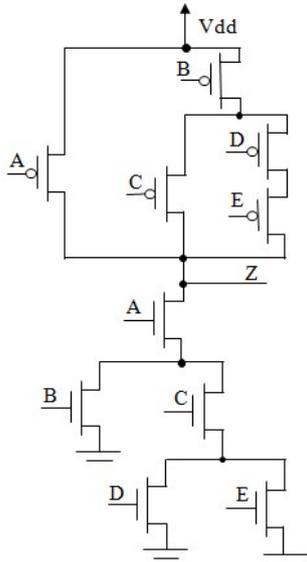


**Grado en Ingeniería de Tecnologías de Telecomunicación.
Electrónica Digital I. Problemas resueltos. Tema IIIa.**

Página 1_1. Encontrar la función lógica que realizan los siguientes circuitos MOS:

a)



Parte PMOS:

D, E en serie => $W_p = D + E$

C, W_p en paralelo => $X_p = C \bullet W_p = C (D + E)$

B, X_p en serie => $Y_p = B + X_p = B + C (D + E)$

A, Y_p en paralelo => $Z_p = A \bullet Y_p = A [B + C (D + E)]$

Parte NMOS

D, E en paralelo => $W_n = D + E$

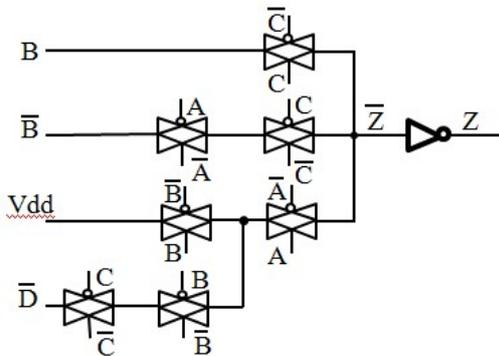
C, W_n en serie => $X_n = C \bullet W_n = C (D + E)$

B, X_n en paralelo => $Y_n = B + X_n = B + C (D + E)$

A, Y_n en serie => $Z_n = A \bullet Y_n = A [B + C (D + E)]$

$$Z = \overline{Z_n} = \overline{Z_p} = \overline{A [B + C (D + E)]}$$

b) Busco cada fila de llaves, y compruebo que no hay combinaciones de valores en las entradas que generen cortocircuito ni alta impedancia, por ejemplo, con un mapa de Karnaugh: busco las casillas que corresponden a los valores de las señales de control, y relleno con el valor de la señal de datos. Después extraigo una función lógica: la función lógica de cada fila es el AND de señales de control que activan las llaves y de la entrada de dato, y genero la función lógica \overline{Z} como el OR de la función de las filas y, si es posible, la reduzco con un mapa de Karnaugh.



Si C es 1, $\overline{Z1}$ es B => $\overline{Z1} = C B$

Si C es 0 y A es 0, $\overline{Z2}$ es \overline{B} => $\overline{Z2} = \overline{A} \overline{C} \overline{B}$

Si A es 1 y B es 1, $\overline{Z3}$ es 1 => $\overline{Z2} = B A$

Si A es 1, y B y C son 0, $\overline{Z4}$ es \overline{D} =>
=> $\overline{Z4} = \overline{C} \overline{B} A \overline{D}$

	CD		$\overline{A} \overline{C} \overline{B}$		
AB	00	01	11	$C B$ 10	
00	1	1	0	0	
01	0	0	1	1	
11	1	1	1	1	
BA					
10	1	0	0	0	
	$\overline{C} \overline{B} A \overline{D}$		\overline{Z}		



	CD				
AB	00	01	11	10	
00	1	1	0	0	
01	0	0	1	1	
11	1	1	1	1	
10	1	0	0	0	
	\overline{Z}				

	CD				
AB	00	01	11	10	
00	0	0	1	1	
01	1	1	0	0	
11	0	0	0	0	
10	0	1	1	1	
	Z				

En el mapa de Karnaugh no hay cortocircuitos ni altas impedancias. Se puede encontrar una función lógica: $\bar{Z} = BC + \bar{A}\bar{B}\bar{C} + AB + A\bar{B}\bar{C}\bar{D} \Rightarrow Z = \overline{BC + \bar{A}\bar{B}\bar{C} + AB + A\bar{B}\bar{C}\bar{D}}$

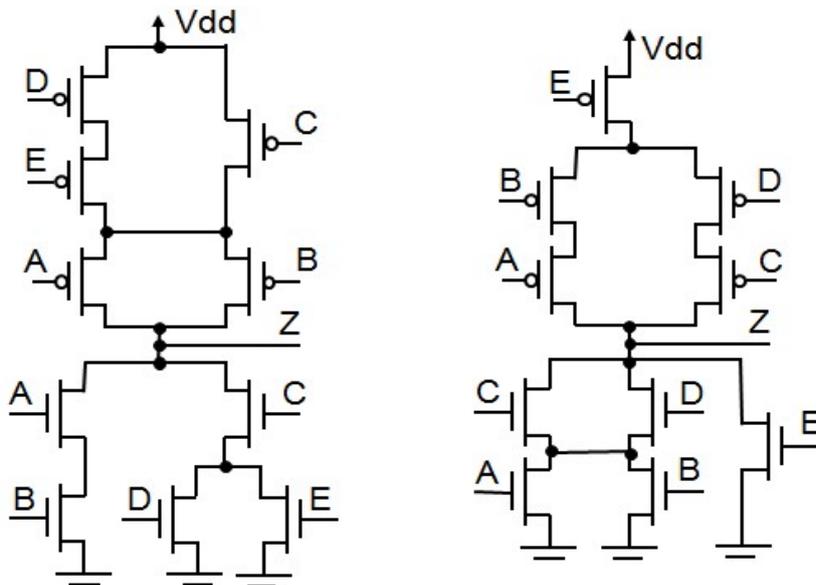
Con los mapas de Karnaugh se puede obtener:

$$\bar{Z} = BC + \bar{A}\bar{B}\bar{C} + AB + A\bar{B}\bar{C}\bar{D} \Rightarrow Z = \overline{BC + \bar{A}\bar{B}\bar{C} + AB + A\bar{B}\bar{C}\bar{D}}$$

$$Z = \overline{BC} + \overline{\bar{A}\bar{B}\bar{C}} + \overline{AB} + \overline{A\bar{B}\bar{C}\bar{D}}$$

Página 1_2. Diseñar las siguientes funciones lógicas usando circuitos CMOS

$$Z = \overline{AB + C(D + E)} \quad Z = \overline{(A + B)(C + D) + E}$$



Página 2_1. Dados los parámetros estáticos típicos de las puertas TTL y de las puertas 4000 CMOS y suponiendo Vcc = 5V. ¿Está garantizado el correcto funcionamiento del circuito si se conectan las salidas de puertas de una tecnología a las entradas de puertas de la otra?

Hay que obtener los parámetros críticos (Voh, Vih, Vol, Vil) de las familias lógicas y comprobar que, al conectar la salida de una puerta A a la entrada de una puerta B, se cumple que Voh(A) > Vih(B), y que Vol(A) < Vil(B). Los valores están en las diapositivas 28 y 29 de teoría.

Familia	Voh	Vih	Vol	Vil	A => B	Voh > Vih	Vol < Vil
TTL	2.4 V	2 V	0.5 V	0.8 V	TTL => MOS	NO	SI
CMOS	4.95 V	3.5 V	0.05 V	1.5 V	MOS => TTL	SI	SI

Al conectar la salida de una puerta MOS a la entrada de una TTL la conexión funciona bien a valor alto H, o bajo L. Pero al conectar la salida de una puerta TTL a una MOS el circuito podría no funcionar como circuito digital.

