Tema 2. Funciones Lógicas

- Algebra de Conmutación.
- Representación de circuitos digitales.
- Minimización de funciones lógicas.

Álgebra de conmutación

- Algebra de Conmutación: Postulados y Teoremas.
- Representación de problemas lógicos. Definición de funciones lógicas. Puertas lógicas y circuitos lógicos. Simplificación de funciones lógicas y de circuitos lógicos.
- Representación de problemas lógicos. Tabla de verdad. Paso de una tabla de verdad a una función lógica: formas canónicas. Funciones lógicas incompletamente especificadas.

Álgebra de Commutación

- Álgebra de Boole. Definición y Postulados.
- Álgebra de conmutación. Operadores lógicos.
- Teoremas del álgebra de Boole.

Álgebra de Boole

- Sistema algebraico formado por un conjunto $B = \{0, a, b, \ldots, 1\}$ finito, y dos operaciones [+, •], que cumplen los siguientes postulados para cualesquiera elementos $X, Y, Z \in B$,

P1. $X + Y \in B$; $X \cdot Y \in B$

P2. Propiedades conmutativa => $X + Y = Y + X$; $X \cdot Y = Y \cdot X$

P3. Propiedades distributivas =>

$$X \cdot (Y + Z) = (X \cdot Y) + (X \cdot Z)$$

$$X + (Y \cdot Z) = (X + Y) \cdot (X + Z)$$

P4. Elemento identidad => $X + 0 = X$; $X \cdot 1 = X$

P5. Elemento complementado => Para X existe $\overline{X} \in B$, tal que $X \cdot \overline{X} = 0$ y $X + \overline{X} = 1$.

Álgebra de Commutación

- Sistema algebraico formado por un conjunto $B = \{0, 1\}$, con las siguientes operaciones [+, •]

X Y	X + Y
0 0	0
0 1	1
1 0	1
1 1	1

X Y	X • Y
0 0	0
0 1	0
1 0	0
1 1	1

X	$\overline{X}$
0	1
1	0

- El álgebra de conmutación cumple los postulados del álgebra de Boole:

P1: Los resultados de $X + Y$ y $X \bullet Y$ son 0 o 1

P2: $0 + 1 = 1 + 0 = 1$; $0 \bullet 1 = 1 \bullet 0 = 0$

P4: $0 + \mathbf{0} = 0$, $1 + \mathbf{0} = 1$; $0 \bullet \mathbf{1} = 0$, $1 \bullet \mathbf{1} = 1$

P5: $0 + \{1\} = 1$, $0 \bullet \{0, 1\} = 0 \Rightarrow \overline{0} = 1$;

$1 + \{0, 1\} = 1$, $1 \bullet \{0\} = 0 \Rightarrow \overline{1} = 0$

La propiedad distributiva se comprueba por *perfecta inducción* o prueba de todas las posibilidades en tres variables booleanas X, Y y Z.

X Y Z	Y + Z	X (Y + Z)	X Y	X Z	XY + XZ
0 0 0	0	0	0	0	0
0 0 1	1	0	0	0	0
0 1 0	1	0	0	0	0
0 1 1	1	0	0	0	0
1 0 0	0	0	0	0	0
1 0 1	1	1	0	1	1
1 1 0	1	1	1	0	1
1 1 1	1	1	1	1	1

X Y Z	Y Z	X + Y Z	X + Y	X + Z	(X + Y) (X + Z)
0 0 0	0	0	0	0	0
0 0 1	0	0	0	1	0
0 1 0	0	0	1	0	0
0 1 1	1	1	1	1	1
1 0 0	0	1	1	1	1
1 0 1	0	1	1	1	1
1 1 0	0	1	1	1	1
1 1 1	1	1	1	1	1

- El álgebra de conmutación guarda correspondencia con la lógica de proposiciones donde se estudian razonamientos en función de los valores verdadero (V) y falso (F) y las operaciones entre ellos AND (Y lógico), OR (O lógico) y NOT (No lógico) mediante la siguiente transformación:

  $0 \Leftrightarrow F,\ 1 \Leftrightarrow V,\ + \Leftrightarrow OR,\ \bullet \Leftrightarrow AND,\ \overline{} \Leftrightarrow NOT$

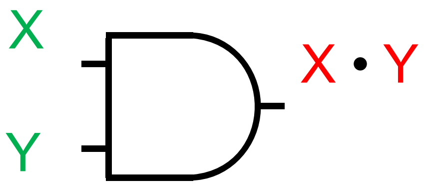
- También guarda relación con la teoría de conjuntos usando la siguiente transformación:

  $0 \Leftrightarrow \{\phi\},\ 1 \Leftrightarrow \{U\},\ + \Leftrightarrow \cup,\ \bullet \Leftrightarrow \cap,\ \{\overline{A}\} \Leftrightarrow \{U\} - \{A\}$

- Por tanto, especificaciones lógicas o basadas en conjuntos pueden resolverse mediante el álgebra de conmutación.

Funciones lógicas básicas

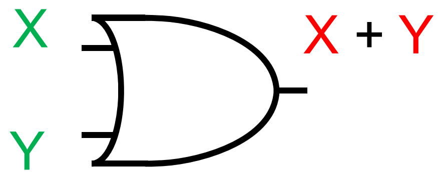
- Operación AND



X Y	X AND Y
F F	F
F V	F
V F	F
V V	V

X Y	X • Y
0 0	0
0 1	0
1 0	0
1 1	1

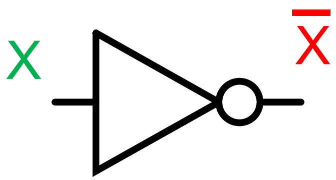
- Operación OR



X Y	X OR Y
F F	F
F V	V
V F	V
V V	V

X Y	X + Y
0 0	0
0 1	1
1 0	1
1 1	1

- Operación NOT



X	NOT X
F	V
V	F

X	$\overline{X}$
0	1
1	0

Teoremas del Álgebra de Boole

- Estos teoremas se demuestran a partir de los postulados del álgebra de boole y se aplican a cualquier álgebra de Boole, incluido el álgebra de conmutación.

