ELECTRÓNICA DIGITAL

En el tema anterior vimos los componentes y circuitos de electrónica analógica (resistencias, condensadores, diodos, transistores, etc), ahora estudiamos los componentes y circuitos de electrónica digital.

El término **ANALÓGICO** se aplica a cualquier información o señal cuyos valores son ***continuos***, es decir que puede tomar cualquier valor dentro de un cierto límite, superior e inferior, como la mayoría de las que representan una magnitud física (temperatura, presión, luminosidad, humedad, etc.).

Las señales analógicas pueden tomar todos los valores posibles de un intervalo.

El término **DIGITAL** se aplica a cualquier información o señal cuyos valores son ***discretos***, es decir que sólo puede tomar valores fijos, dos valores posibles, como si la señal digital fuera variando "a saltos" entre un valor máximo y un valor mínimo.

En los circuitos digitales todas las señales son digitales. Para su análisis y diseño se aplica la Teoría del Algebra de Boole. Fiabilidad, reducido tamaño, barato. Como inconvenientes la complejidad de los circuitos

Algebra de Boole

George Boole desarrolló el álgebra que lleva su nombre, base de la actual electrónica digital.

El álgebra de Boole opera con **variables** que únicamente pueden tomar dos valores, que se designan por cero y uno (0 y 1), y que en los circuitos electrónicos digitales representan si hay o no voltaje, en **lógica positiva**, aunque podría ser al revés (1 si no hay voltaje), si se trabaja con **lógica negativa**.

<u>Variable lógica</u>: símbolo (letra, p.ej: **a**) que <u>representa una proposición sencilla o hecho físico</u> que adoptará valor "1" si es cierto (p.ej.: pulsar botón verde) o "0" si es falso.

<u>Función lógica</u>: es una expresión algebraica formada por una serie de variables binarias relacionadas entre sí por determinadas operaciones. Es decir, representa el resultado de una combinación de proposiciones o variables lógicas: **S = f(a,b…).** Los valores de una función lógica son binarios.

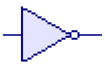
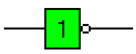
Tabla de verdad: cuadro que recoge todas las combinaciones de las variables de las que depende una función lógica: 2 combinaciones si son dos variables, 8 si son 3 variables,... **2^n combinaciones si son n variables.**

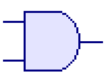
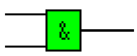
a	b	S
0	0	1
0	1	0
1	0	0
1	1	1

Operaciones básicas: corresponden a la suma, producto y negación, realizadas con variables lógicas. Estas operaciones básicas del álgebra de Boole se implementan mediante **puertas lógicas**: circuitos electrónicos que realizan una operación lógica sencilla con los valores de una o más señales lógicas de entrada para dar automáticamente una señal lógica de salida.

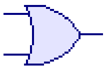
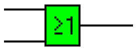
Puertas Lógicas

Las puertas lógicas fundamentales son tres:

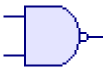
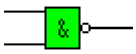
Puerta	Símbolo	Símbolo Normalizado	Tabla Verdad	Función
NOT	a  S	a  S	a | S 0 | 1 1 | 0	S = $\bar{a}$ Realiza la inversión lógica de una proposición

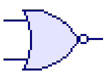
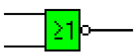
Puerta	Símbolo	Símbolo Normalizado	Tabla Verdad	Función
AND			a b | S 0 0 | 0 0 1 | 0 1 0 | 0 1 1 | 1	S= a·b = ab Realiza cumplimiento simultáneo de las proposiciones: su resultado es "cierto" si ambas son ciertas a la vez

© ALBERTO CASTRO

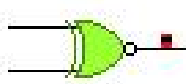
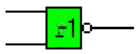
Puerta	Símbolo	Símbolo Normalizado	Tabla Verdad	Función
OR			a b S 0 0 0 0 1 1 1 0 1 1 1 1	S= a+b su resultado es cierto si lo es al menos alguna de las proposiciones

Combinando una puerta AND y una NOT o una OR y una NOT, tenemos un nuevo tipo de puerta.

Puerta	Símbolo	Símbolo Normalizado	Tabla Verdad	Función
NAND			a b S 0 0 1 0 1 1 1 0 1 1 1 0	$S = \overline{a \cdot b} = \overline{ab}$ su resultado es cierto siempre que no se de la simultaneidad en las proposiciones

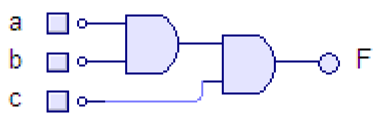
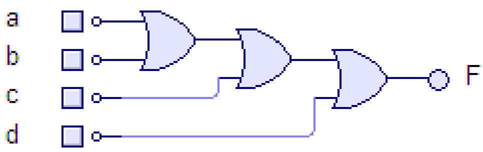
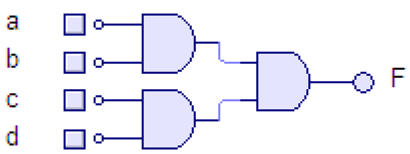
Puerta	Símbolo	Símbolo Normalizado	Tabla Verdad	Función
NOR			a b S 0 0 1 0 1 0 1 0 0 1 1 0	$S = \overline{a + b}$ su resultado es cierto siempre que no lo es ninguna de las proposiciones

