

GOOGLE

La historia de Google empieza en 1995 cuando Larry Page y Sergey Brin se reúnen en Stanford.

Un año después Larry y Sergey, ahora los estudiantes de Stanford ciencias de la computación graduado, han comenzado a colaborar en un motor de búsqueda llamado BackRub. BackRub ha operado en los servidores de Stanford por más de un año-finalmente ha ocupado demasiado ancho de banda para adaptarse a la universidad.

En 1997 Larry y Sergey han decidido que el buscador BackRub necesita un nuevo nombre. Después de algunas ideas, que iban con Google, un juego de la palabra “googol”, un término matemático representado por el número 1 seguido por 100 ceros.



En agosto de 1998 cofundador de Sun, Andy Bechtolsheim escribió un cheque por 100.000 dólares a una entidad que no había existido todavía: una empresa llamada Google Inc. En próxima mes Google ha creado espacio de trabajo en el garaje de Susan Wojcicki a 232 de Santa Margarita, de Menlo Park. También en este mes Larry y Sergey han contratado a Craig Silverstein como su primer empleado. Él es un estudiante graduado de ciencias de la computación en Stanford.

En noviembre de 1999 Charlie Ayers se ha incorporado como primer chef de Google. Ha ganado el trabajo de un cocinero-fuera juzgado por 40 empleados de la empresa.

En próximo año, en el mayo fueron los primeros 10 versiones lingüísticas de Google.com: francés, alemán, italiano, sueco, finlandés, español, portugués, holandés, noruego y danés. En este mes también fueron primeros Premios Web para Google: Logro Técnico (votado por los jueces) y la voz de los Pueblos (votado por los usuarios).

Un mes más tarde Google ha signado una alianza con Yahoo para convertirse en su proveedor de búsqueda predeterminado. Además se ha anunciado el primer índice de mil millones de URL, por lo que Google se ha convertido en el mayor motor de búsqueda. En el septiembre número total de idiomas ha aumentado a 15 porque búsqueda ha estado accesible en chino, japonés y coreano. En el diciembre Google Toolbar fue puesto en libertad. Es un navegador plug-in que permite la búsqueda sin tener que visitar la página principal de Google.

En marzo de 2001 Google.com ya estuvo disponible en 26 idiomas y en próximos meses también búsqueda de imágenes fue creado y ha ofrecido acceso a 250 millones de imágenes. Además en agosto han abierto primera oficina internacional en Tokio, en cambio Larry y Sergey estaban nombrados presidentes de los productos y la tecnología.

En principio de 2004 el índice de búsqueda ha realizado un nuevo hito: 6 millones de artículos, incluyendo 4280 millones las páginas web y las imágenes de 880 millones.

En el marzo ha introducido Google Local, que ha ofrecido listados de negocios pertinentes barrio, mapas y direcciones. (Más tarde, local se combina con Google Maps).

En el octubre fue presentado Google Desktop Search. Esto permite buscar archivos y documentos almacenados en el disco con la tecnología de Google. También ha adquirido Keyhole, una compañía de cartografía digital, cuya tecnología se convertirá más tarde en Google Earth.

Ahora Google tiene muchos productos, por ejemplo: Google Search, Google Reader, Gmail, Google Maps, Google News, Google SketchUp, Google Video, Google Earth, Google Wave, AdSense, AdWords, YouTube, Picasa, Google Chrome, Google Analytics, Google Buzz, Google Desktop, Picnik, Google Product Search, Google Alerts, Google Base, Blogger, Google Blog Search, Google Bookmarks, Google Checkout, Google CSE, Google Code, Google Code Search, Google Compute, Google Desktop.

HISTORIA DE GOOGLE VIDEO

→ <https://www.youtube.com/watch?v=oJLhobZ97q8>

CÓDIGO ENIGMA

La máquina Enigma utilizada por la mayor parte de las comunicaciones alemanas durante la guerra tenía un funcionamiento complejo. Se basaba en cinco rotores que variaban cada vez que se pulsaba una tecla, de manera que cada letra del alfabeto ofrecía un número altísimo de posibilidades. El Ejército alemán complicaba más las cosas cambiando la posición de los rotores una vez al mes. Los mandos alemanes de la época veían a Enigma como indescifrable.

Poco antes del estallido de la guerra, el GC&CS británico se reunió con el Servicio de Inteligencia polaco, que también estaba intentando desentrañar el código Enigma. A partir de la información recibida, Turing empieza a trabajar con otro enfoque, mejorando el método polaco. Su tarea, junto a la de otros criptoanalistas de perfiles variados, se llevaba a cabo en la mansión de Bletchley Park, situada en el condado de Buckinghamshire, en plena campiña inglesa, donde llegaron a trabajar hasta 10.000 personas.



En tres meses desde que recibiera las informaciones del Servicio polaco, Turing fue capaz de descifrar el código alemán. No era suficiente, había que hacerlo de forma rápida, automatizar el proceso. Para ello el matemático diseñó junto a su colega de Cambridge, Gordon Welchman, su propia máquina para contrarrestar la potencia de Enigma, la llamada 'bomba'. El sistema se servía de análisis matemáticos para determinar cuáles eran las posiciones más factibles de los rotores, para que se pudieran probar lo antes posible.

Las máquinas se empezaron a construir en fábrica y entraron en funcionamiento en la primavera de 1940. Durante los meses de verano, jugaron un papel determinante descifrando los mensajes de la

fuerza aérea alemana, que atacó instalaciones militares y ciudades por toda Gran Bretaña. En Bletchley Park un ejército de 'bombas' trabajaba en la retaguardia inglesa para ayudar a la supervivencia de la nación. En 1943 las 'bombas' ya descifraban un total de 84.000 mensajes de Enigma al mes.

Pero los mensajes más importantes para Gran Bretaña eran los que enviaban los submarinos alemanes, que operaban en el Atlántico Norte, y éstos utilizaban un sistema Enigma con una mayor seguridad.

Las comunicaciones que utilizaban los submarinos alemanes se basaban en una máquina Enigma que tenía ocho rotores en lugar de cinco. Las posibles combinaciones aumentaban exponencialmente respecto al código anterior, el que abarcaban las 'bombas'. Turing se centró en la variante naval y, tras la captura de unos documentos alemanes, en los que estaban anotadas las claves para el mes de febrero de 1941, el matemático y su equipo pudieron reconstruir el sistema usado por los alemanes.

Con la ayuda de nuevos documentos, los investigadores del barracón 8, cuyos trabajos Turing dirigía, descifraron el Enigma de los submarinos alemanes. Para que las 'bombas' funcionaran con este código, Turing desarrolló una técnica estadística que permitía conocer la identidad de cada rotor en la máquina cifradora, antes de aplicar su propia máquina descifradora al mensaje.

Cuando en febrero de 1942 los U-Boot complicaron su sistema de transmisiones, en Bletchey Park tuvieron que empezar el trabajo de nuevo. Independientemente del número de rotores que conformaban las máquinas Enigma, en cada comunicación sólo se usaban tres de ellos. La marina alemana comenzó a utilizar uno más en cada mensaje; en total, cuatro. El quebradero de cabeza duró hasta diciembre de 1942, cuando se volvió a descifrar el código.

CÓMO FUNCIONA LA MAQUINA ENIGMA VIDEO:

→ <https://www.youtube.com/watch?v=pWVt3cmO0Tw>

STEVE JOBES VS GATES --- MENTES BRILLANTES

COMPUTADORAS CENTRALES

Altair 8800, perforan papeles, es una forma de comando. Steve Jobs, su amigo, creo algo parecido a lo de Altair pero le agrego "Z" componentes, pantalla y teclado, Steve interesado en vender y su amigo, en crear, ambos se necesitaban.

Steve crea Apple al inicio ellos sufren de fracaso y él empieza a imaginar una nueva computadora a la que llaman Apple Z (1977) contaba con 8 millones de dólares en venta.

Microsoft inicia un software de negocios en 1980 y venden el software a clientes de Apple, Gates quería hacer el software principal de la PC llevando a un trato con IBU, Microsoft TOC paga porque formaran un S.O. que adquiere derecho para formar.

Jobs en 1980, las computadoras de más populares a la venta, ellos entran a las instalaciones de xeros, crean un equipo para integrar a la computadora, en 1983 el ex de la gerencia de PEPSI se

integra a Apple, Jobs invita a --- a integrarse en ambos, a finales de 1983 Microsoft se convierte en Windows, Gates se convierte en rival de Jobs.

En 1984, se inventa a "Macintosh" y Jobs tiene a esta máquina pero era lenta y Apple fue perdiendo sus ventas. Sus aliados votaron porque él cediera su lugar a otro, por lo cual este queda fuera de su propia compañía.

Gates, Microsoft se convierte en el sistema operativo, Windows se apodera y agrega todas las aplicaciones, aunque la gente no quería.

