

REPOSITORIO ACADÉMICO DIGITAL INSTITUCIONAL

Elementos básicos de la arquitectura de computadoras

Autor: Eugenia Yunuén Fernández Verduzco

**Tesina presentada para obtener el título de:
Lic. En Sistemas Computarizados [sic]**

**Nombre del asesor:
Sergio Fco. Barraza Ibarra**

Este documento está disponible para su consulta en el Repositorio Académico Digital Institucional de la Universidad Vasco de Quiroga, cuyo objetivo es integrar organizar, almacenar, preservar y difundir en formato digital la producción intelectual resultante de la actividad académica, científica e investigadora de los diferentes campus de la universidad, para beneficio de la comunidad universitaria.

Esta iniciativa está a cargo del Centro de Información y Documentación “Dr. Silvio Zavala” que lleva adelante las tareas de gestión y coordinación para la concreción de los objetivos planteados.

Esta Tesis se publica bajo licencia Creative Commons de tipo “Reconocimiento-NoComercial-SinObraDerivada”, se permite su consulta siempre y cuando se mantenga el reconocimiento de sus autores, no se haga uso comercial de las obras derivadas.





M.R.

UNIVERSIDAD VASCO DE QUIROGA

RECONOCIMIENTO DE VALIDEZ OFICIAL DE ESTUDIOS
POR ACUERDO 952006 DE LA S.E.P. CLAVE 16PSU0014Q

LICENCIATURA EN SISTEMAS COMPUTARIZADOS

ELEMENTOS BASICOS DE LA ARQUITECTURA
DE COMPUTADORAS

TESINA

QUE PARA OBTENER EL TITULO DE:

LICENCIADO EN SISTEMAS COMPUTARIZADOS

PRESENTA:

EUGENIA YUNUÉN FERNÁNDEZ VERDUZCO

ASESOR:

ING.Y M.A. SERGIO FCO. BARRAZA IBARRA

MORELIA, MICH. MARZO DEL 2000

CONTENIDO

10 Prefacio	32
1. Introducción	14
11.1 Codificadores y decodificadores	4
11.3 Demultiplexores	7
11.5 Circuitos que realizan operaciones aritméticas	7
11.5.2 Restadores	8
2. Antecedentes históricos	4
3. Objetivo	7
4. Expectativa	7
5. Funcionamiento de las computadoras digitales actuales	8
6. Codificación binaria	12
6.1 Sistemas de numeración	12
6.2 Sistema de numeración binario	13
6.3 Sistema de numeración octal	13
6.4 Sistema de numeración hexadecimal	13
6.5 Suma de números binarios	13
6.6 Sustracción de números binarios	13
6.7 Otras representaciones	13
6.7.1 Módulo y signo	13
6.7.2 Complemento a 1 (C-1)	13
6.7.3 Complemento a 2 (C-2)	13
6.7.4 Exceso a 2 elevado a N-1 (exceso 2^{N-1})	13
7. Principios electrónicos	23
7.1 Corriente eléctrica	23
7.2 Resistencia, tensión, intensidad	70
7.3 Ley de Ohm	72
7.3.1 Resistencia en serie y en paralelo	72
8. Algebra de Boole	26
8.1 Operaciones lógicas básicas	26
8.2 Postulados, propiedades y teoremas	77
9. Compuertas lógicas	29
9.1 La compuerta AND	29
9.2 La compuerta OR	29

9.3 La compuerta NOT

10. Circuitos integrados	32
11. Circuitos combinacionales	34
11.1 Codificadores y decodificadores	
11.2 Multiplexores	
11.3 Demultiplexores	
11.4 Conversores de código	
11.5 Circuitos que realizan operaciones aritméticas	
11.5.1 Sumadores	
11.5.2 Restadores	
11.5.3 Comparadores	
12. Sistemas secuenciales	42
12.1 Biestables o Flip-flops	
12.1.1 El biestable RS	
12.1.2 El biestable JK	
12.1.3 El biestable D	
12.1.4 El biestable T	
12.2 Contadores	
12.3 Registros	
13. Elementos básicos de las computadoras	51
13.1 Operaciones con registro	
13.2 Partes fundamentales de una computadora	
13.2.1 La Unidad Aritmético-Lógica (UAL).	
13.2.2 La Unidad de Control	
13.2.3 La Memoria Principal (MP)	
14. Conclusión	70
I. Cambios tecnológicos en las computadoras	72
II. Glosario	74
III. Bibliografía	77

En estas tres últimas décadas, la ciencia cibernética se ha desarrollado vertiginosamente y está en juego de artefacto en una espectacular gama multifacética.

Los científicos, por el puro hecho de ser tales, se expresan en sus descubrimientos de frente a la Ética, con la más exquisita propiedad y probidad. Desafortunadamente no ha sucedido lo mismo cuando sus inventos caen en manos de algunos políticos y no pocos empresarios, quienes valiéndose **Con todo mi amor, para Salvador y nuestros hijos Carlos Alberto y Mariana.** las deshumanizan poniéndolas al servicio de intereses egoístas.

Nuestro anhelo de seres humanos comunes, clama por un progreso que REGULE el quehacer científico-cibernético, en función de una sana libertad perfectamente razonada.

REGULANDO pues, (poniendo en regla), los avances de estos genios, en función de RAZÓN Y LIBERTAD, se produce el triángulo de toda Ética a la que aspiramos todos los seres humanos y cuyos frutos habrán de revertirse además, hacia la política como hacia todos los empresarios.

Alfonso Fernández Díaz

PREL En estas tres últimas décadas, la ciencia cibernética se ha disparado vertiginosamente y estalla como juego de artificio en una espectacular gama multifacética.

Los científicos, por el puro hecho de ser tales, se expresan en sus descubrimientos de frente a la **Ética**, con la más exquisita propiedad y probidad. Desafortunadamente no ha sucedido lo mismo cuando sus inventos caen en manos de algunos políticos y no pocos empresarios, quienes valiéndose de tan geniales herramientas, las deshumanizan poniéndolas al servicio de intereses mezquinos.

Nuestro anhelo de seres humanos comunes, clama por un progreso que **REGULE** el quehacer científico-cibernético, en función de una sana libertad perfectamente razonada.

REGULANDO pues, (poniendo en regla), los avances de estos genios, en función de **RAZÓN Y LIBERTAD**, se produce el triángulo de toda **Ética** a la que aspiramos todos los seres humanos y cuyos frutos habrán de revertirse además, hacia la política como hacia todos los empresarios.

El propósito del capítulo 6 es introducir los diversos conceptos binarios como marco de referencia para un estudio más detallado en los capítulos subsecuentes. Aquí se explicarán los sistemas de numeración más utilizados por dispositivos internos de las computadoras y sus sistemas de numeración.

Alfonso Fernández Díaz.

Algunos de los principios electrónicos básicos como resistencia, corriente eléctrica, tensión, intensidad y Ley de Ohm en un circuito, serán tratados en el capítulo 7.

De forma sintética se tratarán en el capítulo 8 el álgebra de Boole y sus teoremas y postulados.

A partir del capítulo 9, 10, 11 y 12 se tratarán los circuitos integrados y chips que conforman las computadoras. En esta sección se explica la forma como están diseñados internamente y su funcionamiento.

En el capítulo 13 se presentan los elementos fundamentales de las computadoras. Se hablará de la unidad aritmético lógica, de la unidad de **PREFACIO** la memoria principal. El tratamiento de cada uno de los elementos se realizará de forma general.

Al final de esta tesis se encuentra un apéndice que trata acerca de los **El objetivo principal de este trabajo es presentar en forma sencilla los elementos fundamentales que conforman la arquitectura de las computadoras.**

El capítulo 1 es una introducción a los conceptos de computadora, computadora digital, y arquitectura de computadora.

En el capítulo 2 se hace una breve historia de la manera como han evolucionado las computadoras. En esta sección se presenta un esbozo de las etapas históricas fundamentales para comprender mejor cómo se ha alcanzado el desarrollo actual de las computadoras.

El capítulo 5 se explica de forma general la manera como trabaja una máquina, la forma como la computadora recibe los datos, los traduce en lenguaje de máquina, los manipula y los procesa, y da como resultado información útil.

El propósito del capítulo 6 es introducir los diversos conceptos binarios como marco de referencia para un estudio más detallado en los capítulos subsecuentes. Aquí se explicarán los sistemas de numeración más utilizados por dispositivos internos de las computadoras y sus conversiones a otros sistemas de numeración.

Algunos de los principios electrónicos básicos como resistencia, corriente eléctrica, tensión, intensidad y Ley de Ohm en un circuito, serán tratados en el capítulo 7.

De forma sintética se tratarán en el capítulo 8 el álgebra de Boole y sus teoremas y postulados.

A partir del capítulo 9, 10,11 y 12 se tratarán los circuitos integrados y chips que conforman las computadoras. En esta sección se explica la forma como están diseñados internamente y su funcionamiento.

1. INTRODUCCIÓN

En el capítulo 13 se presentan los elementos fundamentales de las computadoras. Se hablará de la unidad aritmético lógica, de la unidad de control y de la memoria principal. El tratamiento de cada uno de los elementos se realizará de forma general.

Al final de esta tesina se encuentra un apéndice que trata acerca de los cambios tecnológicos a nivel hardware que se ha producido en la industria de la computación y acerca de las tendencias en las tecnologías de implementación de las computadoras.

Deseo agradecer a mi padre, por enseñarme a nunca darme por vencida y ser para mi un ejemplo de lucha constante; y a mi madre, por su apoyo incondicional y por ser mi mejor amiga. Los quiero con todo mi corazón.

Por último quiero expresar mi agradecimiento al Ing. Sergio Barraza Ibarra por su gran ayuda para la realización de esta tesina y mi admiración por la forma tan desinteresada de compartir sus conocimientos conmigo.

La necesidad de automatizar las tareas de cálculo y gestión sobre grandes volúmenes de datos, descomponiendo estas tareas en sencillas operaciones como contar, comparar, ordenar, realizadas muchas veces y a gran velocidad, hizo que a finales del siglo XIX los pioneros en el diseño de equipos informáticos, apoyándose en la aritmética binaria y mediante la utilización de tarjetas perforadas codificadas, pudieran gestionar los censos de grandes poblaciones. A estas máquinas las bautizaron con el nombre de ordenador, por ser la operación de ordenar fichas la más popularmente conocida, pero su nombre técnico es el de computadora digital, ya que toma decisiones a través de unos cálculos comparados con otros ya preestablecidos (computar).

Los avances en las computadoras han crecido gracias a las mejoras tecnológicas y arquitectónicas. Estos avances incluyen formas innovadoras de implementar arquitecturas, así como la adopción de muchas de las técnicas de las grandes computadoras.

1. INTRODUCCIÓN

La palabra arquitectura se utiliza en el área de informática con varios significados. Las computadoras digitales han hecho posibles muchos avances científicos, industriales y comerciales que de otra manera nunca se hubieran alcanzado.

La tecnología de las computadoras ha progresado increíblemente en los últimos cincuenta años. Hoy con unos pocos miles de pesos se puede comprar una computadora personal con más prestaciones, más memoria principal y más memoria en disco que una computadora que en 1965 costaba un millón de dólares. Este rápido crecimiento en las prestaciones es consecuencia de los avances en la tecnología utilizada en la construcción de las computadoras y de las innovaciones en los diseños de los mismos.

En un sentido amplio, la palabra computadora puede aplicarse a cualquier objeto que computa o calcula.

Una computadora digital se puede definir como una máquina programable, compuesta por circuitos de microelectrónica, que procesa secuencialmente información digital, a gran velocidad y con notable precisión gobernada por programas y utilizada en gran número de trabajos.

La necesidad de automatizar las tareas de cálculo y gestión sobre grandes volúmenes de datos, descomponiendo estas tareas en sencillas operaciones como contar, comparar, ordenar; realizadas muchas veces y a gran velocidad, hizo que a finales del siglo XIX los pioneros en el diseño de equipos informáticos, apoyándose en la aritmética binaria y mediante la utilización de tarjetas perforadas codificadas, pudieran gestionar los censos de grandes poblaciones. A estas máquinas las bautizaron con el nombre de ordenador, por ser la operación de ordenar fichas la más popularmente conocida, pero su nombre técnico es el de computadora digital, ya que toma decisiones a través de unos cálculos comparados con otros ya preestablecidos (computar).

Los avances en las computadoras han crecido gracias a las mejoras tecnológicas y arquitectónicas. Estos avances incluyen formas innovadoras de implementar arquitecturas, así como la adopción de muchas de las técnicas de las grandes computadoras.

La palabra arquitectura se utiliza en el área de informática con varias acepciones diferentes:

1. Los pioneros en diseño de sistemas en los años sesenta definieron de esta forma la arquitectura de computadoras: *"...como cualquier otra arquitectura, es el arte de determinar las necesidades del usuario de una estructura y de diseñar para satisfacer lo más eficazmente posible, dentro de las restricciones económicas y tecnológicas."*

2. Se entiende por arquitectura de computadoras la *ciencia moderna que integra el estudio del hardware de los equipos que componen un sistema de Información conjuntamente con el software respectivo y los algoritmos de computación que van a ser implantados en los equipos mediante lenguajes de alto nivel orientados a facilitar el trabajo del usuario.*

El hecho de que evolucionen tan rápidamente los diseños de arquitecturas y que día a día se desarrollen nuevas aplicaciones para las computadoras en áreas muy diversas, da origen a una nueva acepción que complementa a las anteriores.

La arquitectura de computadora contempla las técnicas de interconectividad y compatibilidad de equipos, así como portabilidad de software, y estudia el diseño modular en previsión de ampliación de un Sistema Informático, así como su rentabilidad y costes de explotación.

La arquitectura de computadoras se refiere a la estructura física o al *hardware* de la computadora, a las características de sus partes y su interconexión.

Los arquitectos de computadoras al diseñar una computadora deben cumplir ciertos requerimientos funcionales con determinadas ligaduras de precio y rendimiento. Los requerimientos pueden ser características específicas, inspiradas por el mercado.

2. AN Los circuitos digitales se emplean ahora en todo tipo de productos; desde juguetes para niños hasta computadoras, desde sistemas de telemetría en satélites hasta calculadoras manuales. Debido principalmente al desarrollo de los circuitos integrados de bajo costo los circuitos digitales aparecen actualmente en casi todos los productos electrónicos y se espera que esta tendencia continúe.

El crecimiento del rendimiento de las microcomputadoras ha sido el más rápido en cierto modo porque estas máquinas aprovecharon las ventajas que ofrece la tecnología de circuitos integrados. Además desde 1980, la tecnología de microprocesadores ha sido la tecnología para las nuevas arquitecturas y las nuevas implementaciones de arquitecturas antiguas.

Los cambios significativos en el mercado de computadoras han hecho más fácil que antes el éxito comercial de las nuevas arquitecturas. En primer lugar, la eliminación virtual de la programación en lenguaje ensamblador ha reducido drásticamente la necesidad de la compatibilidad del código objeto. En segundo lugar, la creación de sistemas operativos estandarizados, independientes de los vendedores, como por ejemplo UNIX, ha reducido el coste y riesgo de lanzar al mercado una nueva arquitectura.

Desde la fabricación de las computadoras con válvulas electrónicas, pasando por los fabricados con transistores y posteriormente con circuitos integrados de pequeña y media escala de integración, merece especial interés el estudio de arquitecturas a partir de la llegada del microprocesador de 4 bits, que se introdujo en el mercado en 1971 (años después IBM lanzó al mercado la computadora personal).

Al comienzo de los años ochenta, en los grandes sistemas informáticos se introducen los procesadores de 16 bits, de alta velocidad y ya en 1985 son incorporados a las supercomputadoras los microprocesadores de 32 bits. A finales de esta década el desarrollo de la microelectrónica hace posible la aparición en el mercado de microprocesadores de 32 bits con proceso en paralelo.

* Primera generación - bulbos (1940-1952)

Aquellas computadoras que funcionaban con triodos y otras válvulas electrónicas posteriormente desarrolladas, necesitaban de

2. ANTECEDENTES HISTÓRICOS

Generaciones de las computadoras

El desarrollo tan vertiginoso de los componentes electrónicos básicos utilizados en las computadoras en las últimas décadas, ha variado enormemente su configuración y sus prestaciones. También han contribuido a ello la evolución de las memorias fabricadas a base de semiconductores y la evolución de los soportes de información.

En el proceso de evolución histórica de la moderna computadora digital, se han diseñado y construido cientos de diferentes tipos de computadoras. La mayoría de ellas hace tiempo pasó al olvido, pero algunas ejercieron un impacto significativo en las ideas modernas.

- **Generación cero – computadoras mecánicas (1642-1940)**

La primera persona en construir una máquina calculadora que funcionara fue el científico francés Blaise Pascal (1623-1662). Fue diseñado para recolectar los impuestos. Era un aparato enteramente mecánico, hecho de engranes y accionado a mano por una manivela.

Gottfried W. Von Leibniz (1646-1716) construyó una máquina mecánica que podía sumar, restar, multiplicar y dividir.

Charles Babbage (1792-1871) es considerado por muchos como el padre de la computadora actual al construir en 1822 el modelo funcional de las tablas matemáticas para calcular tablas (máquina de las diferencias).

En 1907, Lee de Forest, intentando perfeccionar los receptores de telegrafía inventa la válvula electrónica triodo, uno de los mayores inventos de nuestro tiempo. Con este invento sienta uno de los pilares fundamentales para el desarrollo de la electrónica de radio, conmutación electrónica y el componente básico de construcción de las primeras computadoras.

- **Primera generación – bulbos (1940-1952)**

Aquellas computadoras que funcionaban con triodos y otras válvulas electrónicas posteriormente desarrolladas, necesitaban de

un caldeo a través de sus filamentos siendo el consumo de energía eléctrica muy elevado. Para evitar problemas de calentamiento había que refrigerarlos, y para evitar zonas calientes era necesario separar las válvulas, con lo que el tamaño de aquellos equipos era muy considerable.

La primera computadora electrónica, la ENIAC, entró en funcionamiento en 1945, tenía un volumen aproximado de 111 metros cúbicos, un peso de 30 toneladas, 17.000 válvulas y un consumo medio de 150.000 vatios.

- **Segunda generación - transistores (1952-1964)**

Walter Brattain y John Barden mediante el estudio de los semiconductores inventan el transistor. Este componente reemplazó rápidamente a las válvulas electrónicas en muchas aplicaciones y concretamente en la fabricación de computadoras por su reducido tamaño y mayor fiabilidad.

En 1954 se construyó el TRADIC, la primera computadora con transistores en los laboratorios Bell, en donde se había descubierto el transistor.

- **Tercera generación - circuitos integrados (1964-1971)**

Los trabajos de física de semiconductores, a comienzos de los años sesenta, orientados a la tecnología de fabricación de componentes electrónicos integrados (transistores, diodos, resistencias y condensadores) a muy reducido volumen, marcaron los primeros pasos de la microelectrónica, con la fabricación de circuitos integrados de baja escala de integración (SSI), con un máximo de 12 puertas lógicas, y conocidas popularmente como chips.

A esta generación le pertenecen las minicomputadoras PDP-11 de la firma DEC, así como la CDC-7600.

- **Cuarta generación - computadoras personales (1971-1981)**

La evolución de la microelectrónica y la gran escala de integración (más de 10.000 puertas lógicas por chip), logró el microprocesador, en

1971, integrando en un solo chip todos los elementos del "cerebro" de la computadora, así como todos los elementos de cálculo.

El microprocesador abrió las puertas a la fabricación de los ordenadores personales. Se desarrolló abundante software para el PC, lo que facilitó su estandarización. Rápidamente apareció una gran cantidad de ordenadores clónicos cuyas arquitecturas eran una copia fiel del PC, para que el software pudiera ser intercambiable. Al final de este período el PC original fue mejorado con la puesta en el mercado de ordenadores personales de 16 bits a 6 MHz.

• **Quinta generación - integración a muy grande escala (1981-?)**

Durante la década de los ochenta se desarrollaron microprocesadores de 16 bits, trabajando a velocidades hasta los 25 MHz.

Fueron los microprocesadores de 32 bits los que redujeron notablemente el volumen de las grandes computadoras (mainframes).

A finales de los años ochenta se consiguieron transistores integrados en estos microprocesadores. Las aplicaciones de diseño gráfico con animación, el procesamiento de grandes volúmenes de datos, la transmisión de imágenes vía radio a través de la computadora, la guerra electrónica, la regulación de robótica en trayectoria continua así como la inteligencia artificial, necesitan de un soporte hardware muy potente, y la tecnología avanza a pasos agigantados para cubrir esas necesidades.

5. FUNCIONAMIENTO DE LAS COMPUTADORAS DIGITALES ACTUALES

Si para manejar una computadora dispusiéramos exclusivamente del hardware de los equipos, ese potencial de circuitería electrónica resultaría poco eficiente. En esta situación y con equipos menos potentes se encontraban los pioneros de la informática, todos ellos muy hábiles, y cuyo software tuvieron que desarrollar con paciencia y perseverancia en lenguaje de máquina.

