



**Electrónica
Digital para arquitectura de
computadoras**

José Fortunato Zuloaga Cachay
Consuelo Ivonne Del Castillo Casto
Gilberto Carrión Barco
Fredy William Campos Flores
Lucía Isabel Chamán Cabrera

Electrónica Digital Para Arquitectura de Computadoras

**José Fortunato Zuloaga Cachay
Consuelo Ivonne Del Castillo Casto
Gilberto Carrión Barco
Fredy William Campos Flores
Lucia Isabel Chamán Cabrera**

**Electrónica
digital para arquitectura de
computadoras**



Organización
de las Naciones Unidas
para la Educación,
la Ciencia y la Cultura



Cátedra UNESCO
«Paz, Solidaridad y
Diálogo Intercultural»
Universitat Abat Oliba CEU

Título original:
Electrónica
digital para arquitectura de
computadoras

Primera edición: octubre 2020

© 2020, José Fortunato Zuloaga Cachay
Consuelo Ivonne Del Castillo Casto
Gilberto Carrión Barco
Fredy William Campos Flores
Lucia Isabel Chamán Cabrera

Publicado por acuerdo con los autores.
© 2020, Editorial Grupo Compás
Guayaquil-Ecuador

Grupo Compás es una editorial de la Universidad de Oriente desde el 2017, cada uno de sus textos han sido sometido a un proceso de evaluación por pares externos con base en la normativa del editorial. Este texto ha sido sugerido para su indexación en Latindex, Redib, ErihPlus, mediante ISSN 2600-5743 Folio 28701 Folio Único 24972 Centro de Acopio, Ecuador

El copyright estimula la creatividad, defiende la diversidad en el ámbito de las ideas y el conocimiento, promueve la libre expresión y favorece una cultura viva. Quedan rigurosamente prohibidas, bajo las sanciones en las leyes, la producción o almacenamiento total o parcial de la presente publicación, incluyendo el diseño de la portada, así como la transmisión de la misma por cualquiera de sus medios, tanto si es electrónico, como químico, mecánico, óptico, de grabación o bien de fotocopia, sin la autorización de los titulares del copyright.

Editado en Guayaquil - Ecuador

ISBN: 978-9942-33-333-9



Cita.

Zuloaga. J, Del Castillo. C, Carrión. G, Campos. F, Chamán. L. (2020) Electrónica digital para arquitectura de computadoras Editorial Grupo Compás, Guayaquil Ecuador 313 pag

Índice

Introducción	11
Capítulo 1: Fundamentos de Electrónica Analógica	
1.1. Magnitudes eléctricas	13
1.1.1. Carga eléctrica	13
1.1.2. Corriente eléctrica	14
1.1.3. Voltaje	15
1.1.4. Ley de Ohm	17
1.1.5. Resistencia eléctrica	18
1.1.6. Potencia eléctrica	19
1.1.7. Energía y potencia	20
1.1.8. Corriente alterna	20
1.1.9. Generación de corriente alterna	21
1.1.10. Valores típicos de una señal alterna	22
1.1.10.1. Frecuencia	22
1.1.10.2. Valor eficaz y valor medio	23
1.2. Dispositivos semiconductores	24
1.2.1. Diodos	25
1.2.2. Transistores BJT	28
1.2.2.1. Zonas de trabajo del transistor	29
1.2.3. Amplificadores operacionales	31
1.3. Señales analógicas y digitales	34
1.3.1. Señales analógicas	35
1.3.2. Señales digitales	35
1.3.3. Lógica digital	35
1.3.4. Lógica positiva y lógica negativa	36
Problemas Resueltos N° 01	36
Problemas Propuestos N° 01	45

Capítulo 2: Sistemas y códigos de numeración	
2.1. Representación de datos	47
2.1.1. Representación de números, sistemas de numeración	48
2.1.2. Sistema de numeración binario	48
2.1.3. Conversión del sistema binario a decimal y viceversa	49
2.1.4. Sistema octal y hexadecimal	51
2.1.5. Conversión del sistema octal y hexadecimal a binario y viceversa	52
2.2. Representación de números en formato signo magnitud	54
2.3. Códigos binarios	55
2.3.1. Código BCD	55
2.3.2. Código BCD Aiken	56
2.3.3. Código BCD exceso 3	57
2.3.4. Código Gray	57
2.3.5. Códigos alfanuméricos	59
2.4. Detectores de error	60
2.4.1. Bit de paridad	60
Problemas Resueltos Nº 02	61
Problemas Propuestos Nº 02	64
Capítulo 3: Descripción de circuitos digitales	
3.1. Tablas de verdad	65
3.2. Variables y funciones lógicas	66
3.3. Diagramas de tiempo	66
3.4. Lógica de contactos	67
3.5. Compuertas lógicas	67
3.5.1. Compuerta lógica OR	68
3.5.2. Compuerta lógica AND	69
3.5.3. Compuerta lógica NOT	70
3.5.4. Compuerta lógica NOR	70
3.5.5. Compuerta lógica NAND	71
3.5.6. Compuerta lógica OR EXCLUSIVA	72
3.6. Representaciones alternas de compuertas lógicas	73
3.7. Implementación de circuitos	74
3.8. Diagramas lógicos	74
3.9. Diagramas esquemáticos	75
Problemas Resueltos Nº 03	76
Problemas Propuestos Nº 03	81
Capítulo 4: Álgebra de Boole	
4.1. Definición	83
4.2. Expresiones de una función lógica	84
4.3. Teoremas del Álgebra de Boole	84
4.3.1. Propiedad de Identidad	84
4.3.2. Propiedad conmutativa	85

4.3.3.	Propiedad asociativa	85
4.3.4.	Propiedad distributiva	86
4.3.5.	Propiedad del complemento	86
4.3.6.	Propiedad de idempotencia	87
4.3.7.	Propiedad de unicidad del complemento	87
4.3.8.	Propiedad de los elementos dominantes	87
4.3.9.	Propiedad involutiva	88
4.3.10.	Propiedad de absorción	89
4.3.11.	Propiedad de consenso	90
4.4.	Teoremas De Morgan	91
4.4.1.	Efecto del Teorema de Morgan en las compuertas AND y OR.	92
4.5.	Versatilidad de las compuertas NAND y NOR	92
4.6.	Simplificación de funciones lógicas	95
	Problemas Resueltos N° 04	96
	Problemas Propuestos N° 04	103
Capítulo 5: Estandarización y métodos de simplificación de funciones		
5.1.	Formas canónicas de funciones lógicas	105
5.1.1.	Expresiones canónicas en suma de productos “minitérminos”	107
5.1.2.	Expresiones canónicas en productos de sumas “maxitérminos”	109
5.2.	Método de simplificación de funciones lógicas	111
5.2.1.	Mapas de Karnaugh	111
5.2.1.1.	M.K. de dos variables	112
5.2.1.2.	M.K. de tres variables	113
5.2.1.3.	M.K. de cuatro variables	113
5.3.	Influencia de las variables en las celdas del M.K.	114
5.4.	Proceso de simplificación de funciones lógicas con M.K.	115
5.5.	Funciones incompletamente especificadas	117
	Problemas Resueltos N° 05	118
	Problemas Propuestos N° 05	125
Capítulo 6: Circuitos combinatoriales y circuitos integrados		
6.1.	Introducción a los circuitos combinatoriales	127
6.2.	Implementación de circuitos combinatoriales	127
6.2.1.	Partiendo de una función booleana	128
6.2.2.	Partiendo de una tabla de verdad	130
6.3.	Proceso de desarrollo de circuitos de control lógico	133
6.3.1.	Circuitos de habilitación y deshabilitación	133
6.3.2.	Circuitos de control	134
6.4.	Características de los circuitos integrados digitales	138
6.4.1.	Compuertas lógicas con tres estados	138
6.4.2.	Salida de una compuerta en Colector Abierto (Open colector)	140
6.4.3.	Compuertas con entradas de disparador Schmitt (Schmitt trigger)	141
	Problemas Resueltos N° 06	142
	Problemas Propuestos N° 06	154

Capítulo 7: Codificadores y multiplexores

7.1.	Codificadores	155
7.1.1.	Codificadores sin prioridad	157
7.1.2.	Codificadores con prioridad	158
7.2.	Multiplexores	161
7.2.1.	Diseño de un multiplexor de cuatro canales	163
7.2.2.	Multiplexor comercial de ocho canales	164
7.2.3.	Aplicaciones del multiplexor	165
	Problemas Resueltos Nº 07	167
	Problemas Propuestos Nº 07	175

Capítulo 8: Decodificadores, demultiplexores y comparadores de magnitud

8.1.	Decodificadores	177
8.1.1.	Decodificadores con una salida activada	177
8.1.1.1.	Diseño de un decodificador de ocho salidas con puertas NAND	179
8.1.2.	Decodificadores con varias salidas activadas	182
8.1.2.1.	Visualizador numérico de siete segmentos	182
8.1.2.2.	Decodificadores/controladores de BCD a siete segmentos	184
8.2.	Demultiplexores	186
8.3.	Comparadores de magnitud	187
8.3.1.	Diseño de un comparador de 1 bit de dato en cada canal de entrada	187
8.3.2.	Diseño de un comparador de tres bits e datos en cada canal de entrada	188
	Problemas Resueltos Nº 08	190
	Problemas Propuestos Nº 08	196

Capítulo 9: Circuitos aritméticos

9.1.	Introducción	199
9.2.	Operaciones aritméticas	200
9.2.1.	Suma	200
9.2.2.	Resta	201
9.2.3.	Multiplicación	202
9.3.	Representación de números con signo	203
9.4.	Sistema de complemento a 2	204
9.5.	Operaciones en el sistema de complemento a 2	206
9.5.1.	Suma en complemento a 2	206
9.5.2.	Resta en complemento a 2	207
9.5.3.	Desbordamiento aritmético	208
9.6.	Suma BCD	209
9.7.	Sumador digital	210
9.7.1.	Semisumador	210
9.7.2.	Sumador total	211

9.7.3.	Diseño de un sumador total	213
9.7.4.	Circuitos restadores	213
9.8.	Bloques sumadores prácticos de 4 bits de datos	214
	Problemas Resueltos Nº 09	216
	Problemas Propuestos Nº 09	222
Capítulo 10: Circuitos secuenciales		
10.1.	Introducción a los circuitos secuenciales	223
10.2.	Flip Flop SR	224
10.2.1.	Latch de compuertas NAND	224
10.2.2.	Latch de compuertas NOR	226
10.3.	Sincronización de los biestables	229
10.3.1.	Circuito generador de reloj	229
10.3.2.	Pulsos digitales	232
10.3.3.	Flip Flop S-R sincronizados por reloj	233
10.4.	Flip Flop JK	234
10.4.1.	FF JK Master Slave	234
10.4.2.	FF JK edge – triggered	235
10.4.3.	Entradas asíncronas en el FF JK	235
10.5.	Flip Flop tipo D	235
10.6.	Flip Flop tipo T	236
10.7.	Circuitos integrados comerciales de Flip Flop	237
	Problemas Resueltos Nº 10	238
	Problemas Propuestos Nº 10	242
Capítulo 11: Almacenamiento y transferencia de datos		
11.1.	Registros	243
11.2.	Registros de desplazamiento	244
11.2.1.	Registro de desplazamiento: entrada y salida en serie	244
11.2.2.	Registro de entrada paralelo y salida serie	251
11.2.3.	Registro de entrada serie y salida paralelo	252
11.2.4.	Registro de entrada paralelo y salida paralelo	253
11.3.	Registro de desplazamiento en lazo cerrado	254
11.4.	Registros comerciales 74164, 74165, 74166, 74194	255
11.5.	Bus de datos	256
	Problemas Resueltos Nº 11	257
	Problemas Propuestos Nº 11	259
Capítulo 12: Contadores		
12.1.	Introducción	261
12.2.	Clasificación de los contadores	262
12.2.1.	Contadores asíncronos de rizo	262
12.2.1.1.	Contador asíncrono ascendente	265
12.2.1.2.	Contador asíncrono descendente	266
12.2.1.3.	Contador Módulo N	269

12.2.2.	Contadores síncronos	270
12.2.2.1.	Contador síncrono binario	270
12.2.2.2.	Contador década síncrono	273
12.2.2.3.	Diseño de contadores síncronos utilizando las entradas del Flip Flop JK de manera independiente	275
12.2.2.4.	Contador escalonador	276
	Problemas Resueltos Nº 12	277
	Problemas Propuestos Nº 12	283
Capítulo 13: Dispositivos de memoria		
13.1.	Memoria digital	285
13.2.	Características generales	286
13.3.	Medición de la memoria	286
13.4.	Clases de memoria	287
13.4.1.	Memorias de semiconductores	287
13.4.1.1.	Memorias de acceso aleatorio (RAM)	287
13.4.1.2.	Matriz de memoria semiconductor básica	290
13.4.1.3.	Estructura de una RAM semiconductor	290
13.4.1.4.	Memorias de solo lectura (ROM)	292
13.4.1.5.	Memorias Flash	293
13.4.2.	Memorias magnéticas	295
13.4.3.	Memorias ópticas	296
13.5.	Distribución interna de la memoria	297
	Problemas Resueltos Nº 13	298
	Problemas Propuestos Nº 13	300
Capítulo 14: Estructura básica de un computador		
14.1.	El computador	301
14.2.	Organización y arquitectura	302
14.3.	Estructura y función	302
14.4.	El microprocesador	304
14.4.1.	Unidad Aritmético lógica	305
14.4.2.	Registro temporal de almacenamiento de datos y direcciones	307
14.4.2.1.	Registros visibles al usuario	307
14.4.2.2.	Registros de control y de estado	307
14.4.3.	Unidad de Control	307
14.5.	Funcionamiento del computador	308
	Bibliografía	313

Introducción

El desarrollo vertiginoso de los sistemas computacionales a revolucionado el mundo, pues, desde los artefactos domésticos más sencillos, hasta los más avanzados aparatos de comunicación y equipos de control, cuentan con dispositivos y componentes electrónicos digitales que posibilitan un adecuado manejo y procesamiento de señales basados en niveles lógicos.

Las computadoras personales y los sistemas digitales en general poseen circuitos secuenciales provistos de memoria, en los cuales es común almacenar algún tipo de software que facilita el gobierno de los diversos componentes del que dispone.

Aunque el funcionamiento interno de todos los aparatos digitales es transparente a la mayoría de las personas que las manipulan, es necesario que consideremos la posibilidad de interesarnos cada vez más en comprender el funcionamiento de dichos artefactos, habida cuenta que la aparición de esta tecnología ha venido para quedarse con nosotros.

Los estudiantes de las carreras profesionales de ingeniería de sistemas, así como los de computación e informática, tienen a los diversos sistemas computacionales como herramienta de trabajo, por lo que es necesario que comprendan y dominen las más básicas y avanzadas técnicas digitales para que su desempeño profesional se perfeccione cada vez más y su trabajo se optimice en este mundo tan competitivo.

Este libro tiene como propósito facilitar las ideas básicas, así como los principios y técnicas digitales en un nivel básico e intermedio, que posibilite a quien lo lea y practique, adquirir destrezas y habilidades en el análisis y

diseño de circuitos combinacionales y secuenciales, conocimientos que son la base para incursionar en el mundo de los microcontroladores y para comprender la arquitectura y el funcionamiento de las computadoras personales.

Asimismo, es preciso aclarar que en nuestro medio existe una variedad de textos de muy alto nivel académico en electrónica digital, que obviamente está destinado para aquellas personas que desean especializarse en sistemas digitales. No obstante, es conveniente precisar que en el texto de “Electrónica Digital para Arquitectura de Computadoras” se encontrará la mayoría de los temas tratados en las currículas profesionales de las universidades e institutos de educación superior de nuestro medio, los cuales están expuestos de una manera sencilla y didáctica.

El presente texto está estructurado en catorce capítulos y distribuidos en tres bloques y un capítulo final donde se muestra la estructura básica de un computador personal.

El capítulo 1 expone los fundamentos de la electrónica analógica que son la base para estudiar los circuitos y componentes digitales.

Del capítulo 2 al capítulo 9 se trata todo lo relacionado con los circuitos combinacionales, en ellos, se estudian los sistemas de numeración binario, el álgebra de Boole, métodos de simplificación, codificadores, multiplexores, decodificadores, demultiplexores, comparadores y circuitos aritméticos.

Del capítulo 10 al capítulo 13 se analizan los circuitos secuenciales, los cuales utilizan un sistema generador de reloj para secuenciar sus principales funciones, como son los Flip Flops, Registros, contadores y memorias.

El capítulo 14 presenta la arquitectura básica de un sistema computacional.

Los contenidos tratados en el presente texto, puede desarrollarse convenientemente en un semestre académico, asimismo, por las características de su estructura y desarrollo de los temas, puede ser utilizado por aquellas personas autodidactas o que realizan estudios profesionales a distancia, afianzando sus conocimientos con un conjunto de ejercicios y problemas resueltos y propuestos.

Autores.

Capítulo 1

Fundamentos de Electrónica Analógica

1.1. Magnitudes eléctricas

Las magnitudes eléctricas son todas aquellas cantidades que describen cuantitativamente a los circuitos eléctricos y electrónicos. Las más importantes son: la carga, corriente, voltaje, potencia y energía eléctrica.

1.1.1. Carga eléctrica

Es la propiedad física que poseen los electrones y los protones. Todos los cuerpos pueden cargarse eléctricamente, unos más que otros, pero cuando un cuerpo posee igual cantidad de cargas negativas y positivas se dice que dicho cuerpo es eléctricamente neutro. Cuando hay un desbalance en las cargas eléctricas, se dice que los cuerpos son **negativos** si hay mayor cantidad de electrones y, **positivos** si tienen mayor cantidad de protones o menor cantidad de electrones. La interacción entre los cuerpos electrizados se manifiesta en la repulsión de cargas iguales o la atracción de cargas opuestas.

La unidad de medida de la *carga eléctrica* (Q) es el *coulomb* (C), el cual es equivalente a la electricidad de $6,25 \times 10^{18}$ electrones.

La carga eléctrica del electrón es $-1,602 \times 10^{-19} C$.

Las impresoras de tecnología láser utilizan la propiedad de las cargas eléctricas para imprimir documentos. El proceso de impresión lo realiza una luz láser del sistema óptico de la impresora, esta luz incide sobre un tambor del cartucho de tóner. La parte del tambor iluminada por el láser se carga eléctricamente y atrae partículas de tóner cargadas con carga eléctrica opuesta al tambor. El papel pasa por el tambor y luego a un mecanismo de fijación en el que se calienta y fija el tóner.

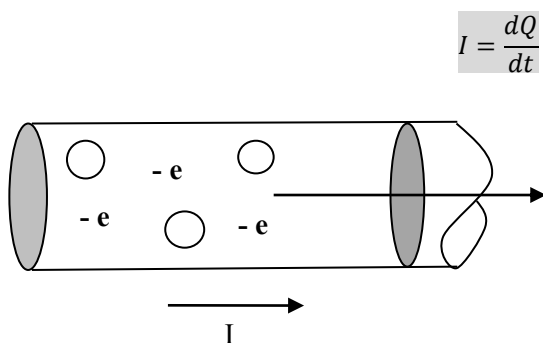
El cuerpo humano también se carga eléctricamente, ya sea por el

rozamiento con el aire o nuestras prendas de vestir. A ésta acumulación de cargas se le llama “cargas electrostáticas”. Ésta electricidad imperceptible para nosotros puede dañar fácilmente a dispositivos electrónicos de tecnología MOSFET.

Para evitar daños en componentes de una computadora por efecto de una *descarga electrostática (ESD)*, es necesario que realicemos un proceso de descarga utilizando el sistema de Tierra, o simplemente tocando por breves segundos cualquier material metálico que haga contacto con Tierra.

1.1.2. Corriente eléctrica

La *corriente eléctrica (I)* se define como el flujo de *cargas eléctricas (Q)* que atraviesan la sección de un conductor eléctrico en un determinado *tiempo (t)*.



$$I = \frac{dQ}{dt}$$

Fig. 1.1

Representación del movimiento de cargas eléctricas en un conductor eléctrico.

El movimiento de los electrones libres en un conductor produce el flujo de cargas en un sentido, como I en la Fig. 1-1; podemos decir que en el sentido contrario circulan cargas positivas. Convencionalmente, el sentido de la corriente electrónica es opuesto al sentido del movimiento de los electrones.

La unidad de medida de la corriente eléctrica es el *Amperio (A)*, donde:

$$1A = \frac{1C}{s}$$

Un *Amperio (A)* es equivalente a la circulación de un *coulomb (C)* o $6,25 \times 10^{18}$ electrones en un segundo.

El instrumento de medida de la corriente eléctrica es el **amperímetro**¹. Éste instrumento se instala en serie con el circuito tal como se muestra en la Fig. 1.2.

¹ El *amperímetro*, el *voltímetro* y el *ohmímetro* se hallan en un solo instrumento llamado **multímetro** o **multitester**.

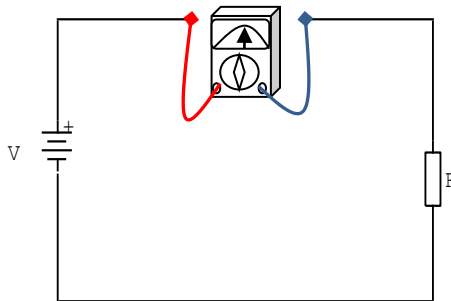


Fig.1.2

Para medir la corriente eléctrica, el amperímetro se instala en serie con el circuito.

Los efectos de la corriente eléctrica cuando circula por el cuerpo humano están en función de la Intensidad de la corriente y el tiempo de aplicación. Cuando circula una corriente entre $1mA$ y $100mA$, el organismo reacciona de distintas formas, desde temblores musculares hasta dolorosas contracciones y paro respiratorio. Si la intensidad de corriente es superior a $100mA$ es probable la muerte por arritmia cardiaca y quemaduras de primer y segundo grado.

Es preciso aclarar que la cantidad de corriente que puede circular por el organismo humano, depende de la resistencia eléctrica del mismo. A mayor resistencia del cuerpo, menor será la corriente que circule por el mismo.

La fuente de poder de una computadora proporciona suficiente corriente para que funcionen los dispositivos y componentes instalados. Por ejemplo, el cable rojo de la salida de la fuente de alimentación abastece con $50A$, aproximadamente.

1.1.3. Voltaje

El *voltaje* (V), denominado también *tensión eléctrica* o *diferencia de potencial*, se define básicamente como el efecto del *trabajo* (W) realizado por el *campo eléctrico* (E) sobre una *carga eléctrica* (Q); lo que en realidad viene a ser la variación de *energía potencial eléctrico* (ΔU) por unidad de carga entre dos puntos de un *campo eléctrico*, es decir, el trabajo (W) es equivalente a la energía potencial (ΔU).

$$V = \frac{W}{Q} = \frac{\Delta U}{Q} = -E \cdot \Delta d$$

La unidad de medida del voltaje es el *voltio* (V)

$$1 \text{ voltio } (V) = \frac{1 \text{ joulio } (J)}{\text{coulomb } (C)} = \left(\frac{1 \text{ newton } (N)}{\text{coulomb } (C)} \right) \times 1 \text{ metro } (m)$$

Nótese que la unidad de la *energía potencial eléctrico* (ΔU) es el **joulio (J)**.

El instrumento de medida del voltaje es el **voltímetro**. Para medir el voltaje de una fuente de poder, o la caída de tensión en la carga de un circuito. El voltímetro

se instala en paralelo con el circuito, tal como se muestra en la Fig. 1.3.

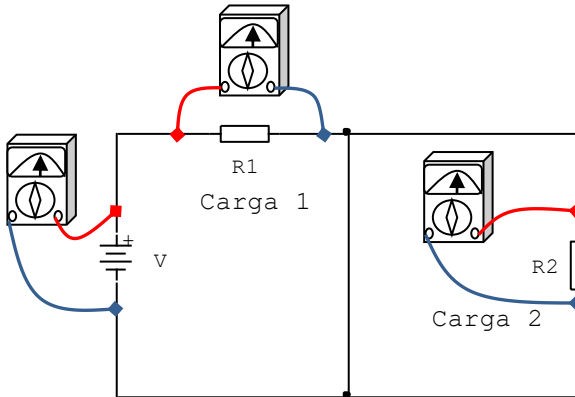


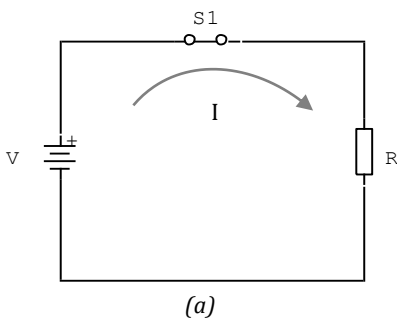
Fig. 1.3

Instalación de un volímetro en paralelo con los componentes en los que se desea saber su voltaje.

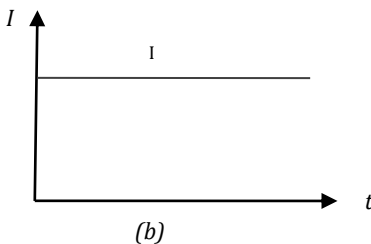
Las pilas, baterías y los generadores eléctricos, son fuentes de voltaje, pues internamente realizan trabajo para separar las cargas positivas de las negativas y así producir una diferencia de potencial que permitirá el flujo de corriente eléctrica en el interior de un circuito.

En general, hay dos clases de fuentes de voltajes: **continua** y **alterna**. La fuente de voltaje continua proporciona *corriente continua o directa (c.d)*, a su vez, la fuente de voltaje alterna genera *corriente alterna (c.a)*, tema que se tratará en (1.1.8 y 1.1.9).

El símbolo de la fuente de voltaje continua es



(a)



(b)

Fig. 1.4

(a) Circuito serie cuya corriente circula del positivo de la fuente hacia el negativo, pasando por la resistencia.

(b) Representación de la corriente continua que circula por un circuito eléctrico, sin cambiar de polaridad en el transcurso del tiempo.

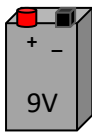
El voltaje producido por las pilas, baterías y acumuladores suministra *corriente directa (c.d)* a un circuito eléctrico o electrónico, en función del tiempo de aplicación. A mayor tiempo de uso la diferencia de potencial entre sus bornes es cada vez menor. Se procede a cargarlos eléctricamente si son “recargables”, sino, se reemplazan completamente por uno nuevo.



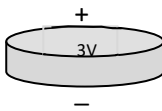
ese momento trabajaban.

La batería de 12V y 50AH, es un acumulador de cargas eléctricas, recargable. Se utilizan en los (Ups) convertidores de 12V *c.d* a 220V de *c.a*, para proveer corriente alterna a los equipos de cómputo ante un eventual corte de energía de la red eléctrica y de ésta manera salvaguardar la “data” y los procesos informáticos que en

El valor nominal de 50AH, indica que la batería puede proporcionar una corriente de 50 Amperios en una hora, o bien, 1 Amperio en 50 horas. Sin embargo, una computadora requiere más de 50 AH, lo que significa que los usuarios sólo dispondrán de aproximadamente 20 minutos para guardar la “data” y terminar el proceso con bastante seguridad.



Batería alcalina de 9V y 150mAH (miliamperio hora), dimensiones 17,5 x 26,5 x 48,5 mm; muy utilizado para la experimentación con circuitos digitales. Se utiliza un regulador de voltaje a 5V, y a éste voltaje se instalan circuitos integrados y demás componentes para comprobar algunas funciones lógicas o el diseño de circuitos electrónicos.



Las placas principales o mainboard de las computadoras disponen de una batería de 3V, en especial del modelo CR-2032. Se trata de una pila de litio, tipo botón, de larga duración. Provee de energía a una memoria del tipo CMOS.

1.1.4. Ley de Ohm

La ley de Ohm establece una relación entre el voltaje, la corriente y la *resistencia eléctrica (R)*. Cuando la *resistencia (R)* es constante, el *voltaje (V)* es directamente proporcional a la *corriente eléctrica (I)*; esto es, a mayor voltaje, circulará mayor corriente por el circuito.

$$V = I \times R$$

Si deseamos que más corriente circule por un circuito serie como el de la Fig. 1.4 (a), deberíamos bajar la resistencia (R) y mantener constante el voltaje de la fuente de alimentación. Para realizar ésta operación se deberá tener en cuenta la potencia eléctrica de la fuente.

1.1.5. Resistencia Eléctrica

La *resistencia eléctrica* (R) se define como la oposición que ofrece un conductor al paso de la corriente eléctrica.

La *resistencia* (R) de un conductor se halla dividiendo el voltaje entre la corriente.

$$R = \frac{V}{I}; \quad \text{Resistencia} = \frac{\text{Voltaje}}{\text{Corriente}}$$

La unidad de medida de la *resistencia eléctrica* (R), es el *Ohmio* (Ω).

$$1 \text{ Ohmio} = \frac{1 \text{ Voltio}}{1 \text{ Amperio}}$$

Es decir, si se aplica un voltaje de $10V$ a un conductor y por él circula una corriente de $2A$, entonces, éste conductor tendrá una resistencia de 5Ω .

El símbolo de la resistencia eléctrica es una línea quebrada en la simbología americana, mientras que en la europea, es un rectángulo.



Simbología americana

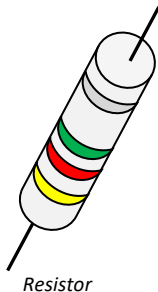


Simbología europea

El instrumento de medida de la *resistencia eléctrica* (R), es el **ohmímetro**.

Para medir la resistencia de un resistor, el ohmímetro se conecta en paralelo, es decir, se conectan cada punta de medida del instrumento a los extremos del dispositivo.

Si el resistor se localiza el interior de un circuito eléctrico, es necesario desconectar previamente la fuente de voltaje.



Los resistores, son dispositivos que presentan una determinada resistencia al paso de la corriente eléctrica. El valor de resistencia de cada uno de ellos se establece según unas franjas de colores en su superficie y leídos según un código de colores.

Todos los componentes de un sistema de cómputo: fuentes de poder, placa principal, discos duros, monitores, impresoras, etc., incorporan en sus circuitos, en su mayoría, resistores de carbón.

1.1.6. Potencia eléctrica

La *potencia eléctrica* (P) de una fuente de energía, es la rapidez con el que transfiere energía potencial eléctrica a un dispositivo determinado.

$$P = \frac{dU}{dt} = d \frac{(QV)}{dt} = \frac{dQ}{dt} \times V$$

La *potencia eléctrica* (P), se puede definir como el producto de la *corriente* (I) por el *voltaje* (V).

$$P = I \times V$$

La unidad de medida de la potencia eléctrica, es el *watt* (W).

$$1 \text{ watt} = \frac{1 \text{ joulio}}{\text{segundo}} = 1 \text{ amperio} \times \text{voltio}$$

El instrumento de medida de la potencia eléctrica es el **wattímetro**. Aunque en electrónica no es común el uso de éste instrumento, en caso se requiera, se puede utilizar un **amperímetro** y un **voltímetro** simultáneamente, y con los valores obtenidos se calcula la potencia del circuito.

Todos los equipos electrónicos obtienen energía eléctrica de una fuente de poder o fuente de alimentación, el cual provee de potencia eléctrica a los dispositivos y componentes del circuito, es decir, le suministra corriente y voltaje.

Un monitor LCD (pantalla de cristal líquido) de 17" consume 35W, mientras que uno TRC (tubo de rayos catódicos), requiere 80W.

Una impresora común de inyección de tinta, funciona con una fuente de alimentación de 12W de potencia, con una salida de 30V y 400mA.

Una computadora actual funciona con una fuente de poder de 600W, mientras que las más antiguas funcionaban correctamente con 150W y 200W; pero, cada vez que el sistema computacional evoluciona y nos procura mejores prestaciones, requiere a cambio, cada vez más potencia.

Aunque el voltaje no varía, sí lo hace la corriente. Si una fuente de 150W aportaba 20A en el cable rojo de 5V; una fuente de 600W provee 50A en el mismo terminal.

Los estabilizadores de voltaje, en los cuales se instalan las computadoras, disponen de potencia que varía según el fabricante y modelo, que van desde 600W a más de 2 000W o 2KW.

1.1.7. Energía y potencia

Es importante comprender la diferencia entre la energía y la potencia eléctrica. Ambas magnitudes están relacionadas de tal manera que la **energía (W)** es directamente proporcional a la **potencia (P)** y al **tiempo (t)** de uso.

$$W = P \times t$$

Cuando la empresa proveedora de energía eléctrica emite un recibo por consumo mensual, lo hace tomando en cuenta el consumo de energía eléctrica y no de la potencia disponible.

Esto quiere decir que el consumo de energía de la fuente de poder de un artefacto con una determinada potencia, será mayor cuanto más tiempo se use dicho artefacto.

La unidad de medida de la energía consumida es el *Watt-segundo (W.s)*. Sin embargo, para cuestiones prácticas, ésta cantidad es demasiado pequeña, es por ello que se utiliza la unidad **Watt-hora (W.h)** ó el **Kilowatt-hora (KW.h)**

$$\text{Kilowatt.hora} = \frac{Pt}{1000}$$

Si en un día promedio, utilizamos la computadora de (500W) durante 5 horas ¿Cuál será el consumo de energía en un mes?

$$\text{Consumo de energía diario} = \frac{500 \times 5}{1000} = 2,5 \text{ KW.h}$$

$$\text{Consumo de energía mensual} = 2,5 \text{ KW.h} \times 30 = 75 \text{ KW.h}$$

Tómese en cuenta a modo de referencia el consumo energético que realiza un foco de filamento incandescente de 100W, y téngase presente que el consumo de energía de cada uno de los equipos utilizados es acumulativo.

1.1.8. Corriente Alterna

Cuando deseamos trabajar con una computadora y todos sus periféricos, los conectamos o instalamos al tomacorriente, del cual obtenemos la energía necesaria para su funcionamiento.

Cuando la corriente es alterna, significa que el sentido de la corriente se invierte periódicamente. Si la corriente es alterna, el voltaje también es alterno, lo que quiere decir, que está variando con el tiempo.

Algunos países como el nuestro utilizan 220V de corriente alterna, mientras

que otros han adoptado 110V, para los mismos propósitos. Las empresas proveedoras de energía eléctrica, nos proporcionan corriente alterna a través de una red eléctrica normalmente trifásica.

La vivienda en su gran mayoría recibe energía eléctrica a través de una red eléctrica monofásica (usa sola fase), sin embargo, las empresas e industrias cuentan con un sistema trifásico, puesto que sus equipos y motores eléctricos funcionan con tres fases.

La corriente alterna a diferencia de la continua, presenta la capacidad de ser transportado a grandes distancias, desde los centros generadores hasta las ciudades o industrias, a través de cables de alta tensión y baja corriente, con la finalidad de reducir las pérdidas de energía por el efecto Joule.

1.1.9. Generación de corriente alterna

La generación de corriente alterna se realiza mediante un dispositivo llamado “generador”, cuyo atributo es convertir energía mecánica en energía eléctrica, aprovechando el fenómeno de inducción electromagnética.

La inducción electromagnética se presenta cuando un conductor en forma de espira gira en el interior de un campo magnético uniforme, creándose una *fuerza electromotriz o diferencia de potencial* en los extremos del conductor, el cual varía o se alterna en el tiempo. El mismo efecto se produce si un campo magnético uniforme gira alrededor de la espira.

El movimiento de la espira lo produce una central hidroeléctrica, térmica, eólica, o nuclear.

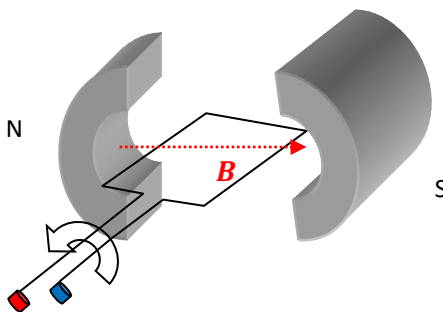


Fig.1.5

Movimiento de una espira en el interior de un campo magnético uniforme.

Cuando la espira gira, la corriente y el voltaje inducido en sus extremos varían tanto en magnitud como en dirección. Una rotación completa de la espira origina un ciclo completo del voltaje y de la corriente. La representación gráfica de este movimiento es una onda sinusoidal, con amplitud máxima cuando el plano de la espira está en paralelo con el campo magnético, y nula, cuando es perpendicular.

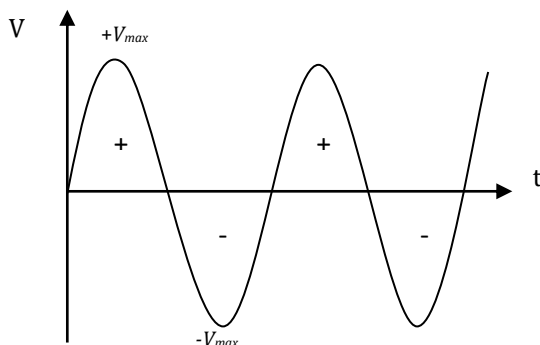


Fig.1.6

Onda sinusoidal que representa la alternancia de la polaridad del voltaje.

1.1.10. Valores típicos de una señal alterna

Una señal alterna presenta valores característicos según la amplitud de la onda y el número de ciclos que realiza la espira por unidad de tiempo.

La *fuerza electromotriz (fem)* o *voltaje alterna (V)*, en los extremos de la espira, se calcula según la siguiente relación.

$$V = V_{max} \text{sen} 2\pi f t$$

1.1.10.1. Frecuencia

La *frecuencia (f)* de una señal alterna, es la cantidad que indica el número de ciclos por segundo que efectúa una espira al girar en el interior de un campo magnético.

La unidad de medida de la frecuencia es el *hertz (Hz)*. El instrumento de medida es el **frecuencímetro**.

La red eléctrica domiciliaria e industrial es de 60 Hz, esto quiere decir, que la espira ha rotado 60 veces en un segundo.

En general, la frecuencia está en relación inversa con el *periodo (t)*.

$$f = \frac{1}{t}$$

En algunos circuitos digitales, especialmente en el generador de pulsos de reloj, se emplea la noción de frecuencia para diseñar e implementar contadores, registros y otros dispositivos secuenciales.

La velocidad del microprocesador, también se expresa en unidades de frecuencia, aunque en este caso, la velocidad se define como el número de ciclos por

segundo, una instrucción puede ser procesada en uno o varios segundos. Un valor típico de velocidad de un microprocesador actual es 3 GHz (Gigahertz) ó 3×10^9 Hz.

1.1.10.2. Valor eficaz y valor medio

El *Valor eficaz* (V_{ef}) o *rms* (*raíz media cuadrática*) de una señal alterna, es aquella corriente senoidal que produce el mismo efecto calórico al circular por una carga óhmica, que si circulara por ella una corriente continua equivalente.

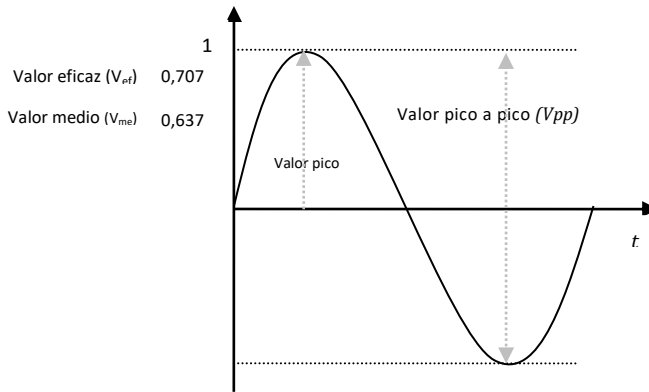


Fig.1.7

Valor máximo (valor pico), Valor pico a pico (V_{pp}), valor eficaz ó rms (root mean square) y valor medio de una señal eléctrica senoidal.

Los valores máximos son aquellos valores de voltajes y corrientes instantáneos, que se dan en determinados momentos, sin embargo, los equipos que funcionan con corriente eléctrica alterna, no trabajan con los valores máximos, sino que lo hacen con los valores eficaces.

El *Valor eficaz* (V_{ef}) y el *valor medio* (V_{med}), de la corriente y voltaje senoidal, se calcula de la siguiente manera:

$$V_{ef} = \frac{1}{\sqrt{2}} V_{max} = 0,707 V_{max}$$

$$V_{med} = \frac{2}{\pi} V_{max} = 0,637 V_{max}$$

Cuando utilizamos un voltímetro para medir la **caída de tensión** (voltaje) en los extremos de un tomacorriente de la red eléctrica (corriente alterna), lo que en realidad medimos es el *Voltaje eficaz* (V_{ef}).

Sabemos que el *voltaje eficaz* (V_{ef}) de la red domiciliaria es **~ 220V** (~ es un símbolo de señal alterna), por tanto, el *Voltaje máximo* (V_{max}) o *Voltaje pico* (V_{pp}) es igual a:

$$V_{max} = \sqrt{2} V_{ef} = (\sqrt{2})(220 \text{ V}) \approx 311 \text{ V}$$

El Voltaje *máximo* (V_{max}) de 311 V es posible medirlo y visualizarlo mediante un osciloscopio.

El símbolo de la fuente de voltaje de corriente alterna es



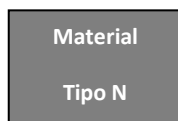
Los equipos electrónicos como la computadora, monitores, impresoras, escaneadores, entre otros, se instalan a la corriente alterna de ($\sim 220 V$), no obstante que todos ellos funcionan con corriente y voltaje continuo; esto se logra gracias a que los equipos disponen de una fuente de alimentación, cuya función es convertir la corriente alterna en corriente continua.

1.2. Dispositivos semiconductores

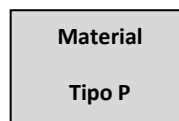
Los dispositivos que procesan las señales alternas de baja potencia como la salida de un micrófono, y las señales de la red eléctrica, se realizan convenientemente con diodos y transistores construidos con material semiconductor.

Los semiconductores, son materiales de germanio o silicio, que presentan pocos electrones libres en su banda de valencia, por tanto, si hacemos circular corriente por éste material, observaríamos que no es un buen conductor, ni tampoco, un buen aislante.

Para que sean buenos conductores de portadores de carga, a cada oblea de silicio le agregan impurezas, proceso llamado “*dopaje*”. Si lo dopan con arsénico o fósforo, se hace un buen portador de electrones (carga negativa) o **material tipo N**. Si lo impurifican con aluminio, indio o galio, el material será un buen portador de huecos (carga positiva) o **material tipo P**.



Silicio Dopado con arsénico o fósforo



Silicio Dopado con aluminio, indio o galio

La utilización de semiconductores a gran escala, para el tratamiento de señales eléctricas, se realizó a partir de 1947, cuando el transistor fue inventado por los físicos J. Bardeen, W. Brattain y W. Shockley, de los laboratorios Bell en Estados Unidos.

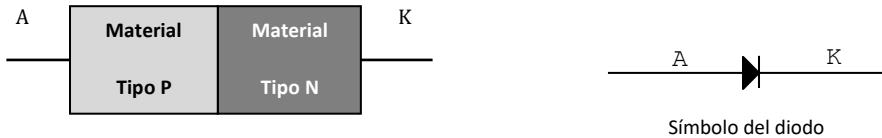
En 1958, el ingeniero norteamericano Jack Kilby, logró la integración de seis transistores en una oblea de germanio. A partir de ese momento, la integración se realizó por cientos, miles y ahora millones de transistores en una sola oblea de silicio.

Los dispositivos que integran elementos electrónicos se llaman “circuitos

integrados” o chips, los hay desde aquellos que se utilizan para la realización de prácticas de electrónica digital, hasta potentes microprocesadores, capaces de procesar cientos de miles de millones de instrucciones por segundo.

1.2.1. Diodos

Un diodo es el más sencillo de los dispositivos semiconductores, pues, se trata de la unión de dos materiales de silicio, uno tipo P y otro tipo N.



Un diodo tiene dos terminales, uno de ellos adherido al material tipo P llamado **Ánodo (A)**, y otro conectado al material tipo N llamado **Cátodo (K)**.

No obstante, su sencillez, el comportamiento eléctrico del diodo es muy valioso, pues puede funcionar como un interruptor, es decir, si se polariza convenientemente puede funcionar como un circuito cerrado o circuito abierto.

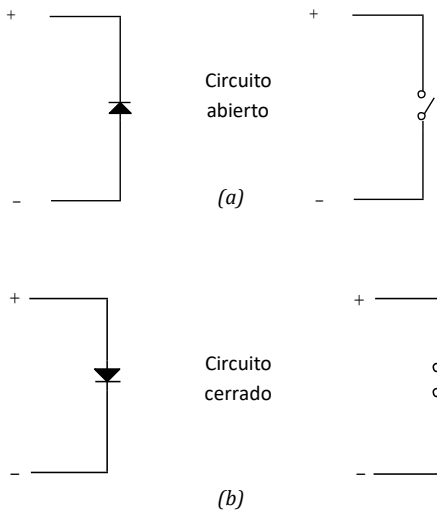


Fig.1.8

(a) Polarización inversa del diodo. Cuando el ánodo se conecta con el negativo de la fuente, y el cátodo con el positivo, se comporta como un circuito abierto.

(b) Polarización directa del diodo. Se comporta como un circuito cerrado, siempre que el ánodo se conecte al positivo y el cátodo al negativo de la fuente de poder.

Además, el diodo es el principal responsable de la conversión de corriente alterna en corriente continua. Si se disponen adecuadamente los diodos, unos polarizados directamente y otros inversamente se obtiene un **“rectificador”** para dejar pasar ondas de una señal senoidal en un solo sentido. A ésta señal se le somete a un **“filtraje”** con ayuda del proceso de carga y descarga de un condensador.

Luego de éste procedimiento, se obtiene en los extremos del circuito de corriente continua un voltaje equivalente al valor máximo de la señal alterna, es decir, si rectificamos la corriente alterna de $\sim 220\text{ V}$ (voltaje eficaz), en los terminales del circuito de corriente continua obtendremos un voltaje continuo de 311 V , este luego de un proceso de filtrado y atenuación.

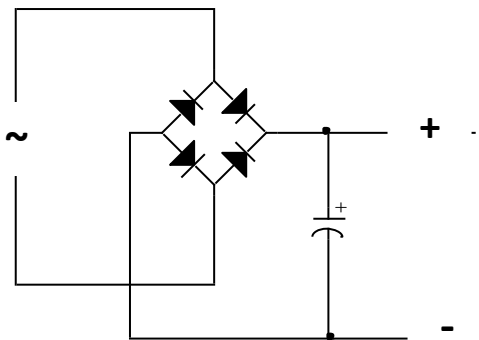


Fig.1.9

Rectificador tipo puente. La entrada recibe un voltaje de corriente alterna (\sim), mientras que la salida presenta un voltaje de corriente continua.

Lo mismo ocurre cuando rectificamos el voltaje en los extremos de un transformador reductor de $\sim 220\text{ V}$ a $\sim 9\text{ V}$. Nótese que tanto la entrada como la salida del transformador poseen corriente alterna. La entrada al rectificador es $\sim 9\text{ V}$ y la salida será aproximadamente $12,7\text{ V}$ de corriente continua.

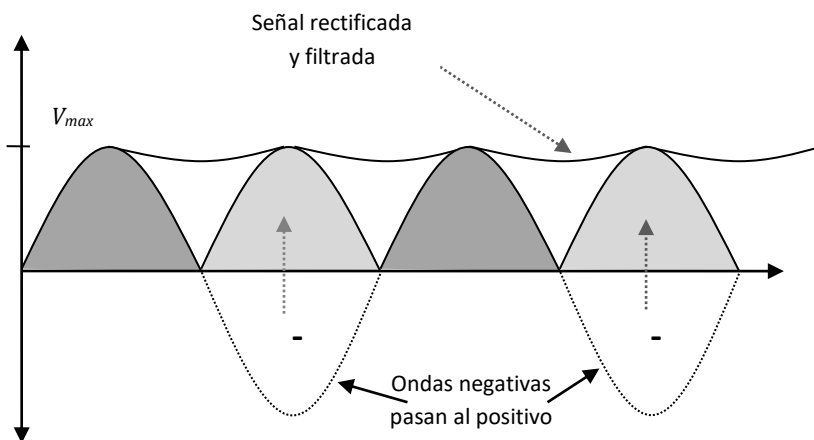


Fig.1.10

Proceso de rectificación. Las ondas negativas pasan al positivo para que el proceso de filtraje establezca la señal continua, al voltaje máximo.

Todas las fuentes de alimentación de los equipos de cómputo y demás artefactos electrónicos, cuentan con un sistema de rectificación para convertir el voltaje de corriente alterna en continua.

Existen varias clases de diodos especiales, entre los que se encuentran los diodos LED (diodos emisores de luz) y cuya importancia lo observamos en distintas aplicaciones de los circuitos electrónicos, en concreto, se utiliza para visualizar las salidas de las principales funciones lógicas en experimentación digital.

Los diodos LED, convierten la energía eléctrica en energía luminosa, ya sea en forma de luz visible o infrarroja.

Para que los diodos LED emitan energía luminosa deberá polarizarse directamente y conectarse en serie con una resistencia de 220Ω o 330Ω . Con ésta configuración, la corriente que atraviesa es del orden de los miliamperios.

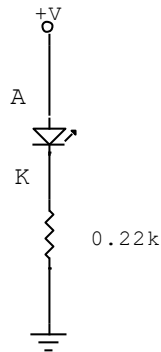


Fig.1.11

Polarización directa de un diodo LED (Light emitting diode), el ánodo conectado al positivo y el cátodo al negativo. El resistor de 220Ω puede instalarse en el terminal del ánodo o del cátodo y el efecto será el mismo.

La forma física de los diodos es muy diversa y está en relación con la aplicación y tratamiento de corriente.

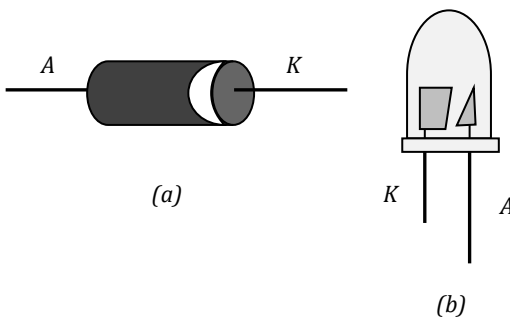


Fig.1.12

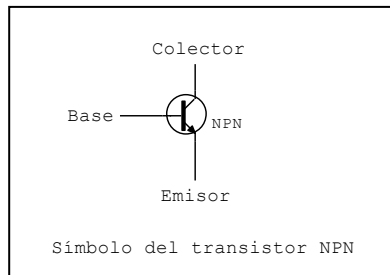
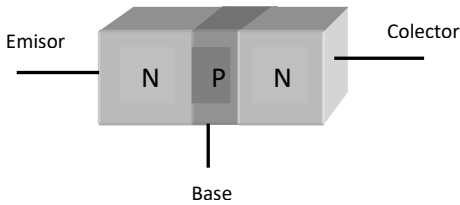
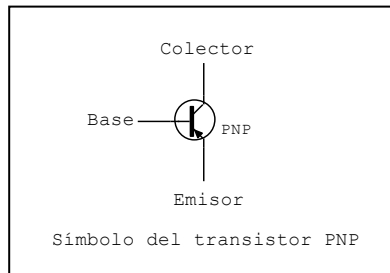
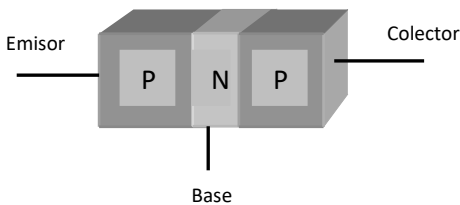
Forma física de un diodo rectificador (a) y de un diodo LED (b). Obsérvese que el cátodo en (a) se visualiza mediante una franja de color claro, mientras que en (b) el pin o patita del cátodo es más pequeño.

1.2.2. Transistores BJT

Los transistores de unión bipolar (BJT), son dispositivos semiconductores que constan de tres materiales semiconductores tipo P y tipo N.

Si el transistor presenta dos materiales tipo P y uno tipo N, entonces será del tipo PNP, en cambio, si consta de dos materiales tipo N y uno tipo P, entonces constituirá un transistor tipo NPN.

El transistor tiene tres contactos o pines llamados: **Emisor (e)**, **Base (b)** y **Colector (c)**.



Cuando se configura el transistor de tal manera que el emisor sea común tanto para el circuito de entrada como para la salida, se llama configuración **Emisor común**.

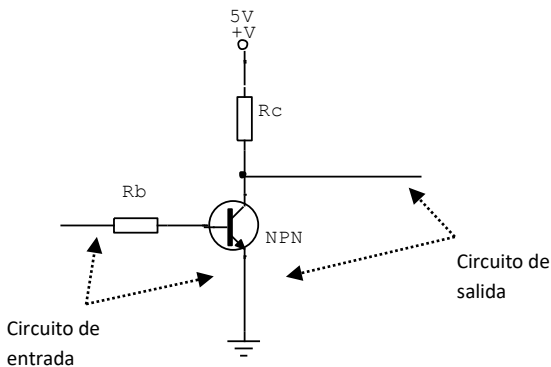


Fig.1.13

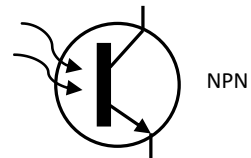
Configuración Emisor común. La Base y el Emisor forman parte de la entrada de señal, mientras que el Colector y Emisor constituyen el circuito de salida de la señal.

En el mundo de la electrónica, el transistor tiene múltiples aplicaciones, ya sea como amplificación de señales, interrupción, conmutación, regulación, etc; función que realiza tanto en el ámbito doméstico como industrial.

Muchos componentes de los equipos de cómputo utilizan un tipo especial de transistores que se activan mediante el “efecto fotoeléctrico”, llamado fototransistor. Si a éste dispositivo se acopla una fuente de luz (LED), entonces constituirá un sensor de posición, ya que, si se interrumpe el haz de luz emitido, el fototransistor se activará indicando la presencia de algo.

Los sensores de posición son muy utilizados en los discos duros, impresoras, lectoras de DVD, y otros mecanismos donde sea necesaria la activación de motores o verificar la presencia de papel, entre otros.

El símbolo de un fototransistor es el siguiente



En electrónica digital, el transistor se utiliza básicamente como interruptor, puesto que, se necesita representar físicamente los dos estados de una variable lógica, el cero y el uno. El **cero** constituye un *circuito abierto*, mientras que el **uno** implica un *circuito cerrado*.

Según la tecnología de fabricación, los transistores se clasifican en dos: transistores bipolares (BJT) y transistores de efecto de campo (FET).

La tecnología MOSFET, una variante mejorada de los transistores FET, permiten una mejor integración en los circuitos integrados, por tal motivo, en la construcción de circuitos integrados digitales, básicamente utilizan ambas tecnologías: **TTL** (transistor transistor Lógico) y los **CMOS** (semiconductor metal oxido complementario).

1.2.2.1. Zonas de trabajo del transistor

El funcionamiento del transistor se efectúa en tres regiones: activa, corte y saturación. Estas zonas de trabajo se determinan teniendo en cuenta la corriente de *Colector* (I_c) y la caída de *tensión entre el Colector y Emisor* (V_{CE}).

El transistor trabaja en la **región activa** ó modo de operación “lineal” cuando es necesario tratar una señal débil y permitir su respectiva amplificación en función a la ganancia del dispositivo.

Lo hace en la **región de corte y saturación** sólo cuando se requiere que el transistor funcione como una variable lógica ó como circuito de conmutación. Este modo de trabajo se conoce como “no lineal”.

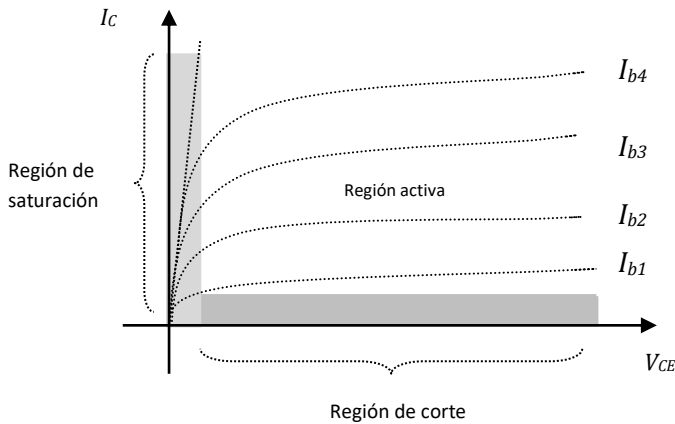


Fig.1.14

Zonas de trabajo del transistor. Nótese que en la región de corte la corriente de Colector es muy pequeña, mientras que en la región de saturación es muy alta. La corriente de Colector es correspondiente con la corriente de Base.

Un transistor trabaja en zona de corte cuando la corriente de base es aproximadamente cero. El voltaje entre Colector y Emisor es aproximadamente el voltaje de la fuente. Esta situación implica que la corriente de Colector sea aproximadamente cero, y la caída de tensión en el resistor de Colector también sea casi nula.

En cambio, para un transistor que trabaja en la zona de saturación, circulará por la Base una corriente de Base máxima, lo que conlleva a que la corriente de Colector sea también máxima. Cuando ocurre éste modo de trabajo, la caída de tensión en el resistor de Colector será aproximadamente el voltaje de la fuente de poder, y la caída de voltaje entre Colector y Emisor será aproximadamente cero.

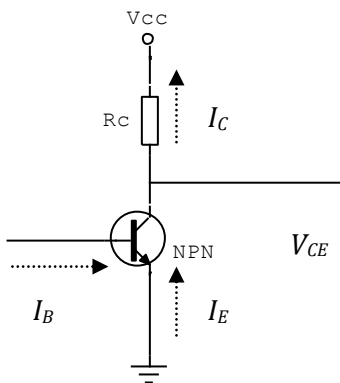


Fig.1.15

Ley de Kirchoff para las corrientes se determina así: La corriente de Emisor es igual a la suma de la corriente de Base con la de Colector. Asimismo, la ley de mallas de Kirchoff, se verifica del siguiente modo: El voltaje de fuente es igual a la suma del voltaje entre Colector y Emisor, más el producto de la corriente de Colector y la resistencia de Colector.

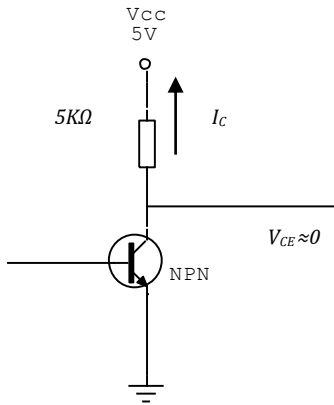
Para analizar estos dos estados o zonas de trabajo, sólo lo haremos en la malla Colector Emisor, asumiendo que en la malla Base Emisor se dan todas las condiciones necesarias para llevar al transistor a corte o saturación.

En la malla Colector Emisor, el voltaje de salida $\equiv V_{CE}$, tenemos:

$$V_{CC} = I_C R_C + V_{CE}$$

Ejemplo: Calcule la corriente de Colector (I_C) que circula por $R_C = 5K\Omega$, si el transistor trabaja en saturación y cuya la fuente de energía (V_{CC}) es +5V.

Cuando el transistor trabaja en la región de saturación $V_{CE} = 0$



$$V_{CC} = I_C R_C$$

$$I_C = \frac{5V}{5\ 000\Omega}$$

$$I_C = 1mA$$

La corriente máxima de Colector que circulará por la malla es 1mA.

En términos reales la región de saturación trabajará con corrientes de Colector ligeramente menores a 1mA, produciéndose a su vez un incremento en el voltaje Colector Emisor.

De igual manera, en la región de corte, la corriente de Colector no será necesariamente cero, sino ligeramente superior, lo que conlleva la presencia de una pequeña caída de tensión en la resistencia de Colector, lo que a su vez disminuye ligeramente el voltaje Colector Emisor.

1.2.3. Amplificadores operacionales

El amplificador operacional (amp op), es un circuito amplificador de alta **ganancia (A_o)** cuya respuesta a una señal de entrada se controla mediante un enlace de retroalimentación.

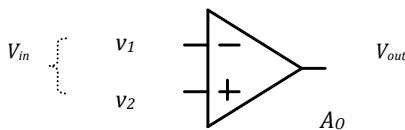


Fig.1.16

Símbolo del amplificador operacional.

En general, el **voltaje de salida (V_{out})** depende del **voltaje de entrada (V_{in})** y de la función de transferencia o **ganancia (A_o)**.

$$V_{out} = A_O V_{in}$$

Las entradas v_1 y v_2 se llaman terminales de entrada diferencial, pues el voltaje de entrada v_i , es la diferencia de voltajes entre ellas.

$$V_{out} = A_O (v_2 - v_1)$$

La entrada rotulada con el signo (-) se denomina *terminal inversora*, mientras que la designada con el signo (+), *terminal no inversora*.

Los valores típicos de un amplificador operacional están en base a la estructura interna del mismo.

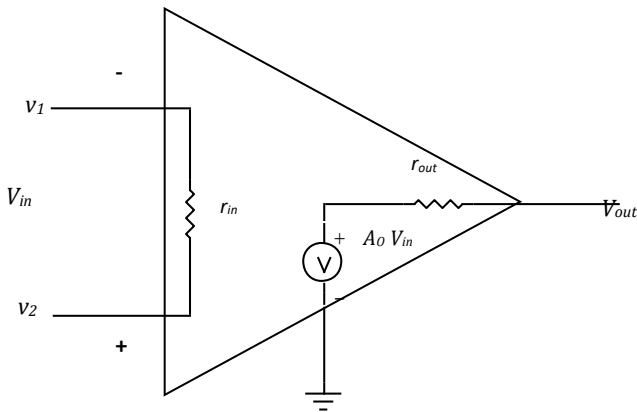


Fig.1.17

Circuito equivalente del amplificador operacional.

- La resistencia de entrada (r_{in}) es muy alta, mayor que 100 k Ω ; por tanto, la corriente que circula por ella es muy pequeña, aproximadamente, cero. Idealmente, no fluye corriente.
- La ganancia (A_O) en lazo abierto (sin retroalimentación) es bastante alta, superior a 100 000. Idealmente se considera infinito.
- La resistencia de salida es muy baja, inferior a 100 Ω . Idealmente se considera cero.
- Idealmente, el voltaje de salida (V_{out}) es cero cuando el voltaje de entrada (V_{in}) es cero. Característica denominada "Tensión de salida sin offset". Sin embargo, las características reales del amplificador originan un voltaje en la salida cuando el voltaje de entrada es cero. Esta diferencia de potencial se denomina "voltaje de desnivel". Algunos amplificadores disponen de terminales para anular dicho voltaje.
- Ancho de banda BW es cero. El amplificador en lazo abierto, la ganancia varía con la frecuencia de la señal, ocasionando que el dispositivo trabaje en un intervalo finito de frecuencias.

Configuraciones del amplificador operacional

Las configuraciones que se pueden realizar con el amplificador son tres:

- Configuración inversora
- Configuración no inversora
- Configuración diferencial

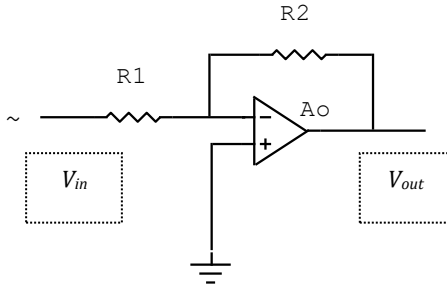


Fig.1.18

Configuración inversora

$$V_{out} = -V_{in} \frac{R2}{R1}$$

La ganancia en lazo cerrado (A_{LC}), está expresado por $\frac{R2}{R1}$; y el signo negativo revela la propiedad de inversión del circuito.

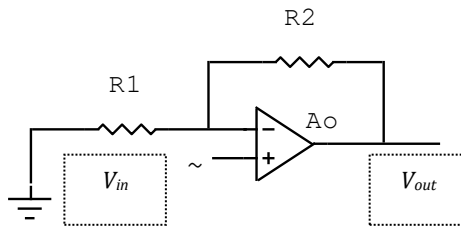


Fig.1.19

Configuración no inversora.

$$V_{out} = V_{in} \left(1 + \frac{R2}{R1}\right)$$

La ganancia de esta configuración está expresado por $\left(1 + \frac{R2}{R1}\right)$; en donde se observa que si $R2$ es cero, la ganancia es la unidad.

Asimismo, se verifica la propiedad de no inversión de la señal de entrada.

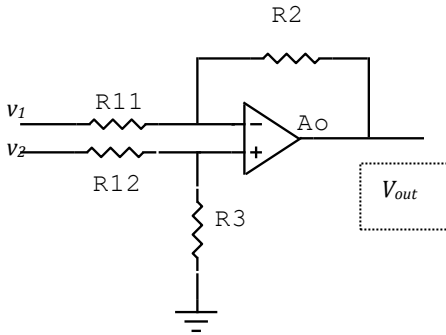


Fig.1.20

Configuración diferencial.

$$V_{out} = \frac{R3}{R12} v2 - \frac{R2}{R11} v1$$

El amplificador operacional cuenta con dos terminales de alimentación (fuente positiva y fuente negativa), aunque también se puede instalar una fuente positiva; dependiendo si en la salida queremos una señal con ondas positivas y negativas o sólo obtener ondas positivas.

Todos los circuitos amplificadores con operacionales se basan en las tres configuraciones anteriores.

En lo que se refiere a las aplicaciones de los amplificadores operacionales, el desarrollo vertiginoso de la computadora no marcó la desaparición de los amplificadores operacionales, al contrario, han logrado establecerse en diversos campos de importancia, desde la adecuación de señales de sensores de temperatura, luz, presión, etc. hasta controles de diversos componentes de envergadura industrial, bioingeniería o uso militar.

1.3. Señales analógicas y digitales

El mundo físico es analógico, pues, las principales magnitudes como la electricidad, presión, temperatura, luminosidad, entre otros, toman un valor dentro de un rango de posibilidades, mientras que, las magnitudes digitales no pueden adoptar cualquier valor, sino que, admiten sólo algunos valores enteros.

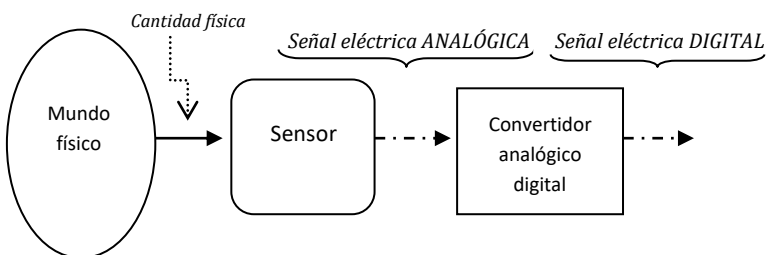


Fig.1.21

Proceso de conversión de la señal analógica en digital

1.3.1. Señales analógicas

Es aquella señal eléctrica proporcionada por el sensor o “transductor”, cuya característica es la variabilidad continua de sus valores.

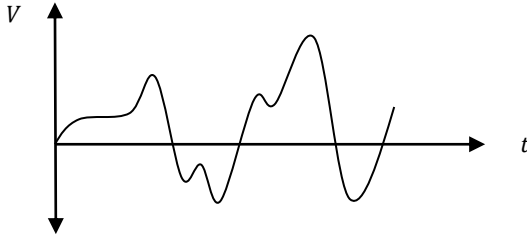


Fig.1.22

Señal eléctrica suministrada por el transductor

1.3.2. Señales digitales

Son señales eléctricas provenientes del convertidor analógico digital, luego de procesar la señal analógica. Se caracteriza porque adopta valores discretos de una posibilidad restringida.

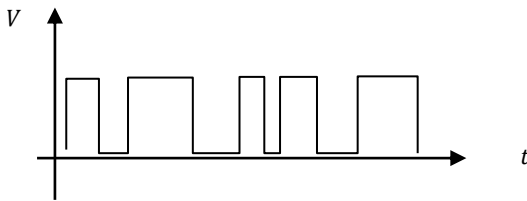


Fig.1.23

Señal eléctrica suministrada por el convertidor analógico digital.

1.3.3. Lógica digital

En los sistemas de cómputo se asume un NIVEL BAJO como CERO y un NIVEL ALTO como UNO. En los aspectos prácticos, sin embargo, los circuitos electrónicos digitales consideran un rango de voltajes y los asumen como UNO o como CERO.

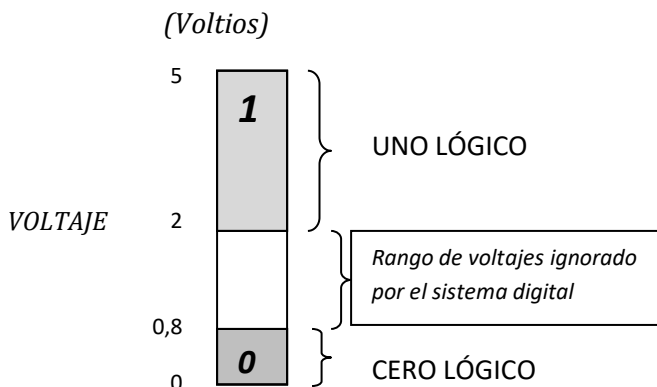


Fig.1.24

Niveles de tensión que asignan el valor lógico UNO y CERO.

1.3.4. Lógica positiva y lógica negativa

En electrónica digital es usual trabajar con lógica de voltajes. Si se asume que el nivel de voltaje alto o más positivo corresponde al UNO lógico y el nivel de voltaje bajo o menos positivo, al CERO lógico, se denomina “lógica positiva”.

Sin embargo, si el nivel de voltaje alto corresponde al CERO lógico y el nivel de voltaje bajo al UNO lógico, se designa, “lógica negativa”.

Problemas Resueltos N° 01

Para resolver los problemas planteados a continuación se debe tener en cuenta los prefijos de los múltiplos y submúltiplos de las cantidades físicas.

<u>Múltiplos</u>			<u>Sub múltiplos</u>		
Prefijo	Valor numérico	símbolo	Prefijo	Valor numérico	símbolo
Tera	10^{12}	T	pico	10^{-12}	p
Giga	10^9	G	nano	10^{-9}	n
Mega	10^6	M	micro	10^{-6}	μ
Kilo	10^3	K	mili	10^{-3}	m

Unidad $10^0 = 1$

1. Realizar las conversiones de las siguientes cantidades:

a. 85 milivoltios (mV) a voltios (V).

Solución:

Se expresa 85 mV en potencia de 10

$$85 \times 10^{-3} V = 0,085 V$$

Que es lo mismo dividir 85 entre 1000, ó correr la coma decimal tres posiciones a la izquierda de 85.

b. 635 microamperios (μA) a amperios (A).

Solución:

635 μA expresado en potencia de 10.

$$635 \times 10^{-6} A = 0,000635 A$$

Es decir, 635 entre 1 000 000, ó correr la coma decimal seis posiciones a la izquierda de 635.

- c. 0,048 Watts (W) a miliwatts (mW).

Solución:

0.048 W se expresa en potencia de 10, ya que 10^{-3} es el prefijo (mili).

$0,048 W = 0,048 \times 10^{-3} \times 10^3$, nótese que también se multiplicó por 10^3 para que la expresión inicial no cambie.

$$(0,048 \times 10^3) \times 10^{-3} W = 48 \times 10^{-3} W = 48 mW$$

- d. 150000 Ohmios (Ω) a megaohmios (M Ω).

Solución:

150 000 Ω se expresa en potencia de 10, pues 10^6 es prefijo de (mega).

$$150\ 000\ \Omega = 150\ 000 \times 10^6 \times 10^{-6}\ \Omega$$

$$\frac{150\ 000}{1\ 000\ 000} \times 10^6 \Omega = 0,15 \times 10^6 \Omega = 0,15\ M\Omega$$

- e. 80 000 milicoulombs (mC) a coulombs (C).

Solución:

80 000 mC se expresa en potencia de 10.

$$80\ 000 \times 10^{-3} C = \frac{80\ 000}{1000} C = 80 C$$

- f. 6 amperios (A) a miliamperios (mA).

Solución:

$6 A = 6 \times 10^{-3} \times 10^3 A$; en este caso, 6 A se multiplicó por 10^{-3} para obtener el prefijo mili, y también se multiplicó por 10^3 para mantener los 6 A.

$$6 \times 10^3 \times 10^{-3} A = 6\ 000 \times 10^{-3} A = 6\ 000 mA$$

2. Determinar la energía potencial eléctrica de una batería utilizada en equipos de energía solar el cual tiene las siguientes especificaciones: 12 V, con 110 amp-hr.

Solución:

Frecuentemente, la cantidad de carga de una batería se expresa mediante el producto amp-hr (amperios)(horas).

De la definición de corriente eléctrica $I = dQ/dt$, tenemos: $Q = (I)(t)$, es decir, $Q = 110\ amp - hr = 110 A \times 3600s = 396\ 000 C$; pues 1hr es igual a 3 600 s.

De la definición de voltaje tenemos: $\Delta U = V \times Q = (12 V)(396\ 000 C) = 4\ 752\ 000\ joulios = 4,75 M$

3. Un monitor LCD de 20" durante su funcionamiento consume 144 000 joulios de energía durante una hora. ¿Cuál es la potencia de dicho monitor?

Solución:

La potencia se calcula: $P = W/t$, donde W se expresa en joulios y t en segundos.

Reemplazando valores tenemos:

$$P = \frac{144\,000\text{ J}}{3\,600\text{ s}} = 40\text{ W}$$

4. ¿Cuánta corriente circula por una motobomba de 0,5 HP de potencia, si se sabe que el voltaje de línea es ~ 220 V?

Solución:

Normalmente la potencia de los motores eléctricos se expresa en Horse Power (HP), cuya equivalencia en Watts (W) es la siguiente: 1 HP = 746 W.

Como la motobomba se conecta al voltaje alterno de 220 V, la corriente se calcula de la siguiente manera: $I = P/V$

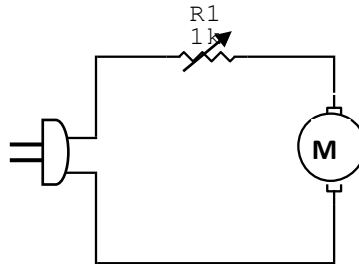
$$I = \frac{(0,5)(746)\text{ W}}{220\text{ V}} = 1,69\text{ A}$$

5. A la línea de voltaje (220 V) se conecta un motor de 440 W a un circuito de control de celeridad, cuya resistencia variable se encuentra en 176 Ω.
- ¿Qué corriente circula por el motor?
 - ¿Cuál es la corriente máxima que puede circular por el motor?
 - ¿Cuál es el valor mínimo de la resistencia variable para que la corriente sea máxima?

Solución:

La resistencia variable de éste circuito representa el control de celeridad de un motor. A menor resistencia, mayor celeridad (velocidad) del motor.

- a. Para saber cuánta corriente está circulando por el motor, se calcula la corriente que circula por el circuito serie, sin considerar la resistencia interna del motor, sólo el valor de la resistencia variable.



$$I = \frac{\text{voltaje de línea}}{\text{resistencia variable}} = \frac{220\text{ V}}{176\ \Omega} = 1,25\text{ A}$$

- b. Para saber cuánta corriente puede circular por el motor, se toma en cuenta la potencia del motor y el voltaje de línea.

$$I = \frac{\text{potencia del motor}}{\text{voltaje de línea}} = \frac{440\text{ W}}{220\text{ V}} = 2\text{ A}$$

- c. Para calcular el valor de la resistencia variable donde la corriente sea máxima, sólo tenemos que dividir el voltaje de línea entre la corriente máxima que

puede circular por el motor.

$$R = \frac{\text{voltaje de línea}}{\text{corriente máxima}} = \frac{220 \text{ V}}{2 \text{ A}} = 110 \Omega$$

Observación: si la resistencia variable es menor de 110 Ω , entonces circulará mayor corriente, es decir, mayor de 2 A, circunstancia que dañará al motor.

6. En el siguiente circuito, calcular:

- Resistencia total (equivalente serie)
- Corriente total
- Potencia disipada por R3

Solución:

- Calculamos la resistencia total. Como el circuito es serie, se suman todas sus resistencias.

$$R_t = 8 \Omega + 10 \Omega + 12 \Omega = 30 \Omega$$

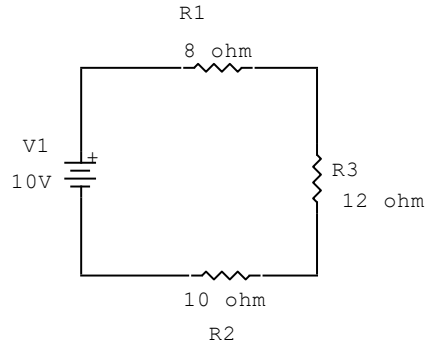
- Calculando la corriente total.

$$I = \frac{10 \text{ V}}{30 \Omega} = 0,333 \text{ A} = 333 \text{ mA}$$

- Calculando la potencia disipada por la resistencia R3. Se sabe que $P = VxI$, y que $V = IxR$; reemplazando en la ecuación anterior tenemos:

$$P = (IxR)xI = I^2R$$

$$P = (333 \text{ mA})^2(12 \Omega) = (333x10^{-3} \text{ A})^2(12 \Omega) = 3996x10^{-3} \text{ W} = 3,99 \text{ mW}$$



7. En el siguiente circuito paralelo, calcular:

- Resistencia equivalente.
- Corriente total.

Solución:

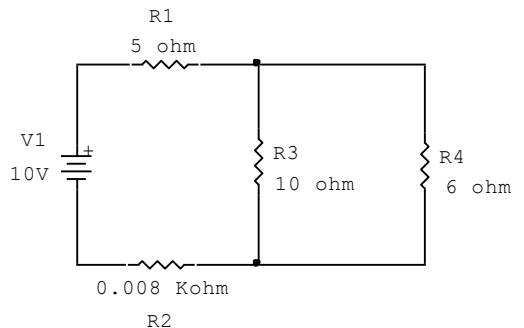
- En primer lugar, se observa que R3 y R4 están en paralelo, por lo tanto, encontramos la resistencia equivalente en paralelo.

$$\frac{1}{Req} = \frac{1}{10\Omega} + \frac{1}{6\Omega} = \frac{4}{15\Omega}$$

$$Req = \frac{15}{4} = 3,75 \Omega$$

Las resistencias R1, R2 y Req están en serie, por tanto:

$$R_t = 5\Omega + 0,008x10^3\Omega + 3,75 \Omega = 16,75 \Omega$$



b. La corriente total que circula por el circuito es:

$$I = \frac{10 \text{ V}}{16,75} = 0,597 \text{ A} = 597 \text{ mA}$$

8. Las resistencias del siguiente circuito disipan las correspondientes potencias:

R1: 3 W

R2: 4 W

R3: 5 W

a. Hallar el voltaje de la fuente del circuito.

b. Hallar la corriente total del circuito.

Solución:

a. Encontramos la caída de voltaje en cada una de las resistencias, ya que el voltaje total del circuito es la suma de los voltajes que caen en cada resistencia.

Si reemplazamos $I = V/R$ en $P = VxI$, tenemos:

$$P = V \left(\frac{V}{R} \right) = \frac{V^2}{R}$$

Voltaje en R1:

$$V1 = \sqrt{P1 \cdot R1} = \sqrt{(3W)(12\Omega)} = 6 \text{ V}$$

Voltaje en R2:

$$V2 = \sqrt{P2 \cdot R2} = \sqrt{(4W)(16\Omega)} = 8 \text{ V}$$

Voltaje en R3:

$$V3 = \sqrt{P3 \cdot R3} = \sqrt{(5W)(20\Omega)} = 10 \text{ V}$$

El voltaje total (Vt) se calcula así:

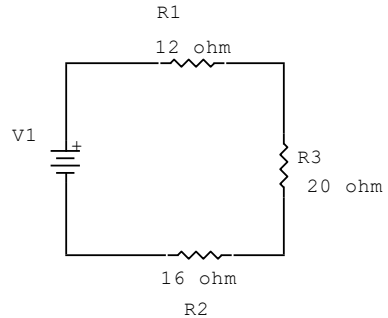
$$Vt = V1 + V2 + V3 = 6 \text{ V} + 8 \text{ V} + 10 \text{ V} = 24 \text{ V}$$

b. Como el circuito es serie, la resistencia total es:

$$Rt = 12 \Omega + 16 \Omega + 20 \Omega = 48 \Omega$$

La corriente total es:

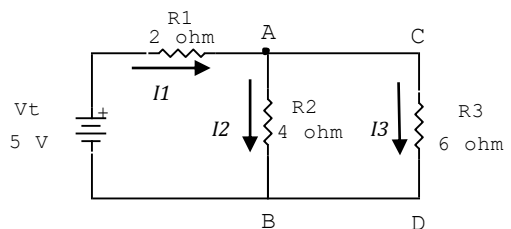
$$I = \frac{24 \text{ V}}{48 \Omega} = 0,5 \text{ A} = 500 \text{ mA}$$



9. En el circuito de la figura calcular la corriente $I3$.

Solución:

Se observa que los puntos A y C tienen la misma diferencia de potencial respecto a B y C que están



a un potencial eléctrico de 0 V. Por lo tanto, decimos que el voltaje que cae en R_2 es igual al voltaje que cae en R_3 .

$$VR_2 = VR_3$$

$$I_2R_2 = I_3R_3$$

Entonces el voltaje total del circuito se distribuye de la siguiente manera:

$$V_t = VR_1 + VR_2$$

$$V_t = I_1R_1 + VR_2$$

$$VR_2 = V_t - I_1R_1 = I_3R_3$$

$$I_3 = \frac{V_t - I_1R_1}{R_3} \quad (*)$$

Para calcular I_3 , tenemos que determinar I_1 .

Se observa también que la corriente I_1 (corriente total) se reparte en el nodo A, en I_2 e I_3 .

$$I_1 = I_2 + I_3,$$

Para calcular I_1 , se debe hallar primero la Resistencia total del circuito, donde R_2 y R_3 están en paralelo, y éstos están en serie con R_1 .

$$R_t = R_1 + R_{eq}$$

$$\frac{1}{R_{eq}} = \frac{1}{R_2} + \frac{1}{R_3} = \frac{R_2 + R_3}{R_2R_3}$$

$$R_t = R_1 + \frac{R_2R_3}{R_2 + R_3}$$

$$R_t = \frac{R_1R_2 + R_1R_3 + R_2R_3}{R_2 + R_3}$$

$$R_t = \frac{((2)(4) + (2)(6) + (4)(6))\Omega^2}{(4 + 6)\Omega} = \frac{22}{5}\Omega = 4,4\Omega$$

Entonces la corriente total I_1 .

$$I_1 = \frac{V_t}{R_t} = \frac{5\text{ V}}{4,4\Omega} = 1,14\text{ A}$$

Reemplazando I_1 en (*) tenemos:

$$I_3 = \frac{5\text{ V} - (1,14\text{ A})(2\Omega)}{6\Omega} = 0,453\text{ A} = 453\text{mA}$$

10. En el circuito rectificador de media onda cuya entrada es 6 V (rms), a 60 Hz y una resistencia de 5 k Ω , encontrar:

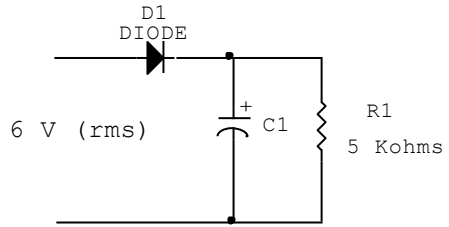
- El periodo de la señal eléctrica.
- El voltaje de caída (DC) en el resistor.
- La corriente máxima que circula por el diodo.

Solución:

- a. El periodo se calcula con el inverso de la frecuencia.

$$t = \frac{1}{f} = \frac{1}{60 \text{ Hz}} = 0,0167 \text{ s} = 16,7 \text{ ms}$$

- b. El voltaje que cae en el resistor de 5 K Ω es voltaje de corriente continua, pues, el diodo D1 rectifica media onda de la señal eléctrica de entrada. El condensador C1 contribuye en mantener la media onda rectificadora a su Valor máximo.



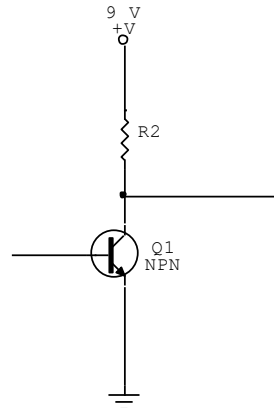
$$\text{Voltaje Mximo } (V_{mx}) = (\sqrt{2})(V_{ef}) = 6\sqrt{2} = 8,48 \text{ V}$$

- c. La corriente mxima que circula por el diodo es:

$$I = \frac{8,48 \text{ V}}{5 \times 10^3 \Omega} = 0,001696 \text{ A} = 1,7 \text{ mA}$$

11. Un transistor trabajando en la zona de saturaci3n presenta una corriente de colector de 750 μA . Si el voltaje de fuente (V_{cc}) es 9 V.

- a. Calcular la Resistencia de colector (R_c).
- b. Si luego se reduce el voltaje (V_{cc}) a 5 V. Calcular la corriente de colector (I_c) en el que el transistor se encuentre en la zona de saturaci3n.
- c. Con el voltaje de fuente anterior, calcular la corriente a partir del cual el transistor comienza a trabajar en la zona de corte.



Soluci3n:

- a. Como el transistor est trabajando en la zona de saturaci3n, el voltaje colector emisor (V_{ce}) es aproximadamente cero. De la ecuaci3n.

$$V_{cc} = I_c R_c + V_{ce}$$

Se calcula la resistencia de colector (R_c).

$$R_c = \frac{V_{cc}}{I_c} = \frac{9 \text{ V}}{750 \times 10^{-6} \text{ A}} = 12 \text{ K}\Omega$$

- b. La resistencia de colector (R_c) se mantiene constante, por tanto, si el voltaje de fuente (V_{cc}) se reduce, la corriente de colector cuando $V_{ce} = 0$, (ver fig. 1.24) ser:

$$I_c = \frac{5 V}{12 \times 10^3 \Omega} = 0,00042 A = 420 \mu A$$

Si el V_{ce} es como máximo 0,8 V, el transistor se mantendrá en la zona de saturación. Por tanto, tenemos:

$$I_c = \frac{(5 - 0,8)V}{12 \times 10^3 \Omega} = 350 \mu A$$

- c. El transistor comienza a trabajar en la zona de corte cuando el voltaje colector emisor (V_{ce}) es como mínimo 2 V.