  La aplicación de estos teoremas permite la modificación o la simplificación de expresiones lógicas por otras equivalentes.

- Principio de dualidad: los postulados presentan dos versiones intercambiando ($1 \Leftrightarrow 0$), y ($+ \Leftrightarrow \bullet$).

  Esto implica que demostrado un teorema determinado, haciendo los intercambios anteriores en la definición del teorema queda determinado un nuevo teorema.

  Por ejemplo: si demuestro que $X + 1 = 1 \Leftrightarrow X \bullet 0 = 0$ queda demostrado por dualidad.

- **T1**. Teorema de la doble complementación: $\overline{\overline{X}} = X$

- **T2**. Teorema de la idempotencia: $X + X = X$; $X \cdot X = X$

- **T3**. Teorema de la identidad: $X + 1 = 1$; $X \cdot 0 = 0$

- **T4**. Teorema de absorción:
  $X + X \cdot Y = X$; $X \cdot (X + Y) = X$

- **T5**. Propiedad asociativa:
  $X + (Y + Z) = (X + Y) + Z$; $X \cdot (Y \cdot Z) = (X \cdot Y) \cdot Z$
  Este teorema indica que se pueden utilizar puertas lógicas de 3, 4, … entradas.

- **T6**. Teorema de DeMorgan:

$$\overline{X + Y} = \overline{X} \cdot \overline{Y}; \qquad \overline{X \cdot Y} = \overline{X} + \overline{Y}$$

- **T7**. Teorema de adyacencia:

$$X \cdot Y + X \cdot \overline{Y} = X; \qquad (X + Y) \cdot (X + \overline{Y}) = X$$

- **T8**. Teorema del consenso:

$$X \cdot Y + \overline{X} \cdot Z + Y \cdot Z = X \cdot Y + \overline{X} \cdot Z$$

$$(X + Y) \cdot (\overline{X} + Z) \cdot (Y + Z) = (X + Y) \cdot (\overline{X} + Z)$$

- **T9**. Teorema de simplificación:

$$X + \overline{X} \cdot Y = X + Y; \qquad X \cdot (\overline{X} + Y) = X \cdot Y$$

Demostraciones de teoremas

- Teorema T3: $X + 1 = 1$

$X + 1 = (X + 1) \cdot 1;$	Postulado P4
$(X + 1) \cdot 1 = (X + 1) \cdot (X + \overline{X});$	Postulado P5
$(X + 1) \cdot (X + \overline{X}) = X + 1 \cdot \overline{X};$	Postulado P3
$X + 1 \cdot \overline{X} = X + \overline{X} \cdot 1;$	Postulado P2
$X + \overline{X} \cdot 1 = X + \overline{X};$	Postulado P4
$X + \overline{X} = 1;$	Postulado P5

- Teorema T4: $X + X \cdot Y = X$

$X + X \cdot Y = X \cdot 1 + X \cdot Y;$	Postulado P4
$X \cdot 1 + X \cdot Y = X \cdot (1 + Y);$	Postulado P3
$X \cdot (1 + Y) = X \cdot (Y + 1);$	Postulado P2
$X \cdot (Y + 1) = X \cdot 1;$	Teorema T3
$X \cdot 1 = X;$	Postulado P4

Representación de problemas lógicos

- Un problema lógico se corresponde con un enunciado en el que se puede describir el problema mediante relaciones entre variables que se pueden definir mediante los valores verdadero y falso (variables lógicas).

La alarma de un coche se enciende cuando se cierran las puertas sin ajustar los cinturones de seguridad, o cuando se enciende el motor estando las puertas abiertas.

Al (alarma encendida) => Encendida (V), Apagada (F)
Pu (puertas cerradas) => Cerrada (V), Abierta (F)
Ci (cinturón ajustado) => Ajustado (V), Suelto (F)
Mo (motor encendido) => Encendido (V), Apagado (F)

- Para la resolución del problema hay que plasmar el enunciado de forma que se pueda expresar como una serie de entradas y salidas de tipo lógico. Hay dos representaciones de los problemas:

  - Funciones Lógicas => Circuitos Lógicos

    Al = F (Pu, Ci, Mo) = Pu • $\overline{Ci}$ + Mo • $\overline{\mathrm{Pu}}$

  - Tabla de verdad:

Pu	Ci	Mo	Al
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

Funciones lógicas

- Una función lógica es una expresión matemática que evalúa cuando una variable lógica toma el valor lógico Verdadero en función de los valores (Verdadero o Falso) de otras variables lógicas operados mediante las operaciones AND, OR y NOT. Normalmente, para escribir las funciones lógicas se usan los valores (0, 1) y los operadores típicos ($\overline{\ }$, •, +) del álgebra de conmutación (de mayor a menor proridad, se pueden alterar mediante paréntesis).

$$Z = F1(X, Y, Z) = \overline{X} + Y \bullet \overline{Z};\quad K = F2(X, Y, Z) = \overline{X + Y} \bullet Z$$

$$T = F3(X, Y, Z) = \overline{(X + Y) \bullet \overline{\overline{Z}}};\quad R = F4(X, Y, Z) = \overline{X + \overline{\overline{Y} \bullet Z}}$$

Además, en circuitos digitales se usa también el operador ⊕ (EXOR), con esta tabla de operación y la misma prioridad que el operador +.