En 1997, Apple estuvo a punto de bancarota, a 12 años la junta le pide a Steve Jobs y este pide ayuda a Gates todo por seguir en el juego. Microsoft invierte dinero en Apple.

En 2001 Steve Jobs crea iPhone la cual revolucionara la era de los celulares y Gates se somete a la filantropía.

Jobs comienza a padecer cáncer de páncreas la cual pide que en zoll deje la compañía Apple queda como la compañía más famosa, Jobs y Gates se vuelven a aliar.

MICROSOFT VS APPLE VIDEO:

→ <https://www.youtube.com/watch?v=OtPY-yMLpwA>

CONFERENCIA (TECNOLOGÍA EN MÉXICO)

NUEVA REVOLUCIÓN 4.0

- Velocidad
- El Alcance
- El impacto en los sistemas

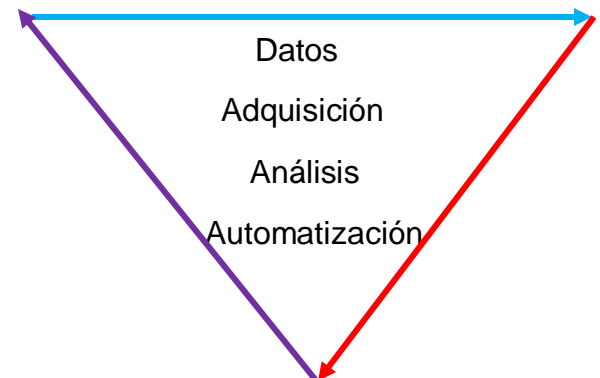
Es un cambio de paradigma que está cambiando en lo que somos es el encuentro de los desarrollos tecnológicos para poder conectarse.

ECONOMIA DIGITAL

Datos → Información → Conocimiento → Sabiduría →

MACROTENDENCIAS

- Digitalización
- Convergencia de Tecnologías
- Convergencia de Mercado
- Ecosistema Digital



SECTOR DIGITAL

- Globalización
- Competencia
- Políticas Públicas

EMPODERAMIENTO DE LA DEMANDA

- Economía Digital
 - ❖ Ecosistema de Plataformas
 - ❖ Generalización de las Plataformas
 - ❖ Economía del Dato

ALGUNAS MACROTENDENCIAS

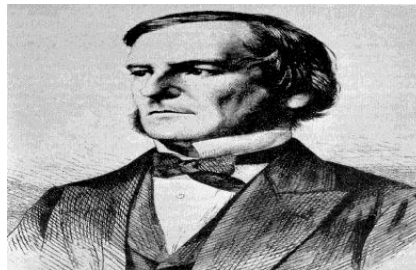
- Cuarta Revolución Industrial
- Sharing Economy
- Fintech

INTERCONECTIVIDAD

- Aumento de suscripciones de banda ancha inalámbrica

APORTACIONES DE GEORGE BOOLE EN EL ALGEBRA BOOLEANA

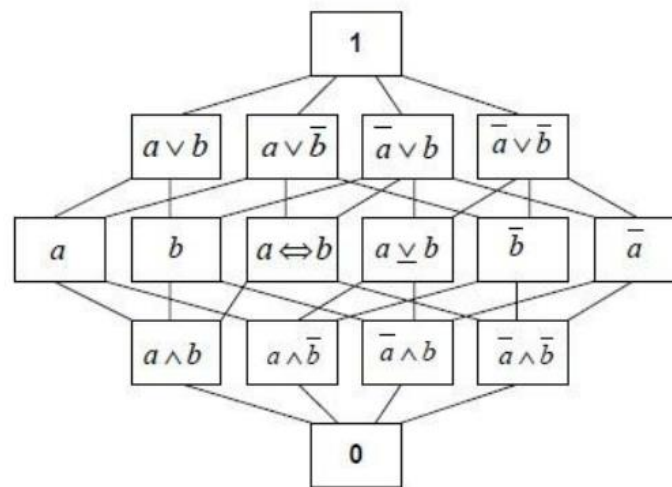
George Boole nació en una familia muy humilde y modesta de Lincoln (Inglaterra). Siendo el primero de cuatro hermanos, se vio obligado a ayudar a sus progenitores desde edades muy tempranas. Su padre, John Boole, era también un zapatero autodidacta, que había aprendido nociones básicas de matemáticas por su cuenta. Él mismo construyó un telescopio con sus propias manos que exhibía orgulloso en el escaparate de la tienda, y fue también quien introdujo al joven George al mundo de las matemáticas y de la óptica. Siendo un adolescente, el célebre investigador también aprendería por su cuenta latín, griego, francés y alemán.



Con el objetivo de ayudar a su familia, George Boole decidió dedicarse inicialmente a la enseñanza. En 1835 fundó su primer colegio, donde aprovechaba los ratos libres para estudiar los trabajos realizados por famosos matemáticos como Laplace y Lagrange. Posteriormente, George Boole se dedicó al álgebra y fue en este área en la que realizó sus mejores contribuciones. El primer trabajo lo publicó en la revista Transaction of the Royal Society. Sería solo el principio de una extraordinaria carrera investigadora, a pesar de su corta duración, pues falleció con solo 49 años.

Las aportaciones que hizo George Boole son :

- Incorporó la lógica (disciplina tradicionalmente dentro de la filosofía) al campo de las matemáticas, estableciendo asimismo sus reglas y axiomas más importantes.
- Creó los fundamentos de la lógica matemática mediante el álgebra que lleva su nombre.
- Dio a entender que existían "clases" que podían agrupar a todos los individuos definidos por una palabra.
- Planteó que podrían utilizarse "xy" para nombrar a la clase de cosas que pertenecen al mismo tiempo al grupo "x" y a "y".
- El álgebra decimonónica del pensamiento de Boole se encuentra detrás de las computadoras electrónicas y es la fuerza de empuje detrás de la inteligencia artificial".



SECCIÓN DE SISTEMAS DE NUMERACIÓN

En la informática se usaron muchos sistemas de numeración como lo fue el sistema binario, decimal, octal y hexadecimal ya que fueron muy útil para la realización de varios programas pero la tecnología ha avanzado tanto que ya estos sistemas están si se puede decir obsoleto. Para la realización de estos programas se tenía que realizar algunas conversiones, de lo cual se les explicara algunos de ellos

En cuanto al software libre suele estar disponible gratuitamente en Internet, o a precio del coste de la distribución a través de otros medios; sin embargo no es obligatorio que sea así y, aunque conserve su carácter de libre, puede ser vendido comercialmente.

Existen diversos tipos de software libre entre ellos están el colibrí y el linux.

SISTEMA DE NUMERACIÓN

- Sistema Binarios:

Es el sistema de numeración que utiliza internamente hardware de las computadoras actuales. Se basa en la representación de cantidades utilizando los dígitos 1 y 0, por tanto su base es dos (número de dígitos de sistemas). Cada dígito de un número representado en este sistema se representa en BIT (contracción de binary digit).

Suma Binaria: Es semejante a la suma decimal, con la diferencia de que se manejan solo dos dígitos (0 y 1), y que cuando el resultado excede de los símbolos utilizados se agrega el exceso (acarreo) a la suma parcial siguiente hacia la izquierda. Las tablas de sumar son:

Tabla del 0 Tabla del 1
0 + 0 = 0 1 + 0 = 1
0 + 1 = 1 1 + 1 = 10 (0 con acarreo 1)

- Sistema Octal: Es sistema de numeración cuya base es 8, es decir, utiliza 8 símbolos para la representación de cantidades. Estos sistemas es de los llamados posicionales y la posición de sus cifras se mide con la relación a la coma decimal que en caso de no aparecer se supone implícitamente a la derecha del número. Estos símbolos son:

0 1 2 3 4 5 6 7

- Sistema Decimal: Es uno de los sistema denominado posicionales, utilizando un conjunto de símbolos cuyo significado depende fundamentalmente de su posición relativa al símbolo, denominado coma (,) decimal que en caso de ausencia se supone colocada a la derecha. Utiliza como base el 10, que corresponde al número del símbolo que comprende para la representación de cantidades; estos símbolos son:

0 1 2 3 4 5 6 7 8 9

- Sistema Hexadecimal: Es un sistema posicional de numeración en el que su base es 16, por tanto, utilizara 16 símbolos para la representación de cantidades. Estos símbolos son:

0 1 2 3 4 5 6 7 8 9 A B C D E F

SÍMBOLOS VALOR ABSOLUTO
A 10
B 11
C 12
D 13
E 14
F 15

3.- DEFINICIÓN Y CONVERSIÓN ENTRE LAS UNIDADES BIT, BYTES, KILOBYTES, MEGABYTES, GIGABYTES Y TERABYTES

- BIT

Un BIT es una manera "binaria " de presentar información; es decir, expresa una de solamente dos alternativas posibles. Se expresa con un 1 o un 0, con un sí o no, verdadero o falso, blanco o negro, algo es o no es, voltaje o no voltaje, un nervio estimulado o un nervio inhibido. (Sabemos que no todo lo que se encuentra en nuestro universo es blanco o negro, pero aún así podemos utilizar esta forma binaria de representación para expresar estados intermedios logrando la precisión deseada).