© ALBERTO CASTRO

Puerta	Símbolo	Símbolo Normalizado	Tabla Verdad	Función
XOR			a b | S 0 0 | 0 0 1 | 1 1 0 | 1 1 1 | 0	$S = a \oplus b$ su resultado es cierto siempre que las proposiciones son de valor diferente

Al combinar varias puertas, formando un circuito, el resultado es una función final que es combinación de las distintas operaciones que se van efectuando.

Si queremos multiplicar o sumar más de dos entradas podemos utilizar, si disponemos, de puertas multientrada (de 3 o más entradas), y si no disponemos de éstas puertas multientrada, podemos combinar varias de dos entradas:

Puerta	Combinación de puertas 2 entradas	Función
AND 3 entradas		$F = abc$
OR 4 Entradas		$F = a + b + c + d$
AND 4 entradas		$F = abcd$

© ALBERTO CASTRO

Leyes de Morgan

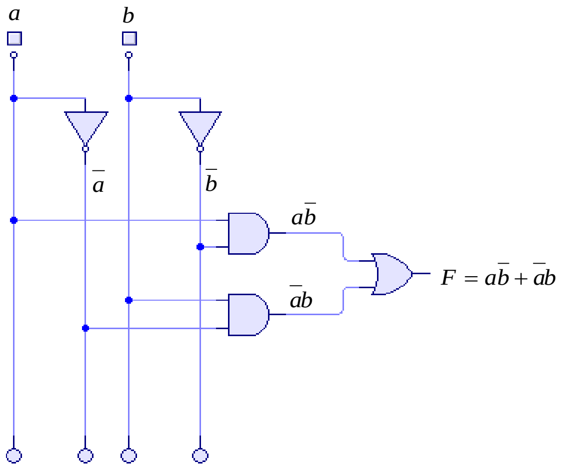
$$\overline{(a+b+c+\ldots)} = \bar{a}\cdot\bar{b}\cdot\bar{c}\ldots$$

$$\overline{(a\cdot b\cdot c\ldots)} = \bar{a}+\bar{b}+\bar{c}+\ldots$$

Ejercicios de ANÁLISIS

Cuando tenemos un circuito dado y se nos pide averiguar su función estamos ante un ejercicio de ANÁLISIS.

Vamos a ver un ejemplo



Para resolver un ejercicio de ANÁLISIS, iremos colocando a la salida de las diferentes puertas, el resultado de "la operación" (cómo se ha indicado en el circuito anterior). A la salida de la última puerta el resultado es el de la función que está implementada en el circuito.

En ocasiones es útil obtener la tabla de verdad del circuito, para lo cual podemos utilizar la fórmula obtenida.

En este caso tenemos dos entradas, por lo que tenemos 2^2=4 estados.

a	b	F	$F = a\bar{b} + \bar{a}b$
0	0	0	$F = 0\cdot\bar{0} + \bar{0}\cdot 0 = 0\cdot 1 + 1\cdot 0 = 0 + 0 = 0$
0	1	1	$F = 0\cdot\bar{1} + \bar{0}\cdot 1 = 0\cdot 0 + 1\cdot 1 = 0 + 1 = 1$
1	0	1	$F = 1\cdot\bar{0} + \bar{1}\cdot 0 = 1\cdot 1 + 0\cdot 0 = 1 + 0 = 1$
1	1	0	$F = 1\cdot\bar{1} + \bar{1}\cdot 1 = 1\cdot 0 + 0\cdot 1 = 0 + 0 = 0$

© ALBERTO CASTRO

Ejercicios de SÍNTESIS

Cuando queremos automatizar o controlar un sistema y queremos diseñar para ese fin un circuito, se trata de un ejercicio de SÍNTESIS. Vamos a ver cómo se hace este proceso de diseñar un circuito digital.

Vamos a ver un ejemplo

Supongamos que tenemos un coche de dos plazas, que queremos que nos indique si tenemos puesto el cinturón de seguridad.

Supongamos que hay un sensor de peso (a) que indica si hay o no conductor, otro sensor (b) que dice si el cinturón del conductor está abrochado y lo mismo para el pasajero, es decir, sensor de peso para saber si hay pasajero (c) y sensor de cinturón abrochado de pasajero (d).

Cómo tenemos cuatro interruptores/sensores (a), (b), (c) y (d), el número de casos, de estados, es $2^4=16$

La salida del sistema, que llamaremos S, será 1 en el caso de que el sistema detecte una situación incorrecta y 0 en el caso de que sea correcta.