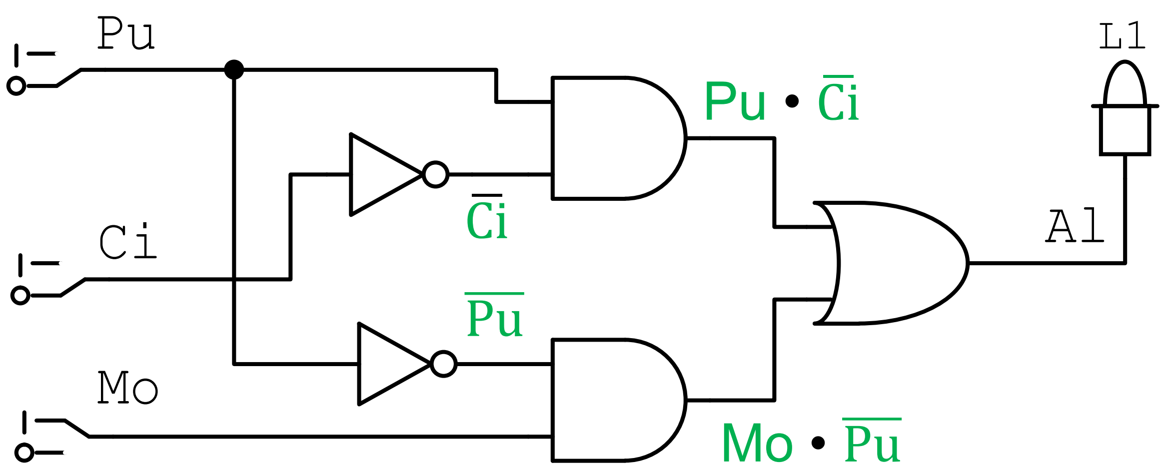
X Y	X ⊕ Y
0 0	0
0 1	1
1 0	1
1 1	0

$$F(X, Y, Z) = (X + Y Z) \oplus \overline{X} Z$$

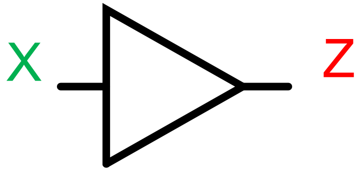
Puertas Lógicas

- Para una representación circuital de las funciones lógicas se utilizan puertas lógicas. Los circuitos lógicos se generan como una conexión de puertas lógicas.

$$Al = F(Pu, Ci, Mo) = Pu \cdot \overline{Ci} + Mo \cdot \overline{Pu}$$



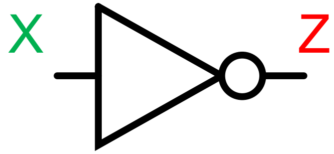
Puerta Buffer: 1 entrada



$Z = F(X) = X$

X	Z
0	0
1	1

Puerta NOT o Inversor: 1 entrada

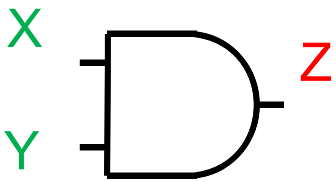


$Z = F(X) = \overline{X}$

X	Z
0	1
1	0

Puerta AND: 2 o más entradas.

La salida es 0 si alguna entrada es 0, si no es 1.

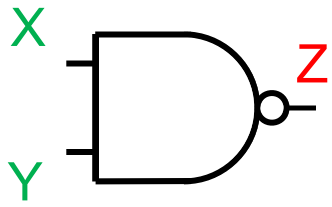


X	Y	Z
0	0	0
0	1	0
1	0	0
1	1	1

$Z = F(X, Y) = X \cdot Y$

Puerta NAND: 2 o más entradas.

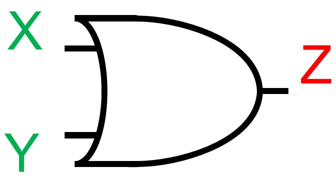
La salida es 1 si alguna entrada es 0, si no es 0.



X	Y	Z
0	0	1
0	1	1
1	0	1
1	1	0

$Z = F(X, Y) = \overline{X \cdot Y}$

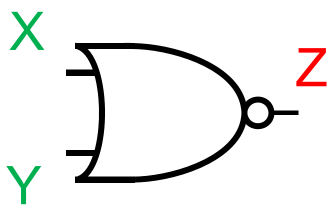
Puerta OR: 2 o más entradas. La salida es 1 si alguna entrada es 1, si no es 0.



X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	1

$Z = F(X, Y) = X + Y$

Puerta NOR: 2 o más entradas. La salida es 0 si alguna entrada es 1, si no es 0.