Por tanto:

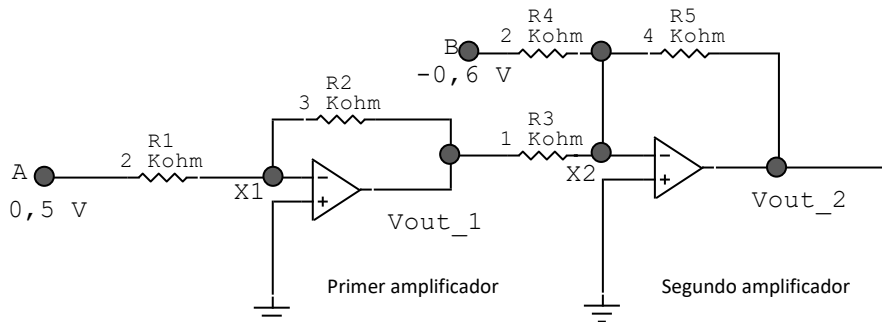
$$V_{cc} = I_c R_c + V_{ce}; \quad 5 V = I_c (12 \times 10^3 \Omega) + 2 V$$

$$I_c = \frac{3 V}{12 \times 10^3 \Omega} = 250 \mu A$$

A partir de 250 μA , el transistor ingresa a la zona de corte. Se mantendrá en ésta zona si la corriente de colector I_c varía hasta hacerse cero.

OBSERVACIÓN: Una corriente de colector entre 250 μA y 350 μA , colocará al transistor en una zona ambigua, no reconocible para un sistema digital.

12. Encontrar V_{out_1} y V_{out_2} del siguiente amplificador operacional con configuración inversora.



Solución:

Evaluando corrientes y voltajes en el primer amplificador.

El punto X1 y X2 se encuentran a un potencial de 0V, puesto que la terminal no inversora se encuentra conectada a tierra.

La diferencia de potencial entre A y X1 es:

$$\Delta V = \text{Voltaje en A} - \text{Voltaje en X1} = 0,5 V - 0 V$$

La diferencia de potencial entre X1 y V_{out_1} es:

$$\Delta V = \text{Voltaje en X1} - \text{Voltaje en } V_{out_1} = 0 V - V_{out_1}$$

Se observa que la corriente entre el punto A y X1 (I_1) es igual que la corriente entre X1 y Vout_1 (I_2), ya que se trata de un circuito serie.

$$I_1 = I_2$$

$$\frac{0,5 V - 0V}{2 K\Omega} = \frac{0 V - V_{out_1}}{3 K\Omega}$$

$$V_{out_1} = -(3) \frac{0,5 V}{2} = -0,75 V$$

Evaluando corrientes y voltajes en el segundo amplificador.

La diferencia de potencial entre Vout_1 y X2 es:

$$\Delta V = \text{Voltaje en } V_{out_1} - \text{Voltaje en } X2 = -07,5 V - 0 V$$

La diferencia de potencial entre B y X2 es:

$$\Delta V = \text{Voltaje en } B - \text{Voltaje en } X2 = -0,6 V - 0V$$

La diferencia de potencial entre X2 y Vout_2 es:

$$\Delta V = \text{Voltaje en } X2 - \text{Voltaje en } V_{out_2} = 0 V - V_{out_2}$$

La corriente que circula entre los puntos Vout_1 y X2 es (I_3).

La corriente que circula entre los puntos B y X2 es (I_4).

La corriente que circula entre los puntos X2 y Vout_2 (I_5)

Tenemos:

$$I_3 + I_4 = I_5$$

$$\frac{-0,75 V - 0V}{1 K\Omega} + \frac{-0,6 V - 0V}{2 K\Omega} = \frac{0V - V_{out_2}}{4 K\Omega}$$

$$\frac{-0,75 V}{1 K\Omega} + \frac{-0,6 V}{2 K\Omega} = \frac{-V_{out_2}}{4 K\Omega}$$

$$V_{out_2} = 4,2 V$$

Problemas Propuestos N° 01

1. Realizar las conversiones de las siguientes cantidades:
 - a. 659 voltios (V) a kilovoltios (KV).
 - b. 87 microamperios (μA) a miliamperios (mA).
 - c. 3 400 watts (W) a kilowatts (KW).
 - d. 2 800 Kilohmios ($\text{K}\Omega$) a megaohmios ($\text{M}\Omega$).

2. ¿Cuál es el valor de la potencia de un aparato eléctrico que convierte 1240 julios de energía en 10 segundos? *Rpta: 124 W.*

3. ¿Qué cantidad de energía necesita para que funcione una computadora de 600 watts durante 3 horas? *Rpta: 6480000 J*

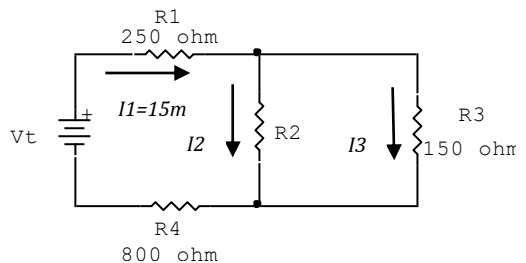
4. Determinar la diferencia de potencial entre los bornes de una batería, la cual provee de 1 620 000 joulios de energía potencial eléctrica y 50 amp-hr. *Rpta: 9V.*

5. Una impresora de inyección de tinta de 0,035 KW de potencia, durante su funcionamiento consume 420 000 J de energía cuando circula 3 000 C de carga. Encontrar la corriente de funcionamiento. *Rpta: 250 mA*

6. La fuente de alimentación de 600 W de una computadora provee en su salida, 50 A en la línea de 5V (cable de color rojo). ¿Qué porcentaje de la potencia total es la potencia de la línea de 5 V? *Rpta: 45,5%*

7. Un motor de paso de disco duro funciona con 12 V ¿Cuál es la potencia del motor si sus bobinas tienen una resistencia de $6,2 \Omega$? ¿Cuál será el voltaje mínimo que se puede aplicar a dicho motor si la corriente mínima que requiere para funcionar es 1,45 A?. Tenga en cuenta que la resistencia de las bobinas es constante. *Rpta: 23,23 W; 9 V.*

8. En el siguiente circuito, calcular:
 - a. La caída de voltaje en la resistencia $R4$.
 - b. El voltaje total teniendo en cuenta que $I3$ es $1/3$ de $I1$.
 - c. La resistencia $R2$.
 - d. La potencia total de la fuente de poder.



Rpta: a) 12 V; b) 18 V; c) 225 Ω ; d) 270 mW.

9. Evaluar la zona de trabajo de un transistor (V_{ce}), cuando $V_{cc}=5V$, $R_c= 10K\Omega$ y
- Cuando $I_c = 0,18$ mA; Rpta: Zona de corte.
 - Cuando $I_c = 0,45$ mA; Rpta: Zona de saturación.

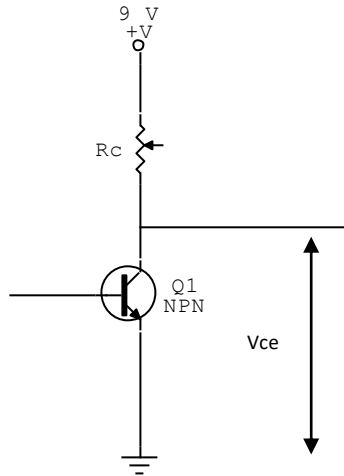
10. ¿Entre qué valores deberá encontrarse la resistencia variable R_c , para que el transistor trabaje en la zona de corte, si se requiere una corriente constante de 350 mA? Considere que la zona de corte puede mantener a V_{ce} entre 2V y 5V.

¿Qué valor debe tener R_c para que el transistor ingrese a la zona de saturación?

Rpta:

R_c deberá oscilar entre 11,43 K Ω hasta 20,00 K Ω . para que el transistor trabaje en la zona de corte.

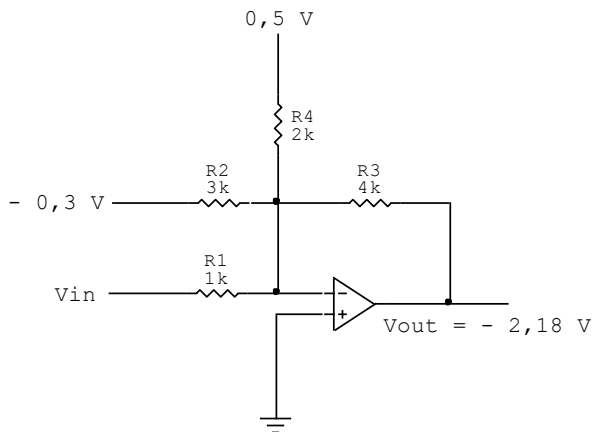
Para que el transistor ingrese a la zona de saturación, la resistencia de colector deberá oscilar entre 23 429 Ω y 25 714 Ω .



11. Encontrar el voltaje V_{in} , en el amplificador operacional de la figura.

Rpta:

$V_{in} = 395$ mV



Capítulo 2

Sistemas y códigos de numeración

2.1. Representación de datos

Las cantidades físicas del mundo exterior son percibidas por sensores o transductores adecuados convenientemente. Dichas cantidades son convertidas en energía eléctrica y luego de un proceso de acondicionamiento son proyectadas en forma de señales, las que deberán organizarse como datos o cantidades numéricas, utilizando un sistema de numeración.

El sistema de numeración decimal es el predilecto para expresar y sistematizar nuestros datos, porque nos es familiar y porque con el podemos realizar un sin número de actividades y transacciones entendible por todos.

Sin embargo, en el mundo de las computadoras, la amplia variedad de datos se representa mediante valores numéricos discretos que adoptan uno de solo dos valores posibles.

Los valores 0 y 1 concuerdan con la naturaleza binaria de la lógica digital y representan lo que ocurre físicamente en el interior del computador, pues dentro de éste, sólo hallaremos corrientes y voltajes que fluctúan en el rango de dos niveles establecidos.

Para relacionar la parte física con la parte lógica de un computador se hará un breve análisis del sistema de numeración binario y su relación con otros sistemas de numeración.

Asimismo, se verá la importancia de los códigos de numeración en el análisis e implementación de circuitos digitales.

2.1.1. Representación de números, sistemas de numeración

Un número cualquiera, sea entero o fraccionario, se puede representar en cualquier sistema de numeración, para tal efecto se tendrá en cuenta la base y los dígitos o cifras que pertenecen al sistema.

Generalmente, la representación de cualquier número se efectúa en notación polinómica, donde cada término se expresa mediante el producto de la potencia de la base por un dígito del sistema.

Los sistemas de numeración son sistemas de valor posicional, pues, el valor de sus cifras depende de la posición donde se encuentren. Los pesos de los órdenes están expresados mediante las potencias de la base del sistema.

En cualquier sistema de numeración se deberá tener en cuenta que el dígito debe ser menor que la base, nunca igual o mayor.

En el sistema decimal, la base es 10, porque diez unidades de un orden forman una unidad del orden inmediato superior. Los diez dígitos decimales son: 0, 1, 2, 3, 4, 5, 6, 7, 8, y 9.

El número **7514,392** en polinomio de potencias se representará así:

$$7514,392 = \underbrace{7 \times 10^3 + 5 \times 10^2 + 1 \times 10^1 + 4 \times 10^0}_{\text{Parte entera}} + \underbrace{3 \times 10^{-1} + 9 \times 10^{-2} + 2 \times 10^{-3}}_{\text{Parte fraccionaria}}$$

Nótese que la parte entera del número corresponde a los términos cuyos exponentes de la base 10 son positivas; y, a la parte fraccionaria, los términos con exponentes negativos.

En el sistema decimal, a la parte fraccionaria se denomina comúnmente *parte decimal*.

2.1.2. Sistema de numeración binario

El sistema binario es el más sencillo de todos, pues, cuenta con solo dos dígitos distintos. No obstante, su sencillez, cualquier número puede ser representado mediante notación polinómica utilizando sólo dos dígitos binarios (bit), el 0 y el 1.

Los sistemas de cómputo utilizan éste sistema de numeración porque es compatible con el funcionamiento interno del mismo, ya que las señales eléctricas que reciben los dispositivos y componentes electrónicos sólo responden a dos niveles de voltaje. Por éste motivo, el sistema binario es el predilecto de cualquier organización y arquitectura de computadoras.

Una de las características del sistema binario es la facilidad y sencillez de las operaciones lógicas y aritméticas.

El número binario $101101,1101_2$, en polinomio de potencias se expresa así:

$$101101,1101_2 = \underbrace{1x2^5 + 0x2^4 + 1x2^3 + 1x2^2 + 0x2^1 + 1x2^0}_{\text{Parte entera}} + \underbrace{1x2^{-1} + 1x2^{-2} + 0x2^{-3} + 1x2^{-4}}_{\text{Parte fraccionaria}}$$

Nótese que el subíndice 2 indica la base del número.

La lectura del número se realiza dígito por dígito y luego se enuncia la base (para cualquier número distinto al decimal).

Ejemplo: leer el número $101101,1101_2$. Se lee: UNO, CERO, UNO, UNO, CERO, UNO, COMA, UNO, UNO, CERO, UNO, en base dos.

2.1.3. Conversión del sistema binario a decimal y viceversa

Cualquier número binario puede convertirse a decimal desarrollando el polinomio de potencias.

Ejemplo: convertir $11011,101_2$ a decimal.

$$\begin{aligned} 11011,101_2 &= 1x2^4 + 1x2^3 + 0x2^2 + 1x2^1 + 1x2^0 + 1x2^{-1} + 0x2^{-2} + 1x2^{-3} \\ &= 16 + 8 + 0 + 2 + 1 + 0,5 + 0 + 0,125 \\ &= 27,525 \text{ (cuando no tiene subíndice indica número decimal)} \end{aligned}$$

Mediante un tablero de valor posicional podemos convertir cualquier número decimal al binario, teniendo en cuenta que las potencias de la parte entera empiezan ascender desde cero a partir de la coma. El bit más significativo (MSB) es el que tiene mayor peso y el bit menos significativo (LSB) es que tiene menor peso.

Ejemplo: convertir los cinco primeros números decimales a binario.

Sistema decimal	Sistema binario		
	2^2	2^1	2^0
	4	2	1
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0

Tabla. 2.1
 Tablero de valor posicional. La fila remarcada indica el peso que adquiere cada orden. Aumenta de derecha a izquierda.

Ejemplo: convertir el número 214 a binario.

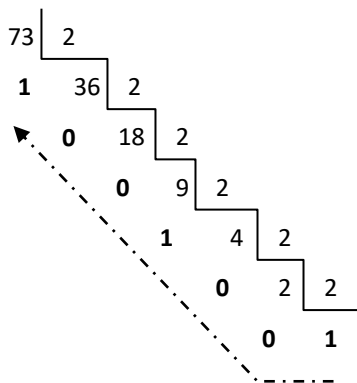
Sistema decimal	Sistema binario							
	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
	128	64	32	16	8	4	2	1
214	1	1	0	1	0	1	1	0

$$214 = 11010110_2$$

Para convertir a binario un número decimal que posee parte entera y fraccionaria, se convierte la parte entera utilizando el tablero posicional o el método de divisiones sucesivas; mientras que la parte fraccionaria se convierte mediante las multiplicaciones sucesivas.

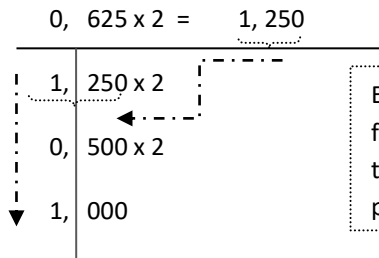
Convertir a binario el número decimal 73,625

Aplicando el método de divisiones sucesivas para la parte entera tenemos:



El número binario de la parte entera se obtiene tomando el último cociente y luego los residuos anteriores.

Aplicando multiplicaciones sucesivas por 2, para la parte fraccionaria.



El número binario de la parte fraccionaria se obtiene tomando la parte entera de los productos sucesivos.

Por tanto: **73,625 = 1001001,101**

2.1.4. Sistema octal y hexadecimal

La mejor manera de representar números octales, hexadecimales y sus equivalentes decimales es utilizando el tablero de valor posicional.

La base del sistema octal es 8. Dispone de ocho dígitos (0, 1, 2, 3, 4, 5, 6, 7) para representar cualquier número.

La base del sistema hexadecimal es 16. Cuenta con dieciséis dígitos (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F), incluye las cifras del 0 al 9 y las letras de la A a la F.

Convierte el número 6425_8 a decimal.

$$\begin{aligned} 6425_8 &= 6 \times 8^3 + 4 \times 8^2 + 2 \times 8^1 + 5 \times 8^0 \\ &= 3072 + 256 + 16 + 5 \end{aligned}$$

$$\boxed{6425_8 = 3349}$$

Convierte el número decimal 351 a octal.

Sistema decimal	Sistema octal		
	8^2	8^1	8^0
351	64	8	1
	5	3	7

$$351 = 5 \times 8^2 + 3 \times 8^1 + 7 \times 8^0$$

$$351 = 320 + 24 + 7$$

$$\boxed{351 = 537_8}$$

Convierte el número $9B4_H$ a decimal.

$$\begin{aligned} 9B4_H &= 9 \times 16^2 + B \times 16^1 + 4 \times 16^0 \\ &= 2304 + 176 + 4 \end{aligned}$$

$$\boxed{9B4_H = 2484}$$

Convierte el número decimal 745 a hexadecimal.

Sistema decimal	Sistema Hexadecimal		
	16^2	16^1	16^0
745	256	16	1
	2	E	9

$$745 = 2 \times 16^2 + E \times 16^1 + 9 \times 16^0$$

$$= 512 + 224 + 9$$

$$\boxed{745 = 2E9_H}$$

2.1.5. Conversión del sistema octal y hexadecimal a binario y viceversa.

El sistema octal y hexadecimal son compatibles entre sí y compatibles con el sistema binario, pues la conversión entre ellos se realiza de manera fácil y sin complicaciones.

La base del sistema octal es 8, y su afinidad con el binario estriba en el hecho que **ocho** es potencia de **dos**; $8 = 2^3$. Esta configuración indica que tres dígitos binarios hacen un dígito octal.

Sistema decimal	Sistema binario			Sistema octal
	2^2	2^1	2^0	
	4	2	1	
0	0	0	0	0
1	0	0	1	1
2	0	1	0	2
3	0	1	1	3
4	1	0	0	4
5	1	0	1	5
6	1	1	0	6
7	1	1	1	7

Tabla. 2.2

Equivalencia de los primeros ocho números en tres sistemas de numeración: decimal, binario y octal

Convierte el número octal 24713₈ a binario.

Solución: a cada dígito octal le corresponde tres dígitos binarios.

2 4 7 1 3
010 100 111 001 011

$$\boxed{24713_8 = 0101001111001011_2}$$

Convierte el número binario 1010111011110100101₂ a octal.

Solución: los dígitos binarios se agrupan de tres en tres, empezando de la derecha hacia la izquierda. Si es necesario, el último grupo se completa con ceros.

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

$$\boxed{1010111011110100101_2 = 1273645_8}$$

Los dígitos 0 remarcados se agregaron para completar el último grupo.

Del mismo modo, la base del sistema hexadecimal es 16 y **dieciséis** es potencia de **dos**; $16 = 2^4$, lo cual indica que cuatro dígitos binarios configuran un dígito hexadecimal.

Sistema decimal	Sistema binario				Sistema hexadecimal
	2^3	2^2	2^1	2^0	
	8	4	2	1	
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	2
3	0	0	1	1	3
4	0	1	0	0	4
5	0	1	0	1	5
6	0	1	1	0	6
7	0	1	1	1	7
8	1	0	0	0	8
9	1	0	0	1	9
10	1	0	1	0	A
11	1	0	1	1	B
12	1	1	0	0	C
13	1	1	0	1	D
14	1	1	1	0	E
15	1	1	1	1	F

Tabla. 2.3

Equivalencia de los primeros dieciséis números en tres sistemas de numeración: decimal, binario y hexadecimal.

Convierte el número hexadecimal 8A3E_H a binario.

Solución: a cada dígito hexadecimal le corresponde cuatro dígitos binarios.

8 A 3 E
 1000 1010 0011 1110

$$\boxed{8A3E_H = 1000101000111110_2}$$

Convierte el número binario 111011111011110101101₂ a hexadecimal.

Solución: los dígitos binarios se agrupan de cuatro en cuatro, empezando de la derecha hacia la izquierda. Si es necesario, el último grupo se completa con ceros.

0011 1011 1111 0111 1010 1101
 3 B F 7 A D

$$\boxed{111011111011110101101_2 = 3BF7AD_H}$$

2.2. Representación de números en formato signo magnitud

Los números binarios con los cuales hemos trabajado, son números que no tienen signo, pues no hemos considerado esa posibilidad. Sin embargo, las computadoras frecuentemente utilizan números con signo para efectuar operaciones aritméticas.

Mediante el formato signo magnitud haremos una primera aproximación en la inclusión de signo positivo o negativo en un número binario; luego, en la sección (9.3), trataremos todo lo relacionado a un formato preferido por los sistemas aritméticos llamado “complemento a dos”.

El formato signo magnitud consiste en ingresar el signo positivo (+) o negativo (-) en un número binario, incorporando un **bit** a la izquierda del bit más significativo (MSB). Convencionalmente, el **bit 0** indica **signo positivo**, y el **bit 1**, **signo negativo**.

El formato signo magnitud está conformado por un bit de signo y un conjunto de bits para la magnitud. Por tanto, el intervalo de números disponibles está en función del número de bits de la magnitud.

Si la magnitud tiene n bits, entonces el número disponible será: $2(2^n)$; entre positivos y negativos, incluyendo el cero tanto para los positivos como para los negativos.

Con 5 bits para la magnitud, el intervalo de los números disponibles será: $2(2^5) = 64$; es decir, desde -31 hasta +31. El cero, en este caso está representado para el positivo como: **00000** y **10000** para el negativo.

Representa en formato signo magnitud el número decimal +67 y -43, con siete bits en la magnitud.

$$\begin{array}{l} +67 = \underbrace{0}_{\text{Bit de signo}} \underbrace{1000011}_{\text{Magnitud}} = 01000011 \\ -43 = \underbrace{1}_{\text{Bit de signo}} \underbrace{0101011}_{\text{Magnitud}} = 10101011 \end{array}$$

Nótese que el bit **0** remarcado de -43, se incluyó para completar los siete bits solicitados.

Obviamente, los computadores utilizan cantidades mayores para procesar información, por lo que su intervalo de valores también se incrementa.

2.3. Códigos Binarios

Los códigos binarios son un conjunto de bits combinados de acuerdo a una regla de formación, los cuales tienen significado en un sistema establecido.

Los códigos binarios se originan en el hecho que tenemos los humanos de comunicarnos con la computadora. Nosotros comunicamos información habitualmente en el sistema decimal, sin embargo, la computadora internamente lo hace en el sistema binario. La conversión de un sistema a otro exige la utilización de códigos numéricos.

El sistema de numeración binario constituye por sí mismo un código denominado CÓDIGO BINARIO NATURAL

Existen otros códigos binarios cuya regla de formación se realiza en base a los pesos que tienen cada orden, a estos códigos se les denomina *“ponderados”*.

Cuando la combinación de bits de dos números consecutivos de un código binario se distinguen por el número de bits, se les designan *“continuos”*.

Cuando la primera y última combinación de un código binario se diferencia en un sólo bit, se llaman *“cíclicos”*.

2.3.1. Código BCD

Un número decimal codificado en binario es una combinación muy utilizada para representar cantidades del mundo físico, pues cuando deseamos ingresar datos en la computadora lo hacemos utilizando números decimales y un dígito a la vez.

En el código **Decimal Codificado en Binario** (BCD), es un código **ponderado**, en la cual, cada número decimal se codifica directamente en un código binario.

Número Decimal	BCD			
	Peso 8	Peso 4	Peso 2	Peso 1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

Tabla. 2.4

Código BCD. Ponderado de izquierda a derecha con los Pesos: 8, 4, 2, y 1.

Para representar los diez dígitos decimales se necesitaron cuatro bits, porque con tres bits sólo podríamos codificar $2^3 = 8$ combinaciones, del 0 al 7; pero, con las dieciséis posibles combinaciones $2^4 = 16$, sólo son aprovechables diez, del 0 al 9; las otras seis combinaciones no se utilizan. Cuando se realizan operaciones aritméticas con dígitos expresados en BCD, es común utilizar el seis como **factor de corrección (110)** para mostrar un resultado BCD.

Expresa en Binario Natural y en BCD el número decimal 83.

$$83 = 1010011_2 \quad \text{en Binario Natural}$$

$$83 = \underbrace{10000011}_{8 \quad 3} \quad \text{en BCD}$$

2.3.2. Código BCD Aiken

Es un código BCD ponderado con los Pesos: 2, 4, 2, y 1. En base a esta distribución se aplica la regla de formación de los códigos, con la finalidad de conseguir la simetría entre dos números opuestos: el primero y el último, el segundo y el penúltimo y así sucesivamente.

Número Decimal	BCD Aiken			
	Peso 2	Peso 4	Peso 2	Peso 1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	1	0	1	1
6	1	1	0	0
7	1	1	0	1
8	1	1	1	0
9	1	1	1	1

Tabla. 2.5

Código BCD Aiken Ponderado de izquierda a derecha con los Pesos: 2, 4, 2, y 1. Nótese que la combinación 0 y 9 son simétricas, así como 1 y 8, 2 y 7, etc.

Expresa en Binario Natural, en BCD y en BCD Aiken, el número decimal 945.

$$945 = 1110110001_2 \quad \text{en Binario Natural}$$

$$945 = 100101000101 \quad \text{en BCD}$$

$$945 = 111101001011 \quad \text{en BCD Aiken}$$

2.3.3. Código BCD exceso 3

Es un código no ponderado. La regla de formación se realiza en el Código BCD, sumando el valor 3 a cada combinación.

Número Decimal	BCD				BCD Exceso 3
	P8	P4	P2	P1	
0	0	0	0	0	0011
1	0	0	0	1	0100
2	0	0	1	0	0101
3	0	0	1	1	0110
4	0	1	0	0	0111
5	0	1	0	1	1000
6	0	1	1	0	1001
7	0	1	1	1	1010
8	1	0	0	0	1011
9	1	0	0	1	1100

Tabla. 2.6

Código BCD exceso 3, No Ponderado. Nótese que la simetría es la principal propiedad de éste código, similar al BCD Aiken.

2.3.4. Código Gray

El código Gray o “reflejado”, es un código no ponderado. La regla de formación de cada combinación se realiza mediante el reflejo “como un espejo” de la combinación anterior; a las combinaciones por encima del espejo se completa con ceros y a los de abajo con unos.

Número Decimal	CODIGO GRAY			
0	0	0	0	0
1	0	0	0	1
2	0	0	1	1
3	0	0	1	0
4	0	1	1	0
5	0	1	1	1
6	0	1	0	1
7	0	1	0	0
8	1	1	0	0
9	1	1	0	1
10	1	1	1	1
11	1	1	1	0
12	1	0	1	0
13	1	0	1	1
14	1	0	0	1
15	1	0	0	0

Tabla. 2.7

Código Gray, No Ponderado. Nótese los tres espejos con líneas negritas. La combinación que está por encima de la línea (espejo) se refleja en la parte inferior, luego se completa con ceros y unos respectivamente.

El Código Gray es “continuo” y es “cíclico”. Es “continuo” porque entre combinaciones consecutivas existe la diferencia de un bit 1. Por ejemplo, entre la combinación 4 (0110), presenta dos bit 1; y la combinación 5 (0111) cuenta con tres bit 1 y así sucesivamente. Es “cíclico”, porque la diferencia entre la primera y última combinación, la diferencia es de un bit.

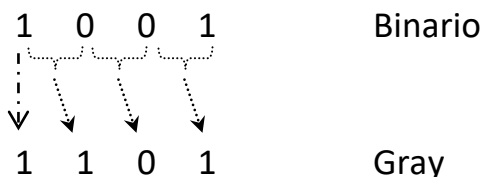
El Código Gray es preferido en sistemas de alta velocidad, ya que su configuración permite una fácil y segura conversión con el sistema binario. Es muy utilizado en robótica y control automático.

Conversión de un número Binario a Gray

El bit más significativo (MSB) es igual para ambos códigos, luego, continuando hacia la derecha se van comparando los bits Binarios: si son iguales el bit Gray será **cero**; si son diferentes será **uno**.

Decimal	Binario	Gray
9	1001	1101

Convierte el número binario 1001_2 a Gray.

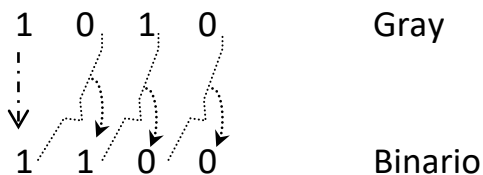


Conversión de un número Gray a Binario

El bit más significativo (MSB) es el mismo para ambos códigos; la comparación se realiza hacia la derecha: se compara el bit Binario con el bit Gray: si son iguales el bit Binario será cero; si son diferentes uno.

Decimal	Binario	Gray
12	1100	1010

Convierte el número Gray 1010 a Binario-



2.3.5. Códigos alfanuméricos

Una computadora recibe del exterior no solo números, sino también, letras, signos de puntuación y otros caracteres especiales, que requieren ser codificados para procesarse convenientemente.

El código alfanumérico de aceptación universal es el **CODIGO ASCII** (*American Standard Code for Information Interchange*), conformado por 128 combinaciones, es decir, utiliza 7 bits: $2^7 = 128$, para representar a todos los caracteres numéricos, alfabéticos y caracteres especiales. Sin embargo, le han agregado otro bit más para obtener doscientas cincuenta y seis combinaciones, $2^8 = 256$; constituyendo lo que se denomina **CODIGO ASCII Extendido**.

Un **BYTE** ó carácter de ocho bits, tiene significado porque forma parte del código ASCII, ya que en función de éste se ingresan datos a la computadora, se procesa y posteriormente se obtiene información comprensible para el usuario.

Código ASCII											
Número Decimal	Número Binario								Número Hexadecimal		Carácter
	128	64	32	16	8	4	2	1	16	1	
35	0	0	1	0	0	0	1	1	2	3	#
37	0	0	1	0	0	1	0	1	2	5	%
50	0	0	1	1	0	0	1	0	3	2	2
55	0	0	1	1	0	1	1	1	3	7	7
64	0	1	0	0	0	0	0	0	4	0	@
65	0	1	0	0	0	0	0	1	4	1	A
69	0	1	0	0	0	1	0	1	4	5	E
72	0	1	0	0	1	0	0	0	4	8	H
73	0	1	0	0	1	0	0	1	4	9	I
76	0	1	0	0	1	1	0	0	4	C	L
78	0	1	0	0	1	1	1	0	4	E	N
79	0	1	0	0	1	1	1	1	4	F	O
80	0	1	0	1	0	0	0	0	5	0	P
83	0	1	0	1	0	0	1	1	5	3	S
92	0	1	0	1	1	1	0	0	5	C	\
123	0	1	1	1	1	0	1	1	7	B	{
125	0	1	1	1	1	1	0	1	7	D	}
126	0	1	1	1	1	1	1	0	7	E	~
127	0	1	1	1	1	1	1	1	7	F	DELETE

Tabla. 2.8

Algunos caracteres del código ASCII. Existe una relación entre los sistemas decimal, binario y hexadecimal

2.4. Detectores de error.

La transmisión de información (códigos binarios) se realiza a través de medios físicos, lo cuales por diversas influencias pueden alterar el contenido de la “data”, convirtiendo un bit 0 a 1 ó un bit 1 a 0; es por ello, que los sistemas digitales implementan un método para detección de errores utilizando un bit adicional al código de información para comprobar su verdadero contenido.

2.4.1. Bit de paridad.

El bit de paridad es el método más simple para detectar errores en la transmisión de datos. El bit de paridad puede ser un 1 ó un 0 que se agrega al código binario para contar el número de 1 que tiene.

El bit de paridad se adiciona al código binario, es decir:

"data" + bit de paridad = código transmitido

Hay dos tipos de bit de paridad:

Bit de paridad par. Se coloca un bit 1 si el número de 1s del código binario es impar, de tal manera, que el total de 1s de la “data” sea PAR. Y se coloca el bit 0 si el número de 1s del código binario es par.

Bit de paridad impar. Se coloca un bit 1 si el número de 1s del código binario es par, de tal manera, que el total de 1s de la “data” sea IMPAR. Y se coloca el bit 0 si el número de 1s del código binario es impar.

Para transmitir la “data”, los fabricantes de sistemas informáticos adoptan uno de los dos tipos de bit de paridad; aunque no corrige el error, ayuda a detectar los posibles errores que se pueden presentar a lo más en 1 bit.

Número decimal	Data Código binario	BIT DE PARIDAD	
		PARIDAD PAR	PARIDAD IMPAR
35	00100011	100100011	000100011
75	01001011	001001011	101001011
117	01110101	101110101	001110101
215	11010111	011010111	111010111
245	11110101	011110101	111110101

Problemas Resueltos N° 02

1. Convertir a decimal el número $1011010,1011_{(2)}$.

Solución:

Se realiza una descomposición polinómica del número binario.

$$1x2^6+0x2^5+1x2^4+1x2^3+0x2^2+1x2^1+0x2^0+1x2^{-1}+0x2^{-2}+1x2^{-3}+1x2^{-4} = 90,6875$$

2. Convertir a hexadecimal el número $365,7_{(8)}$.

Solución:

Tanto el número hexadecimal como el octal son compatibles con el binario.

Cada cifra octal se representa con tres dígitos binarios teniendo en cuenta la parte entera como la parte fraccionaria.

3 6 5, 7₍₈₎

011 110 101, 111₍₂₎. Luego, éste número binario se agrupa de 4 bits, empezando por la coma; tanto hacia la izquierda como a la derecha. Así:

1111 0101, 1110

F 5, E_(H)

3. Convertir el número BCD 1001010001110000 a decimal.

Solución:

El número BCD se agrupa de 4 bits, empezando desde la izquierda.

1001 0100 0111 0000

9 4 7 0

4. Convertir el número $125,123$ a binario.

Solución:

La parte entera 125, se encuentra ya sea por divisiones sucesivas o mediante la tabla del sistema binario.

Número decimal	Sistema Binario						
	64	32	16	8	4	2	1
125	1	1	1	1	1	0	1

La parte fraccionaria $0,123$ se multiplica sucesivamente por 2.

0, 123 x 2

0, 246 x 2

0, 492 x 2

0, 984 x 2

1,968 x 2

1,936 x 2

1,872 x 2

1,744 x 2

Uniando la parte entera con la parte fraccionaria tenemos:

$$125,123 = 1111101,0001111..$$

Los puntos suspensivos indican que los bits de la parte fraccionaria continúan.

5. Convertir $11010110_{(2)}$ a código Gray.

Solución:

Se tiene en cuenta que el primer bit de la izquierda del número binario es el mismo en el código Gray. Luego se compara los bits consecutivos del número binario; si son iguales, en el código Gray es 0; si son diferentes, en el código Gray es 1.

1 1 0 1 0 1 1 $0_{(2)}$ Binario

1 0 1 1 1 1 0 1 Gray

6. Convertir a binario el número en código Gray 1011101101

Solución:

Se tiene en cuenta que el primer bit de la izquierda del número Gray es el mismo en Binario. Luego se compara los bits consecutivos del número Gray con el bit binario; si son iguales, en el código binario es 0; si son diferentes, es 1.

1 0 1 1 1 0 1 1 0 1 Gray

1 1 0 1 0 0 1 0 0 1 $1_{(2)}$ Binario

7. Convertir el número BCD 1001011101010110 a octal.

Solución:

El número BCD 1001011101010110 se convierte a decimal y luego a octal.

1001 0111 0101 0110

9 7 5 6

El número decimal 9 756 se convierte a octal.

$9\ 756 = 23\ 034_{(8)}$

8. Convertir $8A9D_{(H)}$ a binario.

Solución:

Cada dígito hexadecimal se reemplaza por cuatro dígitos binarios.

8 A 9 D_(H)

1000 1010 1001 1101₍₂₎

9. ¿Cuántos bits se requiere para representar el número decimal 675?

Solución:

El número decimal se convierte a binario y se cuenta el número de bits que se empleó en su formación.

Sistema Decimal	SISTEMA BINARIO										Nº Bits	
	512	256	128	64	32	16	8	4	2	1		
675	1	0	1	0	1	0	0	0	0	1	1	10

10. Expresar en formato binario de 8 bit en signo magnitud del número +87.

Solución:

El signo + en formato signo magnitud se representa mediante el bit 0.

Signo **magnitud**

$$0 \quad \mathbf{1010111} = +87$$

11. Expresar en formato signo magnitud de 9 bits, el número -119.

Solución:

El signo - en formato signo magnitud se representa mediante el bit 1.

Signo **magnitud**

$$1 \quad \mathbf{1110111} = -119$$

12. Determinar el valor decimal del número binario 11101110, expresado en formato signo magnitud.

Solución:

Como el número está expresado en signo magnitud, el primer bit de la izquierda representa el signo. Por tanto, el número es negativo.

Signo **magnitud**

$$1 \quad \mathbf{1101110} = -110$$

13. Codifique en BCD exceso 3, el decimal 47

Solución:

Cada dígito decimal se codifica en BCD exceso 3 (téngase en cuenta la tabla 2.6).

4 7

0111 1010 BCD exceso 3

14. Codifique en BCD Aiken el decimal 69.

Solución:

Cada dígito decimal se codifica en BCD Aiken (téngase en cuenta la tabla 2.5)

6 9

1100 1111 BCD Aiken

15. Codifique en ASCII con paridad impar el mensaje "SIPAN".

Solución:

Cada carácter del mensaje se codifica según la tabla 2.8.

S = 01010011

I = 01001001

P = 01010000

A = 01000001

N = 01001110

A los códigos anteriores se agrega el bit de paridad impar.

S = 1 01010011

I = 0 01001001

P = 1 01010000

A = 1 01000001

N = 1 01001110

El bit de paridad se puede adicionar en cualquiera de los extremos del código, sin embargo, es común utilizarlo al extremo del bit más significativo.

Problemas Propuestos N° 02

1. Convertir el número BCD 1000100100100101 a hexadecimal. *Rpta: 22DD_(H).*
2. Convertir el número decimal 235 a código BCD y luego agregue un bit de paridad par. *Rpta: 1001000110101*
3. Convertir el número Gray 101011101 a binario y luego adicione un bit de paridad impar. *Rpta: 0110010110*
4. Expresar el número decimal 345, con paridad par e impar. *Rpta: 1101011001 y 0101011001*
5. Expresé con paridad par el número decimal -453 en formato signo magnitud. *Rpta: 11111000101*
6. Se transmitió información con bit de paridad par. Determine cuál de los siguientes datos se transmitieron correctamente.
 - a. 11011010111011101
 - b. 10110101000110111
 - c. 11010111111010101
 - d. 10101101010011001*Rpta: sólo (d) no es correcto*
7. Determinar el valor decimal de número binario expresado en formato signo magnitud: 10011100100000. *Rpta: -1824.*
8. El siguiente dato es un mensaje codificado en ASCII ¿Cuál es el mensaje?
01001000 01001111 01001100 01000001

Descripción de circuitos digitales

3.1. Tablas de verdad

La tabla de verdad es una herramienta gráfica que permite visualizar todas las posibles combinaciones de los valores de verdad que adoptan las variables de una función lógica y al mismo tiempo determinar la salida de dicha función en base a las combinaciones establecidas.

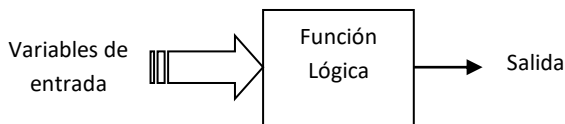


Fig. 3.1

Sistema Digital. Contiene entrada de datos, proceso lógico y salida.

El número de combinaciones posibles de los valores de verdad (**Comb**) depende del número de variables de la función (**# var**), **Comb = 2^{# var}**. Si la función tiene tres variables, entonces el número de combinaciones será: $Comb = 2^3 = 8$; ocho posibles combinaciones.

N° Posibilidades	VARIABLES			FUNCIÓN
	V1	V2	V3	
0	F	F	F	f(FFF)
1	F	F	V	f(FFV)
2	F	V	F	f(FVF)
3	F	V	V	f(FVV)
4	V	F	F	f(VFF)
5	V	F	V	f(VFV)
6	V	V	F	f(VVF)
7	V	V	V	f(VVV)

Tabla 3-1

Tabla de verdad con tres variables de entrada, ocho combinaciones posibles y la salida como función de cada una de las entradas.

3.2. Variables y funciones lógicas

Se entiende como **variable lógica** a aquella magnitud que puede adoptar cualquiera de dos valores posibles. Cuando se habla de valores no significa que sean numéricos, sino que representan estados, por ejemplo: pueden ser alto, bajo; lleno, vacío; activado desactivado; abierto, cerrado, etc.

A cada estado binario le corresponde un valor lógico, que para nuestro caso lo representaremos con un 0 ó con un 1, según la lógica que escojamos. Es preciso enfatizar nuevamente que el 0 y el 1 lógico no son valores numéricos sino niveles lógicos; es decir, si en la entrada o salida de un circuito integrado hay un voltaje entre 2 y 5 voltios, entonces lo representaremos con un 1; en cambio, si los voltajes están comprendidos entre 0 y 0,8 voltios, entonces se representará con un 0.

La variable lógica, además de representarse mediante una letra mayúscula se caracteriza por poseer tres cualidades particulares.

- Puede asumir cualquiera de sólo dos valores posibles.
- Los valores se pueden expresar mediante sentencias declarativas.
- Los dos valores posibles tienen que ser recíprocamente excluyentes.

Se entiende por **función lógica** a aquella relación cuya salida depende de los valores de las variables lógicas de la entrada.

Una función lógica se analiza y se entiende mejor cuando se halla representado en una tabla de verdad, pues en ella se analizan todas las posibles combinaciones de los valores de la variable lógica.

3.3. Diagramas de tiempo

Es la representación en el tiempo de las variaciones que presentan las variables de entrada y la función de salida. Las ondas o señales se grafican en base a dos estados, el cero y el uno, y distribuidos en una línea de tiempo.

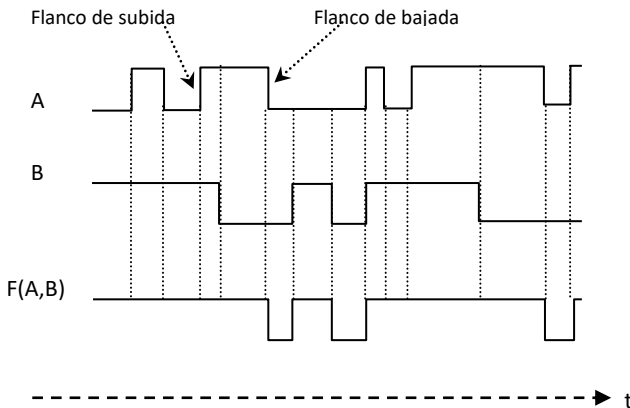


Fig. 3-2

Diagrama de tiempo de una función $F(A,B)$ de dos variables de entrada. Nótese que la función se analiza en cada flanco de los niveles de las variables lógicas.

3.4. Lógica de contactos

En electrónica digital es usual trabajar con interruptores, pues se trata de un dispositivo que funciona como una variable lógica, dado que su funcionamiento tiene sólo dos posibilidades: abierto y cerrado. Cuando el interruptor está abierto se le asigna arbitrariamente el valor lógico (0) para indicar que no circula corriente por el conductor. Cuando el interruptor está cerrado se le da el valor lógico (1) e indica que hay circulación de corriente por el conductor.

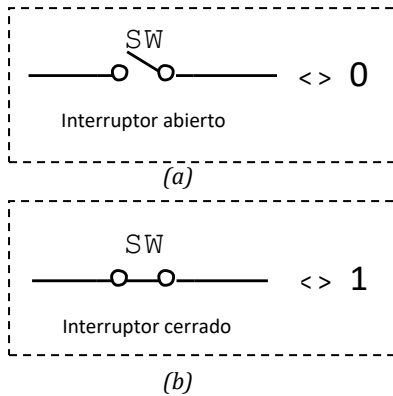


Fig. 3.3

Los interruptores abiertos no conducen corriente eléctrica y adoptan el valor lógico 0; toman el valor lógico 1 si está cerrado e indica que hay circulación de corriente.

3.5. Compuertas lógicas

Son bloques electrónicos que representan la operación que efectúa una función lógica.

Las compuertas lógicas procesan las señales que ingresan desde un conjunto de líneas de entrada, obteniéndose luego, una señal de salida, precisamente el resultado de la función lógica.

Las compuertas lógicas tienen un diseño gráfico de acuerdo a la función que representa, con ellos se organizan funciones lógicas más complejas.

Físicamente las compuertas lógicas se implementan en circuitos integrados, normalmente más de dos. Existen varios fabricantes, los cuales construyen en base a diversas tecnologías.

Fabricante y código
del integrado

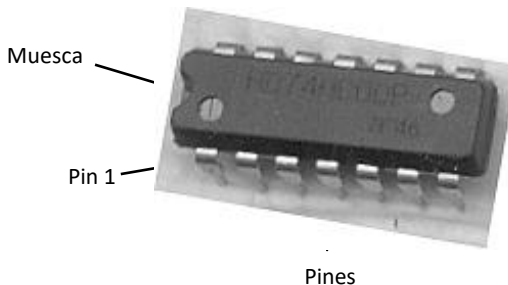


Fig. 3.4

Circuito integrado cuyo patillaje o pines se empieza a contar desde la parte inferior de la muesca. Para realizar lo anterior la muesca tiene que estar a la izquierda tal como en la figura.

3.5.1. Compuerta Lógica OR

La operación lógica que realiza la compuerta OR consiste en obtener un nivel ALTO en la salida, siempre que exista por lo menos una línea en ALTO en la entrada, en caso contrario la salida será BAJO.

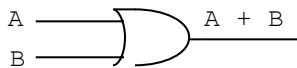
N° Decimal	VARIABLES DE ENTRADA		SALIDA (A + B)
	A	B	
0	0	0	0
1	0	1	1
2	1	0	1
3	1	1	1

Tabla 3.2

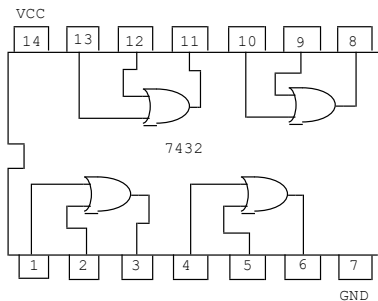
Tabla de verdad de la operación lógica OR, de dos entradas.

La operación OR se llama también SUMA LÓGICA (A+B). En la tabla 3.2, se aprecia que la combinación tres: $1 + 1 = 1$, dado que no se trata de una suma aritmética.

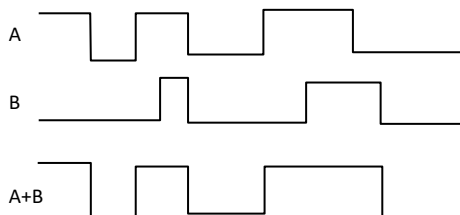
Generalizando: basta que un término de la suma lógica esté en 1 para que la suma sea 1.



(a)



(b)



(c)

Fig. 3.5

(a) Símbolo de la compuerta OR de dos entradas.

(b) Circuito integrado 7432, contiene cuatro compuertas OR de dos entradas. Observe que el pin 7 corresponde a tierra (GND) o negativo y el pin 14 a fuente (Vcc) o positivo.

(c) Diagrama de tiempo de la operación lógica OR. Se verifica que siempre que exista un nivel ALTO en cualquiera de las entradas, la salida será ALTO, mientras que la salida será BAJO cuando las dos entradas están en BAJO.

3.5.2. Compuerta Lógica AND

La operación lógica que realiza la compuerta AND consiste en obtener un nivel ALTO en la salida, siempre que todas las líneas de entrada estén en ALTO. En el caso que una de las entradas esté en nivel BAJO, la salida estará forzosamente en nivel BAJO.

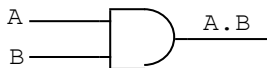
N° Decimal	VARIABLES DE ENTRADA		SALIDA (A · B)
	A	B	
0	0	0	0
1	0	1	0
2	1	0	0
3	1	1	1

Tabla 3.3

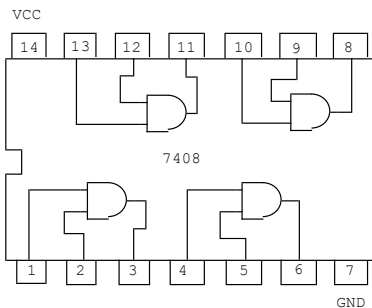
Tabla de verdad de la operación lógica AND de dos entradas.

A la operación AND se le conoce también como PRODUCTO LÓGICO. En la tabla 3.3, se verifica que los resultados de **A.B** coinciden con los resultados del producto aritmético.

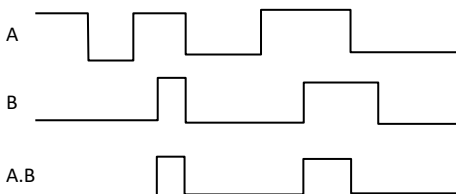
Generalizando: basta que un término del producto lógico esté en 0 para que el producto sea 0.



(a)



(b)



(c)

Fig. 3.6

(a) Símbolo de la compuerta AND de dos entradas.

(b) Circuito integrado 7408, contiene cuatro compuertas AND de dos entradas. Observe que el pin 7 corresponde a tierra (GND) o negativo y el pin 14 a fuente (Vcc) o positivo.

(c) Diagrama de tiempo de la función AND. Se verifica que siempre que exista un nivel BAJO en cualquiera de las entradas, la salida será BAJO, mientras que la salida será ALTO cuando las dos entradas están en ALTO.

3.5.3. Compuerta Lógica NOT

La operación lógica que realiza la compuerta NOT, es invertir la señal de entrada, es decir, si en la entrada de la compuerta se encuentra en nivel BAJO, la salida estará en ALTO. La salida estará en BAJO, sólo si la entrada está en ALTO.

N° Decimal	VARIABLE	SALIDA (\bar{A})
	A	
0	0	1
1	1	0

Tabla 3.4

Tabla de verdad de la operación lógica NOT de una sola entrada.

La operación NOT se denomina también inversión o negación. La salida de la operación NOT se distingue porque la variable aparece con una barra en la parte superior (\bar{A}), lo cual significa negación de la variable A .

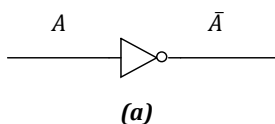
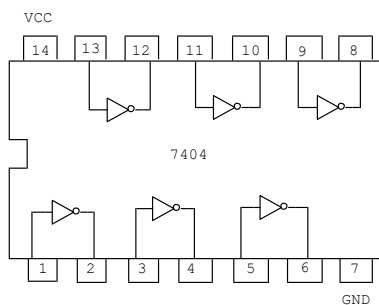


Fig. 3.7

(a) Símbolo de la compuerta NOT.



(b) Circuito integrado 7404, contiene seis compuertas NOT de una entrada.

3.5.4. Compuerta Lógica NOR

Las salidas de la compuerta NOR, son las salidas invertidas de la compuerta OR.



Fig. 3.8

Símbolo de la compuerta NOR. Obsérvese el pequeño círculo en el extremo de la OR, lo cual indica inversión o negación.

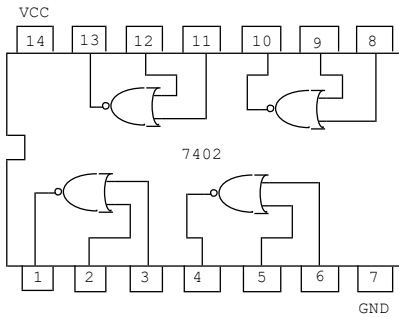


Fig. 3.9

Circuito integrado 7402, contiene cuatro compuertas NOR de dos entradas.

3.5.5. Compuerta Lógica NAND

Las salidas de la compuerta NAND, son las salidas invertidas de la compuerta AND.

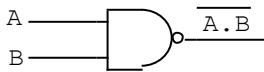


Fig. 3.10

Símbolo de la compuerta NAND. Obsérvese el pequeño círculo en el extremo de la NAND, lo cual indica inversión o negación.

La importancia y versatilidad de las compuertas NOR y NAND se tratará en la sección 4.6.

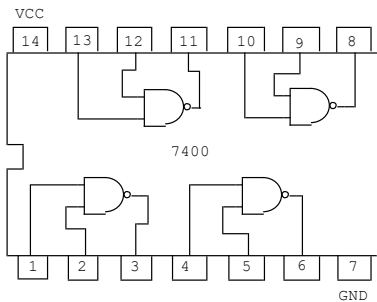


Fig. 3.11

Circuito integrado 7400, contiene cuatro compuertas NAND de dos entradas

3.5.6. Compuerta Lógica OR EXCLUSIVA

Es aquella operación cuya salida se pone en nivel ALTO cuando los valores lógicos de las dos entradas son complementarios, es decir, si uno de ellos está en ALTO y el otro en BAJO. La salida de la OR EXCLUSIVA se pondrá en BAJO sólo cuando ambas entradas tienen el mismo nivel lógico, ALTO, ALTO, ó BAJO, BAJO.

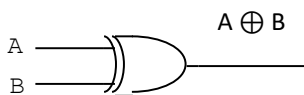
N° Decimal	VARIABLES DE ENTRADA		SALIDA ($A \oplus B$)
	A	B	
0	0	0	0
1	0	1	1
2	1	0	1
3	1	1	0

Tabla 3.6

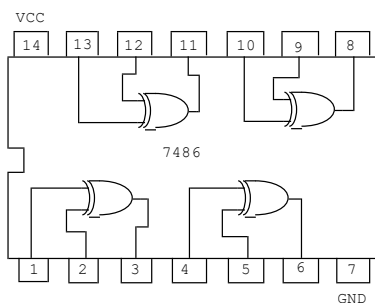
Tabla de verdad de la operación lógica OR EXCLUSIVA (OR-EX).

A diferencia de las operaciones OR, NOR, AND y NAND que pueden recibir más de dos líneas de entrada, la operación OR EXCLUSIVA ($A \oplus B$) recibe solo dos.

Si los niveles lógicos en ambas entradas son 0 ó 1, la salida bajo estas condiciones siempre será 0.



(a)



(b)

Fig. 3.12

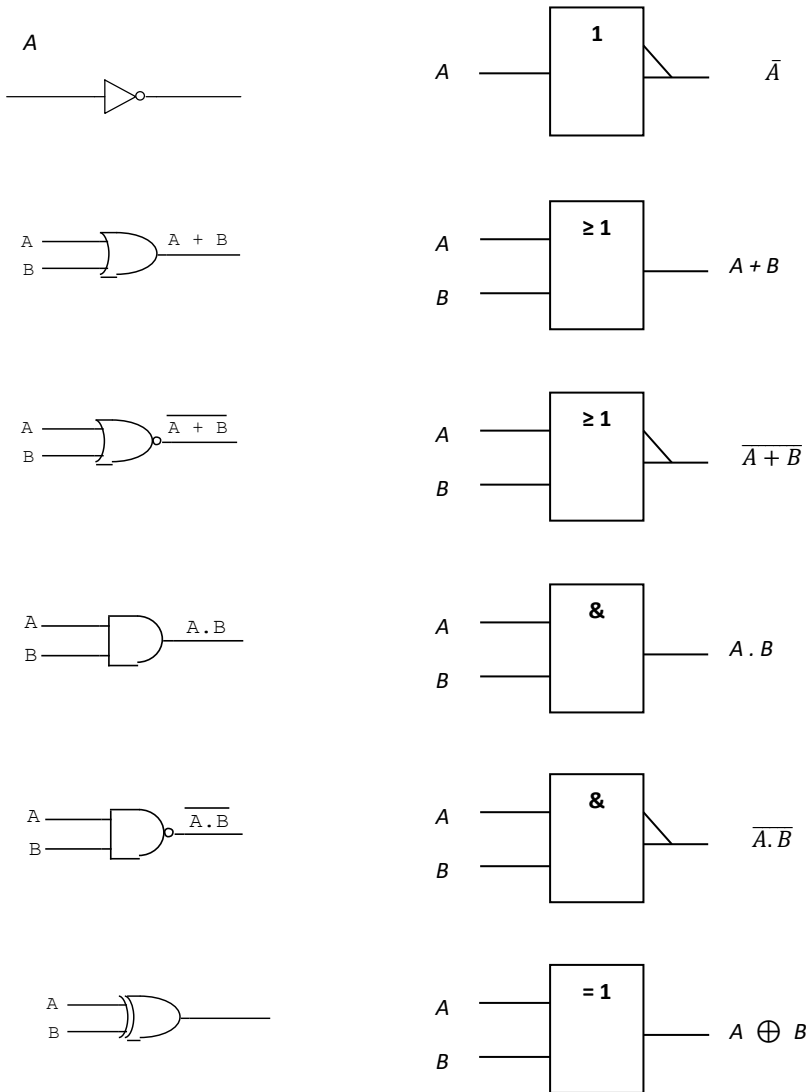
(a) Símbolo de la compuerta OR EXCLUSIVA (OR-EX).

(b) Circuito integrado 7486, contiene cuatro compuertas OR EXCLUSIVA de dos entradas

Las compuertas lógicas se pueden utilizar también como mecanismos de control, para impedir o dejar pasar una señal por la compuerta según indique el cable de comando.

3.6. Representaciones alternas de compuertas lógicas

Para representar cada compuerta lógica hemos utilizado una simbología tradicional, sin embargo, existe otra forma de representar las compuertas lógicas. Mediante una norma estandarizada definida por la IEEE/ANSE (Institute of Electrical and Electronic Engineers)/(American National Standard Institute), las compuertas lógicas se representan mediante símbolos rectangulares.



3.7. Implementación de circuitos

El propósito de estudiar electrónica digital consiste en comprender el funcionamiento interno de un sistema digital, en especial de la computadora. Para lograr este propósito se requiere no solo conocer los principios, leyes y técnicas digitales, sino también implementar físicamente circuitos lógicos y digitales de variada complejidad para complementar y ampliar el aspecto cognitivo.

El aspecto procedimental requiere de diagramas o gráficos que permitan visualizar el trabajo experimental que realizaremos, lo que conlleva a elaborar previamente el diseño del circuito digital que deseamos materializar, por supuesto, con su respectiva comprobación lógica de las entradas y salidas del mismo.

Trataremos básicamente dos clases de diagramas: el lógico y el esquemático.

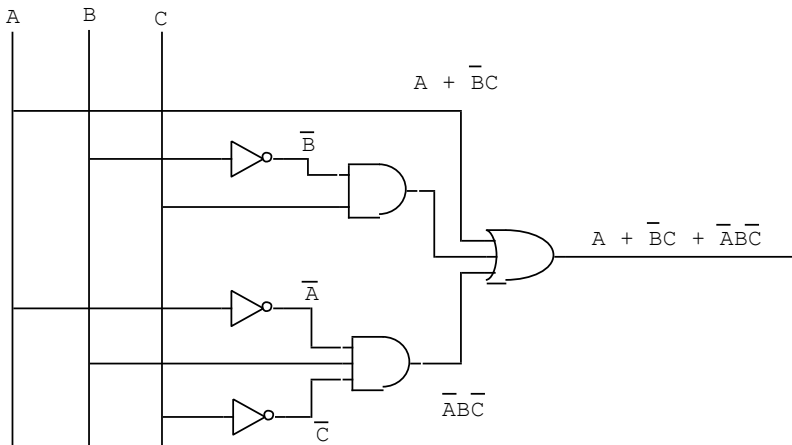
3.8. Diagramas lógicos

El diagrama lógico consiste en alambrear o unir las entradas y salidas de las compuertas lógicas, con el fin de cohesionar el circuito digital y tener un sistema que permita evaluar sus entradas y salidas.

Por ejemplo: implementar el diagrama lógico de la siguiente función lógica:

$$f(A,B,C) = A + \bar{B}C + \bar{A}\bar{C}$$

Nótese que la función $f(A,B,C)$ depende de tres variables lógicas, las cuales se denotan mediante letras mayúsculas y cuyos valores lógicos determinarán la salida de dicha función.



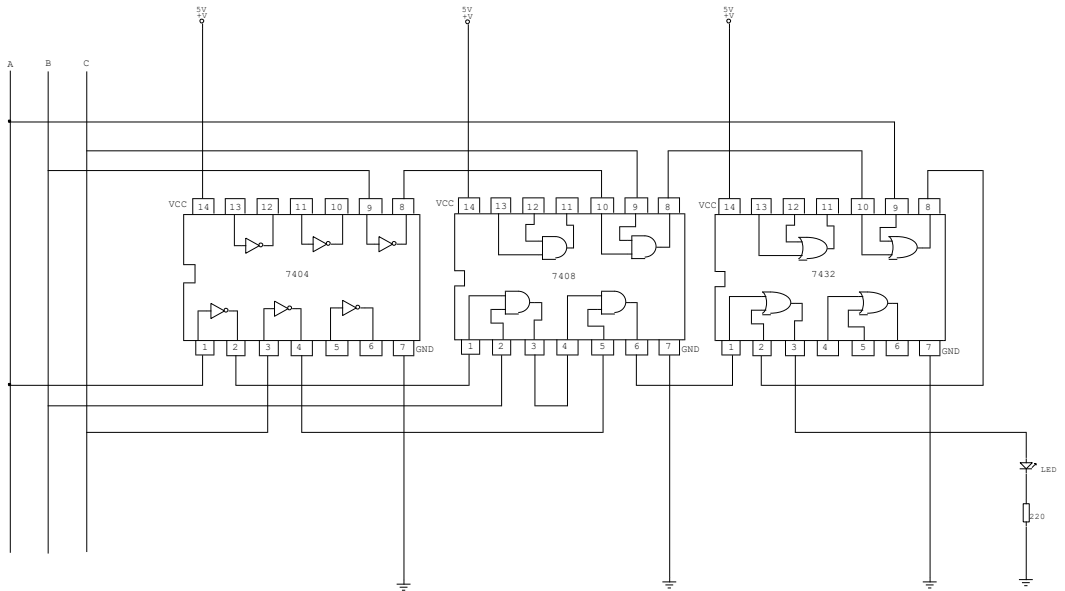
La evaluación de dicha función se realiza utilizando una tabla de verdad de tres variables de entrada.

En secciones posteriores analizaremos los circuitos siguiendo pasos o secuencias establecidas. Para este caso solo nos interesa visualizarlo y por ello se presentó el diagrama lógico.

3.9. Diagramas esquemáticos

El diagrama esquemático es el diagrama más próximo a la implementación física del circuito digital. Consiste en trasladar las compuertas que hay en el diagrama lógico a su respectivo esquema del circuito integrado.

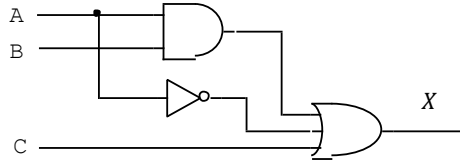
Ejemplo: efectuar la implementación de la función $f(A,B,C) = A + \bar{B}C + \bar{A}\bar{B}\bar{C}$ en diagrama esquemático si se cuenta con los siguientes circuitos integrados: 7404, 7408 y 7432.



Observe que no todas las compuertas lógicas de los circuitos integrados se están utilizando, solo se emplean los que fueron seleccionados de acuerdo a su funcionalidad y el acabado final del circuito digital. Sin embargo, se debe tener en cuenta que si hubiéramos elegido otra compuerta del mismo integrado, el circuito digital seguiría funcionando y proveyendo la misma salida.

Problemas Resueltos N° 03

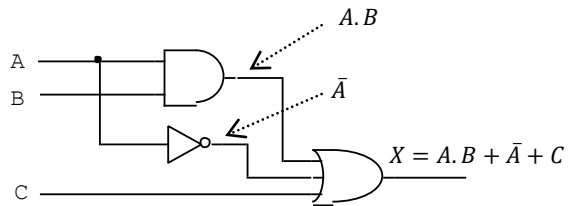
1. Escriba la expresión de la función lógica del siguiente circuito lógico. Utilice una tabla de verdad para determinar todas las posibles combinaciones de las entradas.



Solución:

Para escribir la expresión lógica de la función, se escribe a la salida de cada compuerta la función lógica de sus entradas. Esta operación se realiza de izquierda a derecha.

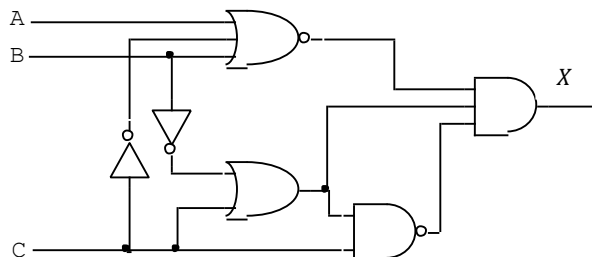
Las salidas de los circuitos lógicos combinatorios responden a cada combinación de los valores de sus variables de entrada; es por ello, que mediante una tabla de verdad se puede determinar todas las posibles combinaciones.



Considerando a la variable lógica A como el bit más significativo (MSB) tenemos la siguiente tabla de verdad.

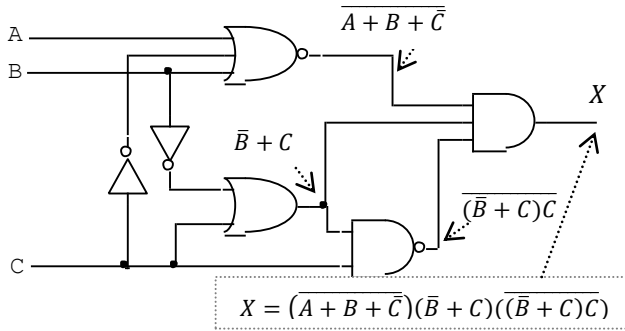
Nº	ABC	A	B	\bar{A}	$f(A,B,C)$
0	000	0	1	0	1
1	001	0	1	1	1
2	010	0	1	0	1
3	011	0	1	1	1
4	100	0	0	0	0
5	101	0	0	1	1
6	110	1	0	0	1
7	111	1	0	1	1

2. Halle la función lógica del siguiente circuito lógico. Determine el valor de la función X en todas las posibles combinaciones de sus variables de entrada.



Solución:

En la salida de cada compuerta lógica se hallará su respectiva función.



Las compuertas lógicas que tienen una pequeña circunferencia en su salida, niegan su salida.

Las entradas negadas en las compuertas se representan mediante compuertas NOT;

también se pueden representar con pequeñas circunferencias en las entradas.

La salida de la función lógica proviene de una compuerta NAND de tres entradas, es por ello, que la función X, está compuesto por el producto de tres términos.

La siguiente tabla visualiza todas las posibles combinaciones.

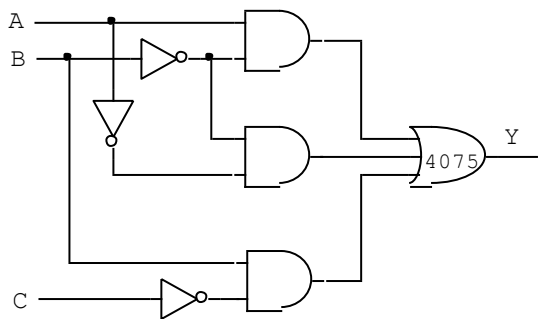
Nº	ABC	$(A + B + \bar{C})$	$(\bar{B} + C)$	$(\overline{(\bar{B} + C)C})$	$X = f(A,B,C)$
0	0 0 0	1	1	1	1
1	0 0 1	0	1	0	1
2	0 1 0	1	0	1	1
3	0 1 1	1	1	0	1
4	1 0 0	1	1	1	0
5	1 0 1	1	1	0	1
6	1 1 0	1	0	1	1
7	1 1 1	1	1	0	1

3. Dada la función $Y = A\bar{B} + \bar{A}\bar{B} + B\bar{C}$ construya el circuito lógico utilizando compuertas AND, OR e Inversores.

Solución:

Se observa que la función lógica Y tiene tres variables (A, B y C); además, es la suma de tres términos producto, es decir: $A\bar{B}$, $\bar{A}\bar{B}$ y $B\bar{C}$.

Primero se construyen las compuertas AND de dos entradas y luego cada una de éstas se enlazan a una compuerta OR.

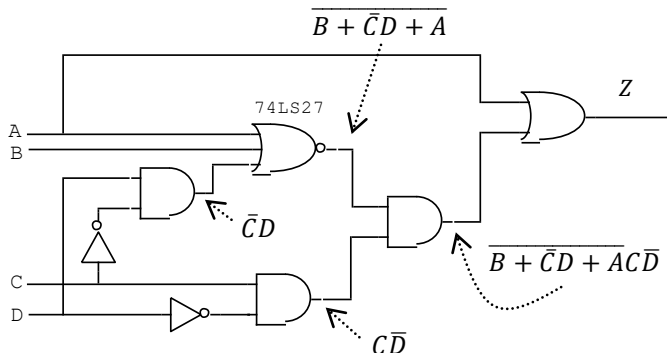


La compuerta NOT se utiliza para negar o invertir los valores de una variable.

4. Construya el circuito lógico y la tabla de verdad de la siguiente expresión lógica.
 $Z = A + B + \overline{C}D + AC\overline{D}$

Solución:

Se observa que la función Z responde a cuatro variables (A , B , C y D); además, la función lógica está sumando dos términos. El primer término es: A ; el segundo es: $B + \overline{C}D + AC\overline{D}$. La



función lógica, por tanto, está constituida por una compuerta OR de dos entradas.

Para construir el segundo término, nos damos cuenta que se multiplica dos términos; uno de ellos es: $B + \overline{C}D + A$ y el otro: $C\overline{D}$.

Nº	ABCD	A	$B + \overline{C}D + A$	$C\overline{D}$	$Z = f(A,B,C)$
0	0000	0	1	0	0
1	0001	0	0	0	0
2	0010	0	1	1	1
3	0011	0	1	0	0
4	0100	0	0	0	0
5	0101	0	0	0	0
6	0110	0	0	0	0
7	0111	0	0	0	0
8	1000	1	0	0	1
9	1001	1	0	0	1
10	1010	1	0	0	1
11	1011	1	0	0	1
12	1100	1	0	0	1
13	1101	1	0	0	1
14	1110	1	0	0	1
15	1111	1	0	0	1

5. Elabore una tabla de verdad, diagrama lógico y diagrama esquemático de la siguiente expresión booleana:

$$M = \overline{A \oplus B} AC + BC$$

Solución:

Tabla de verdad.

Nº	ABC	$\overline{A \oplus B} AC + BC$	$M = f(A,B,C)$
0	000	1 00 0 0	1
1	001	1 00 0 0	1
2	010	0 00 0 0	1
3	011	0 00 1 1	0
4	100	0 01 0 0	1
5	101	0 00 0 0	1
6	110	1 11 1 0	0
7	111	1 00 1 1	0

Diagrama lógico.

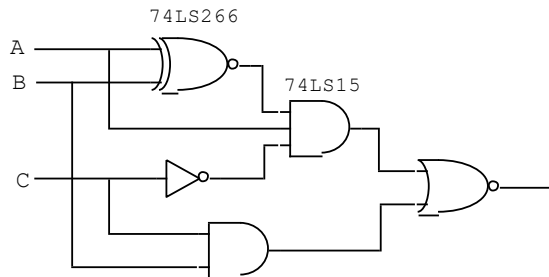
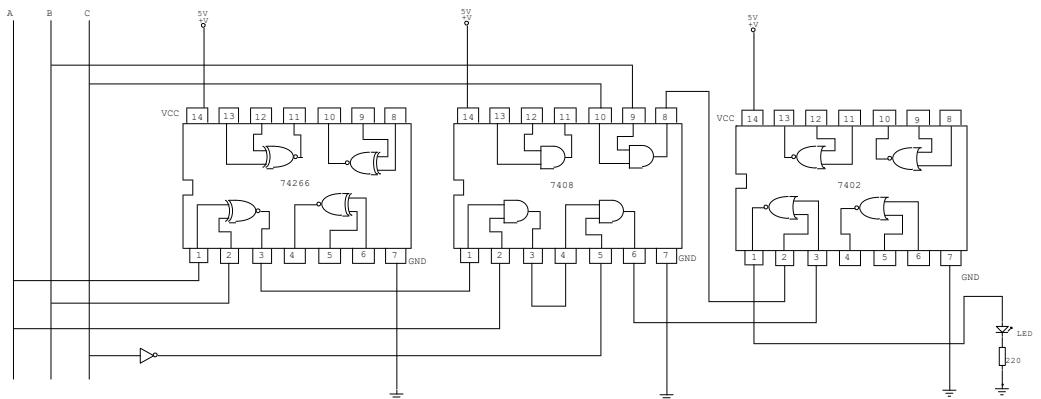


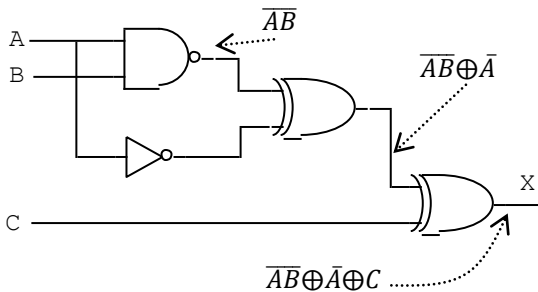
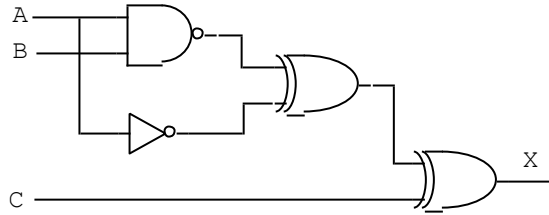
Diagrama esquemático



6. Trace el diagrama de tiempo de la función X del circuito lógico mostrado, según las secuencias de las variables de entrada (A , B y C) dadas.

Solución:

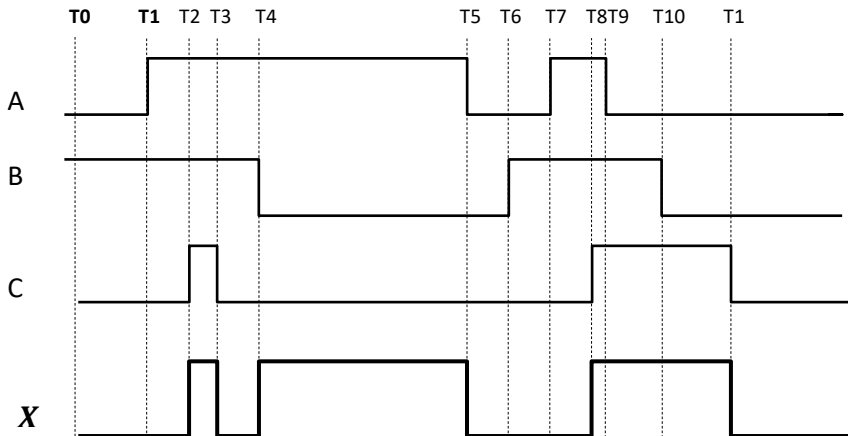
Primero debemos hallar la función X del circuito lógico mostrado; luego mediante una tabla de verdad se encuentra todas las posibles combinaciones que se pueden establecer con los valores de las variables. Se considera a la variable A como MSB.



Nº	ABC	$(\bar{A} \cdot \bar{B} \oplus \bar{A} \oplus C)$	X
0	000	1 0 1 0	0
1	001	1 0 1 1	1
2	010	1 0 1 0	0
3	011	1 0 1 1	1
4	100	1 1 0 0	1
5	101	1 1 0 1	0
6	110	0 0 0 0	0
7	111	0 0 0 1	1

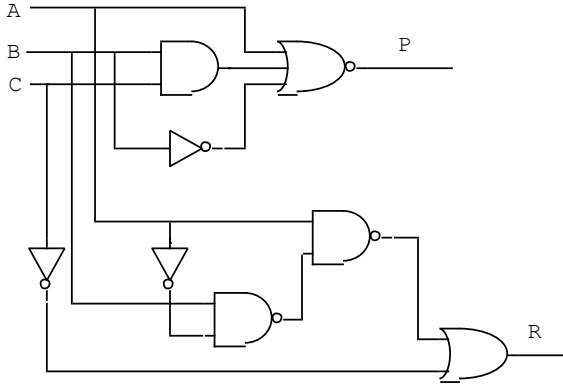
Diagrama de tiempo.

La salida X , variará según como se encuentren las entradas, por ejemplo, entre los tiempos T_0 y T_1 , las variables $A B C$ se encuentran respectivamente en **0 1 0** (fila 2 de la tabla), entonces $X = 0$.



Problemas Propuestos N° 03

- Se sabe que $Q = P + \bar{R}$. Encuentre la función lógica Q y construya su tabla de verdad.

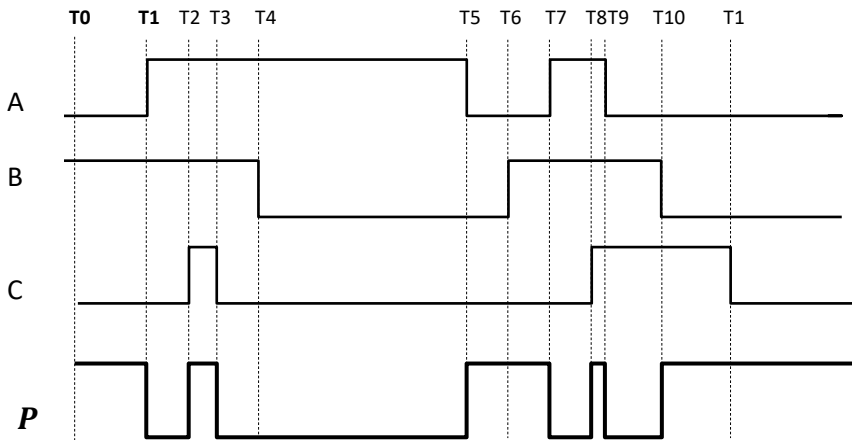


- Construya el diagrama lógico, tabla de verdad y diagrama esquemático de la siguiente función lógica.

$$S = \overline{A\bar{B}\bar{C}} + \overline{\bar{B}C} + \overline{A\bar{B}C}$$

- Dado el siguiente Diagrama de tiempo, Halle la función P en una tabla de verdad. Considere a la variable C como LSB.

Diagrama de tiempo.

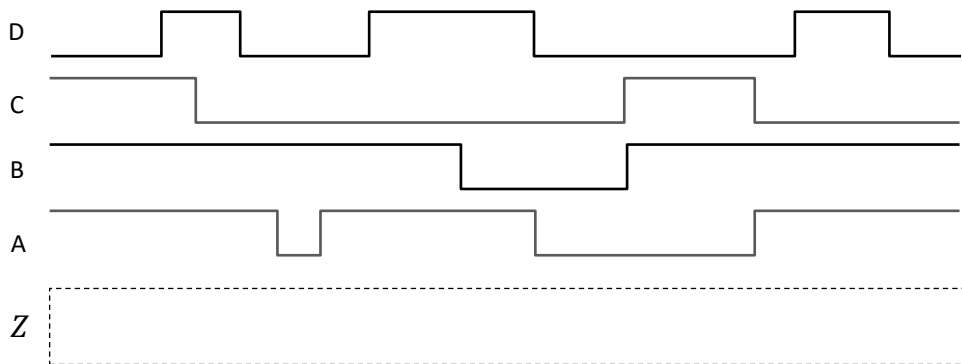


4. Dada la función lógica Z expresada en la tabla de verdad siguiente. Encuentre el diagrama de tiempo de la función teniendo en cuenta que la variable A es **LSB**.

Tabla de verdad.

Nº	$D C B A$	$Z = f(D,C,B,A)$
0	00 00	1
1	00 01	0
2	00 10	0
3	00 11	0
4	01 00	1
5	01 01	1
6	01 10	1
7	01 11	1
8	10 00	0
9	10 01	0
10	10 10	0
11	10 11	1
12	11 00	1
13	11 01	0
14	11 10	0
15	11 11	1

Diagrama de tiempo



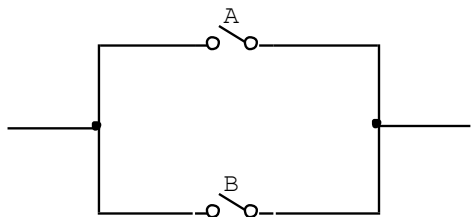
Capítulo 4

Álgebra de Boole

4.1. Definición

El Álgebra de Boole, creada por el matemático inglés George Boole, es una estructura que relaciona la respuesta de un circuito lógico frente a la influencia de las variables de entrada, mediante dos operaciones denominadas “suma lógica” y “producto lógico”. Los teoremas del Álgebra de Boole se utilizan para simplificar funciones lógicas manipulando las expresiones de manera similar al algebra ordinaria, teniendo en cuenta que los valores que adoptan las variables no son numéricas sino valores ó niveles lógicos.

Considerando la lógica de contactos de la sección 3.4; la suma lógica se representa por medio de un circuito paralelo, mientras que el producto por un circuito serie.



(a)



(b)

Fig. 4.1.

(a) La suma lógica de $(A+B)$ representado por un circuito paralelo. Interpretado como $(A \text{ ó } B)$

(b) El producto lógico $(A.B)$ representado por un circuito serie. Interpretado como $(A \text{ y } B)$.

4.2. Expresiones de una función lógica

Una función lógica denotada mediante las operaciones de suma y producto lógico se pueden expresar de dos formas: como Suma de productos y como Producto de sumas.

Suma de productos

$$f(A,B,C) = A\bar{B} + \bar{A}C + B\bar{C}$$

$$f(A,B,C,D) = A\bar{B}C + B\bar{C}D + A\bar{C} + \bar{A}$$

El término \bar{A} , es un término producto, ya que \bar{A} se está multiplicando por 1. Así: $(\bar{A} \cdot 1) = \bar{A}$

Producto de sumas

$$f(A,B,C) = (A + \bar{B})(A + \bar{C})(B + C)$$

$$f(A,B,C,D) = (B + C + \bar{D})(A + B)(\bar{B} + D) \bar{A}$$

El término \bar{A} , es un término suma, ya que \bar{A} se está sumando con 0. Así: $(\bar{A} + 0) = \bar{A}$

4.3. Teoremas del Álgebra de Boole

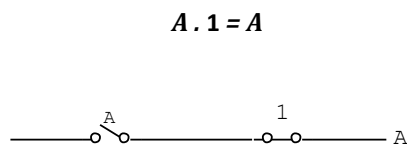
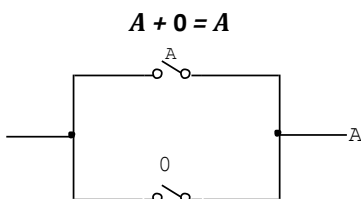
Los teoremas del Álgebra de Boole son herramientas muy potentes para la simplificación de funciones lógicas. Estudiaremos once propiedades cuya aplicación nos ayudará en la reducción de compuertas y circuitos integrados. Las cinco primeras propiedades (conocidos también como postulados) conllevan a la demostración de las siguientes propiedades.

Las demostraciones de las siguientes propiedades se pueden realizar mediante tabla de verdad, lógica de contactos ó algebraicamente.

4.3.1. Propiedad de Identidad

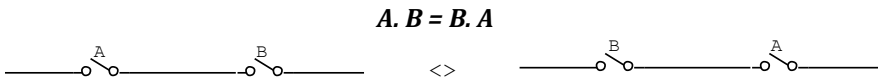
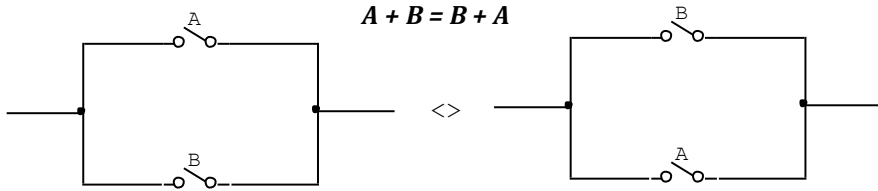
Existen dos elementos de identidad, uno para la suma y otro para el producto. El elemento identidad de la **suma lógica** (+) es **0**; y el elemento identidad del **producto lógico** (.) es **1**.

Al sumar **0** a una variable cualquiera obtenemos la misma variable, del mismo modo, si una variable se multiplica por **1**, se obtiene la misma variable.



4.3.2. Propiedad conmutativa

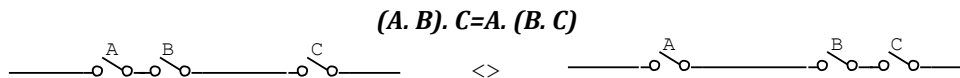
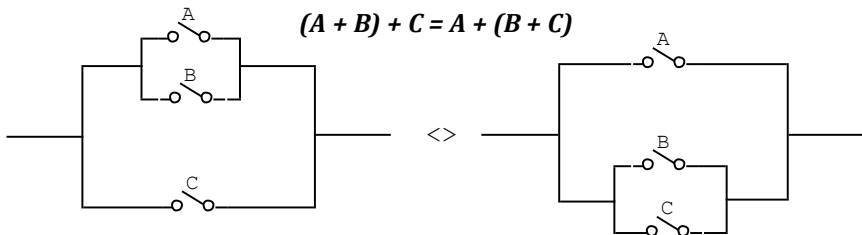
Si al intercambiar la ubicación de cualquier par de variables, el resultado final no cambia. Se cumple tanto para la suma como para el producto.



4.3.3. Propiedad asociativa

Si sumamos dos variables ($A+B$) y luego le sumamos la variable C , es igual a sumar la variable A con la suma de ($B+C$).

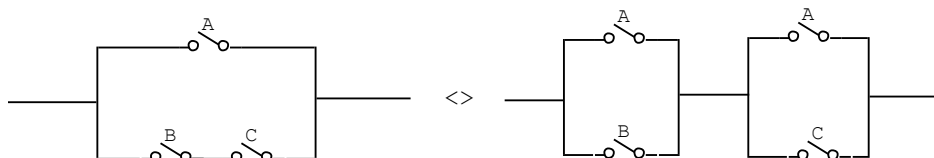
El procedimiento para el producto es análogo a la suma. Esto quiere decir que la propiedad asociativa en el Álgebra de Boole es idéntica al álgebra convencional.