Además de por niveles de voltaje la conexión también debe funcionar por niveles de intensidad. En una conexión, la intensidad que genera la salida de una puerta tiene que ser mayor que la

que requiere la entrada de otra puerta ($|I_o| > |I_i|$), tanto para tensión alta H como baja L. Las intensidades en circuitos TTL se pueden dar en valores numéricos o en unidades de carga (U.L.), donde una U.L.= 40 uA(H)/1.6 mA(L), la intensidad de entrada de una puerta TTL estándar, y varían según el tipo de familia (diapositiva 28 de teoría), I_i es del orden de 1 U.L., e I_o es del orden 10 U.L. En los circuitos MOS 4000, por defecto la intensidad I_i es muy baja (del orden 1 uA como máximo, normalmente menor) y la intensidad de salida depende de la familia (UB, B), pero se puede suponer un mínimo valor de $I_o = \pm 0.5$ mA (intensidad típica $I_o = \pm 1$ mA). En los valores numéricos las intensidades positivas entran en un nudo, ya sea de entrada o de salida de la puerta, y las negativas salen del nudo.

Por intensidad, una salida TTL siempre se puede conectar a una entrada MOS 4000, ya que la I_i del circuito MOS es muy pequeña. Al revés, cuando se conecta una salida MOS a una entrada TTL, no hay problema en las tensiones altas H, ya que en cualquier familia el valor máximo de I_{ih} es $60 \text{ uA} < 500 \text{ uA}$. Sin embargo, en algunas de las familias TTL: TTL ($I_{il} = 1.6 \text{ mA}$), STTL ($I_{il} = 2 \text{ mA}$), e incluso la familia FAST ($I_{il} = 0.6 \text{ mA}$), la tensión baja L podría operar mal, porque $I_{oh_{MOS}} = 0.5 \text{ mA} < I_{ih_{TTL}}$.

Página 2_2. La salida de una puerta de tipo LS TTL está conectada a tres entradas de puertas de tipo LS TTL. ¿A cuántas puertas de tipo S TTL se puede conectar?

Este es un problema que se resuelve por intensidades. Hay que encontrar la intensidad que produce la salida de una puerta LS TTL a tensión alta (I_{oh}) y baja (I_{ol}), y la intensidad que requiere cada entrada de las puertas LS y S TTL a tensión alta (I_{ih}) y baja (I_{il}). Las intensidades se pueden dar en valores numéricos o en unidades de carga (U.L.), donde una U.L.= 40 uA(H)/1.6 mA(L), la intensidad de entrada de una puerta TTL estándar. En los valores numéricos las intensidades positivas entran en un nudo, ya sea de entrada o salida de la puerta, y las negativas salen del nudo.

Una conexión funciona bien si $|I_{oh}| > \sum_i |I_{ih}|$ y $|I_{ol}| > \sum_i |I_{il}|$, para todas las entradas i que estén conectadas a la salida de la puerta. Desarrollo el problema en unidades de carga (diapositiva 28 de teoría).

U.L.	$I_o(LS)$	$I_i(LS)$	$I_i(S)$
L	5	0.25	1.25
H	10	0.5	1.25

Si la salida del circuito LS tiene conectadas 3 entradas LS quedan disponibles para las entradas S:

- L. $I_{disp}(L) = 5 \text{ U.L.} - 3 * 0.25 \text{ U.L} = 4.25 \text{ U.L.}$
- H. $I_{disp}(H) = 10 \text{ U.L.} - 3 * 0.5 \text{ U.L} = 8.5 \text{ U.L.}$

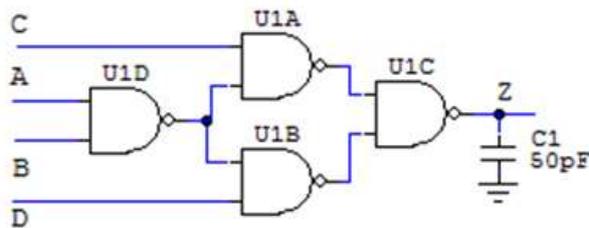
El número N de entradas S que se pueden conectar es la parte entera de:

- NL. $I_{disp}(L)/I_{il}(S) = 4.25 \text{ U.L.} / 1.25 \text{ U.L.} = 3.6 \Rightarrow NL = 3$
- NH. $I_{disp}(H)/I_{ih}(S) = 8.5 \text{ U.L.} / 1.25 \text{ U.L.} = 6.8 \Rightarrow NH = 6$

El número final N de entradas S TTL que se pueden conectar es el mínimo de NL y NH, luego $N = 3$.

Página 2_3. El tiempo de propagación de una puerta HCMOS está dada por una expresión $t_p(C_r) = t_p(C_l) + K (C_r - C_l)$, donde C_l y C_r son la capacidad de la carga de referencia y la carga real, respectivamente. $K = 0.5 V_{cc}/I_{OS}$, donde V_{cc} es la tensión de alimentación e I_{OS} la intensidad de salida en cortocircuito para esa V_{cc} . Obtener el tiempo de propagación máximo del siguiente circuito para $V_{cc} = 5V$, $I_{OS} = 20 \text{ mA}$, $t_{phl}(15\text{pF}) = t_{plh}(15\text{pF}) = 15\text{ns}$, y la capacidad de una entrada $C_{in} = 10 \text{ pF}$.

Obtener la potencia disipada por el circuito, suponiendo que todas las puertas son del mismo circuito "Quad 2-input NAND 74HC00". Suponer que la frecuencia F de operación de A y B es de 1 Mhz , $I_{cc}(\text{max}) = 2\mu\text{A}$ (intensidad estática en la fuente) y la capacidad interna de cada puerta es $C_{pd} = 20\text{pF}$.



El tiempo de propagación máximo desde A ó B hasta Z , se puede medir por un camino formado por 3 puertas NAND $D-A-C$ ó $D-B-C$, según la figura. Elijo uno de ellos, ya que los dos producen el mismo resultado, porque los nudos tienen las mismas capacidades de carga por los dos caminos. Además, los tiempos H_L y L_H son los mismos, por lo que sólo considero un tiempo genérico T_p . Elijo el camino de A a Z a través de las puertas D , A y C . Para activar ese camino hay que realizar una transición H_L ó L_H en A , y para sensibilizarlo hay que fijar las entradas B y C a H (valor no controlante en las otras entradas de las puertas NAND), y la entrada D a L (para fijar a H a la otra entrada de la puerta C).

De esta forma $T_p = T_p(D) + T_p(A) + T_p(C)$.

Para cada puerta el T_p está medido para 15 pf de capacidad de carga, pero la carga real de cada puerta es distinta, en las puertas $C_r(i) = N(i) * C_{in}$, siendo $C_{in} = 10 \text{ pf}$, y $N(i)$ el número de entradas de otras puertas conectadas a la salida de la puerta $\Rightarrow N(D) = 2$, $N(A) = 1 \Rightarrow C_r(D) = 20 \text{ pF}$, $C_r(A) = 10 \text{ pf}$; $C_r(C) = 50 \text{ pf}$, explícitamente. Luego:

- $T_p(D) = 15 \text{ ns} + K * (20 \text{ pf} - 15 \text{ pf}) = 15\text{ns} + K * 5 \text{ pf}$
- $T_p(A) = 15 \text{ ns} + K * (10 \text{ pf} - 15 \text{ pf}) = 15\text{ns} - K * 5 \text{ pf}$
- $T_p(C) = 15 \text{ ns} + K * (50 \text{ pf} - 15 \text{ pf}) = 15\text{ns} + K * 35 \text{ pf}$

Sumando: $T_p = 3 * 15 \text{ ns} + 35 \text{ pf} * K = 45 \text{ ns} + (35 \text{ pf} * 0.5 * 5 \text{ V})/20 \text{ mA} = 45 \text{ ns} + 4.375 \text{ ns}$
 $T_p = 49.375 \text{ ns}$

Para obtener la potencia disipada por el circuito hay que calcular la potencia estática y la dinámica. Como solo se usa un circuito 74HC00 la potencia estática se calcula de los datos de su hoja de características. $P_{est} = I_{cc}(\text{max}) * V_{cc} = 2 \mu\text{A} * 5 \text{ V} = 10 \mu\text{W}$

La potencia dinámica se calcula para cada puerta i como $P_{din}(i) = C(i) * V_{cc}^2 * F$, donde $C(i)$ es la capacidad asociada a cada puerta y F la frecuencia de operación. Al ser V_{CC} y F común para todas las puertas $P_{din} = V_{cc}^2 * F * \sum_i C(i)$. Cada $C(i) = C_{pd} + C_r(i)$.