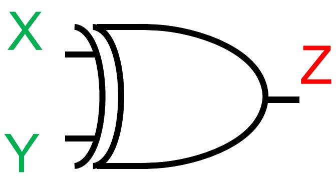


X	Y	Z
0	0	1
0	1	0
1	0	0
1	1	0

$Z = F(X, Y) = \overline{X + Y}$

Puerta EXOR*: 2 o más entradas. La salida es 1 si sólo una entrada es 1, si no es 0. (1) Op: $\oplus$

El número de entradas a 1 es impar. (2)

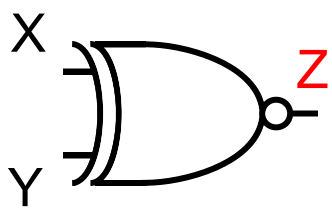


X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	0

$Z = F(X, Y) = X \oplus Y = \overline{X}\,Y + X\,\overline{Y}$

Puerta EXNOR*: 2 o más entradas. La salida es 1 si las entradas son iguales, si no es 0. (1) Op: $\odot$

El número de entradas a 1 es 0 o par. (2)

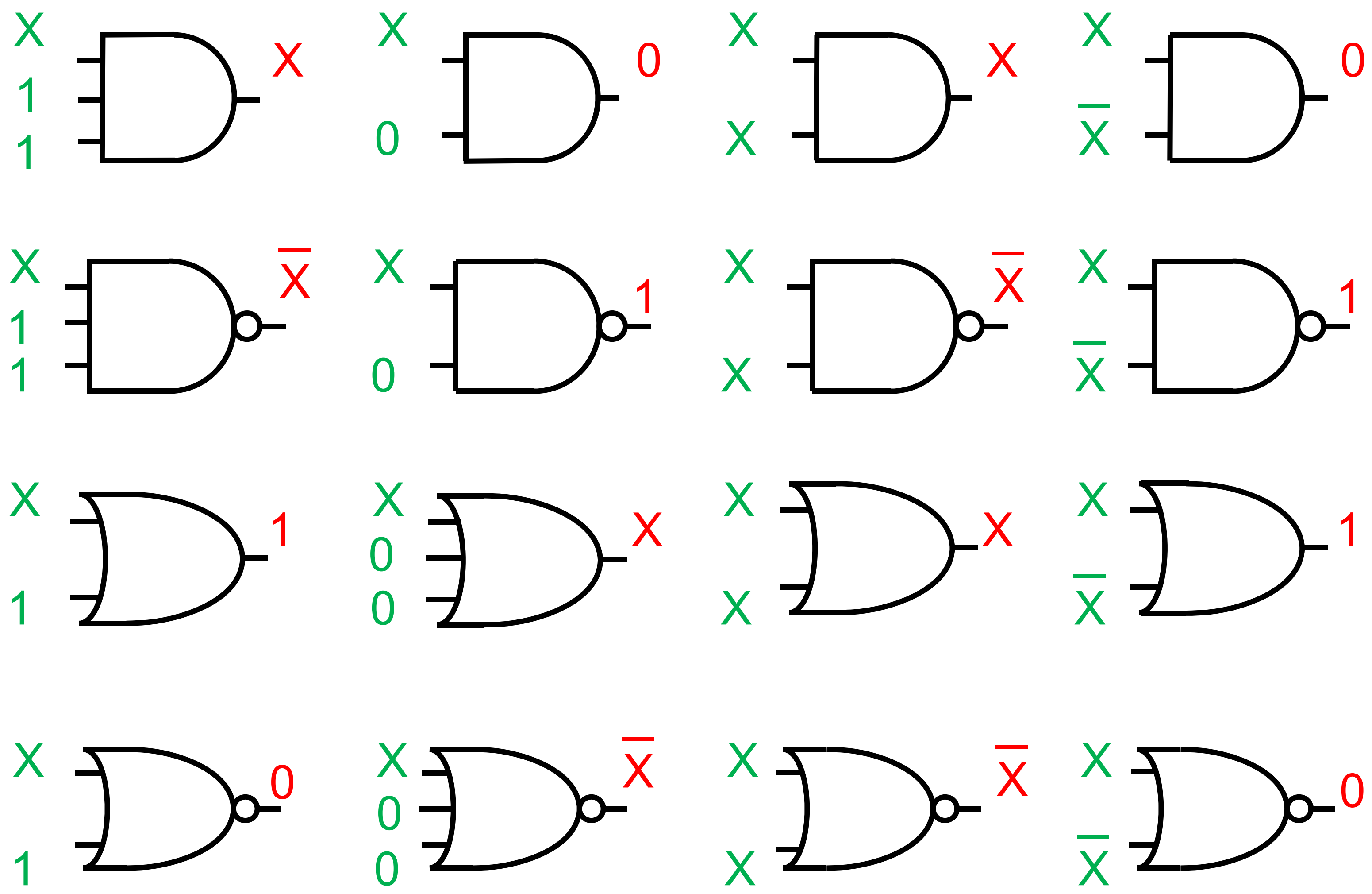


X	Y	Z
0	0	1
0	1	0
1	0	0
1	1	1

$Z = F(X, Y) = \overline{X \oplus Y} = \overline{X}\,\overline{Y} + X\,Y$

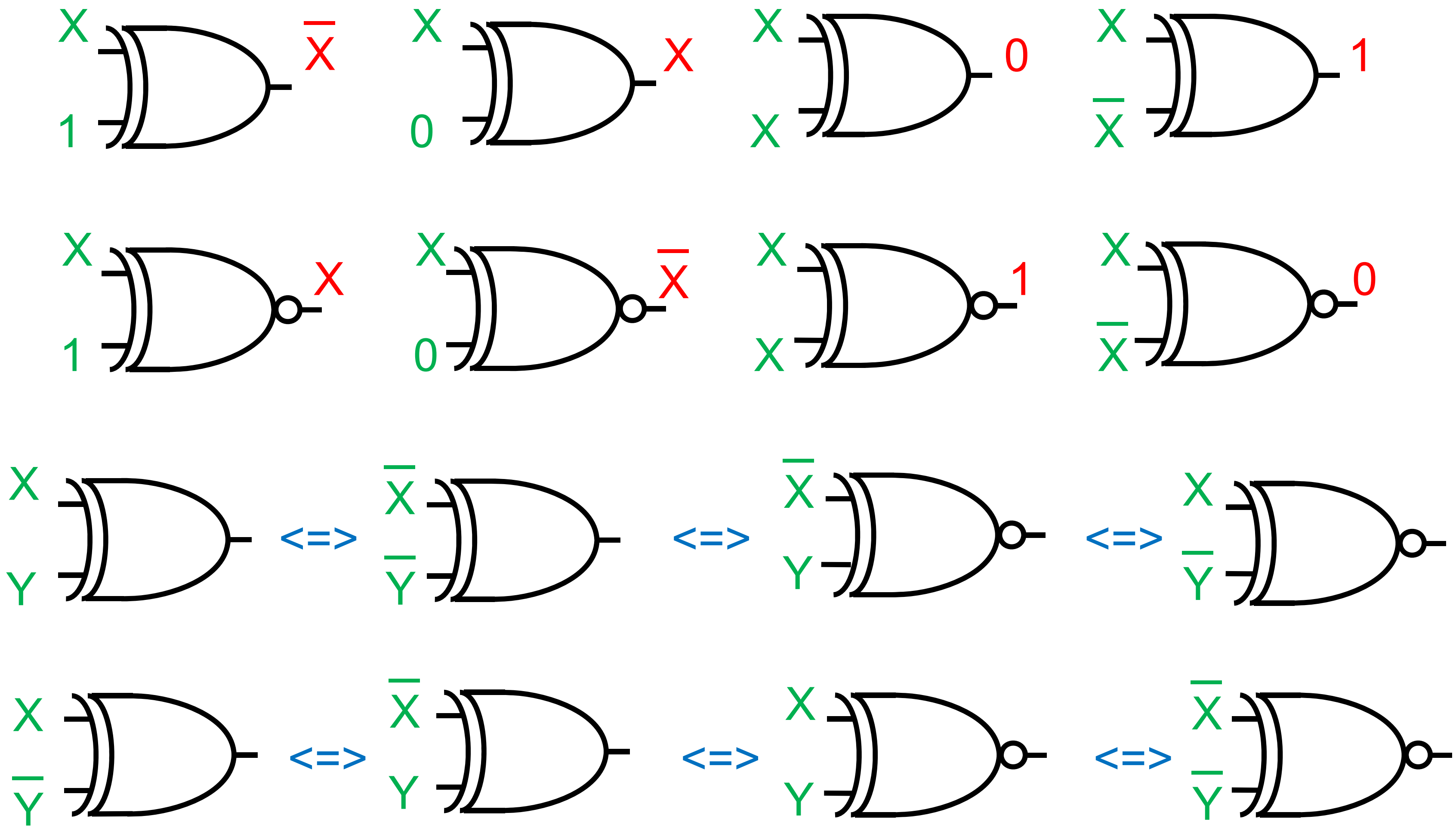
* Para más de 2 entradas las definiciones 1 y 2 son distintas, y $\odot$ no es el complemento de $\oplus$

Operaciones básicas en puertas lógicas



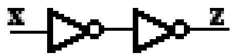
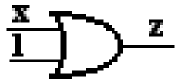
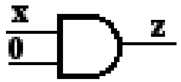
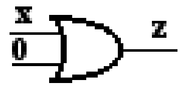
