo de dicha computadora, el programa que gobierna al microcontrolador esta hecho de tal manera que primero se almacenan los datos en un memoria SRAM 6264 y después se envían por el puerto.

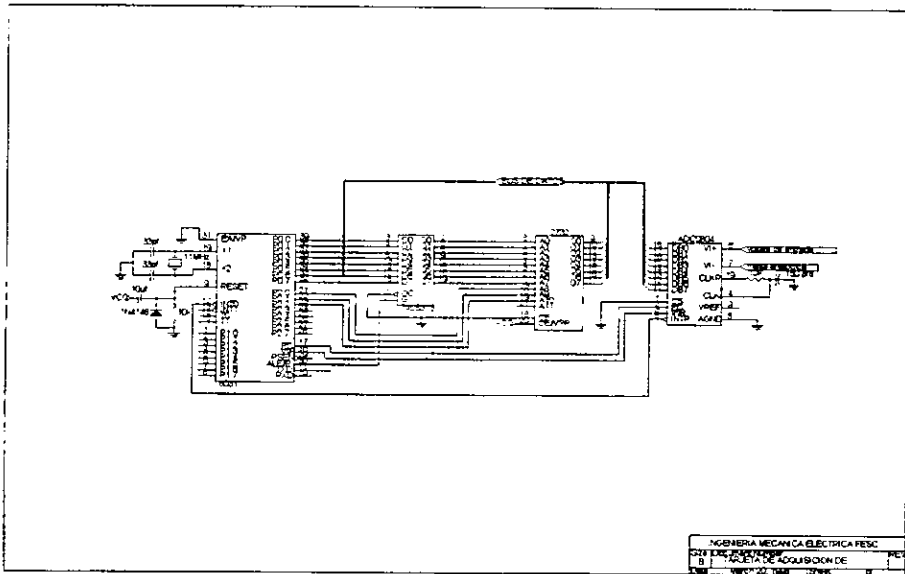


Fig. 6.2 conexión del ADC0804 hacia el microcontrolador

La figura 6.2 muestra una conexión del ADC0804 hacia el microcontrolador en la aplicación de adquisición de datos. El microcontrolador controla cuando se efectuará una conversión generando las señales de CS y WR. Entonces el ADC adquiere los datos que están en la línea de entrada y con las señales de CS y RD, los pone en su BUS de salida después de detectar la señal de NGT generará la señal INTR que indica el fin de la conversión y esta se envía al microcontrolador el cual almacenará lo que tiene en el puerto 0 en el acumulador. Cuando el INTR se va a estado alto el CS y el WR se van a estado bajo, pero el proceso de conversión no empieza hasta que el CS y el WR regresan a estado alto. Las líneas de salida de datos del ADC están en estado de alta impedancia hasta que el microcontrolador activa las señales de CS y RD; en este punto los buffers de datos del ADC son habilitados debido a esto los datos del ADC son enviados al microcontrolador a través del bus de datos. Las líneas de datos regresan al estado de alta impedancia cuando las líneas de CS y RD regresan a estado alto.

Edición del Programa

Para verificar y depurar un programa en ensamblador primeramente es necesario editar el programa en lenguaje ensamblador utilizando mnemónicos de la familia 8051. Esto puede hacerse con cualquier editor de textos estándar como por ejemplo el "EDIT" del MS-DOS ver 5.0 o posterior. El archivo generado se debe guardar con extensión .ASM.

Ejemplo:

A:\EDIT A:\EJEMPLO\PROG1.ASM

Al ejecutar esta instrucción desde la línea de comandos del MS-DOS se abre un archivo de nombre "PROG1.ASM" en el subdirectorio ejemplos de la unidad del disco A. Ya dentro del editor de textos hacer el programa con formato similar al de los ejemplos proporcionados en el capítulo anterior.