Analicemos cada caso:

a	b	c	d	S	
0	0	0	0	0	No está el conductor ni el acompañante y ninguno se ha puesto el cinturón: S=0
0	0	0	1	0	Está abrochado el cinturón sin estar el acompañante. CASO ABSURDO → Elegimos S=0
0	0	1	0	1	Está el acompañante sin cinturón. Debe sonar la alarma en el coche → S=1
0	0	1	1	0	Acompañante con el cinturón puesto → S=0
0	1	0	0	0	Está abrochado el cinturón sin estar el conductor. CASO ABSURDO → Elegimos S=0
0	1	0	1	0	Abrochados los cinturones sin ocupantes. CASO ABSURDO → Elegimos S=0
0	1	1	0	1	Está el acompañante sin cinturón. Debe sonar la alarma en el coche → S=1
0	1	1	1	0	Acompañante con cinturón abrochado y CASO ABSURDO del conductor→ Elegimos S=0
1	0	0	0	1	Conductor sin cinturón → S=1
1	0	0	1	1	Conductor sin cinturón → S=1
1	0	1	0	1	Conductor sin cinturón → S=1
1	0	1	1	1	Conductor sin cinturón → S=1
1	1	0	0	0	Conductor con cinturón y sin pasajero → S=0
1	1	0	1	0	Conductor con cinturón y CASO ABSURDO del pasajero → S=0
1	1	1	0	1	Conductor con cinturón pero pasajero sin él → S=1
1	1	1	1	0	Conductor y pasajero con cinturón → S=0

© ALBERTO CASTRO

Para poder construir el circuito electrónico, previamente debemos obtener la función lógica a implementar.

Esto lo hacemos como sumatorio de TÉRMINOS MÁXIMOS (MAXTERMS) o producto de TÉRMIMOS MÍNIMOS (MINTERMS)

Los MAXTERMS se construyen con los 1's de la tabla verdad del sistema, y los MINTERMS con los 0's

Para construir los MAXTERMS multiplicamos las cuatro entradas *abdc*, y negamos en cada caso, aquellas que corresponden con cero.

En la tabla siguiente obtenemos los MAXTERMS para todos los estados del ejercicio.

	a	b	c	d	MAXTERMS
0	0	0	0	0	$\overline{a}\overline{b}\overline{c}\overline{d}$
1	0	0	0	1	$\overline{a}\overline{b}\overline{c}d$
2	0	0	1	0	$\overline{a}\overline{b}c\overline{d}$
3	0	0	1	1	$\overline{a}\overline{b}cd$
4	0	1	0	0	$\overline{a}b\overline{c}\overline{d}$
5	0	1	0	1	$\overline{a}b\overline{c}d$
6	0	1	1	0	$\overline{a}bc\overline{d}$
7	0	1	1	1	$\overline{a}bcd$
8	1	0	0	0	$a\overline{b}\overline{c}\overline{d}$
9	1	0	0	1	$a\overline{b}\overline{c}d$
10	1	0	1	0	$a\overline{b}c\overline{d}$
11	1	0	1	1	$a\overline{b}cd$
12	1	1	0	0	$\overline{a}\overline{b}\overline{c}\overline{d}$
13	1	1	0	1	$ab\overline{c}d$
14	1	1	1	0	$abc\overline{d}$
15	1	1	1	1	$abcd$

Para sacar la función debemos sumar los términos máximos para los que la salida es 1.

© ALBERTO CASTRO

	a	b	c	d	S	MAXTERMS
0	0	0	0	0	0	
1	0	0	0	1	0	
2	0	0	1	0	1	$\bar{a}\bar{b}c\bar{d}$
3	0	0	1	1	0	
4	0	1	0	0	0	
5	0	1	0	1	0	
6	0	1	1	0	1	$\bar{a}bc\bar{d}$
7	0	1	1	1	0	
8	1	0	0	0	1	$a\bar{b}\bar{c}\bar{d}$
9	1	0	0	1	1	$a\bar{b}\bar{c}d$
10	1	0	1	0	1	$a\bar{b}c\bar{d}$
11	1	0	1	1	1	$a\bar{b}cd$
12	1	1	0	0	0	
13	1	1	0	1	0	
14	1	1	1	0	1	$abc\bar{d}$
15	1	1	1	1	0	