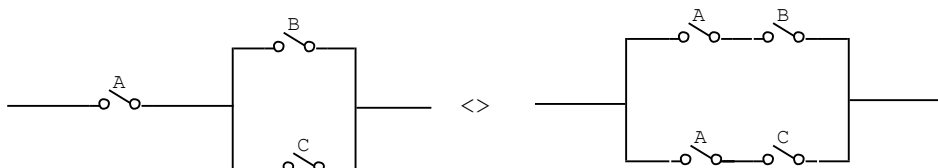
4.3.4. Propiedad distributiva

La propiedad distributiva se define respecto a las operaciones de suma o producto. Esto quiere decir que la operación suma es distributiva respecto a producto y viceversa.

$$A + (B \cdot C) = (A + B) \cdot (A + C)$$



$$A \cdot (B + C) = A \cdot B + A \cdot C$$

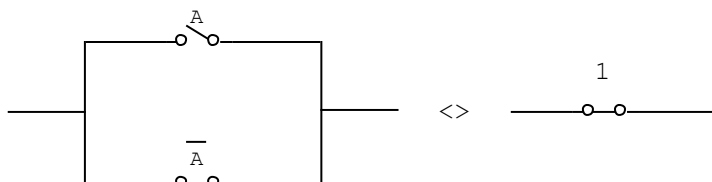


4.3.5. Propiedad del complemento

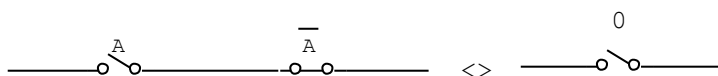
Toda variable lógica tiene su opuesto. Como una variable adopta un único valor de los dos posibles que existe, el opuesto o complemento de la variable adoptará el otro valor. Es decir, si $A = 1$, el complemento de A será igual a 0 .

El complemento de una variable se denota con una barra en la parte superior. De esta manera decimos que el complemento de A es \bar{A} . También se puede decir que el negado de A es \bar{A} .

$$A + \bar{A} = 1$$



$$A \cdot \bar{A} = 0$$



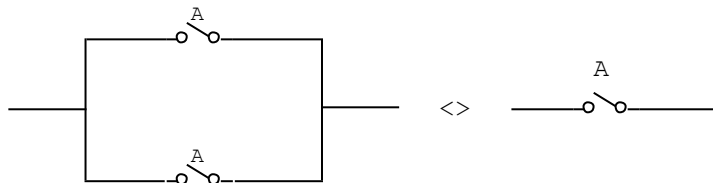
IMPORTANTE

A partir de las propiedades o postulados anteriores es factible la comprobación de otras cuya importancia se aprecia en la simplificación de funciones lógicas.

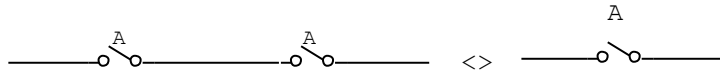
4.3.6. Propiedad de idempotencia

Cuando una variable se suma o se multiplica consigo misma obtenemos la misma variable.

$$A + A = A$$



$$A.A = A$$



4.3.7. Propiedad de unicidad del complemento

De la propiedad del complemento se concluye que el único complemento de la variable A es \bar{A} ; pues al existir sólo dos valores posibles, podemos deducir que si la variable A adopta el valor **1**, su complemento tomará el valor **0**, no hay otra posibilidad; por ello decimos que el complemento de una variable es única.

4.3.8. Propiedad de los elementos dominantes

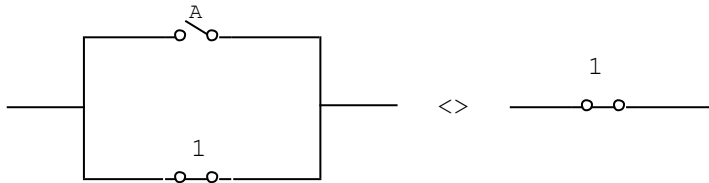
Los elementos dominantes de la suma y el producto lógicos, se deducen de la función OR y AND respectivamente.

Es suficiente que exista un **1** en la suma de productos para que la suma sea **1**. Lo mismo ocurre en el producto de sumas, basta que exista un **0**, para que el producto sea **0**.

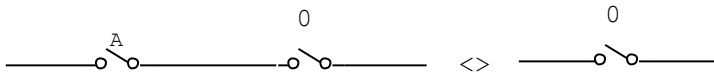
En conclusión:

El **1** es el elemento dominante en la suma y el **0** es el elemento dominante en el producto.

$$A + 1 = 1$$



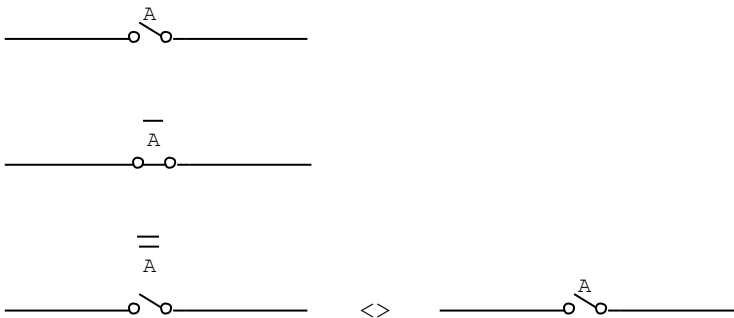
$$A \cdot 0 = 0$$



4.3.9. Propiedad involutiva

Cuando se complementa el complemento de una variable obtenemos la misma variable. Esto quiere decir, si la variable A adopta el valor **0**, su complemento \bar{A} será igual a **1**. Si nuevamente complementamos a \bar{A} , entonces $\overline{\bar{A}}$ será igual a **0**, lo cual es igual a A .

$$\overline{\bar{A}} = A$$



4.3.10. Propiedad de absorción

En la expresión $A + A \cdot B = A$, decimos que la variable A absorbe a la variable B , pues como vemos, la variable A aparece relacionándose con la variable B , luego en el resultado sólo queda A . De igual modo en $A \cdot (A + B) = A$.

Esta propiedad importante lo podemos comprobar algebraicamente o mediante la lógica de contactos.

Algebraicamente.

$A + AB = A$ factorizando A en el primer miembro tenemos;

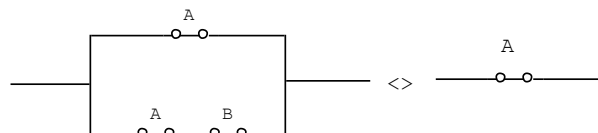
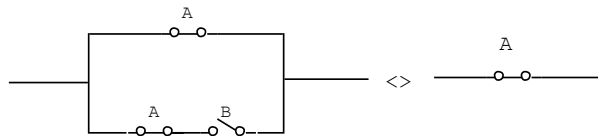
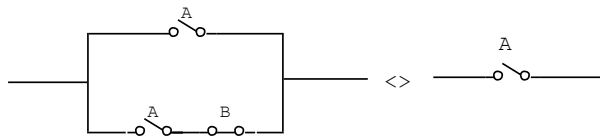
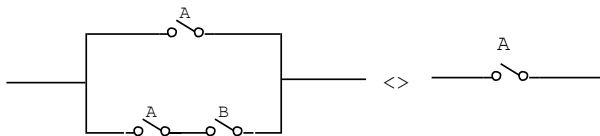
$A(1 + B) = A$ como el factor $(1 + B)$ es igual a 1 . Por la propiedad del elemento dominante.

$A \cdot 1 = A$ teniendo en cuenta la propiedad de identidad, tenemos:

$A = A$

Lógica de contactos

$$A + AB = A$$



4.3.11. Propiedad de consenso

La propiedad de consenso se expresa como suma $A + \bar{A} \cdot B = A + B$ ó como producto $A \cdot (\bar{A} + B) = A \cdot B$

Esta propiedad se comprueba algebraicamente o mediante la lógica de contactos.

Algebraicamente.

$A + \bar{A} \cdot B = A + B$ se aplica la propiedad distributiva respecto al producto en el primer miembro.

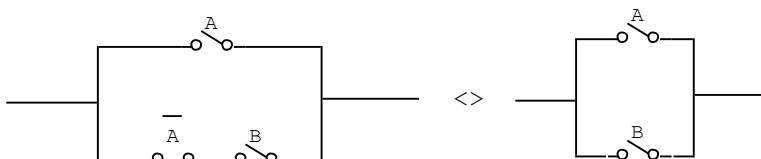
$(A + \bar{A})(A + B) = A + B$ por la propiedad del complemento $(A + \bar{A}) = 1$

$1 \cdot (A + B) = A + B$ por la propiedad de identidad.

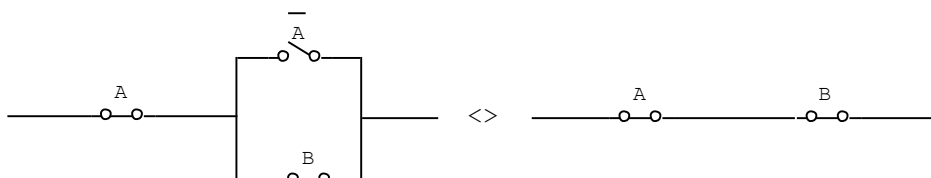
$(A + B) = A + B$

Lógica de contactos

$$A + \bar{A}B = A + B$$



$$A \cdot (\bar{A} + B) = A \cdot B$$



4.4. Teoremas De Morgan

Los teoremas De Morgan, se deben a su creador, el matemático inglés Augusto De Morgan. Estos teoremas consisten en convertir una función lógica suma en producto y viceversa con solo negar cada una de las variables intervinientes.

$$\overline{A + B} = \bar{A} \cdot \bar{B}$$

$$\overline{\bar{A} \cdot \bar{B}} = A + B$$

La demostración de estos dos teoremas se realizará con tablas de verdad.

N°	A	B	A + B	$\overline{A + B}$	\bar{A}	\bar{B}	$\bar{A} \cdot \bar{B}$
0	0	0	0	1	1	1	1
1	0	1	1	0	1	0	0
2	1	0	1	0	0	1	0
3	1	1	1	0	0	0	0



Ambas expresiones tienen el mismo resultado

N°	A	B	A . B	$\overline{\bar{A} \cdot \bar{B}}$	\bar{A}	\bar{B}	$\bar{A} + \bar{B}$
0	0	0	0	1	1	1	1
1	0	1	0	1	1	0	1
2	1	0	0	1	0	1	1
3	1	1	1	0	0	0	0



Ambas expresiones tienen el mismo resultado

Sin embargo, en lugar de las variables podría incluirse toda una expresión lógica como en $\overline{(\bar{A} + B \cdot C) + A \bar{B}}$.

$$\overline{(\bar{A} + B \cdot C) + A \bar{B}} = \overline{(\bar{A} + BC)} \cdot \overline{A \bar{B}}$$

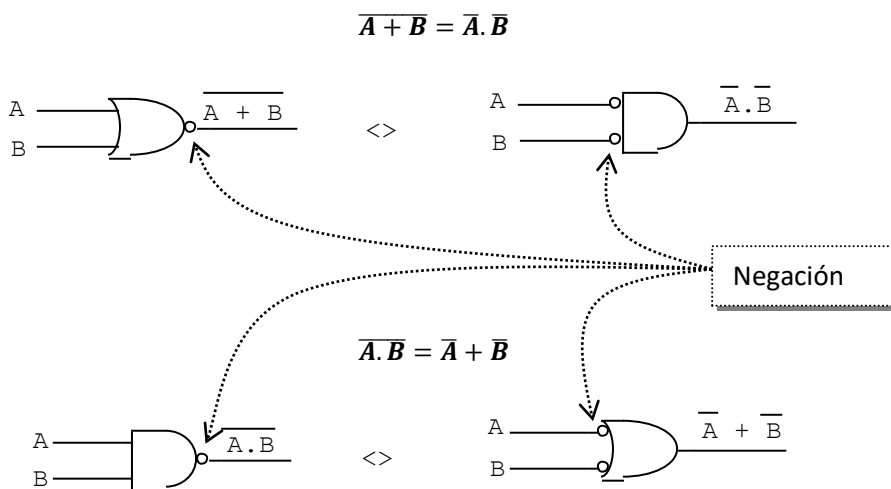
$$= \bar{A} \cdot \overline{(BC)} \cdot (\bar{A} + \bar{\bar{B}})$$

$$\overline{(\bar{A} + B \cdot C) + A \bar{B}} = A \cdot (\bar{B} + \bar{C}) \cdot (\bar{A} + B)$$

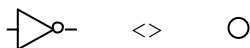
4.4.1. Efecto del Teorema de Morgan en las compuertas AND y OR.

El efecto más notable de éste teorema en las compuertas AND y OR, consiste en obtener una función producto de una función suma y una función suma de un producto. Este procedimiento se logra simplemente negando la función primaria, es decir, convirtiendo la función OR en NOR y la función AND en NAND.

Este efecto se visualiza mejor en sus correspondientes compuertas lógicas, tal como se presenta a continuación.



Las pequeñas circunferencias que se observan en las salidas y entradas de las compuertas indican negación, esto es, en lugar de utilizar una compuerta NOT en forma de triángulo con una circunferencia en su vértice de la salida, sólo se está utilizando la circunferencia.



4.5. Versatilidad de las compuertas NAND y NOR

La versatilidad de las compuertas NAND y NOR se presenta en el diseño y en la implementación de circuitos digitales, ya que en muchas ocasiones se requiere el uso casi exclusivo de un solo tipo de compuertas lógicas; es allí donde las compuertas NAND y NOR presentan su capacidad de adaptarse con facilidad a diversas funciones lógicas.

Una función (OR) se puede convertir fácilmente a función (NAND), aplicándole primero **la propiedad de involución** para que la función OR no cambie y luego a la negación interior (barra interior) **el Teorema De Morgan**.

$$A + B = \overline{\overline{A + B}}$$

De igual manera una función producto (AND) se convierte a una función NOR aplicando involución y Morgan.

Por ejemplo: convertir la siguiente función lógica a una equivalente sólo con compuertas NAND.

$$f(A, B, C) = B\bar{C} + \overline{(A + C)}$$

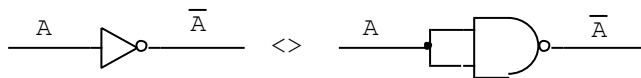
aplicando la propiedad de involución para que la expresión lógica no cambie.

Tenemos:

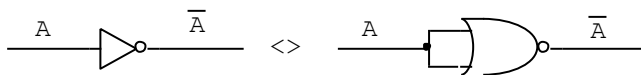
$$\begin{aligned} f(A, B, C) &= \overline{\overline{B\bar{C} + (A + C)}}, \text{ desarrollando la barra interior} \\ &= \overline{\overline{B\bar{C}} \cdot \overline{(A + C)}}, \text{ desarrollando la barra interior de } \overline{(A + C)} \\ &= \overline{\overline{B\bar{C}} \cdot (\bar{A} \cdot \bar{C})} \end{aligned}$$

Se observa que al término $\overline{B\bar{C}}$ de la expresión final, no se aplicó el teorema De Morgan, asimismo, al término $(\bar{A} \cdot \bar{C})$; pues, por conveniencia sólo se requiere una expresión con compuertas NAND, ya sea que sus términos productos estén complementados o no.

La inversión o negación de una variable se realiza puentando las entradas de una compuerta NAND, ya que de esta manera obtenemos una sola entrada con su respectiva salida.



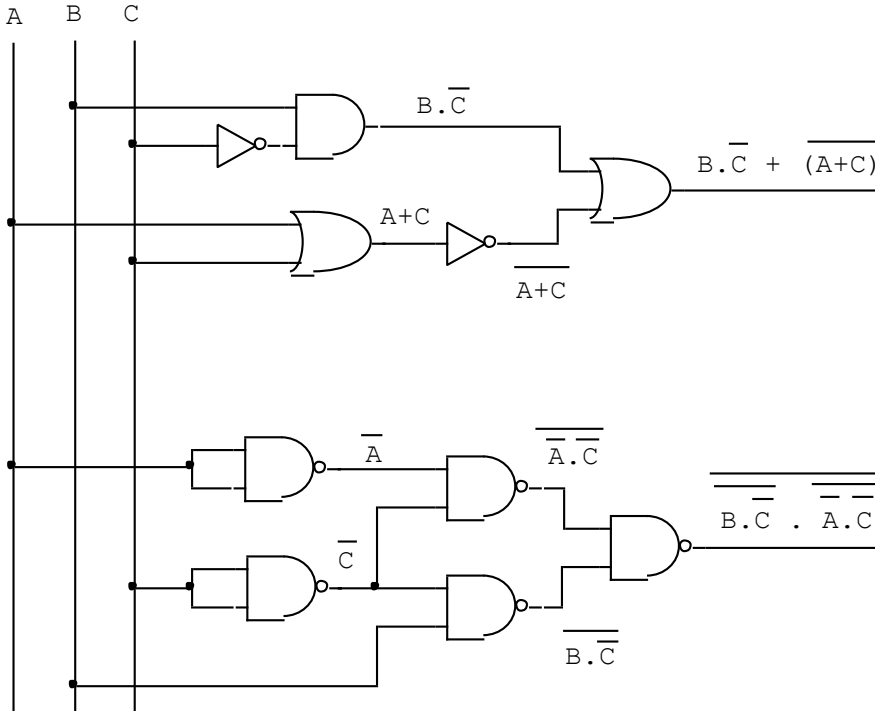
Del mismo modo se puede construir una compuerta NOT con una compuerta NOR.



La variable A no solo puede ser cualquier otra variable, sino, también toda una función que se requiera complementar o negar.

Representación lógica de:

$$B\bar{C} + \overline{(A + C)} = \overline{\overline{B\bar{C}} \cdot \overline{\overline{A + C}}}$$



La ventaja sustantiva de utilizar sólo compuertas NAND, se evidencia en la utilización de circuitos integrados. Por ejemplo, la expresión del primer miembro $B\bar{C} + \overline{(A + C)}$ con su respectivo diagrama lógico, requiere tres circuitos integrados: un C.I. 7404, un C.I. 7408 y un C.I. 7432; sin embargo, la expresión del segundo miembro $\overline{\overline{B\bar{C}} \cdot \overline{\overline{A + C}}}$ con su diagrama lógico con compuertas NAND, requiere dos circuitos integrados 7400. Esta circunstancia permite no solo implementar circuitos digitales de tamaño reducido, sino también, el ahorro económico que conlleva.

Otra ventaja de utilizar un solo tipo de compuertas (NAND o NOR) se manifiesta en el hecho de mejorar la transferencia de señales, reduciendo el inconveniente de retardo de propagación en diferentes tipos de compuertas.

4.6. Simplificación de funciones lógicas

La simplificación de funciones lógicas es un procedimiento lógico matemático para optimizar los recursos materiales, económicos y fundamentalmente del esfuerzo que se realice en la implementación de circuitos digitales.

Para efectuar la simplificación de funciones hasta de cuatro variables emplearemos los teoremas del Álgebra de Boole, posteriormente, en el capítulo 5 aprovecharemos una herramienta visual muy potente para los mismos fines, no obstante ello, los teoremas del Álgebra de Boole tienen sus propios atributos que lo hacen indispensable.

Para simplificar funciones lógicas de manera sencilla, utilizaremos la factorización de términos en donde sea pertinente su aplicación.

Ejemplo: simplificar la siguiente función lógica.

$$f(A, B, C, D) = A\bar{B} + (\overline{ABD} + \overline{B+C})A\bar{B}$$

Solución 1: Aplicando la propiedad de absorción.

$$\begin{aligned} f(A, B, C, D) &= A\bar{B} + (\overline{ABD} + \overline{B+C})A\bar{B}; \text{ tiene la forma de } X + XY, \text{ donde } X = A\bar{B} \\ &= A\bar{B} + A\bar{B}(\overline{ABD} + \overline{B+C}); \text{ aplicando la propiedad de absorción,} \\ &= A\bar{B} \end{aligned}$$

Solución 2: Aplicando el método de factorización.

$$\begin{aligned} f(A, B, C, D) &= A\bar{B} + (\overline{ABD} + \overline{B+C})A\bar{B}; \text{ factorizando el término } A\bar{B}. \\ &= A\bar{B}[1 + (\overline{ABD} + \overline{B+C})]; \text{ por la propiedad del elemento dominante} \\ &\quad \text{en la expresión } [1 + (\overline{ABD} + \overline{B+C})] = 1 \\ &= A\bar{B} \end{aligned}$$

Es común utilizar con frecuencia las propiedades de absorción y de consenso en la simplificación de funciones. La correcta utilización de éstas y otras propiedades posibilitan simplificar dichas funciones a su mínima expresión.

Problemas Resueltos N° 04

1. Halle el resultado de las siguientes expresiones lógicas.

a. $AC + 0 =$

Solución:

El elemento identidad de la suma es el 0. Aplicando la **propiedad de identidad de la suma**, el resultado siempre será el término que se suma con 0.

$$AC + 0 = AC$$

b. $MN + 1 =$

Solución:

Al aplicar la **propiedad del elemento dominante de la suma** el resultado siempre será 1. Es decir, basta que un término de una expresión suma sea 1 para que toda la suma sea 1.

$$MN + 1 = 1$$

c. $PQR.\overline{PQR} =$

Solución:

El resultado de aplicar la **propiedad del complemento del producto** siempre será 0. La propiedad indica que el producto de un término (con una o varias variables) con su complemento es 0. El término en éste ejercicio es PQR y su complemento es \overline{PQR} .

$$PQR.\overline{PQR} = 0$$

d. $XY + YX =$

Solución:

Al aplicar la **propiedad de idempotencia de la suma** a la suma de un mismo término, el resultado es el término anterior. En este caso el primer término es XY , y el segundo es YX , que por cierto, es igual a XY por la propiedad conmutativa.

$$XY + YX = XY$$

e. $AB.0 =$

Solución:

El elemento dominante del producto es 0. Por tanto al aplicar la **propiedad del elemento dominante de un producto**, el resultado siempre es 0. En otras palabras, basta que exista un 0 en un producto para que el resultado sea 0.

$$AB.0 = 0$$

f. $XYZ.1 =$

Solución:

Toda expresión que se multiplique con 1, el resultado siempre será la expresión inicial; a ésta propiedad se conoce como: **propiedad de identidad del**

producto.

$$XYZ \cdot 1 = XYZ$$

g. $AB + AB + C =$

Solución:

A los dos primeros términos ($AB + AB$) se aplica la **propiedad idempotencia de la suma**, luego, este resultado se suma con C .

$$AB + AB + C = AB + C$$

h. $\overline{PQR} + PR + \overline{PQ} =$

Solución:

Se aplica la **propiedad de absorción a la suma** del primer y tercer término: $\overline{PQR} + \overline{PQ}$, cuyo resultado es \overline{PQ} , luego, éste resultado se suma con PR .

$$\overline{PQR} + PR + \overline{PQ} = \overline{PQ} + PR$$

i. $X + \overline{X} + \overline{XYZ} =$

Solución:

Evaluando los dos primeros términos con la ayuda de la **propiedad del complemento** se concluye que es 1, es decir: $X + \overline{X} = 1$, luego, basta que exista un 1 en la suma para que el resultado sea 1.

$$X + \overline{X} + \overline{XYZ} = 1$$

j. $AC + ABC =$

Solución:

Acomodando las variables del segundo término, de tal manera que se pueda aplicar la **propiedad de absorción**, tenemos: $AC + ACB = AC$.

$$AC + ABC = AC$$

k. $\overline{XY} + XYZ =$

Solución:

Se observa que se puede aplicar la **propiedad de consenso**, pues el primer término es \overline{XY} y su complemento $\overline{\overline{XY}}$ es XY . Por tanto.

$$\overline{XY} + XYZ = \overline{XY} + Z$$

l. $PQ(PQ + R) =$

Solución:

Aplicando la **propiedad de absorción** al producto de $PQ(PQ + R)$, tenemos:

$$PQ(PQ + R) = PQ$$

También se puede aplicar la propiedad distributiva, así: $PQ(PQ + R) = PQ \cdot PQ + PQR$

$$PQ + PQR = PQ$$

2. Evaluar la expresión: $f = \overline{AB} + \overline{AB}$

Solución:

La función **OR-EX**: $A \oplus B$, también se puede expresar así: $A \oplus B = \overline{AB} + \overline{AB}$, por tanto, la expresión $\overline{AB} + \overline{AB} = \overline{A \oplus B}$; aplicando el Teorema de Morgan al primer miembro tenemos:

$$\overline{AB} + \overline{AB} = \overline{AB} \cdot \overline{AB}$$

$\overline{AB} \cdot \overline{AB} = (\overline{A} + B)(A + \overline{B})$... Aplicando la propiedad distributiva tenemos:

$$(\overline{A} + B)(A + \overline{B}) = \overline{A}A + \overline{A}\overline{B} + AB + B\overline{B} = \overline{A}\overline{B} + AB, \text{ es decir, una } \mathbf{NOR-EX}.$$

Por tanto:

$$f = \overline{AB} + \overline{AB} = \overline{A \oplus B} = \mathbf{AB} + \overline{AB}$$

3. Simplifique las siguientes expresiones booleanas.

a. $Y = \overline{A}BC + \overline{A}\overline{B}\overline{C} + \overline{A}\overline{B}C + \overline{A}B\overline{C} + \overline{A}BC + ABC$

Solución:

Numerando los términos de la expresión.

$$Y = \overline{A}BC + \overline{A}\overline{B}\overline{C} + \overline{A}\overline{B}C + \overline{A}B\overline{C} + \overline{A}BC + ABC$$

Y es una función de seis términos productos, los cuales se suman lógicamente.

Extraemos factor común de los siguientes términos: 1 con 6; 2 con 3; y 4 con 5.

$Y = BC(\overline{A} + A) + \overline{B}\overline{C}(A + \overline{A}) + \overline{A}B(\overline{C} + C)$, aplicando a los paréntesis, la propiedad del complemento.

$Y = BC + \overline{B}\overline{C} + \overline{A}B$, el primer y segundo término constituye la expresión de una NOR EXCLUSIVA.

$$Y = \overline{B \oplus C} + \overline{A}B$$

b. $X = (A + BC)(\overline{A} + \overline{BC} + \overline{A}\overline{B}) + BC$

Solución:

La función X tiene dos términos que se suman lógicamente. El primer término está compuesto de dos términos que se multiplican lógicamente.

Se observa que en el término producto se puede aplicar la propiedad de consenso, es decir: $(A + BC)(\overline{A} + \overline{BC} + \overline{A}\overline{B}) = (A + BC)\overline{A}\overline{B}$

$X = (A + BC)\overline{A}\overline{B} + BC$, desarrollando el primer término utilizando la propiedad distributiva, tenemos:

$X = A\overline{A}\overline{B} + BC\overline{A}\overline{B} + BC$, eliminando el primer término ($A\overline{A}\overline{B}$) y luego aplicando la propiedad de absorción.

$$X = BC\overline{A}\overline{B} + BC = \mathbf{BC}$$

$$c. P = \overline{\overline{\overline{AC} + \overline{BC}} + \overline{AC}}$$

Solución:

En el interior de la barra superior hay dos términos que se suman lógicamente.

El primer término es: $\overline{AC} + \overline{BC}$ y el segundo es: \overline{AC} .

Por el Teorema de Morgan, tenemos:

$P = \overline{\overline{\overline{AC} + \overline{BC}} . \overline{AC}}$, aplicando propiedad involutiva al primer término, y el Teorema de Morgan al segundo.

$P = (\overline{AC} + \overline{BC})(A + \overline{C})$, desarrollando con la propiedad distributiva.

$P = \overline{ACA} + \overline{BCA} + \overline{ACC} + \overline{BC} \overline{C}$, simplificando por propiedad del complemento, los términos: 1, 3 y 4.

$$P = A\overline{BC}$$

$$d. Q = \overline{A(\overline{B + \overline{C}})}D . A + D$$

Solución:

Se observa que la función Q tiene dos términos: $\overline{A(\overline{B + \overline{C}})}D . A$ y D .

Desarrollando el primer término.

$$Q = (\overline{A} + (\overline{B + \overline{C}}) + \overline{D}) . A$$

$$Q = (\overline{A} + (B + \overline{C}) + \overline{D}) . A$$

$Q = \overline{A}A + AB + A\overline{C} + A\overline{D}$, introduciendo el segundo término D , tenemos:

$$Q = AB + A\overline{C} + \underbrace{A\overline{D} + D}$$

$$Q = \underbrace{AB + A\overline{C} + A} + D$$

$$Q = A + D$$

$$e. M = (A\overline{B} + \overline{C})(A\overline{B} + \overline{D})(\overline{A\overline{B}\overline{D}})$$

Solución:

Para simplificar la expresión anterior se debe desarrollar el tercer término:

$$M = (A\overline{B} + \overline{C})(A\overline{B} + \overline{D})(A\overline{B} + D)$$

También se debe tener en cuenta la propiedad distributiva:

$$(X + YZ) = (X + Y)(X + Z).$$

El primer y segundo término de la expresión tiene el mismo esquema de la propiedad distributiva.

$$M = (\overline{AB} + \overline{C})(\overline{AB} + \overline{D})(\overline{AB} + D)$$

$$M = (\overline{AB} + \overline{CD})(\overline{AB} + D)$$

$$M = \overline{AB} + \overline{C}DD = \overline{AB}$$

4. Utilice sólo compuertas NAND para implementar la siguiente función lógica.

$$f = P\overline{Q} + RS + \overline{P}S$$

Solución:

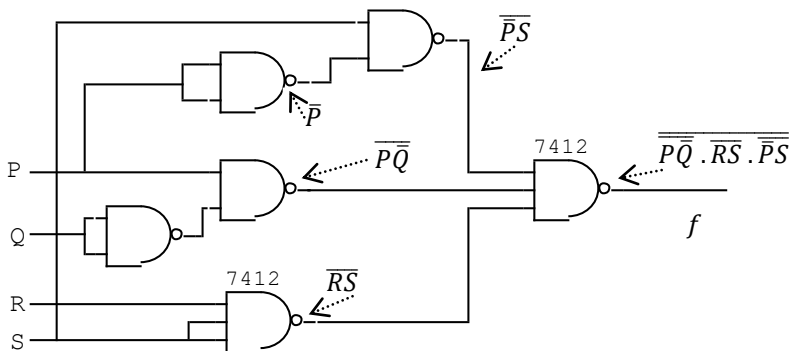
Para implementar funciones lógicas sólo con compuertas NAND, es preciso arreglar la función como suma de productos y luego aplicar la propiedad de involución para que la expresión final sea un producto.

La función f es una suma de productos, por tanto, aplicamos doble negación (propiedad de involución), para que la expresión inicial no cambie.

$f = \overline{\overline{P\overline{Q} + RS + \overline{P}S}}$, desarrollando la *barra interior*, tenemos:

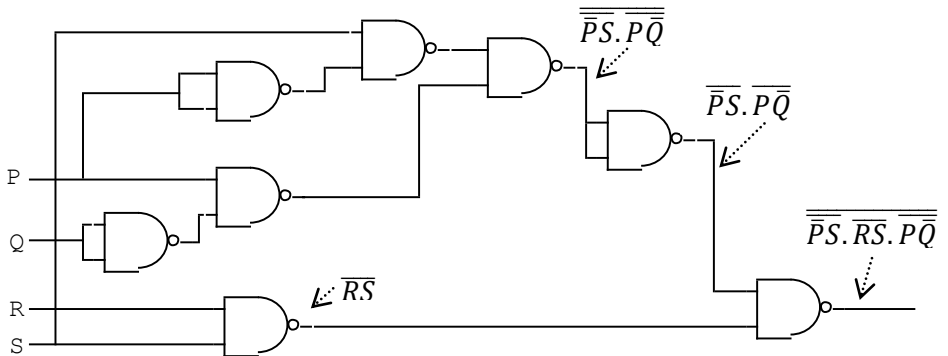
$$f = \overline{\overline{P\overline{Q}} \overline{RS} \overline{\overline{P}S}}$$

Para implementar el circuito digital, se empleará las compuertas NAND, inclusive para implementar la negación.



Con la expresión lógica inicial se requiere tres circuitos integrados: 7404, 7408, y 7432; mientras que, sólo con compuertas NAND, se requiere dos circuitos integrados, uno de ellos es el 7400 y el otro 7412 (compuertas NAND de tres entradas). Observe que, la realización del producto \overline{RS} de dos variables con una compuerta NAND de tres entradas, se puntea o se unen dos terminales o pines tal como se muestra en el diagrama.

También se puede implementar con dos circuitos integrados 7400. Como las compuertas NAND de éste circuito integrado tiene dos pines de entrada, el productos de los tres términos de la función se realiza en dos partes, es decir, primero se multiplica dos términos, se invierte para que la función no cambie, y luego, ésta salida se multiplica con el tercer término.



5. Utilice sólo compuertas NAND para implementar la siguiente función lógica.

$$Y = (P + \bar{Q}R)\bar{S}$$

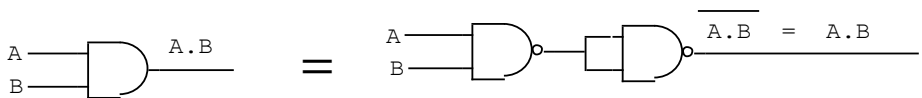
Solución:

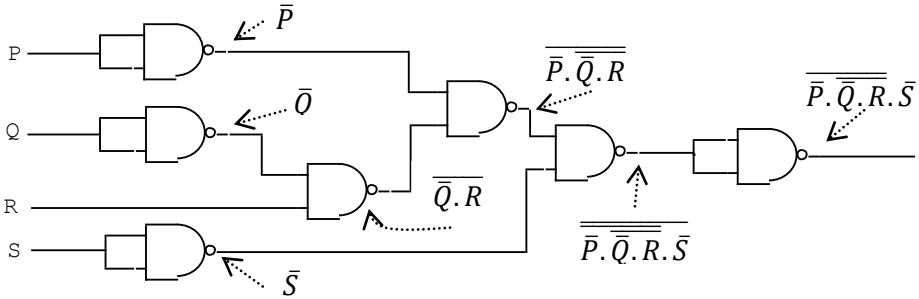
Se observa que la función Y tiene dos términos que se están multiplicando. Para implementar una función con sólo compuertas NAND, todos los términos que tienen suma se convierte a producto con la propiedad de involución. Por tanto, al producto de $(P + \bar{Q}R)$ por \bar{S} , no se aplica involución porque ya es un producto; sólo se aplica al primer término. Así:

$$Y = (\overline{P + \bar{Q}R})\bar{S}, \text{ desarrollando la barra interna, tenemos.}$$

$$Y = (\overline{P \cdot \bar{Q}R})\bar{S}$$

El producto simple se logra con un producto dos veces negado (involución). Así:





6. Utilice sólo compuertas NOR para implementar las siguientes funciones lógicas.

$$Z = (A + \bar{B})(B + \bar{C})(\bar{A} + C)$$

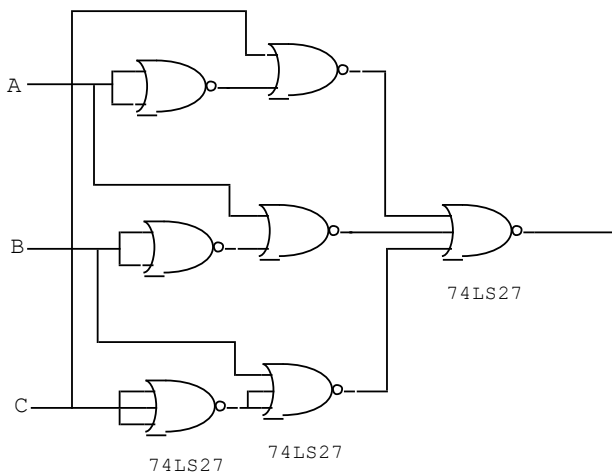
Solución:

Para implementar funciones lógicas sólo con compuertas NOR, es necesario acomodar la función como producto de sumas. Este arreglo es importante, pues, al aplicar la propiedad de involución a la expresión lógica y desarrollando la barra interna o (negación interna), los productos se convierten en sumas, y los términos sumas se complementan.

$$Z = \overline{\overline{(A + \bar{B})(B + \bar{C})(\bar{A} + C)}}$$

$$Z = \overline{\overline{(A + \bar{B})} + \overline{\overline{(B + \bar{C})} + \overline{\overline{(\bar{A} + C)}}}}$$

La implementación se puede realizar con dos circuitos integrados, uno de ellos, el 7402 (cuatro compuertas NOR de dos entradas) y el otro, el 7427 (tres compuertas NOR de tres entradas).



Problemas Propuestos N° 04

1. Halle el resultado de las siguientes expresiones lógicas.

a. $M = \overline{AB\bar{B} + C\bar{C}} + AB + C + \overline{\overline{B\bar{D}}} =$
Rpta: 1

b. $Y = (M + N)(M + \bar{N}) + \bar{M} =$
Rpta: 1

c. $Z = PR(PR + Q) \cdot \overline{P + \bar{R}} =$
Rpta: 0

d. $X = [(B + \overline{CD}) + (B + \overline{CD})A\bar{B}](\overline{B + \overline{CD}}) =$
Rpta: 0

2. Simplifique las siguientes expresiones lógicas.

a. $Z = M\bar{N}(\bar{M}\bar{N} + (P + NQ))P$
Rpta: $Z = \bar{N}P$

b. $Y = P\bar{Q}\bar{R} \overline{(\bar{P} + QR)} \cdot \overline{\bar{S}R} + \bar{Q} + \bar{R}$
Rpta: $Y = QR$

c. $F = (A\bar{B} + \overline{AD})(\overline{CD} + A\bar{B} + \overline{AD}) + A\bar{B} + \overline{AD}$
Rpta: $F = \bar{A} + \bar{B} + \bar{D}$

d. $P = (A\bar{C} + BD)(A\bar{C} + DC)(A + BCD)$
 e. *Rpta: $P = BCD$*

3. Implemente sólo con compuertas NAND la siguiente expresión lógica.

$$X = A + \bar{B}C + B\bar{C}\bar{D}$$

4. Implemente sólo con compuertas NOR la siguiente expresión lógica.

$$Y = (A + B\bar{C})(B + A\bar{D})(D + \bar{A}C)$$

Capítulo 5

Estandarización y métodos de simplificación de funciones

5.1. Formas canónicas de funciones lógicas

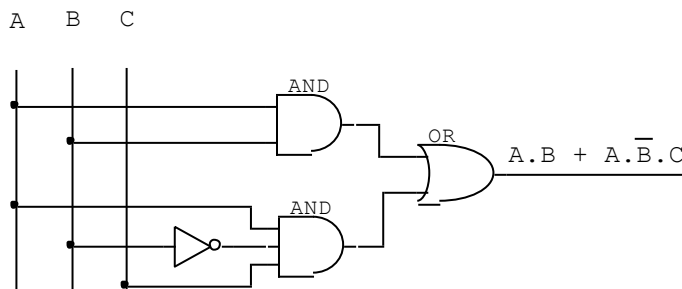
Para expresar cualquier función lógica se necesitan tan solo la suma y el producto lógico que interrelacionan a las variables intervinientes. Con estas dos operaciones, la suma (OR) y producto (AND), es posible describir todo un proceso lógico cuyas entradas son las variables y cuya salida es la respuesta de la función.

Cuando los términos de una operación son productos de variables y además cuando dichos términos se interrelacionan mediante sumas, decimos que se trata de una **Suma de Productos**.

Ejemplo:

- $AB + A\bar{B}C$
- $BC + \bar{D} + ABCD + A\bar{D}$
- $A\bar{C}D + BD + C\bar{D}$

Para implementar una suma de productos, se emplea compuertas **AND** para los términos producto y las salidas de éstos términos se enlazan con una compuerta **OR**.

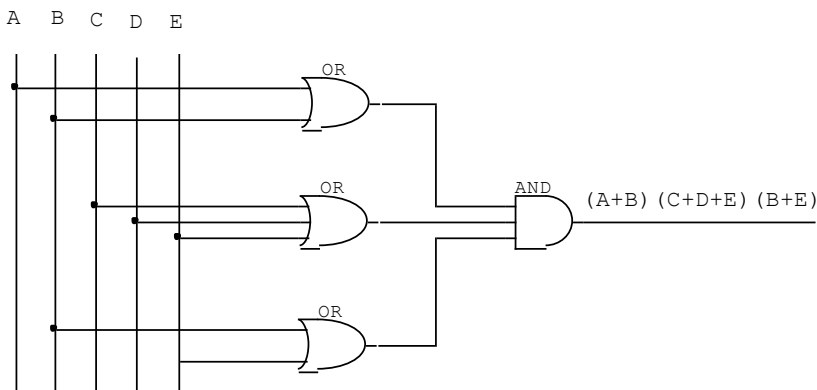


Si los términos son sumas de variables y si dichos términos se operan mediante productos, se trata de un **Producto de Sumas**.

Ejemplo:

- $(A + B)(C + D + E)(B + E)$
- $D(B + C)(A + B + C)$
- $(A + B + C + D)(B + C)(A + D)$

Para implementar un producto de sumas, se empleará compuertas **OR** para los términos; mientras que para la salida, se empleará una compuerta **AND**.



Cualquier expresión lógica puede ser transformada al formato Suma de productos o Producto de sumas aplicando simplemente el Álgebra de Boole.

Convertir las siguientes expresiones en suma o producto según se indique.

A suma de productos:

- a. $A(B + C\bar{D}) = AB + AC\bar{D}$
- b. $AC + B(CD + \bar{A}) = AC + BCD + \bar{A}B$
- c. $(A + \bar{B})(B + \bar{C} + D) = AB + A\bar{C} + AD + B\bar{B} + \bar{B}\bar{C} + \bar{B}D$
- d. $A + \bar{B} + \bar{C} = \bar{A} \cdot (\overline{B + C}) = \bar{A}(B + C) = \bar{A}B + \bar{A}C$

A producto de sumas:

- a. $A + B \cdot C \cdot \bar{D} = (A + B)(A + C \cdot \bar{D}) = (A + B)(A + C)(A + \bar{D})$
- b. $(B + C)(A + CD + \bar{B}) = (B + C)(A + \bar{B} + C)(A + \bar{B} + D)$
- c. $\overline{A \cdot B \cdot C} = \bar{A} + \bar{B}\bar{C} = \bar{A} + BC = (\bar{A} + B)(\bar{A} + C)$

IMPORTANTE:

- En un término suma o término producto, la barra de negación no puede extenderse a más de una variable.
- A todas las variables disponibles en una función lógica se denomina: **Dominio de la función lógica**. Éste se denota mediante una f y entre paréntesis todas las variables con que cuenta, separadas por comas, así: $f(A, B, C, D)$.
- Una expresión lógica se encuentra en su forma canónica o estandarizada, cuando cada término contiene a todas la variables del dominio, ya sea que las variables estén complementadas o no.

5.1.1. Expresiones canónicas en suma de productos “minitérminos”

Son aquellas en las que los **términos productos** de una expresión lógica contienen a todas las variables del dominio de la función.

Toda expresión en **formato suma de productos** se puede estandarizar utilizando la propiedad del complemento respecto de la suma, $A + \bar{A} = 1$. Esta propiedad del Álgebra de Boole tiene la cualidad de no alterar el término utilizado.

Un **término producto** cualquiera al que le falte una de las variables, se le multiplica por $A + \bar{A}$; de ésta manera luego de aplicar la propiedad distributiva obtendremos dos términos con la inserción de uno de ellos con la variable sin complementar, y la otra con la variable complementada.

Dada la siguiente función lógica en **formato suma de productos**, realizar la respectiva estandarización de dicha función.

$$\begin{aligned} f(A, B, C, D) &= \bar{A}B + B\bar{C}\bar{D} + ACD \\ &= \bar{A}B(C + \bar{C}) + B\bar{C}\bar{D}(A + \bar{A}) + ACD(B + \bar{B}) \\ &= \bar{A}BC + \bar{A}B\bar{C} + AB\bar{C}\bar{D} + \bar{A}B\bar{C}\bar{D} + ABCD + A\bar{B}CD \\ &= \bar{A}BC(D + \bar{D}) + \bar{A}B\bar{C}(D + \bar{D}) + AB\bar{C}\bar{D} + \bar{A}B\bar{C}\bar{D} + ABCD \\ &\quad + A\bar{B}CD \end{aligned}$$

$$\begin{aligned} f(A, B, C, D) &= \bar{A}BCD + \bar{A}BC\bar{D} + \bar{A}B\bar{C}\bar{D} + \bar{A}B\bar{C}\bar{D} + AB\bar{C}\bar{D} + \bar{A}B\bar{C}\bar{D} \\ &\quad + \underbrace{ABCD}_{\text{minitérminos}} + \underbrace{A\bar{B}CD}_{\text{minitérminos}} \end{aligned}$$

Cuando cada uno de los términos de una expresión lógica suma de productos que se hallan estandarizados se conoce con el nombre de “**minitérmino**”.

Los minitérminos iguales se simplifican fácilmente invocando la propiedad de idempotencia. Por ejemplo, en el ejercicio anterior el término $\overline{A}B\overline{C}D$, se repite dos veces, por lo tanto, quedará un solo término.

Para dar valor a cada uno de los minitérminos las variables sin complementar de cada uno de ellos adoptan el valor alto ó **1**, y las complementadas el valor bajo ó **0**.

El grupo de **1s** y **0s** formados en cada término se relacionan con el código binario natural, de tal manera que podemos asociarlos con sus respectivos números decimales.

$$\begin{array}{r}
 f(A, B, C, D) = \overline{A}BCD + \overline{A}BC\overline{D} + \overline{A}B\overline{C}D + A\overline{B}\overline{C}D + \overline{A}B\overline{C}\overline{D} + ABCD \\
 \begin{array}{cccccc}
 0111 & 0110 & 0101 & 1100 & 0100 & 1111 \\
 \mathbf{7} & \mathbf{6} & \mathbf{5} & \mathbf{12} & \mathbf{4} & \mathbf{15}
 \end{array} \\
 \\
 +\overline{A}B\overline{C}D \\
 \begin{array}{cccc}
 1011 \\
 \mathbf{11}
 \end{array}
 \end{array}$$

Cada uno de los minitérminos que tienen asociados un número decimal, se pueden representar mediante una sumatoria, en los que se incluyan los minitérminos.

$$f(A, B, C, D) = \sum_m (4, 5, 6, 7, 11, 12, 15)$$

Cada minitérmino adopta el valor lógico **1**, lo que expresado en la tabla de verdad se verifica que en cada fila en donde aparece el número decimal correspondiente equivale el valor lógico **1**, los que no aparece, el valor lógico **0**.

Para verificar lo anterior, construiremos una tabla de verdad teniendo en cuenta el dominio de la función, es decir, el número de filas tendrá que ser 16, ya que $2^{(\text{dominio de la función})} = 2^4 = 16$.

En la tabla de verdad se deberá tener en cuenta los términos de la función y la función propiamente dicha. Luego de procesar la tabla de verdad se deberá constatar que los minitérminos de la sumatoria deberán aparecer en la función con el valor lógico **1**.

Tabla de verdad para evaluar la función $f(A, B, C, D) = \bar{A}B + B\bar{C}\bar{D} + ACD$ y su equivalencia con la sumatoria de miniterminos $\sum_m(4,5,6,7,11,12,15)$.

N°	VARIABLES				TÉRMINOS DE LA FUNCIÓN			FUNCIÓN
	A	B	C	D	$\bar{A}B$	$B\bar{C}\bar{D}$	ACD	$\bar{A}B + B\bar{C}\bar{D} + ACD$
0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	0
2	0	0	1	0	0	0	0	0
3	0	0	1	1	0	0	0	0
4	0	1	0	0	1	1	0	1
5	0	1	0	1	1	0	0	1
6	0	1	1	0	1	0	0	1
7	0	1	1	1	1	0	0	1
8	1	0	0	0	0	0	0	0
9	1	0	0	1	0	0	0	0
10	1	0	1	0	0	0	0	0
11	1	0	1	1	0	0	1	1
12	1	1	0	0	0	1	0	1
13	1	1	0	1	0	0	0	0
14	1	1	1	0	0	0	0	0
15	1	1	1	1	0	0	1	1

5.1.2. Expresiones canónicas en producto de sumas “maxitérminos”

Son aquellas expresiones lógicas en las que los **términos suma** contienen a todas las variables del dominio de la función.

Toda expresión en **formato producto de sumas** se puede estandarizar utilizando la propiedad del complemento respecto al producto, $A.\bar{A} = 0$. Esta propiedad del Álgebra de Boole también tiene la cualidad de no alterar al **término suma**.

Un **término suma** cualquiera al que le falte una de las variables, se le suma $A.\bar{A}$; de ésta manera, luego de aplicar la propiedad distributiva obtendremos dos términos con la inserción de uno de ellos con la variable sin complementar, y la otra con la variable complementada.

Por ejemplo, en los término $(A + C + \bar{D})(B + \bar{C} + D)$, cuyo dominio son las variables A,B,C y D; se suma $B.\bar{B}$ para el primer término, y $A.\bar{A}$ para el segundo.

$(A + C + \bar{D} + B.\bar{B})(B + \bar{C} + D + A.\bar{A})$, resultando.

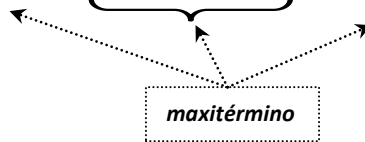
$(A + B + C + \bar{D})(A + \bar{B} + C + \bar{D})(A + B + \bar{C} + D)(\bar{A} + B + \bar{C} + D)$

Dada la siguiente función lógica en formato producto de sumas, realice la respectiva estandarización de dicha función.

$$f(A, B, C, D) = (\bar{A} + C)(\bar{B} + C + \bar{D})$$

$$= (\bar{A} + C + B.\bar{B} + D.\bar{D})(\bar{B} + C + \bar{D} + A.\bar{A})$$

$$f(A, B, C, D) = (\bar{A} + B + C + D)(\bar{A} + B + C + \bar{D})(\bar{A} + \bar{B} + C + D)(\bar{A} + \bar{B} + C + \bar{D})(A + \bar{B} + C + \bar{D})(\bar{A} + \bar{B} + C + \bar{D})$$



Cuando cada uno de los términos de una expresión lógica producto de sumas que se encuentran estandarizados se conoce con el nombre de “**maxitérmino**”.

Si hubiera el caso en que dos o más maxitérminos fueran iguales se procedería a simplificar fácilmente invocando la propiedad de idempotencia, como en el caso del término $(\bar{A} + \bar{B} + C + \bar{D})$.

Para dar valor a cada uno de los maxitérminos las variables sin complementar de cada uno de ellos adoptan el valor bajo ó **0**, y las complementadas el valor alto ó **1**.

El grupo de **1s** y **0s** formados en cada término se relacionan con el código binario natural, de tal manera que podemos asociarlos con sus respectivos números decimales.

$$f(A, B, C, D) = \left(\begin{array}{ccc} (\bar{A} + B + C + D)(\bar{A} + B + C + \bar{D})(\bar{A} + \bar{B} + C + D) \\ (1 \ 0 \ 0 \ 0) \ (1 \ 0 \ 0 \ 1) \ (1 \ 1 \ 0 \ 0) \\ \mathbf{8} \qquad \qquad \mathbf{9} \qquad \qquad \mathbf{12} \\ \\ (\bar{A} + \bar{B} + C + \bar{D})(A + \bar{B} + C + \bar{D}) \\ (1 \ 1 \ 0 \ 1) \ (0 \ 1 \ 0 \ 1) \\ \mathbf{13} \qquad \qquad \mathbf{5} \end{array} \right)$$

Cada uno de los maxitérminos que tienen asociados un número decimal, se pueden representar mediante un producto, en los que se incluyen los maxitérminos.

$$f(A, B, C, D) = \prod_M (5, 8, 9, 12, 13)$$

Cada maxitérmino adopta el valor lógico **0**, lo que expresado en la tabla de verdad se verifica que en cada fila en donde aparece el número decimal correspondiente equivale el valor lógico **0**, los que no aparece, el valor lógico **1**.

N°	VARIABLES				TÉRMINOS DE LA FUNCIÓN		FUNCIÓN
	A	B	C	D	$(\bar{A} + C)$	$(\bar{B} + C + \bar{D})$	$(\bar{A} + C)(\bar{B} + C + \bar{D})$
0	0	0	0	0	1	1	1
1	0	0	0	1	1	1	1
2	0	0	1	0	1	1	1
3	0	0	1	1	1	1	1
4	0	1	0	0	1	1	1
5	0	1	0	1	1	0	0
6	0	1	1	0	1	1	1
7	0	1	1	1	1	1	1
8	1	0	0	0	0	1	0
9	1	0	0	1	0	1	0
10	1	0	1	0	1	1	1
11	1	0	1	1	1	1	1
12	1	1	0	0	0	1	0
13	1	1	0	1	0	0	0
14	1	1	1	0	1	1	1
15	1	1	1	1	1	1	1

5.2. Método de simplificación de funciones lógicas

La aplicación del Álgebra de Boole y los teoremas De Morgan han permitido la simplificación de expresiones lógicas mediante la manipulación algebraica de sus términos.

La noción de minitérminos y maxitérminos posibilita realizar una nueva aproximación hacia la simplificación de funciones, haciendo uso de un proceso visual que lo hace en muchas ocasiones más potente que la manipulación algebraica. Este método gráfico se conoce como Mapas de Karnaugh.

5.2.1. Mapas de Karnaugh

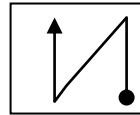
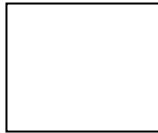
Los mapas de Karnaugh, inventado por el físico e ingeniero norteamericano Maurice Karnaugh, son herramientas gráficas que relacionan cada compartimento o celdas de un mapa con las filas de una tabla de verdad.

Los mapas de Karnaugh son paralelogramos (cuadrados o rectángulos) divididos simétricamente en celdas o compartimentos, los cuales se numeran y se relacionan con cada fila de una tabla de verdad. En cada celda se coloca el minitérmino o maxitérmino correspondiente. Si el dominio de la función tiene dos variables, entonces, el mapa de Karnaugh tendrá cuatro celdas ($\#celdas = 2^2 = 4$).

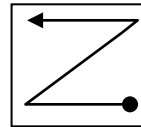
Cuando el dominio de la función (número de variables) es par, el mapa de Karnaugh adopta la forma de un cuadrado; si es impar, la forma de un rectángulo.

Si es cuadrado la numeración se realiza como una **N**. Si es rectángulo la numeración se realiza como una **Z**.

Si el **dominio es par** como: $f(A, B)$ ó $f(A, B, C, D)$



Si el **dominio es impar** como: $f(A, B, C)$ ó $f(A, B, C, D, E)$

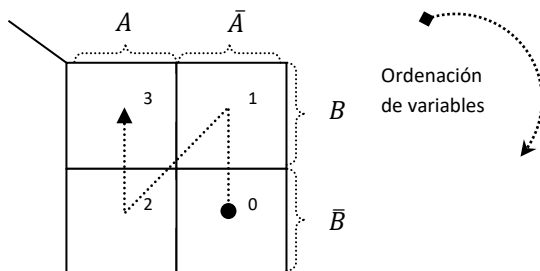


La numeración de las celdas se realiza siguiendo el trazo correspondiente y las numeraciones sucesivas se continúa como si fuera un espejo o un reflejo de la secuencia. Por ello decimos que un mapa de Karnaugh se construye en base al código Gray o Reflejado.

Cada mapa de Karnaugh contiene el dominio de la función, las cuales se distribuyen en el contorno de la misma en sentido horario, tanto la variable sin complementar como la complementada.

5.2.1.1. M.K. de dos variables

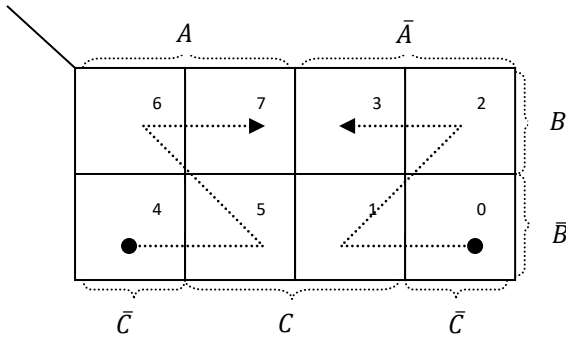
(# celdas = $2^2 = 4$)



N°	Variables		Función
	A	B	$f(A, B)$
0	0	0	
1	0	1	
2	1	0	
3	1	1	

5.2.1.2. M.K. de tres variables

(# celdas = $2^3 = 8$)

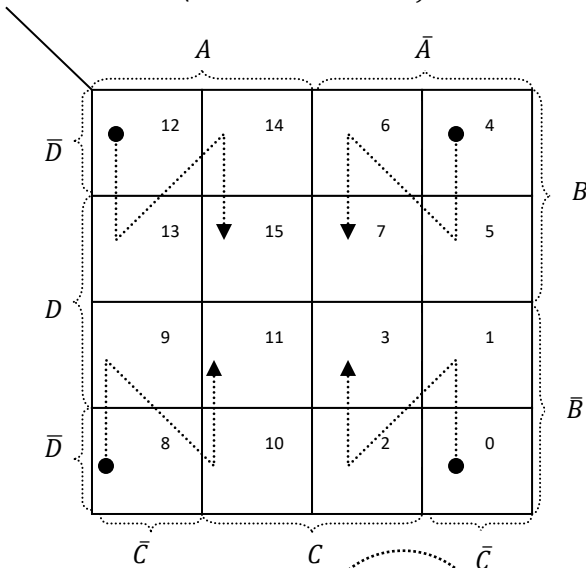


N°	Variables			Función
	A	B	C	$f(A,B)$
0	0	0	0	
1	0	0	1	
2	0	1	0	
3	0	1	1	
4	1	0	0	
5	1	0	1	
6	1	1	0	
7	1	1	1	

Ordenación de variables

5.2.1.3. M.K. de cuatro variables

(# celdas = $2^4 = 16$)



N°	Variables				Función
	A	B	C	D	$f(A,B)$
0	0	0	0	0	
1	0	0	0	1	
2	0	0	1	0	
3	0	0	1	1	
4	0	1	0	0	
5	0	1	0	1	
6	0	1	1	0	
7	0	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	

Ordenación de variables

5.3. Influencia de las variables en las celdas del M.K.

	A	\bar{A}	
B	3	1	
\bar{B}	2	0	

La variable A influye sobre las celdas (2 y 3);

\bar{A} en (0 y 1);

B en (1 y 3);

\bar{B} en (0 y 2).

	A	\bar{A}		
B	6	7	3	2
\bar{B}	4	5	1	0
	\bar{C}	C	\bar{C}	

A en (4, 5, 6 y 7);

\bar{A} en (0, 1, 2 y 3);

B en (2, 3, 6 y 7);

\bar{B} en (0, 1, 4 y 5);

C en (1, 3, 5, y 7);

\bar{C} en (0, 2, 4 y 6).

	A	\bar{A}		
\bar{D}	12	14	6	4
D	13	15	7	5
\bar{D}	9	11	3	1
D	8	10	2	0
	\bar{C}	C	\bar{C}	

A en (8, 9, 10, 11, 12, 13, 14 y 15);

\bar{A} en (0, 1, 2, 3, 4, 5, 6, y 7);

B en (4, 5, 6, 7, 12, 13, 14 y 15);

\bar{B} en (0, 1, 2, 3, 8, 9, 10 y 11);

C en (2, 3, 6, 7, 10, 11, 14 y 15);

\bar{C} en (0, 1, 4, 5, 8, 9, 12 y 13).

D en (1, 3, 5, 7, 9, 11, 13 y 15);

\bar{D} en (0, 2, 4, 6, 8, 10, 12 y 14).

5.4. Proceso de simplificación de funciones lógicas con M.K.

El proceso de simplificación utilizando mapas de Karnaugh se fundamenta en la agrupación de minterminos o maxiterminos en grupos de (1, 2, 4, 8, 16, 32, ...); valores que están basados en las potencias de 2, es decir: $2^0, 2^1, 2^2, 2^3, 2^4, 2^5, \dots$

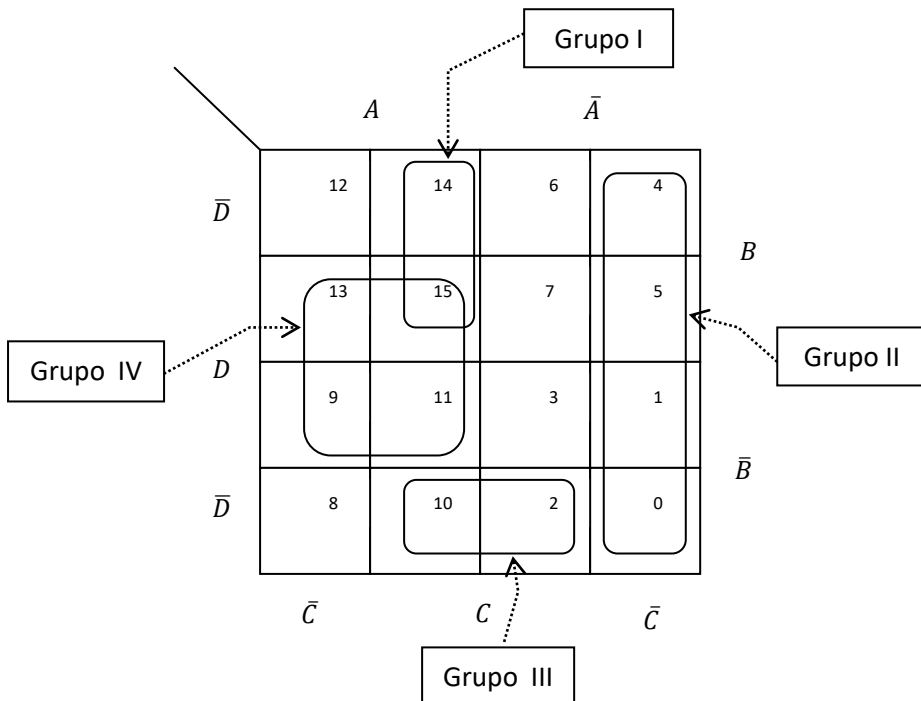
El proceso de simplificación consiste en:

Agrupar la mayor cantidad posible de minterminos o maxiterminos; tal agrupamiento se realiza teniendo en cuenta el fundamento anterior.

Estos grupos se pueden efectuar tanto vertical como horizontalmente. No está permitida la agrupación en diagonal.

Como el proceso es simétrico, se pueden realizar agrupaciones simétricas entre los extremos del mapa de Karnaugh, siempre respetando los fundamentos de agrupamiento.

Los minterminos o maxiterminos se pueden tomar más de una vez si es estrictamente necesario. No deberán formarse grupos innecesariamente.



Como función suma de productos (cuando se agrupan minitérminos):

$$f(A, B, C, D) = \text{Grupo I} + \text{Grupo II} + \text{Grupo III} + \text{Grupo IV}$$

$$f(A, B, C, D) = ABC + \bar{A}\bar{C} + \bar{B}C\bar{D} + AD$$

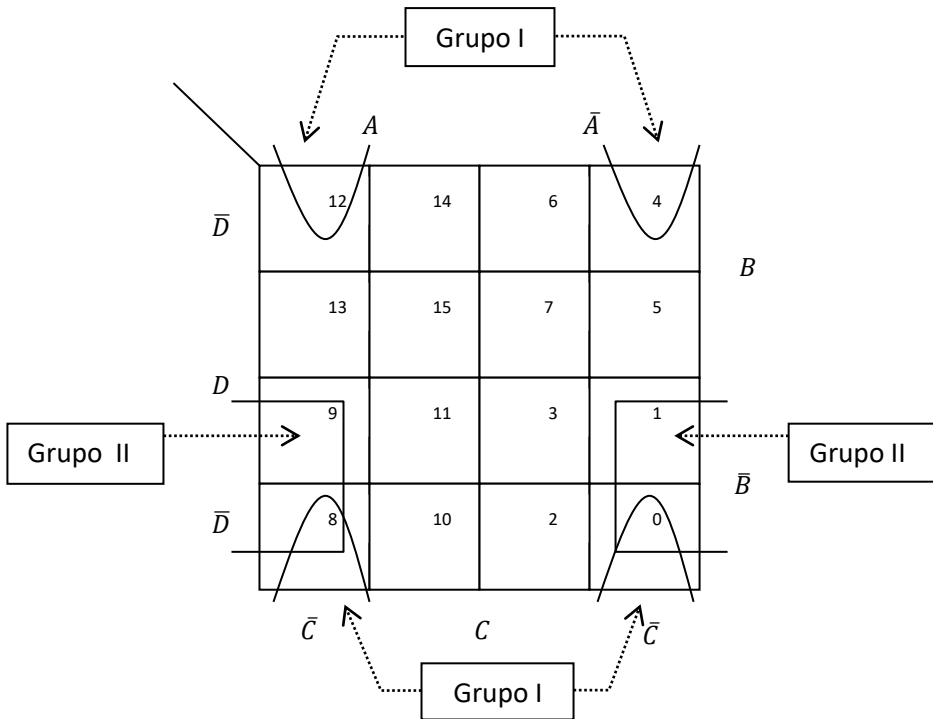
Se observa lo siguiente: las variables que forman cada grupo de la función en formato suma de productos, se toman tal como influyen en las celdas.

Como función producto de sumas (cuando se agrupan maxitérminos):

$$f(A, B, C, D) = (\text{Grupo I})(\text{Grupo II})(\text{Grupo III})(\text{Grupo IV})$$

$$f(A, B, C, D) = (\bar{A} + \bar{B} + \bar{C})(A + C)(B + \bar{C} + D)(\bar{A} + \bar{D})$$

Se observa lo siguiente: las variables que influyen en cada grupo de la función en formato producto de sumas, se toman complementados.



Como función suma de productos (cuando se agrupan minitérminos):

$$f(A, B, C, D) = \text{Grupo I} + \text{Grupo II}$$

$$f(A, B, C, D) = \bar{C}\bar{D} + \bar{B}\bar{C}$$

Como función producto de sumas (cuando se agrupan maxitérminos):

$$f(A, B, C, D) = (\text{Grupo I})(\text{Grupo II})$$

$$f(A, B, C, D) = (C + D)(B + C)$$

5.5. Funciones incompletamente especificadas

Cuando una función no define completamente todas las combinaciones posibles de una tabla de verdad, decimos que la función está incompletamente especificada.

Por ejemplo, el dominio del código BCD es de cuatro variables, esto significa que hay dieciséis posibles combinaciones, de las cuales sólo utiliza diez; las otras seis combinaciones son indiferentes para el código BCD. Estas seis combinaciones no están completamente definidas o especificadas.

Así como en el ejemplo anterior, existen funciones que presentan similares características. Las combinaciones incompletamente especificadas se marcan con una **X**, tanto en la tabla de verdad como en el mapa de Karnaugh.

En el mapa de Karnaugh, la X puede adoptar cualquiera de los dos valores posibles 1 ó 0, a nuestra conveniencia, con la finalidad de optimizar la simplificación.

Ejemplo de una función incompletamente especificada.

$$f(A, B, C, D) = \sum_m(4, 5, 6, 7, 11, 12), i(13, 14, 15)$$

La expresión $i(13, 14, 15)$, indica que las combinaciones 13, 14 y 15 son indiferentes para la función.

Los términos que no aparecen en la suma de minitérminos, significa que son maxitérminos, es decir que adoptan el valor 0.

Problemas Resueltos N° 05

1. Estandarice como suma de productos (minitérminos) y como producto de sumas (maxitérminos), la siguiente expresión booleana.

$$f(X, Y, Z) = X + Y\bar{Z}$$

Solución:

- Estandarizando como suma de productos (minitérminos)

$f(X, Y, Z) = X + Y\bar{Z}$, la expresión lógica está como suma de productos. Agregando las variables faltantes mediante la propiedad del complemento $A + \bar{A} = 1$; tenemos:

$$f(X, Y, Z) = X(Y + \bar{Y})(Z + \bar{Z}) + Y\bar{Z}(X + \bar{X})$$

Se observa que al primer término X le faltan las variables Y y Z y al segundo término, la variable X .

En el desarrollo de la ecuación se debe tener en cuenta el orden de las variables del dominio de la función.

$$f(X, Y, Z) = XYZ + \textcircled{XY\bar{Z}} + X\bar{Y}Z + X\bar{Y}\bar{Z} + \textcircled{X\bar{Y}Z} + \bar{X}YZ$$

Aplicando la propiedad de idempotencia (de la suma) a los términos de la circunferencia.

Dando valores 1 a las variables sin complementar y 0 a las complementadas, tenemos:

$$f(X, Y, Z) = 111 + 110 + 101 + 100 + 010$$

$$f(X, Y, Z) = m7 + m6 + m5 + m4 + m2$$

$$f(X, Y, Z) = \sum_m (2, 4, 5, 6, 7)$$

- Estandarizando como producto de sumas (maxitérminos)

$f(X, Y, Z) = X + Y\bar{Z}$, se expresará como producto de sumas aplicando la propiedad distributiva: $f(X, Y, Z) = (X + Y)(X + \bar{Z})$

A este producto de sumas se agrega las variables que le faltan a cada término. Aplicando la propiedad del complemento $A\bar{A} = 0$; tenemos:

$$f(X, Y, Z) = (X + Y + Z\bar{Z})(X + \bar{Z} + Y\bar{Y})$$

Desarrollando la expresión cuidando que el orden de las variables sea el del dominio de la función.

$$f(X, Y, Z) = (X + Y + Z)(X + Y + \bar{Z})(X + \bar{Y} + \bar{Z})$$

Aplicando la propiedad de idempotencia (del producto) y dando el valor 1 a las variables complementadas y 0 a las variables sin complementar, tenemos:

$$f(X, Y, Z) = (0 + 0 + 0)(0 + 0 + 1)(0 + 1 + 1)$$

$$f(X, Y, Z) = M0.M1.M3$$

$$f(X, Y, Z) = \prod_M (0,1,3)$$

En la estandarización de la función mediante minitérminos y maxitérminos, se observa la complementariedad de ambos métodos.

$$f(X, Y, Z) = \sum_m(2, 4, 5, 6, 7); \quad f(X, Y, Z) = \prod_M(0, 1, 3)$$

Cuando ambas expresiones son colocados en una tabla de verdad, los minitérminos adoptan el valor 1 y los maxitérminos el valor 0.

Nº	X Y Z	Función	$f(X, Y, Z)$
0	0 0 0	$\prod(0)$	0
1	0 0 1	$\prod(1)$	0
2	0 1 0	$\sum(2)$	1
3	0 1 1	$\prod(3)$	0
4	1 0 0	$\sum(4)$	1
5	1 0 1	$\sum(5)$	1
6	1 1 0	$\sum(6)$	1
7	1 1 1	$\sum(7)$	1

2. Estandarice como minitérmino la siguiente expresión booleana.

$$f(A, B, C, D) = \bar{B} + A(\bar{C} + D)$$

Solución:

Para estandarizar una expresión booleana como **suma de productos**, es necesario aplicar la propiedad distributiva: $X(Y + Z) = XY + XZ$

$$f(A, B, C, D) = \bar{B} + A\bar{C} + AD.$$

A cada **término producto** se introducen las variables que faltan para completar el dominio de la función, mediante la propiedad del complemento: $X + \bar{X} = 1$

$$f(A, B, C, D) = \bar{B}(A + \bar{A})(C + \bar{C})(D + \bar{D}) + A\bar{C}(B + \bar{B})(D + \bar{D}) + AD(B + \bar{B})(C + \bar{C})$$

Desarrollando con la propiedad distributiva en cada término, obtenemos: $X(Y + \bar{Y})(Z + \bar{Z}) = XYZ + XY\bar{Z} + X\bar{Y}Z + X\bar{Y}\bar{Z}$; y ordenando alfabéticamente las variables, tenemos:

$$f(A, B, C, D) = ABC\bar{D} + AB\bar{C}\bar{D} + A\bar{B}\bar{C}\bar{D} + A\bar{B}\bar{C}D + ABCD + A\bar{B}CD + A\bar{B}C\bar{D} + A\bar{B}CD + A\bar{B}C\bar{D} + A\bar{B}\bar{C}D + A\bar{B}\bar{C}\bar{D}$$

Dando valor **1** a la **variable sin complementar** y **0** a la **variable complementada**.

$$f(A, B, C, D) = 1101 + 1100 + 1001 + 1000 + 1111 + 1011 + 1010 + 0011 + 0010 + 0001 + 0000$$

$$f(A, B, C, D) = m_{13} + m_{12} + m_9 + m_8 + m_{15} + m_{11} + m_{10} + m_3 + m_2 + m_1 + m_0$$

$$f(A, B, C, D) = \sum_m (0, 1, 2, 3, 8, 9, 10, 11, 12, 13, 15)$$

3. Estandarice como maxitérmino la siguiente expresión booleana.

$$f(A, B, C, D) = (A + BC)(\bar{B} + \bar{C} + D)$$

Solución:

Para estandarizar una expresión booleana como **producto de sumas**, es necesario aplicar la propiedad distributiva: $(X + YZ) = (X + Y)(X + Z)$

$$f(A, B, C, D) = (A + B)(A + C)(\bar{B} + \bar{C} + D).$$

A cada **término suma** se introducen las variables que faltan para completar el dominio de la función, mediante la propiedad del complemento: $X\bar{X} = 0$

$$f(A, B, C, D) = (A + B + C\bar{C} + D\bar{D})(A + C + B\bar{B} + D\bar{D})(\bar{B} + \bar{C} + D + A\bar{A})$$

Desarrollando con la propiedad distributiva en cada término, obtenemos: $(X + Y + Z\bar{Z}) = (X + Y + Z)(X + Y + \bar{Z})$; y ordenando alfabéticamente las variables, tenemos:

$$f(A, B, C, D) = (A + B + C + D)(A + B + C + \bar{D})(A + B + \bar{C} + D)(A + B + \bar{C} + \bar{D})(A + \bar{B} + C + D)(A + \bar{B} + C + \bar{D})(A + \bar{B} + \bar{C} + D)(\bar{A} + \bar{B} + \bar{C} + D)$$

Dando valor **0** a la **variable sin complementar** y **1** a la **variable complementada**.

$$f(A, B, C, D) = (0 + 0 + 0 + 0)(0 + 0 + 0 + 1)(0 + 0 + 1 + 0)(0 + 0 + 1 + 1)(0 + 1 + 0 + 0)(0 + 1 + 0 + 1)(0 + 1 + 1 + 0)(1 + 1 + 1 + 0)$$

$$f(A, B, C, D) = M_0 \cdot M_1 \cdot M_2 \cdot M_3 \cdot M_4 \cdot M_5 \cdot M_6 \cdot M_{14}$$

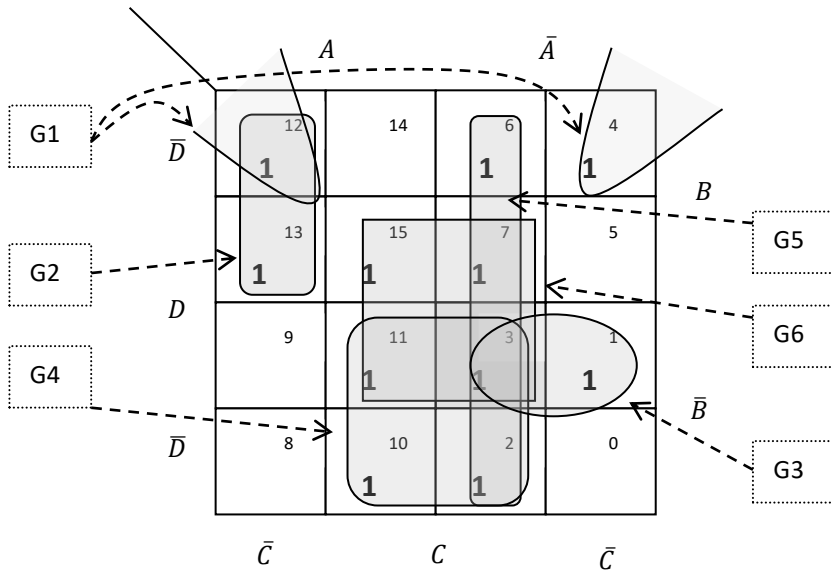
$$f(A, B, C, D) = \prod_M (0, 1, 2, 3, 4, 5, 6, 14)$$

4. Halle la función lógica de los siguientes minitérminos.

$$f(A, B, C, D) = \sum_m (1, 2, 3, 4, 6, 7, 10, 11, 12, 13, 15)$$

Solución:

Se utiliza un Mapa de Karnaugh de cuatro variables. En la celda correspondiente, cada **minitérmino** adopta el valor **1**.



Se forman seis grupos:

- G1: conformado por las celdas (4 y 12)
- G2: conformado por las celdas (12 y 13)
- G3: conformado por las celdas (1 y 3)
- G4: conformado por las celdas (2, 3, 10 y 11)
- G5: conformado por las celdas (2, 3, 6 y 7)
- G6: conformado por las celdas (3, 7, 11 y 15)

Los seis grupos se suman:

$$f(A, B, C, D) = G1 + G2 + G3 + G4 + G5 + G6$$

Asociando convenientemente las variables en cada **grupo (producto de variables)** y tomando cada variable tal como está presente en el grupo, es decir, si las variables que influyen en el grupo G1 son (B, \bar{C}, \bar{D}), entonces, $G1 = B\bar{C}\bar{D}$

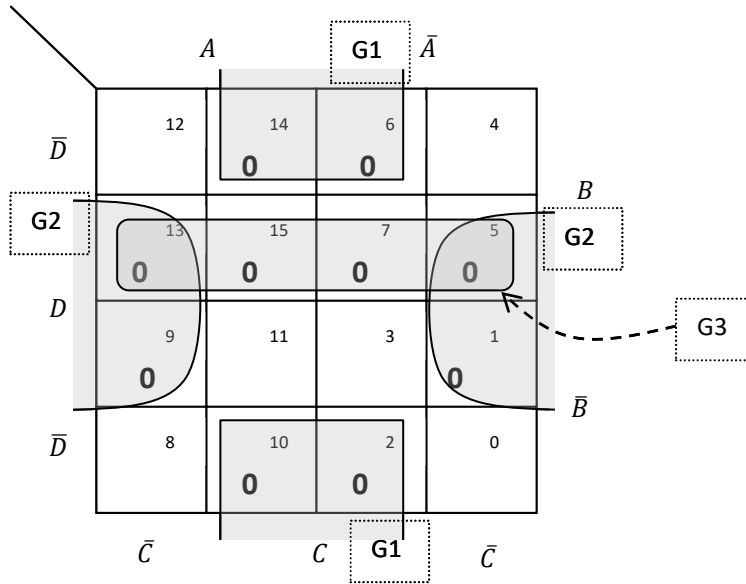
$$f(A, B, C, D) = B\bar{C}\bar{D} + ABC\bar{C} + \bar{A}BD + \bar{B}C + \bar{A}C + CD$$

5. Halle la función lógica de los siguientes maxitérminos.

$$f(A, B, C, D) = \prod_M(1, 2, 5, 6, 7, 9, 10, 13, 14, 15)$$

Solución:

Se utiliza un Mapa de Karnaugh de cuatro variables. En la celda correspondiente, cada **maxitérmino** adopta el valor **0**.



Se forman tres grupos:

G1: conformado por las celdas (2, 6, 10 y 14)

G2: conformado por las celdas (1, 5, 9 y 13)

G3: conformado por las celdas (5, 7, 13 y 15)

Los tres grupos se multiplican:

$$f(A, B, C, D) = G1 \cdot G2 \cdot G3$$

Asociando convenientemente las variables en cada **grupo (suma de variables)** y tomando cada variable del grupo en forma invertida ó negada, tenemos:

Si las variables que influyen en el grupo G1 son (C, \bar{D}) , entonces, $G1 = \bar{C} + D$, porque las variables se toman invertidas.

$$f(A, B, C, D) = (\bar{C} + D)(C + \bar{D})(\bar{B} + \bar{D})$$

6. Halle mediante minitérminos y maxitérminos, la función lógica de la siguiente tabla de verdad.

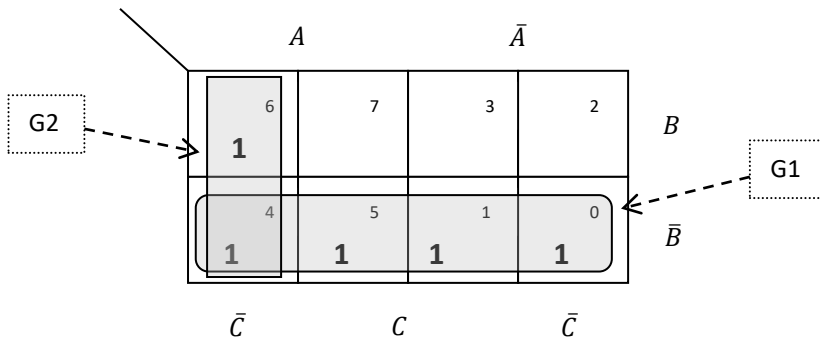
Nº	A B C	f(A,B,C)
0	0 0 0	1
1	0 0 1	1
2	0 1 0	0
3	0 1 1	0
4	1 0 0	1
5	1 0 1	1
6	1 1 0	1
7	1 1 1	0

De la tabla de verdad se obtienen los correspondientes minitérminos y maxitérminos.

$$f(A, B, C) = \sum_m (0, 1, 4, 5, 6)$$

$$f(A, B, C) = \prod_M (2, 3, 7)$$

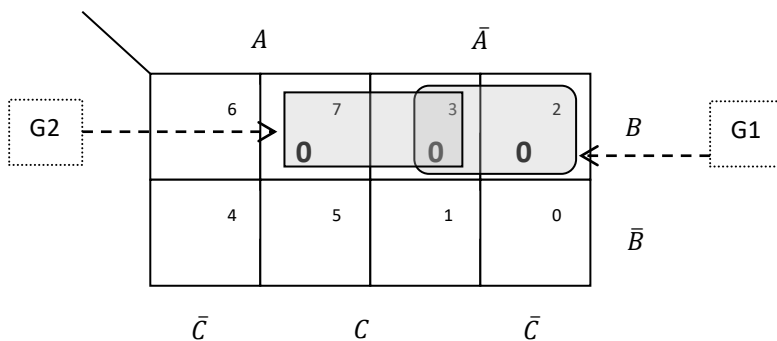
Como suma de productos (minitérminos).



$$f(A, B, C) = G1 + G2 = B + A\bar{C}$$

$$f(A, B, C) = \bar{B} + A\bar{C} = (A + \bar{B})(\bar{B} + \bar{C})$$

Como producto de sumas (maxitérminos).



$$f(A, B, C) = G1.G2 = (A + \bar{B})(\bar{B} + \bar{C})$$

$$f(A, B, C) = (A + \bar{B})(\bar{B} + \bar{C}) = \bar{B} + A\bar{C}$$

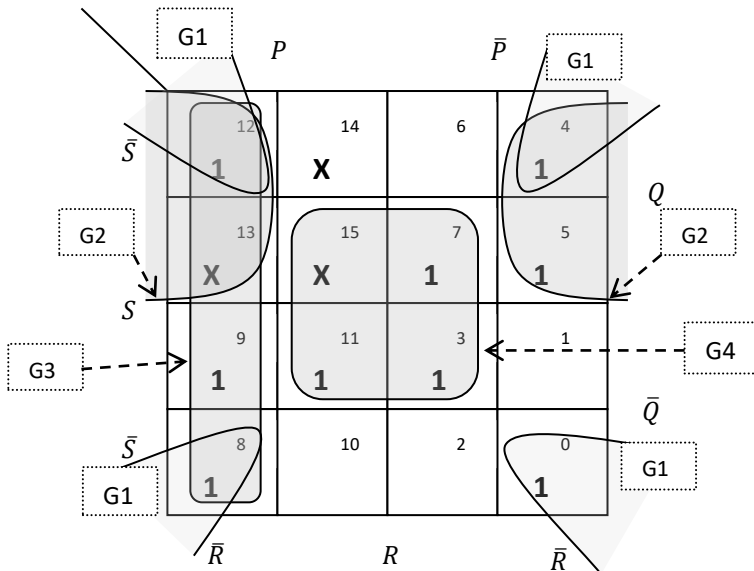
OBSERVACIÓN: si hallamos la función lógica mediante minitérminos o maxitérminos, el resultado es el mismo.

7. Halle la expresión lógica de la función incompletamente especificada.

$$f(P, Q, R, S) = \sum_m (0, 3, 4, 5, 7, 8, 9, 11, 12), i(13, 14, 15)$$

Solución:

Se observa que las celdas 13,14 y 15, contienen términos no especificados, los que ingresan al Mapa de Karnaugh con una X. Estos términos pueden adoptar el valor 0 ó 1, según convenga en la agrupación de los demás términos que si están especificados (minitérminos). No es necesario utilizar todo los términos no especificados.



Se forman 4 grupos:

G1: conformado por las celdas (0, 4, 8 y 12)

G2: conformado por las celdas (4, 5, 12 y 13)

G3: conformado por las celdas (8, 9, 12 y 13)

G4: conformado por las celdas (3, 7, 11 y 15)

Los cuatro grupos se suman:

$$f(P, Q, R, S) = G1 + G2 + G3 + G4$$

$$f(P, Q, R, S) = \bar{R}\bar{S} + Q\bar{R} + P\bar{R} + RS$$

OBSERVACIÓN: el grupo tres (G3) pudo haberse formado con las celdas (9, 11, 13 y 15)

$$f(P, Q, R, S) = \bar{R}\bar{S} + Q\bar{R} + PS + RS$$

Con ambas expresiones se obtienen los mismos resultados si se comprueba con una tabla de verdad.