A continuación se muestra la estructura que debe tener un programa para poder correrlo en el simulador.

```

2000          ORG 100H; Inicio del ensamblado
              LJMP INICIO; Salta al inicio del programa principal
              ORG 2003H; Origen de la interrupción cero.
2003 INTE0:   CLR A;      Vector de interrupción externa cero.
              .          ; Cuerpo de interrupción
              .          ; Otras interrupciones
              ORG 208BH; Inicio del programa principal
208B INICIO:  MOV TVON, #01H
              .          ; Cuerpo del programa principal
          SIGUE: SJMP SIGUE; Bucle infinito

```

Ensamblado del Programa

El paso siguiente es ensamblar el programa utilizando el XASM51; para lo que es necesario tener en el mismo subdirectorio el programa XASM51 y el archivo con extensión .ASM. Utilizando el disco SIEV31, en el subdirectorio ejemplos ensamblar el programa como se indica en el siguiente ejemplo:

A:\EJEMPLOS\XASM51 PROG1.ASM

Donde "PROG1.ASM" es el nombre del archivo de nuestro programa a ensamblar. Si el archivo contiene errores, el programa ensamblador lo indicará mostrando un mensaje. Después de ejecutar

el comando anterior, se generarán los archivos con extensión .PRN y .HEX. El archivo .HEX es un archivo con formato Intel y es utilizado para grabarse en EPROM, conectarse al microcontrolador y así ejecutarlo

Es importante recordar que el programa debe ser ensamblado con origen 0000H cuando se grabe en EPROM.

Ejecución del Programa

Después de haber ensamblado nuestro programa y si el ensamblador no nos indicó ningún error, podremos grabar nuestro programa en la memoria 2732 interconectada al microcontrolador. Para grabar el programa en la memoria se siguieron los siguientes pasos:

1. Verificar que la memoria a emplearse este no contenga datos.
2. Se traslada el archivo con extensión .HEX a el área de datos del PROMAX
3. Se manda grabar este archivo de datos en la memoria.

Al dársele alimentación al circuito y con la memoria colocada en el lugar correspondiente automáticamente se empieza a ejecutar e programa.

El circuito utiliza el puerto serie COM1 de la PC para realizar la comunicación.

Tenemos la desventaja de que si queremos cargar otro programa, se tiene que modificar el contenido de las localidades de memoria, ver o cambiar el valor de los registros, ver el mapa de memoria; todo esto implica repetir el procedimiento citado antes.

Para la transmisión serie es necesario que el sistema de evaluación se encuentre enlazado con la PC utilizando el cable correspondiente y que la tarjeta de adquisición se encuentre alimentada, así como la circuitería asociada a la misma. Cumplido lo anterior se tiene que abrir turbo C para estar preparado para la captura de datos en un archivo designado por el programa de captura. EL cual se muestra a continuación

DESCRIPCIÓN BREVE DEL PROGRAMA XASM51

El programa XASM51.COM, es un ensamblador ejecutable que nos permite convertir instrucciones en mnemónicos de la familia de microcontroladores 8051 de Intel, en código objeto.

El programa genera 2 archivos:

- a) El primero con extensión HEX, es el archivo donde reside el código objeto del programa fuente, es decir, el programa ya ensamblado.
- b) El segundo archivo que genera con extensión PRN, es un archivo auxiliar que contiene la dirección y el código del programa ensamblado, posibles errores detectados, y los mnemónicos del programa fuente.


```

0003          SETB BANDER
0005          MOVX  A,@R0    ; SE ADQUIERE EL DATO
0006          MOVX  @DPTR,A ; SE ALMACENA EL DATO RAM
0007          RETI

```

```

;*****
;***** RUTINA DE SERVICIO DE LA INTERRUPCION DEL TIMER 0 *****
;*****

```

```

000B          ORG 0BH
000B          MOV  TH0,#01FH ; SE CARGA LA BASE DE TIEMPO
000E          MOV  TL0,#00H  ; TIEMPO 62,500 uSEG
0011          DJNZ R4, SALE
0013          SETB RETARD
0015          MOV  R4,#08H   ; ESPERA DE 0.5 mSEG
0017  SALE:   RETI

0100          ORG 100H
0100  TRANSMI: MOV P2,#CAD
0103          MOV  R0,#80H
0105          MOV  R4,#30H
0107          MOV  DPTR,#MEMRAM
010A          MOV  TH0,#1FH  ; SE CARGA LA BASE DE TIEMPO DE
010D          MOV  TL0,#00H  ; 62,500 uSeg
0110          MOV  TMOD,#31H ; CON RELOJ DE 11.059
0113          MOV  TCON,#11H ;INT0 POR TRANSCIENTE
0116          MOV  IE,#83H   ;SE PERMITE LA INT T0
0119          MOV  R7,#3CH   ;R7R6 CONTADORES DE
011C  LAZO:   MOV  R6,#14H   ;1200 DATOS