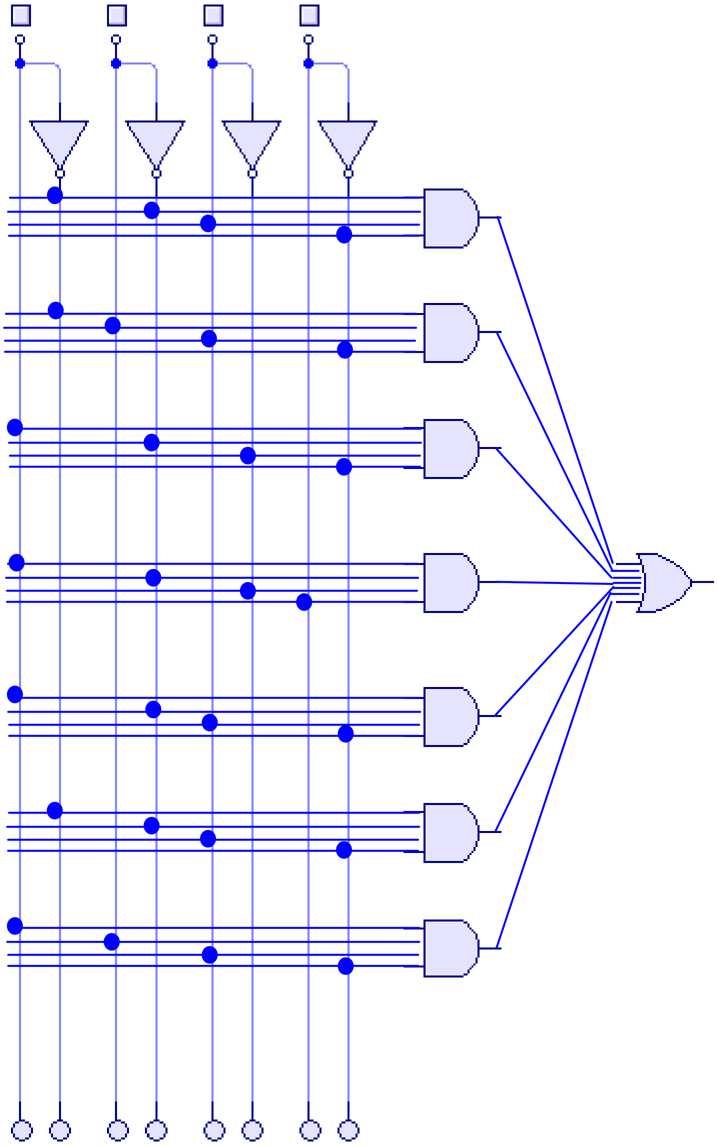
Una forma de expresarlo es:

$$S = \sum(2,6,8,9,10,11,14)$$

o lo que es lo mismo::

$$S = \bar{a}\bar{b}c\bar{d} + \bar{a}bc\bar{d} + a\bar{b}\bar{c}\bar{d} + a\bar{b}\bar{c}d + a\bar{b}c\bar{d} + a\bar{b}cd + abc\bar{d}$$

Sólo queda dibujar el circuito:



Simplificación

Hasta ahora nos hemos puesto a diseñar “sin limitaciones” y para un sistema bien sencillo, nos ha salido un circuito razonablemente complejo. Pero, ¿existe alguna forma de hacer lo mismo de forma más sencilla?

Primero, podemos intentar simplificar, intentando conseguir el mismo efecto, pero con menos casos, es decir prescindiendo de un sensor.

Pensemos un poco, ¿es necesario el sensor que detecta la presencia del conductor? Veamos, si el coche va en marcha “sin conductor”, el problema del cinturón casi el menor de nuestros problemas, ¿no? Con lo cual es posible quitar este interruptor y que el vehículo sólo “pite” si el sensor de cinturón del conductor detecta que no está abrochado.

Así, podemos establecer ahora (a) como el sensor de cinturón abrochado del conductor, (b) como sensor de presencia de pasajero y (c) es un sensor de cinturón abrochado del pasajero.