Problemas Propuestos N° 05

1. Estandariza en términos mínimos (minitérminos) la siguiente expresión booleana.

$$f(A, B, C, D) = A(B + \bar{C})(\bar{A} + \bar{C}\bar{D}) + \bar{B}C$$

Rpta: $m_2 + m_3 + m_9 + m_{10} + m_{11} + m_{12} + m_{13}$

2. Estandariza en términos máximos (maxitérminos) la siguiente expresión booleana.

$$f(A, B, C, D) = (\bar{A}B + CD)(\bar{B} + C + D)(\bar{A} + C\bar{D})$$

Rpta: $M_0. M_1. M_2. M_4. M_8. M_9. M_{10}. M_{11}. M_{12}. M_{13}. M_{14}. M_{15}$

3. Encuentre la función lógica de los siguientes minitérminos.

$$f(P, Q, R, S) = \sum_m (0, 1, 2, 3, 5, 8, 10, 12, 13, 14, 15)$$

Rpta: $f(P, Q, R, S) = \mathbf{PQ} + \mathbf{\bar{P}\bar{Q}} + \mathbf{\bar{Q}\bar{S}} + \mathbf{\bar{P}\bar{R}S} = \mathbf{\bar{P}\oplus\bar{Q}} + \mathbf{\bar{Q}\bar{S}} + \mathbf{\bar{P}\bar{R}S}$

4. Encuentre la función lógica de los siguientes maxitérminos.

$$f(A, B, C, D) = \prod_M (3, 4, 5, 7, 8, 10, 11, 12, 13, 14, 15)$$

Rpta: $f(A, B, C, D) = (\bar{A} + \bar{B})(\bar{C} + \bar{D})(\bar{A} + D)(\bar{B} + C)$

5. Halle la función lógica mediante minitérminos y maxitérminos de la siguiente tabla de verdad.

N°	D	C	B	A	$f(D, C, B, A)$	N°	D	C	B	A	$f(D, C, B, A)$
0	0	0	0	0	1	8	1	0	0	0	0
1	0	0	0	1	1	9	1	0	0	1	0
2	0	0	1	0	1	10	1	0	1	0	0
3	0	0	1	1	1	11	1	0	1	1	0
4	0	1	0	0	0	12	1	1	0	0	1
5	0	1	0	1	1	13	1	1	0	1	1
6	0	1	1	0	1	14	1	1	1	0	1
7	0	1	1	1	1	15	1	1	1	1	1

Rpta: como suma de productos: $f(D, C, B, A) = DC + \bar{D}\bar{C} + \bar{D}B + CA$

Rpta: como producto de sumas: $f(D, C, B, A) = (\bar{D} + C)(D + \bar{C} + B + A)$

Capítulo 6

Circuitos combinacionales y circuitos integrados

6.1. Introducción a los circuitos combinacionales

Se denomina circuitos combinacionales a aquellos circuitos digitales que no cuentan con un enlace de retroalimentación ni de un sistema de memoria. Esta característica conlleva a que la salida del circuito responda únicamente a las combinaciones de los valores de las variables de entrada, esto es, a cualquier variación que exista en la entrada del circuito, la salida variará inmediatamente respondiendo a ésta combinación.

Los codificadores, decodificadores, multiplexores, demultiplexores, comparadores, entre otros, son ejemplos de circuitos combinacionales

6.2. Implementación de circuitos combinacionales

Los circuitos combinacionales como los mencionados anteriormente, están diseñados e implementados en circuitos integrados para ejecutar tareas específicas, sin embargo, hay muchos otros casos en el que se requiere un diseño especial para cumplir también una tarea especial. La implementación de estos circuitos combinacionales se realizan con las conexiones apropiadas en circuitos integrados con compuertas lógicas básicas, teniendo en cuenta a un diagrama lógico previamente elaborado.

La función lógica que se desea implementar, puede ser conocida de antemano y a partir de allí comenzar la implementación, o, puede darse el caso que sólo tengamos los requerimientos de la salida de un circuito, es decir, los minitérminos ó maxitérminos expresados en una tabla de verdad. Para dar solución a este último caso, es necesaria la utilización de una herramienta muy potente de diseño como es el mapa de Karnaugh.

6.2.1. Partiendo de una función booleana

Implemente la siguiente función lógica:

$$f(A, B, C, D) = ABD + BC + \overline{A + D} + \overline{BC}D$$

- **Primer paso:** Se simplifica la función.

$$f(A, B, C, D) = ABD + BC + \overline{A + D} + \overline{BC}D$$

$$f(A, B, C, D) = ABD + BC + AD + \overline{BC}D$$

$$f(A, B, C, D) = \underbrace{ABD + AD} + \underbrace{BC + \overline{BC}D}$$

$$f(A, B, C, D) = AD + BC + D$$

- **Segundo paso:** Elaboración de la tabla de verdad

N°	Variables				Función primaria $ABD + BC + \overline{A + D} + \overline{BC}D$					Función simplificada $AD + BC + D$			
	A	B	C	D	ABD	BC	$\overline{A + D}$	$\overline{BC}D$	f	AD	BC	D	f
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1	1	0	0	1	1
2	0	0	1	0	0	0	0	0	0	0	0	0	0
3	0	0	1	1	0	0	0	1	1	0	0	1	1
4	0	1	0	0	0	0	0	0	0	0	0	0	0
5	0	1	0	1	0	0	0	1	1	0	0	1	1
6	0	1	1	0	0	1	0	0	1	0	1	0	1
7	0	1	1	1	0	1	0	0	1	0	1	1	1
8	1	0	0	0	0	0	0	0	0	0	0	0	0
9	1	0	0	1	0	0	1	1	1	1	0	1	1
10	1	0	1	0	0	0	0	0	0	0	0	0	0
11	1	0	1	1	0	0	1	1	1	1	0	1	1
12	1	1	0	0	0	0	0	0	0	0	0	0	0
13	1	1	0	1	1	0	1	1	1	1	0	1	1
14	1	1	1	0	0	1	0	0	1	0	1	0	1
15	1	1	1	1	1	1	1	0	1	1	1	1	1

Tabla 6.1

Ambas expresiones son iguales

Equivalencia entre función primaria y simplificada.

- **Tercer paso:** Se construye el diagrama lógico de la función simplificada.

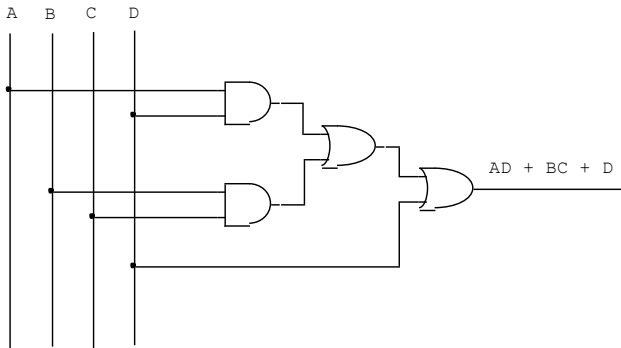


Fig. 6.1

Diagrama lógico de una función simplificada.

- **Cuarto paso:** Se elabora el diagrama esquemático.

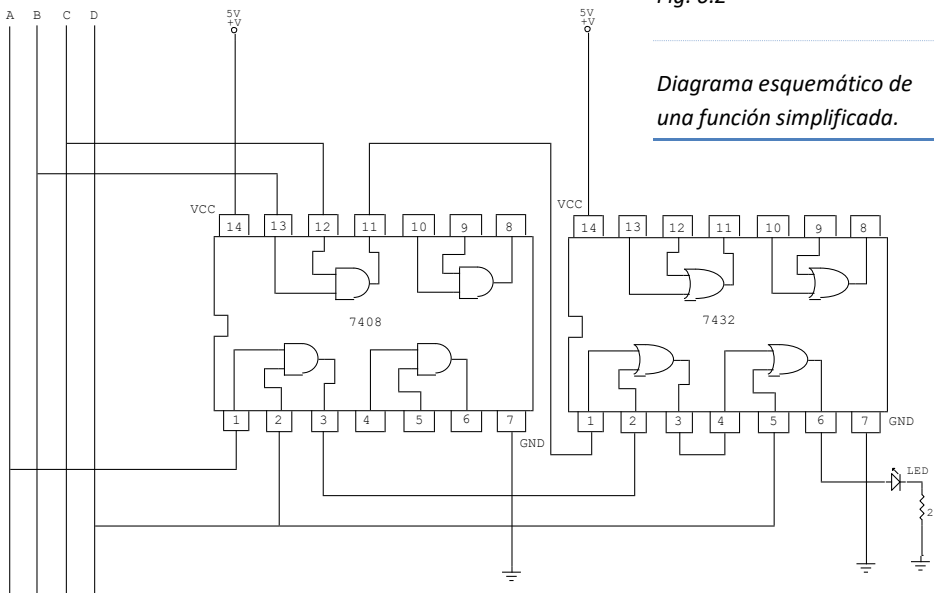


Fig. 6.2

Diagrama esquemático de una función simplificada.

- **Quinto paso:** Se implementa físicamente el circuito digital en una placa de prueba llamada protoboard en el que se insertan los respectivos circuitos integrados y demás dispositivos que permitan visualizar los niveles lógicos de la salida como son los diodos LED. Se debe tener en cuenta la polarización adecuada de los C.I.

6.2.2. Partiendo de una tabla de verdad

El diseño de un circuito digital parte del resultado esperado de una función lógica. Este resultado se arregla en una tabla de verdad evaluando todas las posibilidades que las combinaciones de las variables permiten.

Cuando los resultados esperados de la función lógica se encuentren organizados en una tabla de verdad, se pueden emplear dos métodos para diseñar el circuito digital: el método de la tabla y el método de los mapas de Karnaugh.

Ejemplo: se requiere implementar el resultado de la función de la siguiente tabla de verdad.

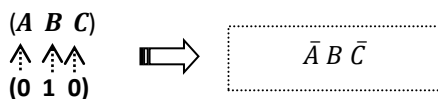
N°	Variables			Función
	A	B	C	$f(A, B, C)$
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	0
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1

Tabla 6.2

Tabla de verdad de una función arbitraria.

POR EL MÉTODO DE LA TABLA

Cada minitérmino se expresa en función de sus variables. Por ejemplo, se observa que el minitérmino de la combinación UNO (0 0 1) corresponden a las variables (A B C); entonces a ésta combinación le corresponderá una variable complementada cuando es cero y sin complementar cuando es 1. Así: $\bar{A} \bar{B} C$.



N°	Variables			Función	Minitérminos expresados con variables
	A	B	C	$f(A, B, C)$	
0	0	0	0	0	
1	0	0	1	1	$\bar{A} \bar{B} C$
2	0	1	0	1	$\bar{A} B \bar{C}$
3	0	1	1	0	
4	1	0	0	0	
5	1	0	1	1	$A \bar{B} C$
6	1	1	0	0	
7	1	1	1	1	ABC

Tabla 6.3

Minitérminos expresados en sus respectivas variables.

La función lógica se expresa así:

$$f(A, B, C) = \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}C + ABC$$

Se simplifica empleando las propiedades del Álgebra de Boole.

$$f(A, B, C) = \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}C + ABC; \text{trabajando con el tercer y cuarto término.}$$

$$f(A, B, C) = \bar{A}\bar{B}C + \bar{A}B\bar{C} + AC(\bar{B} + B)$$

$$f(A, B, C) = \bar{A}\bar{B}C + \bar{A}B\bar{C} + AC; \text{factorizando } C \text{ en el primer y tercer término.}$$

$$f(A, B, C) = C(\bar{A}\bar{B} + A) + \bar{A}B\bar{C}$$

$$f(A, B, C) = C(A + \bar{B}) + \bar{A}B\bar{C}$$

$$f(A, B, C) = AC + \bar{B}C + \bar{A}B\bar{C}$$

Ésta función se puede implementar como se hizo en la sección 6.3.1.

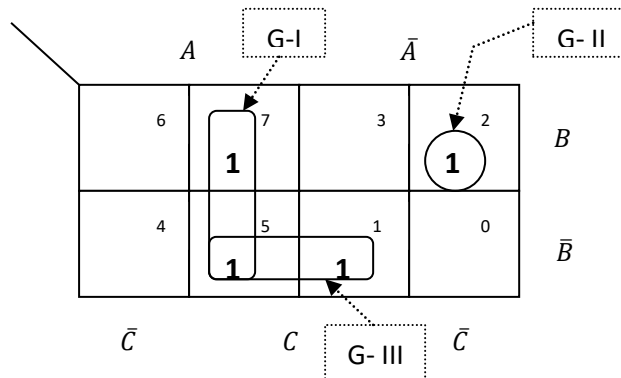
POR EL MÉTODO DEL MAPA DE KARNAUGH

El resultado de la función lo podemos expresar en base a los minterminos o maxiterminos.

Como minterminos. $f(A, B, C) = \sum_m(1,2,5,7)$

Como maxiterminos. $f(A, B, C) = \prod_M(0,3,4,6)$

Trabajando con los minterminos tenemos:



$$f(A, B, C) = GI + GII + GIII$$

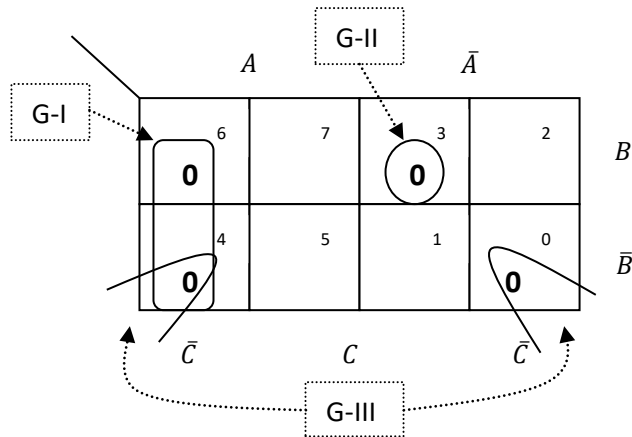
$$f(A, B, C) = AC + \bar{A}B\bar{C} + \bar{B}C$$

$$f(A, B, C) = AC + \bar{B}C + \bar{A}B\bar{C}$$

Se verifica que con ambos métodos se obtiene el mismo resultado.

También, si se desea se puede trabajar con los maxitérminos y se obtendrá el mismo resultado.

Trabajando con maxitérminos.



$$f(A, B, C) = G_I \cdot G_{II} \cdot G_{III}$$

$$f(A, B, C) = (\bar{A} + C)(A + \bar{B} + \bar{C})(B + C)$$

Podríamos implementar el circuito digital con el producto de sumas o si preferimos aplicamos la propiedad distributiva para obtener una suma de productos.

$$f(A, B, C) = (\bar{A} + C)(A + \bar{B} + \bar{C})B + (\bar{A} + C)(A + \bar{B} + \bar{C})C$$

$$f(A, B, C) = (\bar{A} + C)AB + (\bar{A} + C)B\bar{B} + (\bar{A} + C)B\bar{C} + (\bar{A} + C)AC + (\bar{A} + C)\bar{B}C + (\bar{A} + C)\bar{C}C; \text{ aplicando la propiedad del complemento, tenemos.}$$

$$f(A, B, C) = (\bar{A} + C)AB + (\bar{A} + C)B\bar{C} + (\bar{A} + C)AC + (\bar{A} + C)\bar{B}C$$

$$f(A, B, C) = \bar{A}AB + ABC + \bar{A}B\bar{C} + BC\bar{C} + \bar{A}AC + ACC + \bar{A}\bar{B}C + \bar{B}CC$$

$$f(A, B, C) = ABC + \bar{A}B\bar{C} + AC + \bar{A}\bar{B}C + \bar{B}C; \text{ aplicando absorción y consenso.}$$

$$f(A, B, C) = AC + \bar{B}C + \bar{A}B\bar{C}$$

Con este resultado, hemos demostrado que tanto con minterminos como con maxitérminos obtenemos el mismo resultado.

IMPORTANTE

Se tendrá en cuenta que la función lógica, la tabla de verdad y los mapas de Karnaugh están estrechamente vinculados.

El método de la tabla se hace mucho más potente cuanto menos minterminos o maxiterminos queramos operar, ó, cuando se realiza una evaluación previa en el mapa de Karnaugh y se observa que no se pueden formar grupos con más de un mintermino o maxitermino.

6.3. Proceso de desarrollo de circuitos de control lógico

En la práctica digital constantemente nos encontramos con innumerables circuitos sencillos que facilitan muchísimo el trabajo que realizamos. El desarrollo de circuitos con habilitación de compuertas lógicas y circuitos de control, permitirán una mayor dinámica en la manipulación de circuitos digitales.

6.3.1. Circuitos de habilitación y deshabilitación

Las compuertas OR, NOR, AND, NAND y OR EXCLUSIVO, pueden utilizarse para controlar las señales que pasan por ellas y luego proporcionadas a circuitos más elaborados.

DESHABILITACIÓN

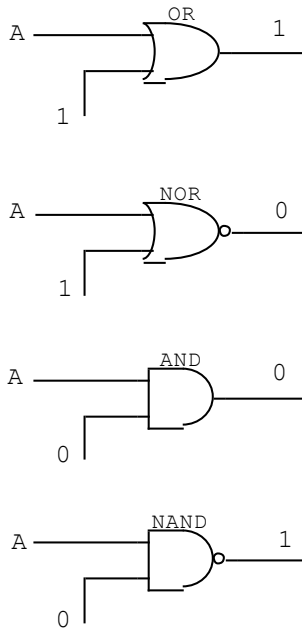


Fig. 6.3

Compuertas lógicas que cumplen la función de deshabilitación. Se observa que las salidas están en 0 o en 1, lo que significa que no están pasando los valores de la variable A.

HABILITACIÓN

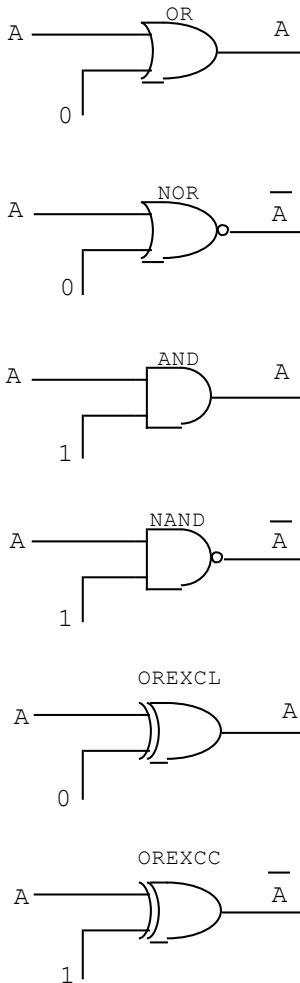


Fig. 6.4

Compuertas lógicas que cumplen la función de habilitación. Se observa que la salida de dichas compuertas tienen a la variable A complementada y sin complementar. La compuerta OR EXCLUSIVA, tiene la propiedad de dejar pasar la variable en un caso sin complementar y en el otro complementada según si el terminal control está en 0 o en 1 respectivamente.

6.3.2. Circuitos de control

Todo circuito de control requiere dispositivos especiales llamados "sensores" para suministrar información a su entrada, y luego de ser procesada es enviada como señal de salida que deberá controlar unidades externas.

Los sensores son dispositivos que responden a una excitación de tipo mecánico, eléctrico, térmico, químico, magnético, etc, generando una señal eléctrica en su salida. Dicha señal puede ser amplificada, pues, el nivel de voltaje y corriente que proporciona un sensor es muy débil.

Los sistemas de cómputo utilizan sistemas de control muy sofisticado para controlar unidades de disco, entrada y salida de datos, entre otros. Los microcontroladores son dispositivos con microprocesador y memoria que pueden ser configurados para trabajar como circuito de control digital en un sistema de cómputo o en aplicaciones especiales como en la “domótica”.

La domótica es el área que trata sobre las casas automatizadas o inteligentes, pues cuenta con un circuito digital que procesa información que le llegan de los múltiples sensores instalados en el edificio y controlan motores para abrir las puertas o cerrarlas, correr cortinas, encender o apagar artefactos, activar el aire acondicionado, regar jardines, etc.

No obstante lo complejo que pueden resultar los circuitos de control tanto en los sistemas de cómputo como los aplicados a trabajos industriales, existen necesidades no tan complicadas que requieren el diseño e implementación de pequeños circuitos de control para activar una alarma o mover pequeños motores.

Un circuito de control contará en su entrada con sensores que proveerán los niveles lógicos apropiados y con ellos se generará la señal de control para los demás circuitos.

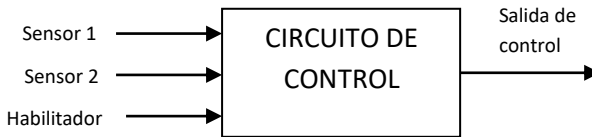


Fig. 6.5

Diagrama de bloques de un circuito de control.

La parte procedimental de los circuitos de control lo realizaremos diseñando un circuito de control con tres entradas (dos sensores y un habilitador del circuito) y una salida.

En primer lugar debemos definir la lógica de activación así como el trabajo que realizan los sensores, enseguida, se elabora su respectiva tabla de verdad en cuya salida se expresan los valores lógicos del circuito de control, posteriormente, en base a los minitérminos o maxitérminos de la salida se simplifica y se confecciona el diagrama lógico y esquemático para finalmente implementar físicamente el circuito.

Los sensores 1 y 2, (Se1, Se2) pueden ser sensores ópticos, los cuales se activan al ser interrumpido el haz de luz entre que inciden sobre ellos.

El emisor de luz puede ser un diodo emisor de luz (LED), un laser, etc. y el receptor podría ser un fotoresistor, fotodiodo o un fototransistor.

Si normalmente el haz de luz incide sobre el receptor de luz que para nuestro caso se trata de un fototransistor, éste conduce corriente de colector y por lo tanto el voltaje colector emisor (V_{CE}) es aproximadamente cero (0 lógico). Al interrumpirse el haz de luz el transistor inmediatamente deja de conducir corriente de colector, pues la corriente de base es cero y el voltaje colector emisor (V_{CE}) es aproximadamente el valor de la fuente de voltaje positiva (1 lógico).

Por lo tanto, los sensores se activan con 1 y se desactivan con 0. La alarma se activará cuando uno o los dos sensores se activen.

El habilitador (Hab), es un terminal que pondrá en funcionamiento el circuito (nivel bajo) o lo aislará del mundo exterior (nivel alto).

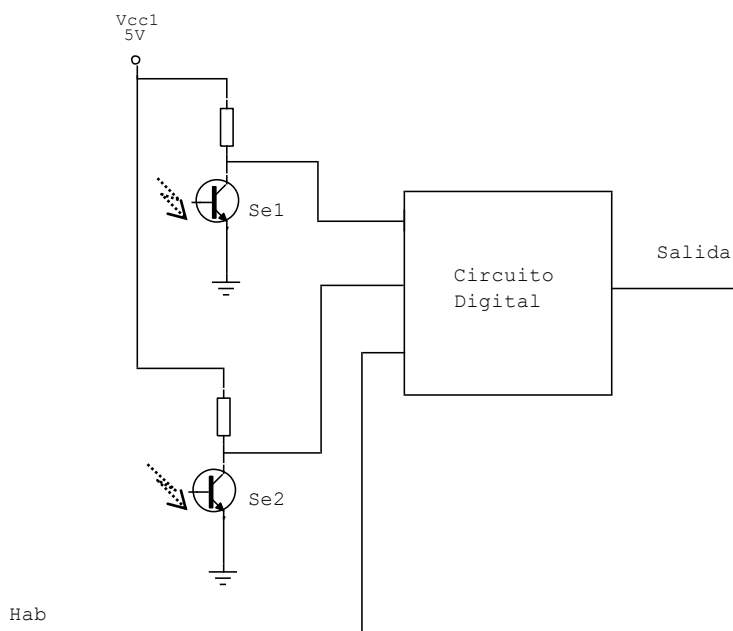


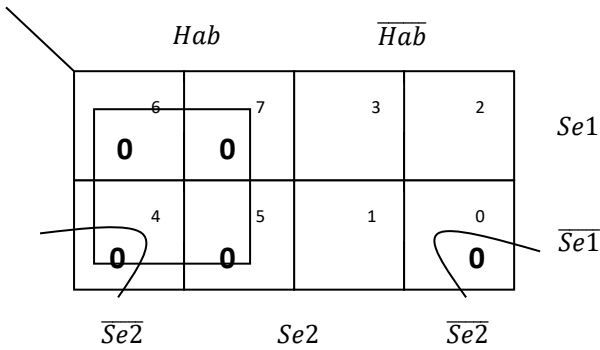
Fig. 6.6

Circuito de control con dos fototransistores como sensores ópticos. Además de las dos entradas de los sensores, existe otra para habilitar o deshabilitar el circuito.

N°	Entradas			Salida Scontrol
	Hab	Se1	Se2	
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

Tabla. 6.4

Tabla de verdad del circuito de control, sólo se activa en las combinaciones 1, 2 y 3.



$Scontrol = \overline{Hab}(Se1 + Se2)$; usa una NOT, una OR y una AND. (se implementa con tres circuitos integrados)

$Scontrol = \overline{Hab}(\overline{Se1 + Se2})$, solo con compuertas NAND

$Scontrol = \overline{Hab}(\overline{Se1.Se2})$, utilizan seis NAND. (se implementa con dos circuitos integrados)

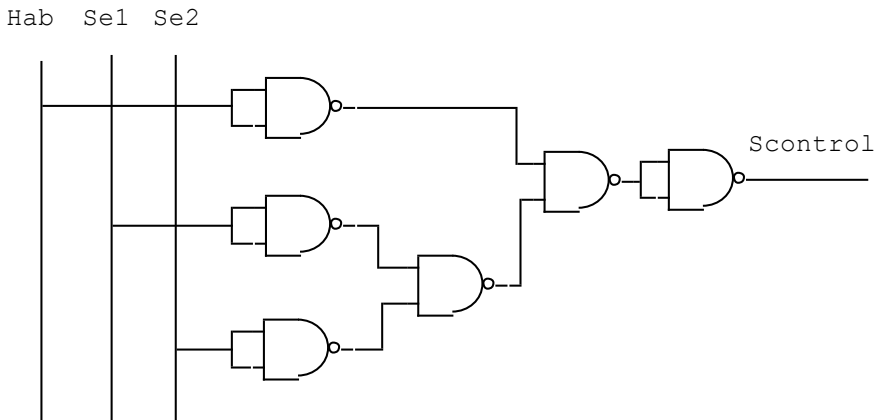


Fig. 6.7

Diagrama lógico del circuito de control construido en base a compuertas NAND, con dos C.I. 7400.

6.4. Características de los Circuitos Integrados Digitales

La implementación física de circuitos digitales se realiza con circuitos Integrados digitales. Estos chips normalmente del tipo DIP (Pines en doble hilera) son fabricados sobre un estrato de silicio con múltiples configuraciones que hacen el equivalente de cientos y hasta miles de transistores.

En la construcción de circuitos integrados básicamente hay dos familias o tecnologías con los cuales se construyen diferentes tipos de compuertas, TTL y CMOS.

La familia **TTL** utiliza como estructura básica el transistor bipolar tipo NPN o PNP para ensamblar sus compuertas lógicas, en cambio, la familia **CMOS** utiliza el transistor unipolar MOSFET de canal tipo P o tipo N.

Ambas familias de circuitos integrados tienen varias características en común, sobre todo porque construyen circuitos digitales equivalentes y con el mismo número de pines.

Un circuito integrado de la serie o familia **TTL** se caracteriza porque utiliza las letras **L** y **S** para designar un tipo de tecnología en la fabricación de compuertas o circuitos lógicos, como por ejemplo, al circuito integrado con cuatro compuertas OR de dos entradas, se denota **74LS32**.

En la familia **CMOS**, se denota mediante las letras **H** y **C**. por ejemplo, el circuito integrado con cuatro compuertas AND de dos entradas es el **74HC08**.

La diferencia sustancial entre ambas familias se manifiesta en un circuito digital cuando los pines de un circuito integrado se encuentran “al aire” o sin conectar. La serie **CMOS** presenta varios problemas cuando las entradas de sus compuertas están “al aire”, lo que proporciona una salida incorrecta ó errática; mientras que, bajo las mismas condiciones de entrada, la serie o tecnología **TTL** asume un **valor alto** ó **1**.

6.4.1. Compuertas lógicas con tres estados.

La salida de una compuerta lógica convencional está basada en dos estados lógicos: ALTO y BAJO, que corresponden a los niveles lógicos 1 y 0, sin embargo, existen circuitos integrados cuyas compuertas pueden tener adicionalmente un tercer estado (no lógico) llamado de “ALTA IMPEDANCIA”. En éste estado la compuerta se comporta como si no estuviera conectado al circuito digital.

Estas compuertas triestado tienen una aplicación especial en la construcción de redes o buses de señal, controlados por la entrada de control de ALTA IMPEDANCIA.

El símbolo de las compuertas triestados son idénticos a las compuertas convencionales al que se le incluye un terminal adicional. Si el terminal hace contacto con el borde de la compuerta con una pequeña circunferencia, entonces decimos que la habilitación se realiza con nivel bajo, en caso contrario, se activa con nivel alto.

Una de las compuertas triestado se incorpora en el Circuito Integrado 74125. Contiene cuatro **buffers** de una entrada y una salida. Cada compuerta tiene un terminal adicional para habilitarlo o deshabilitarlo.

El **buffer** es una compuerta que recibe una señal, lo amplifica proporcionándole más corriente y lo transfiere tal cual. Normalmente si la salida de una compuerta lógica no tiene suficiente corriente para encender un LED por ejemplo, se le acopla un buffer en su salida y el problema se resuelve, sin embargo, si algún dispositivo requiere más corriente se tendrá que instalar un buffer con colector abierto que puede proporciona más corriente que un buffer convencional.

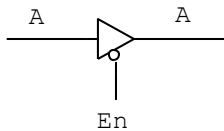


Fig. 6.8

Símbolo de una compuerta buffer triestado.

Habilitación	Salida
(En)	Estado de la compuerta
0	0 ó 1
1	Flotante (alta impedancia)

Tabla. 6.5

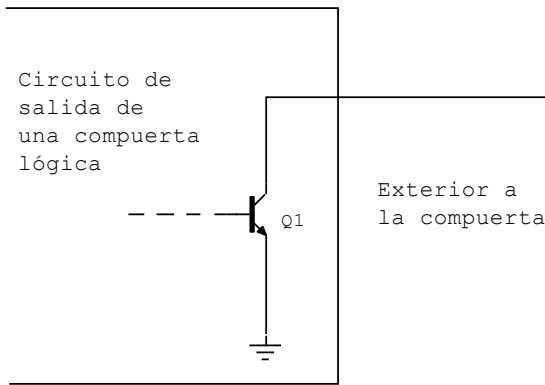
Estado de la compuerta según el valor del terminal de habilitación (En)

La aplicación de las compuertas triestado se dan básicamente en el acoplamiento múltiple de buses de señal, es decir, varias compuertas de este tipo que son la salida de señal de uno o varios circuitos se enlazan a una línea de datos, de dirección o de control, sin que se establezca entre ellos un conflicto en la transmisión de la señal, pues, con el dispositivo de control con que cuentan permite el acceso de uno de ellos a la vez.

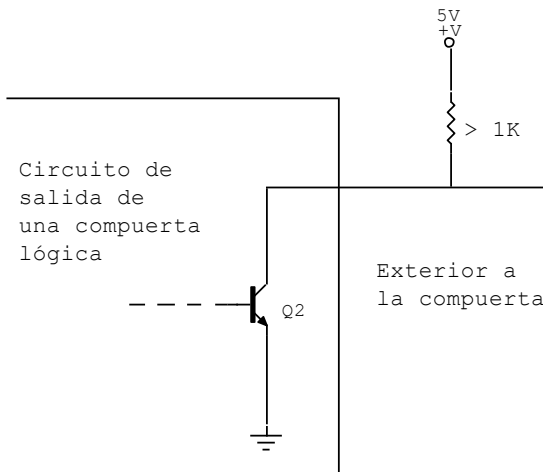
6.4.2. Salida de una compuerta en Colector Abierto (Open colector)

Las compuertas lógicas internamente están implementadas por elementos básicos llamados transistores, ya sea del tipo BJT o FET. En cualquiera de los dos casos, la salida de la compuerta lo proporciona el colector (TTL) o drenador (CMOS).

En una salida convencional, el colector o drenador del transistor mantiene una conexión con la fuente de alimentación a través de una resistencia de colector; sin embargo, en el caso especial de salida donde el colector o drenador no cuenta con resistencia de colector y además no se halla conectado al sistema de alimentación se conoce con el nombre de **colector abierto** "open colector".



(a)



(b)

Fig. 6.9

(a) Salida de una compuerta lógica con colector abierto. Se observa que el colector no tiene conexión con la fuente de alimentación.

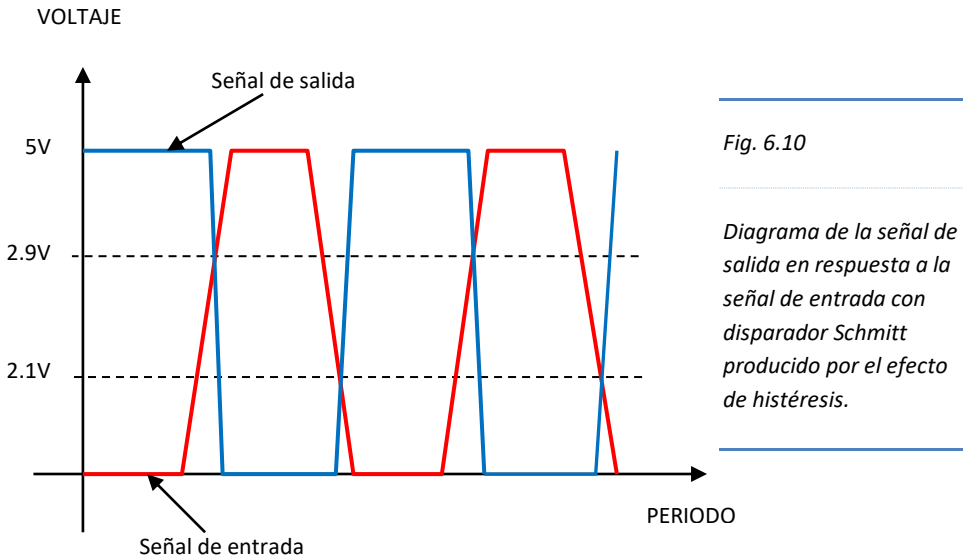
(b) Resistencia de pull up en el colector del transistor de salida con valores mayores que 1K.

Para que éste tipo especial de compuerta funcione adecuadamente es necesario que externamente se conecte una resistencia llamada “pull up” con valores superiores a 1K, para que a través de ésta se conecte a la fuente de alimentación y proporcione los voltajes y corrientes en los niveles adecuados para que funcionen otros circuitos a los que se acopla. La corriente típica que puede proporcionar un buffer con colector abierto es de 25mA y trabajar con voltajes hasta de 30 V de fuente de alimentación.

6.4.3. Compuertas con entradas de disparador Schmitt (Schmitt trigger)

Las compuertas lógicas de los circuitos integrados se comportan de manera anormal cuando inciden sobre ellas algunas señales indeseadas perturbando el normal funcionamiento de un circuito digital. Los ruidos eléctricos son producidos por el encendido o apagado de un artefacto eléctrico, radiación electromagnética o espurios del propio circuito.

Las compuertas con Schmitt trigger tienen la propiedad de responder a niveles de voltaje diferente para los valores lógicos de las entradas. Si la entrada de un inversor con Schmitt trigger como el 7414, se encuentra en 0V, la salida estará en 5V aproximadamente; si el voltaje de entrada empieza a subir, en los 2.9V, la salida pasará a un nivel bajo. Si estando la entrada estabilizada en 5V aproximadamente comenzara a bajar el voltaje, llegando a los 2.9V, la salida no pasará a un nivel alto, ya que deberá bajar hasta los 2.1V aproximadamente para que la salida responda a éste voltaje.



Éste cambio de voltaje es coherente con la lógica digital tratada en la sección 1.3.3 y lo establecido en la Fig. 1.24.

La diferencia de voltaje de 0.8V para activar el nivel alto y bajo de la salida de una compuerta se produce gracias al retardo producido por el efecto llamado “histéresis”.

Las compuertas con disparador Schmitt tienen múltiples aplicaciones en situaciones en donde se quiere evitar los ruidos eléctricos.

El símbolo de una compuerta Schmitt trigger como el inversor 7414 y la compuerta NAND de cuatro entradas incluidas en el C.I. 7413 son los siguientes:

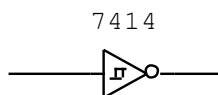
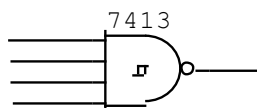


Fig. 6.11



Símbolos de la compuerta NOT 7414 y de la compuerta NAND 7413, con disparador Schmitt.

Problemas Resueltos N° 06

1. Diseñe e implemente un circuito combinacional que genere la palabra “SIPAn” en un display ánodo común (visualizador numérico de siete segmentos).

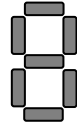
Solución:

Para diseñar un circuito combinacional se tendrá en cuenta lo siguiente

- a. **El número de variables está relacionado con el número de combinaciones.**
La palabra “SIPAn” tiene cinco combinaciones; una combinación por cada letra. Por tanto, se necesita tres variables de entrada, pues con ellas tendremos hasta ocho combinaciones, de las cuales utilizaremos las cinco primeras, y las tres últimas son combinaciones no especificadas.
- b. **Un circuito combinacional puede tener más de una salida. El número de salidas no tiene relación con el número de variables de entrada.**

Un circuito combinacional se diseña para que tenga una o varias salidas, es decir, depende de los requerimientos y necesidades planteadas.

Para implementar el circuito del problema nos piden utilizar como salida un display o visualizador de siete segmentos o diodos (siete salidas), designados con las letras (a-g), tal como se describe en el ítem 8.1.2.1.



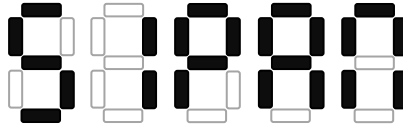
En la combinación 0, para visualizar la letra **S** se deben activar los led: a, c, d, f y g.

En la combinación 1, para visualizar la letra **I** se deben activar los led: b y c; podrían ser también e y f.

En la combinación 2, para visualizar la letra **P** se deben activar los led: a, b, e, f y g.

En la combinación 3, para visualizar la letra **A** se deben activar los led: a, b, c, e, f y g.

En la combinación 4, para visualizar la letra **n** se deben activar los led: a, b, c, e y f.

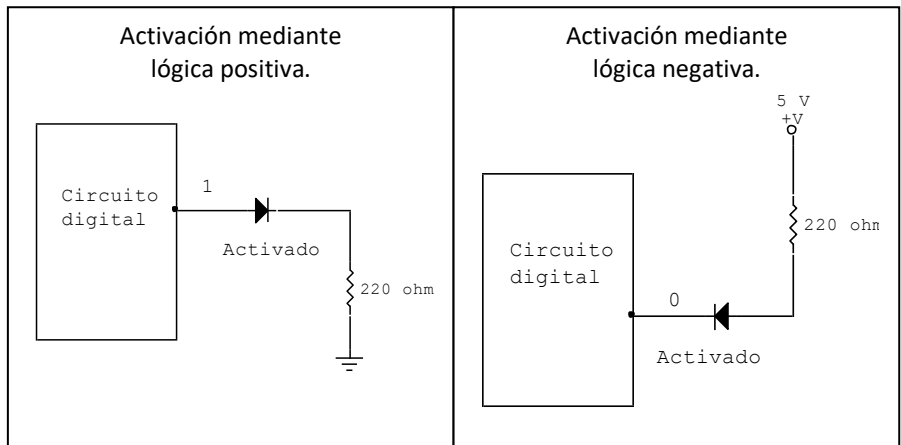


c. Se debe adoptar previamente el valor lógico de la salida.

La lógica de la salida, significa que previo al diseño se debe adoptar la lógica positiva o negativa. Si elegimos la lógica positiva, la salida debe activarse con 1 y desactivarse con 0. Si adoptamos la lógica negativa, la salida debe activarse con 0 y desactivarse con 1.

La visualización de la salida de un circuito lógico se realiza mediante diodos led.

Para activar o polarizar directamente un diodo led según cada lógica.



El display ánodo común funciona con lógica negativa.

Elaborando tabla de verdad.

Nº	VARIABLES DE ENTRADA			SALIDAS							Letra
	A	B	C	a	b	c	d	e	f	g	
0	0	0	0	0	1	0	0	1	0	0	S
1	0	0	1	1	0	0	1	1	1	1	l
2	0	1	0	0	0	1	1	0	0	0	P
3	0	1	1	0	0	0	1	0	0	0	A
4	1	0	0	0	0	0	1	0	0	1	n
5	1	0	1	X	X	X	X	X	X	X	
6	1	1	0	X	X	X	X	X	X	X	
7	1	1	1	X	X	X	X	X	X	X	

Hallando las funciones lógicas de cada salida mediante minitérminos.

$$a = \bar{A}\bar{B}C + A\bar{B}C = \bar{B}C(A + \bar{A}) = \bar{B}C$$

$$b = \bar{A}\bar{B}\bar{C}$$

Diagrama lógico

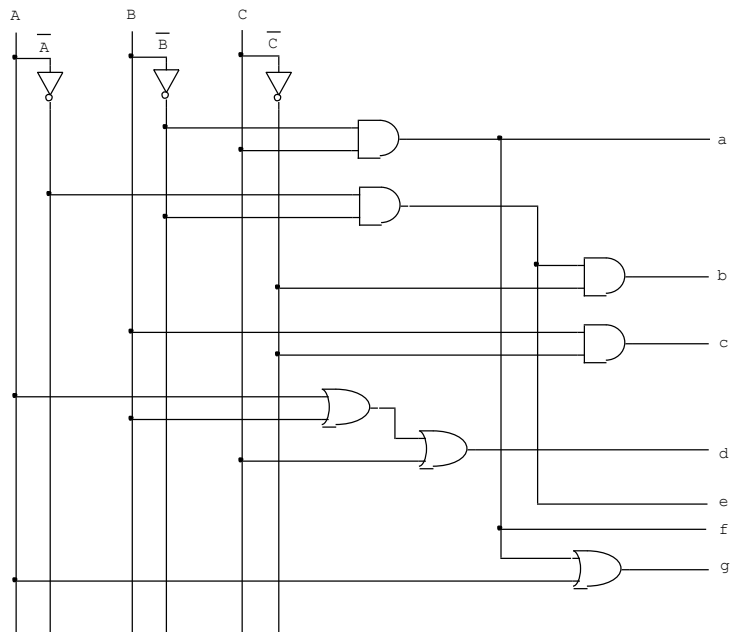
$$c = B\bar{C}$$

$$d = A + B + C$$

$$e = \bar{A}\bar{B}$$

$$f = \bar{B}C$$

$$g = A + \bar{B}C$$

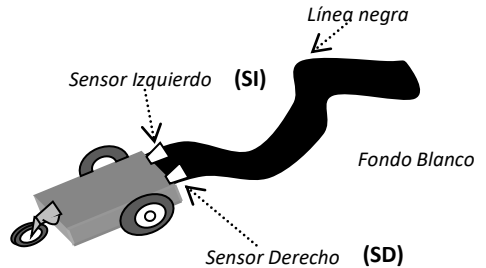


2. Diseñe un circuito digital de control para gobernar el movimiento de las ruedas de un seguidor de línea.

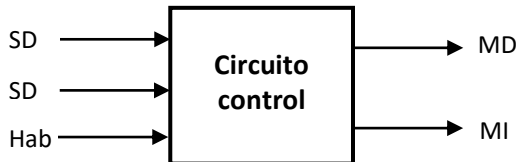
Solución:

El seguidor de línea es un robot móvil que se desplaza siguiendo una línea de color negro, para tal fin dispone de un par de sensores ópticos los cuales pueden distinguir la línea negra del fondo blanco en el que se encuentra.

Existen muchos seguidores de línea, desde los controlados por compuertas lógicas sencillas hasta los controlados por “microcontroladores”.



En nuestro caso, se trata de diseñar un circuito digital de control mediante compuertas lógicas que gobierne el movimiento de las ruedas, las cuales son activadas por dos pequeños motores eléctricos de corriente continua (normalmente los usados en juguetes); que se activan con un nivel lógico **1**, es decir cuando se aplica el voltaje de fuente (Vcc), y se detienen con un nivel lógico **0**, ó 0 voltios.



decir cuando se aplica el voltaje de fuente (Vcc), y se detienen con un nivel lógico **0**, ó 0 voltios.

Para simplificar el diseño, el giro hacia la derecha o hacia la izquierda que realizará nuestro dispositivo móvil se realizará con la combinación de movimientos del motor izquierdo (MI) y del motor derecho (MD) de la siguiente manera:

Sin movimiento: MD = 0; DI = 0 (detenido)

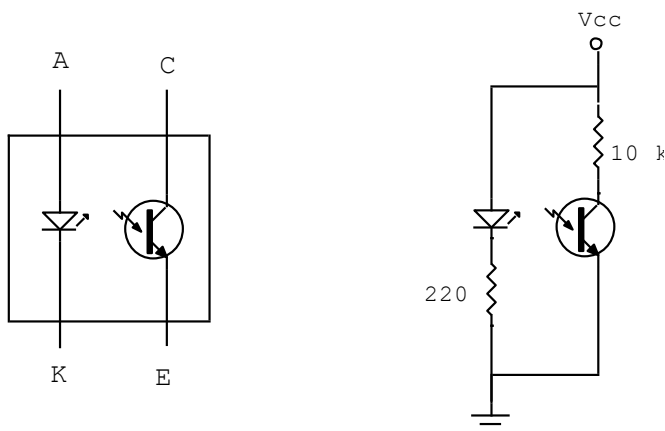
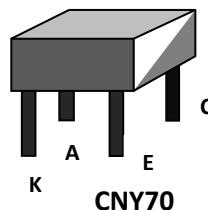
Movimiento hacia la derecha: MD = 0 (detenido); MI = 1 (en movimiento).

Movimiento hacia la izquierda: MD = 1 (en movimiento); MI = 0 (detenido).

Movimiento recto: MI = 1; MD = 1

Nº	MD	MI	Movimiento
0	0	0	Detenido
1	0	1	Hacia la derecha
2	1	0	Hacia la izquierda
3	1	1	En línea recta

Los sensores ópticos están constituidos por un fototransistor **detector** de luz infrarroja. En algunos dispositivos comerciales como el CNY70, el emisor (diodo **emisor** de luz infrarroja) y el detector están ubicados perpendicularmente a la superficie de desplazamiento, de tal manera que si el emisor de luz proyecta su luz sobre la superficie oscura (negra), ésta no se reflejará y no será detectada por el fototransistor; sin embargo, si la superficie es blanca, se reflejará y será captada por el fototransistor.



Si la luz incide sobre una superficie oscura, el fototransistor entrará en corte, pues al no existir corriente de base, no hay corriente de colector y por tanto, el voltaje entre colector y emisor es V_{cc} , es decir, $V_{ce} = 1$.

Si la luz incide sobre una superficie blanca, el fototransistor entrará en saturación, pues, al existir corriente de base, la corriente de colector provoca una caída de tensión en la resistencia de colector, colocando al voltaje entre colector y emisor a 0, es decir, $V_{ce} = 0$.

El movimiento del móvil de acuerdo al nivel lógico de ambos sensores es el siguiente:

Nº	SD	SI	MOVIMIENTO
0	0	0	Detenido
1	0	1	A la izquierda
2	1	0	A la derecha
3	1	1	Línea recta

La línea de habilitación (**Hab**), permite activar o desactivar el dispositivo móvil. Si $Hab = 0$, desactivado; si $Hab = 1$, activado.

Nº	ENTRADA			SALIDA	
	Hab	S.D	S.I	MD	MI
0	0	0	0	0	0
1	0	0	1	0	0
2	0	1	0	0	0
3	0	1	1	0	0
4	1	0	0	0	0
5	1	0	1	1	0
6	1	1	0	0	1
7	1	1	1	1	1

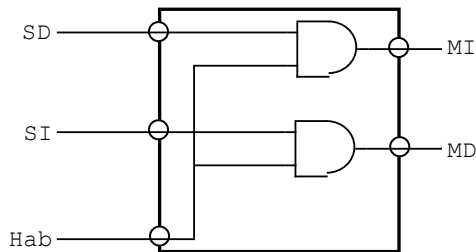
Hallando la expresión lógica de los motores.

$$MD = Hab. \overline{SD}. SI + Hab. SD. SI$$

$$MI = Hab. SD$$

$$MI = Hab. SD. \overline{SI} + Hab. SD. SI$$

$$MD = Hab. SI$$

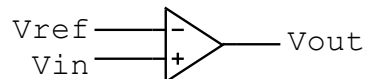


Circuito Control

Observación: la salida de los sensores antes de pasar al circuito de control debe ingresar a un circuito comparador, cuya salida sea una señal digital. Un circuito comparador consiste en un amplificador operacional con una alimentación simple (no simétrica), para una salida digital.

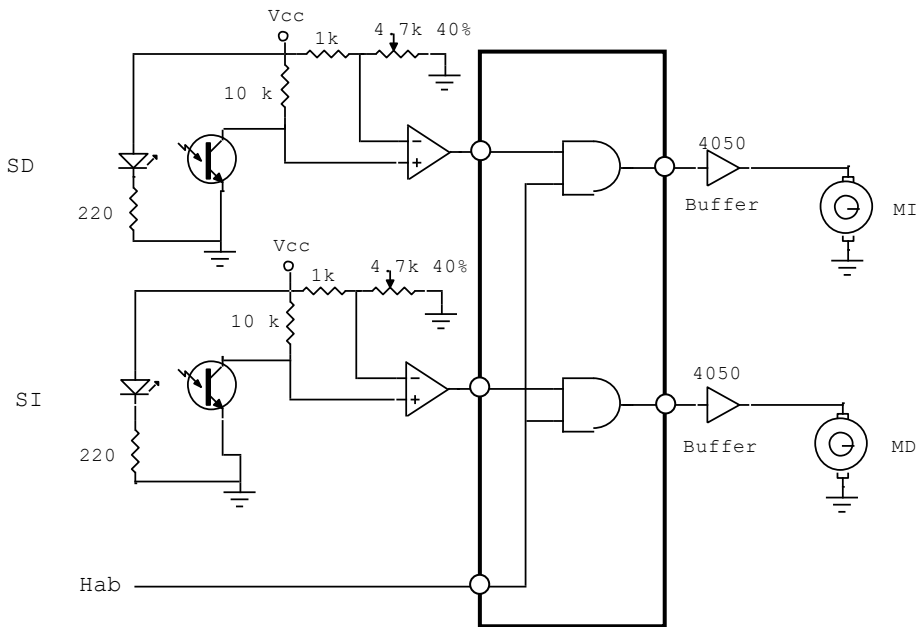
Si $V_{in} > V_{ref}$, entonces $V_{out} = V_{cc}$

Si $V_{ref} > V_{in}$, entonces $V_{out} = 0 V$.



Donde el voltaje de referencia (V_{ref}) se establece en 2.0 V, ajustando convenientemente una resistencia variable de 4.7 K.

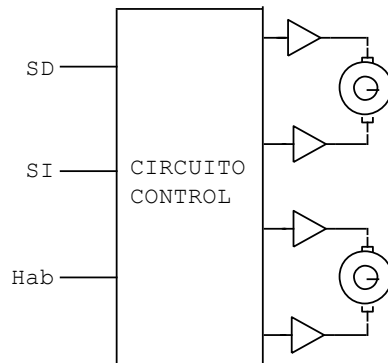
La salida del circuito control, debe pasar a unos buffers antes de acoplarlo a los motores, con la finalidad de proporcionarle potencia suficiente para realizar movimiento y al mismo tiempo aislarlo del circuito control.



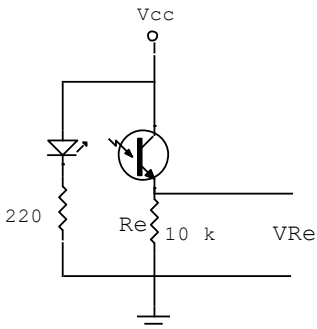
Circuito Control

También se puede diseñar un circuito control para realizar dos movimientos en el motor: hacia adelante y hacia atrás (sentido de giro), lo cual se logra con un circuito integrado como el L293B, que contiene cuatro drivers.

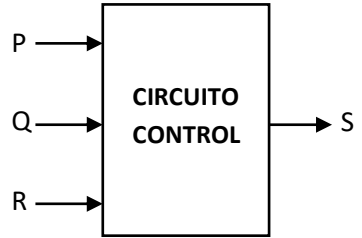
Para los sensores (fototransistores) también se puede configurar dicho transistor con una resistencia de emisor (R_e), de cuya caída de tensión se toma el nivel lógico del sensor; es decir, si la superficie



es negra, el Voltaje en R_e (V_{Re}) = 0 V, pues la corriente de emisor es 0 A; en cambio, si la superficie es blanca, hay corriente de de base y por tanto corriente de emisor, lo que provoca una caída de tensión en R_e . (V_{Re} puede llegar hasta V_{cc} si la corriente de emisor es máxima).



3. El sistema de control de una impresora laser debe generar una señal (S) para detener su funcionamiento y al mismo tiempo mostrar su estado mediante la iluminación de un diodo led cuando se presenta las siguientes situaciones: cuando se haya agotado o no tenga papel en la bandeja alimentadora, ó, cuando se presente un atasco de papel lo cual provoca que los dos sensores en la trayectoria del papel se activen. Cuando la impresora trabaja normalmente, el paso del papel activa un sensor a la vez (Q ó R).



Cuando la bandeja de papel tiene una o varias hojas, el sensor (P) se activa colocándose a nivel Alto; mientras que los dos sensores (Q y R) en la trayectoria de papel se activan con nivel Bajo. Diseñe un circuito lógico sólo con compuertas NAND en el que la salida esté en nivel Bajo cuando se presenta las condiciones anteriores.

Solución:

Para diseñar un circuito digital se debe tener en cuenta la lógica de los sensores y la lógica de la salida del circuito.

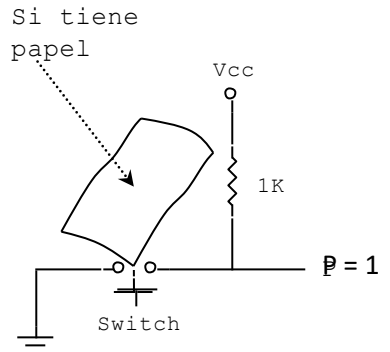
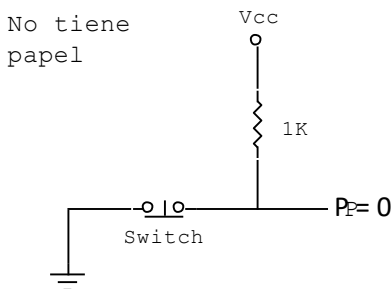
Se observa tres sensores en el circuito control, lo que significa tres variables de entrada.

Evaluando la lógica del sensor de la bandeja alimentadora de papel (P).

Si la bandeja no tiene papel, el nivel lógico de P es **0**.

Si la bandeja tiene papel, el nivel lógico de P es **1**.

El **Sensor P**, es del tipo *switch* o *interruptor normalmente cerrado*, es decir, si la bandeja alimentadora no tiene papel, el *switch* se conecta a tierra ó 0 V; mientras que si la bandeja tiene papel, el *switch* se abre, se desconecta de tierra y se conecta con fuente Vcc, a través de la resistencia de 1k.

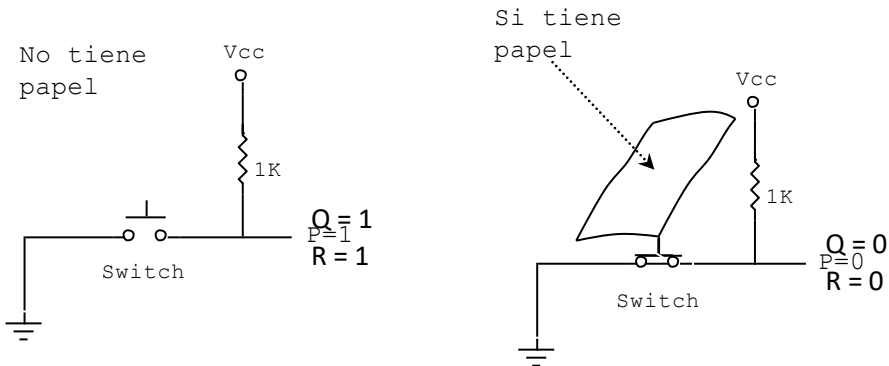


Evaluando la lógica de los sensores de la trayectoria del papel (Q y R).

Si por la trayectoria del papel, no pasa papel, el nivel lógico de Q y R es 1.

Si por la trayectoria del papel, pasa papel, el nivel lógico de Q y R es 0.

Los **Sensores Q y R**, son del tipo de *switch o interruptor normalmente abierto*, es decir, si por la trayectoria de papel, no pasa papel, el *switch* abierto se conecta a fuente o Vcc a través de la resistencia de 1k; mientras que si la trayectoria de papel, tiene papel, el *switch* se cierra, y se conecta directamente a tierra o 0 V.



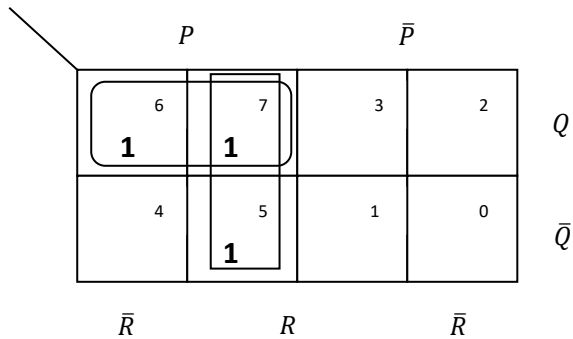
Evaluando la lógica de la salida del circuito.

Cuando se presentan las condiciones planteadas, la salida del circuito deberá restablecerse en nivel bajo ($S = 0$), en los demás casos, deberá establecerse en nivel alto ($S = 1$).

- Si la bandeja alimentadora de papel está en nivel bajo, $P=0$; la salida $S = 0$.
- Si la bandeja alimentadora de papel está en nivel alto, $P=1$, y si los sensores en la trayectoria de papel están en nivel bajo, $Q = 0$ y $R = 0$; la salida $S = 0$.
- En los demás casos, la salida $S = 1$.

Nº	ENTRADA			SALIDA
	P	Q	R	S
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	0
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

Hallando la función lógica (S) del circuito digital mediante Mapa de Karnaugh de tres variables, tenemos:

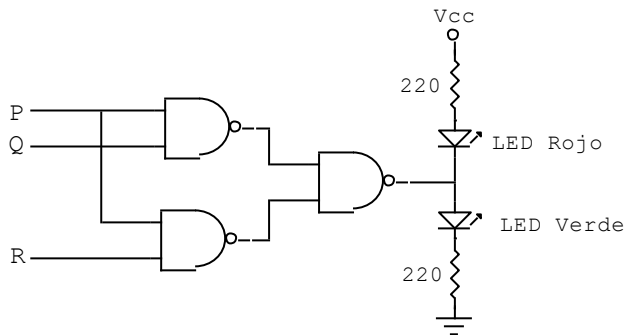


$$S = PQ + PR$$

Para utilizar sólo compuertas NAND, se aplica la propiedad de involución y luego se desarrolla la barra interior mediante el Teorema de Morgan.

$$S = \overline{\overline{PQ} + \overline{PR}} = \overline{\overline{PQ} \cdot \overline{PR}}$$

La visualización del estado de PARO se realiza a través de un diodo LED de color rojo, cuyo cátodo se conecta con la salida del circuito. Y el ánodo de un diodo LED color verde conectado a la salida del circuito, para indicar su estado de FUNCIONAMIENTO.



- Un comité consta de un presidente y tres miembros. Se requiere de un panel electrónico para indicar la decisión del comité. Si las decisiones se toman por mayoría y cuando hay empate prevalece la decisión del presidente; proponga el circuito lógico necesario utilizando bloques combinatoriales, teniendo en cuenta que el dispositivo de votación registra los votos a favor con un nivel bajo, mientras que los votos en contra se registra con un nivel alto. La salida del circuito digital utiliza lógica positiva.

Solución:

El circuito digital tiene cuatro variables de entrada y una sola salida.

Variables de entrada.

P: Presidente.

M1: Primer miembro.

M2: Segundo miembro.

M3: Tercer miembro.

Las decisiones **a favor** se registran con un nivel **Bajo**: P=0; M1=0; M2=0; M3=0; y las decisiones **en contra** con un nivel **Alto**: P=1; M1=1; M2=1; M3=1.

Salida.

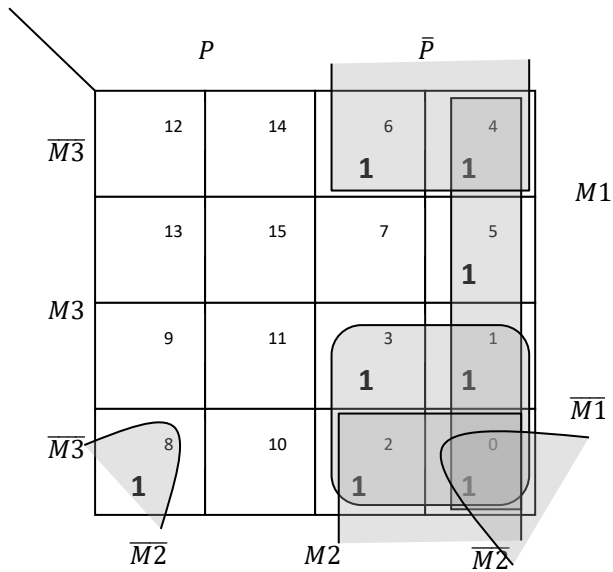
La salida del circuito digital responde a la lógica positiva, es decir, cuando la mayoría está **a favor**, el nivel lógico de **S es Alto**; si está **en contra**, el nivel lógico de **S es Bajo**.

Cuando hay empate (cuando dos integrantes están a favor y dos están en contra), en ésta condición, prevalece la decisión del presidente: Si el presidente está a favor (P=0), la Salida S = 1; si está en contra (P=1), la Salida S=0.

Tabla de verdad.

Nº	ENTRADA				SALIDA
	P	M1	M2	M3	S
0	0	0	0	0	1
1	0	0	0	1	1
2	0	0	1	0	1
3	0	0	1	1	1
4	0	1	0	0	1
5	0	1	0	1	1
6	0	1	1	0	1
7	0	1	1	1	0
8	1	0	0	0	1
9	1	0	0	1	0
10	1	0	1	0	0
11	1	0	1	1	0
12	1	1	0	0	0
13	1	1	0	1	0
14	1	1	1	0	0
15	1	1	1	1	0

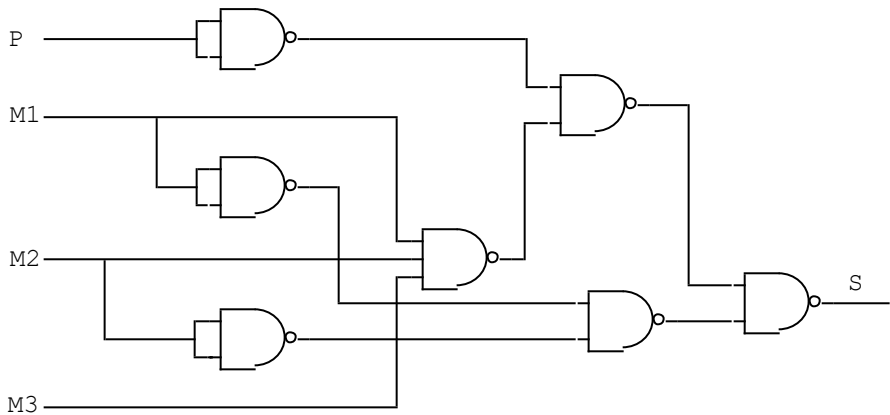
Utilizando un Mapa de Karnaugh de cuatro variables se encuentra la expresión lógica de función de Salida (S).



$$S = \bar{P} \cdot \bar{M1} + \bar{P} \cdot \bar{M2} + \bar{P} \cdot \bar{M3} + \bar{M1} \cdot \bar{M3}$$

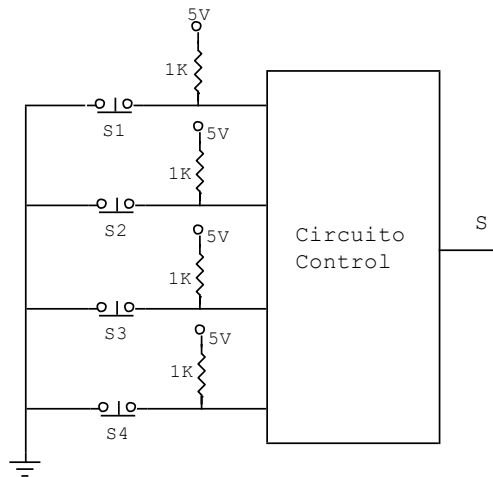
$$S = \bar{P}(\bar{M1} + \bar{M2} + \bar{M3}) + \bar{M1} \cdot \bar{M2} = \bar{P}(\overline{M1 \cdot M2 \cdot M3}) + \bar{M1} \cdot \bar{M2}$$

$$S = \overline{\overline{\overline{P(M1 \cdot M2 \cdot M3)}}} \cdot \bar{M1} \cdot \bar{M2}$$



Problemas Propuestos N° 06

1. Diseñe un circuito combinacional que genere la palabra "FISICA" y cuya visualización se realice en un display cátodo común.
2. Diseñe el circuito de control de los dos motores que activan las ruedas de un seguidor de línea cuyo desplazamiento se realiza a lo largo de una línea blanca y fondo negro.
3. El sistema de control de una impresora cuenta con cuatro sensores de papel (S1, S2, S3 y S4) en estado normalmente cerrado. Los sensores están distribuidos a lo largo de la ruta del papel en el interior del equipo. Cuando el papel pasa por el sensor, éste se abre generando un nivel Alto. Los sensores S1 y S4 nunca se activarán al mismo tiempo. Diseñe un circuito digital cuya salida esté en Alto cuando dos o más sensores están activados. Tenga en cuenta que las condiciones imposibles "nunca", se pueden considerar combinaciones cuya salida pueden ser 0 ó 1 según convenga.



4. Un comité consta de 4 integrantes: un jefe y tres miembros, responsables del acceso a una caja fuerte, la cual se puede abrir bajo las siguientes condiciones:
 - a. Con la aceptación del jefe y cualquier otro miembro.
 - b. Con la aceptación de los tres miembros, sin importar la opinión del jefe.Se requiere de un sistema de control digital en el que cada integrante del comité emite su opinión pulsando un interruptor normalmente abierto. Diseñar el circuito lógico necesario para controlar dicho acceso.

Codificadores y multiplexores

7.1. Codificadores

Los codificadores son circuitos combinatoriales que cumplen la función de conversión entre un código cuyas líneas de entrada es menor o igual a 2^n , a otro código con n líneas de salida. Los codificadores funcionan activando una línea de entrada y el proceso genera una combinación de valores lógicos en las líneas de salida.

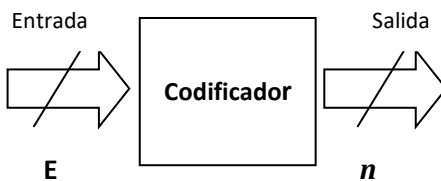


Fig. 7.1

Diagrama de bloques de un codificador, donde las líneas de entrada puede ser menor o igual a 2^n

$$\begin{array}{ccc} \text{Líneas de Entrada} & \text{---} & \text{Líneas de Salida} \\ & \swarrow \quad \nwarrow & \\ & E \leq 2^n & \end{array}$$

El código de numeración octal puede ser convertido a un código binario, solo teniendo en cuenta que la entrada al codificador son **ocho líneas** que corresponden a cada dígito del sistema octal y cuya salida cuenta con **tres líneas** de señal para el código binario.

Un ejemplo típico de un codificador es aquel circuito que mediante la pulsación de una tecla del teclado numérico decimal, lo convierte en código binario entendible por el computador.

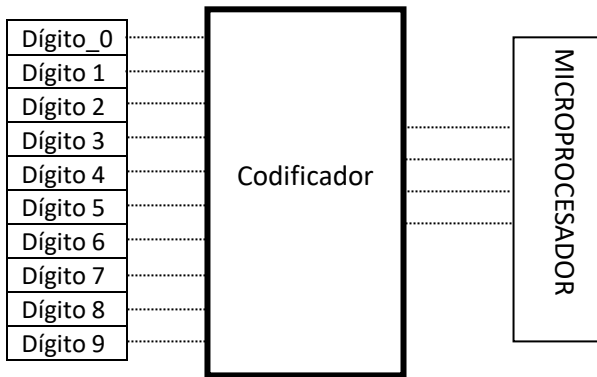
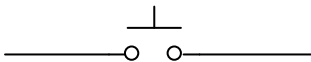


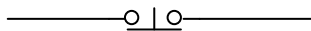
Fig. 7.2.

Diagrama de bloques de un codificador decimal a BCD.

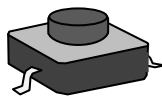
El pulsador ó **tact switch** de una tecla puede estar normalmente abierto (Na) o normalmente cerrado (Nc). En el primer caso, mientras no se accione el dispositivo no hay flujo de corriente, mientras que en el segundo, la corriente está circulando normalmente hasta que se accione el pulsador y la corriente se interrumpe.



(a)



(b)



(c)

Fig. 7.3

(a) Pulsador normalmente abierto (Na).

(b) Pulsador normalmente cerrado (Nc).

(c) Forma física de un pulsador o tact switch.

En la entrada del codificador puede añadirse una extensión adicional para el control del circuito, es decir, para habilitarlo o deshabilitarlo del mundo exterior.

Según la lógica que se utilice (sección 1.3.4), en la salida del codificador pueden acoplarse circuitos inversores para complementar la señal de salida, éste detalle se visualiza en su diagrama lógico mediante unas pequeñas circunferencias en cada línea de salida.

Los codificadores se pueden diseñar sin prioridad o con prioridad.

7.1.1. Codificador sin prioridad

Los codificadores en los cuales se activan simultáneamente dos o más teclas de la entrada y genera un código de error en la salida, se dice que son codificadores sin prioridad, ya que al pulsar al mismo tiempo varias teclas el codificador no responde a ninguno de ellos, por tanto, éste tipo de codificador solamente es útil cuando se pulsa una sola tecla.

Implementación de un codificador sin prioridad de decimal a BCD.

N°	ENTRADA DECIMAL										SALIDA BCD			
	E9	E8	E7	E6	E5	E4	E3	E2	E1	E0	S4	S3	S2	S1
	9	8	7	6	5	4	3	2	1	0	8	4	2	1
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	0	0	1	0	0	0	0	1
2	0	0	0	0	0	0	0	1	0	0	0	0	1	0
3	0	0	0	0	0	0	1	0	0	0	0	0	1	1
4	0	0	0	0	0	1	0	0	0	0	0	1	0	0
5	0	0	0	0	1	0	0	0	0	0	0	1	0	1
6	0	0	0	1	0	0	0	0	0	0	0	1	1	0
7	0	0	1	0	0	0	0	0	0	0	0	1	0	1
8	0	1	0	0	0	0	0	0	0	0	1	0	0	0
9	1	0	0	0	0	0	0	0	0	0	1	0	1	1

Tabla 7.1.

Tabla de verdad de un codificador sin prioridad de entrada decimal y salida BCD.

Evaluando la salida en base a los minitérminos:

$$S4 = E8 + E9$$

$$S3 = E4 + E5 + E6 + E7$$

$$S2 = E2 + E3 + E6 + E9$$

$$S1 = E1 + E3 + E5 + E7 + E9$$

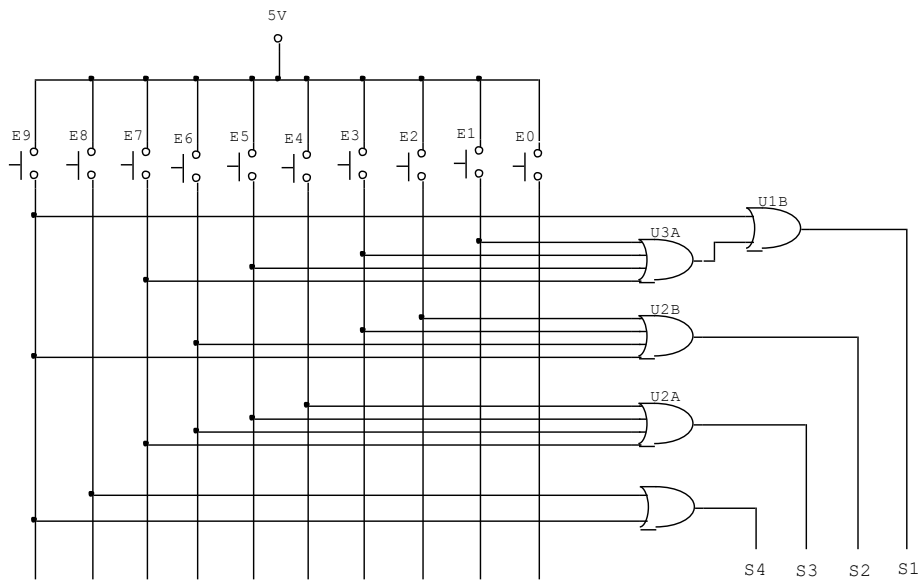


Fig. 7.4

Diagrama lógico de un codificador sin prioridad de Decimal a BCD.

7.1.2. Codificador con prioridad

Los codificadores en los cuales se activan simultáneamente dos o más teclas de la entrada, y el circuito codificador responde a la tecla cuyo dígito sea más alto, decimos que se trata de un codificador con prioridad, pues, sólo responde al nivel más alto del código de entrada e ignora a los demás.

Los codificadores con prioridad tienen un diseño y estructura mucho más complejo que el codificador sin prioridad. La gran mayoría de codificadores comerciales son de este tipo.

El codificador con prioridad 74147 convierte el código decimal

representado por 9 dígitos al código BCD. Las nueve entradas se activan en nivel BAJO, es decir, se conecta a tierra. Su salida se encuentra invertida.

Los dígitos de entrada (E0, E1, E2, E3, E4, E5, E6, E7, E8 y E9) se conectan a través de un pulsador normalmente abierto e instalado a tierra. Cada vez que se pulsa el tact switch la entrada se pone en nivel BAJO e inmediatamente activa la entrada del codificador.

La salida (D, C, B y A) en código BCD, con D como el MSB y A como el LSB, se encuentran invertidas, es decir, el 5 BCD normal es (0 1 0 1), mientras que el 5 BCD invertido es (1 0 1 0).

N°	DÍGITOS DECIMALES DE ENTRADA										SALIDA CÓDIGO BCD			
	E9	E8	E7	E6	E5	E4	E3	E2	E1	E0	D	C	B	A
	9	8	7	6	5	4	3	2	1	0	8	4	2	1
0	1	1	1	1	1	1	1	1	1	0	1	1	1	1
1	1	1	1	1	1	1	1	1	0	X	1	1	1	0
2	1	1	1	1	1	1	1	0	X	X	1	1	0	1
3	1	1	1	1	1	1	0	X	X	X	1	1	0	0
4	1	1	1	1	1	0	X	X	X	X	1	0	1	1
5	1	1	1	1	0	X	X	X	X	X	1	0	1	0
6	1	1	1	0	X	X	X	X	X	X	1	0	0	1
7	1	1	0	X	X	X	X	X	X	X	1	0	0	0
8	1	0	X	X	X	X	X	X	X	X	0	1	1	1
9	0	X	X	X	X	X	X	X	X	X	0	1	1	0

Tabla 7.2

Tabla de verdad del codificador 74147 con prioridad

Una breve inspección de la tabla nos revela que mientras se pulse el dígito de mayor valor, los de menor valor pueden estar activados o desactivados, es decir en **0** ó en **1**, esta condición se representa mediante una **X**, pero, el circuito responderá sólo al de mayor numeración ignorando a los demás. Se constata asimismo, que la combinación de salida (D C B A) para el pulsador **0** está deshabilitada, pues, como se observa, la salida 1 1 1 1, corresponde a 0 0 0 0, ya que la salida se encuentra invertida, tal como se mencionó anteriormente, por ello, el tact switch **0** no se conectará físicamente al codificador.

Por otro lado, en el proceso de implementación del codificador de prioridad se pueden utilizar cuatro compuertas NOT para visualizar normalmente la salida.

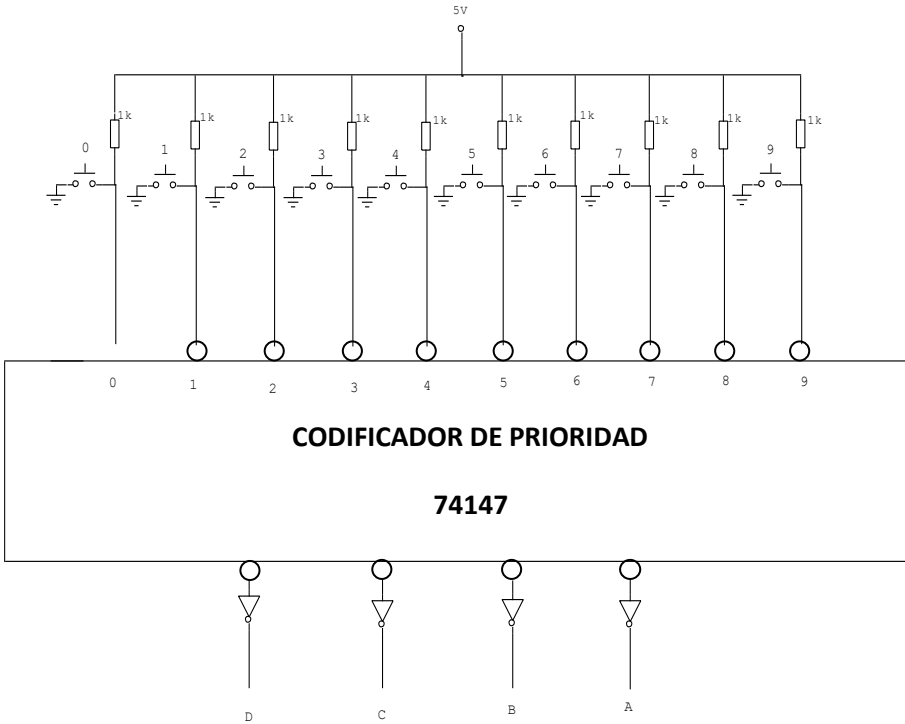


Fig. 7.5

Implementación del 74147 codificador de prioridad.

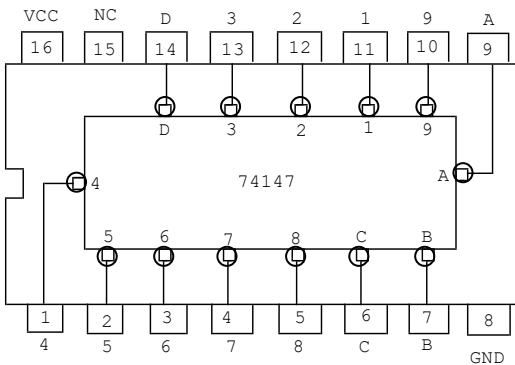


Fig. 7.6

Patillaje del circuito integrado 74147 codificador de prioridad de decimal a BCD.

7.2. Multiplexores

Los multiplexores (MUX) son circuitos combinaciones cuya función es seleccionar una sola línea de señal o “*canal*” de varios que están presentes en la entrada y, derivarlo a la salida. La selección de dicha línea de entrada, se realiza mediante un conjunto de terminales llamado control.

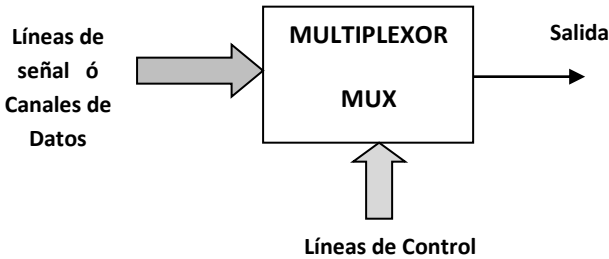


Fig. 7.7

Diagrama de bloques de un multiplexor.

La relación que existe entre las líneas de señal y las líneas de control es el siguiente:

$$D = 2^C$$

de Canales de Datos
de líneas de Control

Por ejemplo, si se desea multiplexar dos canales, es necesario contar con una línea de control.

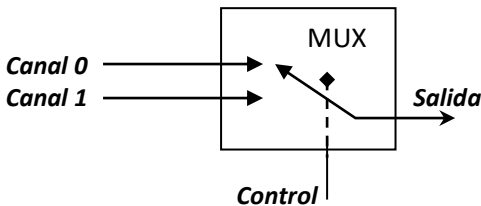


Fig. 7.8

Multiplexor de dos canales

N°	CONTROL	SALIDA
0	0	Canal 0
1	1	Canal 1

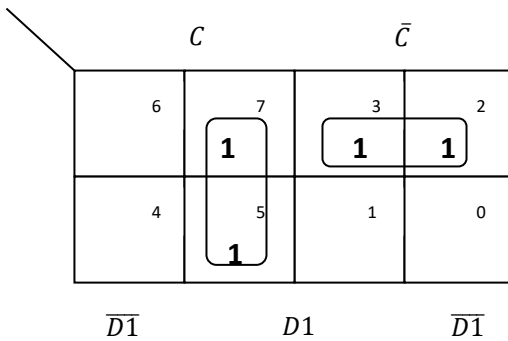
Cuando el **Control** se encuentra en **0**, el **canal 0** pasa a la salida, en cambio, cuando el control está en **1**, es el **canal 1** el que pasa a la salida.

Disponiendo el **Control (C)**, el **Canal 0 (D0)**, el **Canal 1 (D1)** y la **Salida (S)**, en una tabla de verdad obtenemos su diagrama lógico mediante simplificación con un mapa de Karnaugh.

N°	CONTROL	CANAL DE DATOS		SALIDA
	<i>C</i>	<i>D0</i>	<i>D1</i>	<i>S</i>
0	0	0	0	0
1	0	0	1	0
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1

Tabla.7.3

Tabla de verdad de un multiplexor de dos canales



D0

Fig.7.9

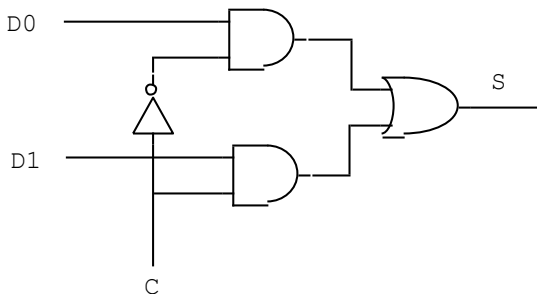
$\overline{D0}$

(a) Mapa de Karnaugh para encontrar la función de un multiplexor de dos canales.

(a)

$$S = \overline{C} \cdot D0 + C \cdot D1$$

(b) Diagrama lógico de un multiplexor de dos canales.



(b)

7.2.1. Diseño de un multiplexor de cuatro canales

Si se desea multiplexar cuatro canales, es necesario contar con dos líneas de control, dicho de otro modo, con dos líneas de control se pueden multiplexar hasta cuatro canales.

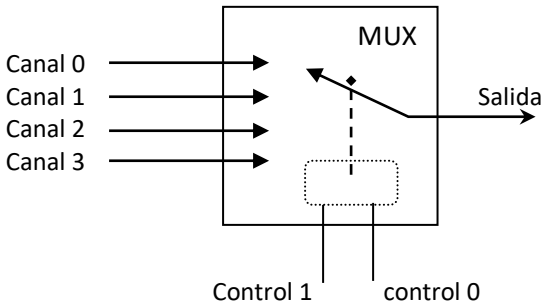


Fig. 7.10

Multiplexor de cuatro canales

N°	CONTROL		SALIDA
	C1	C0	S
0	0	0	Canal 0
1	0	1	Canal 1
2	1	0	Canal 2
3	1	1	Canal 3

Tabla.7.4

Tabla de un multiplexor de cuatro canales

En la Fig. 7.9 (b), se constata que cada Línea de Señal o Canal se enlazan con la línea de control en su respectiva compuerta NAND, los que a su vez se unen a una compuerta OR.

De la Tabla 7.4 se deduce que las combinaciones de **Ct1** y **Ct0** se acoplan en una compuerta **NAND** con su respectivo **Canal**. Estas Líneas de Señal ingresan directamente sin modificación a la compuerta. Finalmente, todas las compuertas AND se unen en una compuerta **OR**.

N°	CONTROL		SALIDA
	C1	C0	S
0	0	0	$S = \overline{C1} \cdot \overline{C0} \cdot D0$
1	0	1	$S = \overline{C1} \cdot C0 \cdot D1$
2	1	0	$S = C1 \cdot \overline{C0} \cdot D2$
3	1	1	$S = C1 \cdot C0 \cdot D3$

Tabla.7.5

Salida de un multiplexor de cuatro canales

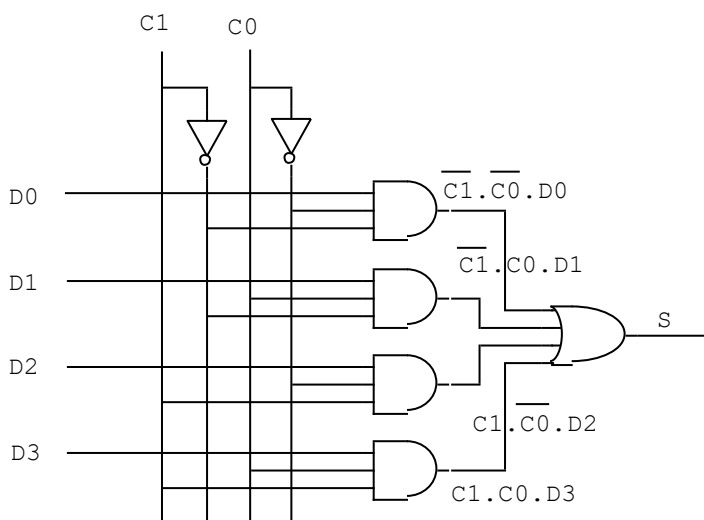


Fig. 7.11

Diagrama lógico de un Multiplexor de cuatro canales

7.2.2. Multiplexor comercial de ocho canales

El circuito integrado 74151 es un multiplexor con tres pines de control (A,B,C), con los que se puede direccionar hasta un máximo de ocho canales de datos (D0 - D7). Además de los pines de control dispone de un terminal llamado STROBE con el que se habilita o deshabilita al multiplexor. Su salida es (Y), mientras que la salida (W) es su complemento.