```

```

011F   TIAD:   JNB RETARD, TIAD
0121           CLR RETARD           ; TIEMPO DE ADQUIS DE 0.5 Seg.
0123           MOVX A,@R0           ; COMIENZA LA CONVERSION
0124   TICONV: JNB BANDER,TICONV; TIEMPO DE CONVERSION
0127           CLR BANDER
0129           INC DPTR
012A           DJNZ R6, TIAD
012C           DJNZ R7, LAZ0
012E           CLR IE.0 ; DESACTIVA LAS INTERRUP.
0130           CLR IE.1 ; DEL T0 Y INT0

```

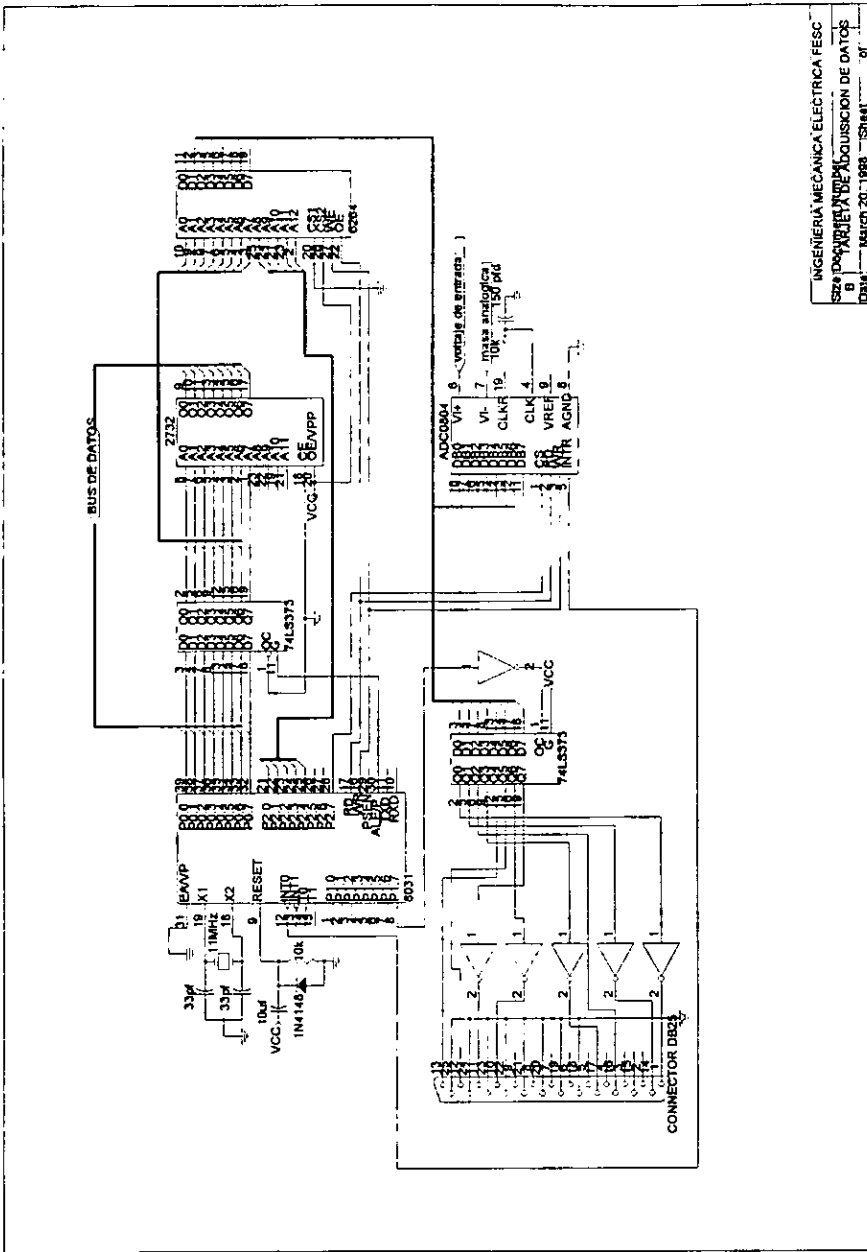
; SE TERMINA LA ADQUISICIÓN DE DATOS Y COMIENZA LA TRANSMISIÓN.

```

0132           SETB P1.7           ; SE ACTIVA EL LATCH DEL PUERTO PARALELO
0134           MOV DPTR,#MEMRAM;
0137           MOV R7,#3CH ; SE RECARGAN LOS CONTADORES
014A   LAZ2:   MOV R6,#14H ; CON 1200 DATOS
014D   LAZ3:   MOVX A,@DPTR ;SE TRANSFIEREN LOS DATOS
0150           INC DPTR
0151           DJNZ R6, LAZ3
0153           DJNZ R7, LAZ2
0000           END
0000           END