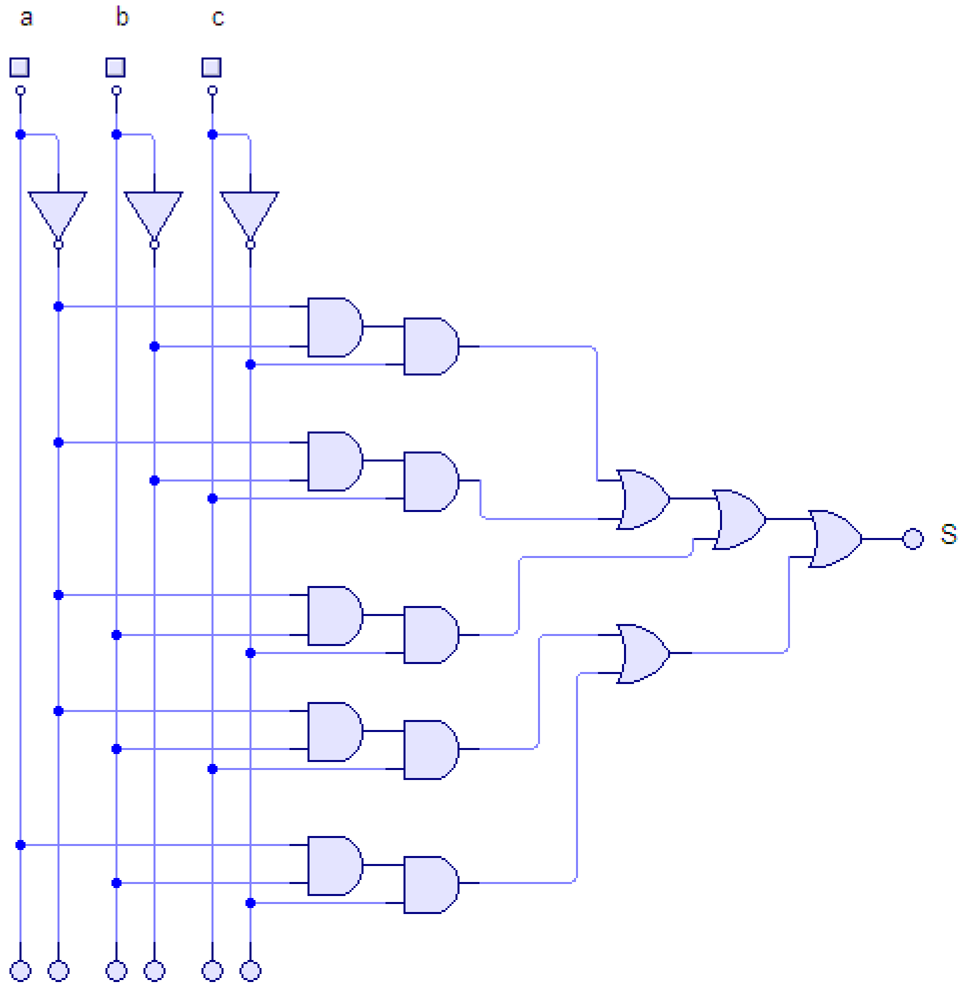
Redefinido el problema, la tabla de verdad del ejercicio, quedará así:

	a	b	c	S	
0	0	0	0	1	El conductor no lleva cinturón→ S = 1
1	0	0	1	1	El conductor no lleva cinturón→ S = 1
2	0	1	0	1	El conductor no lleva cinturón→ S = 1
3	0	1	1	1	El conductor no lleva cinturón→ S = 1
4	1	0	0	0	El conductor lleva cinturón y no hay pasajero→ S = 0
5	1	0	1	0	El conductor lleva cinturón y no hay pasajero que si lleva el cinturón CASO ABSURDO→ S = 0
6	1	1	0	1	El conductor lleva cinturón y hay pasajero que no lleva cinturón→ S = 1
7	1	1	1	0	El conductor lleva cinturón y hay pasajero que lleva cinturón→ S = 0

La función sería:

$$S = \sum(0,1,2,3,6) = \bar{a}\bar{b}\bar{c} + \bar{a}\bar{b}c + \bar{a}b\bar{c} + \bar{a}bc + ab\bar{c}$$

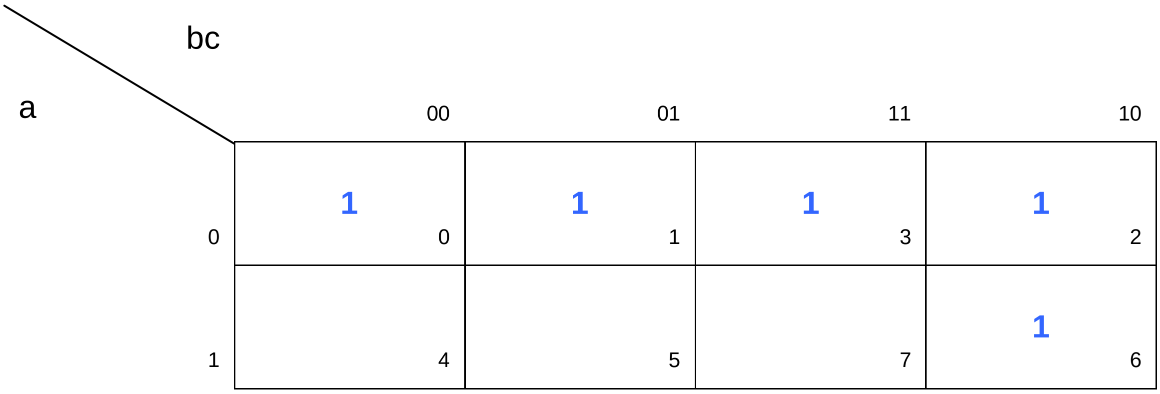
El circuito derivado de esta función sería:



© ALBERTO CASTRO

Además de la simplificación obtenida digamos “racionalizando las entradas”, que no siempre será posible y que es “cualitativa”, podemos hacer una sistemática, que en general nos simplificará la función a través de un procedimiento llamado Mapa de Karnaugh

Para ello, como en la tabla de verdad anterior tenemos 8 estados, vamos a dibujar una cuadrícula con 8 huecos distribuidos en 2 filas y 4 columnas.



Las cuatro celdas horizontales las vamos a nombrar con el siguiente código (llamado GRAY): 00, 01, 11, 10

Las dos celdas verticales las nombramos con 0 y 1.