- $C(D) = 20 \text{ pF} + 2 * 10 \text{ pF} = 40 \text{ pF}$
- $C(A) = C(B) = 20 \text{ pF} + 1 * 10 \text{ pF} = 30 \text{ pF}$
- $C(C) = 20 \text{ pF} + 50 \text{ pF} = 70 \text{ pF}$

$$P_{din} = V_{cc}^2 * F * \sum_i C(i) = (5 \text{ V})^2 * 1 \text{ MHz} * 170 \text{ pF} = 4.25 \text{ mW}$$

$$P = P_{est} + P_{din} = 4.25 \text{ mW} + 0.01 \text{ mW} = 4.26 \text{ mW}.$$

Página 3_1. Dada la siguiente tabla de valores de tensión para entradas A, B y C y salida F (H valor de tensión más alto, L valor de tensión más bajo), indicar en notación decimal como suma de minterms o como producto de maxterms la función lógica correspondiente cuando:

- A.H, B.H, C.H y F.H.
- A.L, B.L, C.L y F.L.
- A.L, B.H, C.L, y F.H.

A	B	C	F
L	L	L	H
L	L	H	L
L	H	L	H
L	H	H	L
H	L	L	L
H	L	H	L
H	H	L	L
H	H	H	H

De la tabla de de tensiones, en valores de tensión altos (H) y bajos (L), sustituyo L por 0 y H por 1 en las señales en polaridad positiva (.H), y L por 1 y H por 0 en las señales de polaridad negativa (.L). Con ello determino la tabla de verdad, que expreso en notación decimal.

A	B	C	F
L	L	L	H
L	L	H	L
L	H	L	H
L	H	H	H
H	L	L	L
H	L	H	L
H	H	L	L
H	H	H	H

	4	2	1	
	.H	.H	.H	.H
A	B	C	F	
0	0	0	0	1
1	0	0	1	0
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	1

	4	2	1	
	.L	.L	.L	.L
A	B	C	F	
7	1	1	1	0
6	1	1	0	1
5	1	0	1	0
4	1	0	0	0
3	0	1	1	1
2	0	1	0	1
1	0	0	1	1
0	0	0	0	0

	4	2	1	
	.L	.H	.L	.H
A	B	C	F	
5	1	0	1	1
4	1	0	0	0
7	1	1	1	1
6	1	1	0	1
1	0	0	1	0
0	0	0	0	0
3	0	1	1	0
2	0	1	0	1

- $F(A, B, C) = \sum(0, 2, 3, 7) = \prod(1, 4, 5, 6)$ para A.H, B.H, C.H y F.H.
- $F(A, B, C) = \sum(1, 2, 3, 6) = \prod(0, 4, 5, 7)$ para A.L, B.L, C.L y F.L.
- $F(A, B, C) = \sum(2, 5, 6, 7) = \prod(0, 1, 3, 4)$ para A.L, B.L, C.L y F.L.

Página 3_2. Dada la función lógica descrita en notación decimal: $F(A, B, C) = \sum(0,2,4,5,7)$ representar la tabla de valores de tensión (valores H y L) para:

- A.H, B.H, C.H y F.H.
- A.L, B.L, C.L y F.L.
- A.L, B.H, C.L, y F.H.

Este problema es el inverso al anterior: dada la tabla de verdad en 0s y 1s, hay que encontrar las tablas de valores de tensión Hs y Ls, sustituyendo 0 por L y 1 por H en las señales en polaridad positiva (.H), y 0 por H y 1 por L en las señales de polaridad negativa (.L).

	4	2	1	
	A	B	C	F
0	0	0	0	1
1	0	0	1	0
2	0	1	0	1
3	0	1	1	0
4	1	0	0	1
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1

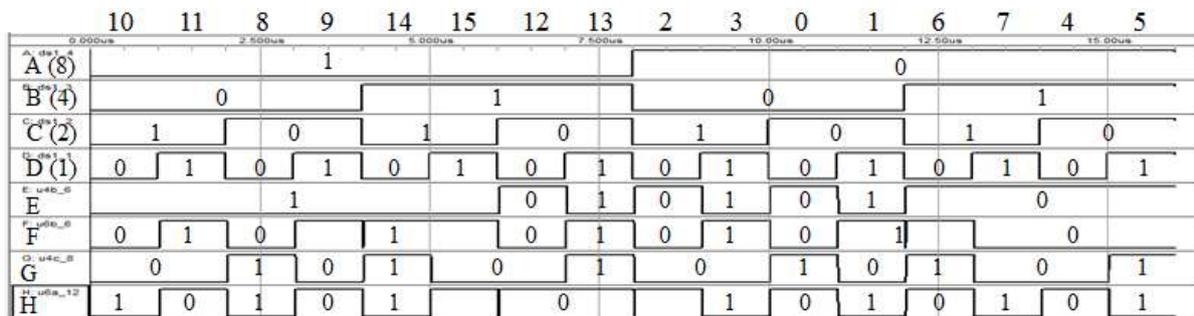
.H	.H	.H	.H
A	B	C	F
L	L	L	H
L	L	H	L
L	H	L	H
L	H	H	L
H	L	L	H
H	L	H	H
H	H	L	L
H	H	H	H

.L	.L	.L	.L
A	B	C	F
H	H	H	L
H	H	L	H
H	L	H	L
H	L	L	H
L	H	H	L
L	H	L	L
L	L	H	H
L	L	L	L

.L	.H	.L	.H
A	B	C	F
H	L	H	H
H	L	L	L
H	H	H	H
H	H	L	L
L	L	H	H
L	L	L	H
L	H	H	L
L	H	L	H

Página 4_1. Indicar la tabla de verdad (en notación decimal) y las formas SOP mínimas de un circuito digital partir de la siguiente simulación analógica para las entradas A.L, B.H, C.L y D.H y las salidas E.L, F.H, G.H y H.L (de arriba hacia abajo en la figura A, B, C, D, E, F, G y H).

Para obtener la tabla de verdad hay que tener en cuenta que en las señales .H (B, D, F y G), la tensión alta H es el 1 lógico y la tensión baja L es el 0; en las señales .L (A, C, E y H) es al revés L es 1 y H es 0. Si aplico esto, puedo obtener los valores 1 y 0 de las entradas y su valor en notación decimal, generando además los 0s y 1s de las salidas.