3. OBJETIVO GENERAL

Presentar los conocimientos básicos para la comprensión de la composición, organización y funcionamiento de la arquitectura de computadoras, mostrando los dispositivos lógicos utilizados en su diseño.

4. OBJETIVOS PARTICULARES

- Identificará los diferentes sistemas de numeración que maneja la computadora.
- Comprenderá las operaciones de suma y sustracción binaria.
- Conocerá los postulados y las operaciones básicas del álgebra de Boole.
- Comprenderá las compuertas lógicas básicas que se utilizan para construir los circuitos integrados.
- Conocerá algunos de los principios electrónicos básicos, como resistencia, voltaje, corriente.
- Identificará los circuitos combinacionales y secuenciales que forman parte de una computadora.
- Entenderá el funcionamiento de las partes fundamentales de una computadora y la forma como se manipula la información en ellas.

La figura 5-1 muestra un diagrama de bloques de la computadora digital. La unidad de memoria almacena los programas al igual que la entrada, la salida y la información intermedia. El procesador realiza las tareas aritméticas y otros procesamientos de información como se especifica por un programa.

La unidad de control supervisa el flujo de información entre las diversas unidades. Recupera las instrucciones, una por una, del programa que está almacenado en la memoria. También supervisa las

5. FUNCIONAMIENTO DE LAS COMPUTADORAS DIGITALES ACTUALES

Si para manejar una computadora dispusiéramos exclusivamente del hardware de los equipos, ese potencial de circuitería electrónica resultaría poco eficiente. En esta situación y con equipos menos potentes se encontraban los pioneros de la informática, todos ellos muy identificados con el hardware, y cuyo software tuvieron que desarrollar con paciencia y perseverancia en lenguaje de máquina.

Es característico de un sistema digital la manipulación de elementos discretos de información. Es posible que dichos elementos sean impulsos eléctricos, dígitos decimales, las letras de un alfabeto, operaciones aritméticas, etc. Los elementos discretos de información se representan en un sistema digital mediante cantidades físicas denominadas señales. Las señales eléctricas como voltajes y corrientes son las más comunes. Las señales en todos los sistemas digitales en la actualidad tienen sólo dos valores discretos y se dice que son binarios.

El diseñador de un sistema digital está limitado al uso de señales binarias debido a la confiabilidad más alta que la de los circuitos electrónicos de valores múltiples. Es decir, puede diseñarse un circuito con diez estados que usa un valor discreto de voltaje para cada estado, pero tendría una confiabilidad muy baja de operación. En contraste, un circuito de transistor que está, ya sea encendido o apagado, tiene dos valores de señal posibles y puede construirse para que sea en extremo confiable (una computadora analógica realiza una simulación directa de un sistema físico. Cada sección de la computadora es la análoga de una porción particular del proceso bajo estudio. Las variables en la computadora analógica se representan por señales continuas, por común voltajes eléctricos que varían con el tiempo.)

La figura 5-1 muestra un diagrama de bloques de la computadora digital. La unidad de memoria almacena los programas al igual que la entrada, la salida y la información intermedia. El procesador realiza las tareas aritméticas y otros procesamientos de información como se especifica por un programa.

La unidad de control supervisa el flujo de información entre las diversas unidades. Recupera las instrucciones, una por una, del programa que está almacenado en la memoria. También supervisa las

instrucciones del programa y el procesador manipula la información como se especifica en el programa.

El programa y la información son introducidos mediante un dispositivo de entrada (p.ej. el teclado). Un dispositivo de salida (p.ej. la impresora) recibe el resultado de los cálculos e imprime los resultados que se presentan al usuario (los dispositivos de entrada y salida son sistemas digitales especiales impulsados por partes electromecánicas y controladas por circuitos digitales electrónicos).

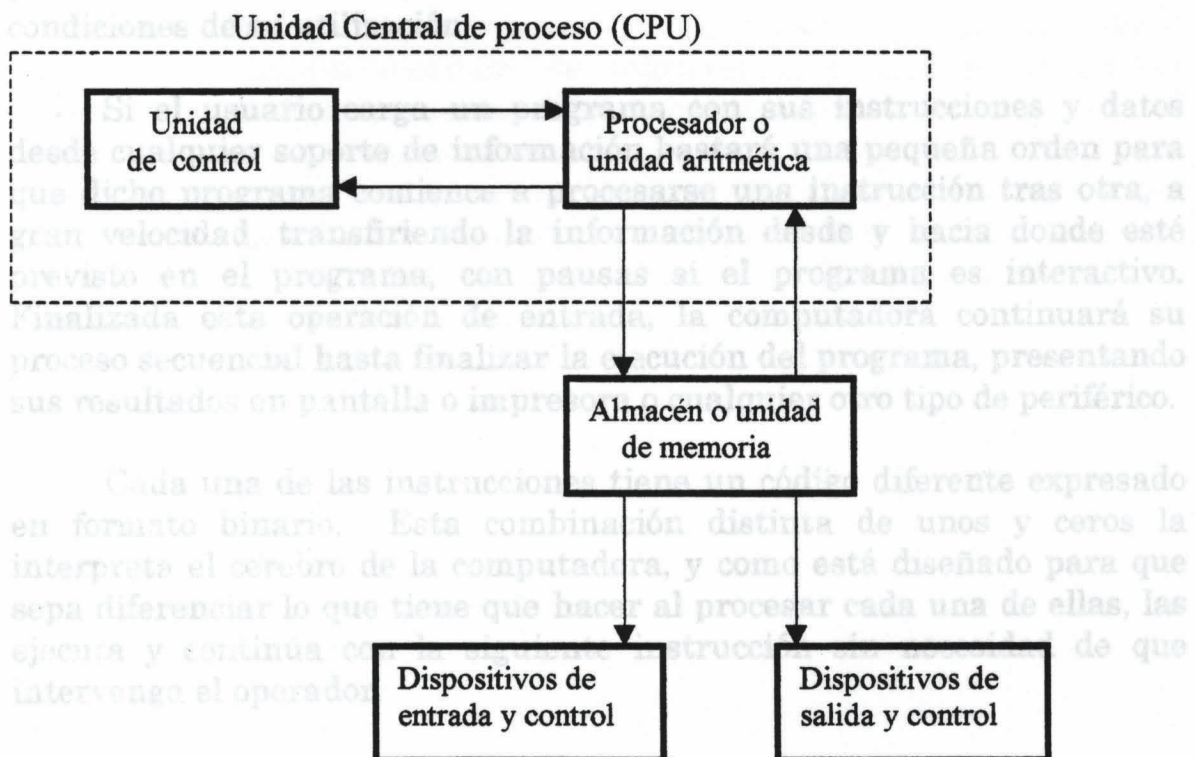


Figura 5-1 Diagrama de bloques de una computadora digital

Un procesador, cuando se combina con la unidad de control forma una unidad central de proceso o CPU. Una CPU encerrada en un pequeño paquete de circuito integrado se denomina microprocesador. Es posible que la unidad de memoria, lo mismo que la parte que controla la interfase entre el microprocesador y los dispositivos de entrada y salida esté encapsulada dentro del paquete del microprocesador. Una CPU

combinada con memoria y control de interfase se conoce como microcomputadora.

Al inicial el arranque, en la gran mayoría de las computadoras, independientemente de su tamaño y potencia, el control pasa mediante circuito cableado a unas memorias (ROM) grabadas con información permanente. Después de la lectura de esta información el circuito de control mandará cargar en la memoria principal, desde algún soporte externo (generalmente disco) los programas del sistema operativo que controlarán las operaciones a seguir, y en pocos segundos aparecerá en pantalla el identificador, dando muestras al usuario de que ya está en condiciones de su utilización.

Si el usuario carga un programa con sus instrucciones y datos desde cualquier soporte de información bastará una pequeña orden para que dicho programa comience a procesarse una instrucción tras otra, a gran velocidad, transfiriendo la información desde y hacia donde esté previsto en el programa, con pausas si el programa es interactivo. Finalizada esta operación de entrada, la computadora continuará su proceso secuencial hasta finalizar la ejecución del programa, presentando sus resultados en pantalla o impresora o cualquier otro tipo de periférico.

Cada una de las instrucciones tiene un código diferente expresado en formato binario. Esta combinación distinta de unos y ceros la interpreta el cerebro de la computadora, y como está diseñado para que sepa diferenciar lo que tiene que hacer al procesar cada una de ellas, las ejecuta y continúa con la siguiente instrucción sin necesidad de que intervenga el operador.

El proceso de una instrucción se descompone en operaciones muy simples de transferencia de información u operaciones aritméticas y lógicas elementales, que realizadas a gran velocidad le proporcionan una gran potencia que es utilizada en múltiples aplicaciones.

Realmente esa información digitalizada en binario, a la que se refiere con unos y ceros, la computadora la diferencia porque se trata de niveles diferentes de voltaje. Cuando se emplean circuitos integrados, los niveles lógicos bajo y alto (representados por 0 y 1), corresponden a valores muy próximos a 0 y 5 voltios en la mayoría de los casos.

6.00 Cuando a las entradas de las compuertas lógicas de los circuitos digitales se les aplica el nivel alto o bajo de voltaje, el comportamiento es muy diferente en cada caso. Por ejemplo, al aplicarles el nivel alto conducen, esto es dejan abierto un circuito. Para que esto ocurra los transistores que constituyen los circuitos integrados trabajan en conmutación, pasando del corte a la saturación. Abriendo y cerrando a gran velocidad los circuitos simultáneamente en sincronismo con los impulsos que proporciona un circuito reloj, la información se transfiere por las diferentes unidades de la computadora, según la instrucción de que se trate hasta conseguir ejecutar el programa por completo.

Como se ha mencionado anteriormente, una computadora digital manipula elementos discretos de información y esos elementos se representan en forma binaria. Los operandos que se utilizan para el cálculo pueden expresarse en el sistema de números binarios. Otros elementos discretos, incluyendo los dígitos decimales, se representan con códigos binarios. El procesamiento de datos se lleva a cabo mediante elementos lógicos binarios que usan señales binarias y las cantidades se almacenan en elementos de almacenamiento binario.

- Sistema binario. Base 2 (2^1).
- Sistema octal Base 8 (2^3).
- Sistema hexadecimal Base 16 (2^4).

El sistema que maneja internamente una computadora es el binario, pero en ocasiones y por comodidad en el manejo de datos se suelen utilizar en octal, en el que una cifra equivale a tres dígitos binarios, y el hexadecimal, en el que una cifra equivale a cuatro dígitos binarios.

6.2 Sistema de numeración binario.

El sistema de números binarios sólo utiliza dos símbolos (1,0); cada número representado en este sistema se denomina bit (binary digit). La forma de contar en binario se muestra en la figura 6.2-1. El número decimal se encuentra a su derecha con su decimal equivalente a la izquierda. Nótese que el bit menos significativo (bms) está en el lugar de las unidades.

En el sistema binario, el valor de cada dígito se basa en el dos y en las potencias de dos. Es decir, el dígito del extremo derecho se multiplica

6. CODIFICACIÓN BINARIA

Para entender cómo funciona una computadora, es necesario saber lo que es la representación binaria de un número en la lógica digital. La primera idea que llevó a realizar una computadora nació del deseo de construir una máquina que simulara las actividades del cerebro humano y que captara la información de estímulos percibidos, la procesara y generara respuestas que fueran interpretadas por distintos órganos.

6.1 Sistemas de numeración

Podemos definir un sistema de numeración como el conjunto de símbolos y reglas que se utilizan para la representación de cantidades. En ellos existe un elemento característico que define el sistema y que se denomina base, siendo ésta el número de símbolos que se utilizan para dicha representación.

En electrónica digital se utilizan tres sistemas de numeración, cuyas bases son potencias de 2. Estos sistemas son:

- Sistema binario. Base 2 (2^1).
- Sistema octal Base 8 (2^3)
- Sistema hexadecimal Base 16 (2^4).

El sistema que maneja internamente una computadora es el binario, pero en ocasiones y por comodidad en el manejo de datos se suelen utilizar en octal, en el que una cifra equivale a tres dígitos binarios, y el hexadecimal, en el que una cifra equivale a cuatro dígitos binarios.

6.2 Sistema de numeración binario.

El sistema de números binarios sólo utiliza dos símbolos (1,0); cada número representado en este sistema se denomina bit (binary digit). La forma de contar en binario se muestra en la figura 6.2-1. El número decimal se encuentra a su derecha con su decimal equivalente a la izquierda. Nótese que el bit menos significativo (bms) está en el lugar de las unidades.

En el sistema binario, el valor de cada dígito se basa en el dos y en las potencias de dos. Es decir, el dígito del extremo derecho se multiplica

por 1 (2^0), el segundo dígito de la derecha se multiplica por 2 (2^1), el tercero se multiplica por 4 (2^2) y así sucesivamente.

Conteo decimal	Conteo binario				
	16	8	4	2	1
0					0
1					1
2				1	0
3				1	1
4			1	0	0
5			1	0	1
6			1	1	0
7			1	1	1
8		1	0	0	0
9		1	0	0	1
10		1	0	1	0
11		1	0	1	1
12		1	1	0	0
13		1	1	0	1
14		1	1	1	0
15		1	1	1	1
16	1	0	0	0	0
17	1	0	0	0	1
18	1	0	0	1	0
19	1	0	0	1	1

Fig. 6.2-1

Un número decimal se puede convertir en número binario mediante la división del número decimal por dos tantas veces como sea necesario para reducir el cociente a un número menor de 2 (0 ó 1), utilizando los residuos por cada paso de la división como número binario. Esto se muestra en la figura 6.2-2

Conteo decimal	Conteo binario	Conteo octal
0	000	0
1	001	1
2	010	2
3	011	3
4	100	4
5	101	5
6	110	6
7	111	7
8	1000	10
9	1001	11
10	1010	12
11	1011	13

Figura 6.3-1 Forma de contar en los sistemas decimales, binario y octal.

El sistema octal también utiliza el valor de posición. La figura 3.3-1 muestra el valor de los cuatro primeros lugares a la izquierda del punto

Divisores sucesivos	Número original y dividendo	Residuo (número binario)
2	33	1
2	16	0
2	8	0
2	4	0
2	2	0
2	1	1

$$33_{10} = 100001_2$$

Fig. 6.2-2 Conversión de los números decimales en números binarios por medio de la división.

6.3 Sistema de numeración octal.

El sistema octal es el de base 8, y los símbolos que utiliza son 0, 1, 2, 3, 4, 5, 6 y 7. La utilidad del sistema octal radica en que posee un símbolo diferente para cada número binario del 000 al 111. La figura 6.3-1 compara cómo se cuenta en los sistemas decimal, binario y octal.

Conteo decimal	Conteo binario	Conteo octal
0	000	0
1	001	1
2	010	2
3	011	3
4	100	4
5	101	5
6	110	6
7	111	7
8	1000	10
9	1001	11
10	1010	12
11	1011	13

Figura 6.3-1 Forma de contar en los sistemas decimales, binario y octal.

El sistema octal también utiliza el valor de posición. La figura 3.3-1 muestra el valor de los cuatro primeros lugares a la izquierda del punto

octal. El dígito menos significativo (dms) es el que está en el lugar de las unidades. El valor o peso de las posiciones es 1, 8, 64, 512, etc.

Potencias de 8	8^3	8^2	8^1	8^0
Valor de posición (en decimales)	512	64	8	1

Figura 6.3-2 Valores de posición en el sistema octal

A continuación se muestra el procedimiento para convertir un número octal a decimal:

Número octal	1	2	3_8	
	64	8	1	→ Valores de posición
	<u>x1</u>	<u>x2</u>	<u>x3</u>	→ Número octal
Número decimal	64	+ 16	+ 3	= 83_{10}

El proceso para convertir números decimales a octales es similar al que se utiliza para convertir decimales a binarios. A continuación se ejemplifica el procedimiento en la figura 3.3-2:

Divisores sucesivos	Número original y dividendo	Residuo (número octal)
8	1327	7
8	165	5
8	20	4
8	4	2

$1327_{10} = 2\ 4\ 5\ 7_8$

Figura 6.3-3 Conversión de números decimales a octales

Decimal	Binario	Hexadecimal	Decimal	Hexadecimal
0	0000	0	$1327_{10} = 2\ 4\ 5\ 7_8$	10
1	0001	1	17	10001
2	0010	2	18	10010
3	0011	3	19	10011
4	0100	4	20	10100

La utilidad del sistema octal está en su facilidad de conversión a binario. Considérese el número octal 532. Para efectuar esta conversión basta memorizar tan sólo los primeros ocho números de la cuenta binaria (000 – 111) y sus respectivos octales equivalentes, como se muestra en la figura 6.3-1. La conversión del octal 532_8 a binario se efectúa como sigue (nótese que cada dígito octal forma un grupo de tres dígitos binarios):

11	1011	B	27	11011	1B
12	1100	C	28	11100	1C
13	1101	D	29	11101	1D
5	101	5_8	30	11110	1E
	011	3_8	31	11111	1F
	010	2_8			

Figura 6.3-1

$532_8 = 101011010_2$

101 011 010

Para convertir de binario a octal se invierte el proceso. El número binario dado se debe dividir en grupo de tres bit cada uno, empezando por la izquierda:

110	111	000	100 ₂	
↓	↓	↓	↓	
6	7	0	4 ₈	$110111000100_2 = 6704_8$
				Valor de posición 4096 256 16 1
				(en decimales)

6.4 Sistema de numeración hexadecimal

El sistema hexadecimal es el sistema de números de base 16. Utiliza símbolos del 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F.

En la figura 6.4-1 se muestra la forma de contar en los sistemas decimal, binario y hexadecimal.

Decimal	Binario	Hexadecimal	Decimal	Binario	Hexadecimal
0	0000	0	16	10000	10
1	0001	1	17	10001	11
2	0010	2	18	10010	12
3	0011	3	19	10011	13
4	0100	4	20	10100	14

5	0101	5	21	10101	15
6	0110	6	22	10110	16
7	0111	7	23	10111	17
8	1000	8	24	11000	18
9	1001	9	25	11001	19
10	1010	A	26	11010	1A
11	1011	B	27	11011	1B
12	1100	C	28	11100	1C
13	1101	D	29	11101	1D
14	1110	E	30	11110	1E
15	1111	F	31	11111	1F

Figura 6.4-1

Para realizar la conversión de hexadecimal a decimal se efectúa el mismo proceso que en la conversión de hexadecimal a decimal. Los valores de posición se muestran en la figura 6.4-2. A continuación se ejemplifica el procedimiento:

Potencias de 16	16^3	16^2	16^1	16^0
Valor de posición (en decimales)	4096	256	16	1

Figura 6.4-2 Valores de posición en el sistema hexadecimal

Conversión de hexadecimal a decimal:

Número hexadecimal	2	B	6_{16}	
	256	16	1	→ Valores de posición
	<u>x2</u>	<u>x11</u>	<u>x6</u>	→ Número octal
Número decimal	$512 + 176 + 6 = 694_{10}$			

La principal ventaja del sistema hexadecimal es su facilidad para convertirlo a binario. A continuación se ejemplifica el procedimiento para la conversión de hexadecimal a binario y de binario a hexadecimal. Cada dígito hexadecimal forma un grupo de cuatro dígitos binarios

Conversión de binario a hexadecimal:

$$\begin{array}{ccc}
 1010 & 1000 & 0101_2 \\
 \downarrow & \downarrow & \downarrow \\
 A & 8 & 5_{16}
 \end{array}
 \quad
 101010000101_2 = A85_{16}$$

Conversión de hexadecimal a binario:

$$\begin{array}{ccc}
 & B & 9_{16} \\
 \downarrow & \downarrow & \downarrow \\
 0011 & 1011 & 1001
 \end{array}
 \quad
 3B9_{16} = 1110111001_2$$

6.5 Suma de números binarios

Las reglas para sumar números binarios son:

- $0 + 0 = 0$
- $0 + 1 = 1$
- $1 + 0 = 1$
- $1 + 1 = 0$ con un 1 que se lleva (acarreo).

Existe una regla especial para sumar columnas de números binarios. Si la cantidad de los unos en cualquier columna individual es mayor de 2, se divide el número de unos entre 2. El número de veces que el 2 dividirá al número de unos, es la cantidad de unos que se lleva a la columna siguiente. Si el número de unos divide exactamente entre 2, entonces el total de la columna se escribe como cero. Si la columna no divide exactamente, entonces tiene residuo 1, y ése 1 se escribe como el total de la columna.