- BYTE

Es la unidad de información formada por ocho bits (01011101). Según cómo estén combinados los bits (ceros o unos), formaran un bytes dependiendo de la cantidad de bytes, formarán kilobytes, un megabytes, gigabytes, etc. Relacionados: Nibble que equivale a medio bytes; DBCS: es el conjunto de caracteres que necesitan dos bytes para aparecer.

- KILOBYTE

Unidad de medida de la cantidad de información en formato digital. Un byte consiste de 8 bits. Un BIT es un cero (0) o un uno (1). Por lo tanto un ejemplo de un byte es 01001001. Esa secuencia de números (byte) puede simbolizar una letra o un espacio. Un kilobytes (Kb) son 1024 bytes y un Megabytes (Mb) son 1024 Kilobytes

- MEGABYTE

El Megabytes (MB) es una unidad de medida de cantidad de datos informáticos. Es un múltiplo binario del byte, que equivale a 220 (1 048 576) bytes, traducido a efectos prácticos como 106 (1 000 000) bytes.

- GYGABYTE

Es una unidad de almacenamiento. Existen dos concepciones de gigabytes (GB). (Debemos saber que un byte es un carácter cualquiera) Un gigabytes, en sentido amplio, son 1.000.000.000 bytes (mil millones de bytes), ó también, cambiando de unidad, 1.000 megas (MG o megabytes). Pero si somos exactos, 1 GB son 1.073.741.824 bytes ó 1.024 MB.

- TERABYTE

Una unidad de almacenamiento tan desorbitada que resulta imposible imaginársela, ya que coincide con algo más de un trillón de bytes. Un uno seguido de dieciocho ceros.

Conversiones

File Storage Capacity by Bits and Bytes					
	Bit	byte	Kilobyte	Megabyte	Gigabyte
bit	1	8	8,192	8,388,608	8,589,934,592
byte	8	1	1,024	1,048,576	1,073,741,824
Kilobyte	8,192	1,024	1	1,024	1,048,576
Megabyte	8,388,608	1,048,576	1,024	1	1,024
Gigabyte	8,589,934,592	1,073,741,824	1,048,576	1,024	1
Terabyte	8,796,093,022,208	1,099,511,627,776	1,073,741,824	1,048,576	1,024
Petabyte	9,007,199,254,740,990	1,125,899,906,842,620	1,099,511,627,776	1,073,741,824	1,048,576
Exabyte	9,223,372,036,854,780,000	1,152,921,504,806,850,000	1,125,899,906,842,620	1,099,511,627,776	1,073,741,824
Zetta byte	9,444,732,965,739,290,000,000	1,180,591,620,717,410,000,000	1,152,921,504,606,850,000	1,125,899,906,842,620	1,099,511,627,776

4.- UNIDADES DE MEDIDA (Definición y Ejemplo):

- HZ:

El hercio es la unidad de frecuencia del Sistema Internacional de Unidades. Proviene del apellido del físico alemán Heinrich Rudolf Hertz, descubridor de la transmisión de las ondas electromagnéticas. Su símbolo es hz. (Que se escribe sin punto). En inglés se llama Hertz (y se pronuncia /jérts/).

- MHZ:

Megahertzio, múltiplo del hertzio igual a 1 millón de hertzios. Utilizado para medir la "velocidad" Megahertzios, es una medida de frecuencia (número de veces que ocurre algo en un segundo). En el caso de los ordenadores, un equipo a 200 MHz será capaz de dar 200 millones de pasos por segundo.

En la velocidad real de trabajo no sólo influyen los MHz, sino también la arquitectura del procesador (y el resto de los componentes); por ejemplo, dentro de la serie X86, un Pentium a 60 MHz era cerca del doble de rápido que un 486 a 66 MHz "bruta" de los microprocesadores.

- **NANOSEGUNDO:**

Un nanosegundo es la milmillonésima parte de un segundo, 10^{-9} . Este tiempo tan corto no se usa en la vida diaria, pero es de interés en ciertas áreas de la física, la química y en la electrónica. Así, un nanosegundo es la duración de un ciclo de reloj de un procesador de 1 GHz, y es también el tiempo que tarda la luz en recorrer aproximadamente 30 cm.

- **MILISEGUNDO:**

Un milisegundo es el período de tiempo que corresponde a la milésima fracción de un segundo (0,001s).

- **MICROSEGUNDO:**

Una millonésima de segundo microsegundo es la millonésima parte de un segundo, 10^{-6} seg

5.- DEFINICION Y TERMINOS DE USO DEL SOFTWARE LIBRE. MENCIONE AL MENOS 2 SOFTWARE LIBRES ACTUALES, SU UTILIDAD Y CARACTERISTICAS FUNDAMENTALES.

- **SOFTWARE LIBRE:**

Software libre es el software que, una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente. El software libre suele estar disponible gratuitamente en Internet, o a precio del coste de la distribución a través de otros medios; sin embargo no es obligatorio que sea así y, aunque conserve su carácter de libre, puede ser vendido comercialmente. ...

- **SOFTWARE LIBRES ACTUALES, SU UTILIDAD Y CARACTERISTICAS:**

Colibrí

- Desarrollado por: Comunidad de Usuarios de Software Libre en Colombia
- Tipo de Recurso: Sitio Web
- Tipo de Destinatario: General
- Tipo Acceso (costo): Libre - Gratuito - Fuente Abierta – GNU

Colibrí es una organización democrática que busca reunir a los interesados en el software libre en Colombia: Documentos, eventos y noticias sobre este tipo de programas en Colombia.

En la sección documentos encontrará un listado de preguntas frecuentes sobre el uso de software libre, así como el proyecto de ley que se está impulsando en este país.

LYNX:

- Desarrollado por: Equipo Interdisciplinario
- Tipo de Recurso: Artículo.

- **Tipo de Destinatario:** General
- **Tipo Acceso (costo):** Libre - Gratuito - Fuente Abierta – GNU

SISTEMAS DE NUMERACIÓN VIDEO

→ <https://www.youtube.com/watch?v=4kenXkQBIEw>

CÓDIGOS BINARIOS

Todo lo que hemos comentado hasta ahora se refiere al llamado código binario natural. Sin embargo existen distintos tipos de códigos binarios, que son empleados según favorezcan el proceso que se va a realizar con ellos, ya que son más eficientes en la ejecución de determinadas operaciones. Veamos los más importantes:

Código binario natural.

Consiste en representar directamente el número decimal en binario, o lo que es lo mismo cada número decimal tiene su correspondiente en binario. Ejemplo $35_{(10)} = 100011_{(2)}$

Códigos BCD (Decimal codificado en binario).

Con estos códigos, para representar un número decimal en binario, se transforman cada una de las cifras que constituyen el número decimal separadamente, en el caso anterior el número 35, transformado en binario sería el resultado de transformar primero 3, y después 5.

Código BCD natural (8421).

Es un código con peso o ponderado, es decir el número decimal equivalente es el resultado de sumar el valor de la posición de los dígitos binarios que constituyen el código. En este código los pesos de los dígitos son las potencias sucesivas de dos, es decir 2^3 (8), 2^2 (4), 2^1 (2) y 2^0 (1), como se puede observar de ahí viene el nombre del código. Es decir se transcriben las cifras decimales a binario y viceversa, según la posición que ocupan. Ejemplo $35 = 0011\ 0111$. Es decir 3 (0011) y 5 (0101).

Código Aiken (2421).

También es un código ponderado, aunque ahora los pesos de las cifras según su posición serán: 2, 4, 2 y 1. Ejemplo $35 = 0011\ 1011$. Es decir 3 (0011) y 5 (1011).

Código exceso tres.

Éste es un código no ponderado, para obtener la relación entre el código decimal y este binario, se suma a cada dígito tres y después se convierte el resultado obtenido en binario 8421. Ejemplo $35 = 0110\ 1000$. Es decir 3 (0110) y 5 (1000).

Código exceso tres paridad impar.

En ocasiones se utilizan códigos que son especialmente útiles para algún cometido concreto, esto sucede con el código que vamos a analizar, se emplea para detectar si ha habido algún error en la transmisión de los datos codificados, de modo que emplea cinco dígitos en lugar de cuatro, pero de ellos el primero es un bit de paridad, para obligar a que cada grupo de cinco bits tenga un número impar de unos; si esto es así, es porque el dato transmitido es correcto, y entonces se procesa la información transmitida que es la que resulta de decodificar los cuatro últimos bits. Ejemplo 35 = 1 0110 0 1000. Es decir 3 (1 0110) y 5 (0 1000).