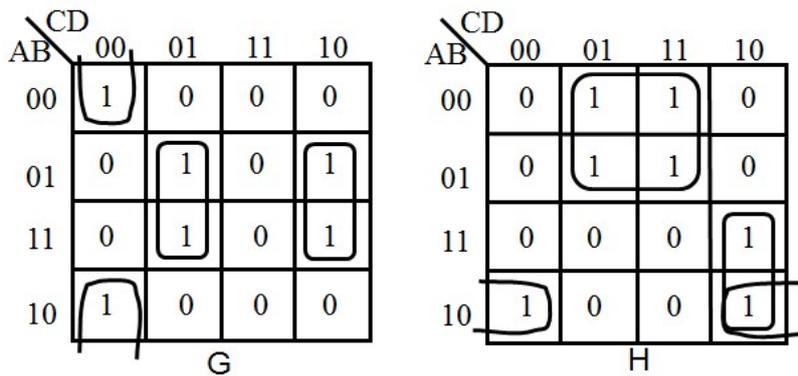
$E.L = E(A, B, C, D) = \sum (1, 3, 8, 9, 10, 11, 13, 14, 15)$
 $F.H = F(A, B, C, D) = \sum (1, 3, 6, 9, 11, 13, 14, 15)$
 $G.H = G(A, B, C, D) = \sum (0, 5, 6, 8, 13, 14)$
 $H.L = H(A, B, C, D) = \sum (1, 3, 5, 7, 8, 10, 14)$

		CD			
		00	01	11	10
AB	00	0	1	1	0
	01	0	0	0	0
	11	0	1	1	1
	10	1	1	1	1

E

		CD			
		00	01	11	10
AB	00	0	1	1	0
	01	0	0	0	1
	11	0	1	1	1
	10	0	1	1	0

F



Usando mapas de Karnaugh:

$$E.L = E(A, B, C, D) = A\bar{B} + AC + AD + \bar{B}D$$

$$F.H = F(A, B, C, D) = AD + \bar{B}D + BC\bar{D} + ABC$$

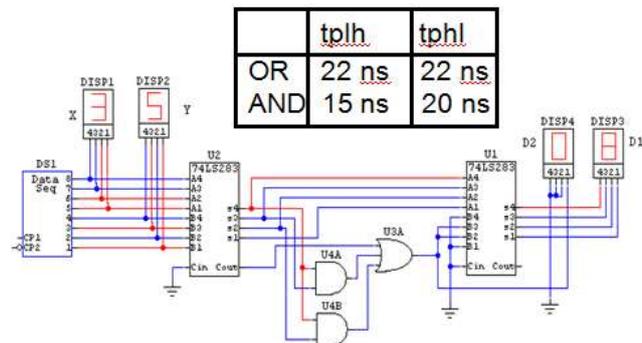
$$G.H = G(A, B, C, D) = \bar{B}\bar{C}\bar{D} + B\bar{C}D + BC\bar{D}$$

$$H.L = H(A, B, C, D) = \bar{A}D + A\bar{B}\bar{D} + AC\bar{D}$$

Página 4_2. Indicar el tiempo de propagación máximo del circuito de la figura usando los tiempos de propagación de sus componentes.

AC CHARACTERISTICS (TA=25°C)

Symbol	Parameter	Limits			Unit
		Min	Typ	Max	
tPLH	Propagation Delay, C0 Input to any Z Output		16	24	ns
tPHL	Propagation Delay, Any A or B Input to Z Outputs		15	24	ns
tPLH	Propagation Delay, C0 Input to C4 Output		11	17	ns
tPHL	Propagation Delay, Any A or B Input to C4 Output		15	22	ns
tPLH	Propagation Delay, Any A or B Input to C4 Output		11	17	ns
tPHL	Propagation Delay, Any A or B Input to C4 Output		12	17	ns



Para medir el máximo tiempo de propagación hay encontrar el camino crítico. Este camino está formado por dos sumadores una puerta AND y una puerta OR, pero lo sumadores tienen hasta cuatro caminos posibles ($A, B \Rightarrow S$; $A, B \Rightarrow Cout$; $Cin \Rightarrow S$; $Cin \Rightarrow Cout$). Hay que elegir qué tiempo se utiliza de cada sumador. El sumador de la derecha no presenta transiciones en Cin (está a 0) ni utiliza Cout (desconectado). Luego su único tiempo es de A o B a S (24 ns según la tabla). El sumador de la izquierda también tiene Cin a 0, y además el camino de Cout va a través de la puerta OR, mientras que los caminos desde S (s_4, s_3 y s_2) van a través de una AND y una OR, luego el camino crítico es:

$$T_p = T_{p83}(A-S) + T_{pAND} + T_{pOR} + T_{p83}(B-S)$$

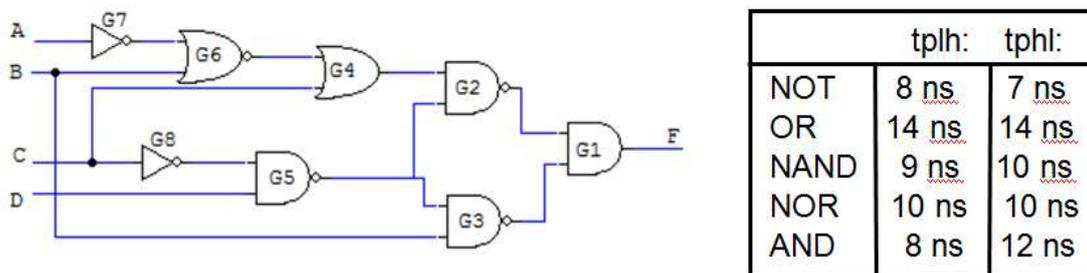
En general en circuitos como sumadores cambios en las entradas pueden producir cambios en las salidas HL y LH, por lo que se debería utilizar el mayor de los dos. En las puertas lógicas si se puede seguir los niveles HL ó LH. En el circuito AND-OR al no haber inversiones hay que usar el máximo de $t_{pHL}(AND) + t_{pHL}(OR)$ (42 ns según la tabla), o $t_{pLH}(AND) + t_{pLH}(OR)$ (37 ns según la tabla). Escogiendo el tiempo mayor:

$$T_p = 24 \text{ ns} + 20 \text{ ns} + 22 \text{ ns} + 24 \text{ ns} = 90 \text{ ns}$$

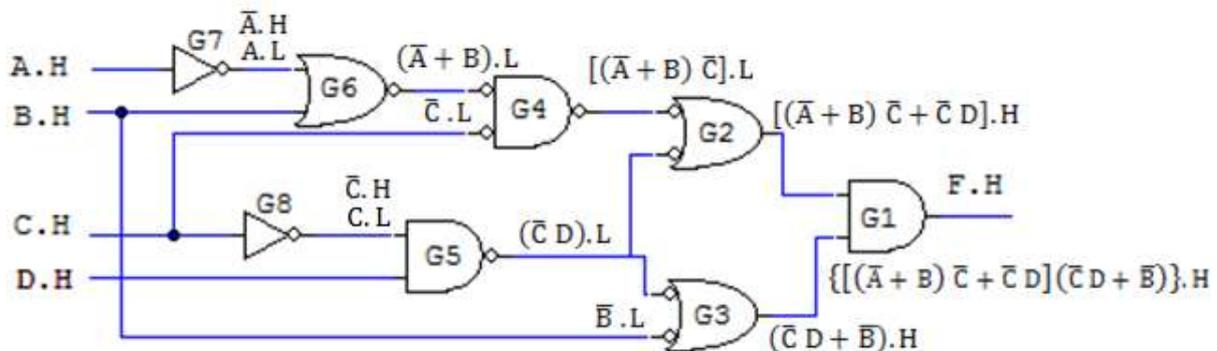
Página 5_1. Encontrar las expresiones lógicas factorizadas correspondiente a los siguientes circuitos digitales. Encontrar el tiempo máximo de propagación del circuito teniendo en cuenta los tiempos de propagación de las puertas lógicas, indicando los valores que hay que fijar en las entradas del circuito.