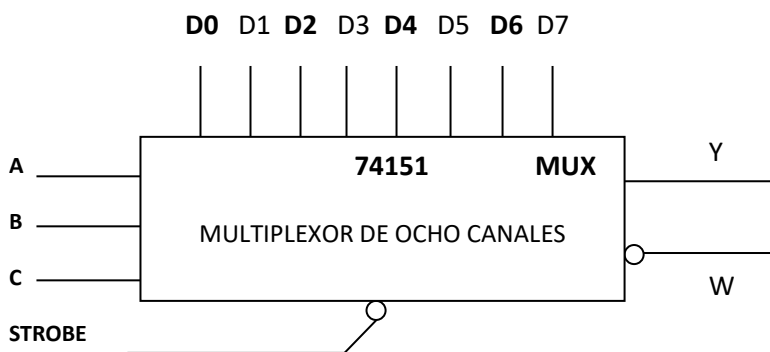


Fig. 7.12

Diagrama de bloque del 74151 Multiplexor de ocho canales con STROBE.

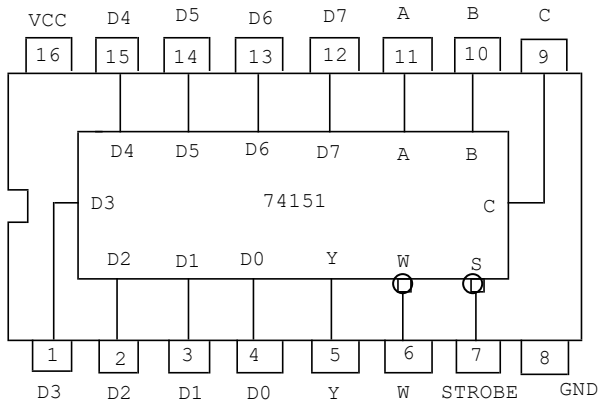


Fig. 7.13

Circuito integrado 74151. Multiplexor de ocho canales con STROBE.

IMPORTANTE:

El terminal STROBE habilita el multiplexor con nivel BAJO, mientras que con un nivel ALTO lo bloquea del mundo exterior.

El circuito integrado 74251, tiene el mismo número de pines y realiza las mismas funciones que el 74151, solo que sus salidas son triestado. Ésta condición es activada por el terminal STROBE.

El circuito integrado 74153 es un doble multiplexor de cuatro canales. El 74157 es un cuádruple multiplexor de dos canales. El 74150 es un multiplexor de 16 canales.

7.2.3. Aplicaciones del multiplexor

Los multiplexores que hemos analizado hasta ahora están relacionados con la selección de líneas de señal o de datos, en la que sólo uno de ellos es transmitido a la salida del circuito, sin embargo, los multiplexores tienen aplicaciones distintas, desde generación de funciones lógicas hasta conversión de señales en formato paralelo a serial.

La generación de funciones lógicas es una aplicación muy importante de los multiplexores, pues permite implementar funciones lógicas elementales y funciones lógicas mucho más complejas.

Para lograr la implementación de dichas funciones, se deberá tener en cuenta que las variables de entrada de la función lógica corresponden a las entradas de control del multiplexor y que los valores combinatoriales de dichas entradas equivalen a las entradas de datos.

Por ejemplo, si deseamos implementar la función lógica NAND de tres variables de entrada, debemos partir de su tabla de verdad, para luego realizar los arreglos pertinentes en el multiplexor para que ejecute la función lógica deseada.

N°	ENTRADA			SALIDA
	C	B	A	NAND
	4	2	1	S
0	0	0	0	1
1	0	0	1	1
2	0	1	0	1
3	0	1	1	1
4	1	0	0	1
5	1	0	1	1
6	1	1	0	1
7	1	1	1	0

Tabla.7.5

Tabla de verdad de una compuerta NAND de tres variables de entrada.

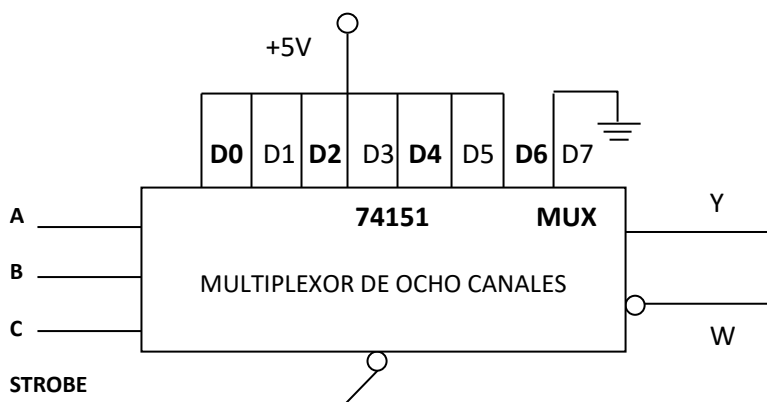


Fig. 7.14

Configuración de un multiplexor de ocho canales para generar la función NAND de tres variables de entrada.

En la Fig. 7.14 se observa que las entradas desde D0 hasta D6 están conectadas a +5V, lo que significa que están conectadas a 1 lógico, en cambio, el terminal correspondiente a D7 se ha instalado a tierra ó 0 lógico. La disposición eléctrica del multiplexor es coherente con la salida de la tabla 7.5; es decir, sólo con la combinación 111, la salida del multiplexor será 0, mientras que en el resto de los casos será 1.

Problemas Resueltos Nº 07

1. Implemente un circuito lógico de un codificador con prioridad 4 a 2, activado con nivel bajo.

Solución:

El codificador con prioridad 4 a 2 convierte un código de 4 bits a uno de 2 bits, de tal manera que si presionamos dos o más bits del sistema de numeración de base 4, obtendremos un número de mayor valor en el sistema de numeración binario equivalente al mayor número de base 4 presionado.

Diseñamos un codificador de prioridad de 4 a 2 elaborando una tabla de verdad en el que las entradas gobernadas por tact switch son: E3, E2, E1 y E0; mientras que la salida tiene dos bits designados por las variables B y A. con A como LSB.

Nº	ENTRADA				SALIDA	
	E3	E2	E1	E0	B	A
0	1	1	1	1	0	0
1	1	1	0	1	0	1
2	1	0	X	1	1	0
3	0	X	X	1	1	1

Considerando que todas las entradas inicialmente están en nivel Alto, pues aún no se ha presionado ningún tact switch; obliga a la salida a restablecerse en nivel Bajo, donde B = 0 y A = 0.

Observando que la línea de Entrada E0 siempre se mantendrá en nivel Alto prescindimos de él por no necesario para el diseño.

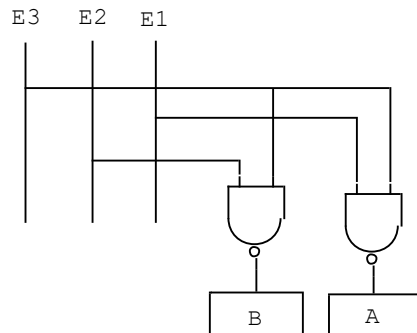
Además, la prioridad de la línea con mayor valor se realizará con otro circuito lógico, por lo que el circuito codificador es similar a un codificador sin prioridad, sólo que en este caso, tendremos en cuenta el nivel bajo de la entrada, es decir, con la entrada negada.

De la tabla tenemos:

$$B = \overline{E2} + \overline{E3} = \overline{\overline{\overline{E2} + \overline{E3}}} = \overline{E2 \cdot E3}$$

$$A = \overline{E1} + \overline{E3} = \overline{\overline{\overline{E1} + \overline{E3}}} = \overline{E1 \cdot E3}$$

El diseño de la prioridad de entrada se consigue estableciendo una relación de selección de una sola línea de salida (S3 ó S2 ó S1) de tres líneas de entrada (E3, E2 y E1), los que a su vez se relacionan así: E3 tendrá salida por S3, E2 tendrá salida por S2, y E1 tendrá salida por S1.

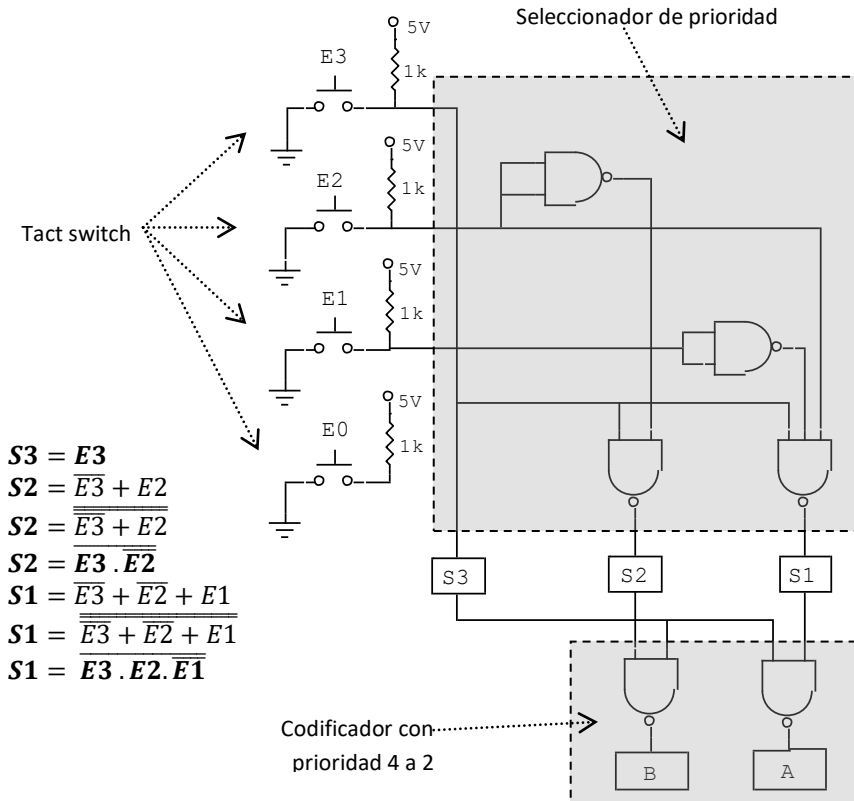


Nº	ENTRADA			SALIDA		
	E3	E2	E1	S3	S2	S1
0	0	0	0	0	1	1
1	0	0	1	0	1	1
2	0	1	0	0	1	1
3	0	1	1	0	1	1
4	1	0	0	1	0	1
5	1	0	1	1	0	1
6	1	1	0	1	1	0
7	1	1	1	1	1	1

En la primera fila se observa que todas las entradas se encuentran activadas, lo que significa que todos los tact switch están presionados. La salida sólo tiene que responder a la línea de mayor valor, es decir a E3, por lo que S3 se activa con nivel bajo. En las filas 1, 2 y 3 responden a E3 por la explicación anterior.

En la fila 4, E3 no está activado, es decir, el tact switch E3 no está presionado, por lo que su salida S3 está desactivado con nivel alto. En la misma fila, E2 y E1 están activados, pero sólo debe responder a E2 por ser de mayor valor, por tanto, S2 se activa con nivel bajo.

Evaluando la salida por el método de la tabla o Mapa de Karnaugh, tenemos:

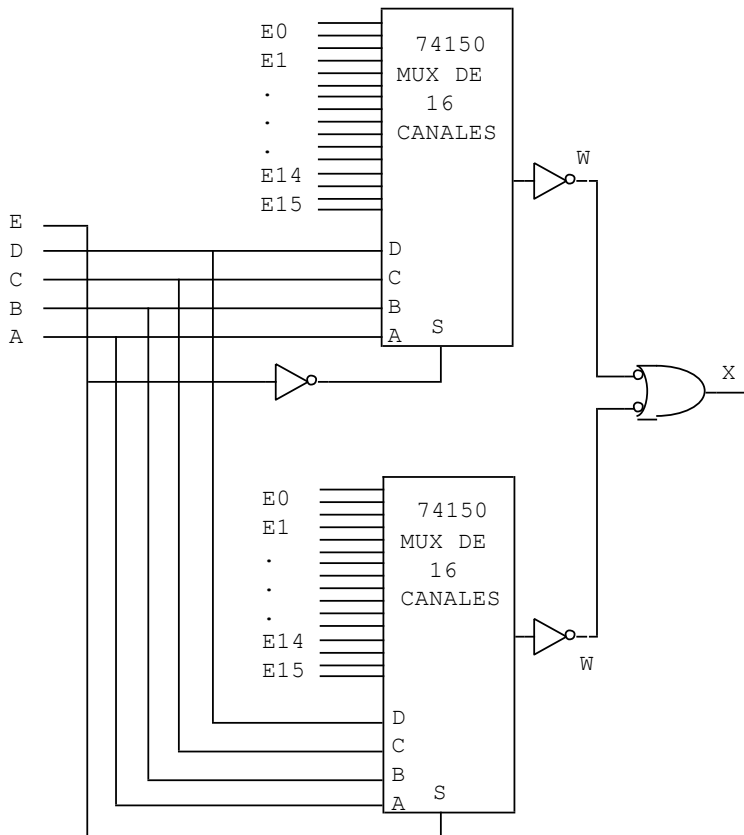


2. Con dos multiplexores de 16 canales de entradas construya un multiplexor de 32 canales.

Solución:

El multiplexor 74150 tiene un conjunto de 16 canales de entrada (E0 – E15) de los cuales sólo uno entrega su data a la salida (W), pero invertida. El mecanismo de selección se realiza a través de un conjunto de líneas llamadas de selección de canal (D,C,B,A) con A como LSB, sin embargo, nada de esto funciona si el multiplexor se encuentra deshabilitado, lo cual se logra mediante una línea llamada STROBE (S). Si STROBE = 1 Habilitado; si STROBE = 0 Deshabilitado.

Para implementar un multiplexor de 32 canales de entradas con dos multiplexores de 16 canales, es necesario alinear los dos Multiplexores cuyas dos salidas ingresan a una compuerta OR previamente complementadas, y controladas por las cuatro líneas de selección de canal y una línea adicional (E) como MSB, conectado a la línea STROBE para habilitar un multiplexor y deshabilitar el otro lo que se logra con un inversor. Las primeras 16 combinaciones habilita sólo al primer multiplexor, las siguientes 16 combinaciones, sólo al segundo multiplexor.



3. Implemente un circuito digital de cuatro variables de entrada A, B, C, y D. con D como MSB, cuya salida S se restablece en nivel Bajo cuando tres o más entradas tienen nivel Alto. Utilice:
- Un multiplexor de cuatro entradas de control solamente.
 - Un multiplexor de tres entradas de control solamente.
 - Un multiplexor de dos entradas de control solamente, y dispone además de compuertas NAND y OR-EXCLUSIVO.

Solución:

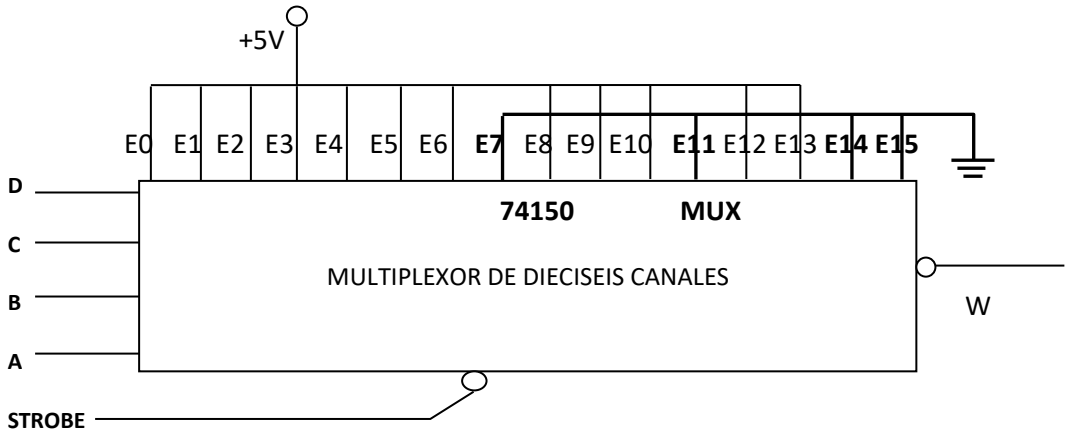
Elaboramos una tabla de verdad para hallar los minitérminos y maxitérminos de la función S.

Nº	ENTRADA				SALIDA
	D	C	B	A	W
0	0	0	0	0	1
1	0	0	0	1	1
2	0	0	1	0	1
3	0	0	1	1	1
4	0	1	0	0	1
5	0	1	0	1	1
6	0	1	1	0	1
7	0	1	1	1	0
8	1	0	0	0	1
9	1	0	0	1	1
10	1	0	1	0	1
11	1	0	1	1	0
12	1	1	0	0	1
13	1	1	0	1	1
14	1	1	1	0	0
15	1	1	1	1	0

De la tabla se observa que $S = f(D, C, B, A) = \prod_M(7, 11, 14, 15)$;

$$S = f(D, C, B, A) = \sum_m(0, 1, 2, 3, 4, 5, 6, 8, 9, 10, 12, 13)$$

- Para implementar la función S con un multiplexor de 16 canales de entrada se utiliza el circuito integrado 74150, en el que las entradas E7, E11, E14 y E15, se conectan a tierra. Las demás entradas E0, E1, E2, E3, E4, E5, E6, E8, E9, E10, E12 y E13, se conectan a nivel alto (+5 V). La salida de la función (W) está invertida.

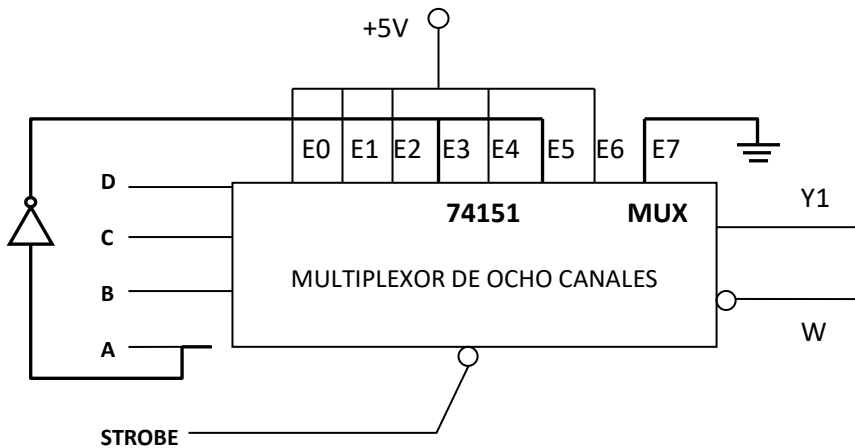


- b. Para implementar la función S con un multiplexor de ocho canales, se puede utilizar un multiplexor 74151, en el que sus variables de control son D , C y B , siendo A una variable adicional al multiplexor.

La tabla de verdad de 16 combinaciones se agrupa de 2 en 2, obteniéndose ocho combinaciones. De cada par de salida (Y) se obtiene 1 salida. Por ejemplo, ambas salidas Y de las combinaciones 0 y 1 tienen valor 1; por tanto $Y1 = 1$. Las combinaciones 6 y 7 son 1 y 0, valores que son los inversos de la variable A en las mismas combinaciones, por lo que $Y1 = \bar{A}$.

Nº	ENTRADA				SALIDA		Nº
	D	C	B	A	Y	Y1	
0	0	0	0	0	1	1	0
1	0	0	0	1	1		
2	0	0	1	0	1	1	1
3	0	0	1	1	1		
4	0	1	0	0	1	1	2
5	0	1	0	1	1		
6	0	1	1	0	1	\bar{A}	3
7	0	1	1	1	0		
8	1	0	0	0	1	1	4
9	1	0	0	1	1		
10	1	0	1	0	1	\bar{A}	5
11	1	0	1	1	0		
12	1	1	0	0	1	1	6
13	1	1	0	1	1		
14	1	1	1	0	0	0	7
15	1	1	1	1	0		

Las entradas E0, E1, E4 y E6 se conectan a fuente; las entradas E3 y E5 se conectan a \bar{A} ; la entrada E7 se alambra a tierra.



- c. Para implementar la función S con un multiplexor de cuatro canales, se puede utilizar el circuito integrado 74153 de dos multiplexores de cuatro canales de entrada, de los cuales se utilizará sólo uno, el otro estará deshabilitado.

Nº	ENTRADA				SALIDA		Nº
	D	C	B	A	Y	Y1	
0	0	0	0	0	1	1	0
1	0	0	0	1	1		
2	0	0	1	0	1		
3	0	0	1	1	1		
4	0	1	0	0	1	$\bar{B} \cdot \bar{A}$	1
5	0	1	0	1	1		
6	0	1	1	0	1		
7	0	1	1	1	0		
8	1	0	0	0	1	$\bar{B} \cdot A$	2
9	1	0	0	1	1		
10	1	0	1	0	1		
11	1	0	1	1	0		
12	1	1	0	0	1	\bar{B}	3
13	1	1	0	1	1		
14	1	1	1	0	0		
15	1	1	1	1	0		

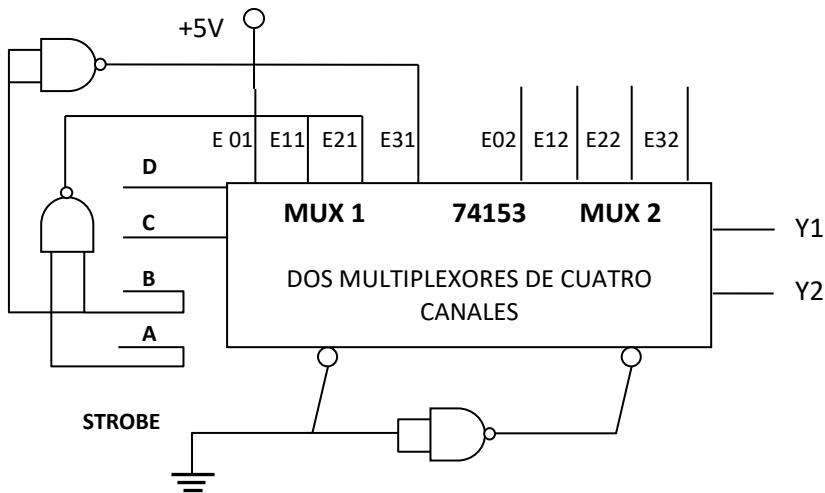
La tabla de verdad de 16 combinaciones se agrupa de 4 en 4, obteniéndose solo cuatro combinaciones, los cuales se logra controlar con dos variables (D y C),

mientras que las variables B y A son variables adicionales al multiplexor. Se observa que las cuatro primeras combinaciones son 1, por tanto, $Y1 = 1$.

Las combinaciones 4, 5, 6 y 7, responde a la función $\overline{B} \cdot \overline{A}$; las últimas cuatro combinaciones responde a \overline{B} .

La entrada E01 del multiplexor 1 (MUX 1) se conecta a fuente; las entradas E11 y E21 se conecta a $\overline{B} \cdot \overline{A}$ y la entrada E32 a \overline{B} .

El multiplexor 2 (MUX 2) se deshabilita con nivel alto, lo cual se logra con un inversor conectado a tierra.



4. El bus de datos de 4 bits de un microprocesador, recibe información de cuatro fuentes de datos de 4 bits cada una. Implemente el circuito de multiplexación utilizando dos circuitos integrados 74153.

Solución:

El Bus de Datos de un microprocesador sólo puede transportar la información presente en un canal de varios existentes.

Para implementar la transferencia de cuatro fuentes de datos de 4 bits cada uno hacia un bus de datos de un microprocesador, se puede realizar mediante la interconexión adecuada de 2 circuitos integrados 74153.

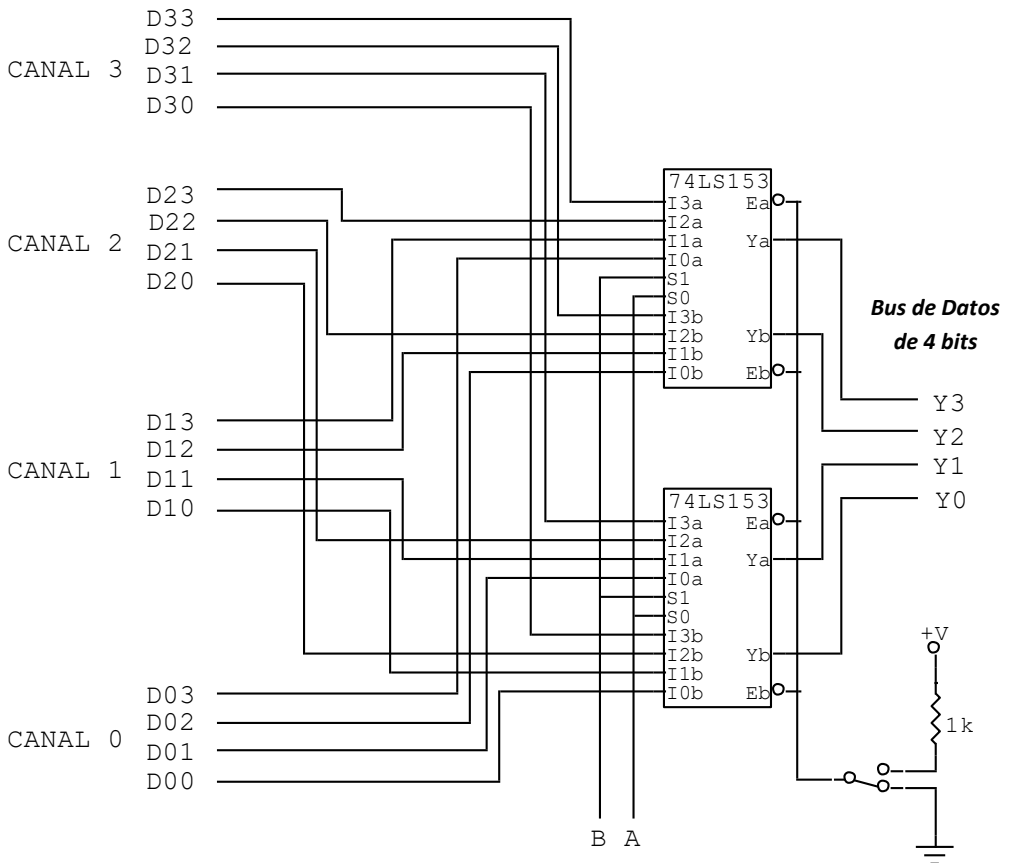
A las entradas (I0, I1, I2 e I3) de los cuatro multiplexores de 4 a 1 se instalan las fuentes de datos de tal manera que las líneas del Canal 0 se conectan a los

cuatro I/O de los multiplexores; las líneas del Canal 1 se conectan a los I1; el canal 2 a los I2, y el canal 3 a los I3.

Las salidas (Ya y Yb) de cada multiplexor constituyen el Bus de Datos de 4 bits (Y3, Y2, Y1, Y0), los cuales reciben la Data de un canal seleccionado por las líneas de selección (S0 y S1), conectados a mos circuitos integrados a las líneas B y A.

La habilitación de los multiplexores se realiza con un pulso negativo en los terminales (Ea y Eb).

Nº	SELECCIÓN		CANAL TRANSFERIDO AL BUS DE DATOS
	B	A	
0	0	0	CANAL 0
1	0	1	CANAL 1
2	1	0	CANAL 2
3	1	1	CANAL 3



Problemas Propuestos N° 07

1. Implemente un circuito lógico de un codificador octal a binario.
2. Utilice el circuito integrado 74153 para implementar la siguiente función lógica $f(A, B, C) = C + \overline{A \oplus B}$
Recomendación: estandarice la función anterior para encontrar minitérminos o maxitérminos.
3. Implemente un multiplexor de 16 a 1, utilizando multiplexores de 4 a 1.
4. Implemente un circuito de multiplexación utilizando circuitos integrados 74151, para acoplar dos canales de datos de ocho bits en un bus de datos de 8 bits.

Decodificadores, demultiplexores y comparadores de magnitud

8.1. Decodificadores

Los decodificadores son circuitos combinatoriales de múltiples líneas de salida; de acuerdo a la activación de una o varias salidas, los decodificadores se clasifican en dos tipos: uno de ellos, cuando se activa una salida, cumple la función de direccionamiento de posiciones de memoria; mientras que cuando se activan varias salidas se trata de un decodificador utilizado para visualizar siete segmentos de LEDs en el denominado “display de siete segmentos”.

8.1.1. Decodificadores con una salida activada

Los decodificadores tienen (n) variables de entrada y un conjunto de líneas de salida (2^n), de las que sólo una se activa.

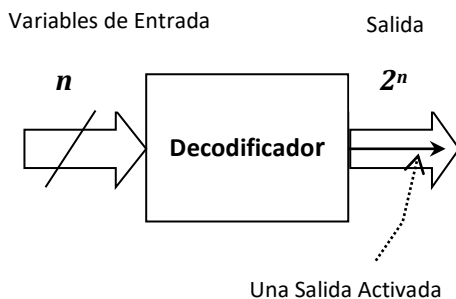


Fig. 8.1

Diagrama de bloques de un decodificador con una salida activada

Si el decodificador tiene 2 líneas de entrada entonces podrá direccionar hasta un máximo de $2^2 = 4$ posiciones de memoria, de las cuales sólo una de ellas estará activa. Según la lógica que se adopte, podrán estar activas con nivel alto si la lógica es POSITIVA o en nivel bajo si es NEGATIVA.

N°	Líneas de entrada		Líneas de Salida			
	B	A	S3	S2	S1	S0
0	0	0	0	0	0	1
1	0	1	0	0	1	0
2	1	0	0	1	0	0
3	1	1	1	0	0	0

Tabla 8.1

Tabla de verdad de un decodificador de 2 a 4 líneas

Hay cuatro salidas (S3, S2, S1 y S0) que se activan uno a la vez; en cada una de ellas se observa que existe un solo minitérmino, por lo que es conveniente aplicar el método de la tabla para obtener sus respectivas expresiones lógicas.

$$S0 = \bar{B} \cdot \bar{A};$$

$$S1 = \bar{B} \cdot A;$$

$$S2 = B \cdot \bar{A};$$

$$S3 = B \cdot A$$

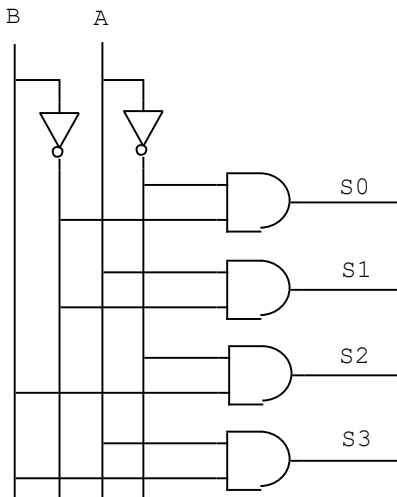


Fig. 8.2

Diagrama de bloques de un decodificador de dos líneas de entrada y cuatro líneas de salida. 1 de 4 líneas.

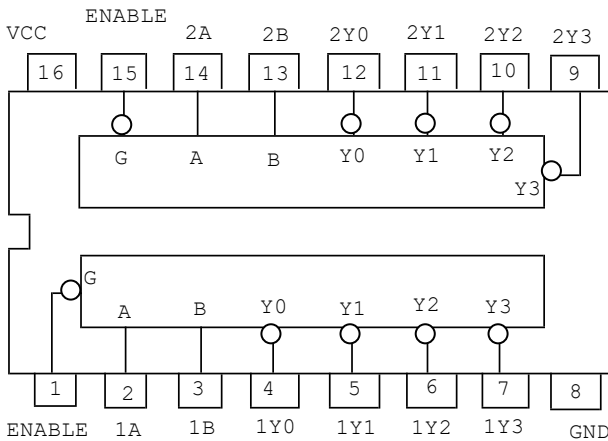


Fig. 8.3

Circuito integrado 74139. Doble decodificador de dos variables de entrada. Los terminales Y0, Y1, Y2 y Y3, son las líneas de salida. El terminal G es el habilitador en nivel BAJO del respectivo decodificador.

8.1.1.1. Diseño de un decodificador de ocho salidas con puertas NAND

Si el decodificador tiene 8 líneas salidas, entonces deberá tener 3 líneas de entrada: $2^3 = 8$. Las tres líneas de entrada se designan mediante las variables (A,B y C), con C como (MSB) el bit más significativo; las ocho líneas de salida se denotan con S, desde S0 hasta S7.

N°	Líneas de Entrada			Líneas de Salida							
	C	B	A	S7	S6	S5	S4	S3	S2	S1	S0
0	0	0	0	1	1	1	1	1	1	1	0
1	0	0	1	1	1	1	1	1	1	0	1
2	0	1	0	1	1	1	1	1	0	1	1
3	0	1	1	1	1	1	1	0	1	1	1
4	1	0	0	1	1	1	0	1	1	1	1
5	1	0	1	1	1	0	1	1	1	1	1
6	1	1	0	1	0	1	1	1	1	1	1
7	1	1	1	0	1	1	1	1	1	1	1

Tabla 8.2

Tabla de verdad de un decodificador 1 de 8 líneas. Salidas activadas en nivel BAJO.

Para obtener la función lógica correspondiente se pueden utilizar minitérminos en mapas de Karnaugh, sin embargo, con el método de la tabla se puede emplear los maxitérminos.

Para obtener las salidas en base a compuertas NAND, es preciso aplicar la propiedad de involución y el teorema De Morgan a las expresiones halladas por el método de la tabla.

$$\begin{aligned}
 S0 &= C + B + A = \overline{\overline{C + B + A}} = \overline{\overline{C} \cdot \overline{B} \cdot \overline{A}} \\
 S1 &= C + B + \overline{A} = \overline{\overline{C + B + \overline{A}}} = \overline{\overline{C} \cdot \overline{B} \cdot A} \\
 S2 &= C + \overline{B} + A = \overline{\overline{C + \overline{B} + A}} = \overline{\overline{C} \cdot B \cdot \overline{A}} \\
 S3 &= C + \overline{B} + \overline{A} = \overline{\overline{C + \overline{B} + \overline{A}}} = \overline{\overline{C} \cdot B \cdot A} \\
 S4 &= \overline{C} + B + A = \overline{\overline{\overline{C} + B + A}} = \overline{C \cdot \overline{B} \cdot \overline{A}} \\
 S5 &= \overline{C} + B + \overline{A} = \overline{\overline{\overline{C} + B + \overline{A}}} = \overline{C \cdot \overline{B} \cdot A} \\
 S6 &= \overline{C} + \overline{B} + A = \overline{\overline{\overline{C} + \overline{B} + A}} = \overline{C \cdot B \cdot \overline{A}} \\
 S7 &= \overline{C} + \overline{B} + \overline{A} = \overline{\overline{\overline{C} + \overline{B} + \overline{A}}} = \overline{C \cdot B \cdot A}
 \end{aligned}$$

El último término obtenido en cada una de las salidas se implementa mediante compuertas NAND. Nótese también que en la implementación de las salidas podrían emplearse compuertas OR, para el mismo propósito, sin embargo, es conveniente hacerlo con compuertas NAND porque obtenemos mejores prestaciones.

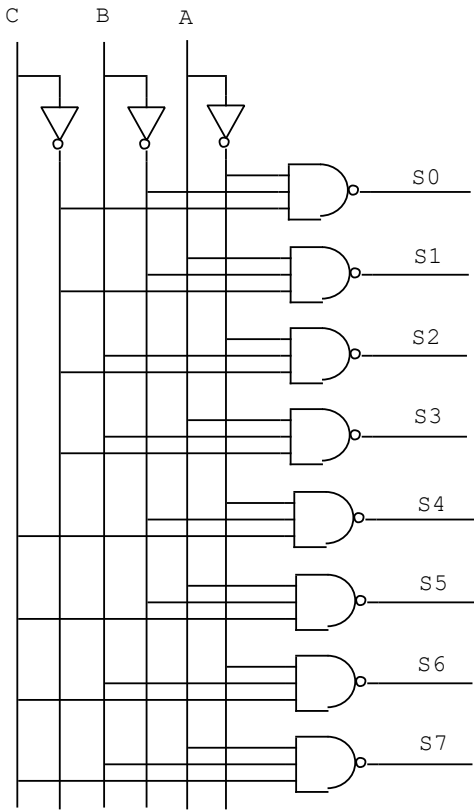


Fig. 8.4

Diagrama lógico de un decodificador 1 de 8 líneas, cuyas salidas se activan en nivel BAJO. Se observa que de cada línea de entrada sale una inversión, de tal manera que si las variables C B A, están en 101, se activará la salida S5. Si en este ejemplo la variable B está en 0, obsérvese que la compuerta NAND de S5, el terminal del centro de dicha compuerta se conecta a la inversión de B.

Si al diagrama lógico de la Fig. 8.4 se le agrega una compuerta AND con tres entradas, cuya salida se aplica a cada compuerta NAND, tendríamos un decodificador 1 de 8 líneas con habilitador. El circuito integrado que incorpora éste esquema es el 74138.

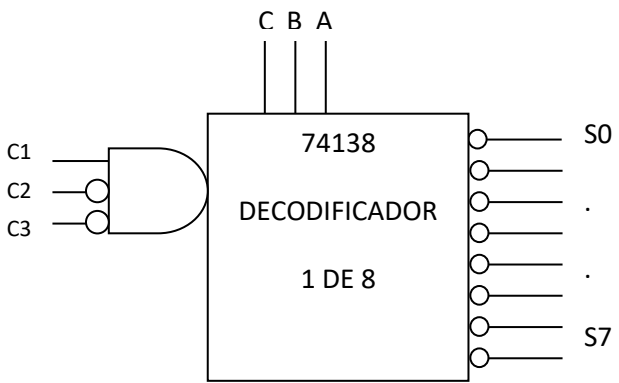


Fig. 8.5

Diagrama de bloques de un decodificador 74138 que activa una salida de ocho disponibles. Cuenta con una compuerta AND como habilitador.

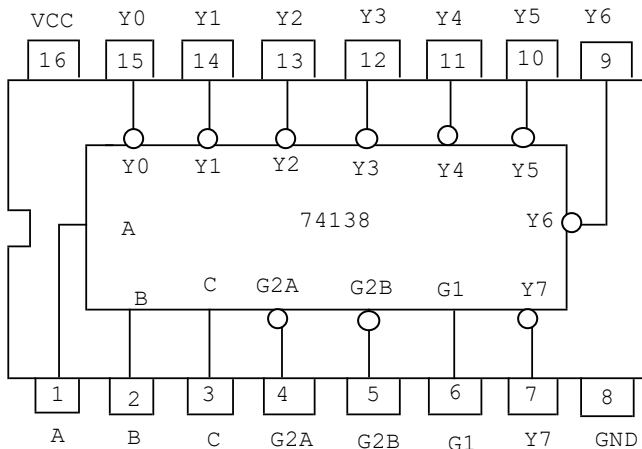


Fig. 8.6

Circuito integrado 74138 decodificador 1 de 8 con habilitador.

La entrada del decodificador está dada por las variables A, B y C; la salida está designada por los terminales Y0 hasta Y7. El habilitador del integrado (C1, C2 y C3) está determinado por los terminales G1, G2B y G2A, respectivamente.

El circuito integrado 74138 se habilita para que responda a la entrada cuando $G1=1$; $G2B=0$; y $G2A=0$; en cualquier otra combinación el dispositivo se bloquea colocando todas las salidas en nivel ALTO.

IMPORTANTE

Los decodificadores del mismo código se pueden instalar en paralelo para obtener un decodificador con mayor número de entradas y como consecuencia de ello, mayor cantidad de líneas de salida.

El circuito integrado 7442 es un decodificador BCD a decimal. El 7443 decodifica un código exceso 3 a decimal. El C.I. 7445, es un decodificador BCD a decimal, idéntico al 7442, con la salvedad que 7445 tiene sus salidas en colector abierto, lo que permite tratar directamente con dispositivos que requieren drenar corriente relativamente altas (80mA).

Los decodificadores que activan sólo una línea de salida, normalmente activan a un conjunto de registros que almacenan un bit. Un conjunto de bits en la línea activada constituye un Bus de Datos.

La generación de funciones se realiza acoplando a una compuerta OR las salidas del decodificador que deseamos estén activas. Es decir, si queremos implementar la función $f(A, B, C) = \sum_m(1,3,5,7)$; las salidas S1, S3, S5 y S7 se conectarán a las entradas de una compuerta OR. Si las salidas del decodificador están complementadas, se puede aplicar la propiedad de involución y el Teorema de Morgan para obtener una compuerta NAND de salida.

8.1.2. Decodificadores con varias salidas activadas

Los decodificadores que activan múltiples salidas se utilizan para casos especiales; una de estas aplicaciones es el manejo de un visualizador numérico de siete segmentos (LEDs).

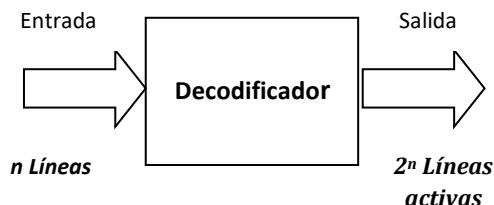


Fig. 8.7

Diagrama de bloques de un decodificador que activa varias líneas de salidas

8.1.2.1. Visualizador numérico de siete segmentos

Todos los circuitos digitales manipulan señales lógicas en el lenguaje binario, ya que la parte física (hardware) funciona con niveles ALTOS y BAJOS de corriente y voltaje.

Para que la información binaria procesada en el hardware se entienda en el mundo exterior, es necesaria la comunicación en sistema decimal y para ello es necesario decodificarlo a través de un dispositivo especial en el que se visualice dígitos decimales.

El visualizador numérico utilizado es aquel que dispone de diodos LEDS adecuadamente organizados.

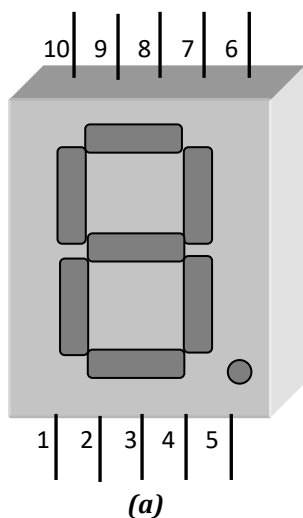
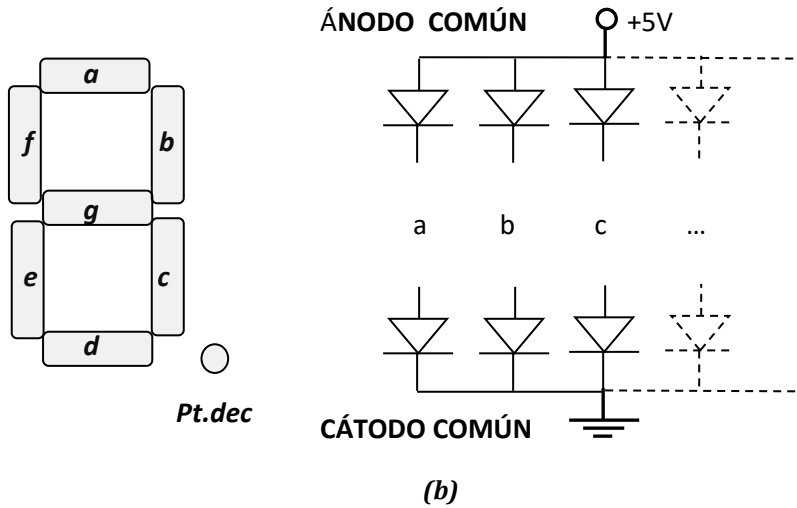


Fig. 8.8

(a) Visualizador numérico de siete segmentos. Dispone de diez pines.

(b) Disposición en el display de los diodos LED como ánodo o cátodo común. Para la polarización directa de los diodos LED, en el display ánodo común, el nivel de voltaje 0 ingresa por el cátodo; mientras que en el de cátodo común el nivel de voltaje 5V ingresa por el ánodo.



Los visualizadores o display están contruidos en base a 7 diodos LED en forma de segmentos (a-g) y un LED adicional para el punto decimal (Pt.dec). Si los ánodos de los LED se hallan conectados entre sí, se denomina display “ánodo común”; si los cátodos están unidos entre si, “cátodo común”.

Nº PIN	LED	DISPLAY ANODO COMÚN	Nº PIN	LED	DISPLAY CÁTODO COMÚN
1	e	Ánodo	1	e	Cátodo
2	d	Ánodo	2	d	Cátodo
3	Ánodo común		3	Cátodo común	
4	c	Ánodo	4	c	Cátodo
5	Punto Decimal		5	Punto Decimal	
6	b	Ánodo	6	b	Cátodo
7	a	Ánodo	7	a	Cátodo
8	Ánodo común		8	Cátodo común	
9	f	Ánodo	9	f	Cátodo
10	g	Ánodo	10	g	Cátodo

Tabla. 8.3

Distribución de segmentos de LED según número de pines de un display. Nótese que la distribución tanto en ánodo común como en cátodo común es la misma; la diferencia está en la polarización.

8.1.2.2. Decodificadores/controladores de BCD a siete segmentos

Es un decodificador de varias salidas activadas con la finalidad de visualizar dígitos decimales a través de la distribución de siete diodos LED en forma de segmentos y un LED adicional para el punto decimal.

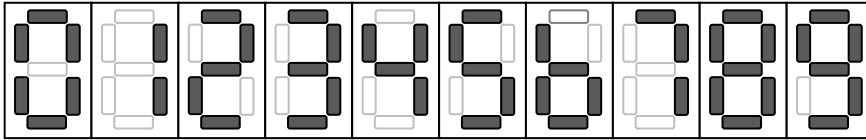


Fig. 8.9

Segmentos del display activados para cada dígito.

El circuito integrado 7447, es un decodificador comercial que controla visualizadores o display ánodo común, esto quiere decir, que el integrado activa a cada diodo LED con un nivel BAJO a través de una resistencia limitadora de corriente.

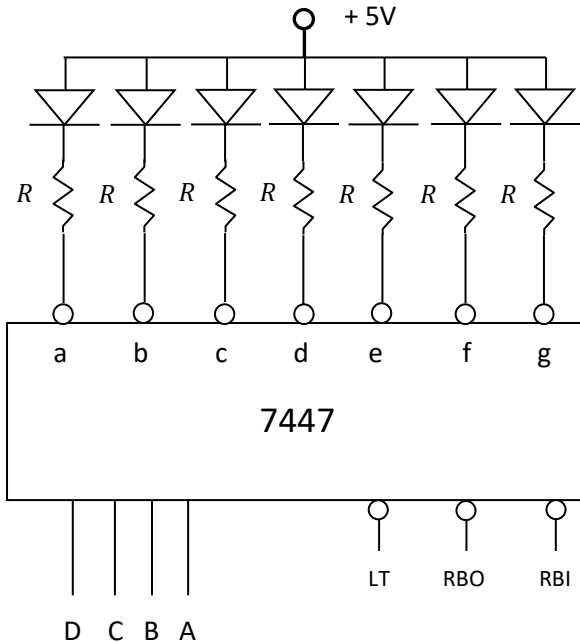


Fig. 8.10

Decodificador 7447 de BCD a siete diodos LED en forma de segmentos, en el que se puede visualizar dígitos decimales. La entrada del decodificador están dadas por las variables (D, C, B, A) con D como MSB. La salida son los terminales (a – g). El control del circuito son los terminales LT, RBO y RBI.

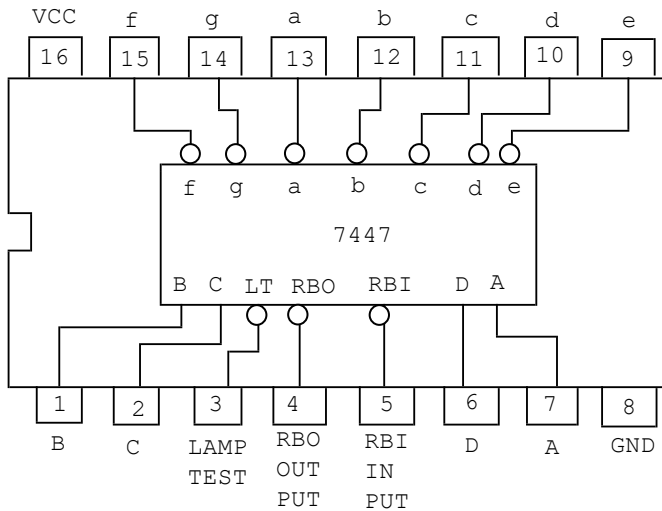


Fig. 8.11

Circuito Integrado 7447, decodificador de BCD a decimal, con salidas en colector abierto. Activa a display ánodo común.

LAMP TEST: terminal de control para verificar el estado del display. Con un nivel bajo se encienden todos los segmentos, independientemente de la entrada.

RBO: (Ripple Blanking Output)

RBI: (Ripple Blanking Input).

Ambos terminales RBO y RBI, se enlazan con sus respectivos decodificadores y actúan para eliminar los ceros no significativos.

IMPORTANTE

La resistencia limitadora de corriente de cada uno de los segmentos LED es de aproximadamente 220 Ω. Dicho valor se calcula en base a la diferencia de voltaje entre (Vcc) y los voltajes que caen en el diodo (Vd) y en el Colector Emisor del Transistor en Colector abierto (Vce); dividido entre la corriente que debe circular por el diodo, aproximadamente 10mA.

$$R = \frac{(V_{cc} - V_d - V_{CE})}{10mA} = \frac{(5 - 2,3 - 0,35)V}{10mA} = 235\Omega \approx 220\Omega$$

220Ω es un valor típico para una resistencia limitadora de corriente de un diodo LED. Cabe resaltar que se debe instalar un resistor para cada uno de los segmentos del display tal como se aprecia en la Fig. 8.10.

El Circuito Integrado 7448 es un decodificador BCD a siete segmentos de display cátodo común. Además, comercialmente se pueden adquirir decodificadores que pueden almacenar una combinación y visualizarse un dígito aunque la combinación que la generó ya no esté presente en la entrada, como el 74HC4511.

8.2. Demultiplexores

Los demultiplexores (DEMUX) son circuitos combinaciones cuya función es distribuir una señal de entrada entre varias líneas de datos que existe en la salida. La selección de la línea activa de la salida, se realiza a través de unos terminales llamadas "líneas de control".

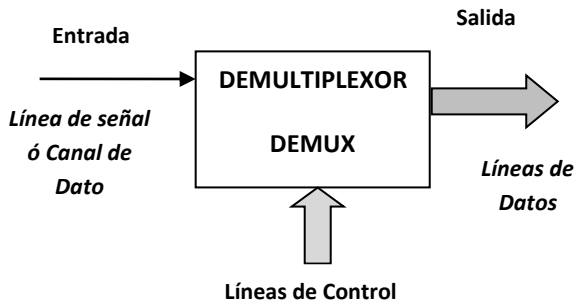


Fig. 8.12

Diagrama de bloques de un demultiplexor.

La relación que existe entre las líneas de datos de salida y las líneas de control es el siguiente:

$$\# \text{ de Canales de Datos de salida} \quad \swarrow \quad \searrow \quad \# \text{ de líneas de Control}$$

$$S = 2^C$$

Si a un decodificador se le agrega una línea de entrada se convierte en un demultiplexor. Si se desea demultiplexar una señal de entrada entre cuatro canales de salida; al circuito decodificador de 1 de 4 (Fig. 8.2) se le agrega una señal de entrada.

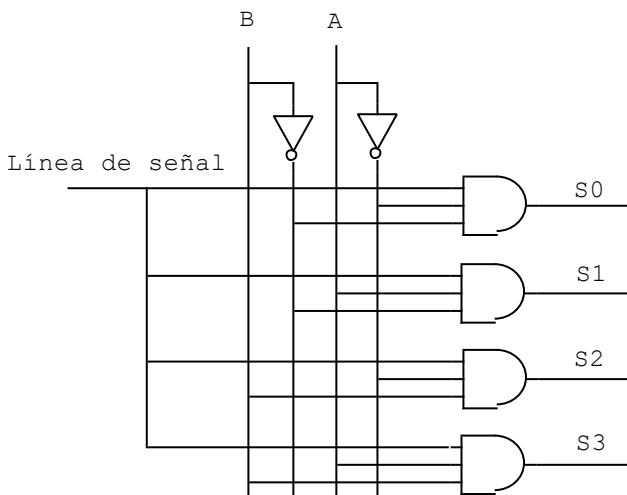


Fig. 8.13

Demultiplexor de 1 a 4. Nótese que la línea de señal se aplica a todas las compuertas; son las líneas de control (A,B) los que determinan que compuerta se activa.

IMPORTANTE

El demultiplexor es una aplicación del decodificador de una salida activada.

Los demultiplexores se aplican en sistemas de transmisión de información, en el cual unen dos bloques conformados por un multiplexor y por un demultiplexor. A éste enlace de datos se llama, sistema de transmisión multiplexada.

Concretamente, los demultiplexores no se fabrican para cumplir sólo ésta función; es por ello que comercialmente lo encontramos acoplado con el decodificador.

8.3. Comparador de magnitud

El comparador de magnitud, es un circuito combinacional que compara los valores de dos canales de datos en la entrada y determina en tres líneas de salida si dichos valores son iguales, y si no lo son, cual es el mayor y cual el menor; es decir, si los canales de datos de entrada (*cada canal puede tener varias líneas de señal*) son A y B, compara los valores que adoptan dichas líneas de datos y determina si $(A = B)$; $(A > B)$ ó $(A < B)$ y lo muestra en la salida.



Fig. 8.14

Diagrama de bloques de un comparador de magnitud.

8.3.1. Diseño de un comparador de 1 bit de dato en cada canal de entrada

Los buses de entrada proveen al circuito con 1 bit de dato. Se construye una tabla de verdad con dos variables de 1 bit y tres líneas de salida.

N°	VARIABLES DE ENTRADA		LÍNEAS DE SALIDA		
	A	B	E	G	L
			A=B	A>B	A<B
0	0	0	1	0	0
1	0	1	0	0	1
2	1	0	0	1	0
3	1	1	1	0	0

Tabla 8.4

Tabla de verdad de un comparador de magnitud de un bit.

De la Tabla 8.4 se deduce las tres líneas de salida:

$$E = \bar{A}.\bar{B} + AB$$

$$G = A.\bar{B}$$

$$L = \bar{A}.B$$

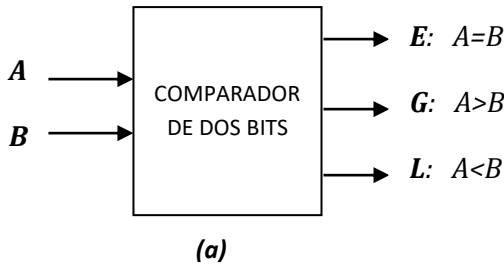
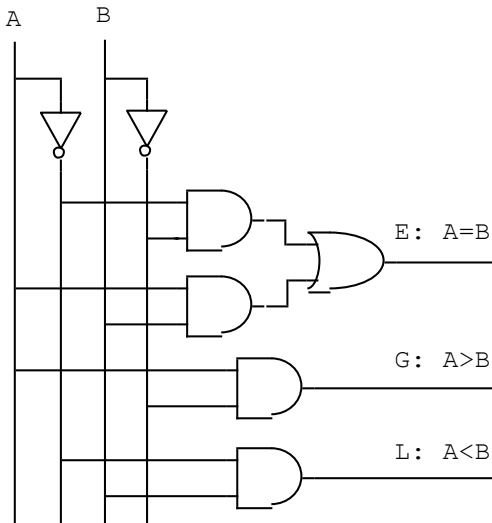


Fig. 8.15

(a) Diagrama de bloques de un comparador de dos bus de datos de un bit cada uno, cuyas salidas son: **E** (Equal); **G** (Greater); **L** (Lesser); que corresponden a $A=B$; $A>B$; y $A<B$, respectivamente.



(b) Diagrama lógico de un comparador de dos canales de un bit cada uno.

8.3.2. Diseño de un comparador de tres bits de datos en cada canal de entrada

El circuito lógico presenta dos buses de datos en la entrada con tres bits cada uno. La salida presenta tres líneas de un bit cada uno.

Designaremos al bus de datos A como: $A_2 A_1 A_0$

Designaremos al bus de datos B como: $B_2 B_1 B_0$

Designaremos a las líneas de salida como: $A = B$; $A > B$; $A < B$

Se construye una tabla de verdad con tres columnas de entrada. En cada columna se compara los bits del mismo **orden** en ambos números de entrada; tal como: ($A_2 B_2$); ($A_1 B_1$); ($A_0 B_0$).

Si en los bits más significativos ($A_2 B_2$) o de mayor orden o peso, uno de ellos es mayor o menor, ésta condición determina la salida, es decir, si ($A_2 > B_2$) ó G_2 entonces, en la columna $A > B$ de la salida se le asignará un 1 ó nivel ALTO; la comparación de los bits menos significativos en ésta misma fila, es indiferente y se marca con X. Utilizando el mismo razonamiento completamos la tabla teniendo en cuenta que si los bits más significativos son iguales, los bits del orden inmediato inferior determinará la condición de salida.

DATOS DE ENTRADA			LÍNEAS DE SALIDA		
$A_2 B_2$	$A_1 B_1$	$A_0 B_0$	$A=B$	$A>B$	$A<B$
$A_2 > B_2$	X	X	0	1	0
$A_2 < B_2$	X	X	0	0	1
$A_2 = B_2$	$A_1 > B_1$	X	0	1	0
$A_2 = B_2$	$A_1 < B_1$	X	0	0	1
$A_2 = B_2$	$A_1 = B_1$	$A_0 > B_0$	0	1	0
$A_2 = B_2$	$A_1 = B_1$	$A_0 < B_0$	0	0	1
$A_2 = B_2$	$A_1 = B_1$	$A_0 = B_0$	1	0	0

Tabla. 8.5

Tabla comparativa de tres bits de entrada de dos buses de datos.

Analizando los minitérminos de las líneas de salida tenemos:

La función igual $E (A=B)$ se determina mediante:

$$E = E_2.E_1.E_0$$

La función mayor $G (A>B)$ se determina de la siguiente manera:

$$G = G_2 + E_2.G_1 + E_2.E_1.G_0$$

La función menor $L (A<B)$ se halla así:

$$L = L_2 + E_2.L_1 + E_2.E_1.L_0$$

Estas tres salidas se pueden construir en base al comparador de un bit de la Fig. 8.15.

Problemas Resueltos N° 08

1. Implemente un decodificador de 5 a 32 utilizando cuatro circuitos integrados 74138.

Solución:

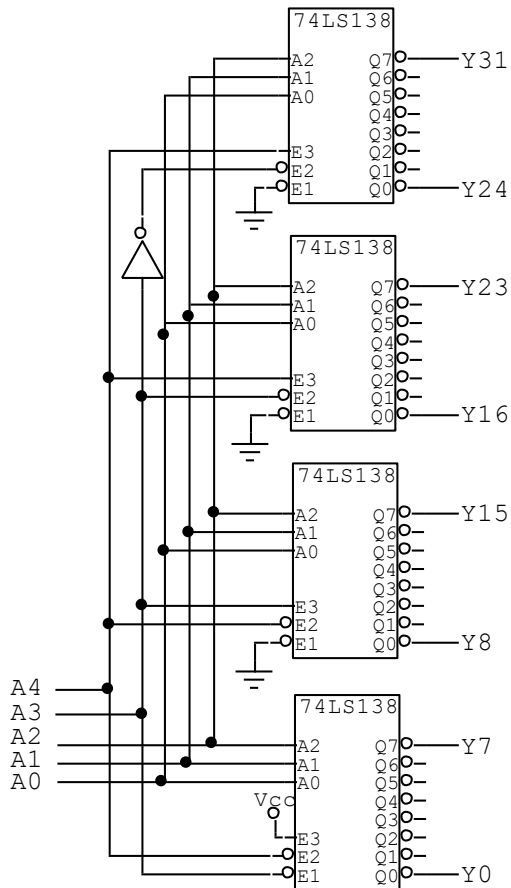
El decodificador 74138 dispone de tres variables ó líneas de selección (A2, A1 y A0) y tres líneas de habilitación (E3, E2 y E1); lo cual coincide con la fig. 8.6, donde las líneas de selección son (C, B y A) y las líneas de habilitación son (G2A, G2B y G1). La salida se numera desde Y0 hasta Y7.

Para implementar un decodificador de 32 líneas de salida (Y0 a Y31) se requiere 5 líneas de selección. Las dos líneas de selección adicionales se obtendrán de las líneas de habilitación.

Como cada decodificador tiene 8 salidas, se requieren cuatro circuitos integrados para obtener las 32.

Para obtener las 32 combinaciones se requieren 5 variables. Las dos variables adicionales (A4 y A3) con A4 como MSB, se obtendrán cableando convenientemente los terminales (E1, E2 y E3).

Evaluando los valores que adoptan las variables (A4 y A3) de una tabla de verdad con 5 variables, se observa lo siguiente:



- Las primeras ocho combinaciones para A4 y A3 son (0 0), por tanto, el primer decodificador se habilita ingresando A4 en E2, A3 en A1 y E3 a fuente.
- La segunda ocho combinaciones para A4 y A3 son (0 1). Para habilitar el segundo decodificador se realiza conectando A4 en E2, A3 en E3 y E1 a tierra.

- La tercera ocho combinaciones para A4 y A3 son (1 0). Para habilitar el tercer decodificador se conecta A4 con E3, A3 con E2 y E1 a tierra.
 - La cuarta ocho combinaciones para A4 y A3 son (1 1). Para habilitar el cuarto decodificador se conecta A4 con E3, $\overline{A3}$ con E2, y E1 a tierra.
2. Cuatro dispositivos distintos transmiten datos de ocho bits hacia un dispositivo semiconductor de memoria. Implemente con un decodificador un circuito habilitador para que solo un dispositivo ingrese su Data en un momento dado. La línea de habilitación de cada dispositivo se realiza con nivel bajo.

Solución:

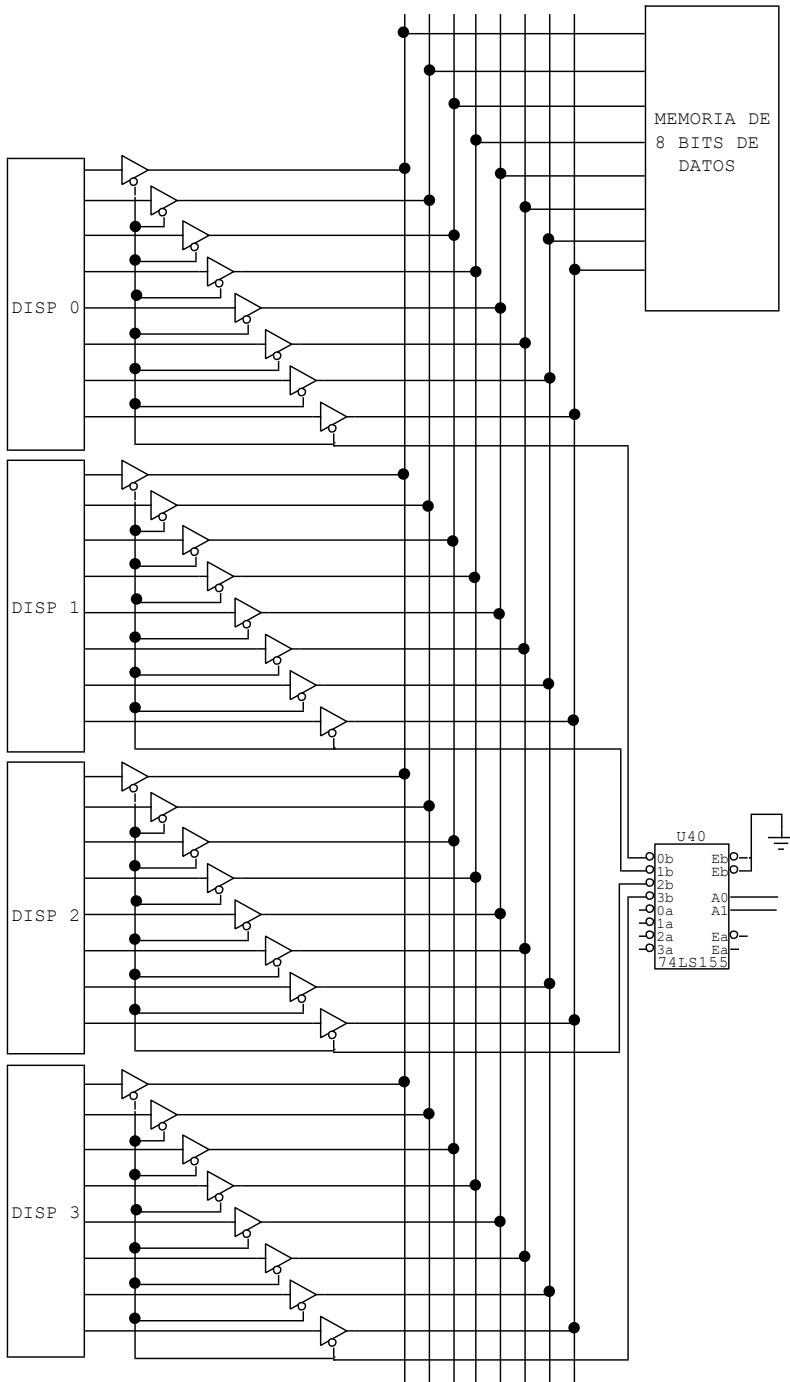
Un bus de datos puede recibir Data de muchos dispositivos (DISP) conectados a él, sin embargo, el ingreso de dichos datos debe realizarse un bus a la vez. Para acoplar las líneas de datos de los diferentes dispositivos (bus de datos de los dispositivos) se utiliza una compuerta buffer triestado, con habilitador en nivel bajo.

Los cuatro dispositivos (DISP 0, DISP 1, DISP 2, DISP 3) acoplan al Bus de Datos, su respectivo bus por medio de un conjunto de buffers. Los habilitadores de cada uno de los buffers se enlazan a las salidas de un decodificador de 2 variables de entrada (seleccionadores) y 4 salidas.

El circuito integrado 74155 dispone de dos decodificadores/demultiplexores de 2 líneas de entrada (A0, A1) ó líneas de selección y 4 líneas de salida (0b, 1b, 2b, 3b) para el decodificador B, el cual se habilita mediante las líneas STROBE y Data (Eb1 y Eb2). El decodificador B se habilita conectando STROBE y Data a un nivel Bajo.

Las salidas activas del decodificador están en nivel Bajo, mientras que las salidas inactivas se encuentran en nivel Alto. El nivel Bajo de la salida del decodificador habilita los buffers del dispositivo que inmediatamente ingresa su Data al Bus de Datos, deshabilitando a los restantes tres dispositivos.

Cuando las líneas de selección (A1 A0) está en (0 0), el dispositivo 0 (DISP 0) ingresa su Data al Bus de Datos; cuando está en (0 1), se habilita el (DISP 1); cuando (1 0), se habilita (DISP 2); si (1 1), se habilita (DISP 3).



3. Implemente la función $f(P, Q, R, S) = \prod_M(1, 5, 6, 11, 15)$, utilice el circuito integrado 74154.

Solución:

El circuito integrado 74154 dispone de 4 variables de entrada (líneas de selección) con los cuales puede direccionar 16 posiciones (S0 – S15) cuyas salidas están invertidas y solo uno puede estar activado.

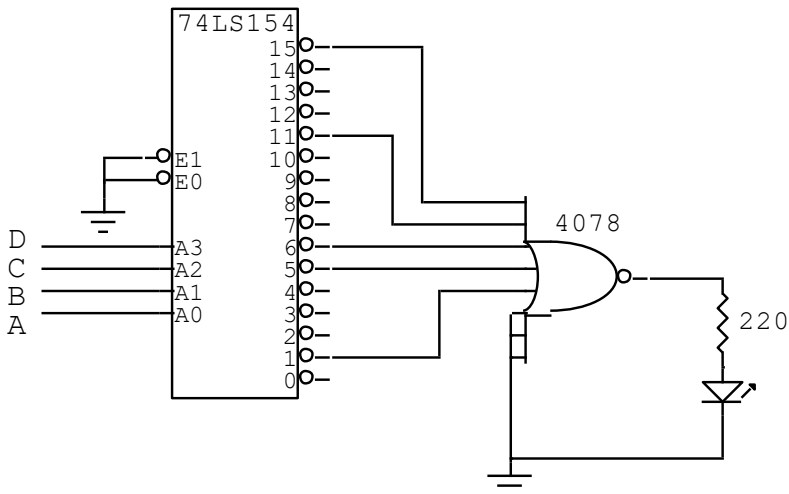
Para implementar funciones lógicas con decodificadores, cada minitérmino se suma lógicamente, mientras que cada maxitérmino se multiplica.

Como las salidas de los decodificadores están complementados, el producto de maxitérminos puede convertirse a suma mediante la propiedad de involución y el Teorema de Morgan.

$$f(D, C, B, A) = \overline{S1} \cdot \overline{S5} \cdot \overline{S6} \cdot \overline{S11} \cdot \overline{S15}$$

$$f(D, C, B, A) = \overline{\overline{\overline{S1} \cdot \overline{S5} \cdot \overline{S6} \cdot \overline{S11} \cdot \overline{S15}}} = \overline{S1 + S5 + S6 + S11 + S15}$$

El circuito integrado CMOS 4078 contiene una compuerta NOR de 8 entradas. Dicha compuerta se conecta a las 5 salidas del decodificador que corresponden a los maxitérminos. Los tres terminales no utilizadas de la compuerta se conectan a tierra. A la salida de la compuerta NOR se conecta un diodo LED para visualizar el nivel Alto de la función (LED iluminado) y el nivel Bajo con el LED apagado.



4. Diseñe un circuito detector de **paridad par** de 3 bits de datos; visualizándose en un display cátodo común la letra "b" si no hay error en la transferencia, y "E" si lo hay. Asimismo, si la transferencia fue correcta determine el bit de paridad.

Solución:

La transferencia de la Data se realiza por un bus de 4 bits, donde el canal de datos es de 3 bits (A, B, C), y el bit de paridad de 1 bit (D). Para diseñar el circuito detector de paridad debemos tener en cuenta que el bit más significativo (MSB) es el bit de paridad.

Tabla de verdad del canal de 3 bits y del bit de PARIDAD PAR.

BIT DE PARIDAD PAR	CANAL DE 3 BITS		
	D	C	A
0	0	0	0
1	0	0	1
1	0	1	0
0	0	1	1
1	1	0	0
0	1	0	1
0	1	1	0
1	1	1	1

El detector de error se diseña en una tabla de verdad de 4 bits, considerando el nivel Bajo (lógica negativa) para las combinaciones de la tabla anterior, por ejemplo, la combinación 1100, adopta el valor 0 en la fila 12.

Nº	DATA				DETECTOR	Nº	DATA				DETECTOR
	D	C	B	A			D	C	B	A	
0	0	0	0	0	0	8	1	0	0	0	1
1	0	0	0	1	1	9	1	0	0	1	0
2	0	0	1	0	1	10	1	0	1	0	0
3	0	0	1	1	0	11	1	0	1	1	1
4	0	1	0	0	1	12	1	1	0	0	0
5	0	1	0	1	0	13	1	1	0	1	1
6	0	1	1	0	0	14	1	1	1	0	1
7	0	1	1	1	1	15	1	1	1	1	0

De la tabla se obtienen la siguiente función lógica.

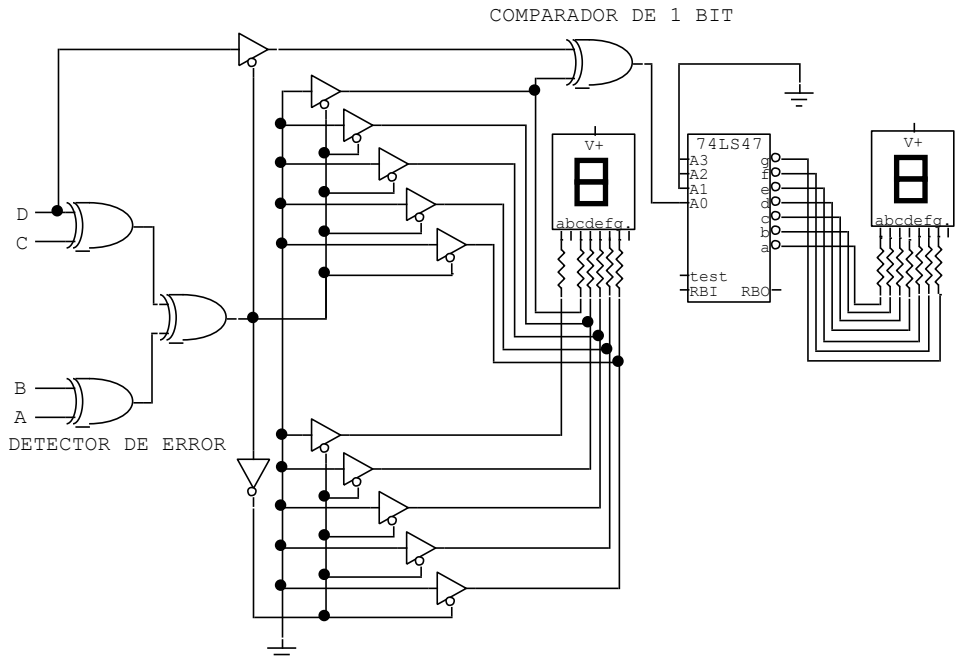
$$f(D, C, B, A) = \bar{B}\bar{A}(D\bar{C} + \bar{D}C) + B\bar{A}(\bar{D}\bar{C} + DC) + \bar{B}A(\bar{D}\bar{C} + DC) + BA(D\bar{C} + \bar{D}C)$$

$$f(D, C, B, A) = \bar{B}\bar{A}(D\oplus C) + B\bar{A}(\overline{D\oplus C}) + \bar{B}A(\overline{D\oplus C}) + BA(D\oplus C)$$

$$f(D, C, B, A) = (D \oplus C)(\bar{B}\bar{A} + BA) + (\overline{D \oplus C})(B\bar{A} + \bar{B}A)$$

$$f(D, C, B, A) = (D \oplus C)(\overline{B \oplus A}) + (\overline{D \oplus C})(B \oplus A)$$

$$f(D, C, B, A) = (D \oplus C) \oplus (B \oplus A)$$



Si la Salida del detector de error es 0, habilita a un conjunto de buffers triestado cuyas entradas se encuentran conectadas a tierra y sus salidas a las entradas (c, d, e, f, g) del display ánodo común, visualizándose la letra "b" e indicando que la transferencia de la data se realizó correctamente. Sin embargo, si la salida es 1, indica que la operación de transferencia se realizó con error, el cual se visualiza en el display la letra "E" mediante la habilitación de las entradas (a, d, e, f, g) a través de buffers conectados a tierra.

La línea D, que transporta el bit de paridad, ingresa a un buffer triestado que se habilita con la salida del detector de error e ingresa al comparador de 1 bit. La otra entrada de este comparador se conecta a tierra a través del buffers que ingresa al terminal (c) del display.

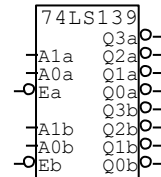
La salida del comparador ingresa al terminal del bit menos significativo del decodificador (A0), mientras que las otras entradas se conectan a tierra para que se muestre en el display un **0** si $A0=0$, y un **1** si $A0=1$.

Problemas Propuestos N° 08

1. Implemente un sistema decodificador de seis variables de entrada. Utilice 4 circuitos integrados 74154 y un C.I. 74139.

Sugerencia:

Las variables de selección (A3, A2, A1, A0), se instala en paralelo a los cuatro C.I. 74154 (decodificador 4 a 16); además, el terminal de habilitación (E0) de cada decodificador se instala a tierra. El otro terminal (E1), de cada integrado, se enlaza con cada una de las cuatro salidas del decodificador 74139 (decodificador 2 a 4).



Las variables de selección de mayor peso (A5 y A4), ingresan a los terminales de entrada de uno de los dos decodificadores del C.I 74139.

2. Dada la siguiente función $f(D, C, B, A) = \sum_m(0, 1, 4, 5, 7, 8, 10, 12, 15)$; implemente con un Circuito Integrado 74138 (Demultiplexor (DMUX) de tres entradas de control solamente (C, B, A).

Sugerencia:

Como las salidas de DMUX están complementadas, se conecta un inversor a cada una.

Nº	VA				FUNCIÓN	Nº	VA				FUNCIÓN
	D	C	B	A			D	C	B	A	
0	0	0	0	0	1	8	1	0	0	0	1
1	0	0	0	1	1	9	1	0	0	1	0
2	0	0	1	0	0	10	1	0	1	0	1
3	0	0	1	1	0	11	1	0	1	1	0
4	0	1	0	0	1	12	1	1	0	0	1
5	0	1	0	1	1	13	1	1	0	1	0
6	0	1	1	0	0	14	1	1	1	0	0
7	0	1	1	1	1	15	1	1	1	1	1

Evaluando la tabla de verdad sólo con tres variables (C, B, A) en el tramo (0-7), los miniterminos son: 0, 1, 4, 5, y 7; en el tramo (8-15), son: 0, 2, 4, y 7.

El terminal (D) si está en 0, habilitará las salidas del tramo (0-7); mientras que si D = 1, habilitará las salidas del tramo (8-15)

Las salidas del C.I. 74138 que pasarán a la compuerta OR son: 0, 1, 2, 4, 5, y 7. Previo a la compuerta OR, cada salida del inversor (de la salida del C.I.) se conecta a una de las dos entradas de una compuerta AND. El otro terminal de las seis compuertas AND se unen mutuamente hacia la variable D, el cual se utiliza para habilitar las compuertas por tramos.

3. Implemente un circuito comparador de 8 bits de datos, utilice dos circuitos integrados 7485 (comparador de 4 bits).

Nota:

El 7485 dispone de ocho terminales de entrada: cuatro para el número A de 4 bits (A3, A2, A1, A0) y cuatro para el número B (B3, B2, B1, B0); tres salidas para determinar si los números comparados son mayor, menor o igual (A<B; A=B; y A>B). Dispone además de tres entradas adicionales (IA<B; IA=B; y IA>B), I=**input**; utilizados para ampliar su capacidad de comparación.

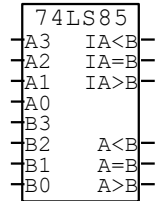


Tabla de verdad del funcionamiento de los terminales de entrada adicionales.

DATOS	ENTRADAS ADICIONALES			SALIDA DE COMPARACIÓN		
	IA<B	IA=B	IA>B	A<B	A=B	A>B
A=B	0	0	1	0	0	1
A=B	0	1	0	0	1	0
A=B	1	0	0	1	0	0
A>B	X	X	X	0	0	1
A<B	X	X	X	1	0	0

Uno de los circuitos integrados 7485, deber realizar la comparación de los cuatro bits menos significativos, mientras que el otro integrado, la comparación de los cuatro bits más significativos.

4. Diseñe un circuito digital, en el que se ingrese dos números decimales de un dígito (0 – 9) cada uno. Mediante un circuito comparador determinar el número mayor el cual pueda ser visualizado en un display cátodo común.

Sugerencia:

Utilice el C.I. 74147 (codificador decimal a BCD), un C.I. 7485 (comparador de 4 bits), el circuito digital del Ejercicio Resuelto N° 4 del capítulo 07 (utilice sólo el canal 0 y canal 1), además conectar el terminal (B) a tierra.

Capítulo 9

Circuitos aritméticos

9.1. Introducción

Los sistemas digitales que cuentan con microprocesador, disponen de una Unidad Aritmético Lógica (ALU) para efectuar operaciones del tipo aritmético como (suma, resta, multiplicación, etc) y del tipo lógico como (OR, AND, OR-EX, etc), sin embargo, muchos circuitos digitales sencillos de aplicación doméstica y comercial, realizan operaciones aritméticas sencillas antes de responder a un determinado estímulo.

Las operaciones aritméticas que realizamos las personas, normalmente lo hacemos en el sistema decimal, ya sea en nuestras operaciones cotidianas o al momento de comunicarnos con un sistema de cómputo. Como ya es de nuestro conocimiento, las computadoras sólo pueden realizar todas sus operaciones en base a niveles de voltaje, que traducidos a 1s y 0s obtenemos una relación con el sistema binario.

En el sistema binario, cabe resaltar un detalle muy importante en cuanto al valor que adopta cada una de las variables. Para efectuar una operación aritmética es necesario que los valores de las variables sean cantidades numéricas binarias, es decir, que el 1 represente la cantidad de una unidad, mientras que el 0, represente la ausencia de algo; sin embargo, para efectuar una operación lógica, los valores que adopten las variables deben cumplir con las características de una variable lógica, aspecto que hemos tratado en la **sección 3.2**

El sistema binario, por tener su base 2, está muy relacionado con el sistema y octal y hexadecimal. Ambos sistemas son susceptibles de operar aritméticamente, conforme a las reglas que también rigen al sistema decimal.

En éste capítulo estudiaremos todo lo relacionado con las operaciones aritméticas como una ampliación del capítulo 3, asimismo, diseñaremos elementales bloques aritméticos y analizaremos los principales circuitos “sumadores” que pueden realizar todo tipo de operaciones aritméticas.

9.2. Operaciones aritméticas

La suma, resta, multiplicación y división son las cuatro operaciones básicas con las cuales podemos construir modelos matemáticos mucho más complejos. Sin embargo, como veremos luego, con una sola operación es posible lograr todo lo anterior.

Para analizar el aspecto procedimental de las operaciones realizaremos las operaciones básicas de suma, resta y multiplicación en el sistema binario, octal y hexadecimal.

9.2.1. Suma

En la realización de una suma binaria se procede como en el sistema decimal, teniendo en cuenta que la base es 2 y por lo tanto dispone de solo dos dígitos para representar y operar cantidades. Cuando se efectúe una operación aritmética con números binarios se deberá tener en cuenta que 10 no significa diez unidades, sino que se trata del **dos BINARIO**, lo mismo ocurre con 11, se trata del tres binario, etc.

Si realizamos la suma de dos variables de 1 bit cada uno tendremos:

N°	VARIABLES DE ENTRADA		SALIDA	
	B	A	C (Carga)	S (Suma)
0	0	0	0	0
1	0	1	0	1
2	1	0	0	1
3	1	1	1	0

Tabla 9.1

Tabla de verdad de la suma de dos variables de un bit cada uno.

Entonces, si sumamos en binario:

$$1 + 1 + 1 = \overset{C}{1} \overset{S}{1}$$

$$1 + 1 + 1 + 1 = \overset{C}{1} \overset{S}{0} \overset{S}{0}$$

El cargo C = 10, no representa diez unidades, sino, dos unidades; si el cargo es igual a 11, indica que se está llevando tres unidades.

EJEMPLO DE SUMA BINARIA.

Sumar: $10111,1011_2 + 111011,101_2 + 101101,11_2$

SUMA BINARIA	SUMA DECIMAL
10111,1011 +	23,6875 +
111011,1010	59,6250
<u>101101,1100</u>	<u>45,7500</u>
10000001,0001	129,0625

EJEMPLO DE SUMA OCTAL.

Sumar: $7532,6_8 + 3521,7_8 + 406,3_8$

7532,64
 3521,7
406,32
 13663,06

N° Dec	N° OCTAL	
	8	1
16	2	0
11	1	3

Téngase en cuenta que en el sistema octal se puede trabajar con el cuadro anterior para formar números mayores que ocho.

En el caso de la segunda columna empezando por la derecha tenemos: $6 + 7 + 3 = 16$ decimal; pero en octal es 20, entonces, el dígito suma es 0, y el dígito cargo es 2 que se lleva al siguiente orden y el procedimiento continúa hasta concluir la suma.

EJEMPLO DE SUMA HEXADECIMAL.

Sumar: $3E9A,3_H + 82F,B_H + C1D,04_H$

3E9A,3
 82F,B
C1D,04
 52E6,E4

N° Dec	N° HEXADECIMAL	
	16	1
38	2	6
34	2	2

9.2.2. Resta

Para los siguientes ejemplos de resta, el minuendo será un número mayor que el sustraendo.

EJEMPLO DE RESTA BINARIA

Restar: $11000_2 - 1011_2$

11000 –
1011
 1101

EJEMPLO DE RESTA OCTAL

Restar: $67231_8 - 54326_8$

$67231 -$
 54326
 12703

N° Dec	N° OCTAL	
	8	1
9	1	1
10	1	2

EJEMPLO DE RESTA HEXADECIMAL

Restar: $5F82_H - 2EAB_H$

$5F82 -$
 $2EAB$
 $30D7$

N° Dec	N° HEXADECIMAL	
	16	1
18	1	2
23	1	7

9.2.3. Multiplicación

EJEMPLO DE MULTIPLICACIÓN BINARIA

Multiplicar: $1011_2 \times 101_2$

1011×101
 1011
 1011
 110111

EJEMPLO DE MULTIPLICACIÓN OCTAL

multiplicar: $473_8 \times 52_8$

473×52
 1166
 3047
 31656

N° Dec	N° HEXADECIMAL	
	8	1
14	1	6
15	1	7
36	4	4
24	3	0

EJEMPLO DE MULTIPLICACIÓN HEXADECIMAL

Multiplicar: $6DA4_H \times 8_H$

$6DA4 \times 8$
 $36D20$

N° Dec	N° HEXADECIMAL	
	16	1
32	2	0
82	5	2

IMPORTANTE

Al realizar el procedimiento de las operaciones aritméticas hemos empleado los mismos algoritmos que los utilizados en el sistema decimal, teniendo en cuenta por ejemplo que **en el sistema octal $5 \times 3 = 17$** y no el decimal 15.

Es necesario aclarar que dichos algoritmos empleados para encontrar las operaciones básicas, son utilizados especialmente por las personas, lo cual se convertido en un proceso familiar, sin embargo, en el interior de una computadora procesa la información algo distinto; es decir, para desarrollar una multiplicación, la ALU efectúa la operación suma con el multiplicando tantas veces como lo indique el multiplicador. La ventaja de la ALU es su velocidad, pues, puede procesar información tan rápidamente que una suma de varios términos lo ejecuta en una fracción de segundo.