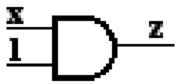
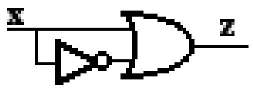
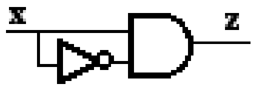
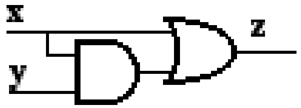
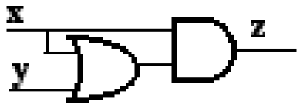
Operaciones básicas en puertas lógicas

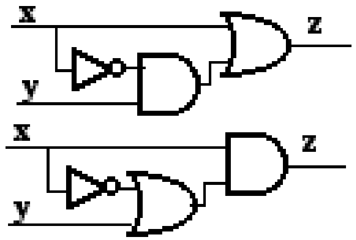
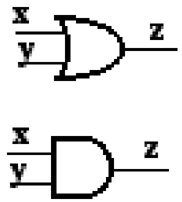
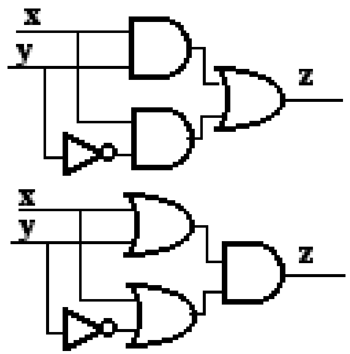
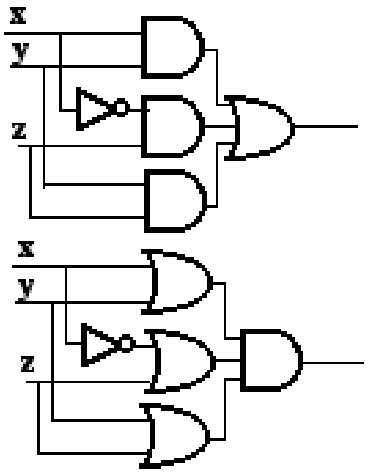
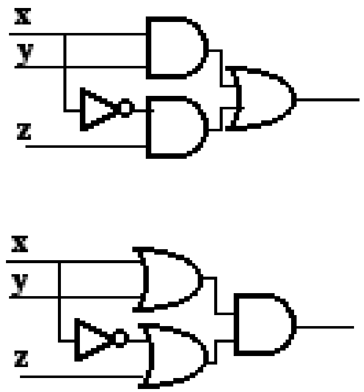
$$X \oplus Y = \overline{X}\,Y + X\,\overline{Y} \qquad\qquad \overline{X \oplus Y} = \overline{X}\,\overline{Y} + X\,Y$$



Simplificación de Funciones Lógicas

- Una misma especificación lógica puede expresarse por muchas funciones lógicas diferentes, sustituyendo términos con ayuda de los teoremas y postulados del álgebra de Boole.
- Funciones lógicas distintas dan lugar a circuitos lógicos distintos. Normalmente nos interesa un circuito lo más pequeño posible => una función lógica con el menor número de términos y operaciones.
- Los postulados y los teoremas del álgebra de conmutación muestran ejemplos de reducciones de circuitos digitales.