```

En la siguiente pagina se presenta el circuito final de la tarjeta de adquisición de datos y despues se presenta el programa que empleado para la captura de datos que complementa al anterior escrito en ensamblador y esta escrito en lenguaje "C".



INGENIERÍA MECÁNICA ELÉCTRICA FESC
 SIZE 100 PARTES DE ADQUISICION DE DATOS
 B
 DATE: MARCH 20, 1999 SHEET 91


```

#include<stdio.h>
#include<stdlib.h>
#include<conio.h>
#include<graphics.h>
#include<dos.h>
#include<string.h>
#include<time.h>
#include <math.h>

int Dor,Dil,Dih, N, x,y;
float V=0,datos[1200], data[1200];
int tarjeta=DETECT,modo;
int z=0;

/*****/
inicio()
{
  initgraph(&tarjeta,&modo,"c:\\tc\\bgi");
  clearviewport();
  gotoxy(10,10);
  printf("Qu, puerto paralelo (LPT_) 1,2 desea utilizar ?");
  gotoxy(65,10);
  scanf("%d",&N);
  gotoxy(10,15);
  printf("Seleccioné el LPT(%d)",N);
  delay(200);
  N=(N-1)*2+8;
  /*Busca dirección del puerto*/
  Dor = peekb(64,N)+256*peekb(64,N+1); /*Registro de salida de datos (OUT)*/
  Dih = Dor+1; /*Registro de entrada, nibble alto (INP)*/
  Dil = Dor+2; /*Resgistro de entrada, nibble bajo (INP)*/
  Cr = Dil; /*Registro de control*/
return 0;
}
/*****/
adquiere()
{
  N=0;

  clearviewport();

  setcolor(CYAN);
  rectangle(1,130,637,330);

  setcolor(GREEN); /* Se coloca el Tamiz */
  for(y=150;y<331;y=y+20){ line(2,y,636,y); }
  for(x=2;x<637;x=x+5) { line(x,131,x,329); }
}

```

```

setcolor(BLUE);
line(2,131,2,329);           /*Eje y*/
line(2,231,636,231);       /*Eje x*/

setcolor(YELLOW);
outtextxy( 2, 120, "señal adquirida          1V/div");

for(x=0;x<636;x=5+x)
{

    /*Envia pulso de inicio de convertidor*/

V=((inportb(Dil) & 15)+(inportb(Dih) & 240))*0.01985;
  datos[x]=V;
  putpixel(x,320-datos[x]*20,BLUE);

  delay(20);
}

getch();

sound(800);delay(80);nosound();   sound(1550);delay(50);nosound();
sound(1900);delay(70);nosound();  sound(7500);delay(70);nosound();

getch();
}
main()
{
  inicio();
  adquiere();
  closegraph();
}