La siguiente tabla muestra la equivalencia entre los distintos códigos binarios analizados.

Decimal	BCD natural (8421)	BCD Aiken (2421)	Exceso 3	Exceso 3 Paridad impar
0	0000	0000	0011	1 0011
1	0001	0001	0100	0 0100
2	0010	0010	0101	1 0101
3	0011	0011	0110	1 0110
4	0100	0100	0111	0 0111
5	0101	1011	1000	0 1000
6	0110	1100	1001	1 1001
7	0111	1101	1010	1 1010
8	1000	1110	1011	0 1011
9	1001	1111	1100	1 1100

Código dos entre cinco.

En los años 40 se utilizó un código más sofisticado llamado dos-entre-cinco, que se basa en que cada conjunto de cinco bits (llamado penta-bit) debe tener únicamente dos unos, uno entre los dos primeros bits y otro en los tres últimos, de forma que se podría detectar posibles errores cuando cada pentabit no cumple esta condición.

También existen otros códigos que no solo son capaces de detectar errores, sino que también son capaces de corregirlos, como es el código de Hamming, formados por siete bits y que es probablemente el más empleado de este tipo.

Otro tipo de códigos son los que tienen la característica de que entre una combinación y la siguiente solamente difieren en un bit, a los códigos que tienen esta característica se les llama progresivos, como es el caso del código Gray.

Se llaman códigos reflejados, aquellos que tienen la característica de la combinación de dos números decimales que sumen nueve, se escriben igual sin más que cambiar 1 por 0 y 0 por 1. Esta característica la tienen los códigos Aiken y exceso tres, como puedes comprobar observando la tabla anterior.

También existen códigos capaces de transmitir información no solo numérica, sino letras, símbolos, operaciones,... de ellos el más común es el código ASCII (American Standard Code for Information Interchange), que es el empleado por casi todos los sistemas informáticos.

Se creó en 1963 por ASA (Comité Estadounidense de Estándares) para reordenar y expandir los caracteres y símbolos empleados en telegrafía por la compañía Bell, en la actualidad tiene 5852 combinaciones diferentes.

CÓDIGOS BINARIOS VIDEO

→ <https://www.youtube.com/watch?v=f9b0wwhTmeU>

ALGEBRA BOOLEANA

Es una rama especial del álgebra que se usa principalmente en electrónica digital. El álgebra booleana fue inventada en el año 1854 por el matemático inglés George Boole.

El álgebra de Boole es un método para simplificar los circuitos lógicos (o a veces llamados circuitos de conmutación lógica) en electrónica digital.

Por lo tanto, también se llama como "Cambio de álgebra". Podemos representar el funcionamiento de los circuitos lógicos utilizando números, siguiendo algunas reglas, que son bien conocidas como "Leyes del álgebra de Boole".

También podemos hacer los cálculos y las operaciones lógicas de los circuitos aún más rápido siguiendo algunos teoremas, que se conocen como "Teoremas del álgebra de Boole". Una función booleana es una función que representa la relación entre la entrada y la salida de un circuito lógico.

La lógica booleana solo permite dos estados del circuito, como True y False. Estos dos estados están representados por 1 y 0, donde 1 representa el estado "Verdadero" y 0 representa el estado "Falso".

Lo más importante para recordar en el álgebra de Boole es que es muy diferente al álgebra matemática regular y sus métodos. Antes de aprender sobre el álgebra de Boole, vamos a contar un poco sobre la historia del álgebra de Boole y su invención y desarrollo.

La síntesis lógica de las herramientas modernas de automatización electrónica se representa de manera eficiente mediante el uso de funciones booleanas conocidas como "Diagramas de decisión binarios".

El álgebra de Boole permite solo dos estados en un circuito lógico, como True y False, High and Low, Yes y No, Open and Close o 0 y 1.

LEYES FUNDAMENTALES

OR	AND	NOT
$A + 0 = A$	$A + 0 = 0$	$A'' = A$
$A + 1 = 1$	$A + 1 = A$	Los dos puntos en la A corresponde a dos barras de negación.
$A + A = A$	$A + A = A$	
$A + A = 1$	$A + A = 0$	

LEYES CONMUTATIVAS

$A + B = B + A$
$A \cdot B = B \cdot A$

LEYES ASOCIATIVAS

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

LEYES DISTRIBUTIVAS

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

OTRAS IDENTIDADES ÚTILES

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

ALGEBRA BOOLEANA → https://www.youtube.com/watch?v=6fXE2_67z_Q

CIRCUITOS MSI Y LSI

CIRCUITOS MSI

Son circuitos de media escala de integración, y contienen entre 10 y 100 puertas lógicas, o de 100 a 1000 transistores. Este tipo de circuitos se pueden clasificar según la función que cumplen dentro de los sistemas digitales en los siguientes tipos:

DE COMUNICACIÓN

Transmiten y modifican información, como los codificadores, decodificadores, multiplexores y demultiplexores.

ARITMÉTICOS

Operan con los datos binarios que procesan, como los sumadores y semisumadores, los comparadores y los restadores.

Circuitos MSI de comunicación

a) Codificadores:

Son dispositivos que transforman una señal expresada en código humano a código binario, y el cual se denominará completo si las entradas son iguales a 2 elevado a la cantidad de salidas (n), e incompleto si las entradas son menores a dicha cantidad. Por otra parte, un codificador puede ser de dos tipos: **con prioridad** y **sin prioridad**; en el primer caso, será capaz de atender a varias entradas simultáneas y determinar el criterio que da prioridad a una señal u otra, mientras que al ser "sin prioridad", sólo aceptará una entrada a la vez.



b) Decodificadores:

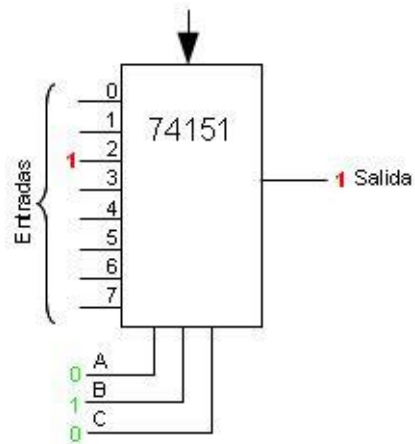
Son la función inversa de un codificador, los cuales disponen de n entradas y 2^n salidas y que, atendiendo a su salida, pueden ser **activos a nivel alto ("1")** o **activos a nivel bajo ("0")**. La mayoría de los decodificadores integran un dispositivo de control mediante una entrada "Enable" o "Habilitada" (la cual puede estar activa a cualquiera de sus dos niveles), la cual en caso de no tener lugar no permitiría la activación del decodificador.

Por otra parte, los decodificadores pueden dividirse en dos tipos: **Excitadores**, para los que a partir de las diferentes entradas posibles se activa más de una salida, siendo estas salidas capaces de dar suficiente corriente como para atacar a otros CI de la misma familia y a otros dispositivos, como displays, lámparas, relés, transductores, entre otros; los **No excitadores**, por su parte, son un tipo de decodificador cuyas salidas solo pueden acoplarse a otros circuitos digitales de la misma familia, ya que dan una corriente muy pequeña, incapaz de activar algún otro componente.



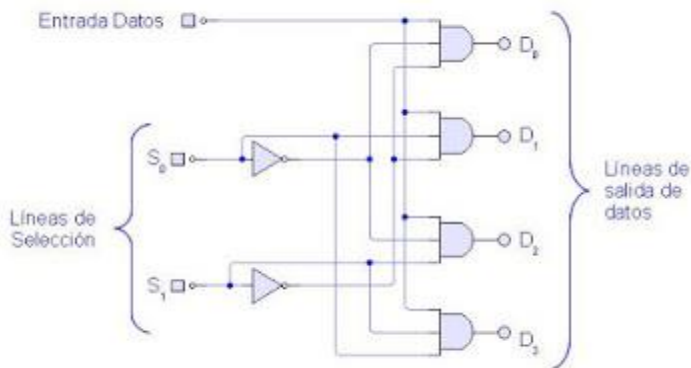
c) Multiplexores:

Son equivalentes a un interruptor mecánico giratorio de varias posiciones, el cual permite dirigir la información binaria procedente de diversas fuentes, a una única línea de salida, para ser transmitida a través de ella a un destino común.



d) Demultiplexores:

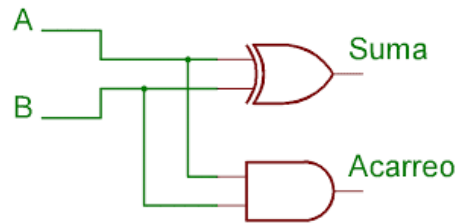
Son circuitos que realizan la función opuesta a un multiplexor, es decir, tiene una única entrada de datos, "n" entradas de selección y un número de salidas menor al 2 elevado al número de entradas de selección, de modo que según se introduzca una u otra combinación por las entradas de selección, se conseguirá comunicar la entrada de datos con la salida seleccionada.