Por ejemplo:

Suma binaria **Suma decimal**

1 0 1	→	5
1 0 1	→	5
1 1 1	→	7
1 0 0 0 1	→	17

6.6 Sustracción de números binarios

Las reglas para la sustracción de números binarios son:

- 0 - 0 = 0
- 1 - 1 = 0
- 1 - 0 = 1
- 0 - 1 = 1 con una cifra prestada de 1



Supongamos que el decimal 33 (binario 100001) se resta del decimal 77 (1001101), la operación se haría de la siguiente manera:

$$\begin{array}{r}
 1001101 \\
 - 100001 \\
 \hline
 101100
 \end{array}$$

6.7 Otras representaciones

Una computadora maneja datos en binario con una limitación de longitud referida al número de bits, y además necesita considerar el signo para poder operar con números negativos. Existen cuatro formas habituales que se utilizan para estas representaciones. Estas son:

- Módulo y signo (MS)
- Complemento a 1 (C-1)
- Complemento a 2 (C-2)
- Exceso a 2 elevando a N-1 (exceso 2^{N-1})

El número de dígitos disponibles lo representaremos por N.

signo -. Para los números positivos, los N-1 bits de la derecha representan el módulo (igual que en MS). El negativo de un número se obtiene complementando todos sus dígitos (es decir, cambiando ceros por unos y viceversa), incluido el bit de signo.

A continuación se muestra la representación en C-1 de los números 10 y -10 para 8 bits.

Número 10

↑
Signo + 0 0 0 0 1 0 1 0
 └──────────────────┘
 módulo

Número -10

(complemento de su positivo)

↑
Signo - 1 1 1 1 0 1 0 1
 └──────────────────┘
 módulo

El rango de representación es el mismo que en MS.

6.7.3 Complemento a 2 (C-2)

Este es un método a través del cual se pueden realizar restas sumando. Utiliza el bit situado más a la izquierda para el signo, correspondiendo el 0 para el signo + y el 1 para el signo -. Para los números positivos los N-1 dígitos a la derecha representan el módulo (igual que en MS y C-1).

El negativo de un número se obtiene en dos pasos:

1. Se complementa el número positivo en todos sus bits (cambiando ceros por unos y viceversa), incluido el bit de signo.
2. Al resultado obtenido en el primer paso se le suma 1 (en binario) despreciando el último acarreo si existe.

A continuación se ejemplifica la representación en C-2 de los números 10 y -10.

Número 10

↑
Signo + 0 0 0 0 1 0 1 0
 └──────────────────┘
 módulo

7. PRINCIPIOS ELECTRÓNICOS

Número -10 Primer paso 1 1 1 1 0 1 0 1

7.1 Corriente eléctrica

Segundo paso 1 1 1 1 0 1 0 1

Se le llama corriente eléctrica a la circulación de cargas eléctricas a través de un material conductor provocando un potencial eléctrico.

Existen corrientes eléctricas, en general, en los conductores que producen en un circuito cargas eléctricas y la energía eléctrica (un conductor es un cuerpo en cuyo interior las cargas libres mueven por la fuerza ejercidas sobre ellas por un campo eléctrico). Las cargas libres en un conductor...

El rango de representación en este caso es asimétrico, lo que representa su mayor inconveniente y viene dado por:

La intensidad de corriente que pasa por un hilo está determinada por la fórmula:

$$-2^{N-1} \leq X \leq 2^{N-1} - 1$$

Para 8 bits:

$$-128 \leq X \leq 127$$

Para 16 bits:

$$-32768 \leq X \leq 32767$$

Para 32 bits:

$$-2147483648 \leq X \leq 2147483647$$

Su principal ventaja es que tiene una única representación para el 0.

i = corriente (en amperios)

6.7.4 Exceso a 2 elevado a N-1 (exceso 2^{N-1})

t = unidad de tiempo

Este sistema no utiliza bit para el signo, con lo cual todos los bits representan un módulo que corresponde al número representado más el exceso, que para N bits viene dado por 2^{N-1} .

Por ejemplo, para 8 bits el exceso es de 128, con lo que el número 10 vendrá representado por $10 + 128 = 138$, y para el -10 tendremos $-10 + 128 = 118$.

Sus representaciones serían

Número 10 1 0 0 0 1 0 1 0 (138 en binario)

Número -10 0 1 1 1 0 1 1 0 (118 en binario)

El 0 tiene una única representación, que para 8 bits sería

Número 0 (0 + 128) 1 0 0 0 0 0 0 0

El rango de representación es asimétrico, y es igual al de C-2.

7. PRINCIPIOS ELECTRÓNICOS

7.1 Corriente eléctrica

Se le llama corriente eléctrica a la circulación de cargas eléctricas a través de un material conductor provocando un potencial eléctrico. Existe corriente eléctrica, en general cuando un generador produce en un circuito cargas eléctricas y la energía para impulsarlas (un conductor es un cuerpo en cuyo interior hay cargas libres que mueven por la fuerza ejercidas sobre ellas por un campo eléctrico). Las cargas libres en un conductor metálico son electrones negativos.

La intensidad de corriente que pasa por un hilo está determinada por la fórmula:

□

$$i = \frac{q}{t}$$

Donde:

i = corriente (en amperios)

q = carga transportada (en coulombios)

t = unidad de tiempo

7.2 Resistencia, tensión, intensidad

No todos los materiales permiten que a través de ellos circulen cargas eléctricas, por eso se presentan dos tipos de materiales bien diferenciados: los conductores y los aislantes.

- Se define como resistencia eléctrica a la oposición de un material al paso de la corriente eléctrica. Se mide en ohmios (Ω), y está dada por la fórmula:

□

$$R = \rho \cdot l / s$$

Donde: Resistencias en serie y en paralelo

ρ es la resistividad en $\Omega \text{ mm}^2/\text{m}$

l es la longitud en m

s es la sección en mm^2

- La tensión es la diferencia de potencial eléctrico que existe entre los extremos de un circuito y la causa del movimiento de los electrones. Se mide en voltios (V) y en circuitos electrónicos se produce por medio de generadores o fuentes de alimentación.

Se llama sentido de la corriente al sentido en el cual se mueven las cargas positivas y es contrario al sentido de las cargas negativas o electrones

- La intensidad es la cantidad de corriente eléctrica (q) que circula por un conductor, es decir, el número de electrones que circulan por unidad de tiempo (t). Se mide en amperios (A) y en general tiene un determinado sentido que suele tomarse al contrario del movimiento de los electrones. Este sentido siempre es el mismo en corriente continua y cambia en corriente alterna. Está dada por la fórmula:

$$I = q/t \quad \therefore \quad q = It$$

7.3 Ley de Ohm

En 1827, el matemático alemán Gregor Simon Ohm, estableció el principio por el que se rige el movimiento de electrones a través de un conductor y que se conoce con el nombre de Ley de Ohm, cuyo enunciado es el siguiente: "La intensidad que circula por un conductor es directamente proporcional a la tensión V entre sus extremos e inversamente proporcional a su resistencia R ". Esta ley se representa mediante la fórmula:

$$I = \frac{V}{R}$$

7.3.1 Resistencias en serie y en paralelo

En un circuito eléctrico, las resistencias pueden aparecer acopladas de diversas formas, en las que resulta necesario conocer el valor equivalente de resistencia en un conjunto.

- Resistencias en serie

Se conecta el final de cada una de ellas con el principio de la siguiente, con la característica de que todas ellas están sometidas al paso de una corriente de igual valor en el caso de tener sometidos a sus extremos una diferencia de potencial. La resistencia equivalente es igual a la suma de las resistencias conectadas.

$$R_{\text{equ}} = R_1 + R_2 + \dots + R_n$$

- Resistencias en paralelo

Se conecta por un lado un extremo de cada resistencia y por el otro el resto, de tal forma que la intensidad que pasa por cada una de ellas no lo hace por las demás; es decir, la intensidad de corriente que circula por el conjunto se reparte entre las distintas resistencias acopladas de modo inverso a la resistencia de cada una de ellas. La resistencia equivalente del conjunto es igual al inverso de la suma de los inversos de cada resistencia.

$$R_{\text{equ}} = \frac{1}{\frac{1}{R_1} + \frac{1}{R_2} + \dots + \frac{1}{R_n}}$$

- Acoplamiento mixto de resistencias

Cualquier combinación de resistencias que no sea serie ni paralelo se denomina mixto. La resistencia equivalente en cada caso vendrá dada por una mezcla adecuada de las dos fórmulas anteriores.

$$R_{\text{equ}} = R_1 + \frac{1}{\frac{1}{R_2} + \frac{1}{R_3}}$$

8. ALGEBRA DE BOOLE

La función intersección también se denomina conjunción, producto lógico.

En 1854, el matemático inglés George Boole desarrolló una teoría matemática que permitió la representación de circuitos de conmutación. El álgebra de Boole tiene cierta similitud con el álgebra convencional y está formado por variables lógicas (a, b, c, \dots), operadores lógicos ($\wedge, \vee, -$) y un conjunto de leyes que rigen ciertas combinaciones de los elementos anteriores.

Las variables lógicas pueden tomar solamente dos valores o estados, verdadero o falso, que pueden ser representados simbólicamente por 1 y 0, respectivamente.

En los circuitos lógicos electrónicos, la interpretación física o equivalencia de estos valores lógicos o estados suele estar asociada a la presencia o ausencia de tensión en las que según el dispositivo o circuito lógico, tendrán dos valores fijos (por ejemplo, 5 y 0 voltios aproximadamente).

8.1 Operaciones lógicas básicas

La función negación también se denomina complementación o inversión.

Las operaciones lógicas básicas son aquellas en las que interviene un solo operador o relación de los que pueden utilizarse (\wedge, \vee o $\bar{}$).

- **Función UNIÓN (\vee)**

La función unión también se denomina disyunción, suma lógica o simplemente O (OR en inglés), y puede ser representada por + y por \cup . Opera entre dos variables o valores lógicos y su valor es 1 si cualquiera de ellos o los dos lo son. Si ambos son 0 el valor de la función es 0.

Tabla de la función $F = a \vee b$

a	b	F
0	0	0
0	1	1
1	0	1
1	1	1

Tabla de la función $F = a \vee b$

Los postulados más importantes son:

- **Función INTERSECCIÓN (\wedge)**

La función intersección también se denomina conjunción, producto lógico o simplemente Y (AND en inglés), y puede ser representada por \cdot y por la Y.

Opera igualmente con dos variables o valores lógicos y su valor es 1 si ambos operandos lo son, siendo 0 en el resto de los casos.

a	b	F
0	0	0
0	1	0
1	0	0
1	1	1

Tabla de la función $F = a \wedge b$

- **Función NEGACIÓN ($\bar{}$)**

La función negación también se denomina complementación o simplemente NO (NOT en inglés), y puede ser representada por \neg o por la palabra NO. Opera sobre una única variable lógica, siendo el valor de la función el contrario al de la variable

a	F
0	1
1	0

Tabla de la función $F = \bar{a}$

8.2 Postulados, propiedades y teoremas

En el análisis de funciones lógicas es necesario el conocimiento de los postulados del álgebra de Boole, fundamentalmente para procesos de simplificación. Supongamos dos variables lógicas, a y b.

Los postulados más importantes son:

- $a \vee 1 = 1$

9. COMPUERTAS LÓGICAS

- $a \vee 0 = a$

- $a \wedge 1 = a$

- $a \wedge 0 = 0$

- $a \vee a = a$

- $a \wedge a = a$

- $a \vee \bar{a} = 1$

- $a \wedge \bar{a} = 0$

- $\bar{\bar{a}} = a$

En el álgebra de Boole se cumplen las siguientes propiedades:

Conmutativa $a \vee b = b \vee a$ $a \wedge b = b \wedge a$

Asociativa $a \vee b \vee c = a \vee (b \vee c)$ $a \wedge b \wedge c = a \wedge (b \wedge c)$

Distributiva $a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$ $a \wedge (b \vee c) = a \wedge b \vee a \wedge c$

También se cumplen los siguientes teoremas:

Absorción $a \vee (a \wedge b) = a$ $a \wedge (a \vee b) = a$

Leyes de De Morgan $\overline{a \vee b} = \bar{a} \wedge \bar{b}$ $\overline{a \wedge b} = \bar{a} \vee \bar{b}$



Símbolo de la compuerta AND

Fig. 9.1-1

9. COMPUERTAS LÓGICAS

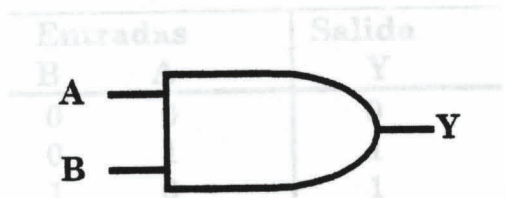
La compuerta lógica es el elemento básico en los sistemas digitales. Es un pequeño circuito, generalmente integrado en una pastilla, que responde desde el punto de vista lógico a una función básica. Las compuertas lógicas operan con números binarios. Por esta razón se les llaman compuertas lógicas binarias. Todos los voltajes usados en las compuertas lógicas serán ALTO o BAJO. Por lo general ALTO voltaje significa 1 binario y BAJO voltaje significa 0 binario.

Las variables que maneja una compuerta lógica son sus entradas, mientras que la función viene dada por su salida. El circuito que configura una puerta lógica está formado por un pequeño número de componentes discretos, como resistencias, diodos, transistores, etc.

Todos los sistemas digitales se construyen usando sólo tres compuertas lógicas básicas. A estas compuertas se les conoce como compuerta AND, compuerta OR y compuerta NOT.

9.1 La compuerta AND

A la compuerta AND se le llama compuerta todo o nada. En la figura 9.1-1 se muestra el símbolo lógico convencional de la compuerta AND. Éste símbolo señala las entradas como A y B. A la salida se le señala como Y. En la tabla de la figura 9.1-2 se muestran todas las posibles combinaciones para las entradas A y B. A esta tabla se le llama tabla de verdad. Las entradas se representan como dígitos binarios (bit). Como se puede ver, sólo cuando ambas entradas A y B son 1, la salida será 1.



Símbolo de la compuerta AND

Fig. 9.1-1

9.3 La compuerta NOT

A la compuerta NOT también se le conoce como inversor. En la figura 9.3-1 se muestra el símbolo lógico para la compuerta NOT. La figura 9.3-2 muestra el tablero de verdad para la compuerta NOT. Si la entrada en la compuerta es 0, la salida será el complemento u opuesto (1). Si la entrada es 1, la salida será el complemento u opuesto (0). Esta inversión también se llama negación o complemento.

Entradas		Salida
B	A	Y
0	0	0
0	1	0
1	0	0
1	1	1

Tabla de verdad para AND

Fig. 9.1-2

9.2 LA COMPUERTA OR

A la compuerta OR se le llama compuerta de "cualquiera o todo". En la figura 9.2-1 se muestra el símbolo de la compuerta OR. La compuerta OR tiene dos entradas, llamadas A y B y una salida Y . La expresión booleana para la función OR está dada por $A + B = Y$. Nótese que el signo (+) significa OR en álgebra booleana (el signo + no significa suma como en el álgebra regular). La tabla de verdad para la compuerta OR se muestra en la figura 9.2-2. La compuerta OR es habilitada (la salida es 1) cada vez que aparece un 1 en alguna o todas las entradas.

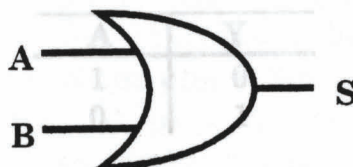


Tabla de verdad para OR

Símbolo de la compuerta OR

Fig. 9.2-1

Entradas		Salida
B	A	Y
0	0	0
0	1	1
1	0	1
1	1	1

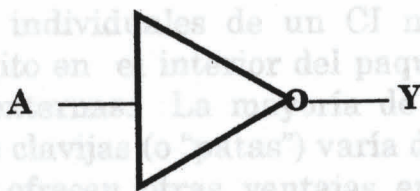
Tabla de verdad para OR

Fig. 9.2-2

10. CIRCUITOS INTEGRADOS

9.3 La compuerta NOT

Los circuitos digitales en forma invariable se construyen con circuitos integrados. La compuerta NOT también se le conoce como inversor. En la figura 9.3-1 se muestra el símbolo lógico para la compuerta NOT. La figura 9.3-2 muestra la tabla de verdad para la compuerta NOT. Si la entrada en la compuerta es 0, la salida será el complemento u opuesto (1). Si la entrada es 1, el circuito dará un 0. Esta inversión también se llama negación o complemento.



Símbolo de la compuerta OR

Fig. 9.3-1

Conforme ha mejorado la tecnología de los CI, el número de compuertas que pueden colocarse dentro de una sola pastilla de silicio ha aumentado en forma considerable. La diferenciación entre los CI que tienen una cuantas compuertas inversas y los que tienen decenas o cientos de compuertas, se hace por una referencia acostumbrada de que un paquete es un dispositivo pequeño, mediana o gran escala de integración. Los fabricantes dividen comúnmente a los circuitos integrados en grupos, basados en la complejidad de los circuitos:

Tabla de verdad para OR

Fig. 9.3-2

- VLSI (integración a muy grande escala)

Un concepto bajo el cual la función de un sistema completo se fabrica como un solo microcircuito. En este contexto un sistema, ya sea digital o lineal, contiene 1000 o más compuertas, o circuitos de similar complejidad.

- MSI (integración a gran escala)

Un concepto bajo el cual un subsistema mayor o las funciones de un sistema se fabrican como un solo microcircuito. En este contexto, un sistema mayor o subsistema, ya sea digital o lineal, es aquel que contiene 100 o más compuertas equivalentes o circuitos de complejidad similar.

10. CIRCUITOS INTEGRADOS

Los circuitos digitales en forma invariable se construyen con circuitos integrados. Un circuito integrado (CI) es un cristal semiconductor pequeño de silicio llamado pastilla, que contiene componentes eléctricos como transistores, diodos, resistencias y capacitores. Los diversos componentes están interconectados dentro de la pastilla para formar un circuito electrónico. La pastilla se monta en un paquete de metal o plástico y se soldan conexiones a las clavijas externas para formar el CI.

Los componentes individuales de un CI no pueden separarse o desconectarse y el circuito en el interior del paquete es accesible sólo a través de las clavijas externas. La mayoría de los CI tienen tamaño estándar y el número de clavijas (o "patas") varía de 8 a 64. El tamaño es muy pequeño, además ofrecen otras ventajas en comparación con los circuitos electrónicos hechos con componentes discretos como son, costo muy bajo, consumo reducido de potencia, alta confiabilidad contra fallas, velocidad de operación mayor y reducción de número de conexiones.

Conforme ha mejorado la tecnología de los CI, el número de compuertas que pueden colocarse dentro de una sola pastilla de silicio ha aumentado en forma considerable. La diferenciación entre los CI que tienen una cuantas compuertas internas y los que tienen decenas o cientos de compuertas, se hace por una referencia acostumbrada de que un paquete es un dispositivo de pequeña, mediana o gran escala de integración. Los fabricantes dividen comúnmente a los circuitos integrados en grupos, basados en la complejidad de los circuitos:

- VLSI (integración a muy grande escala)

Un concepto bajo el cual la función de un sistema completo se fabrica como un solo microcircuito. En este contexto un sistema, ya sea digital o lineal, contiene 1000 o más compuertas, o circuitos de similar complejidad.

- LSI (integración a gran escala)

Un concepto bajo el cual un subsistema mayor o las funciones de un sistema se fabrican como un solo microcircuito. En este contexto, un sistema mayor o subsistema, ya sea digital o lineal, es aquel que contiene 100 o más compuertas equivalentes o circuitos de complejidad similar

11. CIRCUITOS COMBINACIONALES

- **MSI (integración a mediana escala)**
Un concepto bajo el cual un subsistema completo o las funciones de un sistema se fabrican como un solo microcircuito. El subsistema o sistema son menores que los considerados para LSI, pero ya sea digital o lineal contienen de 10 a 100 compuertas equivalentes, o circuitos de complejidad similar
- **SSI (integración a pequeña escala)**
Circuitos integrados de menor complejidad que los de mediana escala. Son varias compuertas lógicas en un solo paquete.

Los circuitos integrados se clasifican en dos categorías generales, lineales y digitales. Los CI lineales operan con señales continuas para proporcionar funciones electrónicas como amplificadores y comparadores de voltaje. Los circuitos integrados digitales operan con señales binarias y están hechos de compuertas digitales interconectadas. En esta investigación nos enfocaremos exclusivamente a circuitos integrados digitales.

En la literatura de los fabricantes de CI aparecen algunos términos que ayudan a comparar los CI. Por ejemplo, para definir un 0 lógico (BAJO) o un 1 lógico (ALTO) en un CI inversor (7404), los fabricantes especifican que para una operación adecuada, una entrada BAJO, debe tomar valores entre 0 y 0.8 voltios. De la misma manera, una entrada ALTO, debe encontrarse entre 2.0 y 5.0 voltios. Entre 0.8 y 2.0 voltios se denomina "región prohibida" y aplicando un voltaje así al CI arrojaría un resultado impredecible.

Una salida BAJO del mismo circuito sería normalmente 0.1 voltios, pero puede llegar a 0.4V. Una salida ALTO normal sería de 3.5 V, pero puede llegar a ser tan baja como 2.4 V. La salida depende del valor de la resistencia de carga en la salida.

Un circuito combinacional consta de tres elementos básicos:

- Variables de entrada (a, b, \dots, n)
- Variables de salida (f_a, f_b, \dots, f_n)
- Circuito interno formado por compuertas lógicas

11. CIRCUITOS COMBINACIONALES

A partir de las compuertas lógicas básicas se construyen circuitos capaces de resolver otras funciones lógicas más complejas.