EXPRESION	CIRCUITO ORIGINAL	CIRCUITO SIMPLIFICADO
$\overline{\overline{x}} = x$		x ———
$x + 1 = 1$ $x \cdot 0 = 0$	 	1 ——— 0 ———
$x + 0 = x$ $x \cdot 1 = x$	 	x ——— x ———
$x + \overline{x} = 1$ $x \cdot \overline{x} = 0$	 	1 ——— 0 ———
$x + x \cdot y = x$ $x \cdot (x + y) = x$	 	x ——— x ———

$x + \bar{x} \cdot y = x + y$ $x \cdot (\bar{x} + y) = x \cdot y$		
$x \cdot y + x \cdot \bar{y} = x$ $(x + y) \cdot (x + \bar{y}) = x$		
$x \cdot y + \bar{x} \cdot z + y \cdot z =$ $x \cdot y + \bar{x} \cdot z$ $(x + y) \cdot (\bar{x} + z) \cdot (y + z) =$ $(x + y) \cdot (\bar{x} + z)$		

Errores comunes en simplificaciones lógicas

- Intentar realizar solo "multiplicaciones" para reducir al final, sin utilizar todos los postulados y teoremas disponibles.

- No utilizar nunca la propiedad distributiva: (X + Y) (X + Z) = X + Y Z

- Aplicar mal los teoremas de Demorgan:

$$\overline{X + Y} = \overline{X} + \overline{Y}$$

$$\overline{X\,Y} = \overline{X}\;\overline{Y}$$

MAL

- No utilizar paréntesis al eliminar complementos de expresiones:

$$\overline{X\,Y}\;Z = \overline{X} + \overline{Y}\,Z$$ MAL

$$\overline{X\,Y}\;Z = (\overline{X} + \overline{Y})\,Z$$ BIEN

Hay que suponer que debajo del complemento siempre hay un paréntesis.

$$(\overline{X\,Y})$$

Reglas para hacer simplificaciones lógicas

- Los postulados conmutativo (P2), del elemento neutro (P4) y complementado (P5) y los teoremas de la doble complementación (T1), idempotencia (T2), identidad (T3) y asociativa (T5) se deben aplicar de forma casi inmediata.

- Los teoremas de Demorgan (T6) y las propiedades distributivas (P3) se utilizan para modificar expresiones para simplificarlas mejor.

- La simplificación se hace con los teoremas de absorción (T4), adyacencia (T7), consenso (T8) y simplificación (T9).

- Los teoremas de Demorgan (T6) pueden aplicarse así:

$$\overline{X+Y+Z} = \overline{X}\,\overline{Y}\,\overline{Z} \qquad \overline{X\,Y\,Z} = \overline{X}+\overline{Y}+\overline{Z}$$

$$\overline{X\,(Y+Z)} = \overline{X}+\overline{Y}\,\overline{Z}$$

$$\overline{\overline{X}\,\overline{Y}+\overline{Z}} = (X+Y)\,Z$$

En expresiones con literales y operadores +, •, los operadores y los complementos de los literales se intercambian, ajustando los paréntesis.

Ejemplos de simplificaciones

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

T. de absorción: $X(X+Y) = X$

$X = \overline{C} + D; \quad Y = E$

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

P. Distributiva: $(X+Y)(X+Z) = X+YZ$

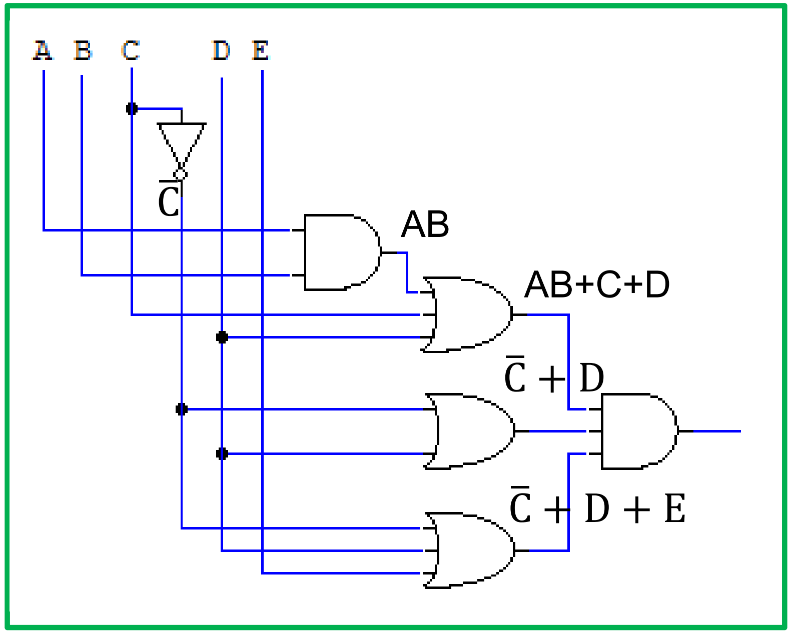
$X = D; \; Y = AB+C; \; Z = \overline{C}$