Circuitos MSI aritméticos

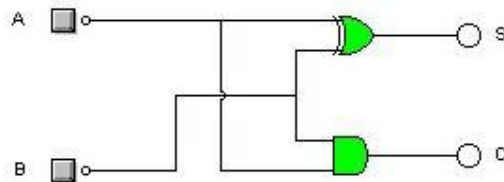
a) Sumadores:

Son circuitos que realizan la suma de dos palabras binarias, siendo distinta de la operación OR, puesto que la operación de suma de números binarios tiene la misma mecánica que la de números decimales.



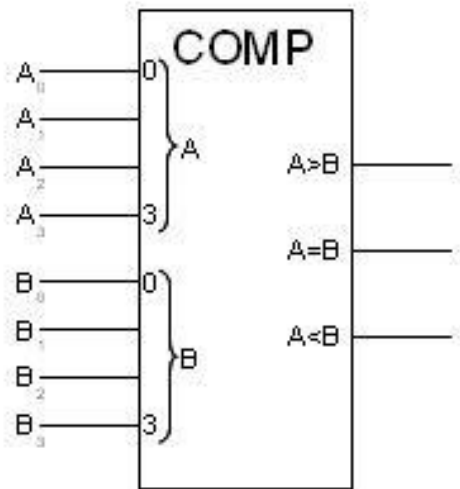
b) Semisumadores:

Son dispositivos capaces de sumar dos bits y dar como resultado la suma de ambos y el acarreo.



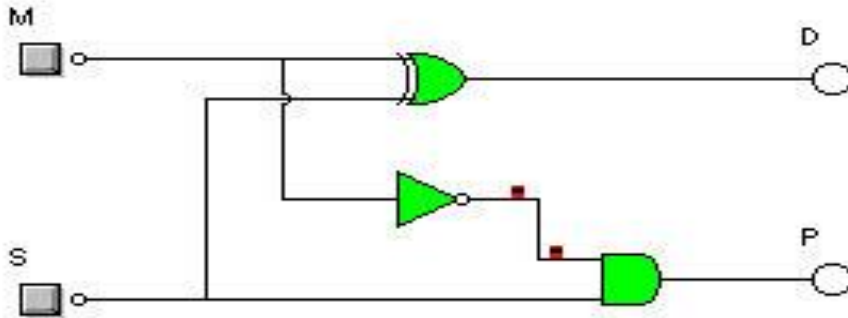
c) Comparadores:

Poseen uno o más pares de entradas, que tienen como función comparar dos magnitudes binarias para determinar su relación, siendo el comparador más básico la compuerta XOR, ya que su salida es 1 si los dos bits de entrada son diferentes y 0 si ambos son iguales.



d) Restadores:

Cada bit del sustraendo se resta de su correspondiente bit del minuendo, para formar el bit de la diferencia. El préstamo ocurre cuando el bit del minuendo es menor al bit del sustraendo, de tal forma que se presta un 1 de la siguiente posición significativa.



CIRCUITOS LSI

Son circuitos de alta escala de integración, y contienen entre 100 y 1000 puertas lógicas, o de 1000 a 10000 transistores. Los ejemplos más importantes de este tipo de circuitos son los microprocesadores y la memoria de semiconductores.

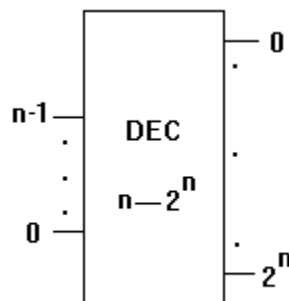


CIRCUITOS MSI Y LSI VIDEO

→ <https://www.youtube.com/watch?v=OaEPmqXesik>

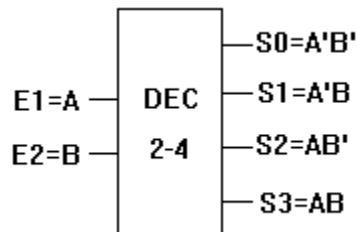
DECODIFICADORES

Un decodificador es un circuito integrado que genera todos los minitérminos correspondientes a n entradas:



En la figura anterior se tiene un decodificador de n entradas y 2^n salidas. Cada salida corresponde a un minitérmino, empezando con la salida superior que corresponde al minitérmino 0.

En la siguiente figura se muestra un decodificador 2-4:



La tabla de verdad correspondiente es:

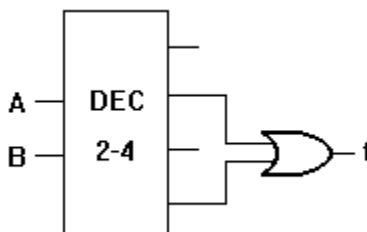
A	B	Y0	Y1	Y2	Y3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

En la tabla de verdad se aprecia que la salida de cada línea es 1 solamente cuando se ha seleccionado el mini término correspondiente al número de la línea, de otra manera la salida es 0.

En la práctica, la mayoría de los decodificadores tienen las salidas complementadas indicándolo mediante una pequeña burbuja o círculo en la salida, es decir, el decodificador genera maxi términos(0) en lugar de mini términos(1).

Una de las aplicaciones de un decodificador es la implementación de funciones Booleanas. Una función Booleana de n variables puede ser implementada fácilmente al unir los minitérminos(maxitérminos) correspondientes a la función utilizando una compuerta OR(NAND).

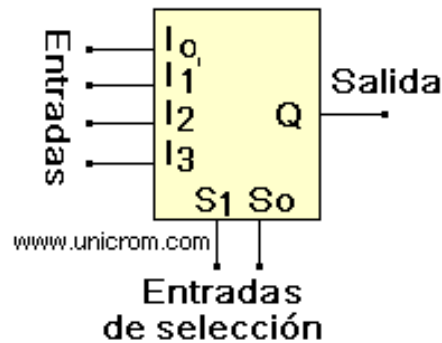
Ejemplo. Implemente $f(A, B) = \sum m(1,3)$ utilizando un decodificador:



DECODIFICADORES VIDEO

➔ <https://www.youtube.com/watch?v=9W8Z070YAYU>

MULTIPLEXORES



El **multiplexor (MUX)** es un [circuito combinacional](#) que tiene varios canales de datos de entrada y solamente un canal de salida. Sólo un canal de la entrada pasará a la salida y este será el que haya sido escogido mediante unas señales de control.

Ejemplo: Si utiliza un **multiplexor** de 4 canales de entrada. Una de los cuatro canales de entrada será escogido para pasar a la salida y ésto se logra con ayuda de las señales de control o selección.

La cantidad de líneas de control que debe de tener el **multiplexor** depende del número de canales de entrada. En este caso, se utiliza la siguiente fórmula: Número de canales de entrada $= 2^n$, donde n es el número de líneas de selección.

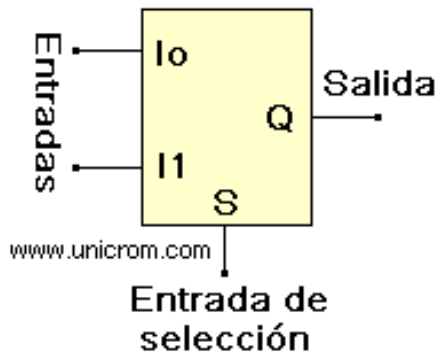
- Para un **multiplexor** de 4 canales de entrada, $n = 2$
- Si la cantidad de canales de entrada fuese 8, las líneas de control serían 3. La fórmula: $8 = 2^n$, $n = 3$
- Si la cantidad de canales de entrada fuese 16, las líneas de control serían 4. La fórmula: $16 = 2^n$, $n = 4$
- Si sólo hay 6 canales de entrada, se utiliza un multiplexor de 3 líneas de control, (don dos líneas de control no es suficiente, pues solo llega hasta 4)
- Si hubiesen 13 canales de entrada se utiliza un **multiplexor** de 4 líneas
- Si hubiesen sólo 2 canales de entrada, sería necesario un multiplexor con una línea de selección.

Normalmente se utilizan **multiplexores** con canales de entrada y salida de . Si se desea lograr canales de dos bits o más, se ponen a trabajar multiplexores en paralelo.

Nota: en alguna literatura se considera a que multiplexores = multiplexadores

Líneas de control para un multiplexador de dos entradas

Dos canales de un solo bit. La única entrada de selección, puede tener $2^n = 2$ posibles valores, donde $n = 1$.



- Con la línea de control en “0” se escoge el primer canal de entrada (canal 0) y lo pasa a la salida.
- Con la línea de control en “1” se escoge el segundo canal de entrada (canal 1) y lo pasa a la salida.