Los circuitos lógicos para sistemas digitales pueden ser combinacionales o secuenciales.

Un circuito combinacional consta de compuertas lógicas cuyas salidas en cualquier momento están determinadas en forma directa por la combinación presente de las entradas sin tomar en cuenta las entradas previas. Un circuito combinacional realiza una operación específica de procesamiento de información, especificada por completo en forma lógica por un conjunto de funciones booleanas. Los circuitos secuenciales emplean elementos de memoria además de las compuertas lógicas. Sus salidas son función de las entradas y el estado de los elementos de memoria. El estado de los elementos de memoria, a su vez, es una función de las entradas previas.

Los Circuitos combinacionales suelen presentarse ya integradas en circuitos comerciales. De ellos los más importantes son:

- Codificadores
- Decodificadores
- Multiplexores
- Demultiplexores
- Conversores
- Operadores aritméticos
- Comparadores

Un circuito combinacional resuelve una función lógica a partir de sus entradas, de tal forma que sus salidas toman valores que sólo dependen del valor de las entradas y de su constitución interna (fig. 11-1). Es decir, para una misma combinación de valores en sus entradas se tendrán siempre los mismos valores en sus salidas.

Un circuito combinacional consta de tres elementos básicos:

- Variables de entrada (a, b, \dots, n) .
- Variables de salida (f_0, f_1, \dots, f_m) .
- Circuito interno formado por compuertas lógicas.

función inversa al codificador, es decir, convierte una entrada binaria en una salida decimal.

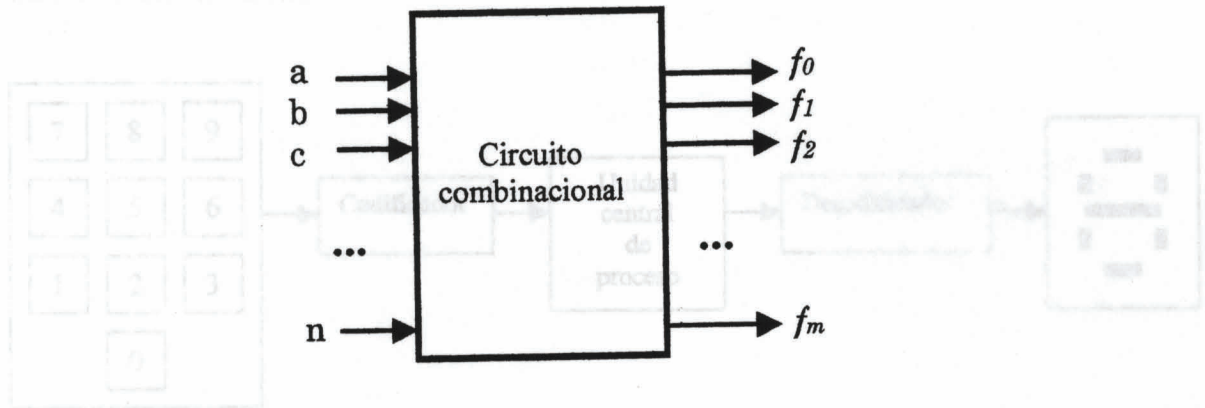


Fig. 11-1 Esquema general de un circuito combinacional

Cada salida del circuito f_i puede depender de todas o parte de las entradas según una función que se puede representar como:

$$F_i = F(a, b, \dots, n)$$

11.1 Codificadores y decodificadores

Un codificador es un circuito combinacional con 2^n entradas y n salidas, cuya misión es presentar en la salida el código binario correspondiente a la entrada activada. Esencialmente, los codificadores traducen una entrada decimal a un número binario.

Existen dos tipos fundamentales de codificadores. Los primeros sólo admiten una entrada activada, codificando en la salida el valor binario de la misma cuando no existe ninguna activa. En los segundos puede haber más de una entrada activada, existiendo prioridad en aquella de valor decimal más alto.

Considérese el diagrama de bloque de una calculadora a mano, como el de la figura 11.1-1, El sistema de entrada a la izquierda es el conjunto de teclas. Entre este conjunto y la unidad del procesador central (CPU) de la calculadora existe un codificador que traduce el número decimal de la tecla oprimida a un código binario. El CPU ejecuta la operación en binario y produce el resultado en código binario. El decodificador traduce del código binario de la CPU a un código especial que ilumina los segmentos en la pantalla. El decodificador realiza la

función inversa al codificador, es decir, convierte una entrada binaria en una salida decimal.

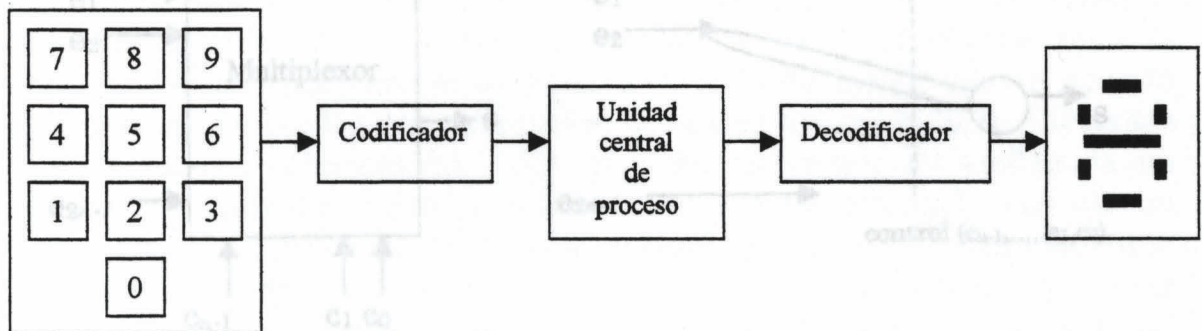


Fig. 11.1-1

11.2 Multiplexores

La multiplexión significa transmitir un gran número de unidades de información sobre un número más pequeño de canales o líneas. Un multiplexor digital es un circuito combinatorial que selecciona la información binaria de una de muchas líneas de entrada y la dirige a una sola línea de salida. En forma normal, hay 2^n líneas de entrada y n líneas de selección cuyas combinaciones bit determinan cuál entrada se selecciona. Como en los decodificadores los multiplexores pueden tener una entrada de habilitación para controlar la operación de la unidad. Cuando la entrada de habilitación se encuentra en un estado binario dado, las salidas están inhabilitadas y cuando está en el estado opuesto el circuito funciona como multiplexor normal (la entrada de habilitación a veces se denomina estroboscopio). En algunos casos dos o más multiplexores se encapsulan en un paquete CI.

El multiplexor es una función MSI muy útil y tiene muchas aplicaciones. Se usa para conectar dos o más fuentes a un solo destino entre unidades computadoras y es útil para construir un sistema de bus común. El circuito actúa como un conmutador múltiple controlado por las entradas. El esquema general de un multiplexor es el representado en la figura 11.2-1

11.4 Conversores de código

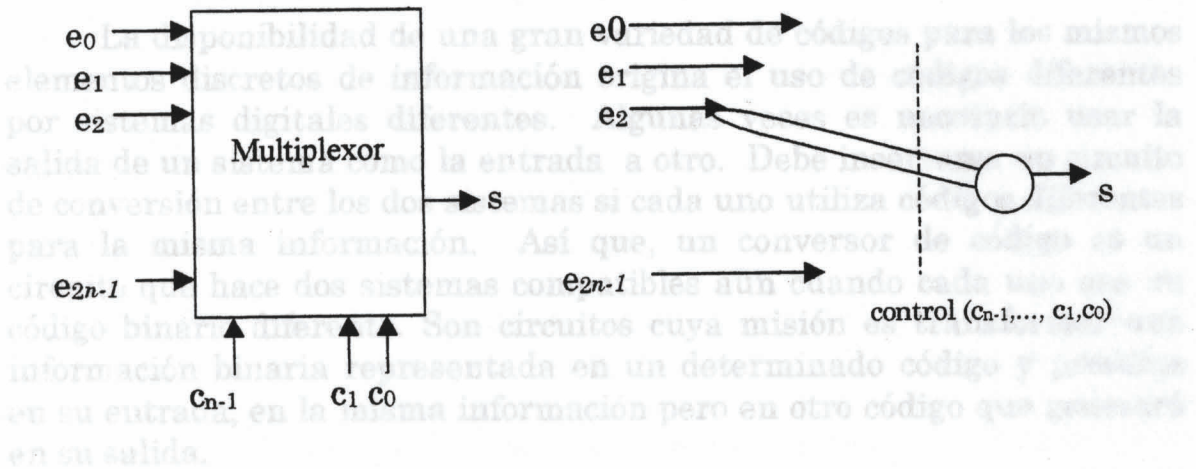


Fig. 11.2-1 Esquema general de un multiplexor

11.3 Demultiplexores

El demultiplexor ejecuta la función contraria de los multiplexores. Es un circuito que recibe información en una sola línea y trasmite esta información en una de 2^n líneas de salida posible. La selección de una línea específica de salida está controlada por los valores bit de n líneas de selección, es decir, tienen una entrada cuyo dato presentan en una de sus salidas, que será seleccionada por las entradas de control, las salidas no seleccionadas estarán en 0. También se le llama selector de datos. El esquema general de un demultiplexor está representado en la figura 11.3-1.

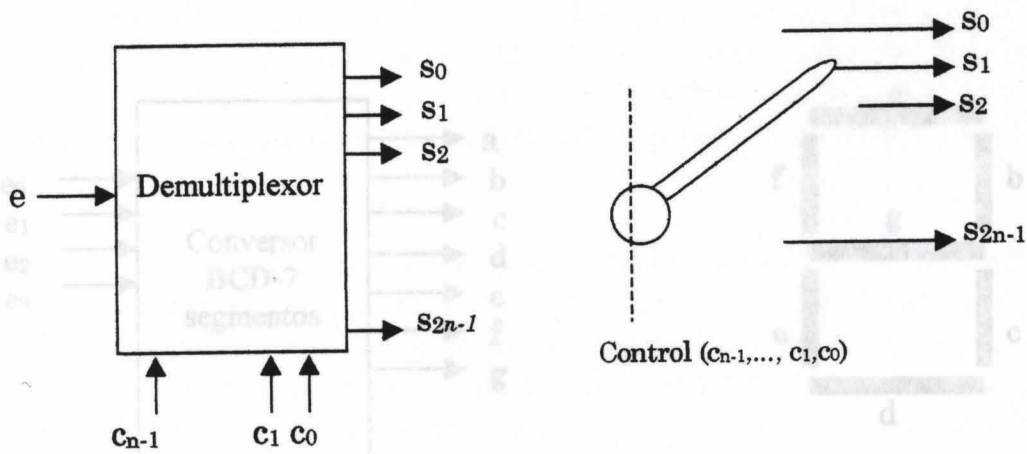


Fig. 11.3-1 Esquema general de un demultiplexor

Figura 11.4-1 Conversor de BCD natural a 7 segmentos

11.4 Conversores de código

La disponibilidad de una gran variedad de códigos para los mismos elementos discretos de información origina el uso de códigos diferentes por sistemas digitales diferentes. Algunas veces es necesario usar la salida de un sistema como la entrada a otro. Debe insertarse un circuito de conversión entre los dos sistemas si cada uno utiliza códigos diferentes para la misma información. Así que, un conversor de código es un circuito que hace dos sistemas compatibles aun cuando cada uno use un código binario diferente. Son circuitos cuya misión es transformar una información binaria representada en un determinado código y presente en su entrada, en la misma información pero en otro código que generará en su salida.

Para convertir un código binario A en el código binario B, las líneas de entrada deben suministrar la combinación bit de elementos como los especifica el código A y las líneas de salida deben generar la combinación bit correspondiente del código B. Un circuito combinacional lleva a cabo esta transformación mediante compuertas lógicas.

Una forma sencilla de diseñar un conversor de código es la de acoplar un decodificador (del código de entrada) a un codificador (al código de salida), auxiliados de unas pocas puertas lógicas sencillas.

Un circuito conversor muy utilizado en dispositivos con displays como elemento de representación de datos es el conversor de BCD natural a 7 segmentos, como se ilustra en la figura 11.4-1

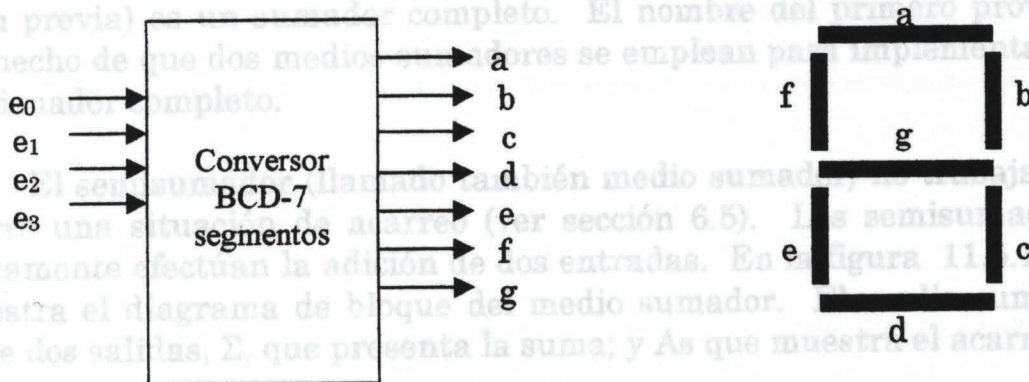


Figura 11.4-1 Conversor de BCD natural a 7 segmentos

11.5 Circuitos que realizan operaciones aritméticas

Son circuitos combinacionales que realizan operaciones sencillas de suma y resta con un determinado número de bits en algún método de representación.

11.5.1 Sumador

Las computadoras digitales realizan una variedad de tareas de procesamiento de información. Entre las funciones básicas encontradas están las diversas operaciones aritméticas. Sin duda la operación aritmética más básica es la adición de dos dígitos binarios. Esta adición siempre consta de cuatro operaciones elementales posibles:

- $0 + 0 = 0$
- $0 + 1 = 1$
- $1 + 0 = 1$
- $1 + 1 = 10$

Las tres primeras operaciones producen una suma cuya longitud es un dígito, pero cuando los bits sumando y adenando son iguales a 1, la suma binaria consta de dos dígitos. El bit significativo más alto de este resultado se denomina acarreo. Cuando los números sumando y adenando contienen más dígitos significativos, la cuanta que se lleva obtenida por la adición de dos bits se añade al siguiente par de orden más alto de bits significativos. Un circuito combinacional que lleva a cabo la adición de dos bits se denomina medio sumador. Uno que lleva a cabo la adición de tres bits (dos bits significativos y una cuenta que se lleva previa) es un sumador completo. El nombre del primero proviene del hecho de que dos medios sumadores se emplean para implementar un adicionador completo.

El semisumador (llamado también medio sumador) no trabajará si ocurre una situación de acarreo (ver sección 6.5). Los semisumadores únicamente efectúan la adición de dos entradas. En la figura 11.5.1-1 se muestra el diagrama de bloque del medio sumador. El medio sumador tiene dos salidas, Σ , que presenta la suma; y A_s que muestra el acarreo.

El semisumador es un elemento básico en la construcción de otros circuitos de adición.

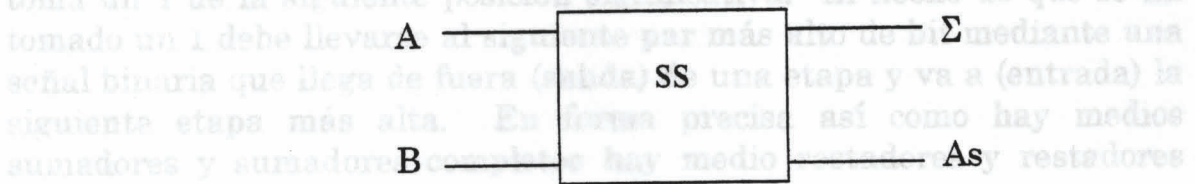


Figura 11.5.1-1 Símbolo de bloque de un semisumador

11.5 Comparadores

El sumador completo o FA (Full Adder) es un dispositivo que sirve para sumar tres cifras binarias. Tiene tres entradas para la adición. Las entradas son A, B y Ae (acarreo de entrada). Las salidas son Σ (suma) y as (acarreo de salida). En la figura 11.5.1-2 se muestra el circuito sumador completo.

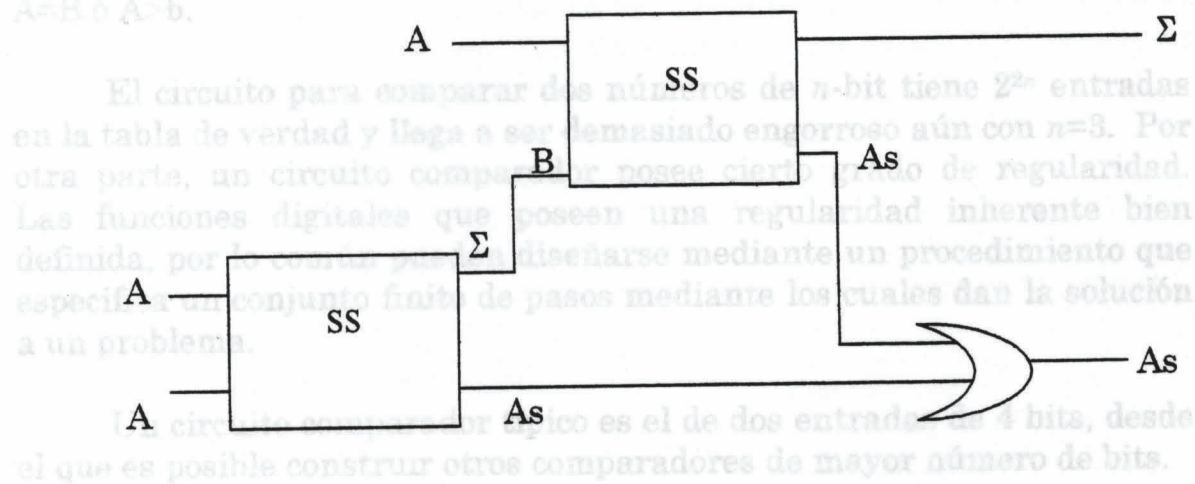


Figura 11.5.1-2 Circuito a partir de semisumadores y compuerta OR

11.5.2 Restadores

La sustracción de dos números binarios puede llevarse a cabo tomando el complemento del sustraendo y agregándolo al minuendo. Por este método, la operación de sustracción llega a ser una operación de división que requiere sumadores completos para su implementación en máquina. Es posible implementar la sustracción con circuitos lógicos en

una forma directa. Por este método cada bit sustraendo de del número se sustrae de su bit minuendo correspondiente significativo para formar un bit de diferencia. Si el bit minuendo es menos que el bit sustraendo se toma un 1 de la siguiente posición significativa. El hecho de que se ha tomado un 1 debe llevarse al siguiente par más alto de bit mediante una señal binaria que llega de fuera (salida) de una etapa y va a (entrada) la siguiente etapa más alta. En forma precisa así como hay medios sumadores y sumadores completos hay medio restadores y restadores completos.

11.6 Comparadores

La comparación de dos números es una operación que determina si un número es mayor que, menor que o igual a otro número. Un comparador de magnitud es un circuito combinacional que compara dos números, A y B y determina sus magnitudes relativas. La salida de la comparación se especifica por tres variables binarias que indican si $A < B$, $A = B$ ó $A > B$.

El circuito para comparar dos números de n -bit tiene 2^{2n} entradas en la tabla de verdad y llega a ser demasiado engorroso aún con $n=3$. Por otra parte, un circuito comparador posee cierto grado de regularidad. Las funciones digitales que poseen una regularidad inherente bien definida, por lo común pueden diseñarse mediante un procedimiento que especifica un conjunto finito de pasos mediante los cuales dan la solución a un problema.

Un circuito comparador típico es el de dos entradas de 4 bits, desde el que es posible construir otros comparadores de mayor número de bits.

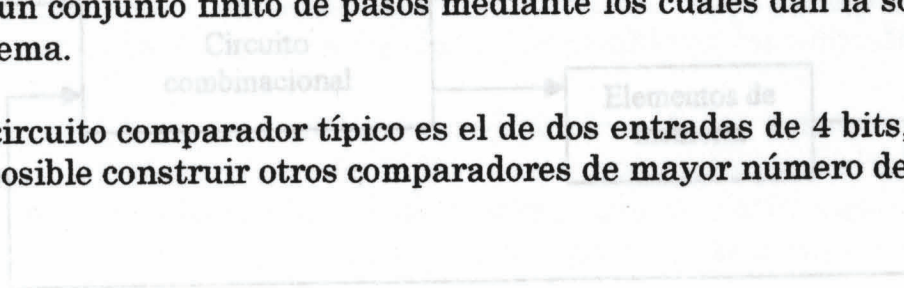


Figura 12.1

El diagrama de bloque muestra que las salidas externas en un circuito secuencial son funciones no sólo de las entradas externas, sino también del estado presente de los elementos de memoria. El siguiente estado de los elementos de memoria también es una función de las

12. SISTEMAS SECUENCIALES

Los circuitos digitales que hasta ahora se han considerado han sido combinacionales, esto es, las salidas en cualquier momento dependen por completo de las entradas presentes en ese tiempo. Aunque cualquier sistema digital es susceptible de tener circuitos combinacionales, la mayoría de los sistemas que se encuentran en la práctica también incluyen elementos de memoria, los cuales requieren que el sistema describa en términos de lógica secuencial.