9.3. Representación de números con signo

En la sección 2.2 se representaron los números binarios en formato signo magnitud, con el cual se satisfacía la necesidad de otorgarle un signo positivo o negativo mediante la inclusión de un bit 0 ó 1 a la magnitud de número.

No obstante la importancia de asignarle un signo a un número binario, éste procedimiento comienza a tener dificultades al momento de operar aritméticamente números con signo, especialmente cuando se trata de números con signos opuestos.

Cuando dos números binarios tienen igual signo, a la suma total se le asigna el signo de los sumandos, sin embargo, cuando los sumandos tienen diferente signo, a la suma total se le afecta el signo del número mayor. En éste caso, cuando se suman dos números con diferente signo, en realidad, aritméticamente se restan las magnitudes y a la diferencia se afecta con el signo del número mayor.

El formato signo magnitud es uno de los métodos para representar números con signo, ya que existen otros sistemas como el complemento a 1 y el complemento a 2, que permiten efectuar operaciones aritméticas con números con signos diferentes y lograr saltar el impase anterior.

El sistema de complemento a 1 consiste en establecer un número positivo asignándole un 0 en el bit de signo, luego, invertir o complementar dicho número incluyendo tanto la magnitud como el signo.

Dado la magnitud del siguiente número binario: 11001, aplicar el método de complemento a 1.

011001, número binario que corresponde a +25

Aplicando el complemento a 1 tenemos:

100110, número binario que corresponde a -25

Pero, el método preferido para representar números con signo es el sistema de complemento a 2.

9.4. Sistema de complemento a 2

Es aquel formato utilizado para representar números con signo y con estos números realizar las operaciones de suma y resta; resta que también se efectúa mediante la operación suma, es decir, la resta se realiza sumando. El procedimiento anterior es considerado por la ALU como una operación muy versátil, pues, con un solo sumador se puede realizar tanto la suma como la resta, y como consecuencia de ello, todas las demás operaciones aritméticas.

El procedimiento para aplicar el complemento a 2 a un número binario es el siguiente:

- A la magnitud de un número binario se le agrega el bit 0 a la izquierda del Bit más significativo, para indicar que se trata de un número positivo.
- A éste número en formato signo magnitud se aplica el complemento a 1.
- Al complemento a 1 se le suma el bit 1 al bit menos significativo (el que está más a la derecha).
- El número así obtenido es el negativo del número original positivo. Se deberá observar que éste número empieza con el bit 1 en el bit de signo, lo cual indica que es un número negativo.

Por ejemplo, aplicar el complemento a 2 del siguiente número binario: 11001
El número binario 11001, es la magnitud, por lo que se agregará el bit 0 a la izquierda para indicar que se trata de un número positivo.

011001, número binario que corresponde a +25.

Aplicando el complemento a 1 tenemos:

100110

Sumando el bit 1 con el bit LSB, tenemos:

$$\begin{array}{r} 100110 + \\ \underline{1} \\ 100111 \end{array}$$

El número binario **100111** corresponde a -25.

A continuación encontraremos los números negativos de los números de cuatro bits de magnitud.

Número Positivo		Número Negativo	
0	00000		
+1	00001	11111	-1
+2	00010	11110	-2
+3	00011	11101	-3
+4	00100	11100	-4
+5	00101	11011	-5
+6	00110	11010	-6
+7	00111	11001	-7
+8	01000	11000	-8
+9	01001	10111	-9
+10	01010	10110	-10
+11	01011	10101	-11
+12	01100	10100	-12
+13	01101	10011	-13
+14	01110	10010	-14
+15	01111	10001	-15

Tabla 9.2

Tabla de verdad de los quince primeros números positivos y negativos.

Obsérvese que el primer bit de la izquierda es el bit de signo y los otros cuatro son los bits de magnitud.

Téngase presente que todos los números positivos son números binarios naturales al que se le agrega un 0 como bit de signo a la izquierda del MSB; en cambio, el número negativo es el que empieza con el bit 1 y cuya magnitud no se relaciona directamente con la magnitud del positivo. Por ejemplo el **+1 = 00001** y su negativo es **-1 = 11111**.

En la tabla 9.2 se aprecia que las magnitudes de los números del 1 al 15 se encuentran simétricamente distribuidos, por ejemplo, la magnitud de +1 (0001) **sin incluir el signo**, aparece en -15 (0001), lo mismo se verifica entre +2 y -14; +3 y -13; +4 y -12, etc.

IMPORTANTE

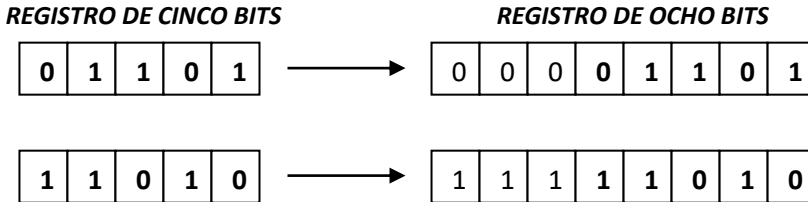
Todo número negativo en complemento a 2 tiene su correspondiente número positivo; para saberlo, simplemente se complementa a 2 al número negativo y se obtiene su número positivo. Por ejemplo, ¿Cuál es su número positivo del siguiente número 1011101 en complemento a 2?

1011101

$$\begin{array}{r} 0100010 + \\ \underline{\quad\quad 1} \\ 0100011 \end{array}$$

0100011 = +35 , por lo tanto el número 1011101=-35

Todo número positivo o negativo en complemento a 2 pueden ser transferidos a registros con mayor número de bits; para ello, a los números positivos se le completa con 0s a la izquierda del bit de signo, mientras que a los números negativos se le completa con 1s. Por ejemplo, transferir el número 01101 y 11010 a un registro de ocho bits.



Un **REGISTRO** es un dispositivo que *almacena datos en forma de bits*.

9.5. Operaciones en el sistema de complemento a 2

Aplicando el método de complemento a 2 se puede realizar operaciones de suma y de resta de números binarios.

9.5.1. Suma en complemento a 2

Para sumar en complemento a 2 se deberá considerar los siguientes casos:

A) Cuando se suman dos números positivos

Por ejemplo, sumar: +5 con +9.

$$\begin{array}{r}
 00101 + \\
 \underline{01001} \\
 01110
 \end{array}$$

Bit de signo Magnitud

$01110_2 = +14_{10}$

B) Cuando se suman dos números de distinto signo. El número positivo mayor.

Por ejemplo, sumar: +7 con -4.

Este cargo se obvia

$$\begin{array}{r}
 0111 + \\
 \underline{1100} \\
 10011
 \end{array}$$

Bit de signo Magnitud

$0011_2 = +3_{10}$

C) *Quando se suman dos números de distinto signo. El número positivo menor.*

Por ejemplo, sumar: -7 con +4.

$$\begin{array}{r}
 1001 + \\
 \underline{0100} \\
 1101
 \end{array}$$

Bit de signo Magnitud

$1101_2 = -3_{10}$

D) *Quando se suman dos números negativos.*

Por ejemplo, sumar: -5 con -9.

Este cargo se obvia

$$\begin{array}{r}
 11011 + \\
 \underline{10111} \\
 1\ 10010
 \end{array}$$

Bit de signo Magnitud

$10010_2 = -14_{10}$

9.5.2. Resta en complemento a 2

Para restar números binarios en complemento a 2 se debe tener en cuenta que al minuendo se le resta el sustraendo, es decir, esta resta se puede realizar mediante una suma con tan solo complementar a 2 el sustraendo y procedemos según los casos de la sección 9.5.1

Restar +5 de +9

La resta de +5 de +9 se puede expresar así: $+9 - (+5) = +9 + (-5)$; se procede como en el caso B.

$$+9 = 01001$$

$$+5 = 00101$$

Operando tenemos:

Este cargo se obvia

$$\begin{array}{r}
 01001 + \\
 11010 \\
 \hline
 1\ 00100
 \end{array}$$

Bit de signo Magnitud

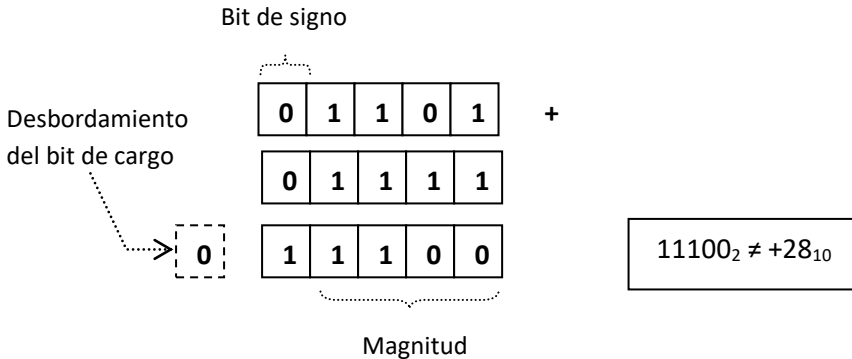
$00100_2 = +4_{10}$

9.5.3. Desbordamiento aritmético

Los números binarios se almacenan en registros con una cantidad determinada de bits. Todas las operaciones que se realicen con dichas cantidades binarias necesariamente tendrán que almacenarse en sus respectivos registros. Cuando se ejecutan operaciones aritméticas cuyos resultados coinciden con el número de bits del registro, no encontraremos errores en el resultado. Sin embargo, si al desarrollar una operación, por ejemplo, la suma de dos números positivos o negativos, el último cargo (bit) rebasa o sale fuera del registro, decimos que existe un desbordamiento aritmético.

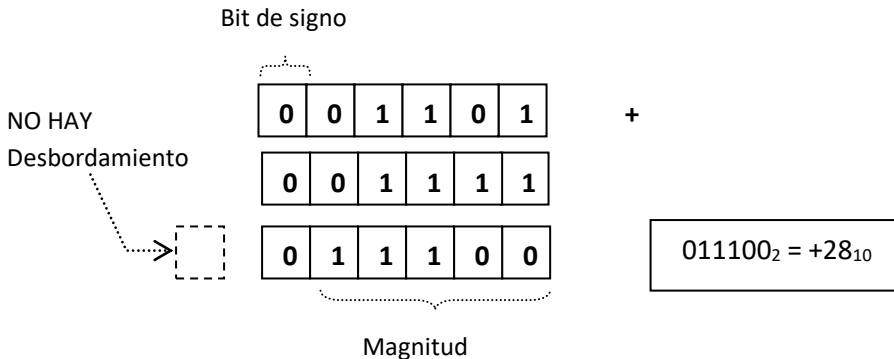
Para detectar un desbordamiento o rebase, todos los circuitos digitales que cuentan con circuitos aritméticos cuentan con circuitos especiales que anuncian al sistema digital que ha ocurrido un error por desbordamiento o rebase.

Por ejemplo, en un registro de cinco bits, sumar +13 con +15



IMPORTANTE

Para sumar correctamente $+13 + (+15)$, el número de bits de los registros deberá ampliarse como mínimo a un registro de 6 bits incluyendo el bit de signo. Así tendremos lo siguiente:



9.6. Suma BCD

Normalmente, en un sistema digital se ingresan números decimales propios de la transferencia de información entre personas, pero al hacerlo se prefiere el sistema BCD, ya que cada dígito decimal (0 – 9) genera un código binario de cuatro bits en el interior del sistema digital, que a su vez se establece como voltajes y corrientes con dos niveles lógicos.

Los cuatro bits del sistema BCD se pueden operar aritméticamente como en el sistema binario, aplicándosele las propiedades correspondientes.

El inconveniente con el sistema BCD se presenta cuando al sumar dos dígitos BCD, su suma es superior a 9; en el caso que sea inferior o igual a 9, el resultado es correcto.

Para salvar el inconveniente mencionado se aplica un factor de corrección, sumando el 6 BCD a la suma > 9.

Sumar en BCD: 23 + 45

$$\begin{array}{r}
 0010 \quad 0011 \quad + \\
 \underline{0100 \quad 0101} \\
 0110 \quad 1000 \\
 \underbrace{\hspace{1.5cm}} \quad \underbrace{\hspace{1.5cm}} \\
 6 \quad 8
 \end{array}$$

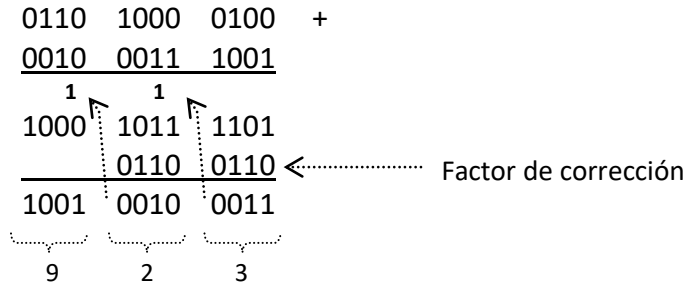
Sumar en BCD: 684 + 239

$$\begin{array}{r}
 0110 \quad 1000 \quad 0100 \quad + \\
 \underline{0010 \quad 0011 \quad 1001} \\
 1000 \quad 1011 \quad 1101 \\
 \underbrace{\hspace{1.5cm}} \quad \underbrace{\hspace{1.5cm}} \quad \underbrace{\hspace{1.5cm}} \\
 8 \quad 11 \quad 13 \\
 \swarrow \quad \searrow \\
 \text{No son BCD}
 \end{array}$$

6 8 4 +
<u>2 3 9</u>
9 2 3

Se aplicará el factor de corrección

0110



9.7. Sumador digital

Es aquel circuito combinacional que efectúa diferentes operaciones aritméticas con una sola operación, la suma. Este circuito combinacional se convierte en secuencial al incorporar registros para almacenar tanto los datos de entrada como de salida.

Los elementos básicos del sumador digital se llama semisumador, mientras que los que realizan una función más completa se llama sumador total.

9.7.1. Semisumador

Es aquel circuito digital que suma dos bits de entrada y devuelve dos bits de salida: el bit suma y el bit cargo.

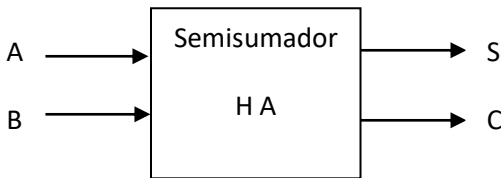


Fig. 9.1

Diagrama de bloques de un semisumador, con dos entradas de un bit cada uno, y dos salidas

De la tabla 9.1, obtenemos las siguientes salidas.

$$\begin{aligned}
 C &= A \cdot B \\
 S &= A \oplus B
 \end{aligned}$$

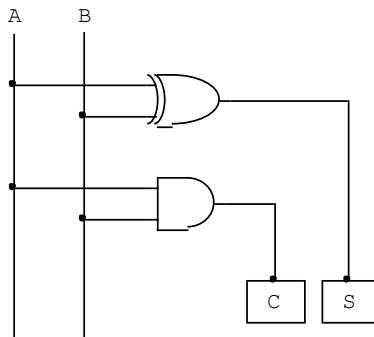


Fig. 9.2

Diagrama lógico de un semisumador, con dos entradas de un bit cada uno, y dos salidas

9.7.2. Sumador total

Es aquel circuito combinacional que agrega al semisumador un terminal más en su entrada, con la finalidad de incluir el cargo previo o de entrada (Cin).

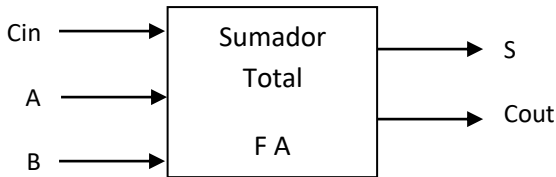


Fig. 9.3

Diagrama de bloques de un sumador total, con dos entradas de un bit cada uno, un bit de carga inicial y, dos salidas.

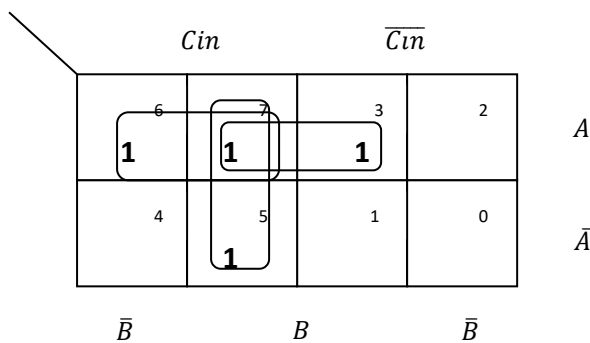
Elaboramos su tabla de verdad.

N°	ENTRADA			SALIDA	
	Cin	A	B	Cout	S
0	0	0	0	0	0
1	0	0	1	0	1
2	0	1	0	0	1
3	0	1	1	1	0
4	1	0	0	0	1
5	1	0	1	1	0
6	1	1	0	1	0
7	1	1	1	1	1

Tabla. 9.3

Tabla de verdad de un sumador completo, con tres variables de entrada y dos salidas.

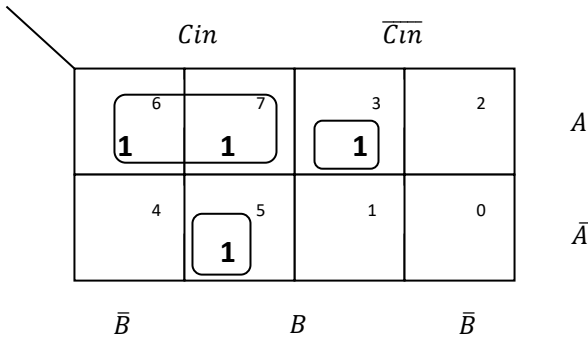
Hallando el **cargo de salida (Cout)** con la ayuda de una tabla de verdad.



$$Cout = Cin.A + A.B + Cin.B$$

No obstante la validez de lo hallado en el M.K, es necesario tener en cuenta que la finalidad de simplificar funciones es optimizar los recursos de los dispositivos y componentes electrónicos, por ello, emplearemos otra forma de agrupar los

minitérminos para llegar a optimizar el cargo de salida del sumador.



$$Cout = Cin \cdot A + \overline{Cin} \cdot A \cdot B + Cin \cdot \bar{A} \cdot B$$

$$Cout = Cin \cdot A + B \cdot (\overline{Cin} \cdot A + Cin \cdot \bar{A})$$

$$Cout = Cin \cdot A + B \cdot (Cin \oplus A)$$

Para hallar la suma (S) utilizaremos el método de la tabla, pues evaluando los minitérminos de S , una tabla de verdad no es de gran ayuda.

$$S = \overline{Cin} \cdot \bar{A} \cdot B + \overline{Cin} \cdot A \cdot \bar{B} + Cin \cdot \bar{A} \cdot \bar{B} + Cin \cdot A \cdot B$$

Aplicando las propiedades del álgebra de Boole tenemos:

$$S = \overline{Cin} \cdot (\bar{A} \cdot B + A \cdot \bar{B}) + Cin \cdot (\bar{A} \cdot \bar{B} + A \cdot B)$$

$$S = \overline{Cin} \cdot (A \oplus B) + Cin \cdot (\bar{A} \oplus \bar{B})$$

$$S = Cin \oplus A \oplus B$$

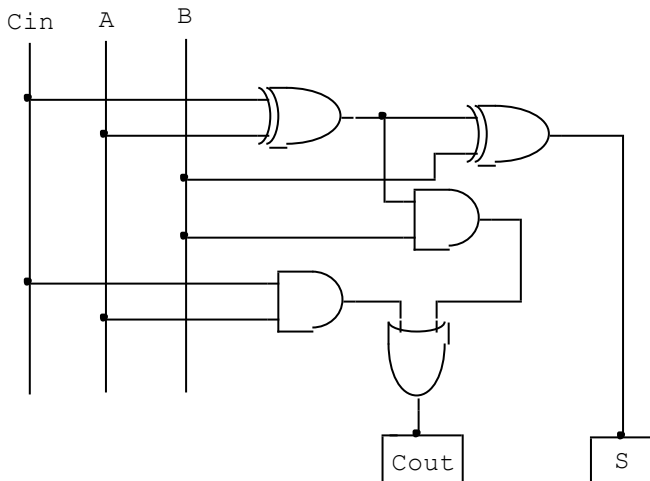


Fig. 9.4

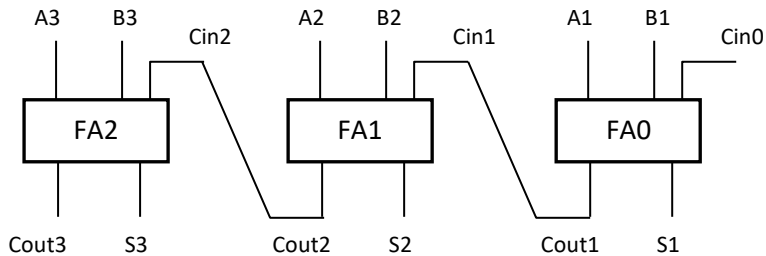
Diagrama lógico de un sumador completo.

9.7.3. Diseño de un sumador total

Para implementar un sumador total de n bits se requerirá n bloques de F.A enlazados entre si a través de los cargos de salida y entrada.

Por ejemplo: implementar un sumador total de dos números de tres bits cada uno.

$$\begin{array}{r} A3 \ A2 \ A1 \ + \\ \hline B3 \ B2 \ B1 \\ \hline \text{Cout3} \ S3 \ S2 \ S1 \end{array}$$



9.7.4. Circuitos restadores

Para restar dos números, se cambia de signo al sustraendo y luego se suma con el minuendo. Para efectuar el cambio de signo en el sustraendo, se utilizan compuertas OR EXCLUSIVO para complementar a 1, luego en el bloque sumador se agrega el bit 1 para complementarlo a 2 y enseguida sumarlo conjuntamente con el minuendo.

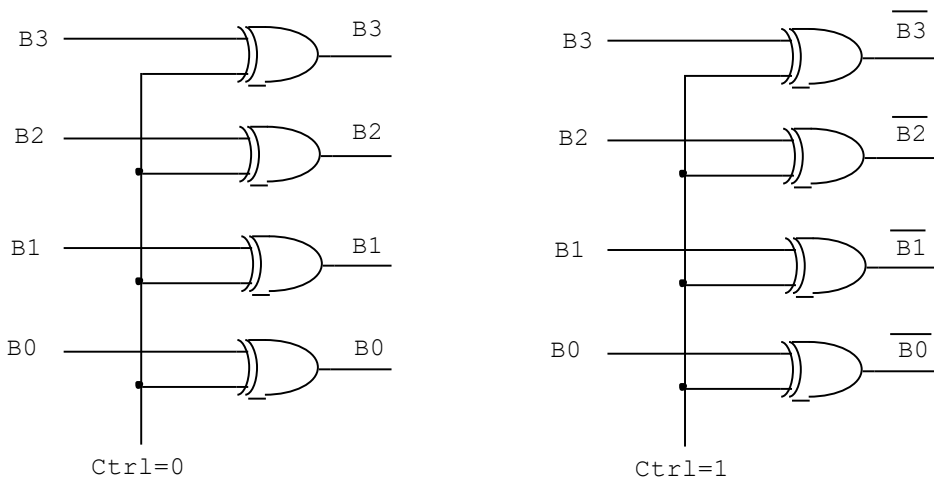


Fig. 9.5
Control para ingresar datos sin complementar ó complementados.

De la Fig. 9.5, se verifica que si el Ctrl=0, los datos salen sin complementarse, en cambio, si Ctrl=1, los datos salen complementados. Con este circuito control podemos efectuar una suma o una resta según convenga.

9.8. Bloques sumadores prácticos de 4 bits de datos

El circuito integrado 7483 puede sumar dos números de cuatro bits cada uno

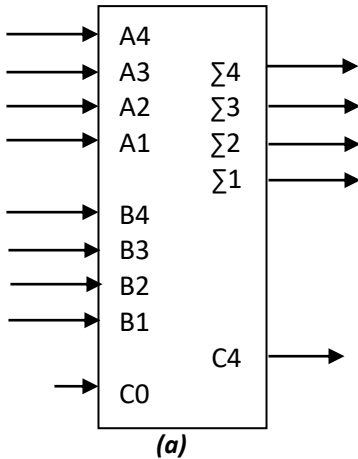
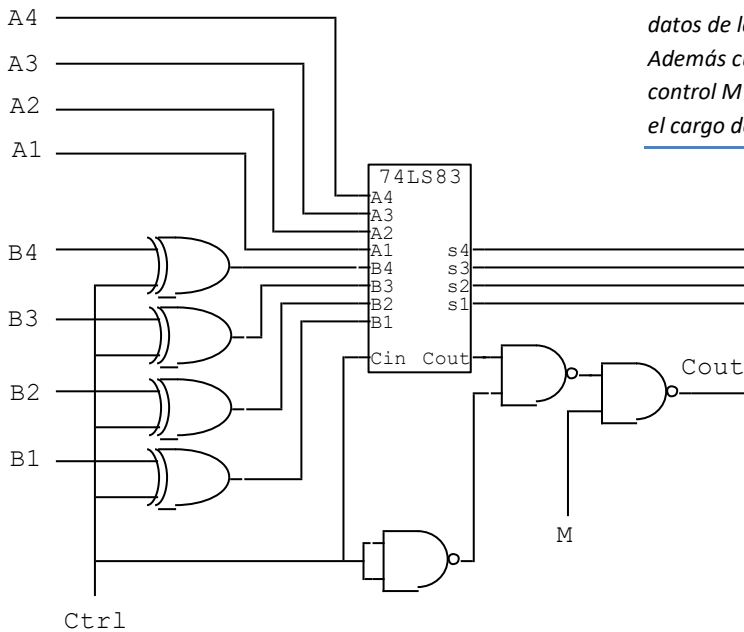


Fig. 9.6

(a) Diagrama de bloques del circuito integrado 7483, que realiza la suma de dos números de cuatro bits cada uno.

(b) Diagrama lógico de un sumador completo 7483, que cuenta con un circuito de control para el ingreso de datos de la variable B. Además cuenta con un control M para permitir o no el cargo de salida.



El circuito sumador de la Fig. 9.6 (b), cuenta con dos controles; uno de ellos, el terminal **Ctrl** para complementar o no complementar los datos de B; el otro control **M**, para permitir o no permitir la salida del cargo final. Ambos controles se detallan en las siguientes tablas de verdad.

N°	Ctrl	OPERACIÓN	Cin	Función
0	0	SUMA	Ingresar 0	El dato B (B1-B4) ingresa al sumador sin complementarse. El cargo inicial es 0.
1	1	RESTA (A-B)	Ingresar 1	El dato B (B1-B4) ingresa al sumador complementado y se suma con 1 del Cin. (COMPLEMENTO A 2)

N°	M	Condición	Función
0	0	A>B	Resultado con signo positivo
1	1	A<B	Resultado con signo negativo

Tabla. 9.4

Tabla de los controles (Ctrl) y (M) del sumador de cuatro bits.

IMPORTANTE:

La señal del control M, puede provenir de la salida de un circuito comparador.

El circuito de complementación se puede implementar con el C.I. 7486, el cual tiene cuatro compuertas OR-EXCLUSIVO.

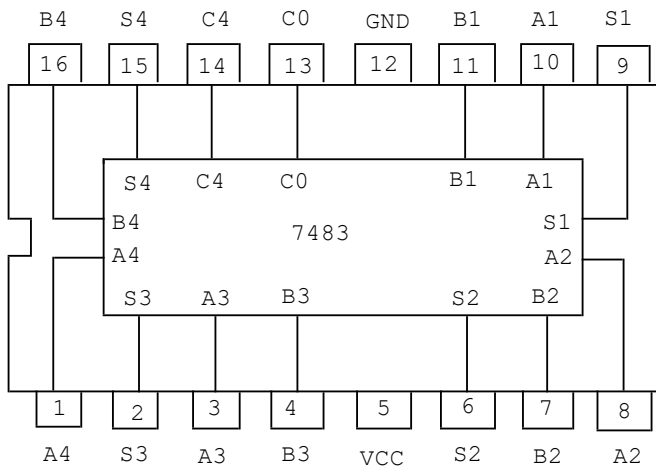


Fig. 9.7

Circuito Integrado 7483, sumador completo FULL ADDER, de dos variables (A y B) de cuatro bits cada uno.

Problemas Resueltos N° 09

1. Explique el fundamento de los números binarios con signo mediante el método de complemento a dos.

Solución:

Para representar números binarios con signo, se utilizó el método de complemento a dos, también denominado, de **complemento a la base**.

El fundamento de éste método consiste en representar un número negativo por una cantidad que le falta para completar a una cantidad que se toma como base, equivalente a $2^{\# \text{ variables del número binario positivo}}$.

Ejemplo:

El número +3 en binario es: 011 (tiene tres variables); entonces, la cantidad que se toma como base es $2^3=8$; donde 8 en binario es: 1000.

Al número +3 le falta el número 5 (101 en binario) para completar 8.
Por tanto, el número -3 es igual a 101 en binario.

Si sumamos: $7+(-3)=4$, en binario **100**

Coherente con el resultado de la suma de:

$$\begin{array}{r} 111+ \\ 101 \\ \hline 1100 \end{array}$$

Se obvia, pues, los números son de tres bits. →

2. ¿Cuál es el algoritmo para calcular el complemento a 2 de un número binario?

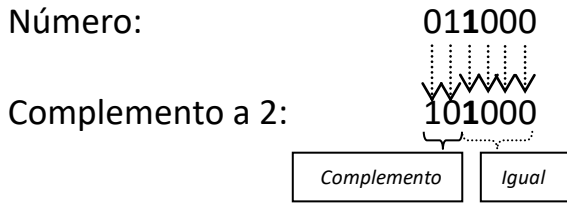
Solución:

Algoritmo para calcular el complemento a 2 de un número binario.

- Evaluamos el número binario de derecha a izquierda.
- Si el primer bit de la derecha es 0, escribimos en la parte inferior, el bit 0; si el bit consecutivo de la izquierda es nuevamente es 0, escribimos a la izquierda del bit anterior, el bit 0.
- El paso anterior se repite hasta encontrar el primer bit 1. El bit 1 encontrado se escribe en la parte inferior.
- A partir el primer bit encontrado, todos los demás bits que quedan hacia la izquierda se escriben sus respectivos complementos en la parte inferior.

Ejemplos:

Hallar el complemento a 2 de los siguientes números binarios:



Número: 01100
Complemento a 2: 10100

Número: 10110
Complemento a 2: 01010

Número: 011
Complemento a 2: 101

Número: 1101100
Complemento a 2: 0010100

3. Un sistema de computadoras utiliza un rango de posiciones de memoria para guardar su Data, que va desde la posición 02F_H hasta 3EF_H y desde 40A_H hasta 6FF_H. Calcular la cantidad total de posiciones de memoria del sistema tanto en hexadecimal como en decimal.

Solución:

Cuando deseamos calcular la cantidad de posiciones de memoria de un sistema, por ejemplo de la posición 0 a la posición 5 tenemos:

Posición
0
1
2
3
4
5

Hay seis posiciones, es decir, hay: (5-0)+1=6

Por tanto, para calcular la cantidad de posiciones de memoria (CPM) se utilizará la siguiente ecuación:

$$\text{CPM} = (\text{posición final} - \text{posición inicial}) + 1$$

$$\text{CPM} = [(3EF - 02F + 1) + (6FF - 40A + 1)]_H$$

$$\text{CPM} = [3C1 + 2F6]_H$$

$$\text{CPM} = [6B7]_H = 1719$$

4. Efectuar la suma de dos números hexadecimales con signo. Compruébelo en el sistema binario con signo (complemento a la base) y en el sistema decimal.

$X = 7B_H$

$Y = 94_H$

Solución:

El dígito más significativo (MSD) de un número hexadecimal con signo, será:

Positivo, si: $0 \leq \text{MSD} \leq 7$,

Negativo, si: $8 \leq \text{MSD} \leq F$

Teniendo en cuenta lo anterior tenemos que $7B_H$ es **positivo** y 94_H es **negativo**.

<u>Hexadecimal</u>	<u>Binario</u>	<u>Decimal</u>
$7B +$	$01111011 +$	$+123 +$
94	10010100	-108
$\hline 10F$	$\hline 100001111$	$\hline +015$

Se obvía

5. Dos números binarios de ocho bits con signo, incluyendo el signo, se transfieren a otra localidad de dieciséis bits. Muestre los bits de dichos números después de su transferencia.

$X = 01011101$ $+93$ en decimal

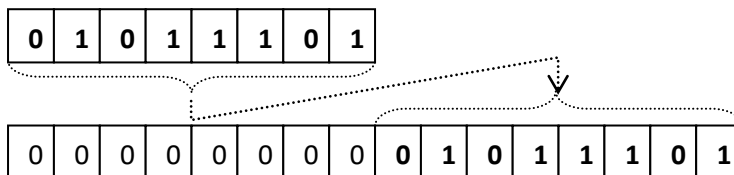
$Y = 10111010$ -70 en decimal

Solución:

Se entiende que los números con signo se expresan en complemento a 2. Por tanto, el número positivo se transfiere a un registro de dieciséis bits tal como está, a las posiciones menos significativas. Las demás posiciones, se completan con 0.

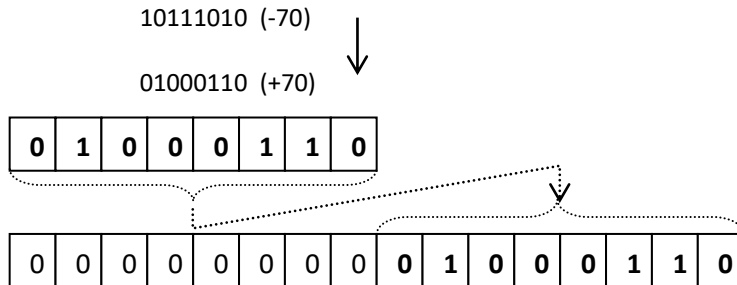
Los números negativos, primero se convierten a positivo, luego se completa con 0 las posiciones sobrantes, y finalmente el número se vuelve a convertir a negativo.

Transfiriendo el número positivo X, de 8 bits, a 16 bits.

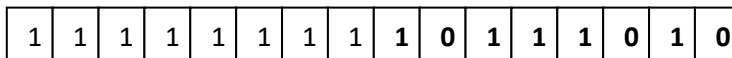


Transfiriendo el número negativo Y, de 8 bits, a 16 bits.

El número negativo 10111010, se expresa en positivo mediante el complemento a 2.



Este número binario (+70) de 16 bits se complementa a 2 para obtener su número negativo (-70)



Se observa que la transferencia de un número binario negativo de 8 bits a 16 bits, se logra completando con el bit 1 a todas las localidades restantes.

En conclusión:

Un número binario positivo se transfiere a un registro con más posiciones, agregando ceros a las localidades restantes.

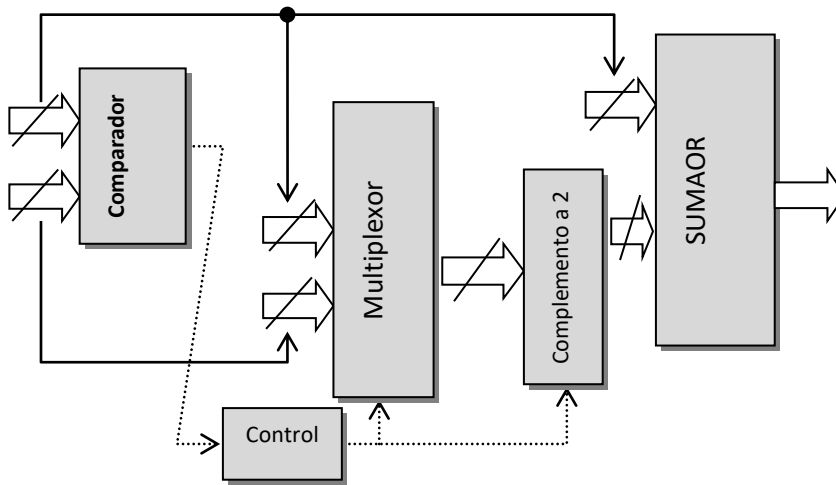
Un número binario negativo se transfiere a un registro con más posiciones, agregando unos a las localidades restantes.

6. Implemente un circuito aritmético, en el que ingresando dos números A y B, codificados en BCD (de cuatro bits), realice las siguientes operaciones:
 - a. Si $A < B$, realiza $A + B$.
 - b. Si $A = B$, realiza $2 * A$
 - c. Si $A > B$, realiza $A - B$.

Solución:

Para implementar un circuito que tiene varias etapas, es necesario elaborar un diagrama de bloques para que guíe el diseño.

- Un bloque para que compare los dos números BCD.
- Un bloque donde ingresen ambos números y proceda a multiplexar según el requerimiento.
- Un bloque aritmético.
- Un bloque de control para que realice las operaciones solicitadas.



Las tres operaciones implican que el número A (**A3 A2 A1 A0**) ingrese directamente al sumador.

Cuando $A < B$; el multiplexor deja pasar el número B (**B3 B2 B1 B0**) hacia el circuito de complemento. Éste deja pasar **B** hacia el sumador, pero sin complementar; allí se suma con **A**.

Cuando $A > B$; el multiplexor deja pasar el número B hacia el circuito de complemento. Éste deja pasar B hacia el sumador, pero complementado; allí se resta de A.

Cuando $A = B$; el multiplexor deja pasar **A**, hacia el circuito de complementación. Éste circuito deja pasar **A** al sumador sin complementarlo. El sumador realiza la suma de: $A + A$, lo que es igual a $2 \cdot A$.

El circuito de control depende de las salidas $A < B$; $A = B$; y $A > B$.

A < B	A = B	A > B	Expresión lógica	Descripción	Función lógica
P	Q	R			
0	0	1	$\bar{P}\bar{Q}R$	Deja pasar B	1
0	1	0	$\bar{P}Q\bar{R}$	Deja pasar A	0
1	0	0	$P\bar{Q}\bar{R}$	Deja pasar B	1

Para que el multiplexor deje pasar el número B:

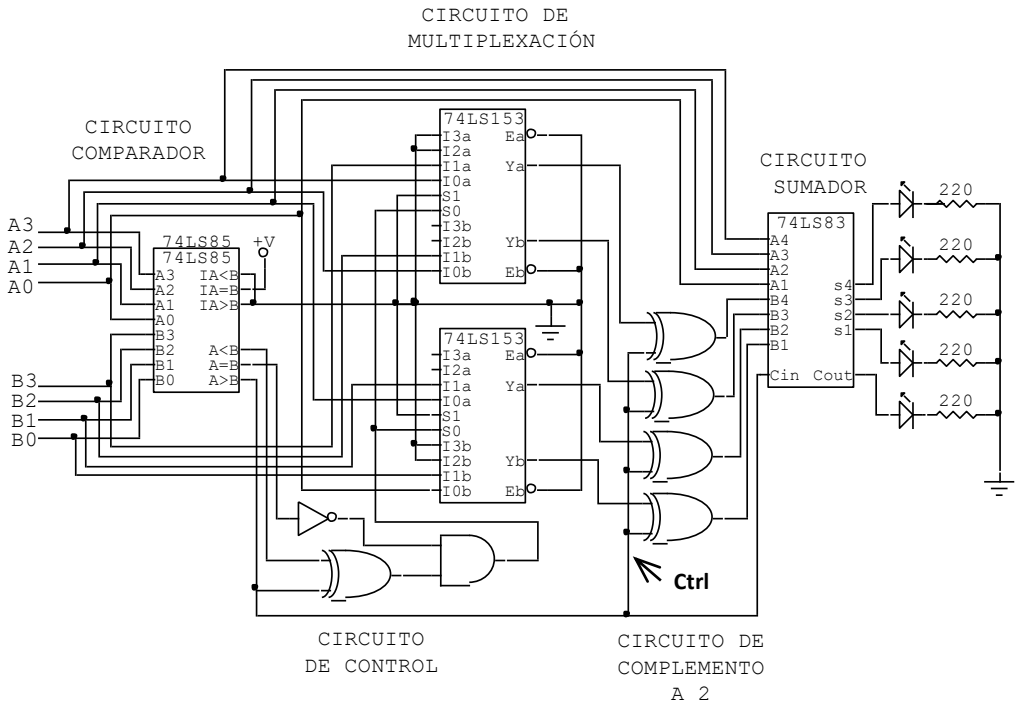
$$f = \bar{P}\bar{Q}R + P\bar{Q}\bar{R} = \bar{Q}(\bar{P}R + P\bar{R}) = \bar{Q}(P \oplus R)$$

Para que el circuito de complementación realice el proceso de complementación del número multiplexado, se debe tener en cuenta lo siguiente:









- El circuito de complementación realiza la complementación de un número cuando el control (Ctrl) es 1.
- El circuito de complementación deja pasar el número sin complementar cuando el control (Ctrl) es 0.





A < B	A = B	A > B	Ctrl	Descripción
0	0	1	1	Si A > B, el circuito complementa a A.
0	1	0	0	Si A = B, el circuito deja pasar el número A, sin complementar.
1	0	0	0	Si A < B, el circuito deja pasar el número B, sin complementar.

Se observa que: **Ctrl = A > B**



Problemas Propuestos N° 09

1. Efectúe las siguientes operaciones de números con signo:
 - a. Binarios:
 -  $1011011101 - 0110110110$
 -  $0110101011 - 1001101110$
 -  $1101011110 - 1011000101$
 -  $0111110010 - 0111010100$
 - b. Octales:
 -  $34 - 26$
 -  $27 - 53$
 -  $62 - 15$
 -  $74 - 61$

Téngase en cuenta que el signo de un número octal corresponde al MSD.
Si $0 \leq \text{MSD} \leq 3$, es un número positivo.
Si $4 \leq \text{MSD} \leq 7$, es un número negativo.
 - c. Hexadecimales:
 -  $2E4 - 68A$
 -  $4B7 - 759$
 -  $C62 - 1F5$
 -  $FFE - D78$
2. Implemente un circuito aritmético, en el que ingresando dos números A y B, codificados en BCD (de cuatro bits), realice las siguientes operaciones:
 - a. Si $A < B$, realiza $B - A$.
 - b. Si $A = B$, realiza $A + B$
 - c. Si $A > B$, realiza $A - B$.
3. Si la entrada **Cin** de la Fig. 9.6 (b) no estuviera conectado a la entrada de la compuerta NAND, que cumple la función NOT (ó conectado al terminal control **Ctrl**), sino a la salida de dicha compuerta. ¿Qué problemas obtendríamos con la SUMA y con la RESTA de dos números?
4. Los pines del circuito integrado de la Fig. 9.7 durante su funcionamiento, tiene los siguientes nivel lógicos: Pin (1) = BAJO; Pin (2) = ALTO; Pin (3) = ALTO; Pin (4) = ALTO; Pin (6) = BAJO; Pin (7) = BAJO; Pin (8) = ALTO; Pin (9) = BAJO; Pin (10) = ALTO; Pin (11) = BAJO; Pin (13) = ALTO; Pin (14) = ALTO; Pin (15) = BAJO; Pin (16) = ALTO. Compruebe su funcionamiento e indique si funciona correctamente o no.

Capítulo 10

Circuitos secuenciales

10.1. Introducción a los circuitos secuenciales

Los circuitos secuenciales, son aquellos circuitos combinatoriales en los que su salida depende tanto de los valores de las variables de entrada como de la salida. Cuando la salida del circuito ingresa nuevamente a la entrada, se llama “retroalimentación”. Si la retroalimentación se realiza mediante sincronización de pulsos de reloj, decimos que el circuito secuencial es “sincrónico”; si se realiza sin pulsos de reloj, se denomina “asincrónico”.

En términos generales, la realimentación es un proceso que implica el uso de memoria, circuito capaz de guardar información en elementos físicos para tal fin.

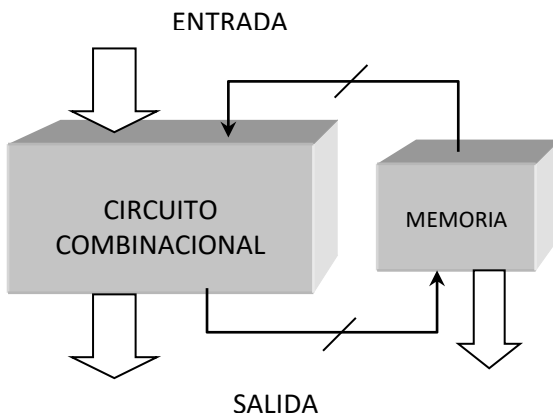


Fig. 10.1

Diagrama de bloques
de un circuito
secuencial

La memoria guarda información bit por bit. Por ello, este dispositivo se compone de varios elementos de memoria denominados BIESTABLES o FLIP FLOP (FF).

Se llaman “biestables”, porque en la salida del circuito están presentes los dos niveles de voltajes complementados y estables (Q y \bar{Q}) según las características de la entrada.

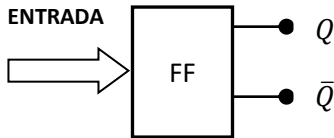


Fig. 10.2

Diagrama de bloques de un biestable o Flip Flop.

IMPORTANTE

Las salidas Q y \bar{Q} son complementarias, es decir, sus valores adoptan uno cualquiera de los dos valores posibles, y son mutuamente exclusivos.

Ambas salidas del Flip Flop se retroalimentan enlazándose con sus entradas.

10.2. Flip Flop SR

Los Flip Flop SR son biestables elaborados con compuertas lógicas, cuyas salidas Q y \bar{Q} se enlazan transversalmente en el proceso de retroalimentación con sus dos entradas denominadas S (Set) y R (Reset).

Según el tipo de compuerta empleado en su construcción se clasifican en Flip Flop de compuertas NAND y Flip Flop de compuertas NOR.

10.2.1. Latch de compuertas NAND

Son Flip Flop elaborados con compuertas NAND.

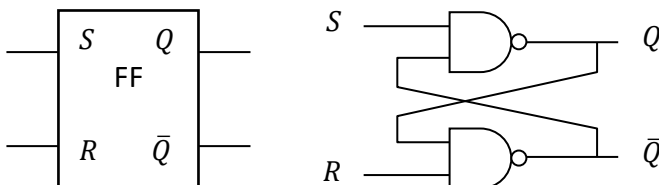


Fig. 10.3

Diagrama lógico de un Latch S-R con compuertas NAND.

N°	ENTRADA		SALIDA
	<i>S</i>	<i>R</i>	<i>Q</i>
0	0	0	$Q = \bar{Q} = 1$; inconsistente ó prohibido
1	0	1	1
2	1	0	0
3	1	1	<i>Q</i> , adopta valores anteriores, historia previa .

Tabla . 10.1

Tabla de verdad o excitadora del Latch S-R con compuertas NAND. Obsérvese que la entrada 00, es una combinación prohibida.

Cuando ambas entradas *S* y *R* del Latch de compuertas NAND están en nivel BAJO, provocan el **establecimiento** de las salidas *Q* y \bar{Q} en nivel ALTO; lo cual es inconsistente con la lógica, pues como se observa claramente, ambas salidas deberían ser complementarias y nunca **establecerse** o **restablecerse** al mismo tiempo. A esta combinación de *S* y *R* también se llama “prohibido”

Establecer la salida, es poner a *Q* en nivel **ALTO**.

Restablecer la salida, es poner a *Q* en nivel **BAJO**.

Cuando $S=0$ y $R=1$, la salida *Q* se establece en nivel ALTO; mientras que si $S=1$ y $R=0$, la salida se restablece en nivel BAJO.

En cambio, si ambas entradas están en nivel ALTO, las salidas se mantienen con los mismos valores anteriores al cambio, es decir, la historia previa se guarda con la combinación $S=1$ y $R=1$. En este caso ambas salidas se mantienen complementadas y coherentes con la lógica digital.

El análisis del comportamiento de un sistema secuencial se realiza a través de los diagramas de tiempo y en función de su tabla de verdad.

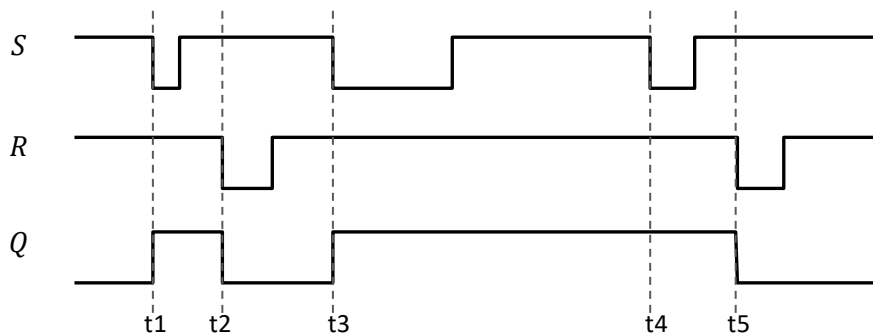


Fig. 10.4

Diagrama de Tiempo del Flip Flop S-R con compuertas NAND.

La salida Q , de la Fig. 10.4 está en nivel BAJO, asumiendo arbitrariamente que guarda su historia previa en ese nivel.

Una aplicación importante del Latch S-R con compuertas NAND es el eliminador de rebote. Al trabajar en electrónica digital se presenta un problema al momento de pulsar los tact switch, interruptores, o simplemente al conectar los cables de entrada a tierra o a fuente. El contacto mecánico en estos dispositivos no se realiza de un solo golpe, sino que lo hacen en varios contactos en lapsos pequeños de tiempo; esta actividad se conoce como “rebote” y es una de las causas más importantes de fallos en los sistemas y entrega de datos.

A la salida de los interruptores se instala un Latch S-R NAND y resolvemos el problema de rebote, tal como se ilustra a continuación.

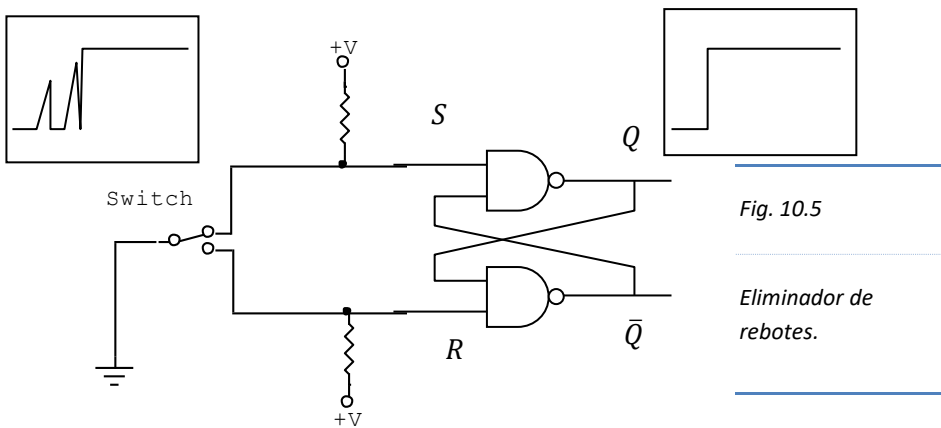


Fig. 10.5

Eliminador de rebotes.

10.2.2. Latch de compuertas NOR

Son Flip Flop elaborados con compuertas NOR. La característica especial respecto a las compuertas NAND es que los niveles ALTOS activan la compuerta.

A diferencia de los Latch de compuertas NAND, las entradas de las compuertas NOR se han intercambiado, es decir, en la entrada superior se encuentra el terminal que restablece R , Reset o también llamado “Clear”, y en el inferior está S .

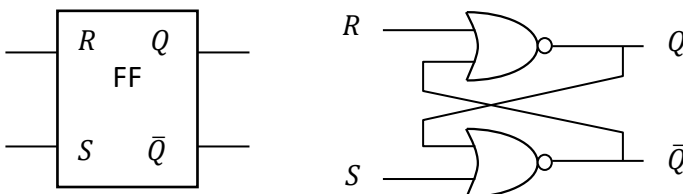


Fig. 10.6

Diagrama lógico de un Latch S-R con compuertas NOR.

N°	ENTRADA		SALIDA
	<i>S</i>	<i>R</i>	<i>Q</i>
0	0	0	<i>Q</i> , adopta valores anteriores, historia previa.
1	0	1	0
2	1	0	1
3	1	1	$Q = \bar{Q} = 0$; inconsistente ó prohibido

Tabla . 10.2

Tabla de verdad o excitadora del Latch S-R con compuertas NOR. Obsérvese que la entrada 11, es una combinación prohibida.

Si ambas entradas *S* y *R* están en nivel BAJO, las salidas se mantienen con los mismos valores anteriores al cambio, es decir, la historia previa se guarda con la combinación $S=0$ y $R=0$, a este estado del Flip Flop se conoce como “memoria”.

Cuando $S=0$ y $R=1$, la salida *Q* se restablece a un nivel BAJO; mientras que si $S=1$ y $R=0$, la salida se establece en nivel ALTO.

En cambio, cuando ambas entradas *S* y *R* del Latch de compuertas NOR están en nivel ALTO, provocan el **restablecimiento** de las salidas *Q* y \bar{Q} a un nivel BAJO; lo cual es inconsistente o “prohibido” en el campo de la lógica digital.

El análisis del comportamiento de un Flip Flop se realiza a través de los diagramas de tiempo y en función de su tabla de verdad.

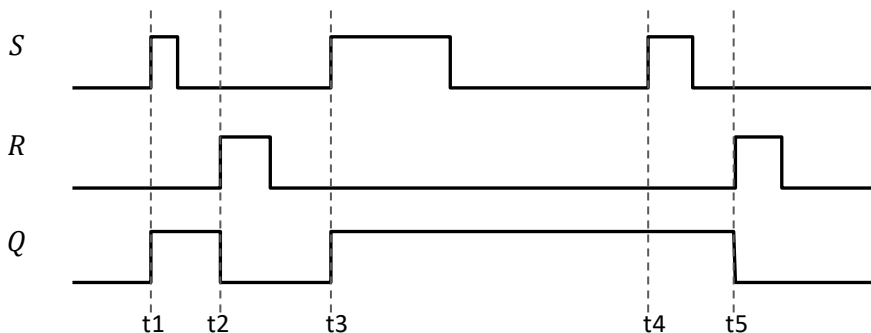


Fig. 10.7

Diagrama de Tiempo del Flip Flop S-R con compuertas NOR. Se asume inicialmente que *Q* está en nivel BAJO.

Una aplicación importante del Latch S-R con compuertas NOR es la activación o desactivación de un circuito digital mediante tact switch normalmente cerrado. El circuito digital puede ser el de poner en marcha un motor o detenerlo. Asimismo, si uno de los tact switch se reemplaza por un sensor óptico como un fototransistor, el circuito se convierte en un activador de alarma.

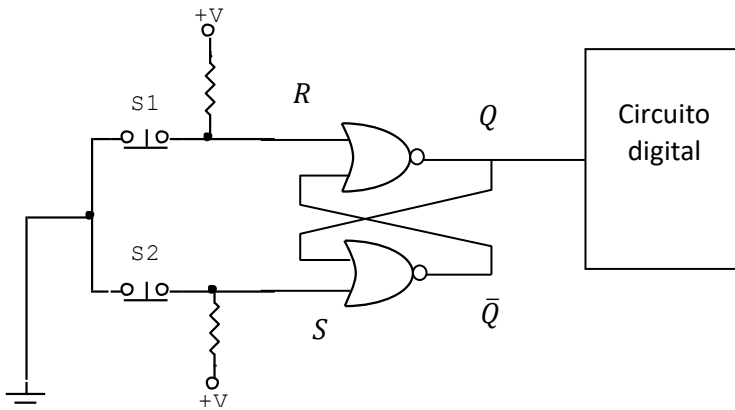


Fig. 10.8

Diagrama lógico de un activador o desactivador de circuito digital. Se puede comportar como sensor de intrusos al acoplarle un fototransistor en S1.

S1 y S2 son tact switch normalmente cerrados, es decir, al pulsarlos se abren y aíslan el sistema de tierra y por tanto S y R se ponen a nivel ALTO (al voltaje que indique +V).

Inicialmente R y S están en nivel BAJO, si luego se pulsa S2, S se pone en nivel ALTO y S1 queda en BAJO, por tanto, Q se establece en ALTO y activa al “circuito digital”. Si se deja de pulsar S2, S vuelve a conectarse a tierra proveyéndole un nivel BAJO; en estas circunstancias Q se mantiene en ALTO pues se encuentra en MEMORIA y se queda con su historia previa. El circuito digital se mantiene activado pues Q se mantiene en ALTO.

Para desactivar dicho circuito, simplemente se pulsa S1, el cual restablece a Q en nivel BAJO. Este pulso desactiva al circuito digital.

Instalando un transistor y un relé en el circuito digital se podría activar y desactivar una bombilla eléctrica y si además los tact switch se activan con pulsos, la activación y desactivación se podría realizar remotamente.

El circuito digital también podría ser una alarma, y para implementarlo, en lugar de S1 podríamos instalar un fototransistor con el Emisor a tierra y el Colector conectado a S. Cuando incide luz en la Base del transistor, éste conduce y provoca una caída de voltaje entre Colector y Emisor con un nivel de voltaje BAJO, esta zona de trabajo del transistor se conoce como SATURACIÓN. Cuando se interrumpe la luz, la corriente de base se interrumpe y el transistor entra en CORTE, provocando una caída de voltaje entre Colector y Emisor equivalente a +V, es decir un nivel de voltaje ALTO, el cual activa la alarma y así se restablezca el haz de luz en el transistor, la alarma no se desactivará pues la salida del Flip Flop se mantendrá con su historia previa, $Q=1$.

10.3. Sincronización de los biestables

Los circuitos secuenciales son fundamentalmente síncronos, es decir, que los biestables cuentan con una entrada de pulsos que sincronizan el funcionamiento de todos los elementos del sistema.

Los pulsos son generados por un circuito llamado circuito generador de reloj.

10.3.1. Circuito generador de reloj

El circuito generador de pulsos de sincronismo es el multivibrador astable. Este tipo de circuito presenta una salida en dos estados de forma inestable que genera una secuenciación de subidas y bajadas de voltaje en un periodo determinado.

Existen dispositivos que permiten generar pulsos de reloj con alta precisión como los cristales piezoeléctricos o no tan altas como los circuitos RC ó el timer, cuya función es proporcionar pulsos de sincronización a los circuitos digitales.

El timer 555, es un temporizador implementado en circuito integrado para cumplir diversas funciones y aplicaciones electrónicas, entre ellas, la de multivibrador astable. Con un sencillo acoplamiento de dos resistencias y un condensador, el timer genera en su salida un tren de pulsos de onda cuadrada con frecuencias apropiadas para efectuar trabajos prácticos y de experimentación.

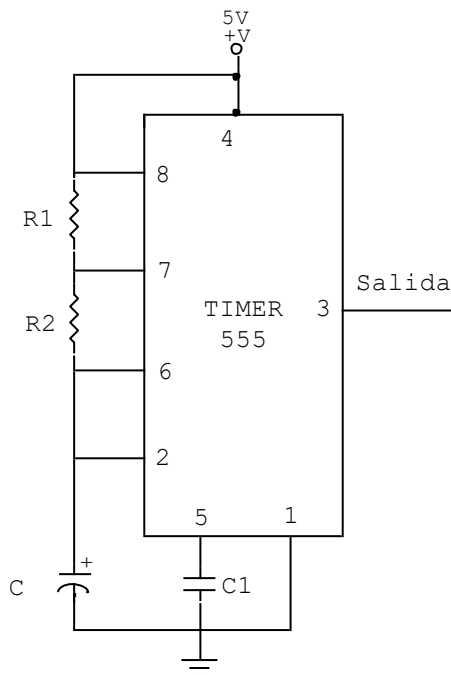


Fig. 10.9

Diagrama de distribución de Pines de un multivibrador astable 555, en el que R1, R2 y C determinan la frecuencia de trabajo. La salida del tren de pulsos de onda cuadrada se realiza por el terminal N° 3.

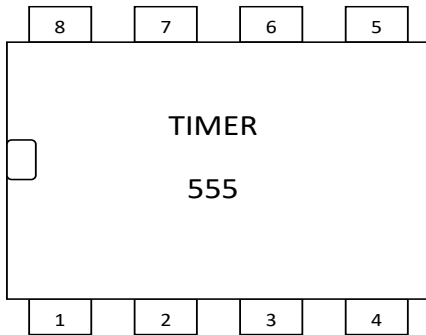


Fig. 10.10

Circuito Integrado 555
TIMER/OSCILADOR de baja
potencia.

Nº PIN	DESCRIPCIÓN DE LOS TERMINALES
1	Masa ó tierra (GND)
2	Comparador de disparo (TRIGGER)
3	Salida (OUTPUT)
4	Habilitación, borrado (RESET)
5	Control (CONTROL VOLTAGE)
6	Umbral (THEREHOLD)
7	Llave electrónica o descarga (DISCHARGE)
8	Alimentación (VCC)

Tabla. 10.3

Descripción de los
terminales del circuito
integrado 555

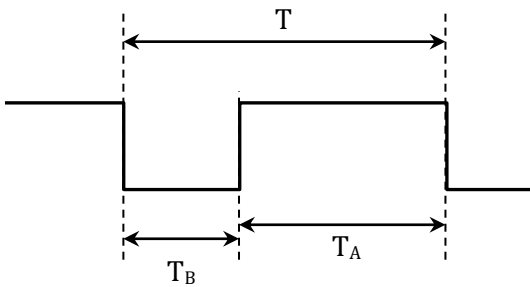


Fig. 10.11

Onda cuadrada de un tren
se pulsos, en el que una
onda completa se realiza en
un periodo T.

$$T_B = 0.693(R_2 \cdot C)$$

$$T_A = 0.693(R_1 + R_2)C$$

$$T = T_A + T_B$$

$$f = \frac{1}{T}$$

$$CT = \frac{T_A}{T} \times 100\%$$

Donde:

T_B: Periodo en Nivel Bajo

T_A: Periodo en Nivel Alto

T: Periodo Total

f: Frecuencia

CT: Ciclo de Trabajo

Ejemplo:

Calcular R1 y R2 si deseamos que un multivibrador astable 555 genere una frecuencia de 0,5 KHz, con un ciclo de trabajo del 60%, si disponemos de un condensador de 0,47 μ F.

Solución:

$$T = \frac{1}{0,5} \times 10^{-3} \text{s} = 2 \text{ms}$$

$$\frac{T_A}{T} \times 100\% = 60\%$$

$$T_A = \frac{3}{5}T = \frac{3}{5} \times 2 \text{ms} = \frac{6}{5} \text{ms} = 1,2 \text{ms}$$

$$T_B = \frac{4}{5} \text{ms} = 0,8 \text{ms}$$

$$R2 = \frac{0,8 \times 10^{-3} \text{s}}{0,693 \times 0,47 \times 10^{-6} \text{F}} = 2,4 \text{K}\Omega$$

$$R1 = \frac{1,2 \times 10^{-3} \text{s}}{0,693 \times 0,47 \times 10^{-6} \text{F}} - 2,4 \text{K}\Omega = 1,2 \text{K}\Omega$$

IMPORTANTE

Datos empíricos demuestran que R1 debe ser mayor o igual a 1K Ω ; que la suma de R1 y R2 no debe superar los 6.6 M Ω ; y que el condensador C debe ser mayor o igual a 500 pF.

En ocasiones, se necesita trabajar en bajas frecuencias, sobre todo para observar mediante LEDs o visualizadores la secuencia realizada por algunos circuitos digitales. Para lograr una frecuencia apropiada se debe disponer de varios valores de condensadores electrolíticos como: 1 μ F; 2,2 μ F; 4,7 μ F y 10 μ F, entre 16V y 35V; además, se debe disponer de un resistor variable o de preajuste entre 500K Ω y 1M Ω .

El condensador C1 de la Fig. 10.9 constituye un dispositivo para evitar inestabilidades en el multivibrador astable, con un valor de 10 nF aproximadamente. Los condensadores cerámicos expresan su valor en picofaradios, por ejemplo: en el condensador 103, el tercer dígito (3) indica la cantidad de ceros que se añadirán a los dos primeros (10); 103 = 10000 pF = **10 nF** = 0,01 μ F

10.3.2. Pulsos digitales

El tren de pulsos consiste en un conjunto de subidas y bajadas de voltaje con una frecuencia determinada, lo que nos da una idea de una onda cuadrada, sin embargo, en términos reales la subida de un nivel bajo a uno alto no lo hace instantáneamente, sino que lo realiza en un determinado tiempo o “periodo de transición” (T_s) y (T_b), lo que nos da una idea de pendiente o “flanco”, tanto de subida como de bajada.

A la onda que se mantiene en nivel ALTO se conoce como pulso POSITIVO, mientras que a la onda que se mantiene en nivel BAJO se denomina pulso NEGATIVO.

En la práctica a los pulsos se considera como una onda cuadrada sin pendiente, por ello, cuando hay una secuencia de pulsos positivos y negativos se denomina “tren de pulsos”

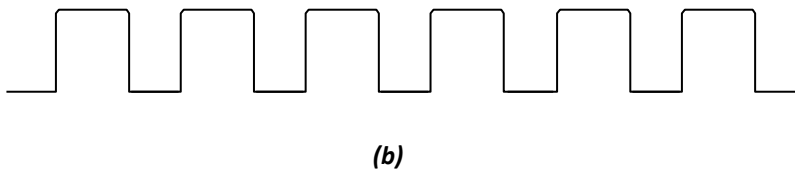
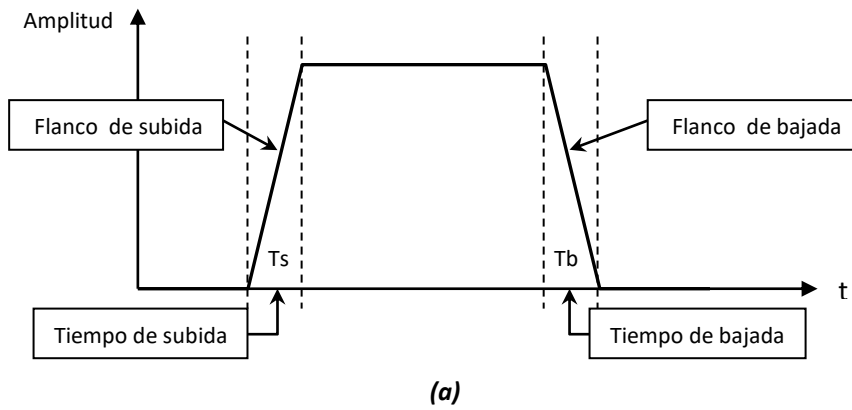


Fig. 10.12

(a) Pulso positivo donde se visualiza los flancos de subida y bajada.

(b) Tren de pulsos ó señales de reloj.

10.3.3. Flip Flop S-R sincronizados por reloj

El Flip Flop sincronizado por reloj tiene un terminal denominado “entrada de reloj” o clock (**Ck**), también se designa como (**CP**), el cual se activa por flanco, ya sea de subida o de bajada. Gráficamente la entrada del reloj se señala mediante un triángulo en el centro de las entradas sincrónicas.

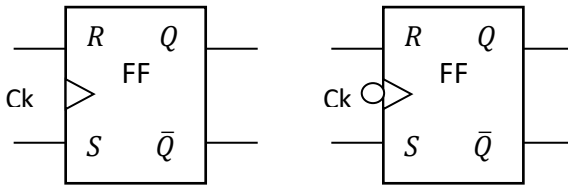


Fig. 10.13

Diagrama de bloques de un Flip Flop con entrada de Reloj, activada con flanco de subida o de bajada.

El Flip Flop S-R, trabaja capturando los niveles lógicos de la entrada en un periodo anterior al “flanco”, para luego disparar la salida Q justo cuando se presenta el flanco de subida o de bajada según lo indique la entrada de reloj del Flip Flop.

Si en la entrada de reloj hay un pequeño círculo, indica que se activa con el flanco de bajada ↓, si no lo tiene, con el flanco de subida ↑

La respuesta de un Flip Flop S-R con compuertas NOR con sincronización de pulsos de reloj que activan por flanco de subida es el siguiente:

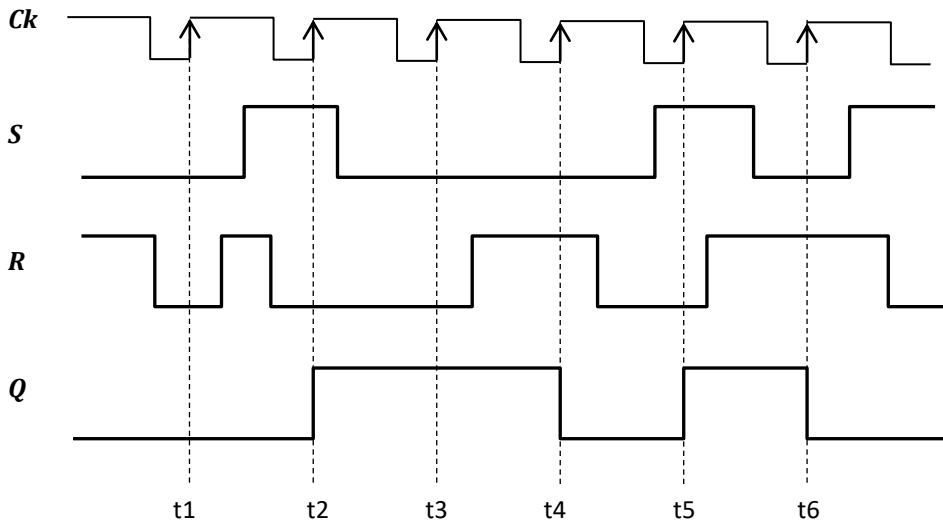


Fig. 10.14

Diagrama de tiempo de un Flip Flop S-R con compuertas NOR. Se asume que inicialmente la Salida Q se encuentra en el nivel BAJO.

IMPORTANTE

S y R pueden estar cambiando en cualquier instante, sin embargo, los niveles lógicos que captura el Flip Flop, son los inmediatamente antes del flanco de subida.

Los niveles de entrada que están prohibidos para el Flip Flop con compuertas NOR, se presenta cuando ambas entradas están en ALTO. En el diagrama se puede observar que en algunos instantes se presenta ésta condición, sin embargo, no es relevante para el sistema mientras no se presente en los flancos de disparo.

10.4. Flip Flop J K

Es aquel biestable que responde a las mismas combinaciones que el Flip Flop S-R a excepción de la combinación 11, en el que el Flip Flop J K, no responde a la prohibición, sino que conmuta (cambia) la salida; en otras palabras el FF JK salva la inconsistencia del FF S-R.

N°	ENTRADA		SALIDA
	J	K	Q_{n+1}
0	0	0	Q_n
1	0	1	0
2	1	0	1
3	1	1	\bar{Q}_n

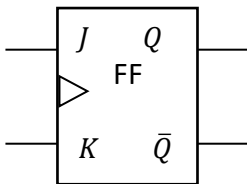


Tabla . 10.2

Tabla de verdad del biestable JK.

Fig. 10.15

Diagrama de bloque de un biestable JK.

10.4.1. FF JK Master Slave

Es un Flip Flop conformado por dos circuitos que se retroalimentan llamados Maestro y Esclavo. La captura de los cambios en los niveles de entrada se realizan en el circuito Maestro durante el pulso positivo (en el nivel activo del reloj), para luego disparar la salida Q en la caída del pulso.

Para evitar cambios indeseables en el proceso de captura de datos, los fabricantes han implementado una variante de FF JK Master Slave llamado "master slave with data lockout".

10.4.2. FF JK edge – triggered

Es aquel Flip Flop que realiza tanto la captura de datos como el disparo de la salida en el flanco de subida o de bajada, según lo indique las características del biestable, obviamente, los datos a capturar deberán estar disponibles antes de la activación del flanco.

10.4.3. Entradas asíncronas en el FF JK

Los Flip Flops pueden disponer básicamente de dos entradas designadas como PRESET (**Pr**), ó también (**S**), y CLEAR (**Clr**) ó (**R**) que no responden a los pulsos de sincronismo sino que la salida del biestable se dispara según la activación de la entrada asincrónica. La activación se realiza en nivel BAJO.

PRESET: pone a Q en nivel ALTO, Q=1.

CLEAR: pone a Q en nivel BAJO, Q=0.

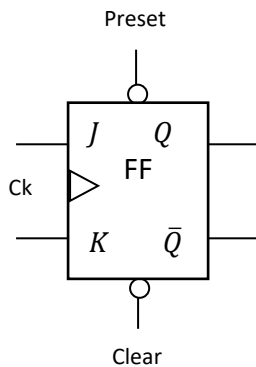


Fig. 10.16

Diagrama de bloque de un Flip Flop JK con Preset y Clear, llamados terminales asíncronicos.

10.5. Flip Flop tipo D

Un tipo especial de Flip Flop JK, es el Flip Flop tipo D, pues entre la entrada J y K se instala una compuerta inversora, quedando como única entrada el terminal J, que se denomina terminal D.

Esta disposición de la entrada restringe a solo dos combinaciones: si D = 0, la salida (Q_{n+1}) será 0, si D=1, (Q_{n+1}) será 1.

N°	SALIDA	
	D	Q_{n+1}
0	0	0
1	1	1

Tabla 10.3

Tabla de verdad del Flip Flop tipo D.

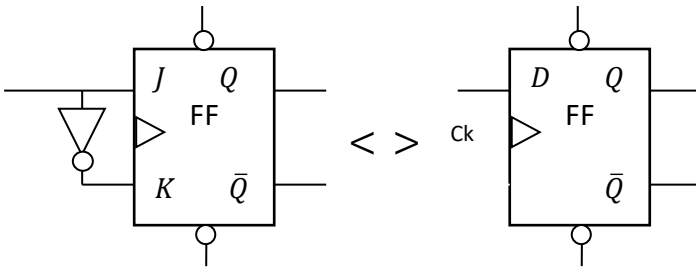


Fig. 10.17

Diagrama de bloque de un Flip Flop tipo D.

10.6. Flip Flop tipo T

Cuando se une o puentea los terminales J y K de un FF JK, tenemos un FF tipo T, pues ambas entradas reciben el mismo dato y activan la salida en función de lo siguiente: si la entrada T=0, la salida (Q_{n+1}) mantendrá su historia previa; mientras que si T=1, (Q_{n+1}) conmutará su nivel lógico anterior.

Como consecuencia de la entrada en ALTO; si dicho terminal se mantiene en 1 mientras los pulsos de reloj se activan en su flanco respectivo, la salida (Q_{n+1}) estará conmutando cada vez que el flanco se dispare. Ésta modalidad de trabajo se aprovecha para generar un divisor de frecuencias.

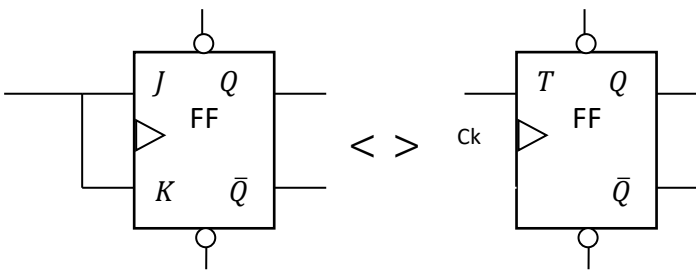


Fig. 10.18

Diagrama de bloque de un Flip Flop tipo T.

Comercialmente no se encuentra Flip Flop tipo T, se puede construir a partir del Flip Flop tipo JK, tal como se observa en la Fig. 10.18.

N°	SALIDA	
	T	Q_{n+1}
0	0	Q_n
1	1	\bar{Q}_n

Tabla 10.4

Tabla de verdad del Flip Flop tipo T.

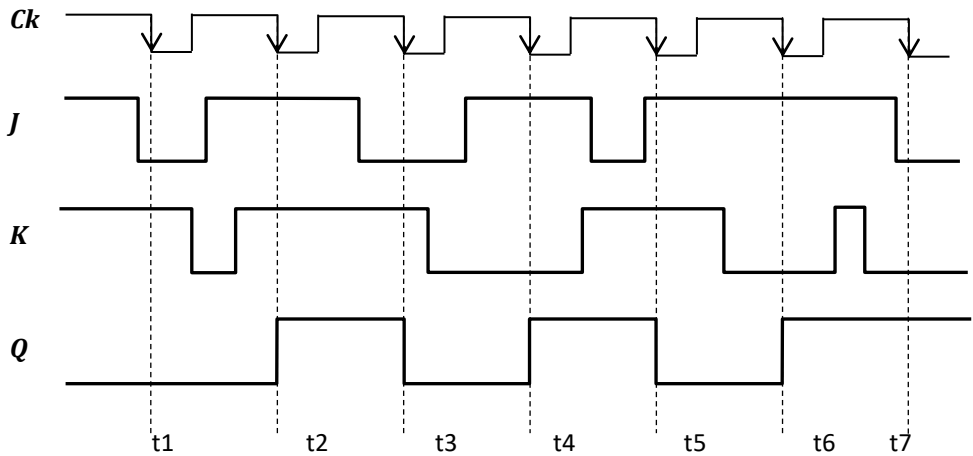


Fig. 10.19
 Diagrama de tiempo de un Flip Flop J-K con compuertas NOR. Se asume que inicialmente la Salida Q se encuentra en el nivel BAJO.

10.7. Circuitos integrados comerciales de Flip Flop

Algunos de los circuitos integrados biestables tipo JK y D son los siguientes:

Tipo JK MASTER SLAVE: 7473, 74H73, 74H76, 74H78.

Tipo JK EDGE TRIGGERED: 74C73, 74LS73, 74C76, 74LS76, 74LS78, 74H103.

Tipo D: 7474(DUAL), 74174(HEX), 74273(OCTAL), 74175(QUAD).

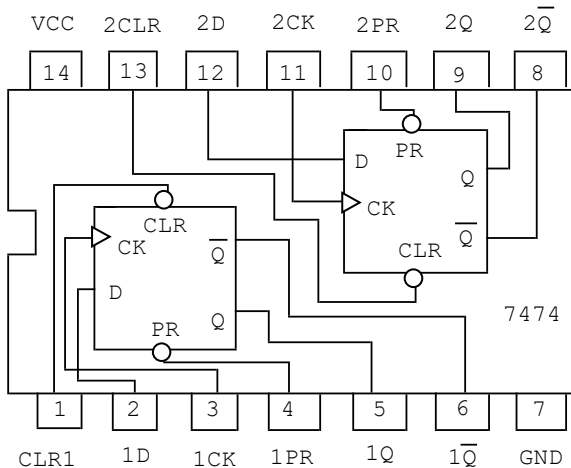


Fig. 10.20

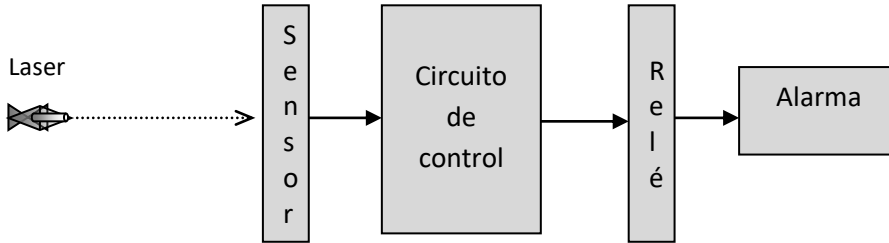
Circuito integrado 7474. Incorpora dos Flip Flop tipo D, con terminales asíncronos de Preset y Clear.

Problemas Resueltos N° 10

1. Implemente un circuito de control para activar una alarma utilizando FF SR, un fototransistor y luz laser.

Solución:

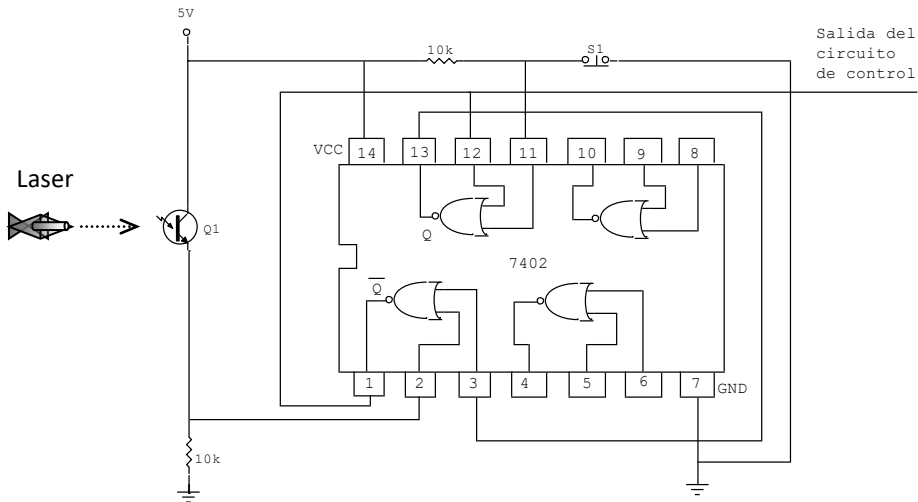
El circuito de control tendrá el siguiente diagrama de bloques.



La luz laser puede provenir de un puntero laser, cuya luz deberá incidir en un fototransistor el cual constituye en el sensor del circuito de control.

Es conveniente que la alarma se encuentre aislado del circuito de control, para ello se utilizará un relé que le permitirá activar el suministro de energía.

Entrada del circuito de control y el control propiamente dicho.

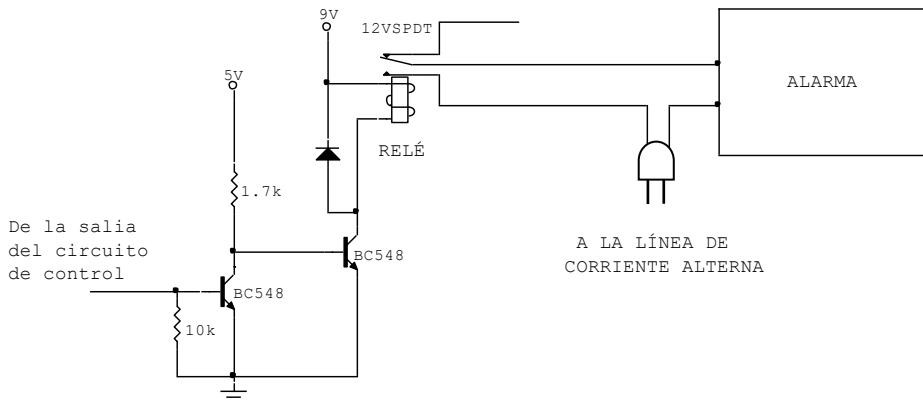


Inicialmente la luz laser incide sobre el fototransistor Q1, el que se satura e ingresa un nivel ALTO en el pin 2 del circuito integrado, luego, se pulsa el tact switch S1 (normalmente cerrado) el cual ingresa a la entrada 11 de la compuerta

NOR un nivel ALTO, por lo que su salida Q (pin 13) se restablece en nivel BAJO, y la salida \bar{Q} (pin 1) se establece en nivel ALTO.

La salida \bar{Q} (pin 1) del circuito de control se mantendrá en nivel ALTO hasta que algo interrumpa la luz laser que incide sobre el fototransistor e inmediatamente éste entra en corte ingresando un nivel BAJO en el terminal 2 del circuito integrado. Esta combinación permite a la salida del circuito de control establecerse en nivel BAJO, el cual coloca en corte al primer transistor BC 548, cuyo voltaje V_{ce} es igual aproximadamente igual a 5 V, permitiendo que el segundo transistor se sature y circule su corriente de colector por el Relé, activando de esta manera la alarma. La desactivación solo se puede realizar pulsando el tact switch S1.

Salida del circuito de control



2. Implemente un circuito de reloj con un periodo de 2 segundos, y 60% de ciclo de trabajo.

Solución:

Usualmente un circuito de reloj implementado con el timer 555 se encuentra en el orden de los kilohertz, sin embargo, el periodo de trabajo de 2 segundos, corresponde a una frecuencia de trabajo muy baja, especialmente para trabajar con dispositivos y componentes electrónicos digitales, de manera que podamos observar los cambios que realizan en las diferentes etapas.

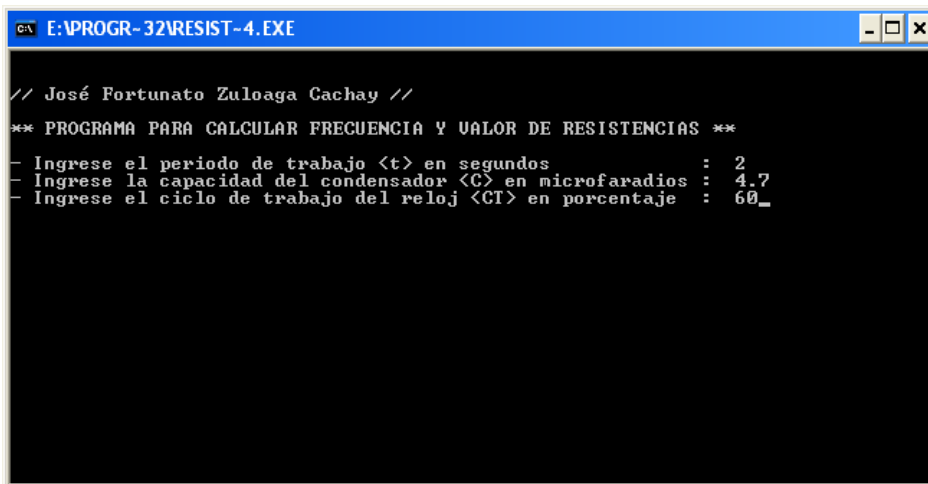
Para calcular valores de resistencia que se conectarán al timer 555, es conveniente utilizar una hoja de cálculo como el Excel, en donde se ingresarán valores del periodo de trabajo (t), la capacidad del condensador (C) y el ciclo de trabajo (CT).

El valor del condensador se asume teniendo en cuenta el periodo de trabajo. La relación entre ambos es directa, es decir, a mayor periodo de trabajo mayor valor del condensador.