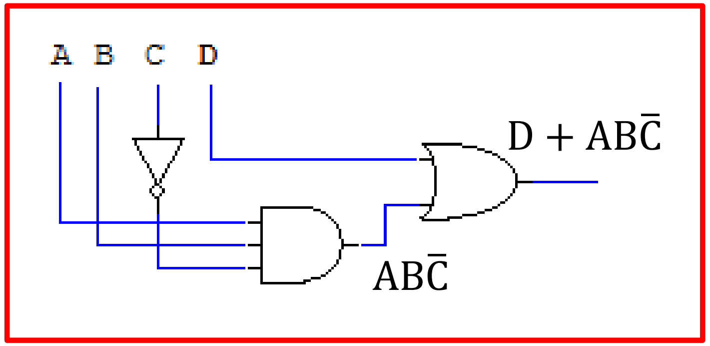
$= D + \overline{C}(C + AB) =$

T. de simplificación: $X(\overline{X}+Y) = XY$

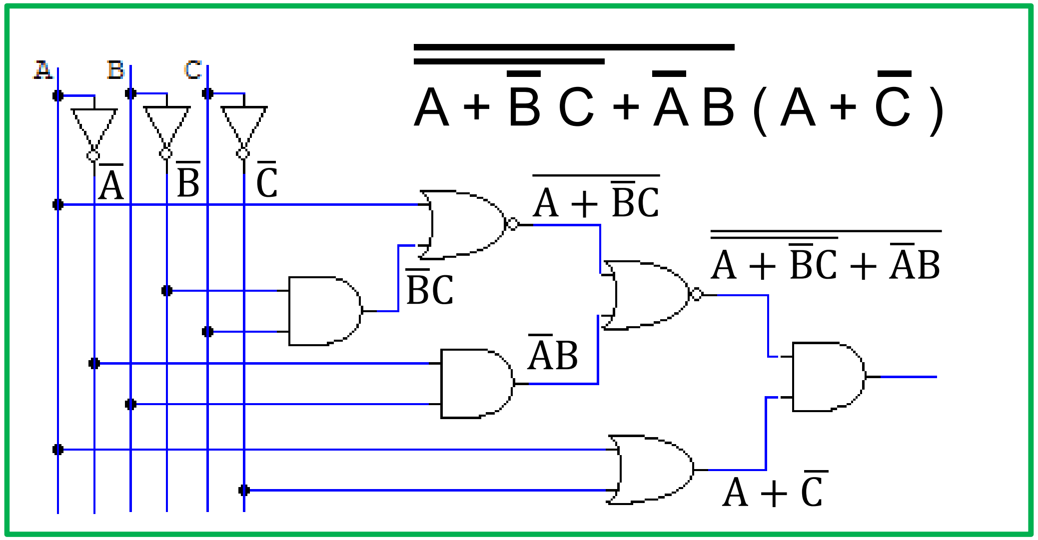
$X = \overline{C}; \; \overline{X} = \overline{\overline{C}} = C; \; Y = AB$

$= D + AB\overline{C}$





Ejemplos de simplificaciones



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

L. de DeMorgan: $\overline{X+Y} = \overline{X}\;\overline{Y}$

$X = \overline{A+\overline{B}\,C}$; $Y = \overline{A}\,B$

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

L. de DeMorgan: $\overline{XY} = \overline{X}+\overline{Y}$; $\overline{\overline{X}} = X$

$X = \overline{A}$; $Y = B$

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

P. Distributiva: $(X+K)(X+Y)(X+Z) = X+KYZ$

$X = A$; $K = \overline{B}C$; $Y = \overline{B}$; $Z = \overline{C}$

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

P. de complemento: $X\,\overline{X} = 0$; $X = C$

T. de idempotencia: $X\,X = X$; $X = \overline{B}$

$$= A + 0\,\overline{B} = A$$

T. de identidad: $X\,0 = 0$; $X = \overline{B}$

P. Elem. Neutro: $X + 0 = X$; $X = A$

Tabla de verdad

- La tabla de verdad es una representación de un problema lógico mediante una tabla en la que se indica el valor lógico que toma la salida(s) en función del valor lógico que toman las entradas.
- Existen problemas que no pueden pasarse de forma directa a una función lógica:

Una sociedad está formada por 5 socios A, B, C, D y E que tienen respectivamente el 25%, 25%, 25%, 15% y 10% de las acciones. Los estatutos de la sociedad indican que una toma de decisión es positiva si el tanto por ciento a favor es mayor del 65%, o si estando entre el 35% y el 65% (ambos inclusive) hay mayoría de votos a favor entre los tres socios más antiguos C, D y E (sin contar su porcentaje respectivo). En caso contrario, la decisión es negativa.

- Este enunciado no puede convertirse fácilmente en una función lógica. Una paso intermedio para llegar al circuito lógico es expresar el problema en una tabla de verdad.