Los circuitos secuenciales pueden ser de dos tipos:

• **Sincronizado** Un diagrama de bloques de un circuito secuencial se muestra en la figura 12.1. Consta de un circuito combinacional al que se conectan elementos de memoria para formar una trayectoria de retroalimentación. Los elementos de memoria son dispositivos capaces de almacenar dentro de ellos información binaria. La información binaria almacenada en los elementos de memoria en cualquier momento dado define el estado del circuito secuencial. El circuito secuencial recibe información binaria de entradas externas. Estas entradas, junto con el estado presente de los elementos de memoria, determinan el valor binario en las terminales de salida. También determinan las condiciones para cambiar el estado en los elementos de memoria.

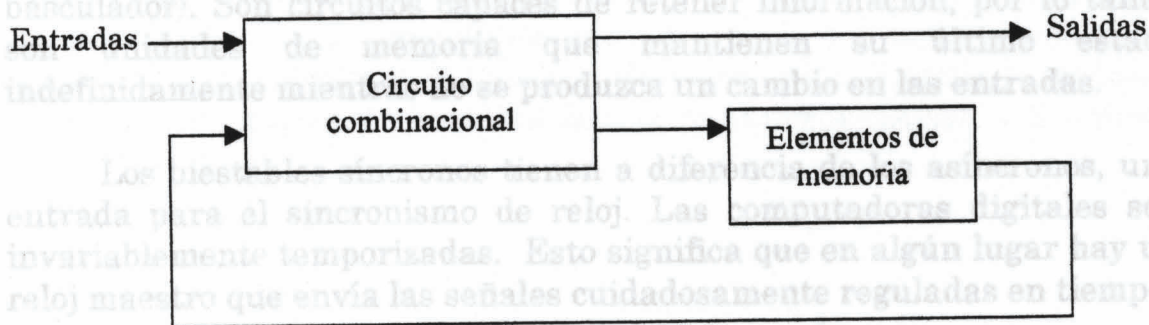


Figura 12.1

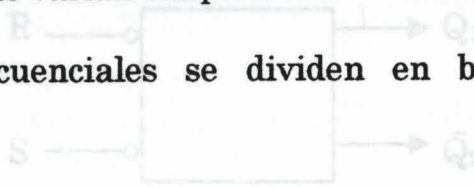
El diagrama de bloque muestra que las salidas externas en un circuito secuencial son funciones no sólo de las entradas externas, sino también del estado presente de los elementos de memoria. El siguiente estado de los elementos de memoria también es una función de las

entradas externas y del estado presente. Por lo tanto, un circuito secuencial está especificado por una secuencia de tiempo de entradas, salidas y estados internos.

La unidad de memoria está formada por un conjunto de elementos denominados biestables o flop-flops, con capacidad de permanecer indefinidamente en un determinado estado lógico (0 o 1) en ausencia de estímulos exteriores.

- Los circuitos secuenciales pueden ser de dos tipos:
- Síncronos. Son aquellos que para realizar la transmisión de datos a través del propio circuito necesitan una señal de control o impulsos de reloj (clock) que además configuran una más de sus entradas.
 - Asíncronos. Son aquellos que no necesitan señal de sincronismo, por lo tanto sus salidas varían simplemente al hacerlo sus entradas.

Los circuitos secuenciales se dividen en biestables, registros y contadores.



12.1 Biestables o flip-flops

Figura 12.1.1-1 Símbolo lógico para el biestable RS

El término biestable o flip-flop se utilizará indistintamente en esta sección y las precedentes (en algunos textos se utiliza el término basculador). Son circuitos capaces de retener información; por lo tanto, son unidades de memoria que mantienen su último estado indefinidamente mientras no se produzca un cambio en las entradas.

Los biestables síncronos tienen a diferencia de los asíncronos, una entrada para el sincronismo de reloj. Las computadoras digitales son invariablemente temporizadas. Esto significa que en algún lugar hay un reloj maestro que envía las señales cuidadosamente reguladas en tiempo.

El reloj es el motor de la computadora y como tal, mide cuidadosamente el tiempo y envía señales regularmente espaciadas que hacen que las cosas ocurran. Las señales de iniciación son llamadas pulsos de reloj.

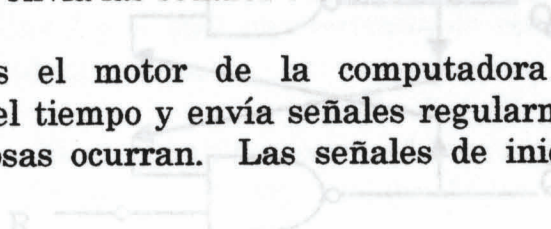


Figura 12.1.1-2 Biestable RS utilizando compuertas NAND

12.1.1 El biestable RS

S	Q	\bar{Q}
0	0	1
1	1	0

Al biestable básico se le llama RS. En la figura 12.1.1-1 se muestra el símbolo lógico del bloque para un basculador RS. Tiene dos entradas a la izquierda, cuyas etiquetas son R (reset) y S (set). El símbolo de este basculador tiene entradas activas BAJO que se indican con los círculos en las entradas S y R. A diferencia de las compuertas lógicas, los biestables tienen dos salidas complementarias, que comúnmente se le llama Q y \bar{Q} (no Q). Q es la salida "normal", la más usual y la otra (\bar{Q}) que es simplemente el complemento de la salida Q y se le llama salida complementaria. En condiciones normales, estas salidas siempre son complementarias, de aquí que si $Q=1$, entonces $\bar{Q}=0$, o bien si $Q=0$ entonces $\bar{Q}=1$.

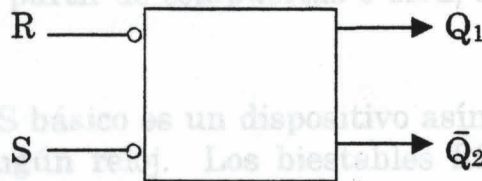


Figura 12.1.1-1 Símbolo lógico para el biestable RS

El biestable RS puede construirse a partir de compuertas lógicas, el de la figura 12.1.1-2 está conectado a partir de dos compuertas NAND (AND más un inversor a la salida). Hay que notar la retroalimentación característica que va de la salida de una compuerta NAND a la entrada de la otra.

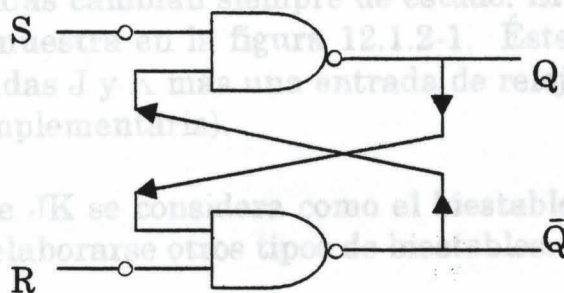


Figura 12.1.1-2 Biestable RS utilizando compuertas NAND

R	S	Q	\bar{Q}
0	0	1	1
0	1	1	0
1	0	0	1
1	1	Sin cambio	Sin cambio

Tabla de verdad del biestable RS con puertas NAND

La primera línea de la tabla de verdad se llama el estado prohibido en el que ambas salidas son 1 o ALTA. Esta condición no se usa en el biestable RS.

Los biestables RS se llaman también enganchadores RS, gracias a su función de detener los datos temporalmente. Los enganchadores RS pueden conectarse a partir de compuertas o bien, comprarse en forma de CI.

El biestable RS básico es un dispositivo asíncrono que no opera en combinación con ningún reloj. Los biestables RS síncronos operan en combinación con un reloj o dispositivo para medir el tiempo.

12.1.2 El biestable JK

El biestable JK es probablemente el más utilizado en los circuitos lógicos secuenciales, debido a su adaptabilidad.

El biestable JK asíncrono es similar al RS, con la diferencia de la eliminación del defecto producido cuando las dos entradas valen 1, en cuyo caso las salidas cambian siempre de estado. El símbolo lógico de un biestable JK se muestra en la figura 12.1.2-1. Éste tiene dos entradas de datos etiquetadas J y K más una entrada de reloj (CK) y las salidas Q (normal) y \bar{Q} (complementaria).

El biestable JK se considera como el biestable universal, a partir del cual pueden elaborarse otros tipos de biestables.

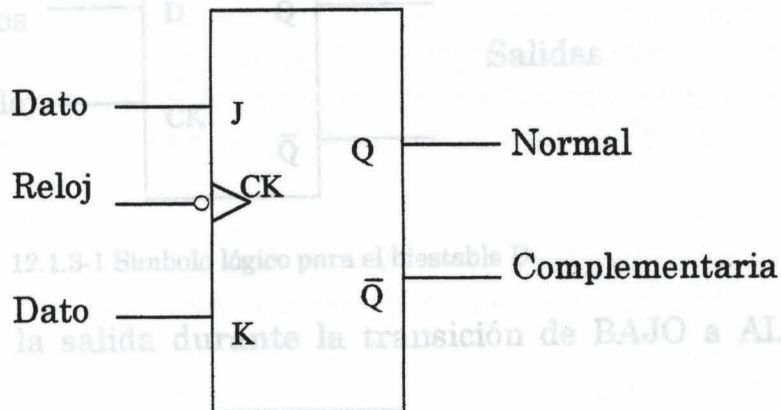


Figura 12.1.2-1 Símbolo lógico del biestable JK

Su tabla de verdad es la siguiente:

J	K	Q	\bar{Q}
0	0	Sin cambio	Sin cambio
0	1	0	1
1	0	1	0
1	1	Estado opuesto	Estado opuesto

Los biestables comerciales pueden tener entradas asíncornas, además de las normales J, K y CK. Usualmente tienen la entrada predisposición (PR), en la cual asíncornamente hace la salida $Q = 1$. Es posible que tengan también la entrada borrar (CLR) para limpiar la salida, haciendo $Q = 0$.

12.1.3 El biestable D

La figura 12.1.3-1 muestra el símbolo lógico para el biestable D. Tiene solamente una entrada de datos D y la entrada de reloj (CK). Las salidas Q y \bar{Q} se encuentran al lado derecho del símbolo. Al biestable D a menudo se le llama flip-flop de retardo o de retraso. Este nombre se debe a que cualquiera que sea la entrada en la entrada de datos (D), se retarda un pulso de reloj antes de llegar a la salida normal (Q). Los datos

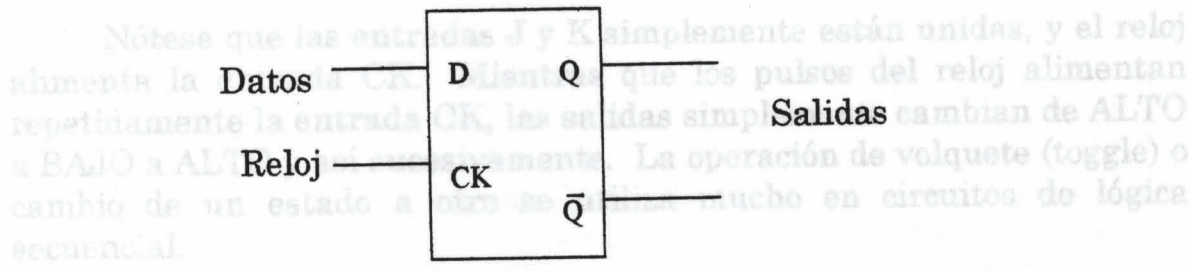


Figura 12.1.3-1 Símbolo lógico para el biestable D. Este símbolo se utiliza en circuitos contadores y registros. Los datos se transfieren a la salida durante la transición de BAJO a ALTO del pulso de reloj.

12.2 Contadores

El biestable RS síncrono puede convertirse en un basculador D añadiendo un inversor. El biestable D comercial es un dispositivo comercial llamado CI 7474.

12.1.4 El biestable T

El biestable T (toggle) síncrono tiene una entrada (T), otra entrada de sincronía (CK) y dos salidas complementarias Q y Q-bar. Su funcionamiento es muy simple, ya que cada vez que se activa su entrada se invierte el valor de sus salidas.

En la figura 12.2.3-1 puede verse un biestable T asíncrono construido a partir de un biestable JK.

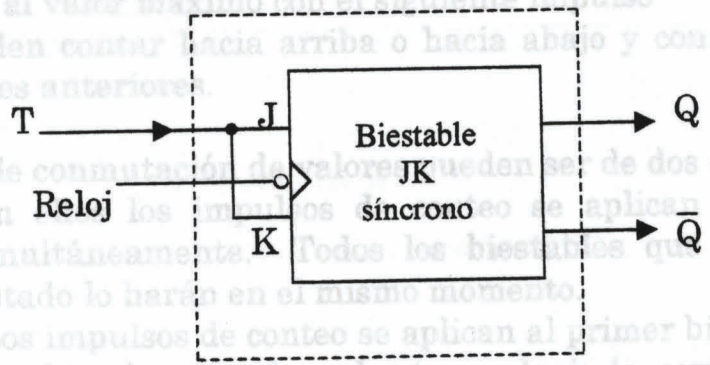


Figura 12.1.4-1 Biestable T síncrono

Un contador síncrono es mucho más rápido que uno asíncrono y mu. Nótese que las entradas J y K simplemente están unidas, y el reloj alimenta la entrada CK. Mientras que los pulsos del reloj alimentan repetidamente la entrada CK, las salidas simplemente cambian de ALTO a BAJO a ALTO y así sucesivamente. La operación de volquete (toggle) o cambio de un estado a otro se utiliza mucho en circuitos de lógica secuencial.

Estos tipos de biestables T son muy utilizados en circuitos contadores y registros, que veremos más adelante.

12.2 Contadores

Un contador es un registro que pasa a través de una secuencia determinada de estados bajo la aplicación de pulsos de entrada. Las compuertas en un contador están conectadas de tal forma que producen una secuencia preescrita de estados binarios en el registro. Aunque los contadores son un tipo especial de registro, es costumbre diferenciarlos dándoles un nombre especial. Los contadores digitales sólo cuentan en binario o en códigos binarios.

Los contadores son útiles para generar variables temporizadas para las operaciones de secuencia y control en un sistema digital. Pueden ser de tres tipos:

- Ascendentes. Cuentan hacia arriba de forma que al llegar a su valor máximo se ponen a cero con el siguiente impulso.
- Descendentes. Cuentan hacia debajo de tal forma que cuando llegan a cero se ponen al valor máximo con el siguiente impulso
- Mixtos. Pueden contar hacia arriba o hacia abajo y con las mismas consideraciones anteriores.

Según la forma de conmutación de valores pueden ser de dos clases:

- Síncronos: En ellos los impulsos de conteo se aplican a todos los biestables simultáneamente. Todos los biestables que tengan que cambiar de estado lo harán en el mismo momento.
- Asíncronos: Los impulsos de conteo se aplican al primer biestable y los demás van cambiando en cadena hasta producir la correspondiente salida.

• Líneas de entrada. Permiten introducir nuevos datos en el registro

- Un contador síncrono es mucho más rápido que uno asíncrono y mucho más complejo.

12.3 Registros

Un registro es un grupo de celdas de almacenamiento, adecuadas para mantener información binaria. Un grupo de flip-flops constituye un registro, ya que cada flip-flop es una celda binaria capaz de almacenar un bit de información. Un registro de n -bit tiene un grupo de n flip-flops y es capaz de almacenar cualquier información binaria que contenga n bits. Además de los flip-flops, un registro puede tener compuertas combinatoriales que realicen ciertas tareas de procesamiento de datos. En su definición más amplia, un registro consta de un grupo de flip-flops y compuertas que efectúan su transición. Los flip-flop mantienen información binaria y las compuertas controlan cuándo y cómo se transfiere la información nueva al registro. Dicha información deberá ser accesible desde el exterior y podrá ser modificable mediante un conjunto de operaciones sencillas. En la figura 12.3-1 se muestra la estructura de un registro básico.

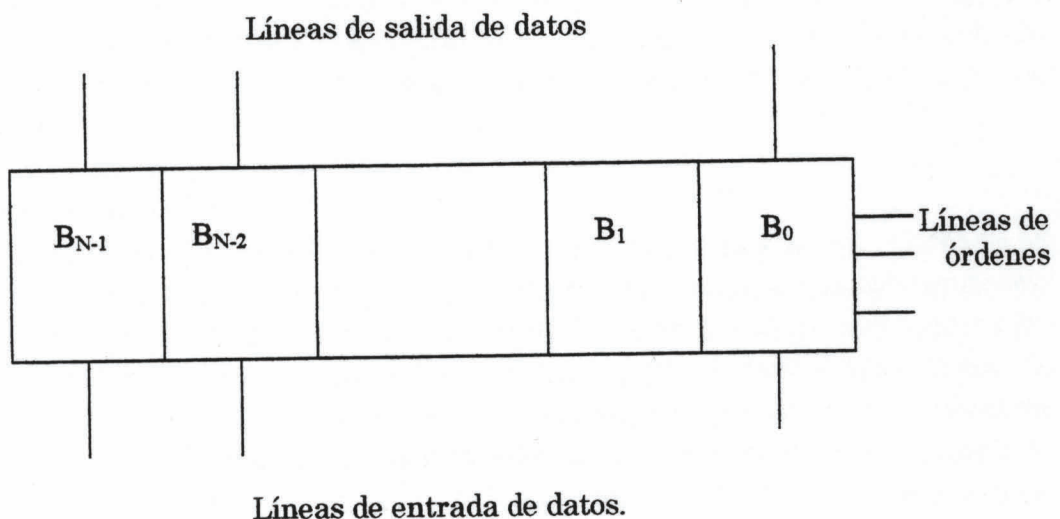


Figura 12.3-1 Estructura de un registro básico

Como vemos en la figura anterior, el registro consta de N biestables, por lo que diremos que su longitud es N . Para comunicarse con el exterior, el registro dispone de:

- Líneas de entrada: Permiten introducir nuevos datos en el registro

- **Líneas de salida:** Permiten extraer la información que contiene el registro
- **Líneas de órdenes.** Su activación permitirá realizar operaciones de lectura y escritura sobre el registro, así como la realización sobre el mismo de una serie de operaciones elementales. ya que estos son la clave en el funcionamiento de los procesadores.

13.1 Operaciones con registros

Todo el conjunto de operaciones que se realizan en el interior de una computadora se puede reducir en esencia, a secuencias de transferencia de información entre sus registros constituyentes.

Un registro es un conjunto de biestables que se consideran como una unidad lógica. Ya que se trata de biestables, el registro será capaz de almacenar información en formato binario.

• *Operación de lectura*

La operación de lectura de un registro consiste en obtener la información binaria que contiene. Para ello se hará uso de una de las líneas de órdenes (*L*). Cada vez que se active dicha línea, el registro presentará la información que contiene al exterior, de donde podrá ser tomada por quien la requiera. Esta operación no destruye el contenido del registro.

• *Operación de escritura*

La operación de escritura en un registro consiste en introducir información binaria nueva en el mismo. Se realiza presentando en las líneas de entrada de datos el valor binario a tomar por cada uno de los biestables que componen el registro y activando una línea de órdenes (*E*). El valor binario que se presenta en las líneas de entrada de los datos, actualiza el contenido de los biestables que componen el registro. Lógicamente, el valor anterior que contuviera el registro se destruye.

• *Transferencia entre registros*

Cuando se realiza una operación de transferencia entre registros se desea que el contenido de un determinado registro, denominado registro fuente, se escriba en otro registro, denominado registro destino.

13. ELEMENTOS BÁSICOS DE LAS COMPUTADORAS de 8 bits de longitud son:

El objetivo de este capítulo es presentar de una forma sencilla los elementos fundamentales que conforman una computadora. Para ello se retomará el concepto registro del capítulo anterior, ya que estos son la clave en el funcionamiento de los procesadores.

13.1 Operaciones con registros Tras realizar una operación de transferencia del registro R_A al R_B registro destino), el contenido final de los dos registros será:

Todo el conjunto de operaciones que se realizan en el interior de una computadora se puede reducir en esencia, a secuencias de transferencia de información entre sus registros constituyentes.

Un registro es un conjunto de biestables que se consideran como una unidad lógica. Ya que se trata de biestables, el registro será capaz de almacenar información en formato binario.