```

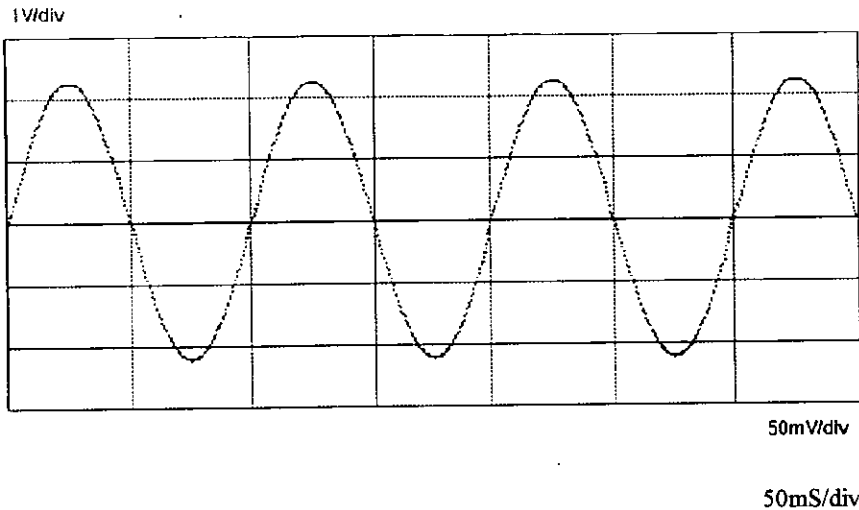
RESULTADOS

Se capturó un arreglo de 1200 datos los cuales se desplegaron de forma gráfica en la pantalla de una computadora como se muestra en la figura 6.4. Estos datos representan el monitoreo de una señal senoidal proveniente de un generador de funciones de 5 Vpp, con una frecuencia de 10 Hz, la intención de esta figura es mostrar que nuestro sistema funciona, no intenta hacer el análisis ni el procesamiento posterior a la captura de los datos, para este caso la velocidad de adquisición es de 1 mS. Como se nota, en nuestros programas se diseñaron para capturar un conjunto de datos que implican una velocidad de adquisición más lenta, es por esto que la velocidad de

captura aparece como 0.5 mS, si se requiere como en el caso de la figura 6.4 una velocidad de adquisición menor, hacer el cambio es muy fácil.

CONCLUSIONES

La portabilidad de nuestro sistema (no se requiere de una PC para poder operarlo, pero si para grabar las memorias) es una de sus ventajas, aunque no fue considerado en el diseño como se puede apreciar la transmisión en tiempo real, sin embargo la adquisición si se hace de esta manera, dos alternativas se presentan para solucionar este problema la primera es considerar en nuestro programa en lenguaje ensamblador un retardo del mismo tiempo en el que se esta adquiriendo y la segunda es considerar este mismo retardo en el programa de lenguaje "C".



Si no se tiene gran requerimiento en el tiempo de adquisición este sistema es muy recomendable a nivel laboratorio pues la gran disminución de precios de los circuitos integrados lo hace accesible, ahora que si se cuenta con los recursos para dedicar un sistema PC exclusivamente para la adquisición de datos, es preferible prescindir del microcontrolador y manejar el convertidor por medio del nibble de salida del mismo puerto paralelo.

Fue una buena experiencia el haber construido el sistema propuesto pero el tiempo empleado para aprender la programación del mismo así como el proceso de estar grabando una y otra vez las memorias EPROM fue engorroso, el tiempo de adquisición sin embargo es lo suficientemente rápido como para permitir aplicaciones que requieran de una velocidad de adquisición de 0.5 m Seg., sin ningún problema. Hay un poco de desincronia a la hora de llevar el archivo de datos capturados a la PC pues esto se tiene que hacer tomando en cuenta el número de datos a capturarse que en nuestro caso sería de 10 minutos.