Las celdas horizontales contienen los valores que toma (a), mientras que los valores horizontales contienen los valores que toman (b) (c).

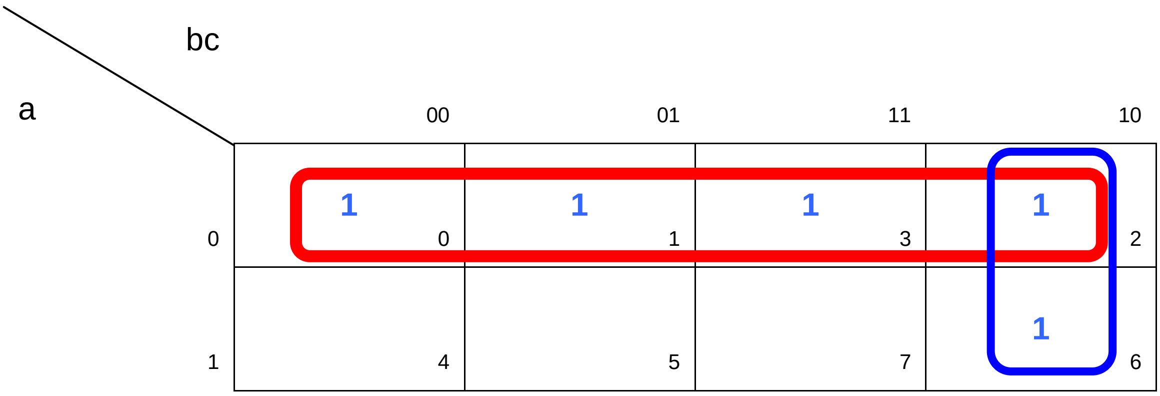
De esta forma podemos nombrar las celdas del 0 al 7 (nº pequeño en la esquina inferior derecha) que se obtiene de la siguiente forma:

Ejemplo: Celda 5 las “coordenadas” bc son 01, y la “coordenada” a es 1, entonces abc=101, que corresponde al número 5 en la tabla de verdad.

Para hacer el proceso de simplificación, colocamos un 1 (los marcados en azul) en cada una de las celdas en las que la tabla de verdad indica que el valor de la salida es 1.

Una vez colocados, debemos hacer agrupaciones de 1’s, comenzando por grupos de 4, continuando con grupos de 2 y concluyendo con grupos de 1.

Estos grupos, pueden ser tomados en horizontal o vertical pero no en diagonal. Así nosotros vamos a tomar dos grupos: uno de 4 y uno de 2



© ALBERTO CASTRO

Analizamos el grupo de 4, señalado en rojo. Los términos que incluye son:

$(0) \rightarrow \overline{a}\overline{b}\overline{c}$

$(1) \rightarrow \overline{a}\overline{b}c$

$(3) \rightarrow \overline{a}bc$

$(2) \rightarrow \overline{a}b\overline{c}$

Vemos que todos los términos tienen en común $\overline{a}$, y que los demás no coinciden, por lo que agrupar este bloque nos da como simplificación: $\overline{a}$

Analizamos ahora el grupo de 2, señalado en azul. Como vemos comparte con el grupo anterior el (2). Podríamos haber cogido un solo término el (6) pero, siempre que se pueda es preferible tomar grupos de 2, y dejar los grupos de 1, para cuando no se pueden combinar de ninguna forma. Los términos de este grupo son:

$(2) \rightarrow \overline{a}b\overline{c}$

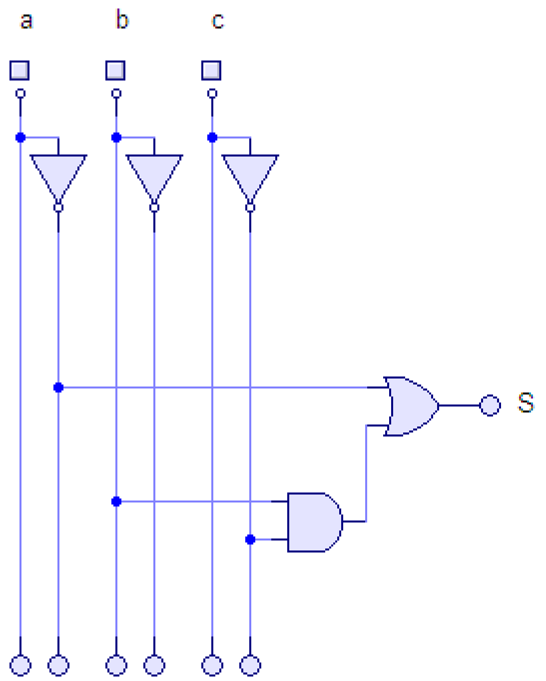
$(6) \rightarrow ab\overline{c}$

Los dos términos tienen en común b y $\overline{c}$, por los que el término que obtenemos de esta simplificación es $b\overline{c}$