- **Operación de lectura** el contenido de R_A a R_B .
La operación de lectura de un registro consiste en obtener la información binaria que contiene. Para ello se hará uso de una de las líneas de órdenes (L). Cada vez que se active dicha línea, el registro presentará la información que contiene al exterior, de donde podrá ser tomada por quien la requiera. Esta operación no destruye el contenido del registro. Un determinado registro R_A de 8 bits de longitud es $(R_A)=00110011$, tras realizar una operación de incremento, su
- **Operación de escritura** 110100.
La operación de escritura en un registro consiste en introducir información binaria nueva en el mismo. Se realiza presentando en las líneas de entrada de datos el valor binario a tomar por cada uno de los biestables que componen el registro y activando una línea de órdenes (E). El valor binario que se presenta en las líneas de entrada de los datos, actualiza el contenido de los biestables que componen el registro. Lógicamente, el valor anterior que contuviera el registro se destruye. de complementación
- **Transferencia entre registros** los biestables que componen el registro.
Cuando se realiza una operación de transferencia entre registros se desea que el contenido de un determinado registro, denominado registro fuente, se escriba en otro registro, denominado registro destino. $(R_A)=00110101$.

- Por ejemplo, si los contenidos de dos registros R_A y R_B , de 8 bits de longitud son:

de puesta a cero o reset de un registro consiste en poner todos sus biestables a cero. Esta operación se realizará con la activación de una línea de órdenes (RES). Por ejemplo, si el contenido de R_A es $(R_A) = 00011011$ y el contenido de R_B es $(R_B) = 11010001$, tras realizar la operación RES resultará $(R_A) = 00000000$.

- Tras realizar una operación de transferencia del registro R_A al registro R_B (R_A registro origen, R_B registro destino), el contenido final de los dos registros será:

de uno o set de un registro, consiste en poner todos sus biestables a uno. Esta operación se realizará mediante la activación de una línea de órdenes (SET). Por ejemplo, si el contenido de R_A es $(R_A) = 00011011$ y el contenido de R_B es $(R_B) = 00011011$, la activación de SET dará como resultado $(R_A) = 11111111$.

En la notación anterior los paréntesis indican "contenido de". Así mismo la expresión:

$(R_A) \longrightarrow R_B$ se definieron anteriormente. Por supuesto que no todas ellas estarán presentes en un registro concreto, debe leerse como "llevar el contenido de R_A a R_B ".

- *Operaciones de incremento y decremento*

Cuando se realiza sobre un registro una operación de incremento, se suma una unidad al contenido del registro. Esta operación se realizará activando una línea de órdenes (INC). Por ejemplo, si el contenido de un determinado registro R_A de 8 bits de longitud es $(R_A) = 00110011$, tras realizar una operación de incremento, su contenido será $(R_A) = 00110100$.

La operación de decremento implica restar una unidad al contenido del registro. Esta operación se realizará activando una línea de órdenes (DEC). Por ejemplo, si el contenido de un registro de 8 bits es $(R_B) = 10001101$, tras realizar una operación de decremento, su contenido será $(R_B) = 10001101$.

- *Operación de complementación*

La operación de complementación o negación consiste en invertir el valor binario de cada uno de los biestables que componen el registro. La realización de esta operación se gobernará mediante la activación de una línea de órdenes (CMP). Por ejemplo, si el contenido de un registro de 8 bits es $(R_A) = 11001010$, tras realizar la complementación, será $(R_A) = 00110101$.

- **Operación de puesta a cero (reset)**

La operación de puesta a cero o reset de un registro consiste en poner todos sus biestables a cero. Esta operación se realizará con la activación de una línea de órdenes que llamaremos RES. Por ejemplo, si el contenido de R_A es $(R_A)=11110010$, tras realizar la operación RES resultará $(R_A)=0000000$.

- **Operación de puesta a uno (set)**

La operación de puesta a uno o set de un registro, consiste en poner todos sus biestables a uno. Esta operación se realizará mediante la activación de una línea de órdenes que llamaremos SET. Por ejemplo, si el contenido de R_A es $(R_A)=00110011$, la activación de SET dará como resultado $(R_A)=11111111$.

- La figura 13.1-1 muestra la estructura general de un registro con todas las líneas de órdenes que se definieron anteriormente. Por supuesto que no todas ellas estarán presentes en un registro concreto, sino sólo aquellas que sean necesarias par la función que dicho registro vaya a desempeñar.

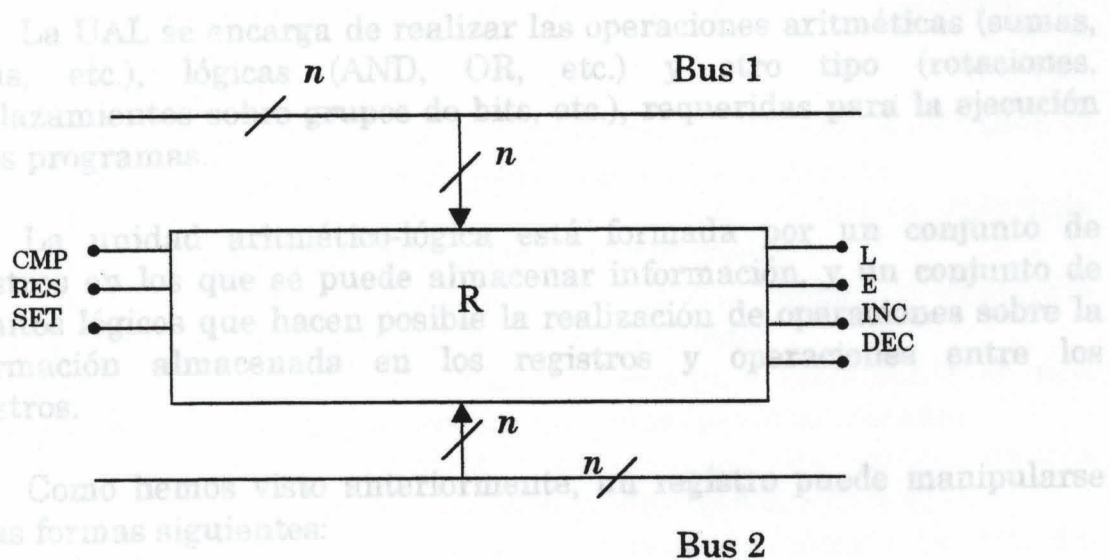


Figura 13.1-1 Estructura de un registro con todas sus líneas de órdenes definidas

1. El registro puede limpiarse o ponerse a cero.
2. El contenido del registro puede desplazarse para obtener el complemento a uno.
3. El contenido del registro puede desplazarse a la izquierda o a la derecha.
4. El contenido del registro puede incrementarse o decrementarse.

13.2 Partes fundamentales de una computadora

En esta sección está dedicada a los tres elementos presentes en cualquier arquitectura básica: la unidad aritmético-lógica (UAL), la unidad de control (C) y a la memoria principal (MP).

- La unidad de control se encarga de gobernar el funcionamiento global de la computadora, trayendo las instrucciones que están en la memoria principal, e indicando cuál es su cometido al conjunto de elementos restantes (UAL, MP, y E/S).
- La unidad aritmético-lógica es la que se encarga básicamente de realizar operaciones sobre los datos que maneja un programa.
- La memoria principal se encarga de almacenar las instrucciones que componen un programa y los datos que maneja.

13.2.1 La unidad aritmético-lógica (UAL)

La UAL se encarga de realizar las operaciones aritméticas (sumas, restas, etc.), lógicas (AND, OR, etc.) y otro tipo (rotaciones, desplazamientos sobre grupos de bits, etc.), requeridas para la ejecución de los programas.

La unidad aritmético-lógica está formada por un conjunto de registros en los que se puede almacenar información, y un conjunto de circuitos lógicos que hacen posible la realización de operaciones sobre la información almacenada en los registros y operaciones entre los registros.

Como hemos visto anteriormente, un registro puede manipularse de las formas siguientes:

1. El registro puede limpiarse o ponerse a cero.
2. El contenido del registro puede complementarse para obtener el complemento a 1 o a 2 en binario.
3. El contenido del registro puede desplazarse a la izquierda o a la derecha.
4. El contenido del registro puede incrementarse o decrementarse.

Y anteriormente se han descrito también algunas las operaciones entre registros. Estas incluyen:

1. Transferir el contenido de un registro a otro.
2. Sumar o restar el contenido de un registro al contenido de otro.

La mayoría de las operaciones que realiza una UAL consiste en éstos, o en una secuencia de conjuntos de estos dos tipos de operaciones. Instrucciones complicadas como la multiplicación y la división requieren de un gran número de estas operaciones, pero estas operaciones se realizan utilizando secuencias de operaciones simples.

En la figura 13.2.1-1 se muestra la estructura de una UAL básica. En dicha figura podemos ver los elementos que la constituyen. La UAL se comunica con el exterior mediante un bus denominado bus de datos, compuesto por nt líneas.

- **Circuito combinacional:** Es el que se encarga de realizar propiamente las operaciones con los datos. Este circuito tiene unas entradas de órdenes para indicarle el tipo de operación que desea realizar con los datos de entrada. A continuación enumeramos las órdenes más típicas:

SUM: Cuando se activa, el circuito combinacional presenta a su salida la suma aritmética de las cantidades presentes en sus entradas.

RES: Cuando se activa, el circuito combinacional realiza la diferencia aritmética de las cantidades presentes en sus entradas.

MUL: cuando se activa, el circuito combinacional realiza la diferencia aritmética de las cantidades presentes a sus entradas.

INC : Cuando se activa, el circuito combinacional incrementa en una unidad el contenido de una de sus dos entradas (predeterminada).

DEC: Cuando se activa, el circuito combinacional decrementa en una unidad el contenido de una de sus dos entradas (predeterminada).

NOT: Cuando se activa, el circuito combinacional invierte los bits del número binario presente en una de sus dos entradas.

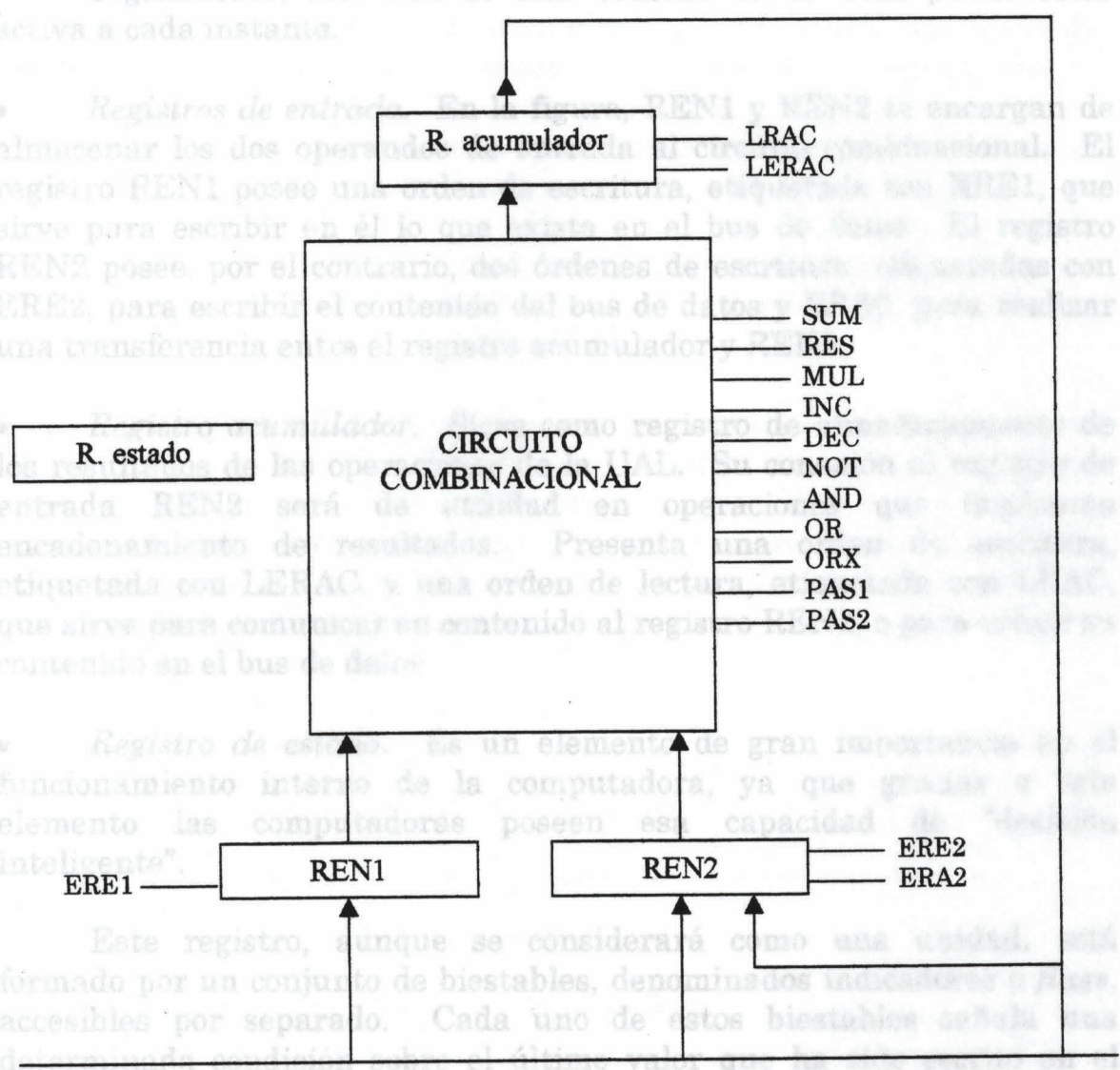
AND: Cuando se activa, el circuito combinacional presenta a su salida el producto lógico de los números binarios presentes en sus entradas.

OR: Cuando se activa, el circuito combinacional presenta a su salida la suma lógica de los números binarios presentes en sus entradas.

PAS2: Cuando se activa, el circuito combinacional presenta a su salida el contenido del registro REN2.

Logicamente, sólo una de esas órdenes en la UAL puede estar activa a cada instante.

Registros de entrada. En el sistema REN1 y REN2 se encargan de almacenar los dos operandos de la operación combinacional. El registro REN1 posee una orden de escritura, etiquetada con ERE1, que sirve para escribir en él lo que exista en el bus de datos. El registro REN2 posee, por el contrario, dos órdenes de escritura, etiquetadas con ERE2, para escribir el contenido del bus de datos en el registro. Una transferencia entre el registro acumulador y el bus de datos se realiza a través de un controlador de estado, etiquetado con R. estado, que indica al acumulador que debe leer o escribir en el bus de datos.



Registro de estado. Este elemento de gran importancia en el funcionamiento interno de la computadora, ya que gracias a este elemento las computadoras poseen esta capacidad de "pensar inteligente".

Este registro, aunque se considerará como una unidad, está formado por un conjunto de biestables, denominados indicadores y pas, accesibles por separado. Cada uno de estos biestables recibe una determinada condición sobre el último valor que ha almacenado el acumulador. Los indicadores que se encuentran más frecuentemente en este registro son:

Z: Biestable indicador de cero. Adquiere el valor uno cuando el último número binario es cero.

Figura 13.2.1-1 Estructura de una UAL básica

- ORX:** Cuando se activa, el circuito combinacional realiza la suma lógica exclusiva de sus dos entradas y la presenta en su salida.
- PAS1:** Cuando se activa, el circuito combinacional presenta a su salida el contenido del registro REN1.
- PAS2:** Cuando se activa, el circuito combinacional presenta a su salida el contenido del registro REN2.

O. Bien Lógicamente, sólo una de esas órdenes en la UAL puede estar activa a cada instante.

- **Registros de entrada.** En la figura, REN1 y REN2 se encargan de almacenar los dos operandos de entrada al circuito combinacional. El registro REN1 posee una orden de escritura, etiquetada con ERE1, que sirve para escribir en él lo que exista en el bus de datos. El registro REN2 posee, por el contrario, dos órdenes de escritura, etiquetadas con ERE2, para escribir el contenido del bus de datos y ERA2, para realizar una transferencia entre el registro acumulador y REN2.

- **Registro acumulador.** Sirve como registro de almacenamiento de los resultados de las operaciones de la UAL. Su conexión al registro de entrada REN2 será de utilidad en operaciones que impliquen encadenamiento de resultados. Presenta una orden de escritura, etiquetada con LERAC, y una orden de lectura, etiquetada con LRAC, que sirve para comunicar su contenido al registro REN2, o para volcar su contenido en el bus de datos.

- **Registro de estado.** Es un elemento de gran importancia en el funcionamiento interno de la computadora, ya que gracias a este elemento las computadoras poseen esa capacidad de "decisión inteligente".

Este registro, aunque se considerará como una unidad, está formado por un conjunto de biestables, denominados indicadores o *flags*, accesibles por separado. Cada uno de estos biestables señala una determinada condición sobre el último valor que ha sido escrito en el acumulador. Los indicadores que se encuentran más frecuentemente en este registro son:

Z: Biestable indicador de cero. Adquiere el valor uno cuando el último número binario escrito en el acumulador es cero.

N: Biestable indicador de negativo: Adquiere el valor uno cuando el último número binario escrito en el acumulador es negativo.

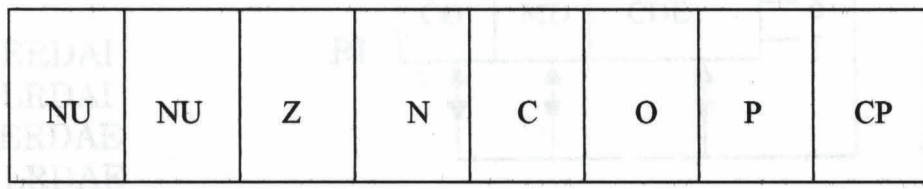
C: Biestable indicador de acarreo. Adquiere el valor uno cuando se escribe en el acumulador un número procedente de una operación de la UAL que generó acarreo en el último bit.

O: Biestable indicador de desbordamiento: Adquiere el valor uno cuando se escribe en el acumulador un número procedente de una operación de la UAL no representable con el número de bits asignado al acumulador

P: Biestable indicador de paridad. Adquiere el valor uno cuando el número de unos de la cantidad binaria escrita en el acumulador es par.

CP: Biestable de acarreo parcial. Adquiere el valor uno cuando se escribe en el acumulador el resultado de una operación de la UAL que produjo acarreo en el cuarto y el quinto bit.

En la figura 13.2.1-2 se muestra la estructura del registro de indicadores. La actualización de este registro se realiza de forma automática, siguiendo las operaciones de la UAL, y de forma transparente al usuario. Sin embargo, suele ser común que éste pueda poner a cero (reset) o a uno (set) algunos o todos los biestables de dicho registro. Depende de cada arquitectura cuáles de ellos en concreto.



UN = No utilizado

Figura 13.2.1-2 Registro de indicadores típico

13.2.2 La unidad de control (UC)

La unidad de control puede calificarse como el "director" de la computadora, ya que es la parte que gobierna totalmente su funcionamiento.

La unidad de control se encarga de traer de la memoria las instrucciones que componen un programa, de interpretarlas y de generar el conjunto de órdenes que gobiernan a todos los elementos de la computadora, en la secuencia adecuada para que la instrucción se ejecute de modo correcto. En la figura 13.2.2-1 se muestra la estructura de una unidad de control típica.