Líneas de control para un multiplexador de 4 entradas

Son necesarias 2 líneas de selección o control para lograr direccionar las 4 entradas. Las combinaciones posibles de las líneas de control o selección son:



00 para la primera entrada (entrada 0) 01 para la segunda entrada (entrada 1) 10 para la tercera entrada (entrada 2) 11 para la cuarta entrada (entrada 3)

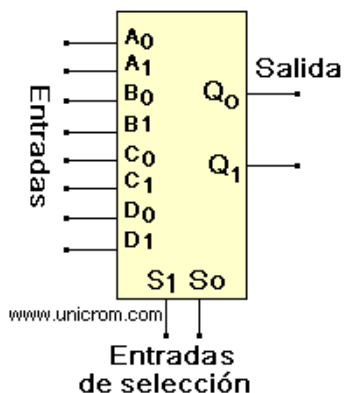
Líneas de control para un multiplexador de 8 entradas

Son necesarias 3 líneas de selección para lograr direccionar las 8 entradas. Las combinaciones serían:

- 000 para la primera entrada (entrada 0)
- 001 para la primera entrada (entrada 1)
- 010 para la primera entrada (entrada 2)
- 011 para la primera entrada (entrada 3)
- 100 para la primera entrada (entrada 4)
- 101 para la primera entrada (entrada 5)
- 110 para la primera entrada (entrada 6)
- 111 para la primera entrada (entrada 7)

Como se puede observar, las líneas de selección, representan el [número en binario](#) que escoge la entrada que pasará a la salida.

Multiplexor de 4 canales de entrada de dos bits



Cada canal de entrada tiene 2 bits (A0 – A1, B0 – B1, C0 – C1 y D0 – D1) y el canal de salida también tienen 2 bits (Q0, Q1). (ver el gráfico de la derecha). Las entradas de selección son siempre 2. (S1 y S0). En este caso cada vez que se selecciona una entrada, se pasar un canal (2 bits) a la salida (también de 2 bits).

Este **multiplexador** se puede implementar con dos multiplexadores de 4 canales de un bit cada uno conectados en paralelo. El multiplexador “0” tienen las entradas X0 y la salida Q0 y el multiplexador “1” tiene las entradas

X1 y la salida Q1. donde X puede ser A, B, C o D. Las entradas de selección S1 y S0 son comunes a ambos **multiplexadores**.

MULTIPLEXORES VIDEO

→ https://www.youtube.com/watch?v=w_w7sOZ58Tg

CONTADORES

Un contador electrónico básicamente consta de una entrada de impulsos que se encarga de conformar (escuadrar), de manera que el conteo de los mismos no sea alterado por señales no deseadas, las cuales pueden falsear el resultado final. Estos impulsos son acumulados en un contador propiamente dicho cuyo resultado, se presenta mediante un visor que puede estar constituido por una serie de sencillos dígitos de siete segmentos o en su caso mediante una sofisticada pantalla de plasma.

Empezaremos por considerar un circuito de entrada que nos permita tomar la señal motivo del conteo, para lo cual hemos de pensar en la forma de tomar la señal a medir. Para que el mencionado circuito sea lo más universal posible:

- Entrada de alta impedancia (Z). El circuito no debería absorber demasiada señal para no inducir errores.
- Dicha señal, la deberemos escuadrar de forma segura.

El circuito constará de un separador de corriente continua mediante un condensador cerámico de baja capacidad (47nf/400V), para detectar las señales de alta frecuencia si es el caso, la salida se conectará a un diferenciador constituido por una puerta lógica, para una mayor seguridad dicha puerta será un disparador Schmitt (trigger Schmitt). Si utilizamos un transistor, éste debe ser de alta velocidad.

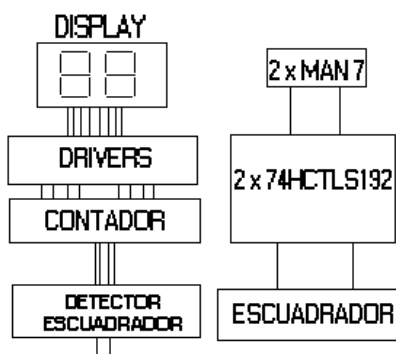


Fig. 1

El circuito de la figura 2, representa la entrada descrita, en la figura 3, más elaborada, se aprecia el circuito de entrada completo que puede servirnos en la mayoría de los casos, en la figura 4, se presenta un nuevo circuito con mejores prestaciones al que se le ha añadido el mencionado disparador Schmitt formado por la puerta de alta velocidad 74HCT14 (6 inversores Schmitt). En el cual resaltamos el punto (A) ya que en él los impulsos detectados aún no están totalmente escuadrados.

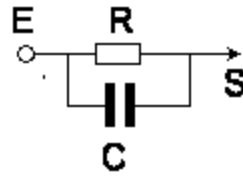


Fig. 2

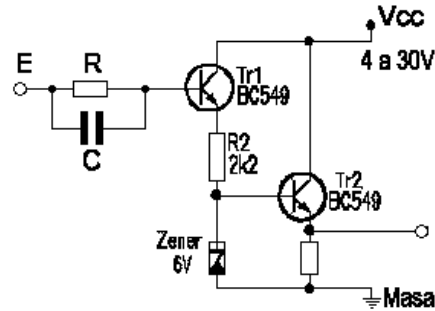


Fig. 3

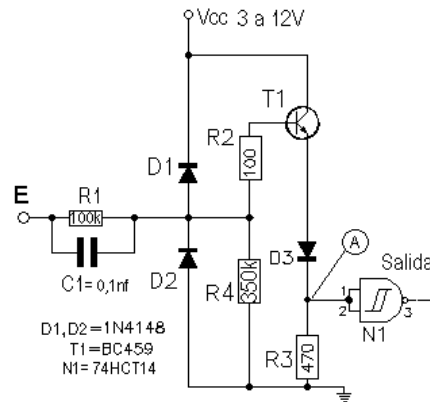


Fig. 4

En la figura 2, se pueden apreciar dos componentes, el condensador separa la tensión continua y la resistencia eleva la impedancia lo suficiente para nuestro cometido. En la figura 3, se propone un circuito que utiliza un transistor como seguidor de emisor con un limitador de tensión a 6V mediante el diodo zener, al que le sigue un nuevo transistor separador que mejora la salida, permitiendo así, una alimentación con un margen bastante amplio.

En la figura 4, los limitadores de tensión son los dos diodos D1 y D2, su punto común como se aprecia es conectado a la base de un transistor mediante un divisor de tensión, y la salida del seguidor de emisor, nos da la señal que utilizaremos para el contador, no obstante, se ha intercalado un inversor o puerta trigger para escuadrar al máximo dicha señal.

Anteriormente vimos cómo detectar y escuadrar los impulsos que posteriormente se han de contar. Ahora, trataremos el que es el corazón del contador propiamente dicho, es decir, la parte del circuito que se encarga de contar, almacenar y acumular cada pulso al siguiente de la cuenta anterior y si es el caso cambiar de decada.

El circuito básico que se use, dependerá de la tecnología disponible, esto lo podemos apreciar mejor en la tabla siguiente:

FAMILIA	TIPO	Relación V/C
TTL	74LS192	+ velocidad + consumo
CMOS	CD4510B	- velocidad - consumo
HCTLS	74HCTLS192	+ velocidad - consumo

Atendiendo a que la velocidad es muy importante y en cualquier momento se puede requerir esta capacidad, nos centraremos en la opción de la serie 74HCTLS192 o en su defecto por la 74LS192 ya que se trata del mismo dispositivo, lo único que cambia es el consumo, la tensión de trabajo y poco más.

El dispositivo 74HCTLS192, constituye un contador asíncrono reversible con entrada paralela, preparado para efectuar el conteo decimal en código binario BCD.

Para cargar las salidas a un determinado estado, se aplican los datos a las entradas Da, Db, Dc y Dd y se aplica el nivel bajo L a la patilla 11 'load', esta operación de carga es independiente del reloj y del estado del contador.

En la figura 1, se muestra la disposición de las patillas como se puede ver.

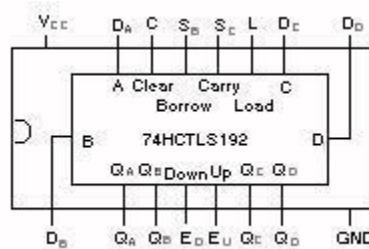


Fig. 1

El circuito se completará añadiendo 3 puertas NAND correspondientes a un 74HCTLS00, una de ellas conectada como inversor y unos pocos elementos como 1 pulsador 4 preseletores, 1 conmutador y unas resistencias de 1k5 y ¼ W. También se puede hacer que el contador se ponga a una determinada cuenta de forma automática

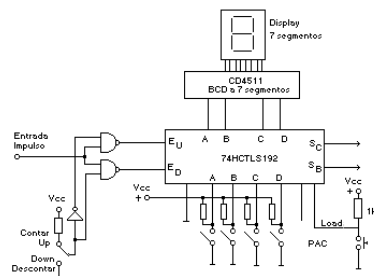


Fig. 2 Esquema del contador para 1 dígito.