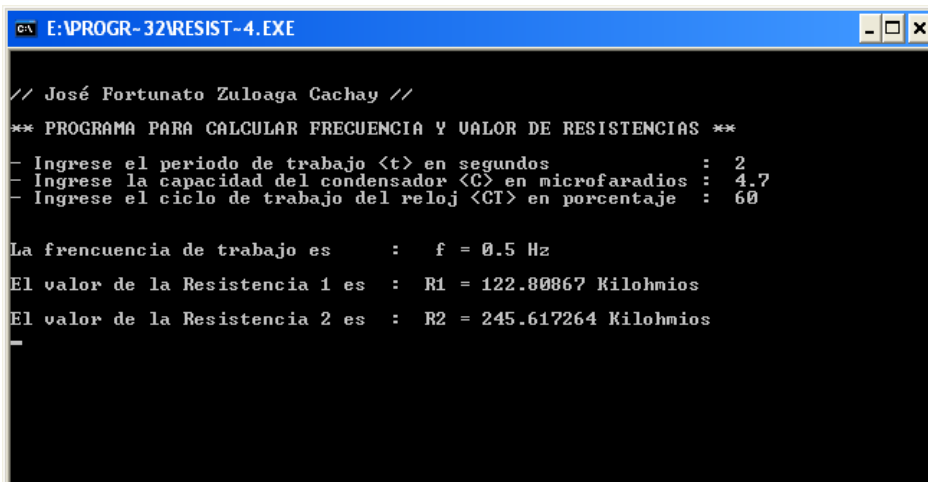
Para baja frecuencia, como el periodo de trabajo entre 1 y 3 segundos, se puede utilizar condensadores entre $1\mu\text{F}$, $4.7\mu\text{F}$ y $10\mu\text{F}$.

Para realizar cálculos repetitivos es conveniente utilizar un lenguaje de programación estructurada como el C, en el que se ingresan los datos propuestos y se obtienen las respuestas deseadas, tal como a continuación se presenta.

Para resolver lo anterior se asume como valor del condensador igual a $4.7\mu\text{F}$.



```
E:\PROGR- 32\RESIST-4.EXE
// José Fortunato Zuloaga Cachay //
** PROGRAMA PARA CALCULAR FRECUENCIA Y VALOR DE RESISTENCIAS **
- Ingrese el periodo de trabajo <t> en segundos : 2
- Ingrese la capacidad del condensador <C> en microfaradios : 4.7
- Ingrese el ciclo de trabajo del reloj <CT> en porcentaje : 60_
```



```
E:\PROGR- 32\RESIST-4.EXE
// José Fortunato Zuloaga Cachay //
** PROGRAMA PARA CALCULAR FRECUENCIA Y VALOR DE RESISTENCIAS **
- Ingrese el periodo de trabajo <t> en segundos : 2
- Ingrese la capacidad del condensador <C> en microfaradios : 4.7
- Ingrese el ciclo de trabajo del reloj <CT> en porcentaje : 60
La frecuencia de trabajo es : f = 0.5 Hz
El valor de la Resistencia 1 es : R1 = 122.80067 Kiloohmios
El valor de la Resistencia 2 es : R2 = 245.617264 Kiloohmios
-
```

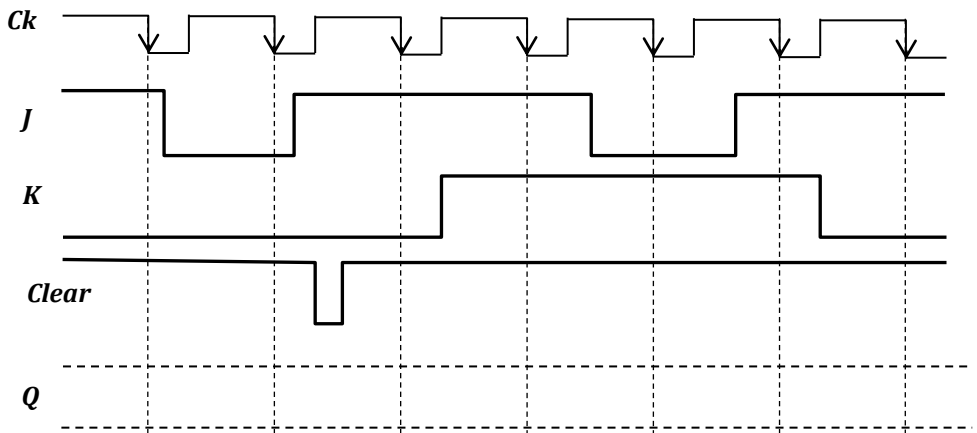
Los valores de las resistencias (tómese como referencia la Fig. 10.9) que se instalará en el Timer 555 son:

$R1 = 120\text{ K}\Omega$

$R2 = 250\text{ K}\Omega$

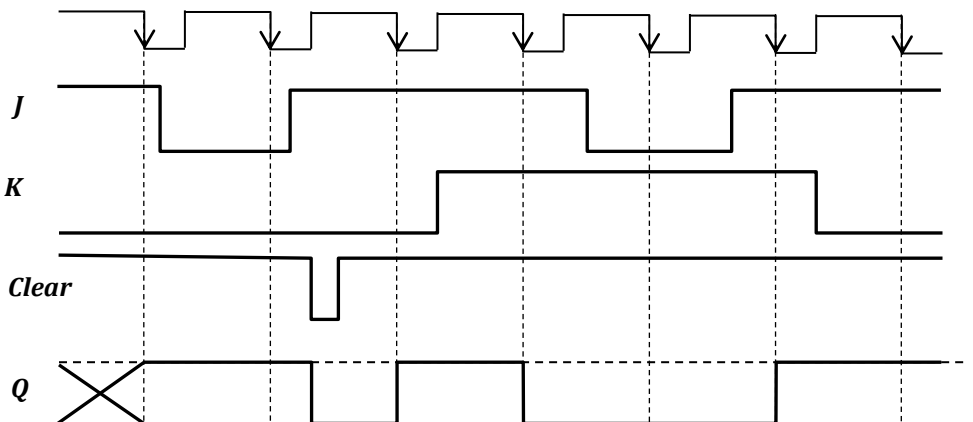
$C = 4.7\text{ }\mu\text{F}$

3. Evaluar la salida Q con las siguientes señales de entrada del FF JK con entrada asíncrona (Clear o Reset)



Solución:

Recordemos que las entradas asíncronas se activan con nivel bajo, y actúan sobre la salida Q, en cualquier momento, no necesariamente en el flanco de activación.



Problemas Propuestos N° 10

1. Implemente un circuito de reloj con una frecuencia de 0.8Khz, y 75% de ciclo de trabajo.

Rpta:

Considerando $0.47\mu\text{F}$ como valor del condensador tenemos:

$$R1 = 1.91\text{K}\Omega \approx 2\text{K}\Omega$$

$$R2 = 0.96\text{K}\Omega \approx 1\text{K}\Omega$$

2. Calcular la frecuencia y el ciclo de trabajo de un circuito de reloj, si disponemos de los siguientes componentes: un condensador de $1\mu\text{F}$, resistencias de $10\text{K}\Omega$ y $50\text{K}\Omega$.

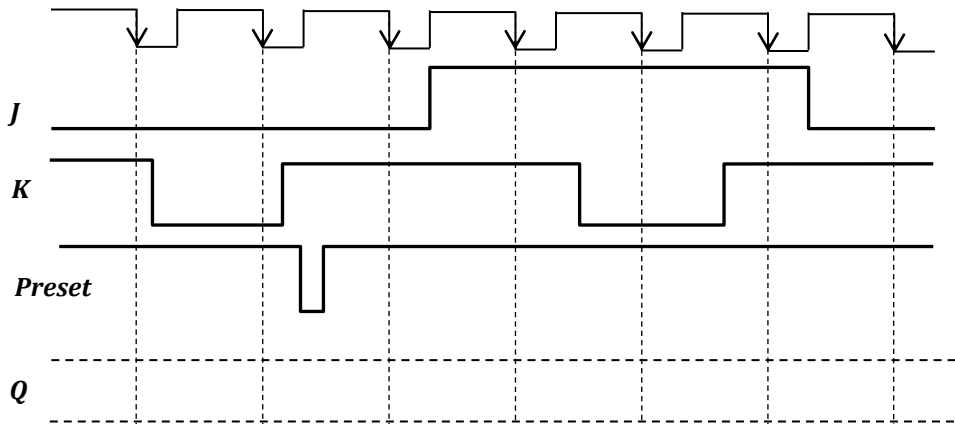
Rpta:

Considerando a $R1 = 10\text{K}\Omega$ y $R2 = 50\text{K}\Omega$, tenemos:

$$f = 13.1 \text{ Hz}$$

$$CT = 54.5\%$$

3. Evaluar la salida Q con las siguientes señales de entrada del FF JK con entrada asíncrona (Preset).



Capítulo 11

Almacenamiento y transferencia de datos

11.1. Registros

La característica principal de los circuitos secuenciales es la utilización de un sistema de almacenamiento de datos o información.

El biestable o Flip Flop, es el elemento que tiene la propiedad de guardar un bit en memoria mediante una combinación adecuada de sus entradas, por lo que agrupados pueden almacenar información por varios bits; pues como se sabe, los sistemas digitales necesitan un almacenamiento normalmente en bytes (conjunto de ocho bits).

Los dispositivos que guardan información en grupos de varios Flip Flop tipo D, se llaman “registros”. Los FF tipo D, se conectan en cascada enlazando la salida de uno con la entrada del otro, y de esta manera se forma un array o conjunto de bits almacenados.

A diferencia de la memoria de gran almacenamiento de información en forma temporal o permanente, los registros se caracterizan por contener unos pocos elementos de almacenamiento, es decir, su capacidad no excede de los dos bytes, cuyos datos los guarda normalmente en forma temporal.

Los microcontroladores así como los microprocesadores cuentan con registros especiales para el almacenamiento temporal de sus datos. Uno de estos registros especiales se llama Acumulador o también denominado registro de trabajo, cuya función es fundamental en el procesamiento de información que realiza la Unidad Aritmético Lógico (ALU).

11.2. Registros de desplazamiento

Cuando la información que ingresa a un registro se desplaza de alguna forma, se conoce como Registro de desplazamiento; aunque la propiedad de los registros es la transferencia de los datos o información a otros registros temporalmente o la transferencia a la memoria principal del sistema digital.

Es preciso aclarar, que un sistema de información recibe en su entrada un conjunto de datos, constituidos por bits de un código numérico o alfanumérico, los cuales son procesados y luego exteriorizados a través de la salida en forma de información; por tanto, se entiende que la información son datos procesados.

Por el hecho mismo que el registro transfiere información, los datos que ingresan a él pueden deberse a la información que sale de otro registro; por ello, en este caso, no distinguimos claramente la diferencia entre datos e información.

El desplazamiento que pueden realizar los datos en los registros, está en función del formato de entrada y salida del registro.

Los datos pueden ingresar en serie y salir en serie.

Los datos pueden ingresar en serie y salir en paralelo.

Los datos pueden ingresar en paralelo y salir en serie.

Los datos pueden ingresar en paralelo y salir en paralelo.

11.2.1. Registro de desplazamiento: entrada y salida en serie

En este registro la información entra en serie, se desplaza en serie y luego en la salida la información sale en serie. Los datos pueden ingresar un bit a la vez, es decir, si la información consta de 8 bits, deberá ingresar ocho veces.

Los datos en forma de bits ingresan por un solo terminal de entrada, denominado IN, luego se transfiere al siguiente Flip Flop hasta que dicho bit aparezca en la salida llamada OUT.

La operación de entrada y salida de datos es sincronizada por un reloj (clock) el cual enlaza en paralelo a todos los Flip Flop del registro a través de CK ó (CP). Con cada pulso de reloj el bit que ingresa por IN se desplaza al siguiente y así sucesivamente hasta que el primer bit que ingresó aparezca en la salida. Si el dato a ingresar tiene 4 bits, entonces, el reloj deberá emitir 4 pulsos para que la información se guarde en el registro; si emite más de 4 pulsos la información se perderá si es que no se logra almacenar en otro registro.

Cuando por gestión del microprocesador, el dato almacenado en el registro es solicitado por otro registro o dispositivo, el reloj deberá emitir los pulsos necesarios para desplazar dicho dato. Por ejemplo, si el registro tiene almacenado cuatro bits, el

reloj deberá producir cuatro pulsos para desplazarlo hacia el otro registro o dispositivo de almacenamiento.

Implementar un registro de 4 bits

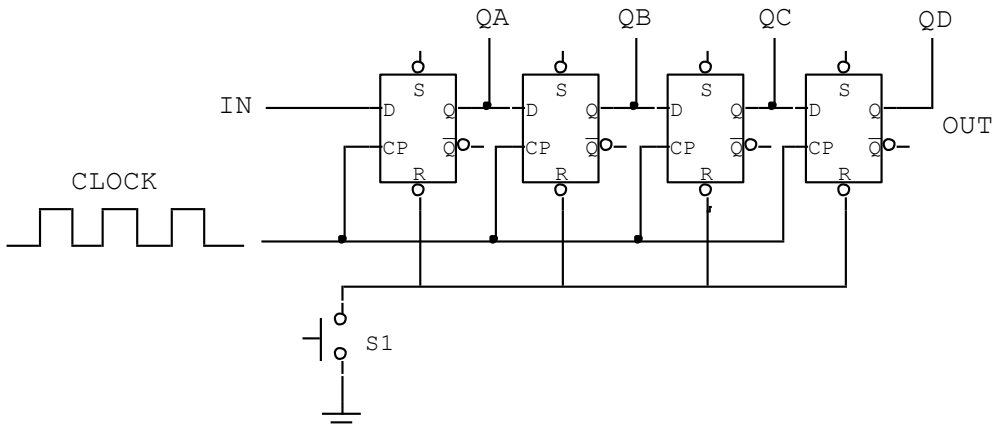


Fig. 11.1

Registro de desplazamiento de cuatro bits de izquierda a derecha.

Inicialmente se presiona el pulsador S1 (se activa con nivel BAJO) y todas las salidas se restablecen en nivel BAJO ó 0 “se resetean”. El pulsador es un tact switch normalmente abierto.

El proceso de entrada de datos o escritura se realiza de la siguiente manera:

- El registro se borra (todos los FF en cero).
- Los datos van a entrar en serie por IN.
- Con cada pulso de reloj entra un bit a un FF y el anterior bit se desplaza.
- En el caso anterior requiere 4 pulsos de reloj porque necesita desplazar 4 bits.

IMPORTANTE

El registro de la Fig. 11.1, es un registro que desplaza los datos hacia la derecha, pues los datos ingresan por IN y aparecen por OUT; sin embargo, se pueden desplazar los datos de derecha a izquierda ingresando dichos datos por OUT y la salida por IN.

La señal del Clock proviene de la salida de un multivibrador astable como el timer 555 con una frecuencia determinada. Esta frecuencia debe ser tal que permita la retención o captura de datos inmediatamente antes que se produzca el disparo por flanco, ya que siempre en estos arreglos se produce un “retardo de propagación”.

Si la entrada de reloj (CP) no tiene una pequeña circunferencia en la entrada, quiere decir que el disparo se realiza en el flanco de subida.

Las entradas de restablecimiento de los biestables (clear) se enlazan entre sí en los terminales asincrónicos Clr ó (R). con la finalidad de resetear o poner en 0 a todos los Flip Flops del registro.

INGRESO DE DATOS

Ingresa el dato binario 1011 en un registro de desplazamiento SERIE – SERIE, de izquierda a derecha como el de la Fig. 11.1. Visualizar dicho desplazamiento mediante diagrama de tiempo.

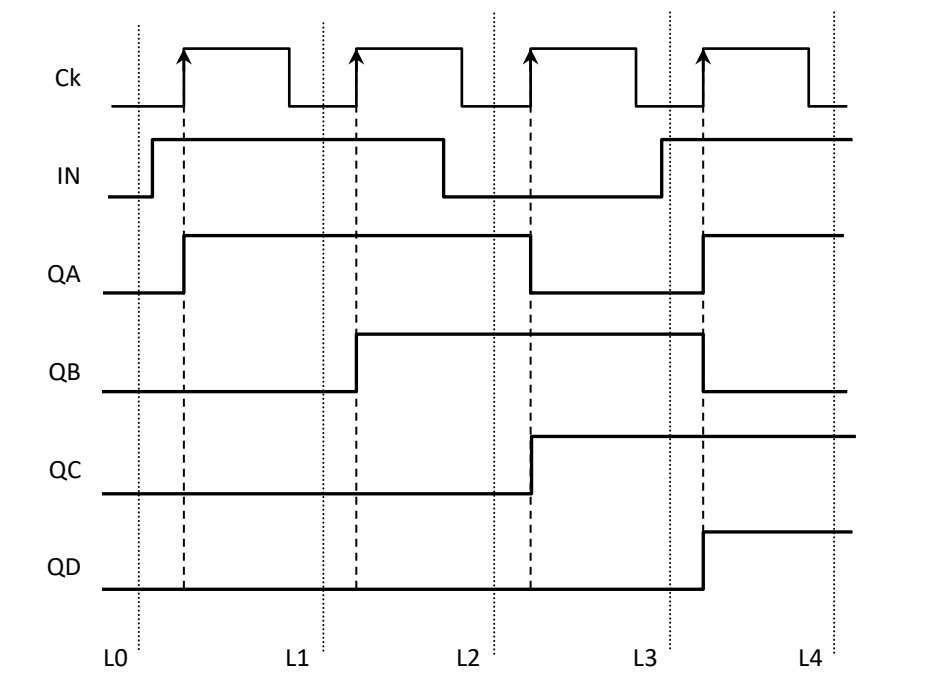


Fig. 11.2

Diagrama de tiempo de un registro SERIE-SERIE de cuatro bits en el que ingresa el dato 1011.

La lectura del diagrama de tiempo se realiza en las líneas punteadas L0, L1, L2, L3 y L4, básicamente en las salidas de los Flip Flops QA, QB, QC y QD.

	QA	QB	QC	QD
L0	0	0	0	0
L1	1	0	0	0
L2	1	1	0	0
L3	0	1	1	0
L4	1	0	1	1

Tabla 11.1

Lectura de salidas de cada uno de los Flip Flops del Registro de desplazamiento.

Se observa en la lectura L0 que las salidas de los Flip Flops están reseteados, ya que inicialmente se han activado los clear de cada biestable.

En la lectura L1, QA responde a la entrada IN en el primer pulso, las demás salidas se mantienen en nivel BAJO.

En la lectura L2, QA responde a IN, mientras que QB responde a QA en el segundo pulso; QC y QD se mantienen en BAJO.

En la lectura L3, QA responde a IN, QB responde a QA, mientras que QC responde a QB en el tercer pulso; QD se mantiene en BAJO.

En la lectura L4, QA responde a IN, QB a QA, QC a QB y QD a QC, en el cuarto y último pulso. Si visualizamos la salida QA QB QC QD verificaremos que se trata del dato de entrada, es decir, 1 0 1 1.

DESPLAZAMIENTO DE DERECHA A IZQUIERDA

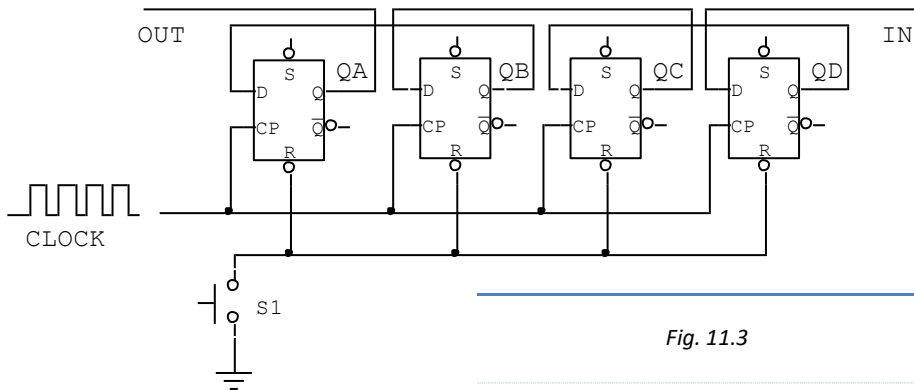


Fig. 11.3

Registro de desplazamiento de cuatro bits, de derecha a izquierda.

CONTROL DE LA DIRECCIÓN DEL DESPLAZAMIENTO

Para diseñar un circuito combinacional llamado “control de dirección”, se tendrá en cuenta tres entradas y una salida. Dos de las entradas corresponderán a las salidas de dos Flip Flops; y la salida corresponderá a la entrada de un Flip Flop, tal como se detalla a continuación.

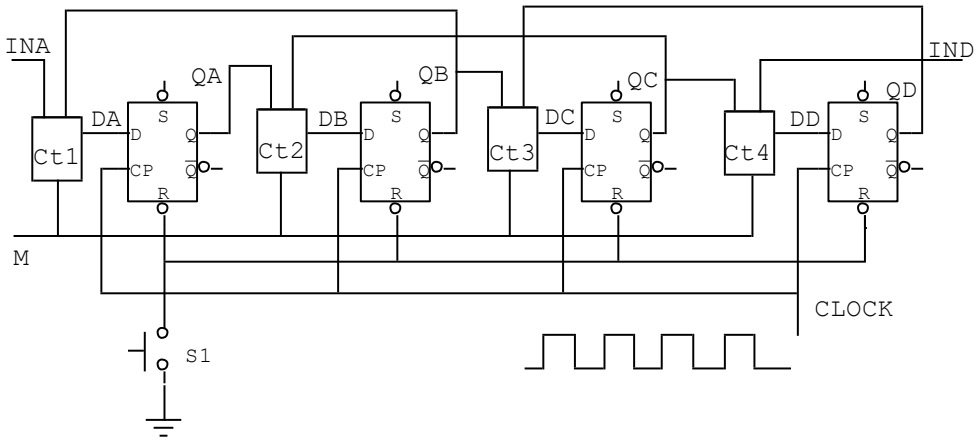


Fig. 11.4

Registro de desplazamiento de cuatro bits, con control de dirección.

Control 1 (Ct1): Tiene como entradas a INA, QB y un habilitador de dirección designado como “M”; y una salida DA.

Control 2 (Ct2): Tiene como entradas a QA, QC y un habilitador de dirección designado como “M”; y una salida DB.

Control 3 (Ct3): Tiene como entradas a QB, QD y un habilitador de dirección designado como “M”; y una salida DC.

Control 4 (Ct4): Tiene como entradas a QC, IND y un habilitador de dirección designado como “M”; y una salida DD.

OBSERVACIÓN

El habilitador de dirección “M” de cada bloque combinacional se enlazan en uno solo.

Si $M=0$, el registro desplazará datos de izquierda a derecha; si $M = 1$, desplazará datos de derecha a izquierda. Criterio adoptado discrecionalmente.

DISEÑO DEL CONTROL 2

Todos los demás controles tienen la misma estructura lógica del Control 2, sólo deberá tenerse en cuenta sus correspondientes entradas y salidas.

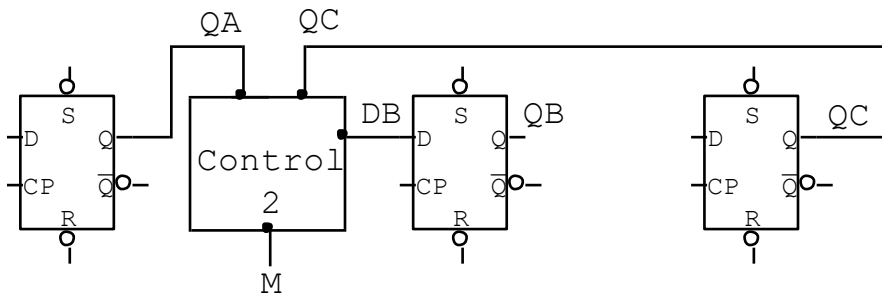


Fig. 11.5

Circuito combinacional (Control 2) para direccionar los datos de izquierda a derecha o de derecha a izquierda mediante el habilitador de dirección "M".

Construimos una tabla de verdad con tres entradas (M , QA y QC) y una salida DB .

N°	INGRESO			SALIDA
	M	QA	QC	DB
0	0	0	0	0
1	0	0	1	0
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1

Tabla 11.2

Tabla de verdad de un control de dirección de datos

Se utiliza un mapa de Karnaugh para hallar la función lógica DB y luego se implementa mediante diagrama lógico.

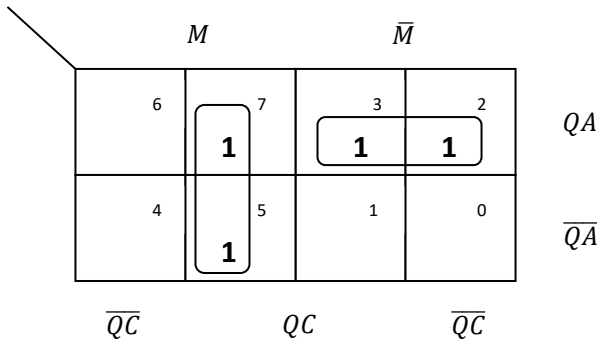


Fig. 11.6

Mapa de Karnaugh para hallar la función lógica del control de dirección de datos.

Hallando la función lógica DB.

$$DB = M.QC + \bar{M}.QA$$

$$DB = \overline{\overline{M.QC} + \overline{\bar{M}.QA}}$$

$$DB = \overline{\overline{M.QC} . \overline{\bar{M}.QA}}$$

Implementando la función lógica DB, mediante compuertas lógicas NAND

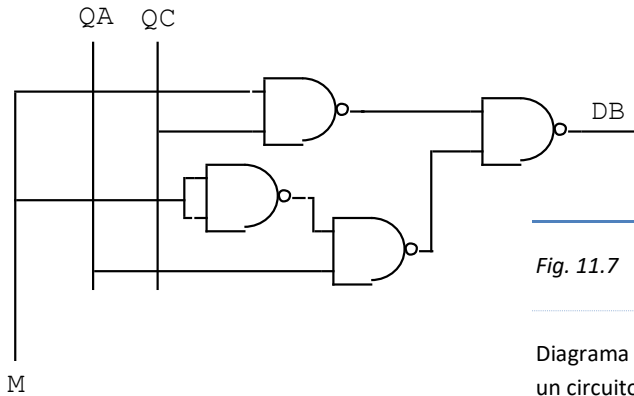


Fig. 11.7

Diagrama lógico con compuertas NAND de un circuito combinacional de control de direccionamiento de datos hacia la izquierda o hacia la derecha.

11.2.2. Registro de entrada paralelo y salida serie

Para ingresar datos en paralelo se utiliza la entrada asincrónica PRESET (Pr) o (S), en el que se aplica la salida de una compuerta NAND, cuyas entradas son: el Habilitador (H) y la entrada de dato.

Si el habilitador (H) está en BAJO, $H=0$, la entrada de datos (A, B, C y D) se inhabilita y no ingresan a los respectivos Flip Flops. Si $H=1$, los datos se habilitan e ingresan a los Flip Flops; véase al respecto en la sección 6.3.1.

INGRESO DE DATOS

Para ingresar datos en paralelo en el registro de desplazamiento, se deberá resetear las salidas de los Flip Flops mediante el switch S1.

Ingresar los datos A, B, C y D en sus respectivos terminales mientras H se encuentre en nivel BAJO.

Colocar H en nivel ALTO, para que los datos pasen a los respectivos Flip Flops. Se deberá tener en cuenta que las entradas asincrónicas (Pr) o (S) se activan en nivel BAJO y no requieren ningún pulso de reloj.

Cuando los datos A, B, C, y D ya se han registrado como (QA, QB, QC y QD), requieren cuatro pulsos de reloj para que dichos datos se desplacen en serie hacia otro registro o dispositivo de almacenamiento.

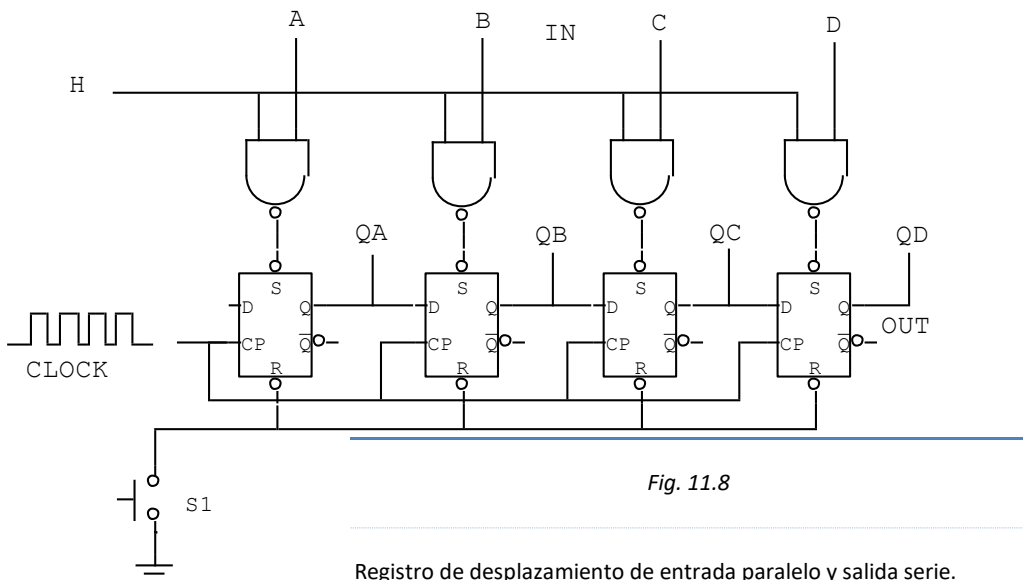


Fig. 11.8

Registro de desplazamiento de entrada paralelo y salida serie.

11.2.3. Registro de entrada serie y salida paralelo

En este tipo de registro, los bits de datos ingresan por el terminal IN en cuatro pulsos de reloj; luego de dichos pulsos, los datos se registran en sus respectivos Flip Flops.

Para desplazar en paralelo los datos registrados, se utilizan los terminales asincrónicos de un registro en el que se desea guardar la información. Para proceder de la manera descrita anteriormente, es necesario que al inicio de la operación se reseteen ambos registros.

IMPORTANTE:

Es necesario tener en cuenta que el registro de desplazamiento serie utiliza pulsos de reloj para almacenar los datos, mientras que la transferencia de datos en paralelo no lo requiere, pues hace uso del terminal asincrónico Preset (Pr) o (S).

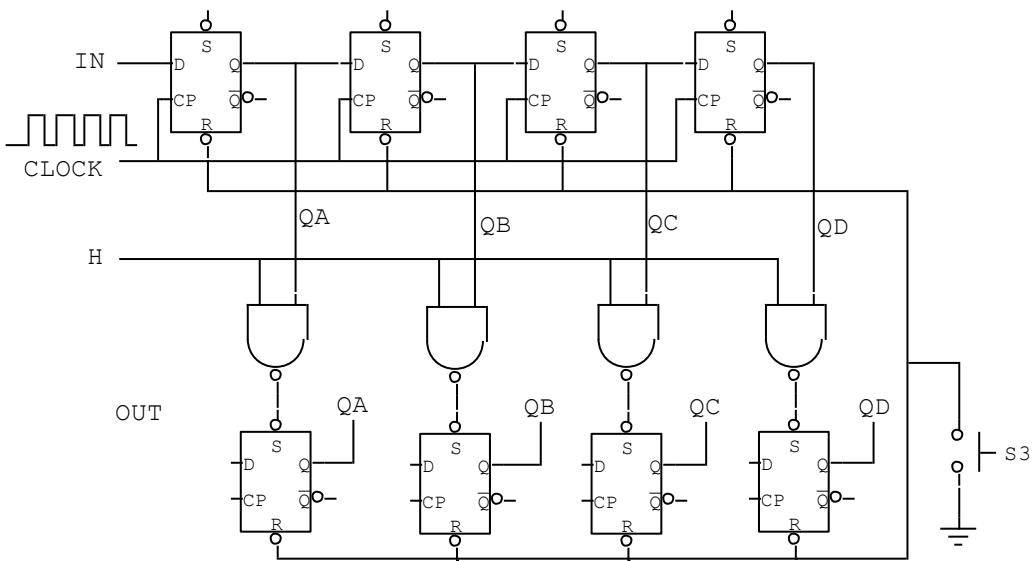


Fig. 11.9

Registro de desplazamiento de entrada serie y salida paralelo.

11.2.4. Registro de entrada paralelo y salida paralelo

El registro de entrada paralelo y salida paralelo utilizan las entradas asincrónicas Preset (Pr) o (S), y no requieren pulsos de reloj para efectuar dicha operación.

Para que funcione adecuadamente dicha transferencia de datos, es imprescindible que los registros se restablezcan en nivel BAJO, mediante el switch S1.

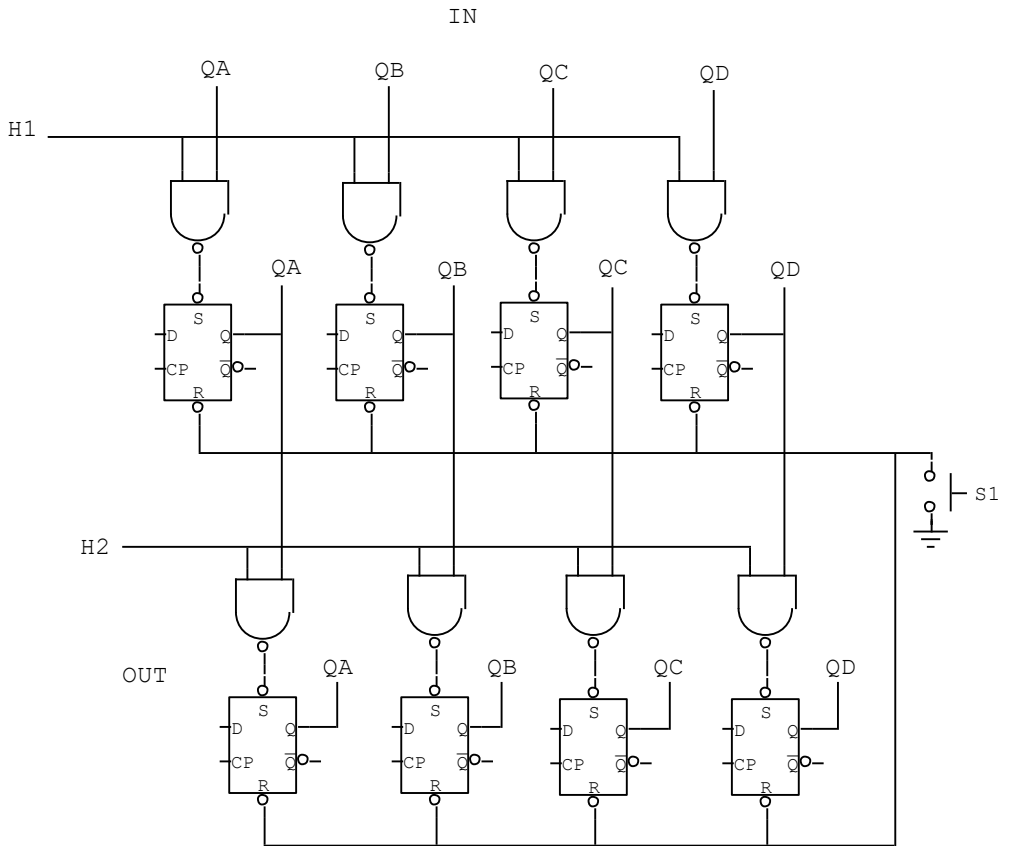


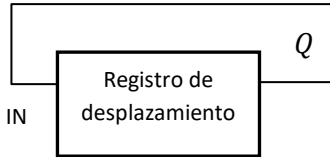
Fig. 11.10

Registro de desplazamiento de entrada paralelo y salida paralelo.

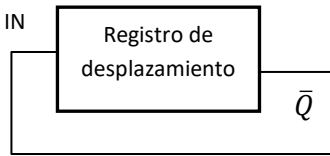
11.3. Registro de desplazamiento en lazo cerrado

Los registros en lazo cerrado se obtienen enlazando la salida (OUT) del último Flip Flop con la entrada (IN).

Si la entrada (IN) del registro de desplazamiento recibe los datos Q del último Flip Flop, recibe el nombre de **Contador anillo**; si recibe los datos \bar{Q} del último Flip Flop, recibe el nombre de **Contador Johnson**.



(a)



(b)

Fig. 11.11

(a) Contador anillo.

(b) Contador Johnson.

CONTADOR ANILLO DE CUATRO BITS

QA	QB	QC	QD
0	0	0	1
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

Fig. 11.12

Desplazamiento de bits del contador anillo. El nivel ALTO de QD se obtiene estableciendo en nivel ALTO el último Flip Flop de la derecha mediante un pulso en BAJO aplicado en el terminal asincrónico PRESET antes de inicial la operación.

CONTADOR JOHNSON DE CUATRO BITS

QA	QB	QC	QD
0	0	0	0
1	0	0	0
1	1	0	0
1	1	1	0
1	1	1	1
0	1	1	1
0	0	1	1
0	0	0	1
0	0	0	0

Fig. 11.13

Desplazamiento de bits en un contador Johnson. El nivel ALTO en QA se produce porque en la entrada del Primer Flip Flop se aplica la salida en ALTO de \bar{QD} . Inicialmente el registro se resetea tal como se observa en la primera fila.

11.4. Registros comerciales 74164, 74165, 74166, 74194

Existen diversos registros de desplazamiento comerciales que efectúan transferencias en uno o más formatos de entrada y salida de datos.

El circuito integrado 74164, es un Registro de desplazamiento de 8 bits, de entrada serial y salida paralelo.

El circuito integrado 74165, es un Registro de desplazamiento de 8 bits, de entrada paralelo y salida serial.

El circuito integrado 74166, es un Registro de desplazamiento de 8 bits, de entrada serial o paralelo y salida serial.

El circuito integrado 74194, es un Registro de desplazamiento de 4 bits, que permite todos los formatos de entrada y salida; y además en el desplazamiento serie es bidireccional, es decir, desplaza los datos a la izquierda o hacia la derecha.

La configuración del formato de entrada y salida de datos se realiza mediante dos terminales denominados S0 y S1.

S0	S1	Formato de entrada y salida de datos
0	0	Inhabilitación del desplazamiento
0	1	Desplazamiento de derecha a izquierda
1	0	Desplazamiento de izquierda a derecha
1	1	Carga de datos en paralelo

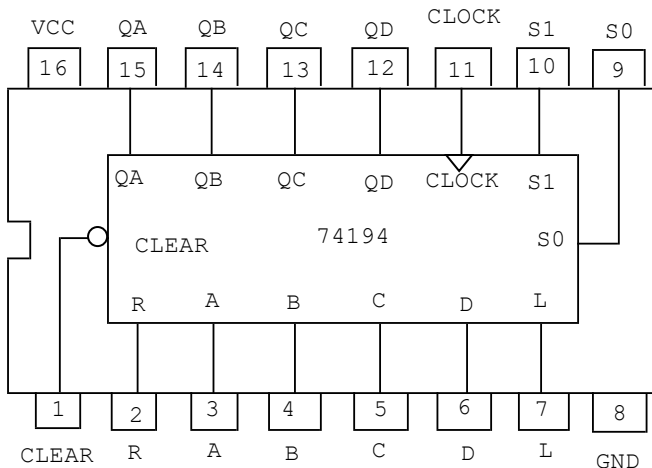


Fig. 11.14

Circuito Integrado 74194. Registro de desplazamiento de cuatro bits.

DESCRIPCIÓN DE TERMINALES DEL 74194

Entrada.

- Pin N° 2: designado como R; entrada serial de desplazamiento hacia la derecha.
- Pin N° 7: designado como L; entrada serial de desplazamiento hacia la izquierda.
- Pines N° (3-6): designados como (A, B, C, y D); entrada en paralelo.

Salida:

- Pines N° (12-15): designados como (QD, QC, QB y QA); salida serial o paralelo.

Reseteo:

- Pin N° 1: designado como CLEAR; se activa con un pulso negativo.

Control:

- Pines N° 9 y 10: designados como S0 y S1; control de formato de entrada y salida de datos.

Reloj:

- Pin N° 11: designado como CLOCK; entrada se señal de reloj. Las entradas de control deberán activarse cuando la señal de reloj esté en ALTO.

11.5. Bus de datos

Son aquellas líneas por donde ingresan o salen datos, es decir, son aquellos caminos por donde se desplazan los datos.

El bus de datos puede contener una o más líneas de transmisión. Si el bus de datos tiene una línea de datos, entonces, los bits entrarán o saldrán de uno en uno; a éste tipo de bus se conoce como "bus serial".

Los sistemas secuenciales normalmente utilizan varias líneas de datos, cuya transmisión lo realiza en paralelo, tanto de entrada como de salida.

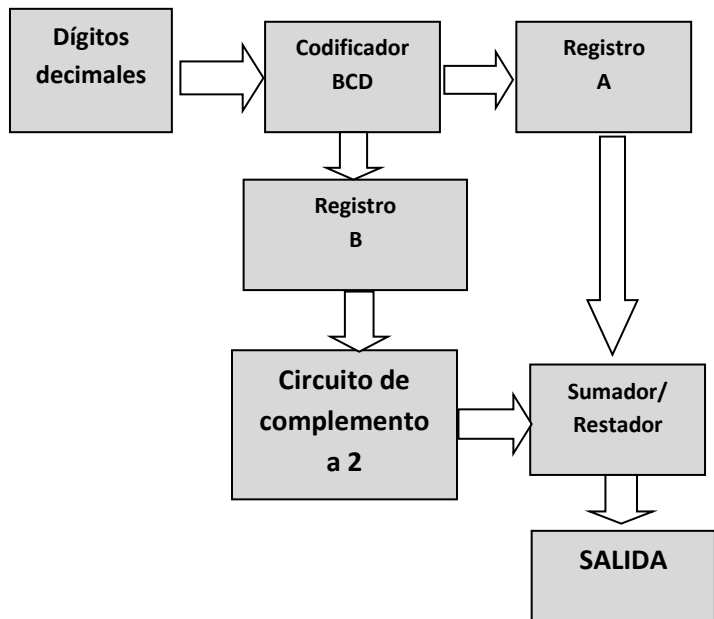
Problemas Resueltos N° 11

1. Implementar un circuito sumador restador de 4 bits de datos.

Solución:

Para la implementación de un circuito sumador restador, es conveniente utilizar registro de datos para almacenar el número ingresado por un codificador de prioridad, luego, por medio de bits seleccionar el registro deseado y seleccionar el tipo de operación a realizar.

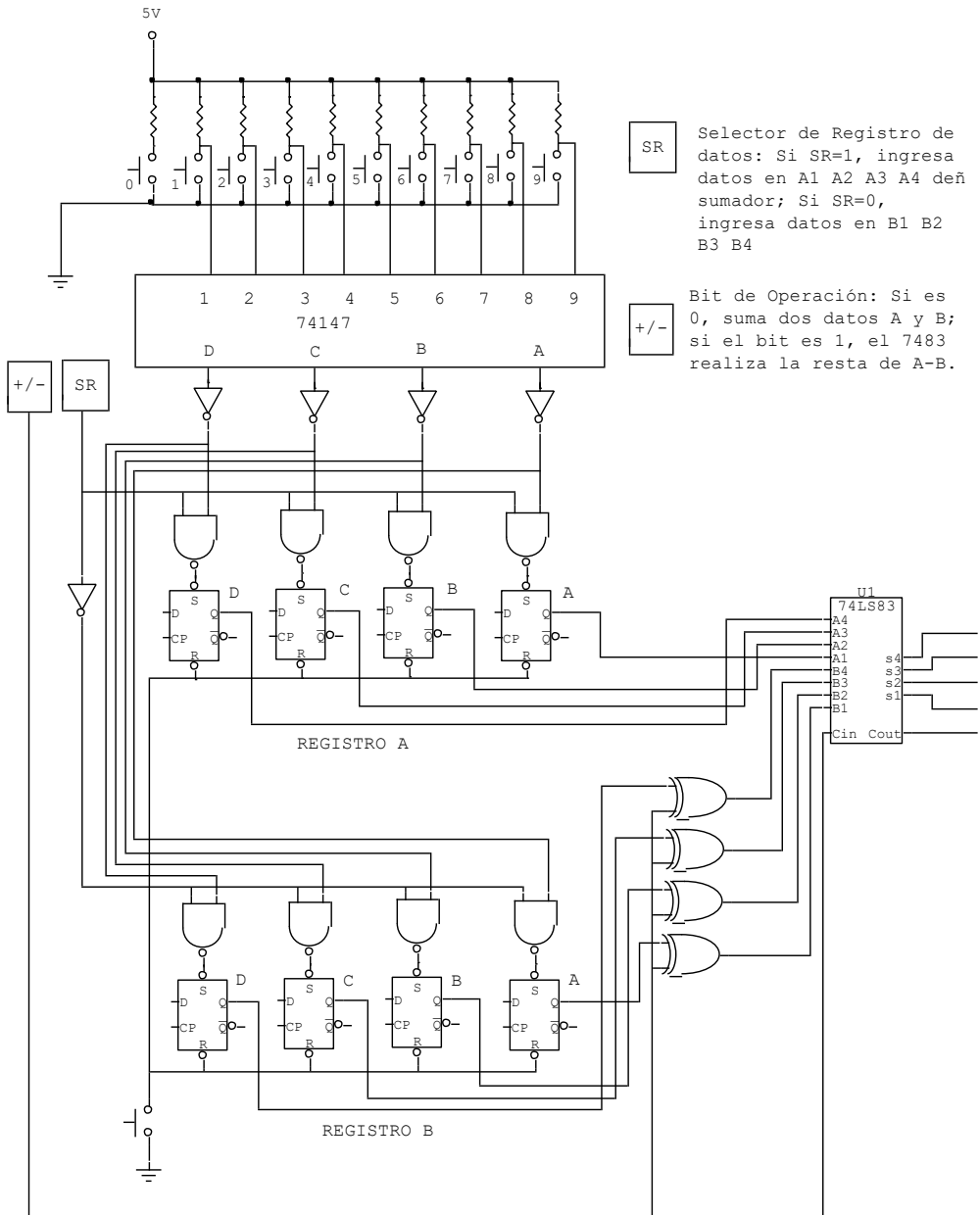
El diagrama de bloques adecuado puede ser el siguiente:



El sumador restador debe contar con un circuito de complemento a 2 para gestionar el tipo de operación que realizará el 7483.

El circuito cuenta con dos bits de control:

El Bit selector de registro y el selector de operación, se operan por separado teniendo en cuenta que el almacenamiento de datos en los registros deben previamente resetearse antes de ingresar los datos de operación.



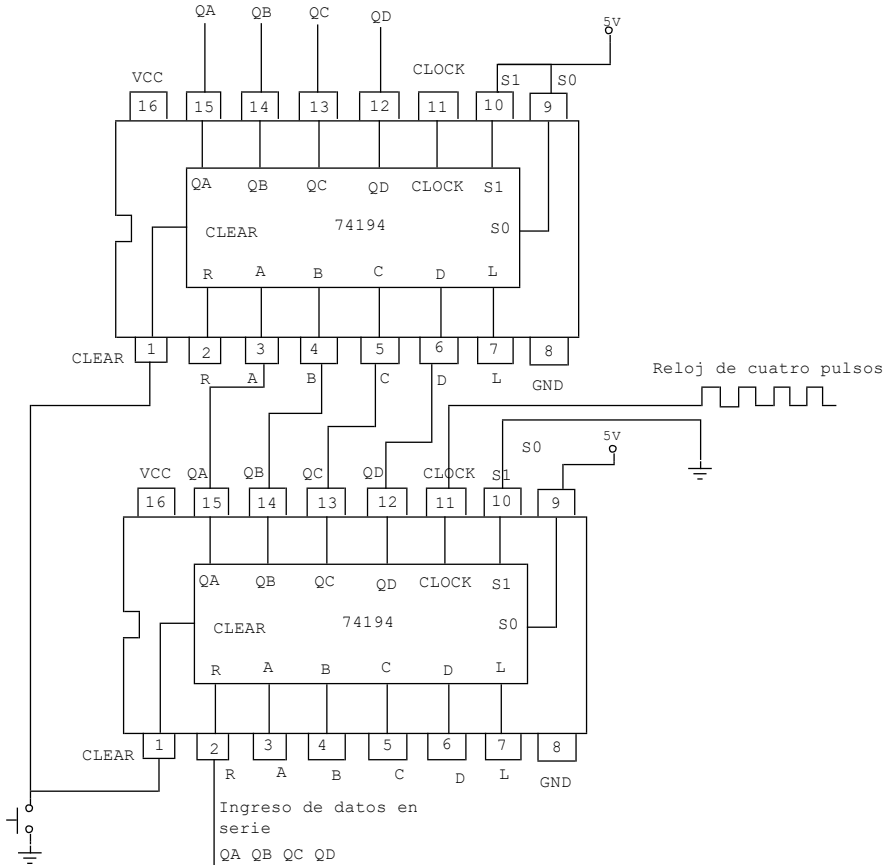
SR Selector de Registro de datos: Si SR=1, ingresa datos en A1 A2 A3 A4 deñ sumador; Si SR=0, ingresa datos en B1 B2 B3 B4

+/- Bit de Operación: Si es 0, suma dos datos A y B; si el bit es 1, el 7483 realiza la resta de A-B.

- Implemente con dos registros 74194, una carga de datos en serie y luego la transferencia de dichos datos en paralelo

Solución:

La implementación constará de dos etapas, en la primera de ellas ingresará el dato en serie, y la salida se transfiere en paralelo.



Problemas Propuestos Nº 11

- Diseñe e implemente un circuito sumador restador cuyo ingreso y transferencia de datos se realice en forma serial.

Sugerencia: En lugar del sumador 7483, puede utilizar un circuito sumador completo de la fig. 9.4, dado que el 7483 es un sumador de datos en paralelo y no es útil para los requerimientos de la pregunta.

Capítulo 12

Contadores

12.1. Introducción

Los contadores son circuitos secuenciales cuya salida refleja el número de pulsos de reloj aplicados en su entrada.



Fig. 12.1

Diagrama de bloques de un contador digital, cuya entrada recibe pulsos de reloj y su salida refleja el número de pulsos aplicados

En la vida cotidiana es común observar y utilizar contadores digitales, ya sea que cuenten ascendente o descendentemente. La mayoría de éstos dispositivos se utilizan en equipos electrodomésticos como: lavadoras automáticas, hornos microondas, equipos de sonido, televisores, etc.

Normalmente, los contadores se programan para que cuenten la cantidad de segundos, minutos u horas que deberá funcionar o apagarse el equipo.

Existen otro tipo de contadores que cuenta la cantidad de eventos que se producen, tales como el velocímetro digital de un automóvil ó los contadores de objetos de una determinada tienda ú oficina.

La estructura interna de un sistema de cómputo dispone de un contador de programa para gestionar el procesamiento de un programa computacional.

La estructura interna de los contadores está constituida por Flip Flops tipo T, los cuales se implementan uniendo las entradas de los Flip Flops JK.

12.2. Clasificación de los contadores

La clasificación de los contadores se realiza en función de la aplicación de los pulsos de reloj en los Flip Flops.

Si la aplicación de dichos pulsos se hace en el primer Flip Flop y los siguientes utilizan los pulsos de las salidas Q de los Flip Flop que lo preceden, se conoce como contadores asíncronos; sin embargo, si los pulsos de reloj se aplican a todos los Flip Flops del contador en forma simultánea, se conoce como contadores síncronos.

Los contadores asíncronos pueden ser:

- Binario
- Década
- Módulo N
- Up/Down

Los contadores síncronos a su vez pueden ser:

- Binario
- Década
- Módulo N
- Reversible
- Programable

12.2.1. Contadores asíncronos de rizo

Los contadores asíncronos tipo rizo se caracterizan por su sencillez y fácil aplicabilidad. Su construcción se realiza mediante el enlace de un conjunto de Flip Flops tipo T. Sin embargo, la limitación de este tipo de contador es su velocidad de operación, pues, sólo el primer Flip Flop del contador recibe la señal de reloj, los siguientes Flip Flops reciben señal para sus respectivas entradas de reloj de las salidas Q ó \bar{Q} de los Flip Flops anteriores, tal como se aprecia en la Fig. 12.2..

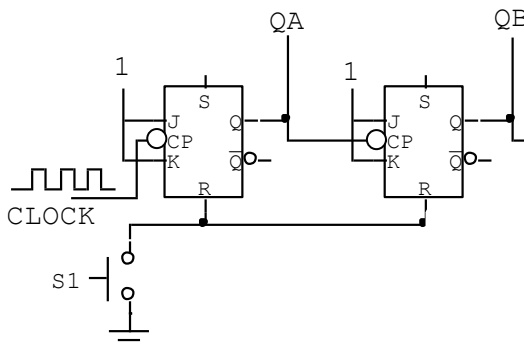


Fig. 12.2

Contador asíncrono de rizo de dos bits. Nótese que el Flip Flop A recibe la señal de reloj (clock) en CP, mientras que el Flip Flop B, recibe señal de reloj de la salida QA.

La configuración anterior acarrea una serie de eventos en las señales de salida de cada uno de los Flip Flops del contador, tal como se observa en la figura 12.3. cuyas salidas QA y QB inicialmente se encuentran en nivel bajo por efecto de S1 (Fig. 12.2). La respuesta de los Flip Flops se produce en los flancos de bajada de la señal de reloj (clock) para el primer Flip Flop y en el flanco de bajada de la señal de QA para el segundo Flip Flop.

Sin embargo, por el retardo de propagación que sufre la señal de salida de un Flip Flop, QA responde a la señal de reloj (Clock) en un lapso de tiempo entre t_1 y t_2 , asimismo, el flanco de bajada de QA ocurre entre t_3 y t_4 . Este evento determina que la señal de QB responda en un tiempo entre t_4 y t_5 .

Mientras exista suficiente tiempo entre los flancos de bajada del Clock para que las señales de salida de los Flip Flops respondan a sus respectivas señales de reloj, no hay problema relevante en el funcionamiento del contador.

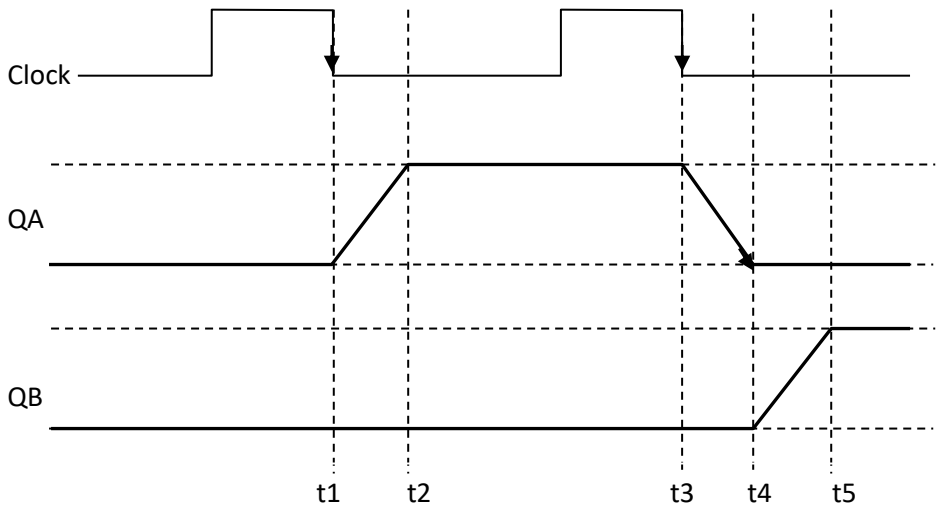


Fig. 12.3

Respuesta de las salidas de los Flip Flops de un contador asíncrono de dos bits.

El problema se presenta cuando la velocidad del reloj (clock) es muy rápido (cuando aumenta la frecuencia de pulsos), es decir, cuando entre los flancos de bajada del reloj no hay suficiente tiempo para que respondan las salidas de los Flip Flops del contador, tal como se aprecia en la Fig. 12.4.

La señal QA, salida del Flip Flop A que inicialmente está en nivel bajo, cambia hacia el nivel alto en el flanco de bajada en t1. En t2 QA debería empezar su cambio hacia el nivel bajo, sin embargo, el nivel de QA en t2 aún no ha cambiado completamente a nivel alto sino hasta t3. En t4 el flanco de bajada restablece la salida QA a nivel bajo hasta t6, momento en que QB se establece en nivel alto.

Los cambios anteriores son inconsistentes, pues QA debería establecerse en nivel alto antes de t2 y QB antes de t4.

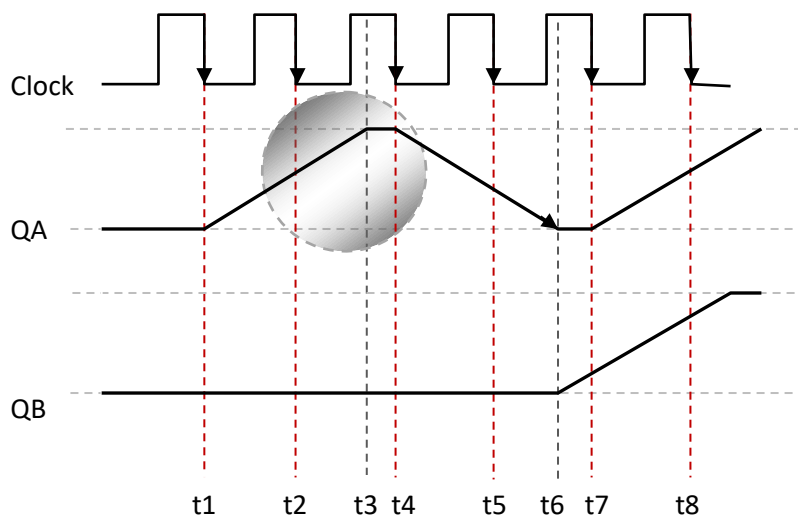


Fig. 12.4

Respuesta de las salidas inconsistentes de los Flip Flops de un contador asíncrono de dos bits.

Para que los contadores asíncronos superen la inconsistencia anterior es necesario tener en cuenta el valor de la frecuencia máxima ($F_{m\acute{a}x}$) en la relación siguiente:

$$F_{m\acute{a}x} < \frac{1}{N(T_p)}$$

Donde:

F = Frecuencia de funcionamiento del contador).

N = número de biestables (FLIP FLOPS) que dispone el contador.

T_p = Tiempo de propagación de la señal (tiempo que tarda la señal en pasar por el biestable).

Según la hoja de datos de los principales fabricantes de Flip Flops, la velocidad con la que se propaga la señal oscila entre 20 y 35 Mhz. Si se asume como velocidad máxima la frecuencia menor (20Mhz), entonces, tendremos que el tiempo de transferencia es 50 ns.

Lo anterior quiere decir que si operamos los Flip Flops por debajo de 20Mhz, los fabricantes garantizan que los circuitos ensamblados funcionarán correctamente.

Ahora, si disponemos de 4 Flip Flops ensamblados en un contador asíncrono, la frecuencia máxima de operación ($F_{m\acute{a}x}$) deberá ser menor de 5Mhz.

12.2.1.1. Contador asíncrono ascendente

El contador asíncrono ascendente consta de Flip Flops tipo T, cuya entrada T se conecta al nivel alto. Asimismo, el primer Flip Flop recibe señal de sincronización del reloj ó clock. Las entradas de reloj de los demás Flip Flops reciben señal de las salidas Q del Flip Flop anterior, tal como se aprecia en el contador asincrónico de tres bits de la Fig. 12.5.

Debemos recordar que un contador de tres bits deberá contar desde 0 a 7, es decir ocho combinaciones, con QA como el bit menos significativo (LSB) y QC como el bit MSB.

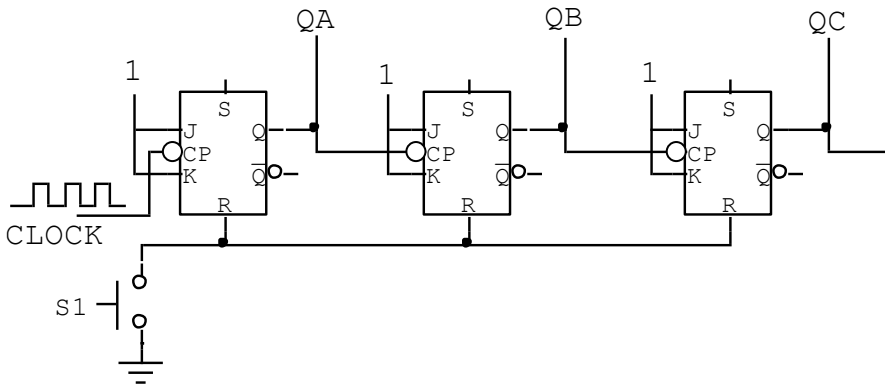


Fig. 12.5

Contador asíncrono ascendente de tres bits..

Inicialmente el contador se resetea a través de S1 restableciendo las salidas en nivel bajo ($QA = QB = QC = 0$). En el primer flanco de bajada del reloj (Ck), QA cambia a 1, QB y QC, se mantienen en nivel bajo. En el segundo flanco de bajada de Ck, QA, cambia 0, originando un flanco de bajada que dispara a QB. QB se mantiene en alto hasta el cuarto flanco de bajada de Ck, donde a su vez Qc se dispara al nivel alto. La secuencia continúa indefinidamente o hasta cancelar la señal de reloj.

La lectura de cada salida (QC, QB y QA) del contador se realiza en el tiempo en bajo de reloj (Ck). Ejemplo: 000, 001, 010, 011, 100, 101, 110, 111, 000, etc. tal como se aprecia en la Fig. 12.6.

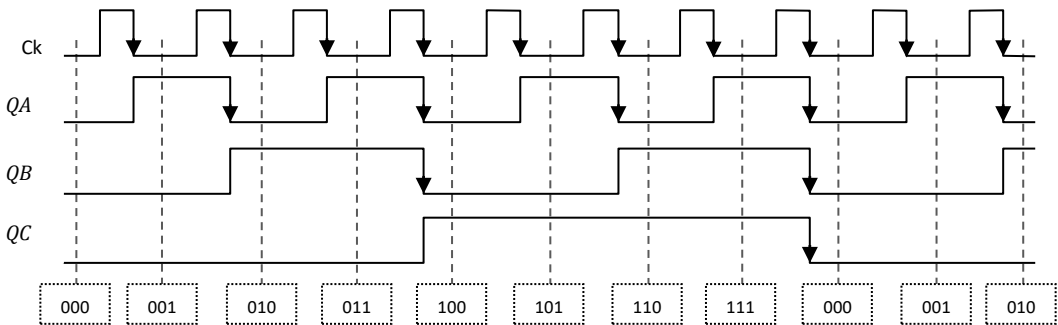


Fig. 12.6

Secuencia de conteo de un contador asíncrono ascendente de tres bits..

12.2.1.2. Contador asíncrono descendente

El contador asíncrono descendente recibe en su primer Flip Flop la señal de reloj correspondiente de Clock (Ck). Los siguientes Flip Flops B y C, reciben su señal de reloj de las salidas complementadas de Q, es decir, el Flip Flop B recibe señal de \overline{QA} y el Flip Flop C recibe señal de \overline{QB} . Tal como se aprecia en la Fig. 12.7.

La secuencia de conteo descendente se inicia reseteando las salidas QA, QB y QC del contador de la Fig. 12.7, mediante la pulsación de S1. La lectura de conteo se realiza en las salidas QA, QB y QC, aunque como podrá verificarse en la Fig. 12.8, la transición o cambios en la salida del contador se produce con las señales producidas en \overline{QA} y \overline{QB} , que ingresan a la entrada de reloj del Flip Flop B y Flip Flop C respectivamente.

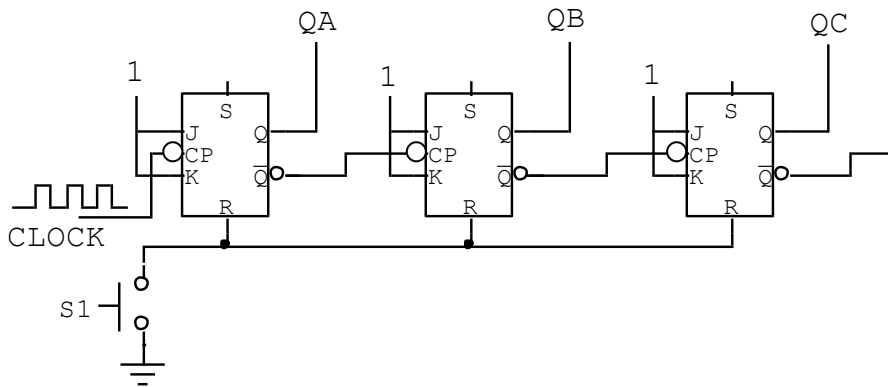


Fig. 12.7

Contador asíncrono descendente de tres bits..

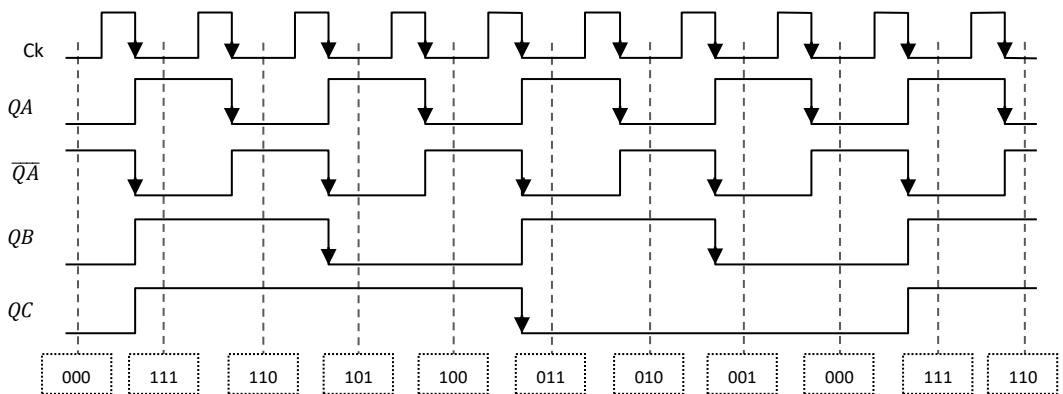


Fig. 12.7

Secuencia de conteo de un contador asíncrono descendente de tres bits. Es importante aclarar que si bien el Flip Flop B y C, reciben señal de las salidas de \overline{QA} y \overline{QB} , los cambios se analizan en las salidas QA, QB y QC.

CONTROL DE CONTADOR ASCENDENTE Y DESCENDENTE

El control de contador consta de un habilitador "M", el que puede adoptar sólo dos valores, el 0 y el 1.

El siguiente caso se desarrolla para el Flip Flop A y Flip Flop B de un contador asincrónico, cuando el habilitador M adopta el valor 0, en este caso, la salida de QA ingresa a la entrada de reloj del Flip Flop B; Si M adopta el valor 1, entonces, la Salida complementada de QA, ingresa a la entrada de reloj del Flip Flop B, tal como se observa en la tabla N° 12.1

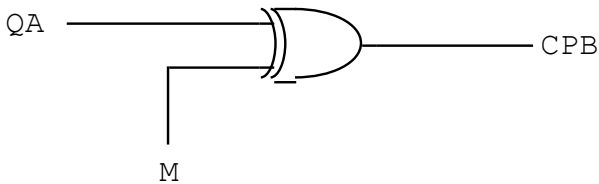
N°	M	QA	Reloj (Ck B)
0	0	0	0
1	0	1	1
2	1	0	1
3	1	1	0

Tabla. 12.1

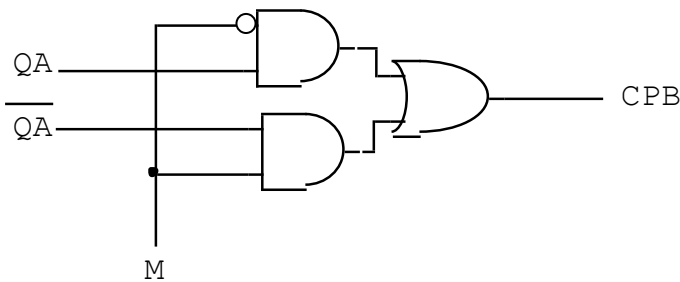
Tabla con dos variables: M (habilitador) y la salida QA del Flip Flop A.

La entrada de reloj del Flip Flop B (Ck B) coincide con las combinaciones de una función OR-EXCLUSIVA, la cual se puede implementar mediante una compuerta Or-Exclusiva o mediante compuertas And, Or y Not.

$$Ck_B = \bar{M}QA + M\bar{Q}A$$



(a)



(b)

Fig. 12.8

(a) Circuito control para seleccionar contador asincrónico ascendente o descendente mediante compuertas Or-Exclusiva.

(b) Circuito control equivalente con compuertas And, Or, y Not.

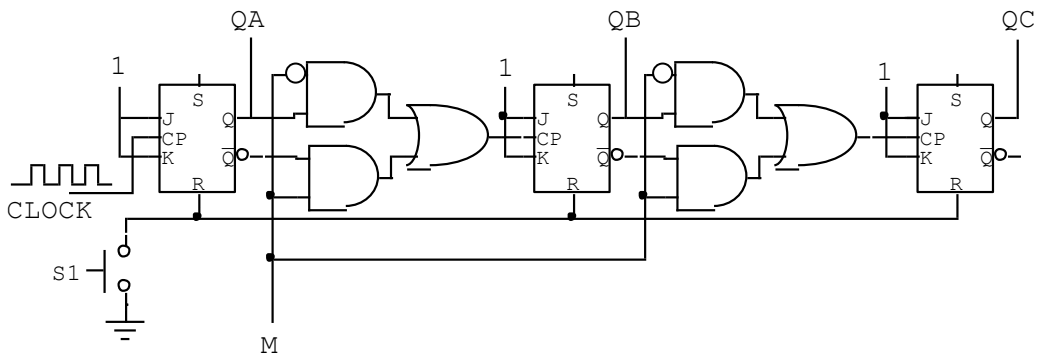


Fig. 12.9

Contador asíncrono ascendente/descendente controlado mediante un circuito control habilitado por un terminal M.

12.2.1.3. Contador Módulo N

Es aquel contador que cuenta desde 0 hasta N-1. Se construye mediante Flip Flops tipo T, de tal manera que el número (n) de Flip Flops cumple con la siguiente relación $2^n > N$.

Para que el contador cuente hasta el módulo N, se utiliza una compuerta NAND cuyas entradas son las salidas del contador que se ponen en 1 cuando el contador cuenta hasta N.

La salida de la compuerta NAND se aplica a todos los terminales Clear de los Flip Flops para restablecer el contador en nivel bajo.

Ejemplo: Construir un contador década.

N = 10 (decimal)

$$2^n > 10$$

N = 4 (Flip Flops)

N = $1010_2 = (QD \ QC \ QB \ QA)$ con QD como MSB y QA como LSB, en donde QD y QB se ponen en alto, por tanto, la compuerta NAND recibirá en su entrada dichas salidas en Alto.

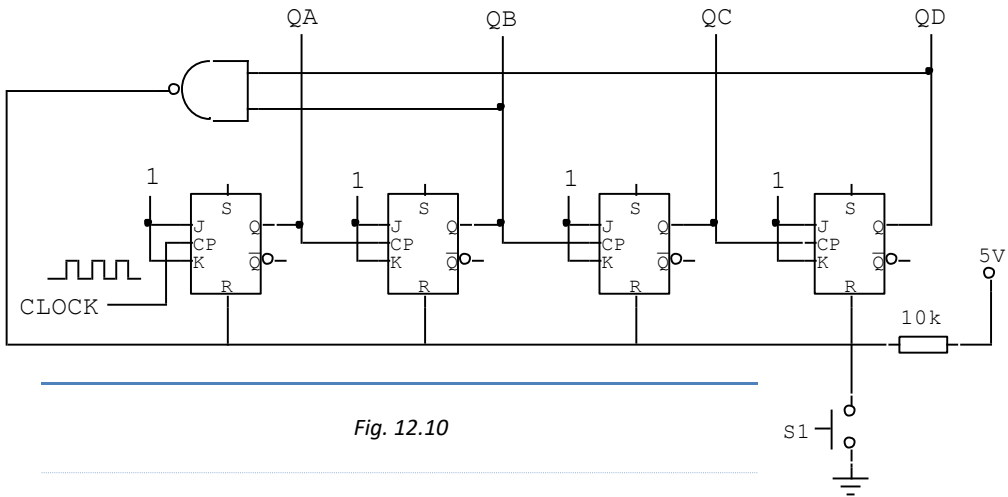


Fig. 12.10

Contador asíncrono módulo 10 (Contador década). Cuando QD y QB se establecen en nivel Alto, la salida de la compuerta NAND se restablece en nivel Bajo y resetea a todo el contador.

12.2.2. Contadores síncronos

Son aquellos contadores cuyos Flip Flops reciben pulsos de sincronización de reloj en cada una de las entradas de reloj. Consecuentemente, todos los Flip Flops reciben al mismo tiempo la señal de reloj y ya no se produce el rizado de los contadores asincrónicos. Por tanto, la frecuencia de trabajo del Contador síncronico puede ser mayor que un Contador asincrónico.

Los Contadores síncronos se construyen con Flip Flops tipo J K, cuyas entradas es una combinación de sus propias salidas.

12.2.2.1. Contador síncrono binario

Los contadores binarios se organizan con Flip Flops: FFA-FFB-FFC-FFD, teniendo en cuenta lo siguiente: MSB para FFD; LSB para FFA,

Contador Binario módulo 16

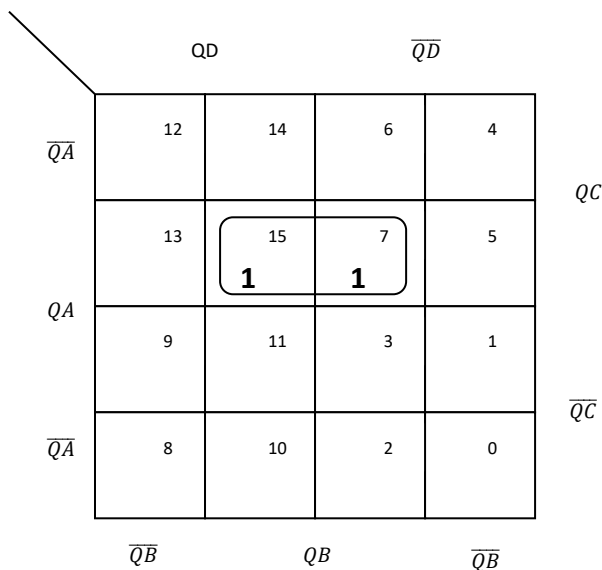
Para construir un Contador binario módulo 16 se realiza con cuatro Flip Flops (Tabla N° 12.2)

N°	SALIDA				ENTRADA			
	QD	QC	QB	QA	JD-KD	JC-KC	JB-KB	JA-KA
0	0	0	0	0	0	0	0	1
1	0	0	0	1	0	0	1	1
2	0	0	1	0	0	0	0	1
3	0	0	1	1	0	1	1	1
4	0	1	0	0	0	0	0	1
5	0	1	0	1	0	0	1	1
6	0	1	1	0	0	0	0	1
7	0	1	1	1	1	1	1	1
8	1	0	0	0	0	0	0	1
9	1	0	0	1	0	0	1	1
10	1	0	1	0	0	0	0	1
11	1	0	1	1	0	1	1	1
12	1	1	0	0	0	0	0	1
13	1	1	0	1	0	0	1	1
14	1	1	1	0	0	0	0	1
15	1	1	1	1	1	1	1	1
0	0	0	0	0				

Tabla. 12.2

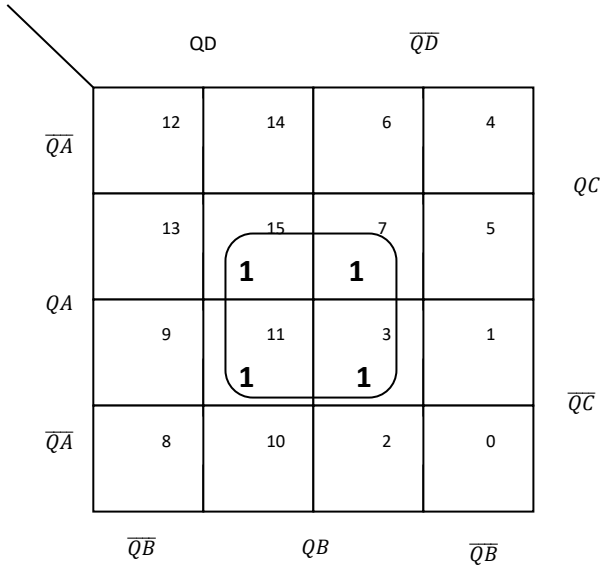
Tabla de verdad de un Contador módulo 16. Debe tenerse en cuenta que las entradas son JK y las salidas Q.

Para JD KD



$$JD_KD = QC.QB.QA$$

Para JC KC



$$JC_KC = QB \cdot QA$$

$$JD_KD = QC \cdot QB \cdot QA$$

$$JB_KB = QA \text{ (Directamente de la tabla 12.2)}$$

$$JA_KA = 1 \text{ (Directamente de la tabla 12.2)}$$

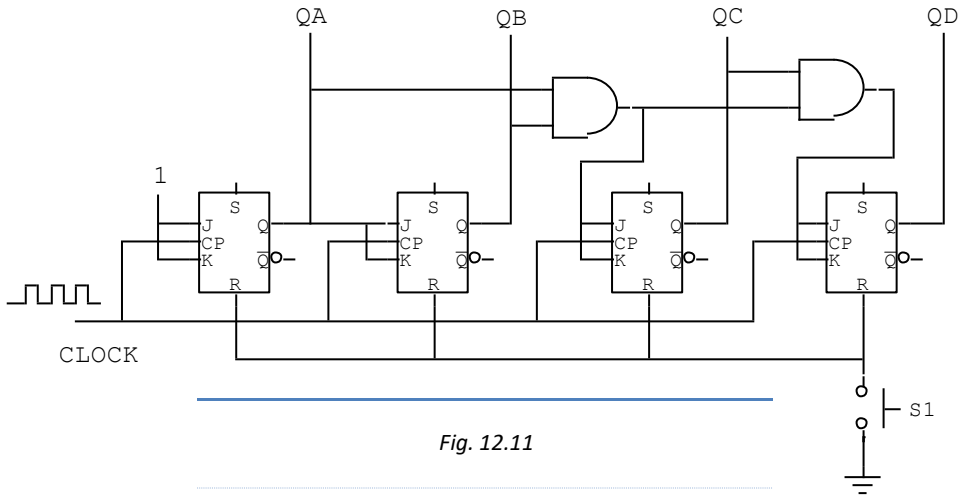


Fig. 12.11

Contador Síncrono módulo 16

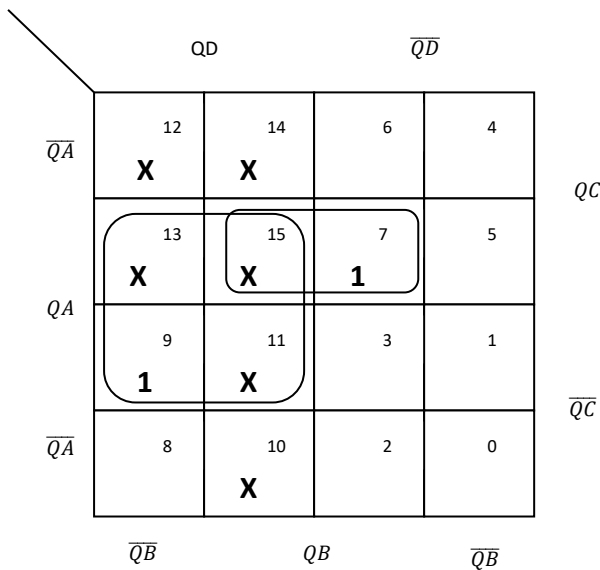
12.2.2.2. Contador década síncrono

El contador década es aquel circuito conformado por cuatro Flip Flops tipo T, los cuales al recibir los pulsos de reloj en sus respectivas terminales, generan niveles lógicos en su salida QD_QC_QB_QA, 0000 hasta 1001, es decir 10 combinaciones, reiniciando nuevamente el conteo en la combinación 0000.

N°	SALIDA				ENTRADA			
	QD	QC	QB	QA	TD	TC	TB	TA
0	0	0	0	0	0	0	0	1
1	0	0	0	1	0	0	1	1
2	0	0	1	0	0	0	0	1
3	0	0	1	1	0	1	1	1
4	0	1	0	0	0	0	0	1
5	0	1	0	1	0	0	1	1
6	0	1	1	0	0	0	0	1
7	0	1	1	1	1	1	1	1
8	1	0	0	0	0	0	0	1
9	1	0	0	1	1	0	0	1
0	0	0	0	0				

Tabla. 12.3

Tabla de verdad de un Contador Década. Debe tenerse en cuenta que las entradas son tipo T.



$$TD = QD \cdot QA + QC \cdot QB \cdot QA$$

$$TD = QA(QD + QC \cdot QB)$$

		QD	\overline{QD}	
\overline{QA}	12	14	6	4
	X	X		
QA	13	15	7	5
	X	X	1	
\overline{QA}	9	11	3	1
		X	1	
\overline{QA}	8	10	2	0
		X		
	\overline{QB}	QB	\overline{QB}	

$$TC = QB \cdot QA$$

		QD	\overline{QD}	
\overline{QA}	12	14	6	4
	X	X		
QA	13	15	7	5
	X	X	1	1
\overline{QA}	9	11	3	1
		X	1	1
\overline{QA}	8	10	2	0
		X		
	\overline{QB}	QB	\overline{QB}	

$$TB = \overline{QD} \cdot QA$$

$$TA = 1$$

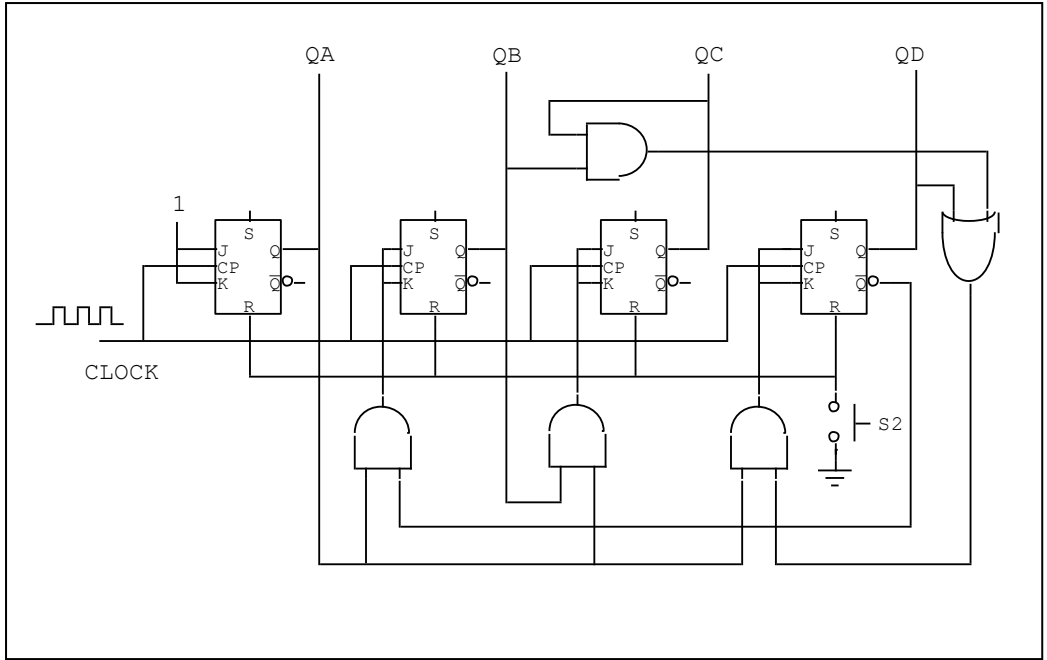


Fig. 12.12

Contador Década Síncrono

12.2.2.3. Diseño de contadores síncronos utilizando las entradas del Flip Flop JK de manera independiente.

Los contadores síncronos se pueden diseñar utilizando las entradas J y K del Flip Flop JK, teniendo en cuenta las combinaciones mostradas en la tabla

Qn	Qn+1	Jn	Kn	Jn Kn
0	0	0	0	0 X
		0	1	
0	1	1	0	1 X
		1	1	
1	0	0	1	X 1
		1	1	
1	1	0	0	X 0
		1	0	

Tabla. 12.4

Tabla de las posibles combinaciones de las entradas JK de verdad de un Contador módulo 16. Debe tenerse en cuenta que las entradas son JK y las salidas Q.

N°	SALIDA				ENTRADA			
	QD	QC	QB	QA	JD-KD	JC-KC	JB-KB	JA-KA
0	0	0	0	0	0 X	0 X	0 X	1 X
1	0	0	0	1	0 X	0 X	1 X	X 1
2	0	0	1	0	0 X	0 X	X 0	1 X
3	0	0	1	1	0 X	1 X	X 1	X 1
4	0	1	0	0	0 X	X 0	0 X	1 X
5	0	1	0	1	0 X	X 0	1 X	X 1
6	0	1	1	0	0 X	X 0	X 0	1 X
7	0	1	1	1	1 X	X 1	X 1	X 1
8	1	0	0	0	X 0	0 X	0 X	1 X
9	1	0	0	1	X 0	0 X	X 1	X 1
10	1	0	1	0	X 0	0 X	X 0	1 X
11	1	0	1	1	X 1	0 X	X 1	X 1
0	0	0	0	0				

Tabla. 12.5

Tabla de verdad de un Contador módulo 12. Con las entradas JK de manera independiente de un Flip Flop JK.

$$JD = QC \cdot QB \cdot QA$$

$$KD = QB \cdot QA$$

$$JC = \overline{QD} \cdot QB \cdot QA$$

$$KC = QB \cdot QA$$

$$JB = QA$$

$$KB = QA$$

$$JA = 1$$

$$KA = 1$$

12.2.2.4. Contador escalonador

Cuenta con secuencia desordenada.