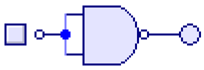
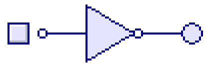
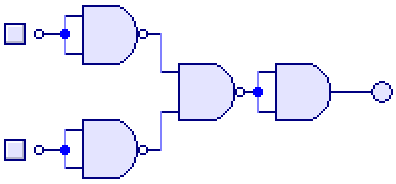
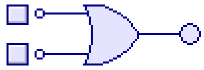
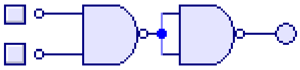
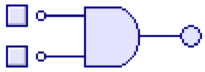
Para concluir el proceso de simplificación por Karnaugh, sólo queda sumar los dos términos provenientes de la simplificación, con lo que la función queda:

$$S = \overline{a} + b\overline{c}$$

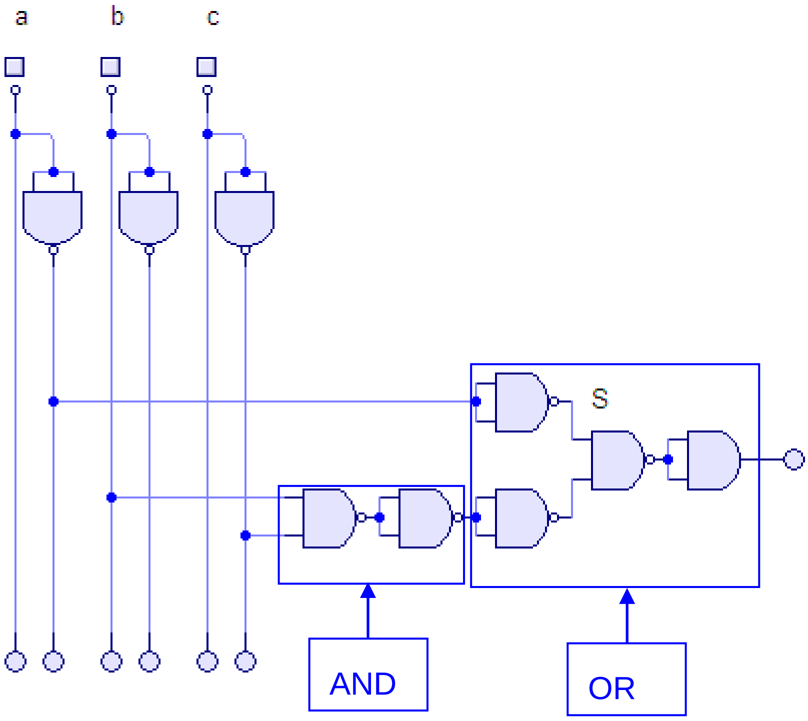
Una vez obtenida la función, sólo resta dibujar el circuito:



Una vez simplificado, puede interesarnos reducir el número de tipos de puerta. Con una puerta NAND, se pueden hacer equivalencias de los tres tipos de puertas, veámoslas:

NOT		
OR		
AND		

Utilizando estas equivalencias, el circuito anterior quedaría de esta forma:

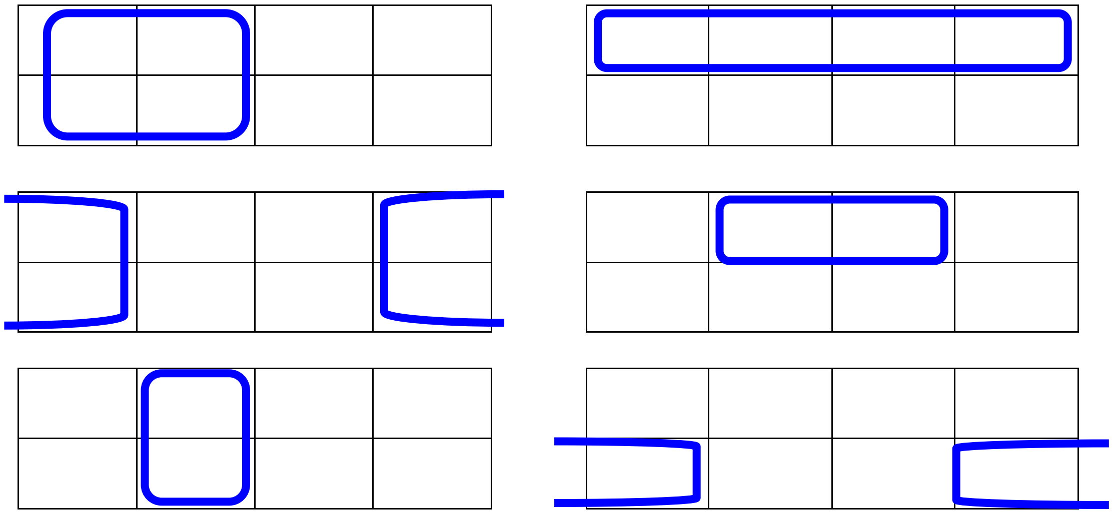


Tenemos el ejercicio simplificado y con un sólo tipo de puerta lógica.