Como se desprende del circuito para incrementar el número de dígitos, tan solo habrá que añadir tantos contadores como dígitos se deseen y conectarlos en serie, prescindiendo de las subsiguientes puertas de los nuevos contadores y conectando todas las patillas de carga al mismo pulsador de igual forma que las patillas de PAC.

Por otra parte se encuentran los convertidores de BCD a 7 segmentos, compuesto por un dispositivo de la serie CMOS por ser el más típico al igual que por ser compatible con los HCT.

→ <https://www.youtube.com/watch?v=rLSnUPIpls0>

SUMADORES

Un sumador es un circuito que realiza la suma de dos palabras binarias. Es distinta de la operación OR, con la que no nos debemos confundir. La operación suma de números binarios tiene la misma mecánica que la de números decimales.

Por lo que en la suma de números binarios con dos o más bits, puede ocurrir el mismo caso que podemos encontrar en la suma de números decimales con varias cifras: cuando al sumar los dos primeros dígitos se obtiene una cantidad mayor de 9, se da como resultado el dígito de menor peso y "me llevo" el anterior a la siguiente columna, para sumarlo allí.

En la suma binaria de los dígitos $1 + 1$, el resultado es 0 y me llevo 1, que debo sumar en la columna siguiente y pudiéndose escribir 10, solamente cuando sea la última columna a sumar. A este bit más significativo de la operación de sumar, se le conoce en inglés como carry (acarreo), equivalente al "me llevo una" de la suma decimal.

Semisumador. Es un dispositivo capaz de sumar dos bits y dar como resultado la suma de ambos y el acarreo. La tabla de verdad correspondiente a esta operación sería:

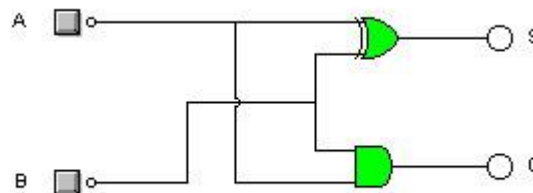
Entradas		Salidas	
A	B	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Con lo que sus funciones canónicas serán:

$$C = AB$$

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

Que una vez implementado con puertas lógicas, un semisumador tendría el circuito:



Sumador completo. Presenta tres entradas, dos correspondientes a los dos bits que se van a sumar y una tercera con el acarreo de la suma anterior. Y tiene dos salidas, el resultado de la suma y el acarreo producido. Su tabla de verdad será:

ENTRADAS			SALIDAS	
A	B	C^{-1}	C	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

Sus funciones canónicas serán:

$$S = \bar{A}\bar{B}C^{-1} + \bar{A}BC^{-1} + A\bar{B}C^{-1} + ABC^{-1}$$

$$C = \bar{A}BC^{-1} + A\bar{B}C^{-1} + ABC^{-1}$$

Que una vez simplificadas quedarían:

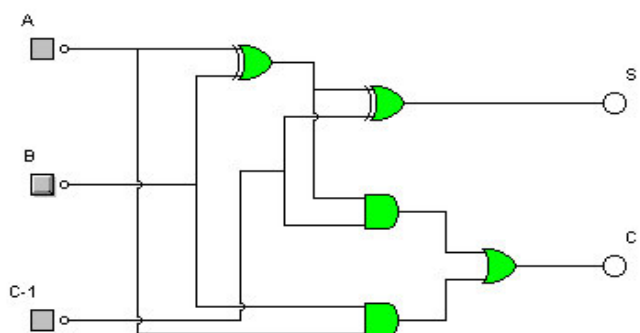
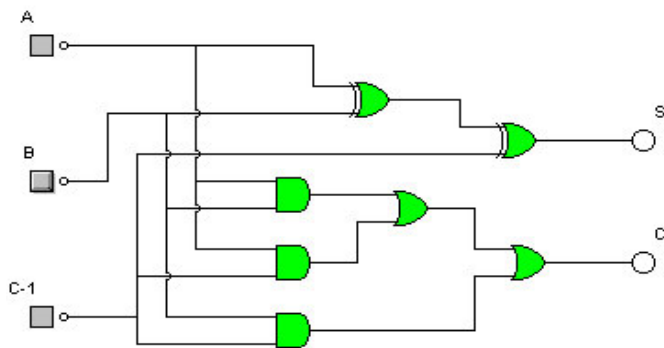
$$S = A \oplus B \oplus C^{-1}$$

$$C = AB + AC^{-1} + BC^{-1}$$

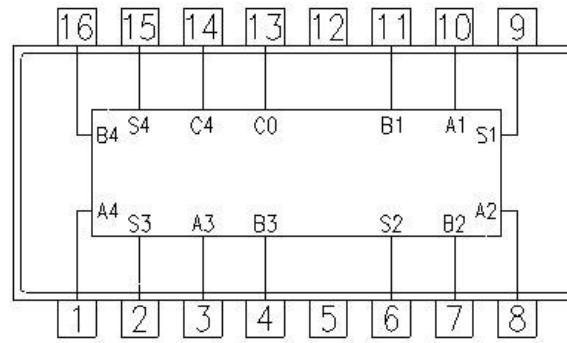
O bien:

$$C = AB + (A \oplus B)C^{-1}$$

Una vez implementado con puertas lógicas el sumador presentaría cualquiera de los siguientes circuitos:

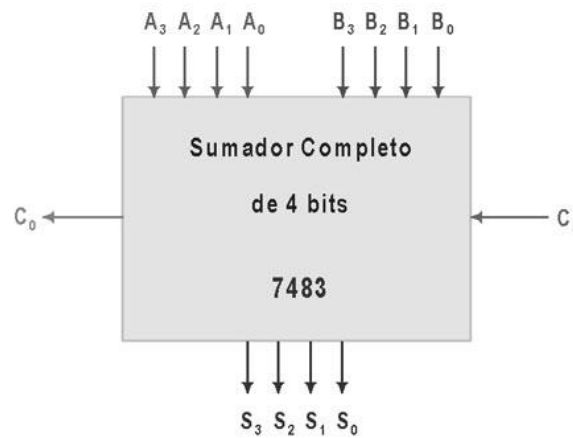


Aunque, como ya hemos dicho en otros casos, en realidad estos circuitos no se cablean con puertas lógicas, si no que forman parte de circuitos integrados como el CI 7483, que es un sumador de cuatro bits.

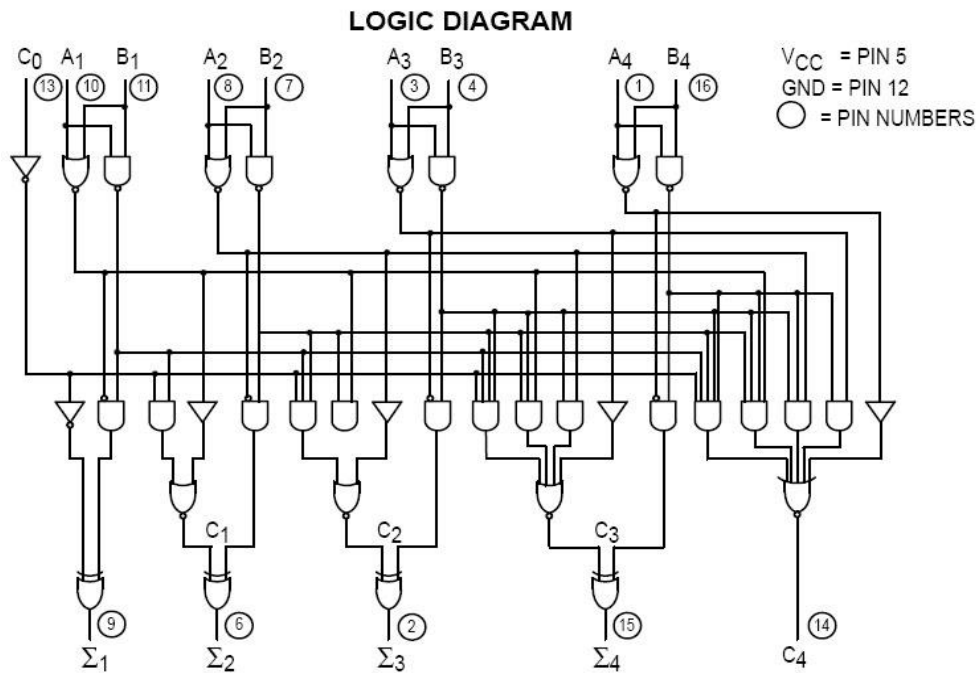


7483

El esquema:



El esquema mostrado en la figura es el conexionado interno que presenta dicho sumador de 4 bits, configurado dentro del CI 7483.