Este problema puede resolverse de varias formas, pero yo suelo usar la conversión de puertas lógicas (diapositiva 40 de teoría), mostrando los circuitos, de forma que en sus nudos internos coincidan las polaridades de la salida de la puerta que genera el nudo y las de las entradas de las puertas conectadas. Si esto se consigue, el circuito queda como sucesivos niveles AND/OR/AND/OR ..., que se puede extraer como una función lógica factorizada. Solo queda entonces encontrar los literales de las entradas, según su polaridad y sus conversiones y operaciones descritas en la diapositiva 42 de teoría, y obtener la polaridad de la salida.

a)



Hago la conversión de puertas, y sobre ella genero las funciones lógicas hasta las salidas. Al no estar definida las polaridades de las entradas y de la salida, las supongo todas .H.

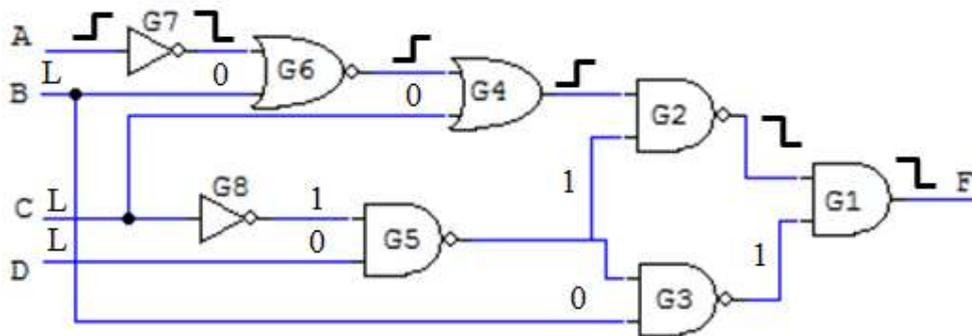


$$\text{Luego } F.H = \{[(\bar{A} + B) \bar{C} + \bar{C} D](\bar{C} D + \bar{B})\}.H$$

Para medir el tiempo de propagación busco el camino crítico que, por lo general, será el camino con más puertas. En este circuito el camino crítico va de A a F, por las puertas G7, G6, G4, G2, G1. Solo muestro una transición desde la entrada a la salida. La otra se consigue cambiando las subidas por bajadas, y viceversa. Las puertas inversoras cambian la transición de la entrada a la salida, las puertas no inversoras la mantienen igual. Para sensibilizar el camino de A a F hay que poner las otras entradas de las puertas del camino a valores no controlantes: 0 (o L) en puertas OR o NOR, y 1 (o H) en puertas AND o NAND. Las puertas EXOR y EXNOR sensibilizan tanto con 0 (mantiene en la salida la transición de la entrada) o 1 (cambia en la salida la transición de la entrada).

De la figura se tiene que, para medir los tiempos, B = C = D = 0 = L. En función de los tiempos de propagación de la tabla del enunciado, considerando los tiempos HL o HL de las puertas según lo que haga la transición en su salida, se calcula uno de los tiempos de propagación de

A a F, y cambiando los HL por LH y viceversa, se obtiene el otro. El tiempo de propagación máximo es 53 ns.

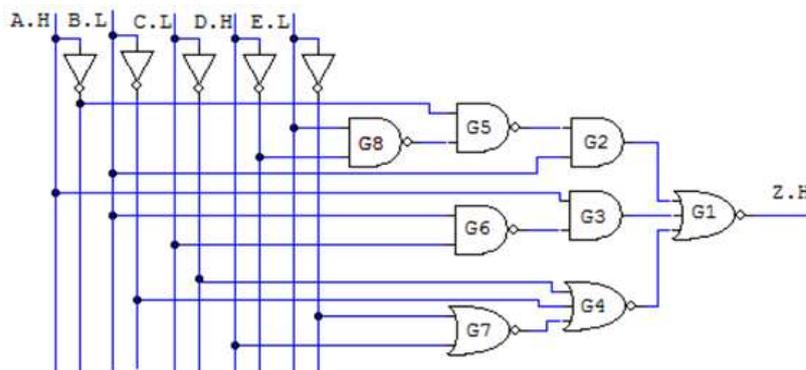


$$T_{pHL}(A-F) = T_{pHL}(G7) + T_{pLH}(G6) + T_{pLH}(G4) + T_{pHL}(G2) + T_{pHL}(G1) = 7 \text{ ns} + 10 \text{ ns} + 14 \text{ ns} + 10 \text{ ns} + 12 \text{ ns} = 53 \text{ ns}$$

$$T_{pLH}(A-F) = T_{pLH}(G7) + T_{pHL}(G6) + T_{pHL}(G4) + T_{pLH}(G2) + T_{pLH}(G1) = 8 \text{ ns} + 10 \text{ ns} + 14 \text{ ns} + 9 \text{ ns} + 8 \text{ ns} = 49 \text{ ns}$$

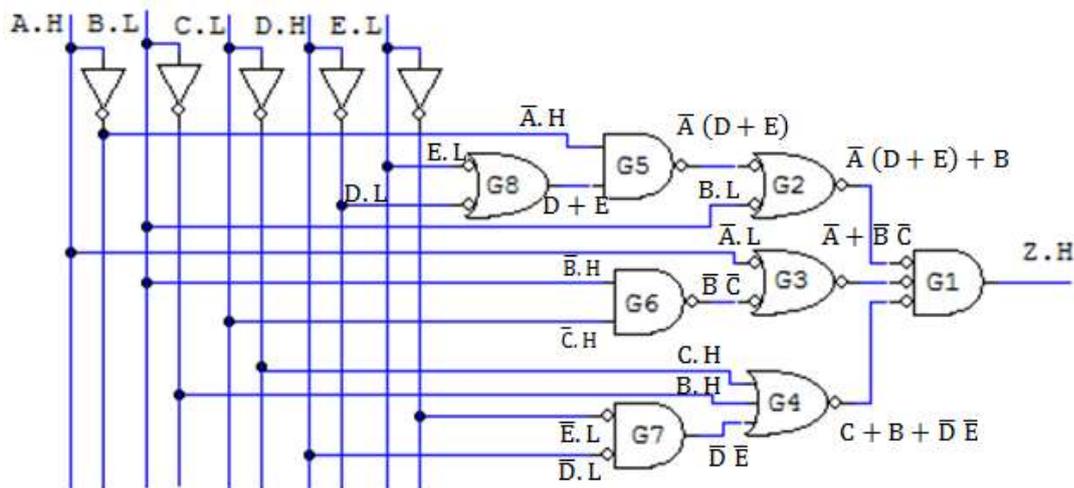
El tiempo de propagación máximo es 53 ns.

b)

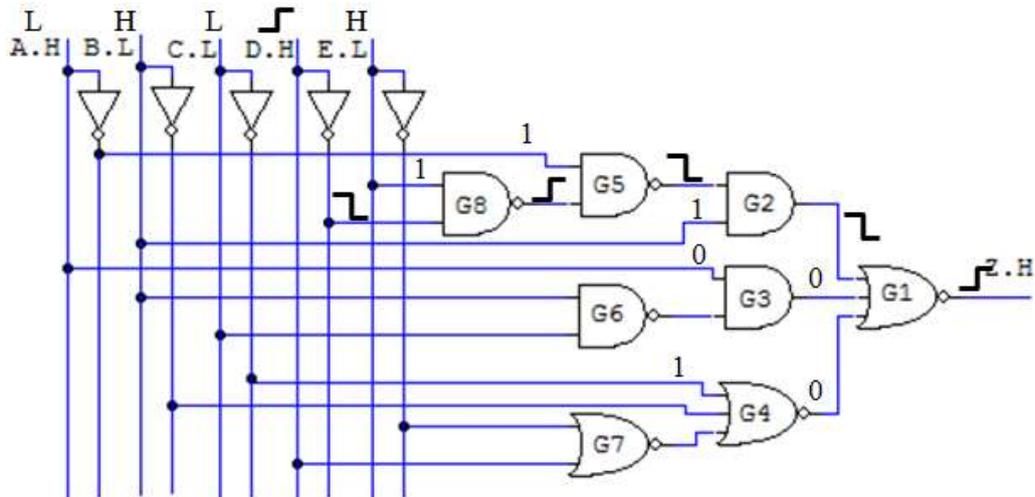


	<u>t_{plh}</u> :	<u>t_{phl}</u> :
NOT	10 ns	9 ns
NAND	14 ns	14 ns
NOR2	12 ns	10 ns
NOR3	15 ns	17 ns
AND	15 ns	20 ns

Como se tiene Z.H ajusto las puertas para que la puerta de salida no tenga burbuja de inversión. Se obtiene $Z.H = [\bar{A} (D + E) + B] (\bar{A} + \bar{B} \bar{C}) (C + B + \bar{D} \bar{E})$.