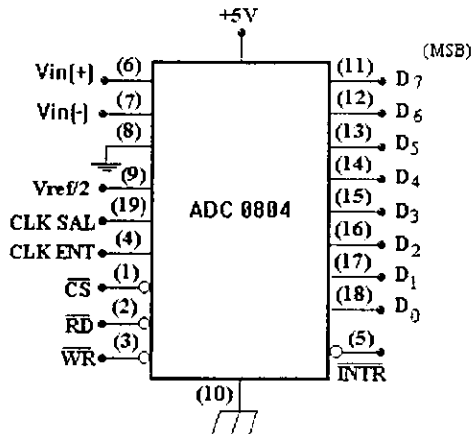
ADC de aproximaciones sucesivas ADC0804

Muchos fabricantes ofrecen ADCs con un amplio margen de cualidades y características de operación. Echemos un vistazo a uno de los dispositivos más populares para tener una idea de lo que actualmente se emplea en sistemas de fácil aplicación. Tal dispositivo es el ADC0804, que es un CI de 20 terminales fabricado con tecnología CMOS y que lleva a cabo una conversión A/D usando el método de aproximaciones sucesivas. Algunas de sus características importantes son las siguientes:

- Tiene dos entradas analógicas: $V_{IN(+)}$ y $V_{IN(-)}$ que permiten entradas **diferenciales** es decir, el voltaje real de entrada, v_{IN} , es la diferencia entre los voltajes aplicados a estos pines [el voltaje analógico de entrada $V_{IN} = V_{IN(+)} - V_{IN(-)}$]. En medidas hechas en un solo punto, la entrada analógica se aplica a $V_{IN(+)}$, mientras que $V_{IN(-)}$ se conecta a tierra analógica. Durante la operación normal, el convertidor usa $V_{CC} = +5$ V como voltaje de referencia, y la entrada analógica puede ir de 0 a 5 V que es el valor de la escala completa.
- Convierte el voltaje analógico de entrada a una salida digital de ocho bits. La salida digital es de tres estados (tristate buffered) debido a lo cual puede ser conectada fácilmente a un arreglo de bus de datos. Con 8 bits, la resolución es de 19.6 mV (5 V/255).
- Tiene internamente un circuito generador de reloj que produce una frecuencia de $f = 1/(1.1 RC)$, donde R y C son valores de componentes de conexión externa. Una frecuencia típica del reloj es 606 kHz usando una resistencia $R = 10$ k Ω y $C = 150$ pF. Se puede usar si se desea una señal externa de reloj conectando en el pin rotulado como CLK IN.
- Si se usa una frecuencia de reloj de 606-kHz, el tiempo de conversión es aproximadamente de 100 μ s.
- Tiene conexiones de tierra separadas para voltajes digitales y analógicos. El pin 8 corresponde a la tierra analógica que se conecta al punto de referencia común del circuito analógico que genera el voltaje analógico. El pin 10 es la tierra digital que es aquella usada por todos los circuitos digitales en el sistema. (note la diferencia de simbología usada para estas tierras). La tierra digital es inherentemente ruidosa debido a los rápidos cambios de corriente que ocurren mientras los dispositivos digitales cambian de estado. Si bien no es necesario usar una tierra analógica por separado, hacerlo asegura que el ruido de la tierra digital no cause switcheo prematuro del comparador analógico dentro del ADC.

Este integrado está diseñado para ser fácilmente conectado a un microprocesador (o microcontrolador en nuestro caso). Por esta razón, los nombres de algunas de las entradas así como de las salidas de este ADC0804 están basadas en funciones que son comunes a los sistemas basados en microprocesadores. Las funciones de estas entradas y salidas se definen como sigue :

CS (Chip Select) Esta entrada tiene que estar en un estado BAJO para que tengan efecto las entradas de RD o WR. Con el CS en estado alto, las salidas digitales estan en un estado de alta impedancia y no se puede efectuar ninguna conversión.



RD (Output Enable) Esta entrada se usa para habilitar los buffers digitales de salida. Con las líneas CS = RD = BAJO, los pines digitales de salida tendran niveles lógicos que representan el resultado de la última conversión A/D.

WR (Start Conversion) Un pulso de bajada se tiene que aplicar a esta entrada para señalar el comienzo de una nueva conversión.

INTR (End of Conversion) Esta señal de salida se ira a estado alto al inicio de una conversión, y regresará a estado bajo para señalar el final de la conversión.