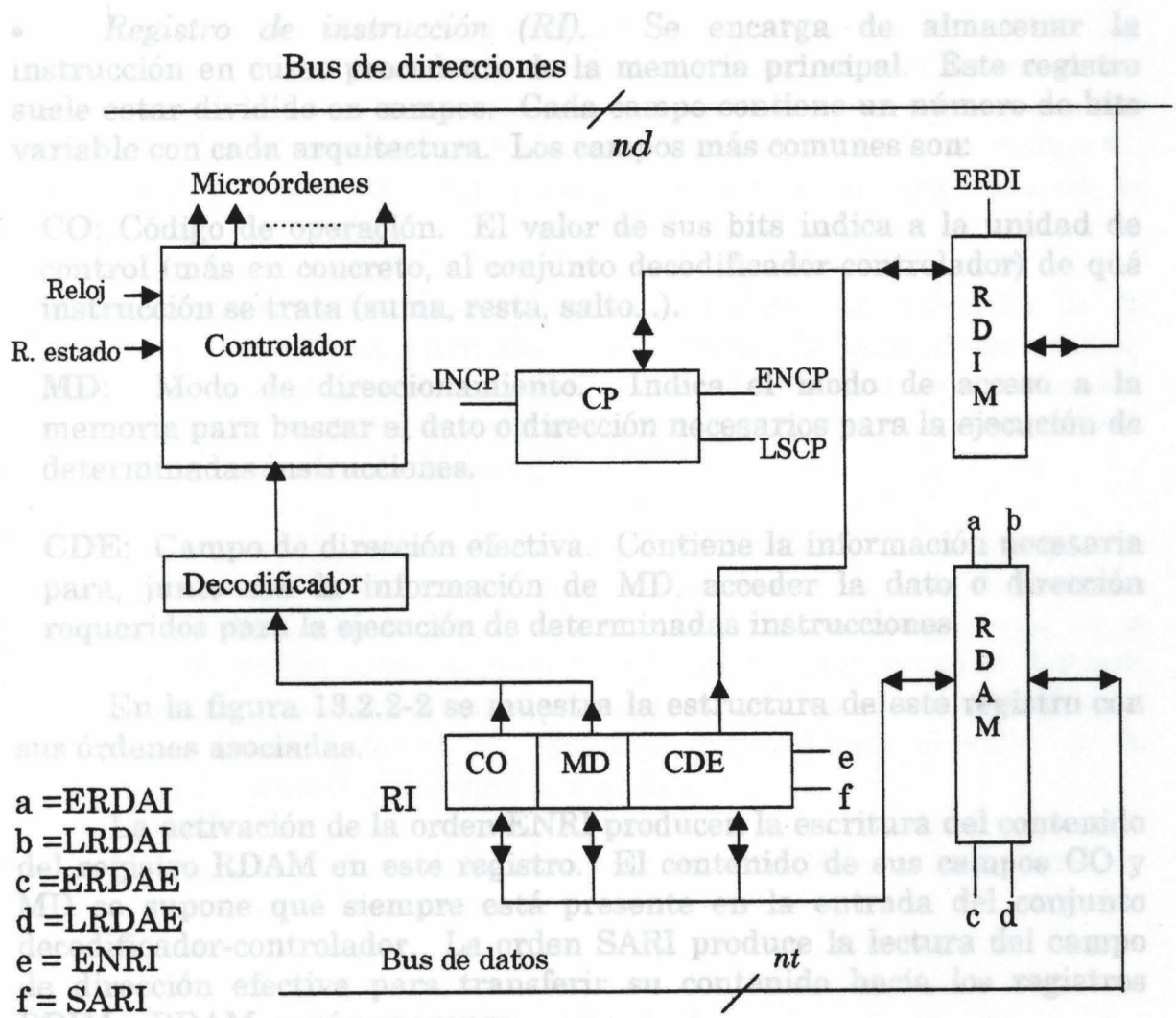


Figura 13.2.2-1 Estructura de una unidad de control típica

Sus conexiones al exterior, con el resto de los elementos que componen la computadora se realizan en este caso a través de dos buses: el bus de datos, que ya encontramos en la UAL, y un nuevo bus, denominado bus de direcciones, compuesto por nd líneas, cuyo contenido fundamental es transmitir a la memoria principal (MP) a través del registro RDIM las direcciones que se desea leer o en las que se desea escribir. La unidad de control poseerá sus propios buses de datos y de direcciones que denominaremos buses de direcciones y de datos internos.

A continuación se presentan las partes fundamentales de una unidad de control típica:

Figure 13.2.2-2 Estructura del registro de instrucción

- **Registro de instrucción (RI).** Se encarga de almacenar la instrucción en curso procedente de la memoria principal. Este registro suele estar dividido en campos. Cada campo contiene un número de bits variable con cada arquitectura. Los campos más comunes son:

CO: Código de operación. El valor de sus bits indica a la unidad de control (más en concreto, al conjunto decodificador-controlador) de qué instrucción se trata (suma, resta, salto...).

MD: Modo de direccionamiento. Indica el modo de acceso a la memoria para buscar el dato o dirección necesarios para la ejecución de determinadas instrucciones.

CDE: Campo de dirección efectiva. Contiene la información necesaria para, junto con la información de MD, acceder la dato o dirección requeridos para la ejecución de determinadas instrucciones.

En la figura 13.2.2-2 se muestra la estructura de este registro con sus órdenes asociadas.

La activación de la orden ENRI producen la escritura del contenido del registro RDAM en este registro. El contenido de sus campos CO y MD se supone que siempre está presente en la entrada del conjunto decodificador-controlador. La orden SARI produce la lectura del campo de dirección efectiva para transferir su contenido hacia los registros RDIM o RDAM, según convenga.

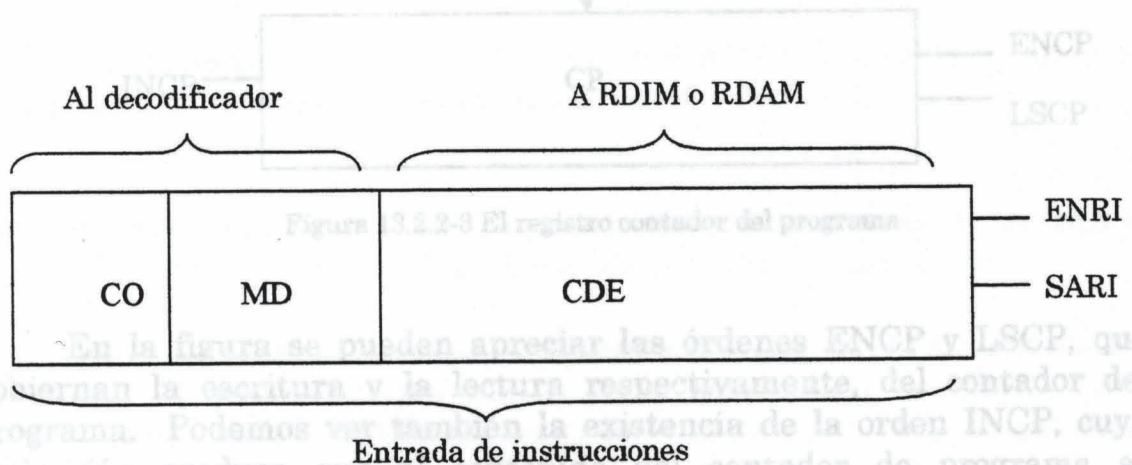


Figura 13.2.2-2 Estructura del registro de instrucción

• **Registro contador del programa (CP).** La ejecución de los programas almacenados en memoria se realiza primordialmente en orden secuencial. Se ejecuta una instrucción tras otra, según su orden consecutivo en la memoria principal. No obstante, cuando se realiza un salto se pasa a ejecutar una instrucción escrita en otra zona de la memoria.

El registro contador de programa contiene la dirección de la siguiente instrucción a ejecutarse. Por tanto, la unidad de control, cuando busque la instrucción siguiente, lo hará utilizando la información de la posición de memoria en la que se encuentra, que es precisamente el contenido del contador del programa.

Al finalizar la ejecución de una instrucción, lo primero que se deberá hacer será colocar en el contador del programa la posición de memoria en donde se encuentra la siguiente instrucción que se va a ejecutar. Para ello, si no se trata de una instrucción de salto, bastará incrementar adecuadamente el contador de programa. Si se trata de una instrucción de salto habrá que escribir directamente el valor de la posición de la siguiente instrucción a ejecutar.

En la figura 13.2.2-3 Se muestran la estructura y conexiones de registro contador de programa.

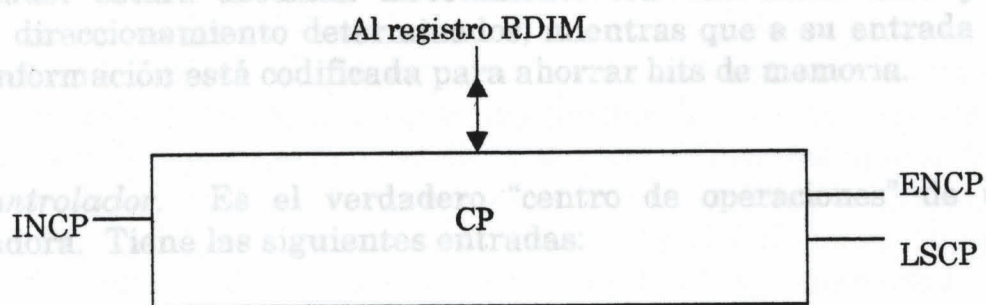


Figura 13.2.2-3 El registro contador del programa

En la figura se pueden apreciar las órdenes ENCP y LSCP, que gobiernan la escritura y la lectura respectivamente, del contador del programa. Podemos ver también la existencia de la orden INCP, cuya activación produce que el contenido del contador de programa se

incremento un número de unidades predeterminado. Esta orden es típica de los registros contadores de programa.

- **Registro de direcciones de memoria (RDIM).** Se utiliza para almacenar las direcciones de memoria en las que se van a escribir o de las que se van a leer datos. La orden ERDI gobierna la escritura en este registro. El contenido de este registro se supone siempre presente en sus líneas de salida, conectadas al bus de direcciones externo, sin necesidad de una orden explícita.

- **Registros de datos de memoria (RDAM).** Almacena temporalmente los datos que se intercambian con la memoria en las operaciones de lectura y de escritura. Una de sus conexiones se considerará como líneas de entrada y salida de conexión directa con el bus de datos externo a la unidad de control, gobernados por las órdenes ERDAE y LRDAE. La otra conexión del RDAE con este bus de datos interno se gobierna con dos órdenes: ERDAI para la escritura en el registro y LRDAI para la lectura en dicho bus de los datos del registro.

- **Decodificador.** Es un circuito combinatorial que toma como entrada los campos CO + MD del registro de instrucción y activa un conjunto de salidas conectadas directamente como entradas al controlador. Simplifica el trabajo de éste ya que cada salida del decodificador estará asociada directamente con una instrucción y un modo de direccionamiento determinados, mientras que a su entrada esa misma información está codificada para ahorrar bits de memoria.

- **Controlador.** Es el verdadero "centro de operaciones" de una computadora. Tiene las siguientes entradas:

Salidas del decodificador: Aquellas que estén activas le indican el tipo de instrucción y el modo de direccionamiento asociados.

Reloj del sistema: Una computadora es un dispositivo esencialmente síncrono. El orden en que tienen lugar los eventos es crítico. Nada sucede fuera de los instantes marcados por el reloj. Por lo tanto, el controlador debe tener información de los pulsos de reloj del sistema. Si existen más de un reloj todos ellos informarán de su estado al

controlador. Muchos circuitos digitales utilizan relojes para su sincronización. Un reloj, en este contexto es un circuito que emite una serie de impulsos determinados con precisión.

Registro de estado. El secuenciador conoce en todo momento el valor de los biestables de este registro de la UAL, de modo que puede gobernar adecuadamente las instrucciones de salto condicional.

Como puede verse en la figura 13.2.2-2, el controlador presenta una serie de salidas denominadas microórdenes. De estas salidas se vale el controlador para gobernar el funcionamiento de la computadora. Las microórdenes están conectadas directamente a las órdenes de los registros (lectura, escritura...) de la UAL a todos los elementos operativos del ordenador.

13.2.3 La memoria principal (MP)

La memoria principal constituye la parte del ordenador que almacena instrucciones y datos. La memoria es un conjunto ordenado de celdas denominadas posiciones de memoria. A cada posición se puede acceder por medio de un número que la identifica unívocamente. Dicho número se conoce con el nombre de dirección de memoria.

La memoria de una computadora no está concentrada realmente en un solo sitio; los dispositivos de almacenamiento están dispersos por toda la máquina. Es deseable que la velocidad de operación de esta sección de la computadora sea tan rápida como sea posible, ya que la mayoría de transferencias de datos desde y hacia la sección de procesamiento de la máquina se hace a través de la memoria principal. Por desgracia los elementos disponibles actualmente de mayor velocidad, para cumplir satisfactoriamente esta función, no poseen la capacidad de almacenamiento que se requiere, por lo que se añade una memoria auxiliar o secundaria a la mayoría de las computadoras. Esta sección de la memoria se caracteriza por el bajo costo por dígito almacenado, aunque tiene una velocidad de operación más lenta que la de memoria principal.

En la figura 13.2.3-1 se muestra una unidad de memoria principal con sus conexiones de comunicación con el exterior.

A continuación se describen las principales líneas de conexión de la memoria principal.

Líneas de direcciones. Su valor binario indica la dirección de la celda de memoria a la que se accede. En una memoria con nd líneas se podrán direccionar hasta 2^{nd} posiciones distintas. El número de líneas de direcciones coincidirá con el número de líneas del bus de direcciones.

Líneas de datos. En las líneas de datos se presentan los datos que se desean escribir en la memoria, o bien, la memoria presenta los datos que se desean leer de la misma. El número de estas líneas, nt , coincidirá con el número de bits por posición de memoria, y generalmente, con el número de líneas del bus de datos.

Línea LEC. Cuando esta línea está activa, se indica a la memoria que se desea realizar una operación de lectura de datos.

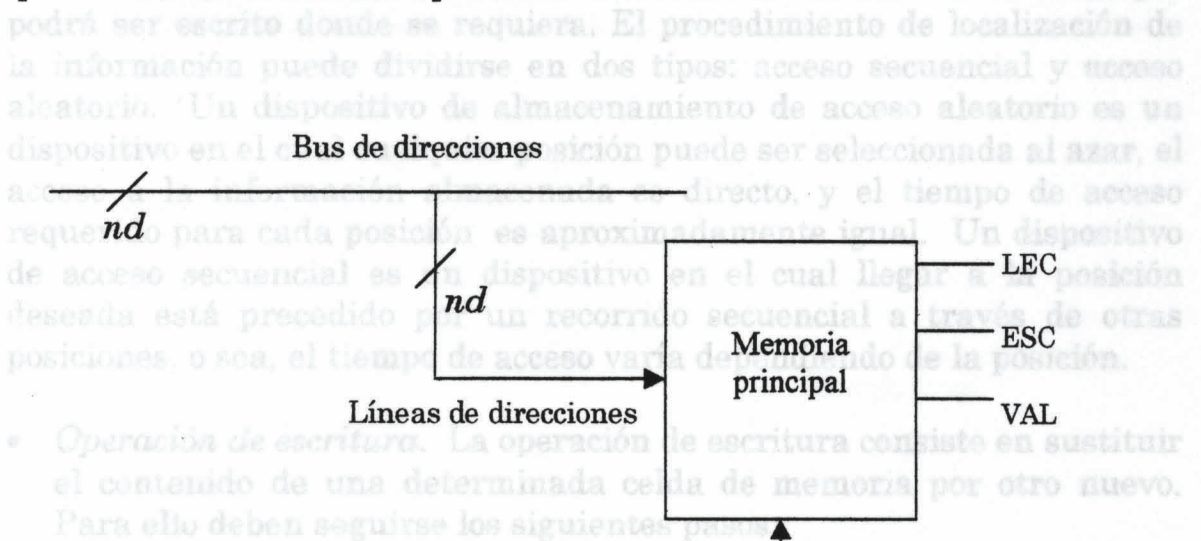


Figura 13.2.3-1 Una unidad de memoria principal

Línea ESC. Cuando esta línea está activa se indica a la memoria que se desea realizar una operación de escritura de datos.

Línea VAL. Valida la memoria. Su activación previa es necesaria para que la activación de las demás líneas produzca el efecto deseado.

Las operaciones de lectura y escritura no se pueden encadenar a
cualquiera de las dos operaciones principales que se pueden realizar sobre las
memorias son, lectura y escritura.

• **Operación de lectura.** Consiste en obtener el contenido de alguna
de las celdas de memoria. Para ello es necesario realizar los siguientes
pasos:

1. Activar la línea VAL.
2. Colocar la dirección de la celda que se desea leer en las líneas de direcciones.
3. Activar la línea LEC.

Al cabo de un tiempo, que se denomina tiempo de acceso a memoria, la
memoria presentará el contenido de la celda seleccionada en sus líneas
de datos y, por lo tanto en el bus de datos del ordenador, desde el que
podrá ser escrito donde se requiera. El procedimiento de localización de
la información puede dividirse en dos tipos: acceso secuencial y acceso
aleatorio. Un dispositivo de almacenamiento de acceso aleatorio es un
dispositivo en el cual cualquier posición puede ser seleccionada al azar, el
acceso a la información almacenada es directo, y el tiempo de acceso
requerido para cada posición es aproximadamente igual. Un dispositivo
de acceso secuencial es un dispositivo en el cual llegar a la posición
deseada está precedido por un recorrido secuencial a través de otras
posiciones, o sea, el tiempo de acceso varía dependiendo de la posición.

• **Operación de escritura.** La operación de escritura consiste en sustituir
el contenido de una determinada celda de memoria por otro nuevo.
Para ello deben seguirse los siguientes pasos:

1. Activar la línea VAL.
2. Colocar en las líneas de direcciones la dirección de la posición de memoria donde se desea escribir.
3. Colocar en las líneas de datos el valor binario del dato que se desea escribir.
4. Activar la línea ESC.

El valor presente en las líneas de todos (y, por lo tanto en el bus de
datos) pasará a sustituir el valor anterior de la celda seleccionada.

Las operaciones de lectura y escritura no se pueden encadenar a cualquier ritmo. Hay que dejar transcurrir un cierto tiempo entre la realización de una operación en memoria y la realización de la operación siguiente. Este tiempo se conoce como ciclo de memoria.

La figura 13.2.3-2 muestra una memoria principal de alta velocidad organizada en palabra de longitud fija. Como indica la figura, una memoria dada está dividida en N palabras, donde N es generalmente una potencia de 2, y a cada palabra se le asigna una dirección o posición en la memoria. Cada palabra tiene el mismo número de bits, llamados longitud de la palabra.

Las direcciones o números de dirección en la memoria, van consecutivamente partiendo de la dirección 0 llegando hasta la dirección más grande. Entonces en la dirección 0 encontraremos una palabra; en la dirección 1 una segunda palabra; en la dirección 2 una tercer palabra y así hasta la última palabra en la dirección más alta.

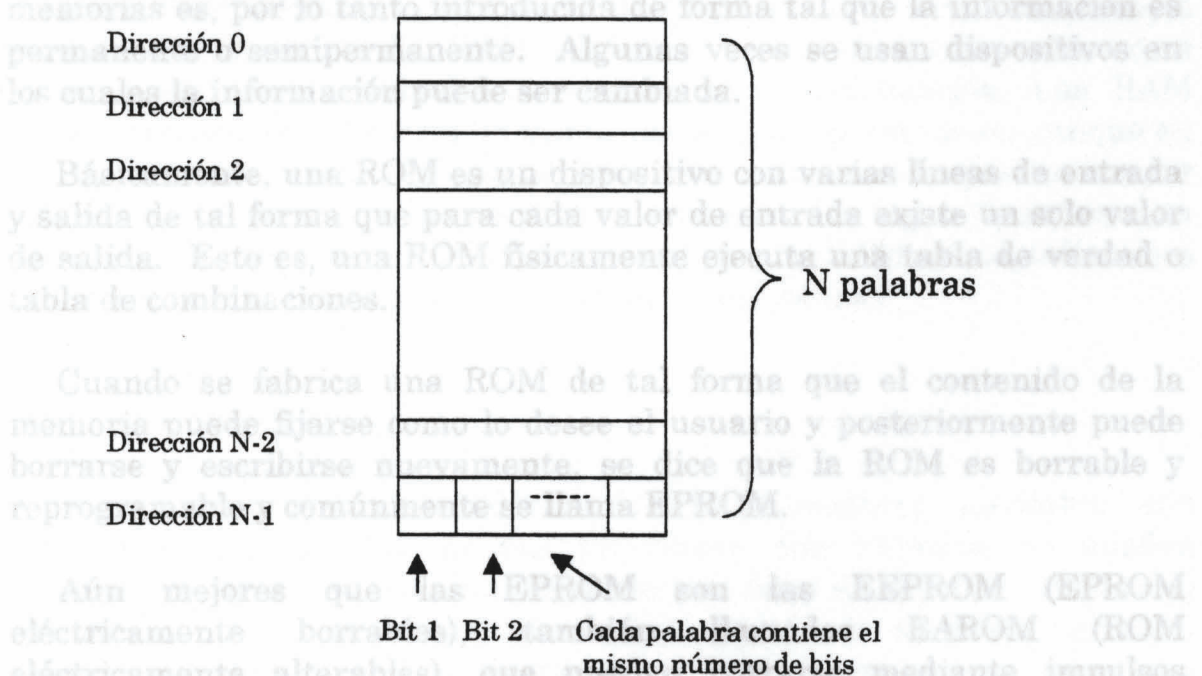


Figura 13.2.3-2 Palabras de una memoria de alta velocidad

Generalmente la computadora puede leer una palabra o escribir una palabra en cada posición de memoria. Para una memoria con palabras de

8 bits, si escribimos la palabra 01001011 en la dirección de memoria 17, posteriormente leemos esta misma posición, y un tiempo después (y no hemos escrito otra palabra en ella) leeremos nuevamente la palabra 01001011. Esto significa que la memoria es de lectura no destructiva, dado que el leer no destruye o cambia la palabra almacenada. (Debido a que las memorias vienen generalmente con un número de palabras igual a 2^n para cualquier n, si una memoria tiene $2^{14} = 16,384$ palabras, se hará referencia a ella como una memoria de 16 k ya que siempre se entiende que la memoria tiene la capacidad de 2^n completa. Por lo tanto una memoria de 2^{15} palabras de 16 bits se denomina memoria de 32k de 16 bits).

Memoria de sólo lectura

Existe un tipo de almacenamiento llamado memoria de solo lectura, ROM (read only memory) que tiene la característica única de que pueden ser leídas, pero no escritas. La información almacenada en esas memorias es, por lo tanto introducida de forma tal que la información es permanente o semipermanente. Algunas veces se usan dispositivos en los cuales la información puede ser cambiada.