Para obtener el tiempo de propagación el camino crítico empieza en D a través de la puerta NOT, G8, D5, G2 y G1. Para activar el camino hay un pulso en D, y A = C = L y B = E = H.



Los tiempos de propagación de D a Z se miden según los tiempos de propagación de las puertas y las transiciones HL o LH de la salida de las puertas. Del tiempo HL al tiempo HL hay que cambiar las transiciones de cada puerta.

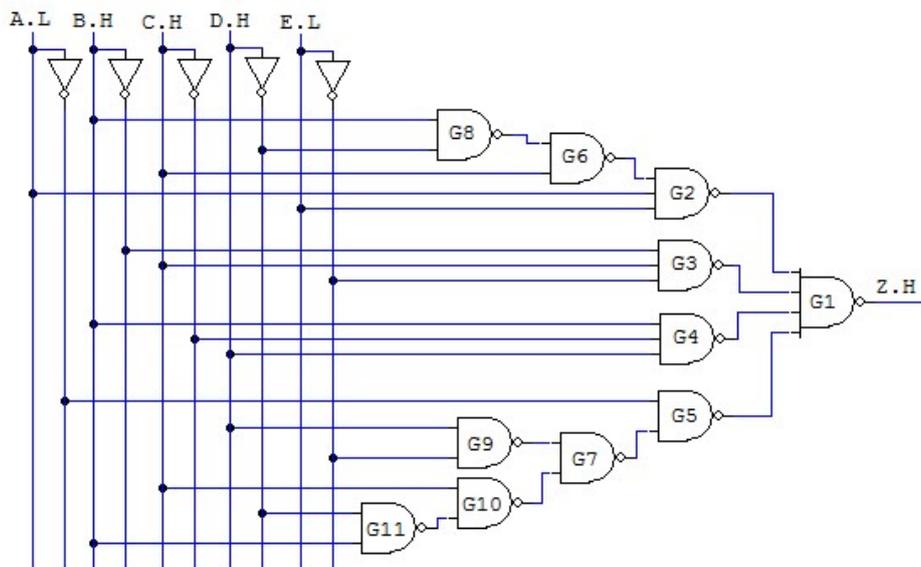
$$\begin{aligned}
 T_{PLH}(D-Z) &= T_{PHL}(\text{NOT}) + T_{PLH}(G8) + T_{PHL}(G5) + T_{PHL}(G2) + T_{PLH}(G1) = \\
 &= 9 \text{ ns} + 14 \text{ ns} + 14 \text{ ns} + 20 \text{ ns} + 15 \text{ ns} = 72 \text{ ns}
 \end{aligned}$$

$$\begin{aligned}
 T_{PLH}(D-Z) &= T_{PLH}(\text{NOT}) + T_{PHL}(G8) + T_{PLH}(G5) + T_{PLH}(G2) + T_{PHL}(G1) = \\
 &= 10 \text{ ns} + 14 \text{ ns} + 14 \text{ ns} + 15 \text{ ns} + 17 \text{ ns} = 70 \text{ ns}
 \end{aligned}$$

El tiempo de propagación máximo es 72 ns.

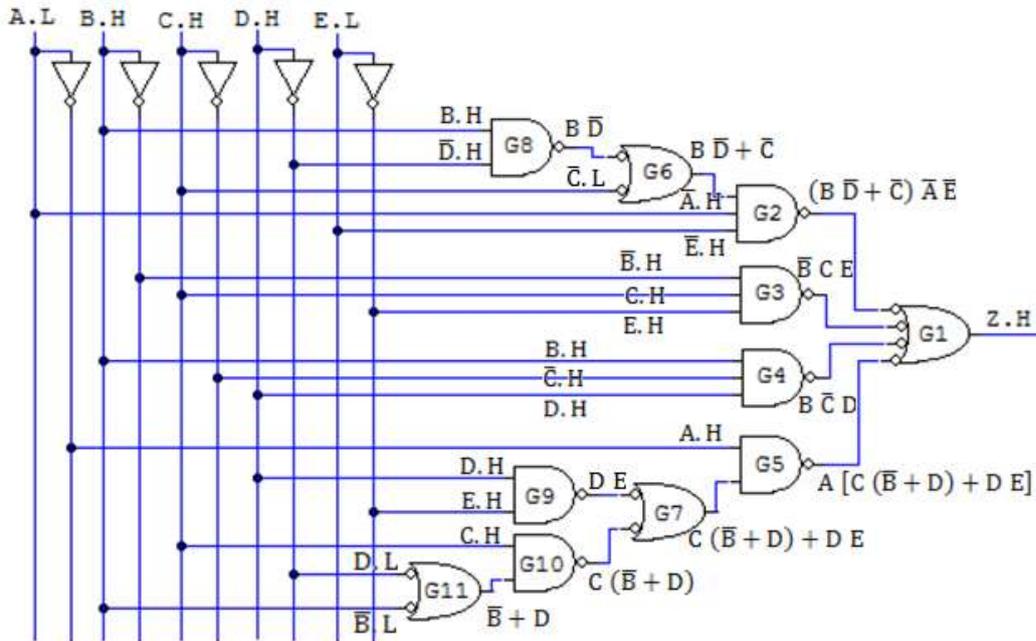
Página 6. Encontrar la expresión lógica factorizada correspondiente a los siguientes circuitos digitales. Suponiendo que todas las puertas tienen el mismo tiempo de propagación t_p , encontrar el camino crítico y los valores en las entradas que permiten su activación.

a)

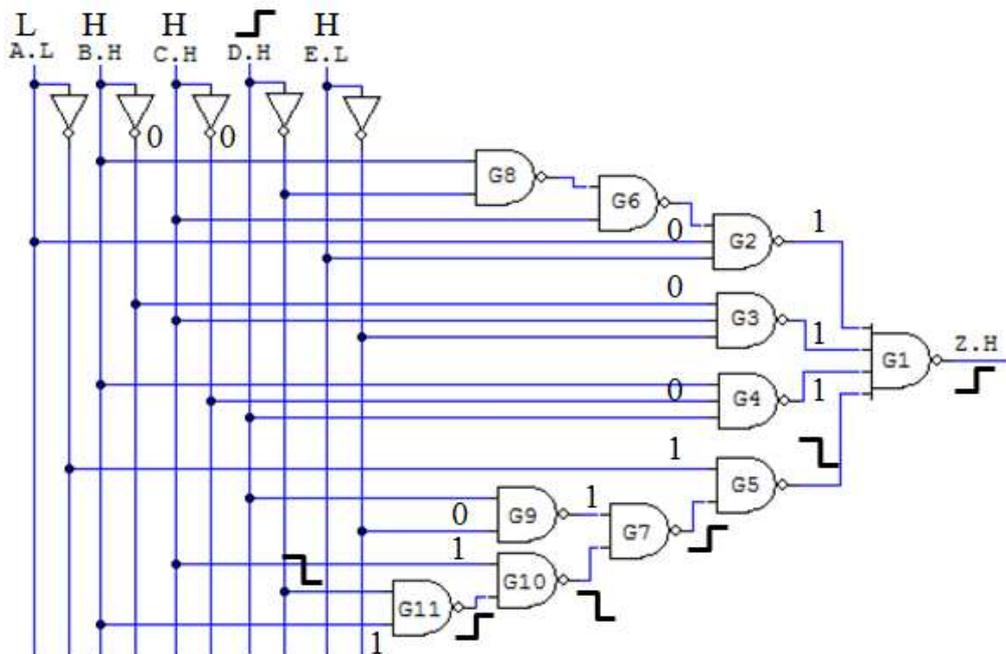


Sustituyo las puertas para obtener una forma factorizada. Como la salida es .H, empiezo sustituyendo la puerta de salida. Obtengo Z.