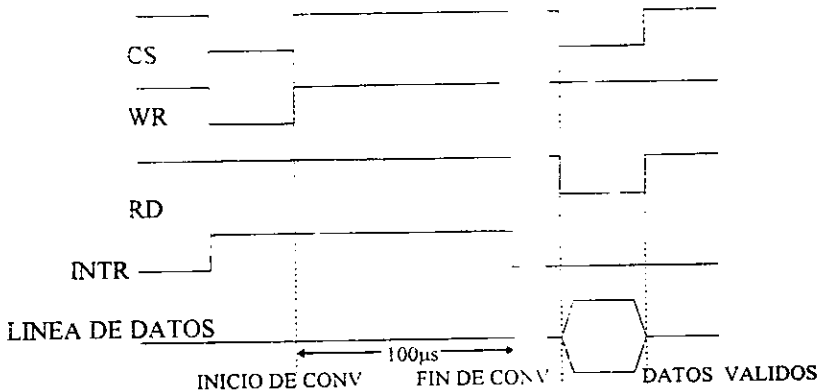
Vref/2 Esta es una entrada opcional que se puede usar para reducir el voltaje de referencia interno y por lo tanto cambiar el rango analógico de entrada que el convertidor puede manipular. Cuando esta entrada no esta conectada, este se fija a 2.5 V (Vcc/2), siendo que Vcc se usa como el voltaje de referencia. Conectando un voltaje externo a este pin, el voltaje de referencia interno se cambia a dos veces este voltaje, y el rango analógico de entrada se cambia de acuerdo a este. La tabla de abajo muestra esto.

Vref/2	Rango Analógico de entrada (V)	Resolución (mV)
Abierto	0- 5	19.6
2.25	0-4.5	17.6
2.0	0-4	15.7
1.5	0-3	11.8

APENDICE

CLK OUT Se conecta una resistencia a este pin para usar el reloj interno. La señal de reloj aparece en este pin.

CLK IN Se usa para una entrada de reloj interno, o para conectar un capacitor cuando se usa la señal de reloj interno.



LSB (least-Significant Bit): En un sistema de código binario, este es el bit que tiene el valor más pequeño o peso. Este valor es el voltaje de escala completa dividido entre 2^n , donde n es la resolución del convertidor.

Error de linealidad: En el caso peor es la desviación entre la línea entre los puntos finales (Cero y de escala completa). Puede expresarse como porcentaje de escala completa o en fracciones de un LSB.

Offset Error (Zero Error). Es la diferencia entre el voltaje de entrada ideal ($1/2$ LSB) y el voltaje actual de entrada que se necesita para hacer la transición (entre) de cero a 1 LSB, el error de Offset se expresa generalmente en LSBs.

Resolución: es el incremento analógico más pequeño correspondiente a un cambio en el código del convertidor de 1 LSB. donde el número de niveles analógicos es igual a 2^n .

Bibliografía

- Boylestad R., Nashelsky L. "Electronica Teoría de Circuitos". Prentice Hall 1989.
- Cruz A. "Detector de Movimientos de la cabeza para ser usado en estudios de de nistagmo congénito". Tesis de M.C Bioeléctronica, CINVESTAV-IPN 1997.
- Eisen A. "Data Acquisition Made Easy". Electronic Design/May 1 1996, pág 111.
- Hintz K., Tabak D. "Microcontrollers: architecture. implementation and programming". Mc Graw Hill 1992.
- Rojas A. "Emsamblador Básico". Marcombo 1993.
- Schildt H. "C the complete reference". McGraw Hill.
- Schildt H. "C MANUAL DE REFERENCIA". Osborne McGraw-Hill 1989.
- Vega S. A. "Manual y aplicaciones del Microcontrolador 8051". Informe Técnico CINVESTAV Departamento de Ingeniería Eléctrica.
- "Macro-Assamblar User`s guide". Intel MCS-51. McGraw-Hill
- "8-Bit Embedded Controllers". Intel 1990.
- "Personal Computer XT Hardware Reference Library". IBM 1983.
- "RCA SK Series Solid State Replacement Guide". RCA 1996.