Características del sumador de 4 bits 7483:

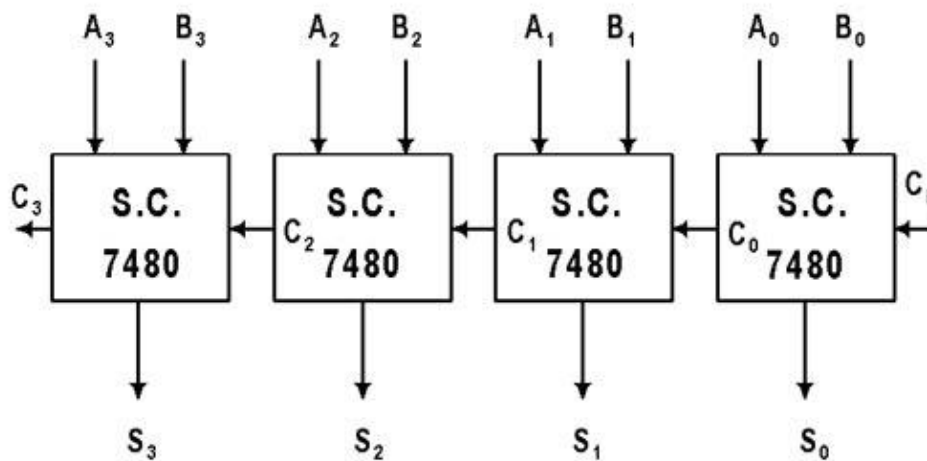
Es un sumador completo que ejecuta la suma de dos números binarios de cuatro bits. Hay salida de suma por cada bit y el acarreo resultante (C4), se obtiene del cuarto bit.

Está diseñado para velocidades medias-altas de funcionamiento, con bits múltiples de suma en paralelo y acarreo en serie.

- Tensión de alimentación.....4,5V a 5,25V.
- Temperatura de funcionamiento.....0 a 70°C.
- Cargabilidad de salida normalizada C4.....5 U.L.
- Cargabilidad de las salidas de suma.....10 U.L.
- Tensión de entrada alta mínima.....2V.
- Tensión de entrada de alta máxima.....0,8V.

Para sumar números de más de un bit, también se recurre al conexionado de sumadores binarios en paralelo, donde el acarreo de la suma de dos dígitos será una entrada a sumar en el paso siguiente.

En este caso se precisan tantos semisumadores como bits tengamos que sumar. El montaje de la figura posterior tiene un funcionamiento idéntico al del CI 7483, aunque presenta incompatibilidades a nivel de pines.



Circuitos Sumadores

➔ https://www.youtube.com/watch?v=gD65hr_iATw

OTROS CIRCUITOS LÓGICOS: ALU

ALU son las siglas de Arithmetic Logic Unit, es decir, Unidad Lógico Aritmética.

Se trata de un circuito integrado con la capacidad de realizar diferentes operaciones aritméticas y lógicas (es decir, del álgebra de Boole), con dos palabras de n bits. Se pueden encontrar como circuitos independientes, y también como bloque funcional dentro de los microprocesadores y microcontroladores.

En general, las operaciones matemáticas están codificadas en binario natural y en complemento a 2 para las restas, pero se pueden codificar en otros códigos, como por ejemplo BCD natural.

El más conocido es 74LS181, que es una ALU de 4 bits, que puede realizar hasta 32 funciones diferentes (16 lógicas y 16 aritméticas), trabaja con números binarios de 4 bits, aunque se pueden conectar en cascada para aumentar el número de bits. Este circuito integrado tiene como entradas:

- Los cuatro bits del operando A.
- Los cuatro bits del operando B.
- Entradas de selección (para seleccionar la operación a realizar, entre 16).
- Entrada de acarreo, por si viene de un integrado con el resultado de menor peso.
- Entrada de control, para seleccionar si la operación a realizar deber ser aritmética o lógica.
- Como salidas tiene los 4 bits del resultado, más una salida comparador ($A = B$) y salidas de acarreo.

Como curiosidad decir que este circuito integrado trabaja con lógica inversa en las entradas de datos y en las salidas, es decir, que para estos pines se invierte el significado de los 1 y los 0. Aunque es posible hacerlo trabajar con lógica directa.

Configuración de pines del CI 74LS181:

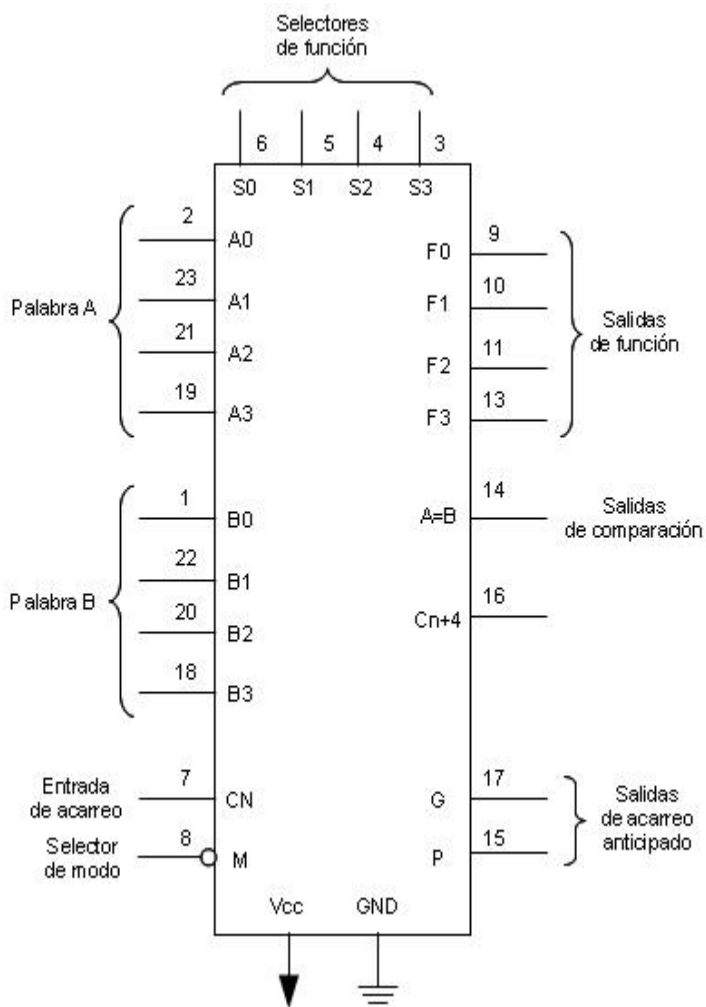


Tabla con las funciones que pueden realizarse con el 74LS181:

Códigos de Selección				Funciones lógicas	Funciones aritméticas*	
					M=0	
S3	S2	S1	S0	M=1	Cn=1(sin acarreo)	Cn=0 (con acarreo)
0	0	0	0	\overline{A}	A	$A+1$
0	0	0	1	$\overline{A+B}$	$A+B$	$(A+B)+1$
0	0	1	0	\overline{AB}	$A+\overline{B}$	$(A+\overline{B})+1$
0	0	1	1	0	-1	0
0	1	0	0	\overline{AB}	$A+A\overline{B}$	$A+A\overline{B}+1$
0	1	0	1	\overline{B}	$(A+B)+A\overline{B}$	$(A+B)+A\overline{B}+1$
0	1	1	0	$A\oplus B$	$A-B-1$	$A-B$
0	1	1	1	$A\overline{B}$	$A\overline{B}-1$	$A\overline{B}$
1	0	0	0	$\overline{A+B}$	$A+AB$	$A+AB+1$
1	0	0	1	$\overline{A\oplus B}$	$A+B$	$A+B+1$
1	0	1	0	B	$(A+\overline{B})+AB$	$(A+\overline{B})+AB+1$
1	0	1	1	AB	$AB-1$	AB
1	1	0	0	1	$A+A$	$A+A+1$
1	1	0	1	$A+\overline{B}$	$(A+B)+A$	$(A+B)+A+1$
1	1	1	0	$A+B$	$(A+\overline{B})+A$	$(A+\overline{B})+A+1$
1	1	1	1	A	$A-1$	A

*: Expresadas en complemento a 2

+: Operador OR en funciones lógicas y signo más en operaciones aritméticas

-: Signo aritmético menos

$\overline{}$: Barra de inversión lógica

ALU VIDEO:

→ <https://www.youtube.com/watch?v=ukjiTrlVvtQ>

Certificate of Completion

This certificate is awarded to

Jaqueline

for successful completion of
The Hour of Code
and demonstrating an understanding of
the basic concepts of Computer Science.



www.code.org

Hadi Partovi

Hadi Partovi, Co-founder and Chief Executive Officer, Code.org

To learn beyond your first hour, visit Code.org
Microsoft made the generous gift to sponsor your learning.