Básicamente, una ROM es un dispositivo con varias líneas de entrada y salida de tal forma que para cada valor de entrada existe un solo valor de salida. Esto es, una ROM físicamente ejecuta una tabla de verdad o tabla de combinaciones.

Cuando se fabrica una ROM de tal forma que el contenido de la memoria puede fijarse como lo desee el usuario y posteriormente puede borrarse y escribirse nuevamente, se dice que la ROM es borrable y reprogramable y comúnmente se llama EPROM.

Aún mejores que las EPROM son las EEPROM (EPROM eléctricamente borrables), también llamadas EAROM (ROM eléctricamente alterables), que pueden borrarse mediante impulsos eléctricos, sin necesidad de que las introduzcan en un receptáculo especial para exponerlas a la luz ultravioleta. Las EEPROM difieren de las RAM en que el tiempo necesario para grabar o borrar un byte es miles de veces mayor, a pesar de que los tiempos de acceso en lectura de las ROM, PROM, EPROM EEPROM Y RAM son similares (al máximo unas centenas de nanosegundos).

Memoria de acceso aleatorio (RAM) Hay rutinas de software que son necesarias para este movimiento de segmentos de programas. Estas rutinas Las memorias que pueden utilizarse para leer o escribir se denominan memorias vivas o RAM (random acces memory). Este último nombre es bastante inadecuado puesto que todas las pastillas de memoria son accesibles en forma aleatoria, pero el término ya se ha arraigado.

Hay dos variedades de memorias RAM: Las estáticas y las dinámicas.

Las RAM estáticas se construyen a partir de circuitos similares al del biestable D básico. Tienen la propiedad de retener su contenido tanto tiempo como estén conectadas a la fuente de alimentación, sean segundos, minutos, horas o aun días.

Las RAM dinámicas, por el contrario, no usan circuitos biestables, sino que están construidas como un conjunto de pequeños capacitores que pueden estar cargados o descargados. Como la carga eléctrica tiende a fugarse, cada bit de la RAM dinámica debe refrescarse cada pocos milisegundos, para impedir la pérdida de su información. Las RAM dinámicas son más difíciles de interconectar que las estáticas, aunque en muchas aplicaciones esta desventaja está compensada por su mayor capacidad. Algunas memorias dinámicas retienen la lógica de refresco en la propia pastilla, dando así gran capacidad y facilidad de conexión a los circuitos. Estas pastillas se denominan "casi estáticas".

Memoria virtual

En muchas computadoras, los programadores advierten con frecuencia que algunos de sus programas más extensos no pueden alojarse en memoria para su ejecución por falta de espacio. Incluso si hubiera memoria suficiente para un programa, la memoria principal puede ser compartida con otros usuarios haciendo que un programa cualquiera ocupe una fracción de memoria que puede no ser suficiente para que el programa se ejecute.

Si no cabe todo un programa en la memoria principal física disponible, se trasladan partes de él (páginas), del almacenamiento secundario a la memoria principal, cuando deben ejecutarse, desplazando

las páginas que hayan quedado ociosas. Hay rutinas de software que son necesarias para este movimiento de segmentos de programas. Estas rutinas de manejo son parte de un conjunto de programas que constituyen el sistema operativo de la computadora.

Las técnicas generales de trasladar automáticamente el programa y los bloques de datos necesarios a la memoria principal física para su ejecución se denominan técnicas de memoria virtual. Los programas, y por lo tanto la CPU, hacen referencia a un espacio de instrucciones y de datos independiente del espacio físico de la memoria principal. Las direcciones binarias que emite la CPU, ya sea para instrucciones o para datos, se conocen como direcciones virtuales o lógicas. El mecanismo que opera en estas direcciones virtuales y las traduce a localizaciones verdaderas de la jerarquía física por lo general se implantan empleando una combinación de componentes de hardware y software.

El número de los sistemas de memorias virtuales aumenta con rapidez. Muchos nuevos diseños de computadoras, incluyendo los sistemas de microcomputadoras de alto rendimiento, tienen memoria virtual como parte de su hardware básico y de su software operativo. Otros sistemas se diseñan de tal manera se facilite este implante.

El lector podrá darse cuenta que uno de los elementos más importantes de la computadora son los registros ya que éstos son la clave en el funcionamiento de los procesadores. Todo el conjunto de operaciones que se realizan en el interior de una computadora se puede reducir en esencia, a secuencias de transferencia de información entre sus registros constituyentes.

El diseño lógico es un arte, más que una ciencia exacta. En el diseño del hardware existe un axioma: *más pequeño es más rápido*. Las partes más pequeñas de hardware, generalmente, son más rápidas que las más grandes. Este sencillo principio es aplicable, particularmente a las memorias, por dos razones. Primero, en las máquinas de alta velocidad, la propagación de la señal es una causa importante de retardo: las memorias más grandes tienen más retardo de señal y necesitan más niveles para decodificar las direcciones. Segundo, en muchas tecnologías se pueden obtener memorias más pequeñas, que son más rápidas que en memorias más grandes. Estas características afectan el costo relativo y desempeñan un papel determinante en cuanto a las capacidades de una computadora, aspectos que debe tomar muy en cuenta el profesional de sistemas.

15. CONCLUSION

Esta tesina contiene sólo los aspectos más fundamentales de la arquitectura. El diseñador de sistemas puede realizar excelentes aplicaciones de software sin tener conocimientos de arquitectura de computadoras, sin embargo, el estudio y comprensión del hardware de la computadora, le permitirá utilizarla a su máximo rendimiento, aprovechando mejor su capacidad y haciendo posible la optimización de las aplicaciones desarrolladas en ellas. Esto se logrará a partir de desarrollos de sistemas que utilicen la programación a bajo nivel y la manipulación de la información a nivel de hardware, obteniendo aplicaciones que puedan optimizar la velocidad de operación de un programa, el espacio en memoria y la posibilidad de crear algunas aplicaciones que sin las bases del conocimiento de la lógica digital, no sería posible realizarlas.

La arquitectura de computadoras involucra también el estudio del diseño y funcionamiento de los dispositivos de almacenamiento secundarios, pero debido a que es un tema muy amplio, en este trabajo se identificaron como elementos básicos de la computadora la Unidad Aritmético Lógica, la Unidad de Control y la Memoria Principal únicamente.

El lector podrá darse cuenta que uno de los elementos más importantes de la computadora son los registros ya que estos son la clave en el funcionamiento de los procesadores. Todo el conjunto de operaciones que se realizan en el interior de una computadora se puede reducir en esencia, a secuencias de transferencia de información entre sus registros constituyentes.

El diseño lógico es un arte, más que una ciencia exacta. En el diseño del hardware existe un axioma: *más pequeño es más rápido*. Las partes más pequeñas de hardware, generalmente, son más rápidas que las más grandes. Este sencillo principio es aplicable, particularmente a las memorias, por dos razones. Primero, en las máquinas de alta velocidad, la propagación de la señal es una causa importante de retardo: las memorias más grandes tienen más retardo de señal y necesitan más niveles para decodificar las direcciones. Segundo, en muchas tecnologías se pueden obtener memorias más pequeñas, que son más rápidas que en memorias más grandes. Estas características afectan el costo relativo y desempeñan un papel determinante en cuanto a las capacidades de una computadora, aspectos que debe tomar muy en cuenta el profesional de sistemas.

I. Cambios tecnológicos en las computadoras

Esta tesina contiene sólo los aspectos más fundamentales de la arquitectura de computadoras. Existe una gran variedad de arquitecturas, pero todas están constituidas por los mismos dispositivos electrónicos básicos. Considero que, si se comprenden algunos de los principios del diseño lógico, el Licenciado en Sistemas podrá rápidamente sacar provecho de éstos y obtener grandes beneficios en su vida profesional.

cada uno o dos años, la tecnología DRAM se cuadruplica cada tres o cuatro años. Estos cambios en la tecnología llevan a situaciones que pueden habilitar una técnica de implementación que antes no era posible (por ejemplo, cuando la tecnología MOS logró ubicar entre 25.000 y 50.000 transistores en un único circuito integrado, fue posible construir un microprocesador de 32 bits en una sola pastilla integrada. Al eliminar los cruces entre chips dentro del CPU, fue posible un enorme incremento en el costo/rendimiento).

La cantidad de memoria necesaria por el programa medio ha crecido en un factor de 1.5 a 2 por año. Esto se traduce en un consumo de bits de direcciones a una frecuencia de $\frac{1}{4}$ bit por año.

En la siguiente figura se muestran las tendencias en las tecnologías de implementación de las computadoras.

<i>Tecnología</i>	<i>Tendencias de rendimiento y densidad</i>
Tecnología de CI lógicos	El número de transistores en un chip aumenta aproximadamente el 25 por 100 por año, duplicándose en tres años. La velocidad de los dispositivos aumenta casi a esa rapidez.
DRAM semiconductor	La densidad aumenta en un 60 por 100 por año, cuadruplicándose en tres años. La duración del ciclo ha mejorado muy lentamente, decreciendo aproximadamente una tercera parte en diez años.
Tecnología de disco	La densidad aumenta aproximadamente el 25 por 100 por año, duplicándose en tres años. El tiempo de acceso ha mejorado en un tercio en diez años.

I. Cambios tecnológicos en las computadoras

Los cambios tecnológicos a nivel hardware que se producen en la industria de la computación no son continuos sino que, con frecuencia, se presentan en pasos discretos. Por ejemplo, los tamaños de las DRAM (memoria dinámica de acceso aleatorio) aumentan siempre en factores de 4, debido a la estructura básica de diseño. Entonces en lugar de duplicarse cada uno o dos años, la tecnología DRAM se cuadruplica cada tres o cuatro años. Estos cambios en la tecnología llevan a situaciones que pueden habilitar una técnica de implementación que antes no era posible (por ejemplo, cuando la tecnología MOS logró ubicar entre 25.000 y 50.000 transistores en un único circuito integrado, fue posible construir un microprocesador de 32 bits en una sola pastilla integrada. Al eliminar los cruces entre chips dentro del CPU, fue posible un enorme incremento en el coste/rendimiento).

La cantidad de memoria necesaria por el programa medio ha crecido en un factor de 1.5 a 2 por año. Esto se traduce en un consumo de bits de direcciones a una frecuencia de $\frac{1}{2}$ bit por año.

En la siguiente figura se muestran las tendencias en las tecnologías de implementación de las computadoras.

Tecnología	Tendencias de rendimiento y densidad
Tecnología de CI lógicos	El número de transistores en un chip aumenta aproximadamente el 25 por 100 por año, duplicándose en tres años. La velocidad de los dispositivos aumenta casi a esa rapidez.
DRAM semiconductora	La densidad aumenta en un 60 por 100 por año, cuadruplicándose en tres años. La duración del ciclo ha mejorado muy lentamente, decreciendo aproximadamente una tercera parte en diez años
Tecnología de disco	La densidad aumenta aproximadamente el 25 por 100 por año, duplicándose en tres años. El tiempo de acceso ha mejorado en tercio en diez años

II. Glosario

B Si una arquitectura ha de tener éxito, debe ser diseñada para que sobreviva a los cambios en la tecnología hardware, tecnología software y aplicaciones características. El diseñador debe ser consciente, especialmente de las tendencias en la utilización de la computadora y de la tecnología de las computadoras. Después de todo, una nueva arquitectura a nivel lenguaje de máquina que tenga éxito puede durar decenas de años*. Los cambios tecnológicos deben incrementar con éxito la vida de una máquina. Para planificar la evolución de una máquina se debe ser consciente de los rápidos cambios que experimentan las tecnologías de implementación.

PUERTA LÓGICA que responde desde el punto de vista lógico a una función básica. Las variables que maneja una compuerta surgen con sus entradas, mientras que la función viene dada por su salida. El circuito que configura una puerta lógica está dada por un pequeño número de componentes discretos, como son resistencias, diodos, transistores, etc.

FALLO DE CACHE (caché miss) cuando la CPU no encuentra un dato que necesita en la caché se produce lo que se llama un fallo de caché y el dato se recupera de la memoria principal.

FALLO DE PÁGINA (page fault) Se produce cuando la CPU referencia un elemento de una página que no está presente en la memoria caché o en la principal. Entonces la página completa es transferida del disco a la memoria principal.

FLIP-FLOP Elemento electrónico básico para almacenar información binaria.

INFORMACIÓN DIGITAL BINARIA Es el sistema de dos valores que representa la base con que trabajan las computadoras actuales.

LENGUAJE ENSAMBLADOR Lenguaje de programación próximo al lenguaje de máquina. Utiliza palabras y símbolos para representar los códigos máquina del computador.

MEMORIA CACHE Memoria pequeña y veloz que se inserta entre la CPU y la memoria principal.

*El núcleo de la IBM 360 ha sido utilizado desde 1964.

II. Glosario

BUS Cualquier línea a la que confluyen un conjunto de dispositivos con objeto de efectuar transferencias de información. Es una ruta eléctrica común entre múltiples dispositivos.

CACHÉ memoria pequeña y rápida localizada cerca de la CPU que contiene la mayor parte del código o de los datos recientemente accedidos.

COMPUERTA LÓGICA Es un pequeño circuito, generalmente integrado en una pastilla, que responde desde el punto de vista lógico a una función básica. Las variables que maneja una compuerta lógica son sus entradas, mientras que la función viene dada por su salida. El circuito que configura una puerta lógica está dada por un pequeño número de componentes discretos, como son resistencias, diodos, transistores, etc.

FALLO DE CACHÉ (caché miss) cuando la CPU no encuentra un dato que necesita un la caché se produce lo que se llama un fallo de caché y el dato se recupera de la memoria principal.

FALLO DE PÁGINA (page fault) Se produce cuando la CPU referencia un elemento de una página que no está presente en la memoria caché o en la principal. Entonces la página completa es transferida del disco a la memoria principal

FLIP-FLOP Elemento electrónico básico para almacenar información binaria.

INFORMACIÓN DIGITAL BINARIA Es el sistema de dos valores que representa la base con que trabajan las computadoras actuales.

LENGUAJE ENSAMBLADOR Lenguaje de programación próximo al lenguaje de máquina. Utiliza frases y palabras para representar los códigos máquina del microprocesador.

MEMORIA CACHÉ (Buffer) Memoria pequeña y veloz que se inserta entre la memoria principal (más grande y lenta) y la CPU.

MEMORIA DE ACCESO ALEATORIO (RAM) Memoria que puede transferir información almacenada a una salida (lectura) y es capaz de recibir información nueva de entrada para almacenamiento (escritura), /a diferencia de la memoria (ROM). Un nombre más apropiado para esta memoria sería memoria de lectura-escritura.

MEMORIA DE SOLO LECTURA (ROM) Es un dispositivo de memoria (o almacén) en el cual se almacena un conjunto fijo de información binaria. La información binaria primero debe especificarla el usuario, y entonces se inserta en la unidad para formar el patrón requerido para la conexión. Las ROM se obtienen con eslabones internos especiales que pueden fusionarse o romperse. Una vez que se establece un patrón para una ROM, permanece fijo aun cuando se encienda la potencia y se apaga otra vez.

MICROPROCESADOR Usualmente se utiliza el este término para denominar a cualquier CPU contenida en una sola pastilla, aún cuando algunas de ellas tengan la arquitectura y el poder de cómputo de pequeñas macrocomputadoras.

REGISTRO. Grupo de celdas binarias. Es un conjunto de flip-flops, donde cada flip-flop es capaz de almacenar un bit de información. Ya que una celda almacena un bit de información se deduce que un registro con n celdas puede almacenar cualquier cantidad discreta de información que contenga n -bits

RELOJ. Con objeto de sincronizar las operaciones a una base fundamental de tiempo, existe en la computadora un pulso periódico muy exacto, cuyo nivel baja y sube alternativamente y con el cual se celebra toda actividad dentro de la misma. A este pulso se le llama reloj y constituye una de las partes de la unidad de control.

SISTEMA DIGITAL. Es una combinación de dispositivos diseñados para manipular cantidades físicas o información que estén representadas en forma digital, es decir, que solo pueden tomar valores directos.

SISTEMA ANALÓGICO. Contienen dispositivos que manipulan cantidades físicas representadas en forma analógica. Aquí las cantidades varían sobre un intervalo continuo de valores.

UNIDAD ARITMÉTICO LÓGICA. Dispositivo en donde se ejecutan la mayoría de las operaciones.

UNIDAD DE CONTROL Dispositivo que se encarga de coordinar las operaciones de almacenamiento y proceso de información

UNIDAD DE ENTRADA. Dispositivos por medio de los cuales la computadora acepta información, como por ejemplo el teclado, plumillas luminosas, scáners, etc.

UNIDAD DE MEMORIA. Es una colección de celdas de almacenamiento junto con circuitos asociados necesarios para transferir la información de entrada y salida del almacenamiento. Dispositivo de la computadora cuya función es almacenar programas y datos. *La memoria principal o almacenamiento primario* es una memoria rápida, capaz de operar a velocidades electrónicas. Contiene gran número de celdas semiconductoras de almacenamiento, cada una capaz de almacenar un bit de información.

- ARQUITECTURA DE COMPUTADORES, UN ENFOQUE ESTRUCTURADO, John D. Lenk, ed. Diana, México, 1990.
- MANUAL DE CIRCUITOS DE LÓGICA John D. Lenk, ed. Diana, México, 1990.
- ARQUITECTURA DE ORDENADORES, E. Alcalde, F. Ormaechea, J. Portillo, F. García Merayoed, McGraw-hill, 1991, España
- DISEÑO CON CIRCUITOS INTEGRADOS TTL, Compañía Editorial Continental, 1980, México.
- DICCIONARIO DE FÍSICA, Ed. Ediplesa, 1980, Madrid.
- ELECTRICIDAD Y MAGNETISMO, Francis W. Sears, ed. Aguilar, 1970, Madrid.
- ORGANIZACIÓN DE COMPUTADORAS. V. Carl Hsmacher, Zvonko G. Vranceic, Safwat G. Zaky, Ed. McGraw-Hill, 2ª ed, México
- ORGANIZACIÓN DE COMPUTADORAS, UN ENFOQUE ESTRUCTURADO, Andrew S. Tanenbaum, ed. Prentice-Hall Hispanoamericana, S.A., 3ª ed, México.

III. Bibliografía

- ARQUITECTURA DE COMPUTADORES, UN ENFOQUE CUANTITATIVO. John L. Hennessy y David A. Patterson, ed. McGraw-hill, 1993, España.
- PRINCIPIOS DIGITALES, Roger L. Tokheim, ed. McGraw Hill, 1982, México.
- MANUAL DE CIRCUITOS DE LÓGICA John D. Lenk, ed. Diana, 1981, México.
- ENCYCLOPEDIA OF COMPUTER SCIENCE AN ENGINEERING, ed. Van Nostrand Reinhold Company, 2ª ed., Estados Unidos de N.A.
- INTRODUCCION A LA CIENCIA DE LAS COMPUTADORAS, ENFOQUE ALGORÍTMICO, Jean-Paul Tremblay y Richard B. Bunt., ed Mc-Graw Hill, 1990, México.
- ARQUITECTURA DE ORDENADORES, E. Alcalde, F. Ormaechea, J. Portillo, F. García Merayoed, McGraw-hill, 1991, España.
- DISEÑO CON CIRCUITOS INTEGRADOS TTL, Compañía Editorial Continental, 1980, México.
- DICCIONARIO DE FÍSICA, Ed. Ediplesa, 1980, Madrid.
- ELECTRICIDAD Y MAGNETISMO, Francis W. Sears, ed. Aguilar, 1970, Madrid.
- ORGANIZACIÓN DE COMPUTADORAS. V. Carl Hamacher, Zvonko G. Vranesic, Safwat G. Zaky, Ed. McGraw-Hill, 2ª ed, México.
- ORGANIZACIÓN DE COMPUTADORAS, UN ENFOQUE ESTRUCTURADO, Andrew S. Tanenbaum, ed. Prentice-Hall Hispanoamericana, S.A., 3ª ed, México.

- TEORÍA DE CIRCUITOS, Lawrence P. Hulesman, ed. Prentice-Hall Hispanoamericana, S.A., 2a. ed. México.
- FUNDAMENTOS DE COMPUTADORAS DIGITALES, Thomas C. Bartee, McGraw-Hill, 5ª ed., 1990, España.
- ARQUITECTURA DE COMPUTADORAS Y PROCESAMIENTO PARALELO, Kai Hwang y Fayé A. Briggs, 1988, México.
- FUNDAMENTOS DE DISEÑO DIGITAL CON APLICACIÓN A LA ARQUITECTURA DE COMPUTADORAS. César R. De la Cruz Laso, ed. Trillas, 5ª. ed., 1988, México.
- FUNDAMENTOS DE LOS MICROPROCESADORES. Roger L. Tokheim, McGraw-Hill, 2ª ed. 1990, México.
- <http://www.list200.com.tr/es/index/index.asp>
- <http://www.dlsi.ua.es/~marco/tema3/sld043.htm>
- <http://www.geocities.com/CollegePark/Housing/9955/resumen4.html>
- <http://www.una.ac.cr/u/ararce/Arquitectura/temas.html>