N°	SALIDA			ENTRADA		
	QC	QB	QA	JC-KC	JB-KB	JA-KA
5	1	0	1	X 1	0 X	X 1
0	0	0	0	0 X	1 X	1 X
3	0	1	1	1 X	X 1	X 1
4	1	0	0	X 0	1 X	1 X
7	1	1	1	X 1	X 0	X 1
2	0	1	0	1 X	X 0	0 X
6	1	1	0	X 1	X 1	1 X
1	0	0	1	1 X	0 X	X 0
5	1	0	1			

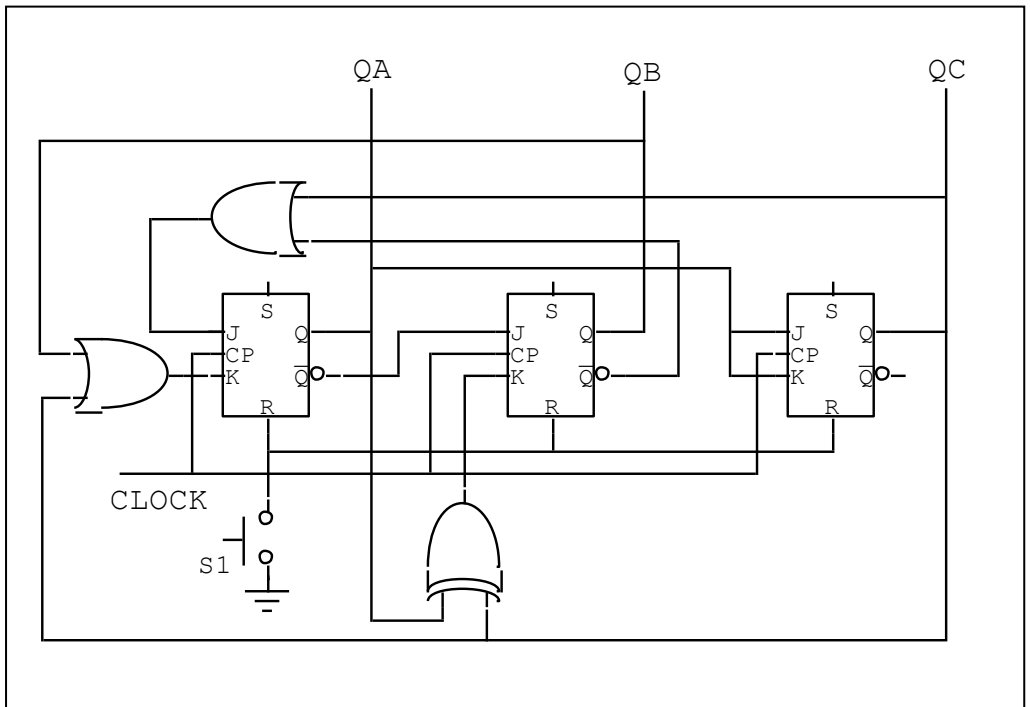
$$JC = KC = QA$$

$$JB = \overline{QA}$$

$$KB = QC \oplus QA$$

$$JA = QC + \overline{QB}$$

$$KA = QC + QB$$



Problemas Resueltos N° 12

1. Diseñar un dado digital

Solución:

Un dado presenta 6 caras, cada cara presenta un número consecutivo desde 1 hasta 6 (1, 2, 3, 4, 5, 6). Dicho dado electrónico deberá presentar sólo dígitos, para ello, en una tabla de verdad se colocará los dígitos empezando por 1 hasta 6 y luego se repetirá la secuencia: 1, 2, 3, 4, 5, 6, 1.

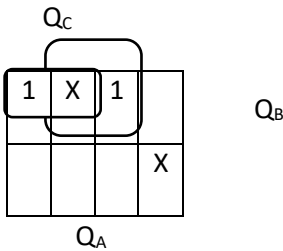
Después de completar la tabla (Q_C , Q_B , Q_A) **por filas**, se procederá a completar la tabla ($J_C K_C$, $J_B K_B$, $J_A K_A$), **por columnas**.

Nº	Q_C	Q_B	Q_A	$J_C K_C$	$J_B K_B$	$J_A K_A$
1	0	0	1	0	1	1
2	0	1	0	0	0	1
3	0	1	1	1	1	1
4	1	0	0	0	0	1
5	1	0	1	0	1	1
6	1	1	0	1	1	1
1	0	0	1			

Normalmente, en los contadores se trabaja con Flip Flops tipo T, por tanto: Si $J_C K_C = 0$, entonces Q_C se mantendrá con su historia previa; y si $J_C K_C = 1$, Q_C conmutará, es decir cambiará de 0 a 1, o de 1 a 0. El mismo procedimiento se sigue para la columna $J_B K_B$ y $J_A K_A$.

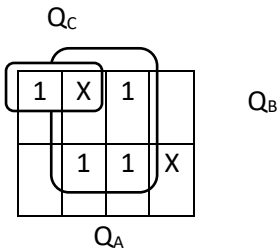
La información de las columnas ($J_C K_C$, $J_B K_B$, $J_A K_A$) se trabaja con mapas de Karnaugh, en donde, X (valores incompletamente definidos) se coloca en las celdas 0 y 7, pues estas dos filas no se considera en la tabla de verdad, por tanto: puede ser 0 o 1 según convenga.

Para $J_C K_C$



$$J_C K_C = Q_C Q_B + Q_B Q_A$$

Para $J_B K_B$



$J_B K_B = Q_C Q_B + Q_A$
 $J_A K_A = 1$; PUES TODA LA COLUMNA $J_A K_A$ ES 1.

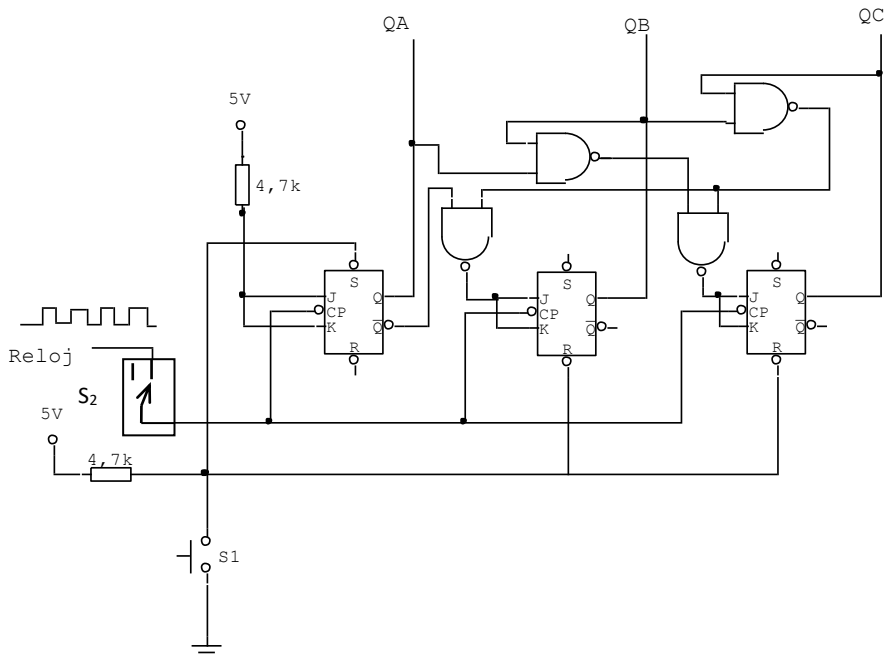
RESUMEN:

$J_C K_C = Q_C Q_B + Q_B Q_A$
 $J_B K_B = Q_C Q_B + Q_A$
 $J_A K_A = 1$

Para utilizar sólo compuertas NAND se aplica la ley de involución y el teorema de Morgan a $J_C K_C$ y $J_B K_B$.

$$J_C K_C = \overline{Q_C Q_B + Q_B Q_A} = \overline{Q_C Q_B} \cdot \overline{Q_B Q_A}$$

$$J_B K_B = \overline{Q_A + Q_C Q_B} = \overline{Q_A} \cdot \overline{Q_C Q_B}$$



Observen que cuando se pulsa S1 (para resetear) el Flip Flop A, el menos significativo (el que está a la izquierda), se pone a 1, pues la entrada del pulso negativo ingresa por la entrada asincrónica SET (S) ó también denominado Preset (P). En los siguientes Flip Flops, el pulso negativa ingresa al reset. Entonces, al inicio (reseteando) tenemos: 0 0 1, y cuando el contador empieza a contar con el siguiente pulso de reloj, la cuenta estará en 0 1 0 y así sucesivamente hasta 1 1 0; luego pasará a 0 0 1.

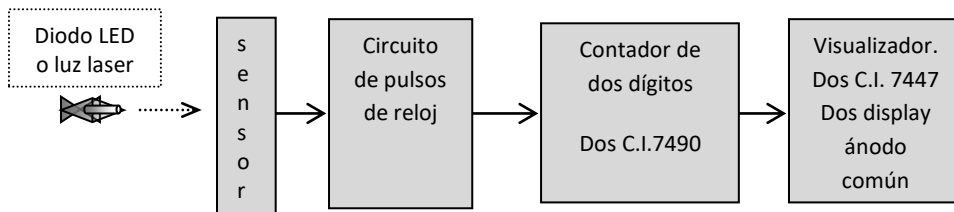
La salida QC, QB, QA, se pueden conectar a la entrada de un decodificador 7447 y la salida de éste a un display ánodo común.

La señal de reloj se conecta al contador (dado digital) a través del switch S2, el cual conecta y desconecta según convenga (cuando se conecta cuenta, cuando se desconecta deja de contar), ya que la frecuencia del reloj debe ser suficiente para que el conteo se realice varias veces en un segundo, por ejemplo 60 Hz; esta frecuencia de trabajo se puede lograr acoplado convenientemente en un timer 555 los siguientes componentes: R1 = 5K Ω , R2 = 10K Ω y C = 1 μ F.

2. Diseñar un contador de objetos

Solución:

El diagrama de bloques puede ser el siguiente:



Iluminación del sensor

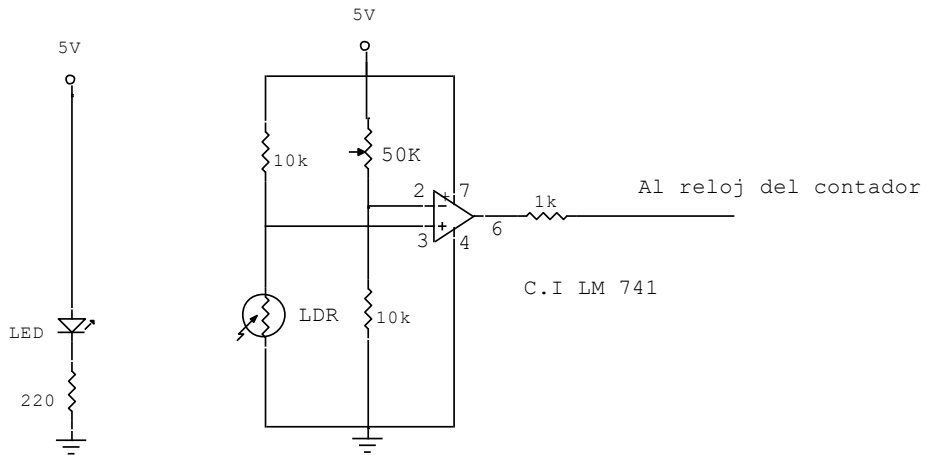
Para contar objetos pequeños, en la que la luz incidente y el sensor se encuentren a pequeñas distancias, es conveniente utilizar un diodo emisor de luz (LED) el que se instala cerca del sensor con su respectiva resistencia de 220 Ω . Para contar objetos más grandes, es necesario utilizar una luz más intensa como la luz laser, ya que esta fuente de iluminación se instalará a mayor distancia del sensor, la que se calibrará adecuadamente para que la luz incida directamente sobre el sensor.

Sensor

El sensor está constituido principalmente por una celda fotosensible, como puede ser una resistencia del tipo LDR (light dependent resistor), el cual es un dispositivo semiconductor cuya resistencia varía inversamente con la intensidad de la luz incidente, es decir a mayor iluminación del diodo LED ó la luz laser que recae sobre la fotocelda, su valor óhmico disminuye, y cuando la iluminación decrece, la resistencia de la fotocelda aumenta linealmente.

Circuito de pulsos de reloj, con C.I. LM741

Para comparar la variación de voltaje producido por el LDR, se utiliza convenientemente un circuito comparador como el LM741, cuya salida se aplica a la entrada de reloj del contador 7490.



Potenciómetro de 50K

LDR = Fotorresistor

En la entrada no inversora se conecta al LDR en un divisor de tensión, el cual variará su resistencia según la intensidad luminosa que incida sobre él y variará obviamente su respectiva caída de tensión. Dicho voltaje se compara con el divisor de tensión de la entrada inversora, el cual cuenta con un potenciómetro de 50KΩ que sirve de ajuste del circuito.

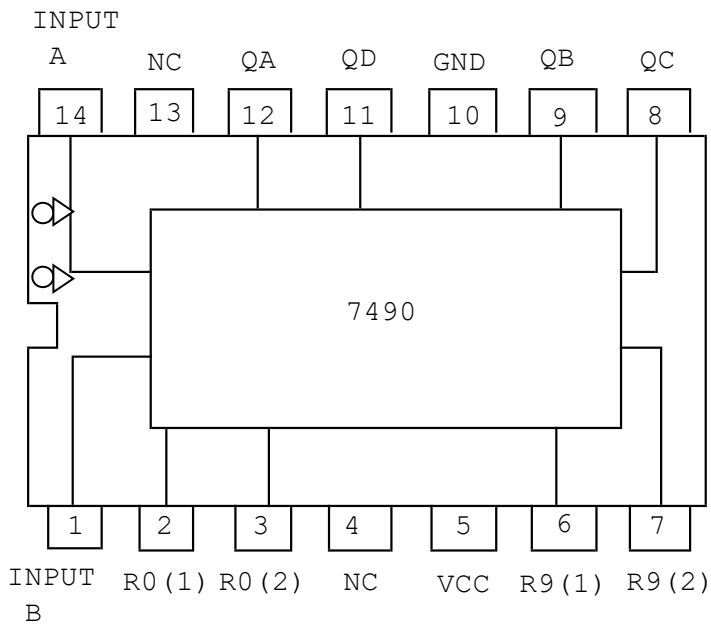
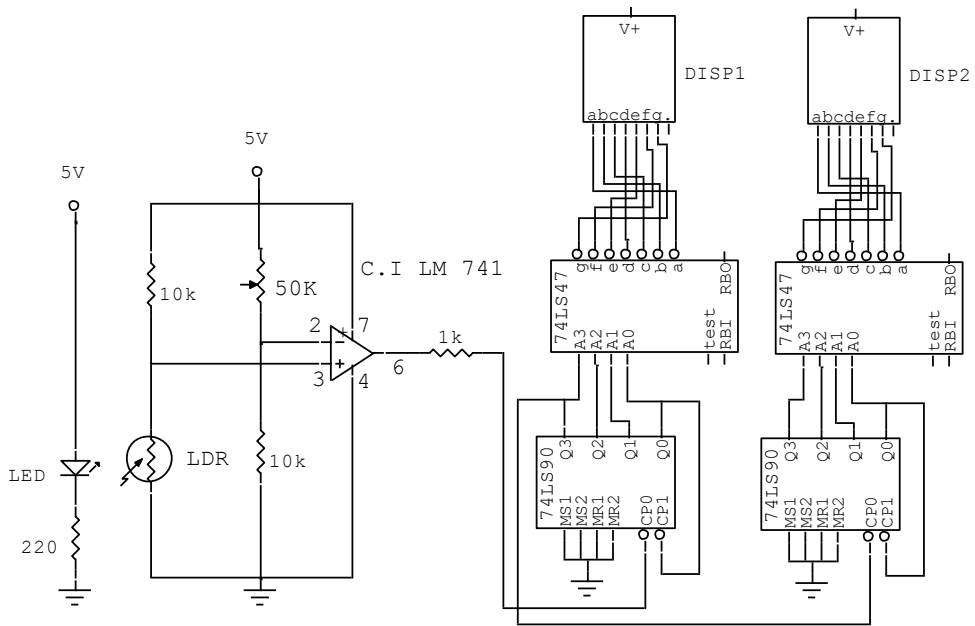
La presencia de luz en el LDR, produce un nivel bajo en la salida (pin 6) del comparador, pues, el LDR presenta una baja resistencia y por tanto una baja caída de tensión; sin embargo, cuando la presencia de luz se corta, el LDR aumenta considerablemente su resistencia y consecuentemente aumenta su caída de tensión, lo cual obliga a la salida del comparador a pasar a nivel alto.

Esta salida variable que pasa de nivel alto a bajo y viceversa se conecta a la entrada de reloj del contador.

Contador de dos dígitos con C.I. 7490 y visualizador con C.I. 7447 y display ánodo común.

Dos contadores 7490 con sus respectivos decodificadores 7447 y sus display ánodo conectado en cascada, posibilitan el conteo de hasta 99 objetos. Uno de los contadores recibirá la señal de reloj del circuito de pulsos, en el pin N° 14 (INPUT A ó CP0), mientras que el otro contador, recibirá los pulsos de la salida del pin N° 11 (QD ó Q3), en su entrada de reloj N° 14. La salida de QA ó Q0, se conecta a la entrada N° 1 (INPUT B ó CP1), en ambos contadores.

Los terminales [R0₍₁₎, R0₍₂₎] ó [MR1,MR2] y [R9₍₁₎, R9₍₂₎] ó [MS1,MS2], se conectan a tierra.



Problemas Propuestos N° 12

1. Diseñar un contador escalonador que cuente de la siguiente manera: 0, 2, 4, 6, 1, 3, 5, 7.
2. Utilizando el contador 7490, implemente un contador que cuente hasta 59,. Utilice los terminales 2 y 3 para resetear el contador. Recuerde que R0(1) y R0(2), ambos conectados a nivel ALTO reestablecen a cero al contador.

Sugerencia: instalar los terminales Qc y QB del contador de las decenas a una compuerta AND, la salida de ésta compuerta se conectará a: R0(1) y R0(2). Esto es comprensible, puesto que mientras Qc y QB no tengan ambos un nivel ALTO (0 1 1 0) correspondiente al número 6, la salida de la compuerta AND se encontrará en nivel BAJO.

3. Implemente un contador que cuente hasta 12, utilizando C.I. 7490.

Sugerencia: instalar los terminales QA del contador de las decenas y los terminales QB y QA del contador de las unidades a una compuerta AND, la salida de ésta compuerta deberá conectarse a: R0(1) y R0(2). Esto es porque al contar a trece (13) los contadores deberán resetearse; es decir cuando QA del contador de las decenas se encuentra en nivel ALTO (0 0 0 1) “UNO” y QB y QA del contador de las unidades tengan ambos un nivel ALTO (0 0 1 1) “TRES” correspondiente al número 3, la salida de la compuerta AND se encontrará en nivel BAJO.

Dispositivos de memoria

13.1. Memoria digital

La memoria de un sistema digital es el componente de todo circuito secuencial cuya función es guardar información a corto y largo plazo. Físicamente, la memoria del computador es el espacio real donde se almacena gran cantidad de datos e instrucciones en forma de bits. El espacio físico de la memoria es el material donde se guardan los bits.

La memoria de la computadora está jerarquizada. La de más alto nivel se refiere a una memoria especial que se encuentra incorporado en el microprocesador. Esta memoria se conoce como “memoria caché” cuya característica básica es su alta velocidad de transferencia de datos.

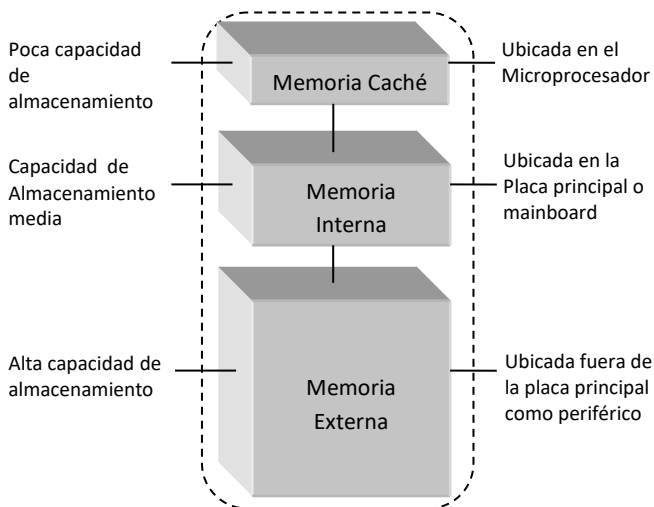


Fig. 13.1

Diagrama de bloques de la jerarquía de memorias de una computadora.

Otra memoria importante es conocida como “memoria interna” la que es direccionada exclusivamente por el microprocesador y se encuentra ubicada en la placa principal de la computadora.

La “memoria externa” es la memoria de menor jerarquía del sistema aunque no la menos importante, ya que en ella se almacenan los datos a largo plazo. Físicamente se ubica en el exterior de la placa principal.

13.2. Características generales

La computadora cuenta con memoria de diversas tecnologías, tipos, velocidad, prestaciones y costos que hacen de éste dispositivo uno de los más importantes del sistema computacional.

Las principales características de las memorias son las siguientes:

- a) **Capacidad de almacenamiento:** referido a la cantidad de información que se puede almacenar en términos de bits.
- b) **Velocidad de transferencia de datos:** es el tiempo que emplean los datos en alojarse o salir de memoria. A menor tiempo de acceso, mayor velocidad.
- c) **Tecnología de fabricación:** aspecto que considera las técnicas de almacenamiento de información y a los recursos tecnológicos empleados en su construcción tomando en cuenta la organización del sistema computacional.
- d) **Fabricante:** es aquella empresa que fabrica la memoria. Ejemplo: Kingston, A Data, Simpletech, Samsung Semiconductor, Corsair, Seagate, etc.

13.3. Medición de la memoria

Las unidades de memoria almacenan datos básicamente de uno a ocho bits, es decir, físicamente una unidad de memoria guarda un bit, luego, estas unidades son agrupadas en ocho bits lo que conforma una **unidad básica de información** denominada Byte. A continuación se expresan las unidades básicas de memoria y los múltiplos de dicha unidad.

1 Bit	equivalente a	1 ó 0
1 Nibble	equivalente a	4 bits
1 Byte	equivalente a	8 bits

Múltiplos del Byte

El prefijo Kilo equivale a la base 2 elevado a la décima potencia. $2^{10} = 1024$

1 Kilobyte (Kbyte)	equivale a	1024 Bytes = 8192 bits
1 Megabyte (Mbyte)	equivale a	1024 Kilobyte = 1048576 Bytes
1 Gigabyte (Gbyte)	equivale a	1024 Mbyte = 1048576 Kbytes
1 Terabyte (Tbyte)	equivale a	1024 Gbyte = 1048576 Mbytes

13.4. Clases de memoria

Las memorias según la función que desempeñan en el sistema de computadoras pueden denominarse:

- a. **Memoria Principal** (memoria interna), es aquella memoria considerada como el banco de trabajo del microprocesador, es decir, aquella donde el microprocesador realiza todo el trabajo de procesamiento de datos en forma temporal.
- b. **Memoria Auxiliar** (memoria externa), referida a la memoria que almacena datos, instrucciones, e información en general a largo plazo.

Las memorias de acuerdo al material donde se guardan los bits pueden ser: de material semiconductor como el silicio, ó de material magnético ubicado normalmente en discos.

13.4.1. Memorias de semiconductores

Son aquellas memorias cuya estructura física se basa en material semiconductor, normalmente el silicio. En ésta estructura cristalina se implementan unidades de almacenamiento de bits y dependiendo de la tecnología empleada pueden ser condensadores o Flip Flops para las memorias de acceso aleatorio y diodos para las memorias de solo lectura.

13.4.1.1. Memorias de acceso aleatorio (RAM)

La **Memoria de Acceso aleatorio** o **Random (RAM)**, por definición, es aquella en la que se accede a cualquier posición de memoria, sin pasar necesariamente por una posición anterior. Por otro lado la RAM es una memoria semiconductor en la que se guarda los bits en dispositivos del tipo condensador para **Memorias RAM Dinámicas (DRAM)**, o del tipo Flip Flops para **Memorias RAM eStáticas (SRAM)**. En ambos casos el bit se *guarda temporalmente* mientras dure la alimentación de energía que le provee la fuente de poder. La información almacenada en una memoria RAM se pierde al perderse la energía (volátil), solo se rescata la información almacenada o guardada en una memoria externa como el disco duro.

La memoria principal de un sistema computacional es del tipo RAM Dinámica normalmente ensamblados en unidades compactas denominadas

“módulos”, por ejemplo, módulos tipo SIMM, DIMM, DDR, DDR-2, DDR-3, etc.

DRAM (RAM Dinámica)

Es aquella memoria en donde las unidades básicas de almacenamiento de datos (bits) se realizan en dispositivos denominados “condensadores”. Estos dispositivos tienen la propiedad de almacenar cargas eléctricas, característica que se refleja en la presencia de una diferencia de potencial o voltaje en sus terminales.

Si el voltaje es alto (alrededor de microvoltios) el dispositivo almacena un bit 1, si es bajo, un bit 0.

La desventaja de este tipo de memoria es la propiedad del condensador denominado “descarga”, en la que el periodo de carga de un condensador ensamblado en una oblea de silicio es la rápida pérdida de carga, lo que se refleja en la pérdida de voltaje y el consecuente cambio del bit 1 al bit 0, hecho que influye en la performance de un sistema computacional.

Los sistemas de memoria (DRAM) solucionan esta eventual pérdida de información mediante un mecanismo de “recarga” denominado “refresco de energía”, en el que periódicamente inyectan carga a los bits 1 para que mantengan su nivel de voltaje apropiado, aunque como consecuencia de lo anterior la velocidad de procesamiento de datos se vea afectado.

La ventaja de la memoria DRAM es la sencillez del dispositivo, motivo por el cual es posible colocar mayor cantidad de condensadores por unidad de volumen en un material semiconductor. Por ejemplo, actualmente existen módulos de memoria DRAM de hasta 4 Gbyte de capacidad.

Estructura de una DRAM

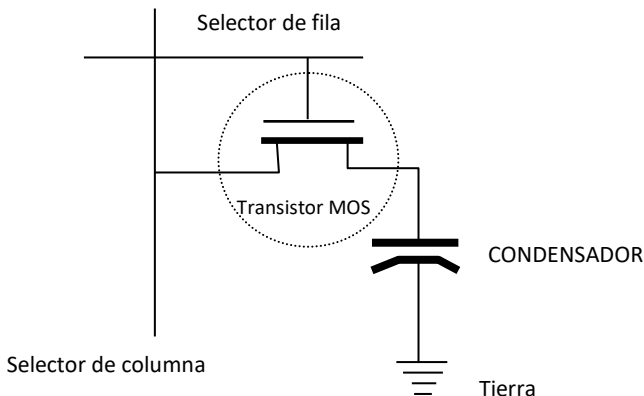


Fig. 13.2

Elemento dinámico que guarda un bit de dato a través de un CONDENSADOR. El dispositivo requiere de un frecuente refresco de energía debido a la pérdida que sufre en el tiempo. Esta debilidad es compensada por la gran cantidad de elementos que se pueden integrar en una oblea de silicio.

SRAM (RAM eStática)

La memoria RAM eStática tiene la característica de almacenar los bits de datos en dispositivos electrónicos denominados Flip Flops, implementados con transistores tipo MOSFET. Estos dispositivos cuentan con un mecanismo de almacenamiento de bits seguros y estables cuyo potencial (voltaje) no decae en el tiempo.

El almacenamiento de un bit 0 ó 1 lo puede realizar mediante la manipulación de sus terminales de entrada de datos y la entrada de reloj de sincronización, o mediante sus terminales de entrada asincrónica.

La desventaja de la SRAM radica en el espacio que ocupa cada dispositivo de almacenamiento, lo que conlleva a colocar un bajo número de Flip Flops por unidad de volumen del material semiconductor, motivo por el cual la capacidad de las RAM estáticas oscila en el orden de los bytes y los Kbytes.

La ventaja de las SRAM se encuentra en la “Alta velocidad de transferencia de datos”, hecho que es aprovechado por el microprocesador para trabajar con los datos mayormente utilizados.

La mayoría de los sistemas computacionales cuentan con dos clases de memorias SRAM: **la memoria Cache**: normalmente ubicado en el interior del microprocesador; y **la memoria CMOS**: ubicada en la placa principal, destinada para almacenar datos de configuración de hardware. Guarda sus datos mediante una batería tipo moneda, por ejemplo el CR-2032, para evitar la pérdida de información.

Estructura de una SRAM

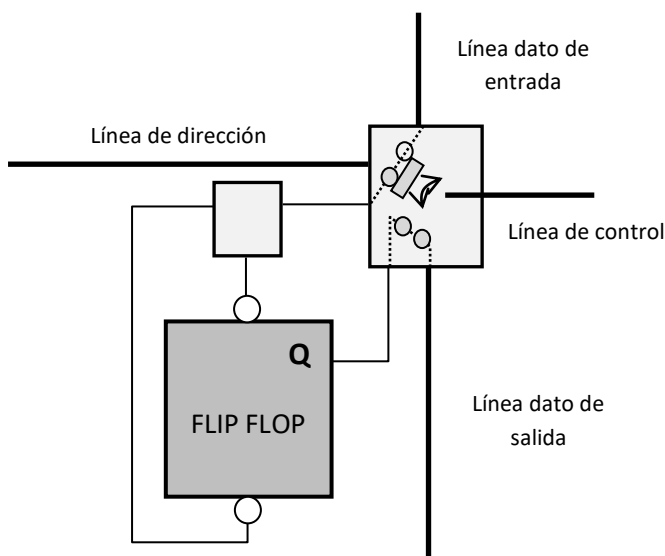


Fig. 13.3

Elemento estático que guarda un bit de dato a través de un Flip Flop. Este dispositivo no requiere refresco de energía, sin embargo su debilidad se encuentra en el hecho de integrarlo en una oblea de silicio. La baja capacidad de almacenamiento respecto a la dinámica es compensada por la rapidez de respuesta del dispositivo.

13.4.1.2. Matriz de memoria semiconductora básica

Cada elemento de almacenamiento en una memoria puede almacenar un bit, es decir un 1 o un 0, en un lugar denominado celda de almacenamiento. Las memorias están formadas por matrices de celdas en las cuales se guardan un determinado número de bits de acuerdo a la capacidad máxima de la matriz. cada celda de almacenamiento se puede especificar mediante una fila y una columna.

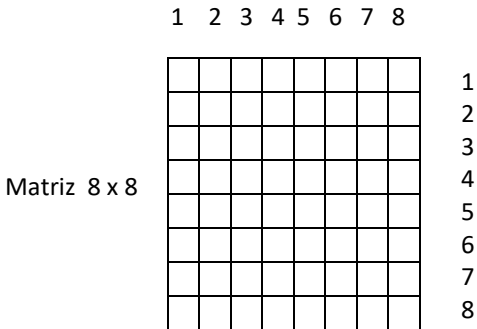


Fig. 13.4

Matriz de memoria 8x8. Cada una de las celdas guarda un bit de dato. La activación de las filas se realiza mediante un decodificador. Las líneas de las columnas permiten el ingreso de los bits de datos.

Una memoria se calcula mediante el número de palabras que puede almacenar multiplicado por el tamaño de la palabra. Por ejemplo, una memoria de 16 Kbyte x 8, puede almacenar 16 384 palabras de ocho bits.

13.4.1.3. Estructura de una RAM semiconductora

Una memoria RAM se puede representar mediante un bloque que presenta tres tipos de buses: Bus de Direcciones, Bus de Datos y, Bus de Control. Cada uno de ellos cumple una función específica en la memoria y es a partir de los dos primeros con los cuales se puede calcular su capacidad de almacenamiento, mientras que el Bus de control determina la lectura o escritura en el dispositivo de memoria.

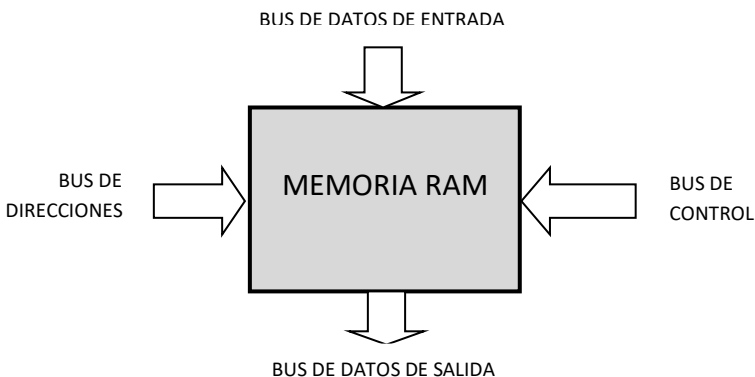


Fig. 13.5

Diagrama de bloques de una memoria RAM de acceso aleatorio. Se observa que tiene tres buses: de dirección, de datos y, de control.

Cálculo de la capacidad de almacenamiento de una memoria RAM.

$$\text{Capacidad de Almacenamiento} = (2^{N^{\circ} \text{ de bits de direcciones}})(N^{\circ} \text{ de bits de datos})$$

La estructura de una RAM comprende básicamente bloques que direccionan posiciones de memoria, líneas de ingreso y salida de datos y, un bloque que habilita o deshabilita la memoria, asimismo, que la RAM lea (operación de lectura) o escriba (operación de escritura).

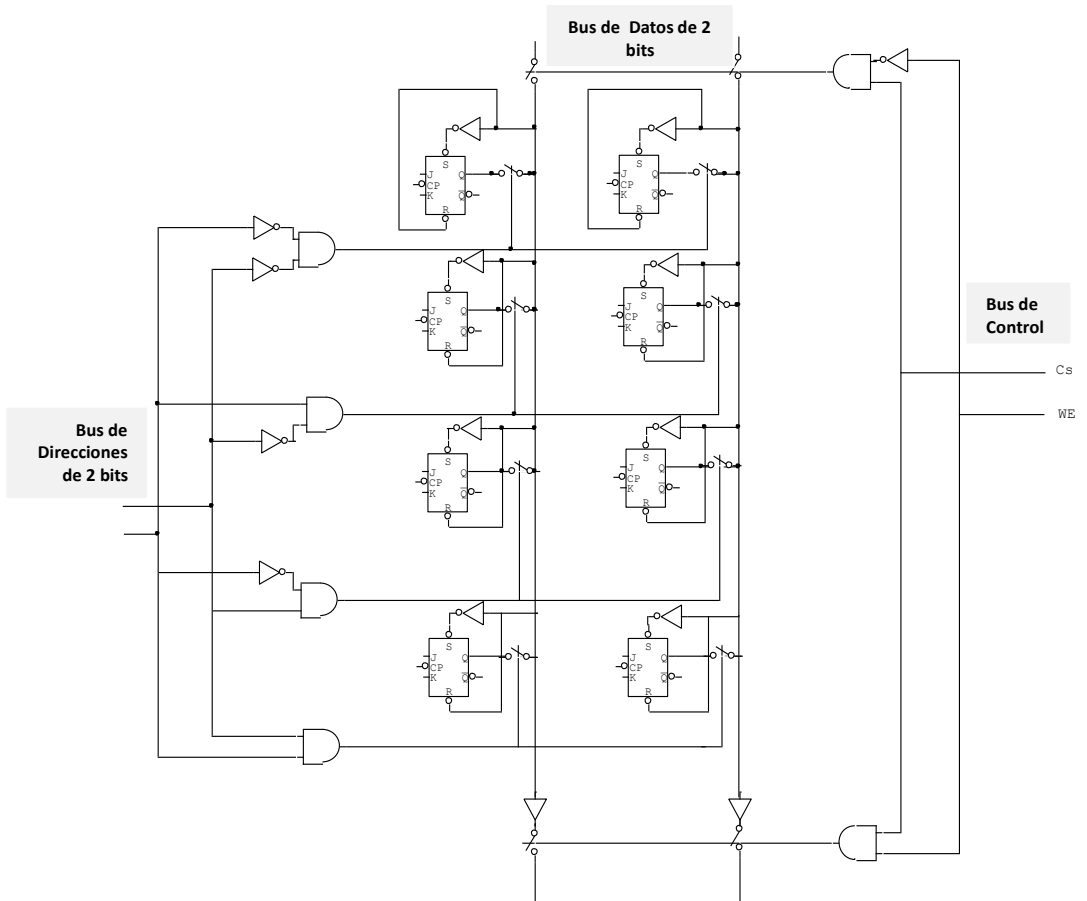


Fig. 13.6

Estructura de una memoria RAM de 8 bits

13.4.1.4. Memorias de sólo lectura (ROM)

La memoria ROM (Read Only Memory) es aquella memoria de solo lectura, en la que los datos o programas se guardan permanentemente o hasta que se vuelva a re grabar para actualizarlo o reemplazarlo. No requiere de alimentación de energía para mantener la información guardada.

Normalmente en las memorias ROM durante el proceso de fabricación se graban dos programas denominados: Bios Post y el Setup. El Bios Post establece la condición inicial para que el sistema computacional sea revisado durante el encendido de la máquina, y de esta manera chequea el estado de los componentes como memoria, microprocesador, etc.

El Setup permite la configuración del sistema computacional mediante un formato del programa, cuyos datos ingresados no se guardan en la memoria ROM sino en la memoria SRAM conocida como CMOS, la cual se encuentra habitualmente en los circuitos integrados especializados (chipset) el que requiere de una batería de 3V para guardar dicha información.

Existen otros tipos de memoria ROM, generalmente de acuerdo a la forma de grabar y mantener los datos, entre estos tenemos:

PROM: es una **ROM Programable**, disponible para que cualquier usuario con los dispositivos adecuados grave datos en la memoria de acuerdo a sus necesidades. Una vez grabado los datos ya no puede re-grabarse.

EPROM: son memorias **PROM** borrables (**Erasable**), en la que se puede volver a grabar información borrando previamente la anterior. Para borrar la data anterior es necesario aplicarle rayos ultravioletas por un determinado tiempo.

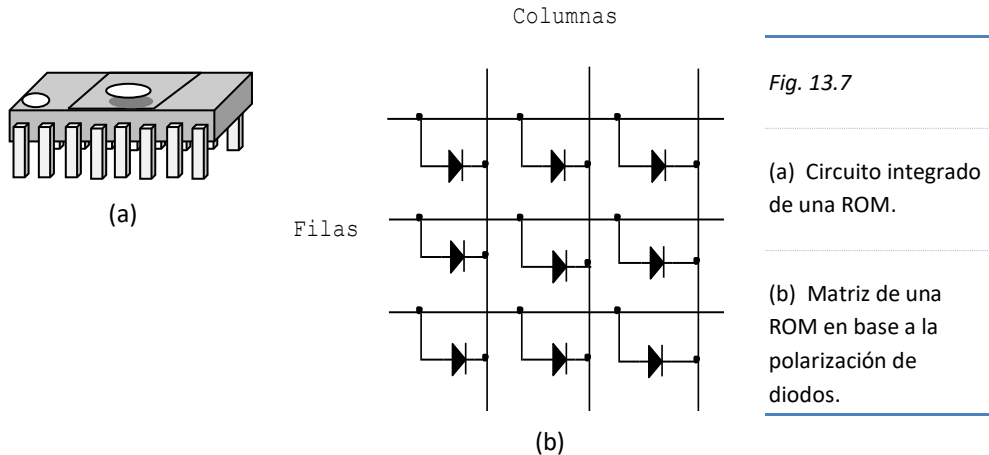
EEPROM: son memorias EPROM borrables **Eléctricamente**, en el que se puede grabar y borrar mediante pulsos eléctricos aplicados desde una tarjeta especialmente diseñada para tal fin.

La estructura interna de una memoria ROM se basa en dispositivos semiconductores como los diodos o como dispositivos de protección como los fusibles. En ambos casos, la configuración de los dispositivos para que conduzcan corriente adoptan el valor alto (1), y adoptan el valor (0) cuando los diodos no conducen corriente por estar fundidos o cuando los fusibles se han escindido. Este proceso se realiza al momento de su construcción teniendo en cuenta los programas y datos que se desean almacenar y los cuales no se pueden modificar, pues físicamente la estructura de los dispositivos son inalterables.

La distribución de los dispositivos internos de una memoria ROM se realiza en forma de MATRIZ, en la que las filas corresponden a las líneas de direccionamiento, mientras que las columnas transportan la información almacenada.

La lectura de dicha información se realiza activando la dirección correspondiente (fila) mediante el bus de direcciones, y la data almacenada se lee a través del bus de datos (columnas).

La estructura interna de las memorias EPROM y EEPROM están constituidas por otros dispositivos dinámicos como los transistores de tecnología MOS.



13.4.1.5. Memorias Flash

Es aquella memoria cuya tecnología exhibe las mejores características de las memorias RAM y ROM juntas. Es decir son memorias de gran capacidad de almacenamiento, de lectura y escritura de datos y, de almacenamiento permanente (no volátil) sin necesidad de alimentación de energía. A lo anterior se suma la portabilidad y su costo accesible de acuerdo con su densidad de almacenamiento.

La estructura interna de una memoria flash consta de un conjunto de transistores de tecnología MOS. Cada uno de estos transistores guarda un bit (0 ó 1) de acuerdo a la cantidad de carga eléctrica suministrada en la **operación de escritura**, es decir, cuando conectada a un sistema computacional se graban datos en la memoria flash. En este proceso a mayor cantidad de carga eléctrica en el transistor MOS se almacena un bit 0 y con ausencia de carga, se almacena un 1. La característica principal de este proceso consiste en que el almacenamiento de carga en el transistor MOS puede durar hasta 100 años sin requerir alimentación de energía.

En la operación de lectura, el sistema computacional aplica un voltaje positivo en el terminal de control del transistor MOS, el cual actúa de la siguiente manera: si posee carga (0), el transistor no se activará, y si no tiene carga, el transistor se activará. Esta situación es interpretada por el sistema computacional como bit 0 ó 1 respectivamente.

El formateo o borrado de la memoria flash es realizado por el sistema computacional retirando toda la carga de los transistores MOS con que dispone, colocando cada una de las celdas de la memoria en 1s (unos).

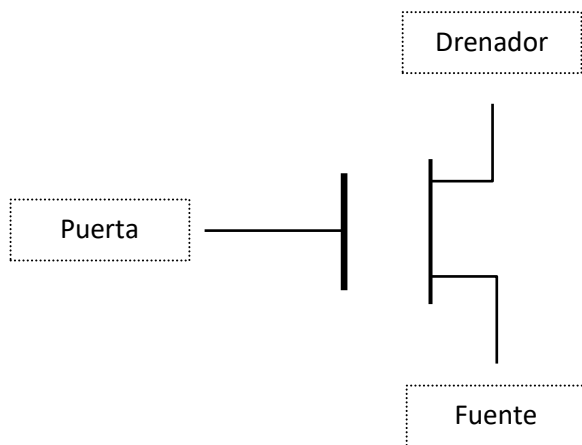


Fig. 13.8

Transistor de tecnología CMOS, en el que por medio de la puerta y los voltajes apropiados en la fuente y drenador se pueden almacenar cargas eléctricas.

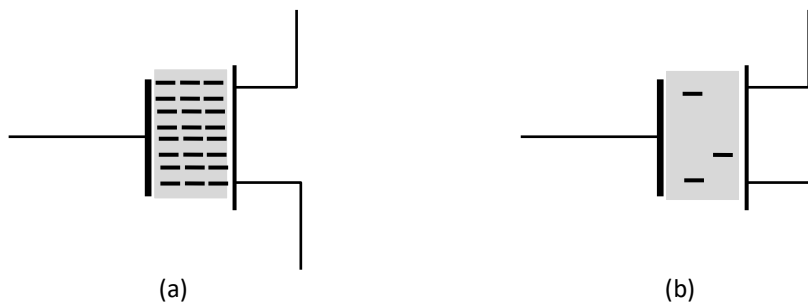


Fig. 13.9

(a) Transistor CMOS con carga eléctrica negativa entre sus terminales, lo que indica el almacenamiento de un bit 0.

(b) Transistor CMOS sin carga eléctrica negativa entre sus terminales, lo que indica el almacenamiento del un bit 1.

13.4.2. Memorias magnéticas

Las memorias magnéticas utilizan la propiedad de campos magnéticos para inducir corriente eléctrica en un devanado (alambre enrollado) y del mismo modo un devanado con corriente eléctrica inducir campos magnéticos en un material ferromagnético.

Los dispositivos de almacenamiento magnético son aquellas unidades que presentan gran capacidad de almacenamiento de información tales como: los discos duros y cintas magnéticas de respaldo, los cuales son masivamente utilizados por los sistemas computacionales y que por este tiempo sus capacidades bordean las decenas y hasta centenas de Terabytes.

Estos dispositivos fundamentalmente disponen de material ferromagnético sobre la superficie de discos o cintas y un cabezal de lectura y escritura constituido por un diminuto electroimán, que realizan las operaciones básicas para extraer y guardar información llamada “operación de lectura y escritura”.

La superficie del disco o cinta están constituidos por microscópicos “imanes” que al principio cada uno de ellos se encuentra orientado en la misma dirección dispuesta de fábrica, luego, cuando se graba información, dichos “imanes” se reorientan adecuadamente. Por ejemplo, cuando un imán ubicado horizontalmente se ha orientado N-S (Norte – Sur) indica la presencia de un bit 1, si lo hace de S-N, indica un bit 0; obviamente el valor del bit depende de la orientación del imán y de la lógica adoptada por el fabricante del dispositivo.

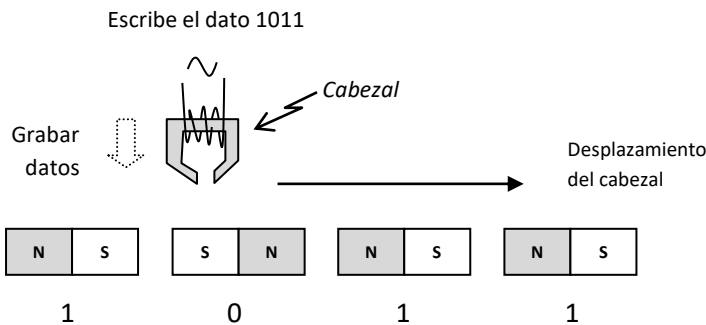


Fig. 13.10

Operación de escritura. el devanado del cabezal recibe señal eléctrica de acuerdo a los datos que se requiere grabar, por ejemplo, el dato 1011. El cabezal se desplaza de izquierda a derecha induciendo la polaridad correspondiente en cada uno de los “imanes” del material magnético.

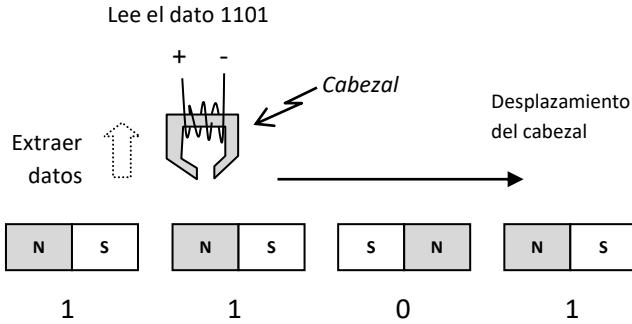


Fig. 13.11

Operación de lectura. El devanado del cabezal presenta una diferencia de potencial en sus terminales debido a la inducción magnética recibida de los “micro imanes”, por ejemplo, el dato 1101 contenido en el material magnético inducirá un voltaje en el devanado según la orientación de los respectivos imanes.

13.4.3. Memorias ópticas

Son aquellas memorias que utilizan la propiedad de la luz para grabar o extraer datos de los dispositivos de almacenamiento, los más conocidos son el CD ROM y el DVD.

El mecanismo de las memorias ópticas consiste en un disco plástico con una recubierta de material reflectante y un diodo emisor de luz laser. El material reflectante de un CD y DVD del tipo ROM (memoria de solo lectura) ya no se puede modificar una vez que han sido grabado datos en él.

Para el caso de CD y DVD RW (regrabable) el material se comporta como zonas claras y oscuras dependiendo si se quieren grabar bits 1 ó 0 respectivamente. La luz laser emitida por el diodo es más intensa en la **operación de escritura** para permitir la grabación de datos generando “zonas” según requerimiento de los datos a guardarse.

En la **operación de lectura** la luz emitida del laser es reflejada en las zonas claras generadas en el proceso de escritura y es interpretada como el almacenamiento de un bit 1, y es interpretado como un bit 0 cuando la luz incide en zonas oscuras impidiendo su reflexión.

Un detector óptico situado en el interior del dispositivo captura la luz reflejada por las zonas previamente grabadas y lo convierte en impulsos eléctricos que determinan si registró un bit 1 o un bit 0.

Con esta secuencia de 1s y 0s se puede almacenar aunque no indefinidamente pero si por unas cuantas veces en el mismo CD ó DVD del tipo RW.

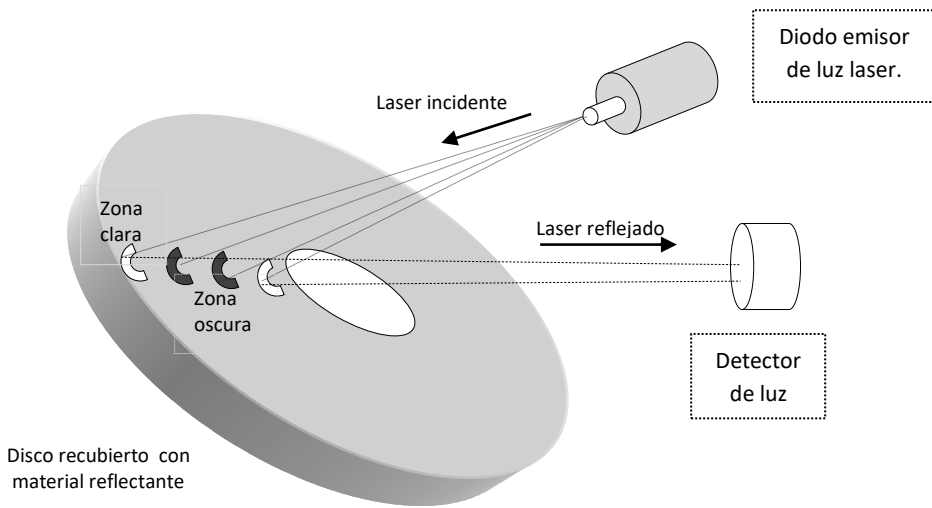


Fig. 13.12

Mecanismo de lectura en memorias del tipo CD y DVD RW. El detector lee el dato 1001.

13.5. Distribución interna de la Memoria

En la Memoria Principal se aloja el programa o secuencia de instrucciones que debe seguir la computadora para realizar el procesamiento de los datos, el cual está almacenado en una parte de la memoria llamado "Memoria de Instrucciones", mientras que los datos y resultados de las operaciones se almacenan temporalmente en el resto de la memoria disponible denominado "Memoria de Datos".

La matriz de memoria que consta de Instrucciones y datos cuenta con un Registro de Direcciones (MAR) alimentado por el Bus de direcciones. La salida de la MAR ingresa a un decodificador de Direcciones, el cual ubica la posición de memoria de una instrucción o de un dato. Cuando dicha información es extraída de la matriz pasa a un Registro de Datos (MDR), el que a su vez da salida a las instrucciones a través del Bus de Salida de Instrucciones y los datos a través del Bus de entrada y salida de datos (RAM)

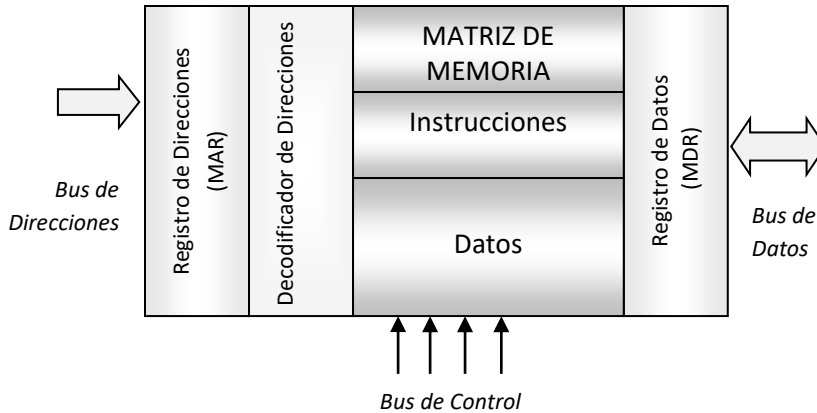


Fig. 13.13

Distribución interna de la memoria principal de un sistema computacional.

Problemas Resueltos Nº 13

1. Implementar un circuito visualizador de palabras, utilizando una memoria ROM programable de tipo EEPROM.

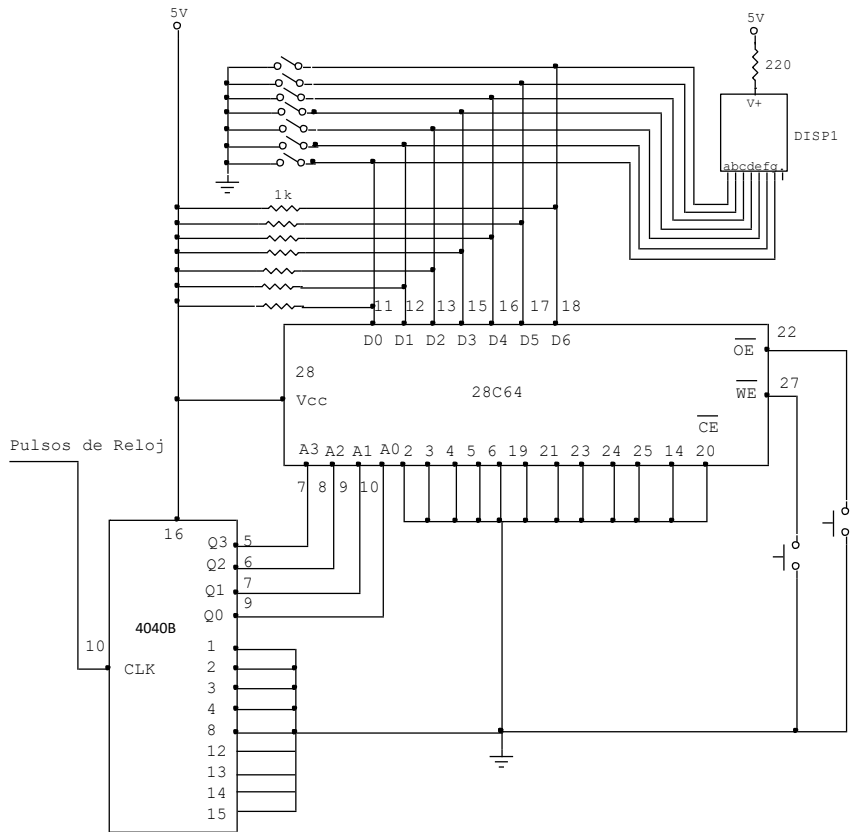
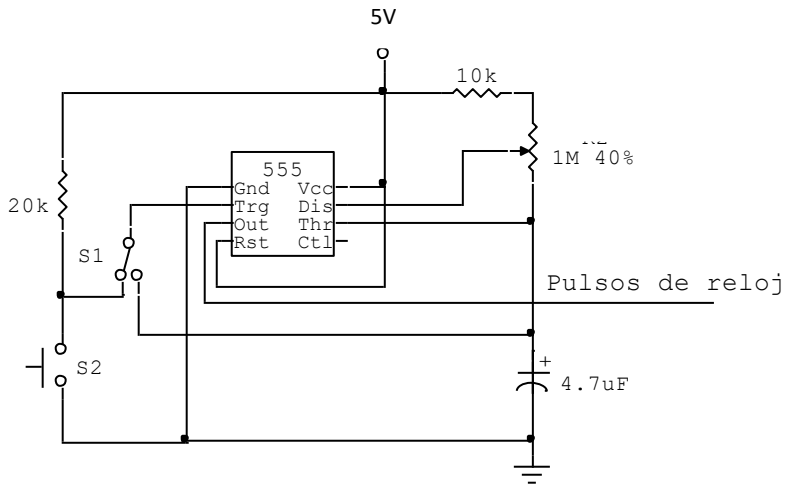
Solución:

Para implementar el circuito visualizador de palabras se puede utilizar la memoria EEPROM 28C64A, que tiene una capacidad de almacenamiento equivalente a $8K \times 8 = 64KB$ (kilobyte).

La memoria 28C64 tiene dos modos de operación: de escritura y de lectura:

Proceso	\overline{CE} Pin Nº 20	\overline{OE} Pin Nº 22	\overline{WE} Pin Nº 27
Escritura o Grabación	0	1	0
Lectura	0	0	1

Para realizar ambos procesos, es conveniente que el generador de pulsos de reloj se configure de dos modos: como multivibrador astable o como monoastable, lo cual se consigue para ambos casos con el timer 555, conectando el pin Nº 2 (Trg) con el pin Nº 6 (Thr) se comporta como Astable, y conectado con tierra a través del tact switch S2 se comporta como Monoastable.



a) Proceso de escritura o grabación.

Para realizar la operación de escritura, el terminal \overline{OE} se conecta a tierra a través de un pulsador de doble posición.

Se escribe letra por letra a través de los switch que se conectan al display y al 28C64.

Se utiliza el timer 555 en modo monoestable e ingresa un pulso de reloj al contador 4040B cuando se pulsa el switch S2.

Para grabar la letra ingresada, el terminal \overline{WE} se conecta a tierra y el terminal \overline{OE} a fuente o 1 lógico.

Si deseamos grabar una secuencia de letras, por ejemplo, LA SIPAN ES TU FUTURO, se procederá conforme a los pasos anteriores para cada letra.

b) Proceso de lectura.

En primer lugar se desconectan cada uno de los switch que se conectan al display.

Se configura los terminales \overline{WE} a fuente o 1 lógico, y \overline{OE} a tierra o 0 lógico.

El timer 555 se configura en el modo de multivibrador astable a través del switch S1, cuya frecuencia se ajusta con el potenciómetro de $1M\Omega$.

Problemas Propuestos N° 13

- Una memoria tiene 1024 posiciones de memoria puede almacenar 16 bits en cada posición. ¿Cuál es la capacidad de dicha memoria en bytes)
(a) 512 K (b) 1024 K (c) 2048 K (d) 2 K
- Una memoria de 1048576 posiciones de memoria, tiene un bus de direcciones de:
(a) 20 bits (b) 15 bits (c) 12 bits (d) 10 bits

Capítulo 14

Estructura básica de un computador

14.1. El computador

El computador es una máquina electrónica que realiza determinadas acciones en base a instrucciones recibidas, las cuales son almacenadas y procesadas a gran velocidad.

El computador está compuesto de una parte física llamada “hardware” y otra parte lógica llamada “software”.



<http://www.minedu.gob.pe/fotosmed/0389-F.htm>

Fig. 14.1

Fotografía de los computadores portátiles distribuidos por el Ministerio de Educación de Perú.

La tecnología del hardware evoluciona vertiginosamente en estos últimos días, de tal forma que los computadores son más compactos y más potentes.

El hardware, es el conjunto de circuitos electrónicos que conforman el sistema de computadoras, tales como: la placa principal, el microprocesador, la memoria, el disco duro, la fuente de poder, etc.

El software, es el conjunto de programas y aplicaciones conformado por las rutinas de instrucciones (almacenados en una unidad de memoria) que coordinan el funcionamiento del sistema y la ejecución de trabajos diversos, tales como: sistemas operativos, ofimática, diseño, juegos, búsqueda, etc.

Existe un software especial grabado en la memoria ROM (estructura electrónica del sistema) denominado "Firmware", el cual contiene un grupo de programas (BIOS – Basic Input/Output System) que sirven de intermedio entre el software y el hardware de la computadora.

El computador se encuentra comunicado con el mundo exterior a través de sus periféricos de entrada y salida, pues, permanentemente recibe y entrega información a través de sus periféricos respectivos.

14.2. Organización y arquitectura

La organización y la arquitectura de computadoras se ajusta más a la descripción que sobre dichos temas realiza William Stallings en su libro "Organización y Arquitectura de Computadores", quien manifiesta, que la organización de computadoras son atributos del hardware son transparentes o que en todo caso no son accesibles a un programador tales como: las unidades funcionales y sus interconexiones que materializan especificaciones arquitectónicas, entre ellas se incluye las señales de control, interfases entre el computador y los periféricos y la tecnología de la memoria usada.

Mientras que arquitectura de computadoras se refiere a los atributos de un sistema que son visibles a un programador, es decir, aquellos componentes que influyen en la ejecución lógica de un programa, por ejemplo: conjunto de instrucciones, el número de bits usados para representar varios tipos de datos (caracteres, número), mecanismos de entrada y salida, técnicas de direccionamiento de memoria, entre otros.

Por tanto, podemos decir que la arquitectura de computadoras puede sobrevivir muchos años, mientras que la organización de computadoras cambia con la evolución tecnológica de sus componentes.

14.3. Estructura y función

Estructura: es el modo en que los componentes de un sistema se interrelacionan entre sí.

Función: es la operación que realiza cada componente individualmente dentro de la estructura.

La Estructura de un computador consta de los siguientes componentes:

- Unidad Central de Procesamiento (CPU)
- Memoria Principal
- Periféricos de Entradas y Salidas
- Sistema de Interconexiones

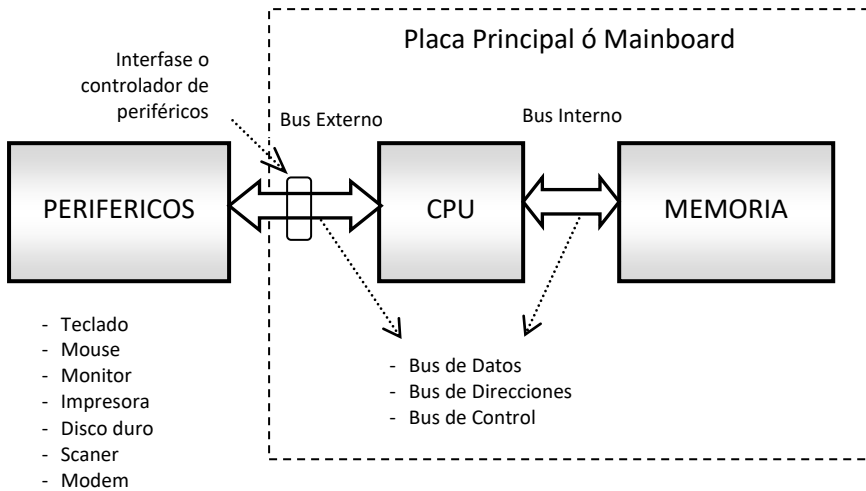


Fig. 14.2

Diagrama de bloques de la estructura básica de un computador

Las Funciones básicas que un computador puede llevar a cabo son los siguientes:

- Procesamiento de datos
- Almacenamiento de datos
- Transferencia de datos
- control

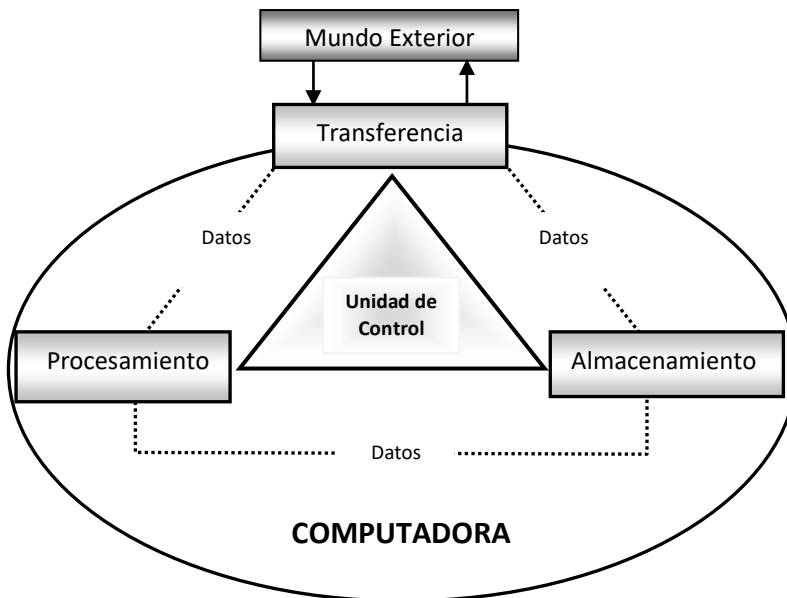


Fig. 14.3

Diagrama de bloques de las funciones básicas de un computador

14.4. El microprocesador

La Unidad Central de Proceso (CPU), también llamado “microprocesador”, es un circuito electrónico que cumple la función de cerebro de la computadora. La función principal del microprocesador es recoger, decodificar y ejecutar las instrucciones que residen en memoria, siendo capaz de transferir datos desde la memoria externa a sus propios registros y viceversa. Asimismo, tiene la capacidad de realizar tareas de orden y limpieza del sistema, desde proveer de refresco de energía a la memoria RAM dinámica, hasta tomar medidas en caso falle la fuente de alimentación. Todas estas características hacen que el microprocesador sea considerado como el componente más importante de una computadora y por tanto el más costoso del sistema.

Los componentes principales de un microprocesador son: la Unidad Aritmético Lógica (ALU), Registros de Trabajo y la Unidad de Control. En los Registros de Trabajo se almacenan datos, direcciones u otras informaciones pertinentes.

El desempeño de un microprocesador depende de varios indicadores, entre

ellos la velocidad del reloj, la velocidad del bus interno y externo, la capacidad de la memoria caché, el número de bits de datos, entre otros.

La velocidad del microprocesador se expresa en unidades de frecuencia, lo que significa que la frecuencia del reloj se da en términos de ciclos por segundo, sin embargo, es bueno aclarar que el procesamiento de instrucciones se puede realizar en uno o varios ciclos de reloj.

La velocidad del bus interno y externo se determina mediante la velocidad denominado FSB (Frontal Side Bus), el que refiere la cantidad de datos que se desplazan desde y hacia el microprocesador.

La capacidad de memoria caché incorporado en el microprocesador influye en la velocidad del mismo, ya que a mayor capacidad mayor velocidad y viceversa.

Si el bus de datos y de direcciones es mayor, el microprocesador tendrá mejores prestaciones que otro microprocesador de menores características de bus, que redundará en la velocidad de este importante dispositivo.

Los microprocesadores o CPU, se fabrican en un solo chip o circuito integrado.

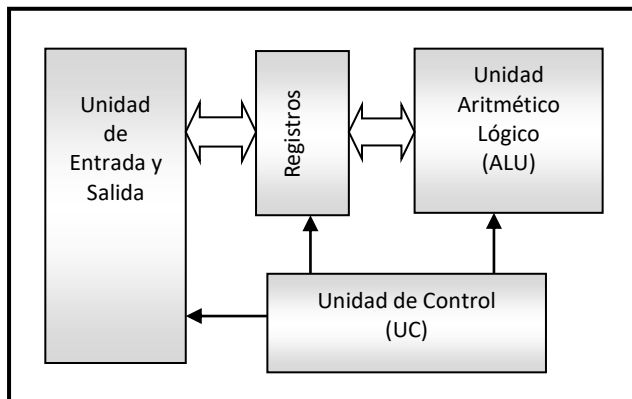


Fig. 14.4

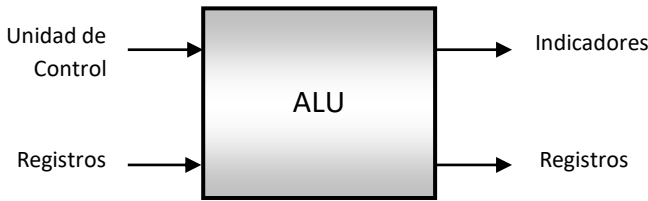
Diagrama de bloques de un Microprocesador.

14.4.1. Unidad Aritmético Lógica

Es aquella Unidad que realiza realmente los cálculos matemáticos y lógicos con los datos, así como las comparaciones entre las mismas; asimismo, ejecuta rutinas de mantenimiento del sistema y toma las medidas necesarias según lo indiquen sus instrucciones previamente almacenadas.

Los datos se presentan a la ALU a través de Registros, y en estos se almacenan también los resultados de las operaciones efectuadas, en el que se

activarán algunos indicadores como los (flags) de acuerdo al resultado de las operaciones. Por ejemplo, un indicador de desbordamiento se indicará mediante un 1 si el resultado de la operación excede al número de bits del Registro de almacenamiento.



Entradas y Salidas de la ALU

Fig. 14.5

Diagrama de bloques de las entradas y salidas de la ALU.

Las operaciones que realiza la ALU son:

- Aritmético (suma, resta, incremento, decremento)
- Lógico (AND, OR, OR-EXCLUSIVO, complementación)
- Otras funciones (transferencia de registros, rotaciones: derecha, izquierda)

La Unidad Aritmético Lógica recibe señales digitales desde la Unidad de Control, de acuerdo a estas señales la ALU ejecutará una determinada instrucción.

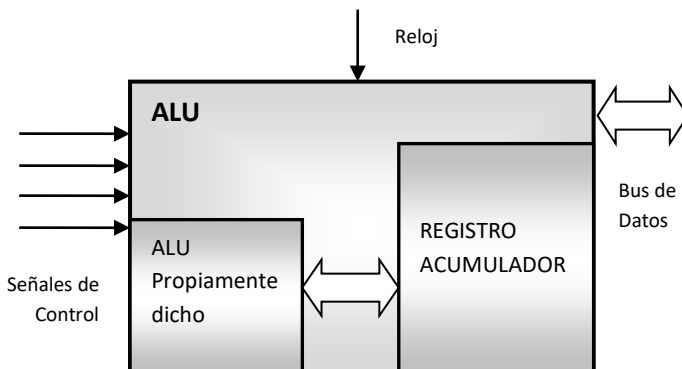


Fig. 14.6

Diagrama de bloques de la estructura interna de la ALU.

El Acumulador es un registro muy importante, pues se considera el banco de trabajo de la ALU. El Acumulador funciona como registro de doble sentido ya que se comporta como origen y destino de datos. Como registro origen, el Acumulador puede contener información producto de una operación previa. Como registro destino, se puede grabar temporalmente el resultado de una operación que se está ejecutando; esta acción de grabar, borra la información anterior. Así pues, el Acumulador es el soporte y el registro de trabajo de las instrucciones que ejecuta el microprocesador.

14.4.2. Registro temporal de almacenamiento de datos y direcciones

Dentro de la Unidad Central de Proceso existen registros de uso general que complementan el trabajo del Acumulador, que funcionan fundamentalmente como memoria, pero, con un nivel de jerarquía mayor que la memoria principal y que la memoria caché.

Los registros de la CPU son de dos tipos:

14.4.2.1. Registros visibles al usuario

Son aquellos registros que pueden ser accedidos por el usuario a través de programas de bajo nivel. Se clasifican básicamente en:

- **Registro de datos** (Memory Data Register, **MDR**), es un registro de poca capacidad pero de alta velocidad de transferencia. Contiene datos que se están leyendo ó que se escribirán en la memoria principal.
- **Registro de direcciones** (incluye puntero de pila, de segmento y Registro de índice).
- **Registro de condición** (incluye indicadores o “flags”)

14.4.2.2. Registros de control y de estado

Son aquellos registros transparentes al usuario. Esencialmente son cuatro los registros responsables de la ejecución de una instrucción.

- **Contador de Programa** (“Program Counter”, **PC**), contiene la dirección de la instrucción a ejecutar.
- **Registro de Instrucción** (“Instruction Register”, **IR**), contiene la instrucción recientemente ejecutada.
- **Registro de Dirección de Memoria** (“Memory Address Register, **MAR**), contiene la dirección de una posición de memoria.
- **Registro Intermedio de Memoria** (“Memory Buffer Register”, **MBR**), contiene el dato a escribir en memoria o el que se ha leído recientemente.

14.4.3. Unidad de Control

La Unidad de Control es un circuito secuencial cuyo trabajo consiste en recibir una instrucción, almacenarlo, decodificarlo y luego generar los niveles lógicos apropiados en las señales de control para que se ejecute ésta instrucción.

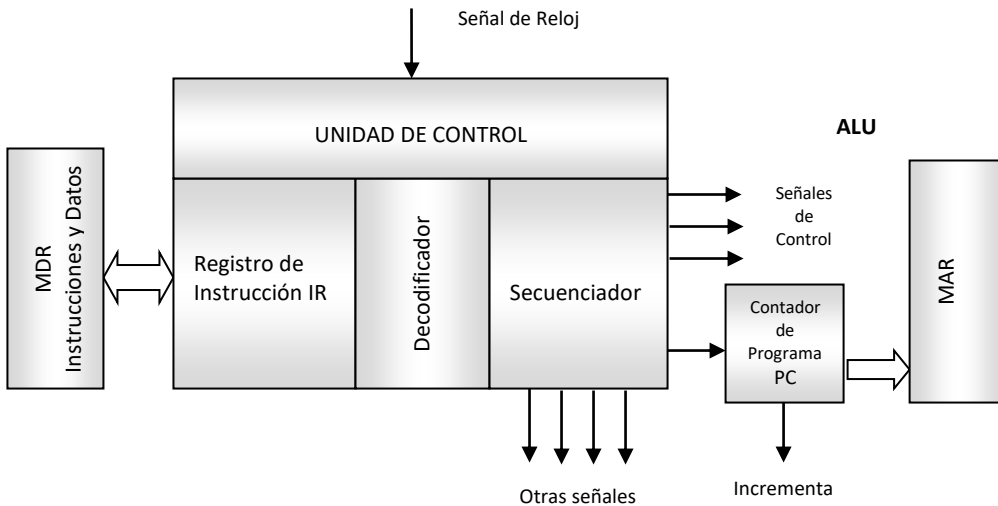


Fig. 14.7

Diagrama de bloques de la Unidad de Control

La unidad de control recibe instrucciones del Registro de Datos (MDR) y lo almacena en su Registro de Instrucción (IR) para luego ser decodificado y generar las señales de control apropiadas para que la ALU realice las operaciones asignadas. Asimismo, el secuenciador envía una señal al contador de programa para que incremente su valor, el que a su vez selecciona una dirección de otra instrucción en el Registro de Direcciones (MAR).

14.5. Funcionamiento del computador

La función básica que realiza un computador es la ejecución de programas. El programa a ejecutar está compuesto por un conjunto de instrucciones almacenadas en memoria. La (CPU) Unidad Central de Procesamiento, es precisamente la que se encarga de ejecutar las instrucciones especificadas en el programa.

Los principales componentes del computador interactúan para ejecutar un programa. El punto de vista más sencillo considera el procesamiento de una instrucción constituido por dos etapas: La CPU lee (capta la instrucción de memoria, y la ejecuta).

Fases de una Instrucción.

Son dos: Búsqueda y Ejecución.

Búsqueda: la unidad de control se encarga de hacerlo. La salida del Contador de Programa viaja por el bus de direcciones y su contenido indica la dirección de la Instrucción que se va a ejecutar, luego ésta dirección es recepcionada por la memoria en su Registro MAR, la memoria decodifica esta dirección y de la celda pertinente (matriz) se extrae información y se almacena en el MDR, el cual puede almacenar tanto instrucciones como datos (por el bus de datos puede viajar datos e instrucciones). La unidad de control toma cuenta del caso y envía ésta información directamente al registro de instrucción, luego la decodifica e incrementa el contador de programa. Esta fase es general para cualquier instrucción.

Ejecución: la ejecución del programa consiste en la repetición del proceso de captación de instrucción y ejecución de instrucción, en el que se genera diferentes secuencias de microinstrucciones y señales de control, los cuales viajan por todo el sistema hasta que se ejecuta completamente la macroinstrucción.

Por ejemplo: Si un dispositivo externo proporciona al ordenador la información sobre un número de piezas trabajado por una máquina A y este número se guarda en una posición de memoria, lo mismo sucede con una máquina B; la misión del ordenador es sumar estos 2 números y visualizar el resultado en una pantalla (display).

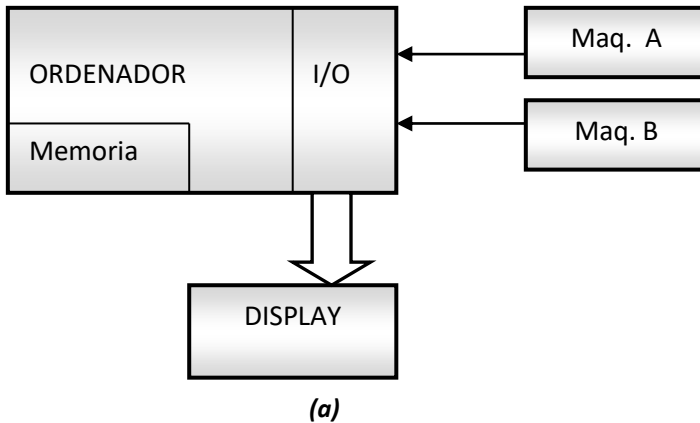
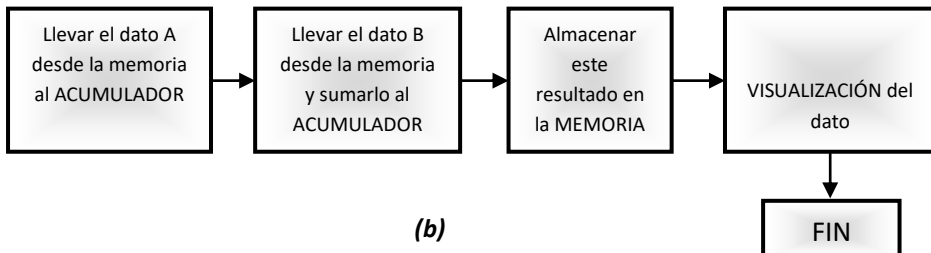


Fig. 14.8

(a) Diagrama de bloques del ingreso y salida de datos en un computador

(b) Diagrama de bloques de la ejecución de un programa.



Cada paso puede equivaler a 1, 2,3, ½, etc. Instrucciones.

Se define “carga del acumulador” a la acción de traer información desde la memoria y la almacenamos en el acumulador, la información que tenía el acumulador se pierde y la información que se trajo desde la memoria permanece aun en ella.

La operación inversa se llama almacenar en memoria y que consiste en que la información del acumulador se va a depositar en la memoria.

Fuente (conserva la información y de ella sale la información)

Destino (copia la información)

(cargar) LOAD LDA (cargar al acumulador)

(llevar) STURE STA (llevar del acumulador)

La información del contador de programa debe coincidir con la dirección.

La numeración de la dirección escrito en el sistema hexadecimal.

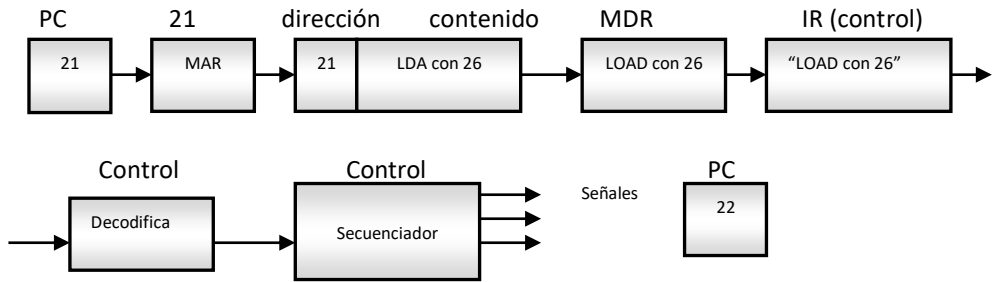
MAPA DE LA MEMORIA PRINCIPAL

Dirección	Contenido
MAR	MDR
21	LDA con 26
22	Sumar A con 27
23	STA a 28
24	Visualización
25	FIN
26	Dato A
27	Dato B
28	Resultado A+B

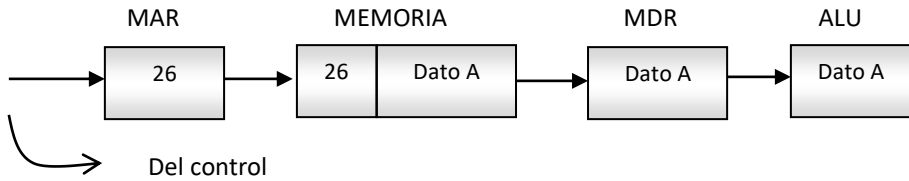
En el mapa de memoria se observa que el Registro de Direcciones (MAR) almacena posiciones de memoria, mientras que el Registro de Datos (MDR) almacena tanto instrucciones (LDA, sumar, STA, etc) como datos (Dato A, Dato B, Resultado A+B, etc)

En la fase de búsqueda inicia en el Contador de Programa **PC** indicando la dirección 21, y finaliza cuando el **PC** se incrementa a 22.

Fase de búsqueda



Fase de ejecución



Bibliografía

Angulo, José M. (2007). Sistemas Digitales y Tecnología de Computadores. España. Editorial Paraninfo.

Floyd, Thomas. (2000). Fundamentos de Sistemas Digitales. Séptima edición. México. Editorial Prentice Hall.

Fowler, Richard. (1994). Electricidad Principios y Aplicaciones. Primera edición. España. Editorial Reverté S.A.

García, Francisco. (2007). Prácticas Calificadas de Electrónica Digital. Facultad de Ciencias Físicas y Matemáticas. Universidad Nacional Pedro Ruíz Gallo.

Hermosa, Antonio. (1993). Electrónica Digital Fundamental. Barcelona. Editorial Marcombo.

Malvino, Albert y Leach, Donald. (1993). Principios y Aplicaciones Digitales. Cuarta edición. España. Editorial Marcombo.

Mano, Morris. (2005). Diseño Digital. Tercera edición. México. Editorial Pearson Educación.

Marcovitz, Alan. (2005). Diseño Digital. Segunda edición. México. Editorial Mc Graw Hill.

Novo Pío (2008). Lógica digital y microprogramable. Primera edición. España: Editorial Marcombo. Versión digital en www.e-libro.com.

Ojeda, Francisco. (1994). Problemas de Electrónica Digital. España. Editorial Paraninfo.

Ruiz, Francisco. (2007). Electrónica Digital Fácil Para Electricistas y Técnicos de Mantenimiento. México. Editorial Alfaomega.

Tocci, Ronald y otros. (2007). Sistemas digitales. Principios y aplicaciones. Décima edición. México. Editorial Pearson Educación.

Tokheim, Roger. (2008). Electrónica Digital. Principios y Aplicaciones. Sétima edición. México. Editorial McGraw-Hill.

Wakerly, Jhon. (2001). Diseño Digital. Principios y Prácticas. Tercera edición. México. Editorial Pearson Prentice Hall.

Zuloaga, José. (2009). Guía Didáctica de Electrónica Digital. Escuela Profesional de Ingeniería de Sistemas. Universidad Señor de Sipán.

Linkografía

Circuit Maker versión estudiantil, para diagramar y diseñar circuitos digitales

http://www.ciens.ucv.ve/ciencias/progr_aplic.htm

Circuitos varios

<http://www.electronred.es.vg/>

Circuitos y Sistemas Digitales

<http://apuntesgratis.oposicionesyempleo.com/electronicadigital.pdf>

Curso de electrónica digital

<http://www.unicrom.com/ElectronicaDigital.asp>

Manual de Circuitos Digitales I

<http://fcqi.tij.uabc.mx/docentes/jjesuslg/LAB-D1-00-2.PDF>

Manual de Prácticas

http://laimbio08.escet.urjc.es/docencia/EDII/EDI_II_Manual_practicas.pdf

Manual de Prácticas

<http://www.hspdigital.org/Documentos/HSP/EDII/LabED-II.pdf>

Prácticas de Electrónica Digital

<http://olmo.pntic.mec.es/jmarti50/digital/index.html>

Simbología de electrónica digital

<http://www.simbología-electrónica.com/simbolos/digital.htm>

Descubre tu próxima lectura

Si quieres formar parte de nuestra comunidad, regístrate en <https://www.grupocompas.org/suscribirse> y recibirás recomendaciones y capacitación



   @grupocompas.ec
compasacademico@icloud.com

José Fortunato Zuloaga Cachay

Ingeniero de Sistemas y Computación, Universidad Católica Santo Toribio de Mogrovejo, Chiclayo, Perú. Licenciado en Física, Universidad Nacional Pedro Ruíz Gallo, Lambayeque, Perú. Maestro en Educación con mención en Docencia y Gestión Educativa. Universidad César Vallejo, La Libertad, Perú. jzuloaga@usat.edu.pe, ORCID: 0000-0003-2363-0817, https://scholar.google.com/citations?hl=es&pli=1&user=-aIDq_QAAAAJ

Consuelo Ivonne Del Castillo Castro

Ingeniera en Computación e Informática, Universidad Nacional Pedro Ruíz Gallo, Lambayeque, Perú. Maestra en Ingeniería de Sistemas, Universidad César Vallejo, Lambayeque, Perú, cdelcastilloc@unprg.edu.pe, ORCID: 0000-0002-1512-006X, <https://scholar.google.com/citations?hl=es&user=cbZ6hUAAAAAJ>

Gilberto Carrión Barco

Ingeniero en Computación e Informática, Universidad Nacional Pedro Ruíz Gallo, Lambayeque, Perú. Licenciado en Administración Pública, Universidad Señor de Sipán. Magister en Docencia Universitaria, Universidad César Vallejo, La Libertad, Perú. Maestro en Ingeniería de Sistemas, Universidad Nacional Pedro Ruíz Gallo, Lambayeque, Perú. Maestro en Gestión Pública, Universidad César Vallejo, Lambayeque, Perú. Doctor en Ciencias de la Computación y Sistemas, Universidad Señor de Sipán, Lambayeque, Perú. gcarrion@unprg.edu.pe, ORCID: 0000-0002-1104-6229, <https://scholar.google.es/citations?user=5NTGy7kAAAAJ&hl=es>

Freddy William Campos Flores

Ingeniero en Computación e Informática, Universidad Nacional Pedro Ruíz Gallo, Lambayeque, Perú. Maestro en Ingeniería de Sistemas, Universidad César Vallejo, Lambayeque, Perú, fcampos@unprg.edu.pe, ORCID ID: 0000-0002-9624-2930, <https://scholar.google.es/citations?user=S1vZSJMAAAAJ&hl=es>

Lucía Isabel Chamán Cabrera

Ingeniera Electrónica, Universidad Privada Antenor Orrego, Trujillo, Perú. Maestría en Administración con Mención en Gerencia Empresarial, Universidad Nacional Pedro Ruíz Gallo, Lambayeque, Perú, luciachc81@gmail.com, ORCID: 0000-0003-2901-1427, <https://scholar.google.es/citations?user=vFZcls4AAAAJ&hl=es>



@grupocompas.ec
compasacademico@icloud.com



Organización
de las Naciones Unidas
para la Educación,
la Ciencia y la Cultura



Cátedra UNESCO
«Paz, Solidaridad y
Diálogo Intercultural»
Universitat Abat Oliba CEU



Grupo de capacitación e investigación pedagógica



@grupocompas.ec
compasacademico@icloud.com