A B C D E	%	nº votos	Z
0 0 0 0 0	0	0	0
0 0 0 0 1	10	1	0
0 0 0 1 0	15	1	0
0 0 0 1 1	25	2	0
0 0 1 0 0	25	1	0
0 0 1 0 1	35	2	1
0 0 1 1 0	40	2	1
0 0 1 1 1	50	3	1
0 1 0 0 0	25	0	0
0 1 0 0 1	35	1	0
0 1 0 1 0	40	1	0
0 1 0 1 1	50	2	1
0 1 1 0 0	50	1	0
0 1 1 0 1	60	2	1
0 1 1 1 0	65	2	1
0 1 1 1 1	75	3	1

A B C D E	%	nº votos	Z
1 0 0 0 0	25	0	0
1 0 0 0 1	30	1	0
1 0 0 1 0	40	1	0
1 0 0 1 1	50	2	1
1 0 1 0 0	50	1	0
1 0 1 0 1	60	2	1
1 0 1 1 0	65	2	1
1 0 1 1 1	75	3	1
1 1 0 0 0	50	0	0
1 1 0 0 1	60	1	0
1 1 0 1 0	65	1	0
1 1 0 1 1	75	2	1
1 1 1 0 0	75	1	1
1 1 1 0 1	85	2	1
1 1 1 1 0	90	2	1
1 1 1 1 1	100	3	1

- La tabla de verdad de un problema lógico es única. Sin embargo, un problema lógico puede expresarse por muchas funciones lógicas diferentes (aunque equivalentes).

- De una función lógica se puede obtener la tabla de verdad operando.

$$Al = F(Pu, Ci, Mo) = Pu \cdot \overline{Ci} + Mo \cdot \overline{Pu}$$

Pu Ci Mo	$\overline{Ci}$	$\overline{Pu}$	Pu $\overline{Ci}$	Mo $\overline{Pu}$	Al
0 0 0	1	1	0	0	0
0 0 1	1	1	0	1	1
0 1 0	0	1	0	0	0
0 1 1	0	1	0	1	1
1 0 0	1	0	1	0	1
1 0 1	1	0	1	0	1
1 1 0	0	0	0	0	0
1 1 1	0	0	0	0	0

- De una tabla de verdad se puede obtener una función lógica siguiendo este razonamiento:

  La función es 1 si los valores de las entradas coinciden con los de una **u** otra (OR) de las filas de la tabla de verdad que producen 1.

  Coincidir con una fila significa que todas las entradas (AND) tienen el valor de la entrada en la fila, donde 1 es la entrada y 0 la entrada complementada.

Forma canónica SOP (suma de minterms)

$$F(Pu, Ci, Mo) = \overline{Pu}\,\overline{Ci}\,Mo + \overline{Pu}\,Ci\,Mo + Pu\,\overline{Ci}\,\overline{Mo} + Pu\,\overline{Ci}\,Mo$$

Pu	Ci	Mo	Al
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

$\overline{Pu}\,\overline{Ci}\,Mo$

$\overline{Pu}\,Ci\,Mo$

$Pu\,\overline{Ci}\,\overline{Mo}$

$Pu\,\overline{Ci}\,Mo$

minterms

Reduciendo la función por T. de Adyacencia

Forma estándar SOP (suma de productos)

$$F(Pu, Ci, Mo) = \overline{Pu}\,Mo + Pu\,\overline{Ci}$$

- Otro razonamiento posible es:

  La función es 1 si los valores de las entradas no coinciden con ninguna (AND) de las filas de la tabla de verdad que producen 0.

  No coincidir con una fila significa que el valor de una u otra (OR) de las entradas es distinto del valor en la fila, para lo que 1 es la entrada complementada y 0 sin complementar.

Forma canónica POS (producto de Maxterms)

$$F(Pu, Ci, Mo) = (Pu + Ci + Mo)\,(Pu + \overline{Ci} + Mo)\,(\overline{Pu} + \overline{Ci} + Mo)\,(\overline{Pu} + \overline{Ci} + \overline{Mo})$$

Pu	Ci	Mo	Al	Maxterms
0	0	0	0	$Pu + Ci + Mo$
0	0	1	1	
0	1	0	0	$Pu + \overline{Ci} + Mo$
0	1	1	1	
1	0	0	1	
1	0	1	1	
1	1	0	0	$\overline{Pu} + \overline{Ci} + Mo$
1	1	1	0	$\overline{Pu} + \overline{Ci} + \overline{Mo}$

Reduciendo la función por T. de Adyacencia

Forma estándar POS (producto de sumas)

$$F(Pu, Ci, Mo) = (Pu + Mo)\,(\overline{Pu} + \overline{Ci})$$

Notación decimal de una tabla de verdad

- Para indicar una función lógica mediante su tabla de verdad se suele usar una notación decimal. Se supone que las entradas forman un código binario con pesos de derecha a izquierda 1, 2, 4, 8, …. Se indica la función como un sumatorio de combinaciones que producen 1 o como un productorio de las combinaciones que producen 0.

	Pu	Ci	Mo	Al
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	1
5	1	0	1	1
6	1	1	0	0
7	1	1	1	0

$$F(Pu, Ci, Mo) = \sum (1, 3, 4, 5) = \prod (0, 2, 6, 7)$$

$$\overline{F}(Pu, Ci, Mo) = \sum (0, 2, 6, 7) = \prod (1, 3, 4, 5)$$

Funciones incompletamente especificadas

- Existen problemas en los que no están definidas todas las combinaciones de las entradas.

  Indicar si una palabra de un código NBCD es múltiplo de 3.

  De las 16 combinaciones solo tienen sentido de 0 (0000) a 9 (1001). Las combinaciones 10-15 no tienen sentido, para ellas la salida no está definida, puede ser 0 o 1 según convenga. Se dice que la salida es "don't care" (no importa): $\varnothing$.

  Para el ejemplo:

  $$F(a3, a2, a1, a0) = \sum (0, 3, 6, 9) + \sum_{\varnothing}(10, 11, 12, 13, 14, 15)$$
  $$F(a3, a2, a1, a0) = \prod (1, 2, 4, 5, 7, 8) \bullet \prod_{\varnothing}(10, 11, 12, 13, 14, 15)$$