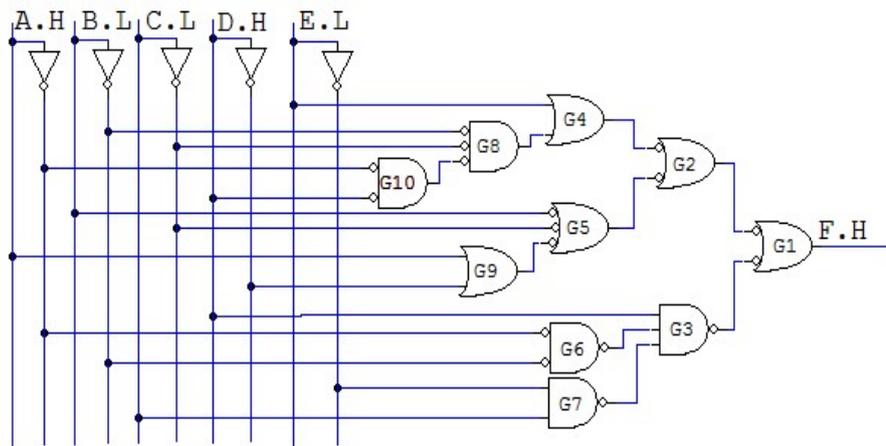
$$Z.H = (B \bar{D} + \bar{C}) \bar{A} \bar{E} + \bar{B} C E + B \bar{C} D + A [C (\bar{B} + D) + D E]$$



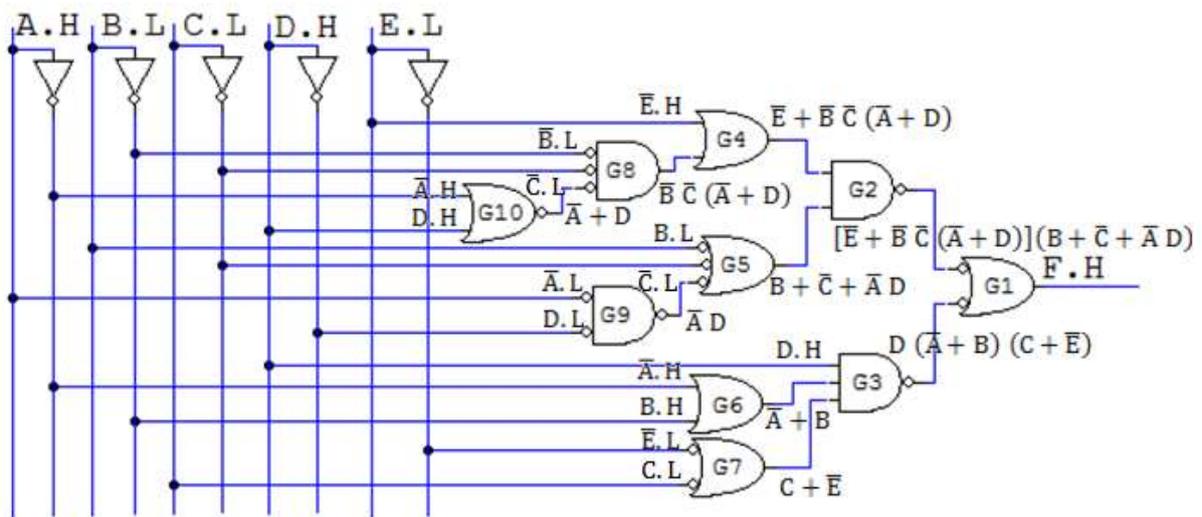
El camino crítico parte de D hasta Z pasando por una puerta NOT y las puertas G11, G10, G7, G5 y G1. Son 6 puertas, luego $T_p = 6 T_p(\text{NAND})$. Al ser puertas NAND, las otras entradas de las puertas del camino crítico deben estar a 1 lógico. Para activar en ese camino en G11 un 1 en la otra entrada de la puerta implica que B es H; por G10, C está H; por G7, una de las dos entradas de G9 está a 0, D no puede ser porque es la señal pulsada, luego debe ser la otra, lo que implica que E es H (por el inversor); por G5, A es L (por el inversor). Además, por la puerta G1, las salidas de G2, G3 y G4 tienen que estar a 1; A a L fuerza G2 a 1, B a H con el inversor fuerza G3 a 1, y C a H con el inversor fuerza G4 a 1. Luego, $A = L$ y $B = C = E = H$. En D se introduce las transiciones que activan el camino crítico.



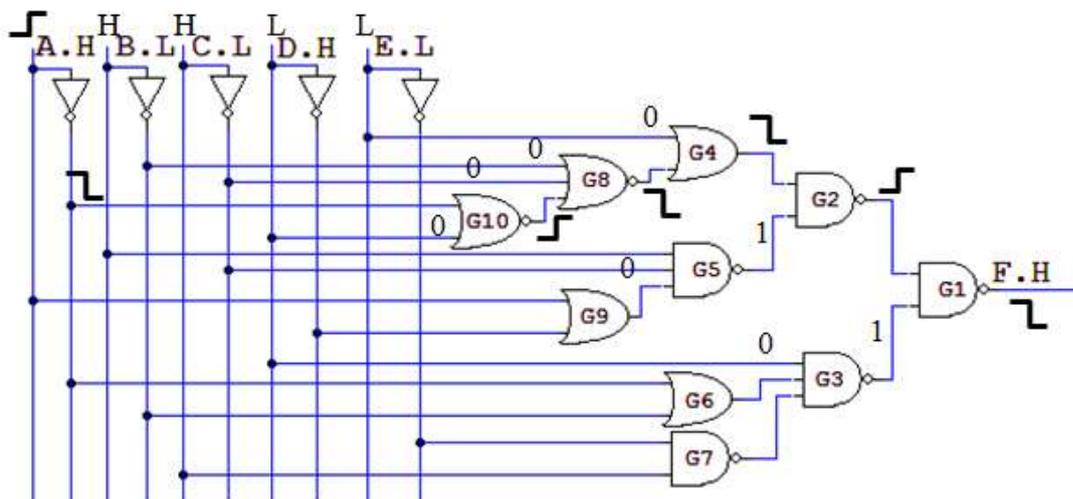
b)



Teniendo en cuenta que la salida es .H, hago las conversiones de puertas que permiten que los nudos internos tengan la misma polaridad entrad la puerta fuente del nudo y las entradas de las puertas conectadas al nudo. Modifico G2, G6, G7, G9 y G10. Una vez hecho, encuentro la función $F.H = [\bar{E} + \bar{B} \bar{C} (\bar{A} + D)](B + \bar{C} + \bar{A} D) + D (\bar{A} + B) (C + \bar{E})$.