© ALBERTO CASTRO

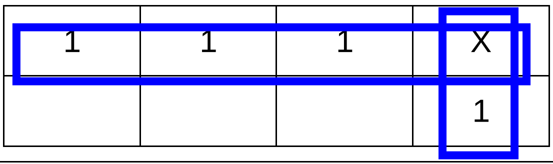
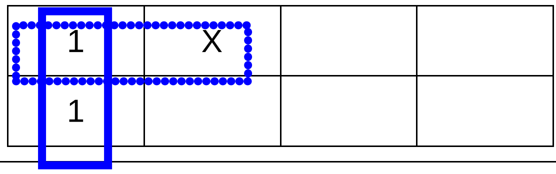
Algunos Consejos para hacer agrupaciones

A la hora de agrupar, hemos dicho que podemos hacerlo en horizontal y vertical y NUNCA en diagonal. Veamos qué agrupaciones están permitidas:



Casos Absurdos

En los ejercicios de síntesis, al estudiar cada estado, nos han salido algunos casos que hemos denominado Absurdos (esencialmente aquellos, en los que no había pasajero pero si tenía puesto el cinturón). Nosotros hemos optado por decidir un valor, 0 ó 1, pero lo correcto es marcarlos con una X. De esta forma, al mapa de Karnaugh, trasladamos los 1´s y las X's, y luego elegimos el valor de la X según favorezca o no la realización de grupos de simplificación.

	
Es mejor elegir X con valor 1, para conseguir dos grupos de simplificación	Es mejor elegir X=0, ya que así con un solo grupo de simplificación (el punteado no sería necesario) ya tenemos la función.

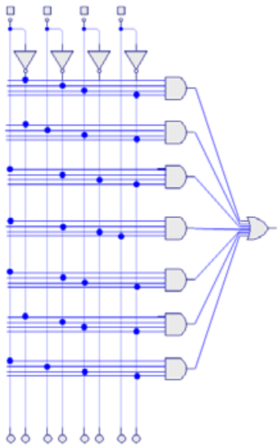
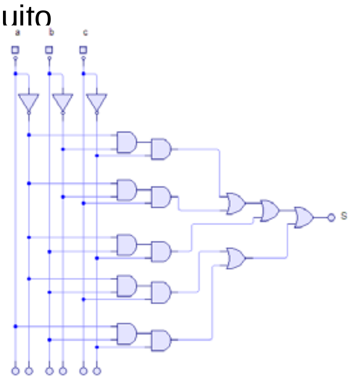
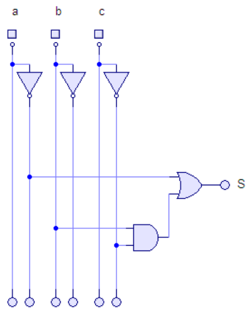
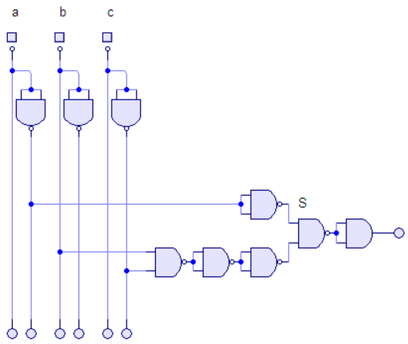
Resumen

Hemos resuelto el ejercicio de realizar un sistema que avise si llevamos incorrectamente puestos los cinturones de seguridad.

Para ello hemos hecho sucesivas aproximaciones.

1. Con 4 sensores y sin simplificar
2. Simplificando sensores, con 3 sensores y sin simplificar por Karnaugh
3. Con 3 sensores y simplificado por Karnaugh

FASES DE UN EJERCICIO DE SÍNTESÍS

<table>
<tr><td>FASE 1
Sin simplificar</td><td>Función
$$S = \bar{a}\bar{b}c\bar{d} + \bar{a}bc\bar{d} + a\bar{b}\bar{c}\bar{d} + \bar{a}\bar{b}cd + a\bar{b}c\bar{d} + a\bar{b}cd + abc\bar{d}$$</td><td>Circuito
</td></tr>
<tr><td>FASE 2
Reduciendo numero de sensores</td><td>Función
$$S = \bar{a}\bar{b}\bar{c} + \bar{a}\bar{b}c + \bar{a}b\bar{c} + \bar{a}bc + ab\bar{c}$$</td><td>Circuito

</td></tr>
<tr><td>FASE 3
Simplificando por Karnaugh</td><td>Función
$$S = \bar{a} + b\bar{c}$$</td><td>Circuito

</td></tr>
<tr><td>FASE 4
Usando sólo puertas NAND</td><td>Función
La misma que en la Fase anterior pero implementada con puertas NAND.</td><td>Circuito

</td></tr>
</table>