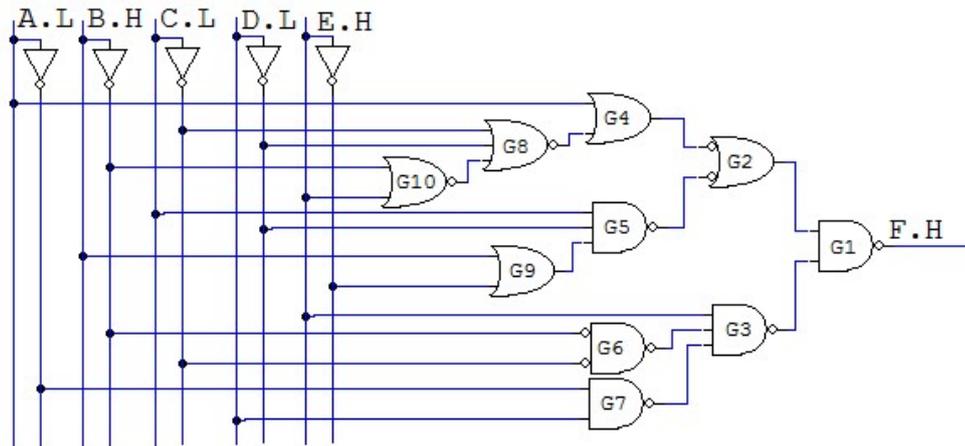


El camino crítico del circuito va desde A a F, pasando por una puerta NOT y las puertas G10, G8, G4, G2 y G1. Luego el $T_p = 6 t_p$, donde t_p es el tiempo de propagación de una puerta.



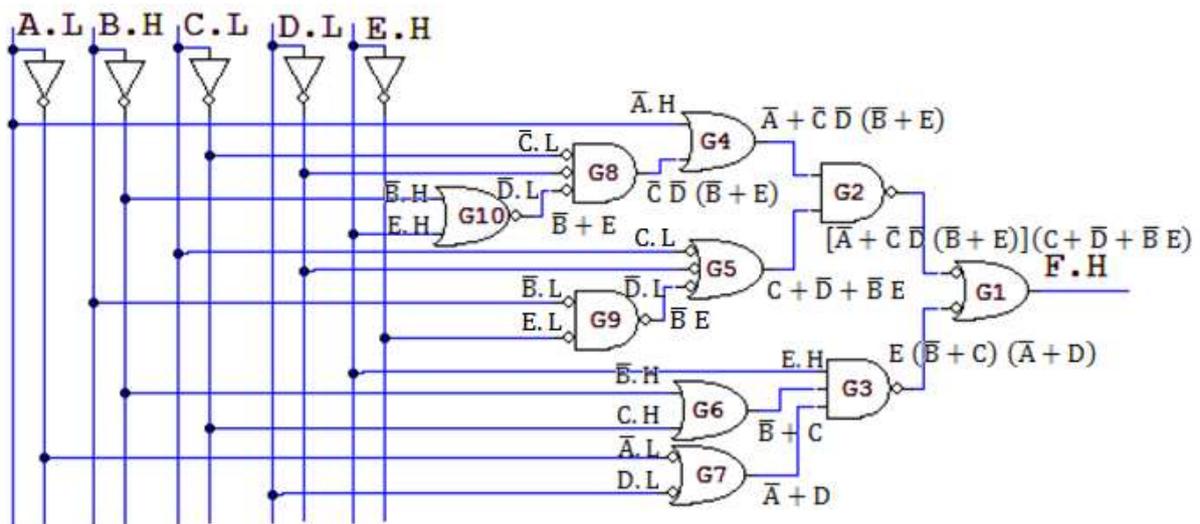
Sitúo todas las puertas en su forma positiva, para visualizar mejor los valores no controlantes en las otras entradas de las puertas del camino crítico. En la entrada A se aplica la transición; fijo D a L por la puerta 10 (NOR), fijo B y C a H (con los inversores) por la puerta 8 (NOR) y fijo E a L por la puerta 4. Además, C a H, fija la salida de su inversor a 0 y la salida de G5 a 1, y D a L fija la salida de G3 a 1, con lo que todo el camino queda sensibilizado.

c)

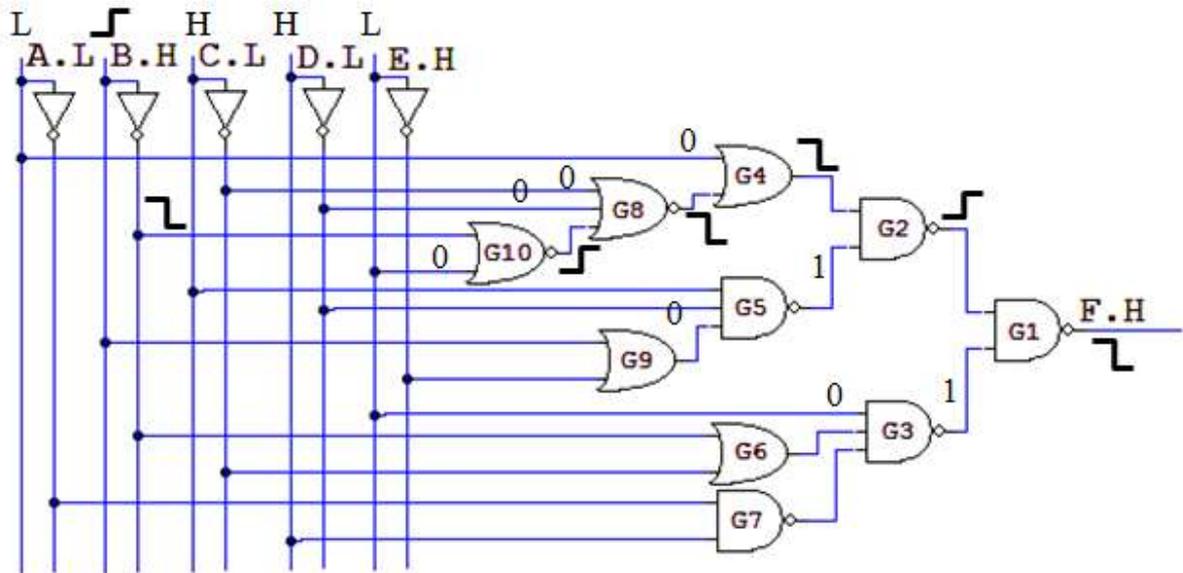


Teniendo en cuenta que la salida es .H, hago las conversiones de puertas que permiten que los nudos internos tengan la misma polaridad entre la puerta fuente del nudo, y las entradas de las puertas conectadas al nudo. Modifico G1, G2, G5, G6, G7, G8 y G9. El circuito resultante es similar al del apartado b), donde solo cambian las conexiones a las entradas y la polaridad de estas. Siguiendo el mismo procedimiento que en b) encuentro la función F.

$$F.H = [A + \bar{C} \bar{D} (\bar{B} + E)](C + \bar{D} + \bar{B} E) + E (\bar{B} + C) (\bar{A} + D)$$



El camino crítico del circuito va desde B a F, pasando por una puerta NOT y las puertas G10, G8, G4, G2 y G1, como en el apartado b). Luego el $T_p = 6 t_p$, donde t_p es el tiempo de propagación de una puerta. Como en el apartado b), sitúo todas las puertas en su forma positiva. En la entrada B se aplica la transición; fijo E a L por la puerta 10 (NOR), fijo C y D a H (con los inversores) por la puerta 8 (NOR) y fijo A a L por la puerta 4. Además, D a H, fija la salida de su inversor a 0 y la salida de G5 a 1, y E a L fija la salida de G3 a 1, con lo que todo el camino queda sensibilizado.



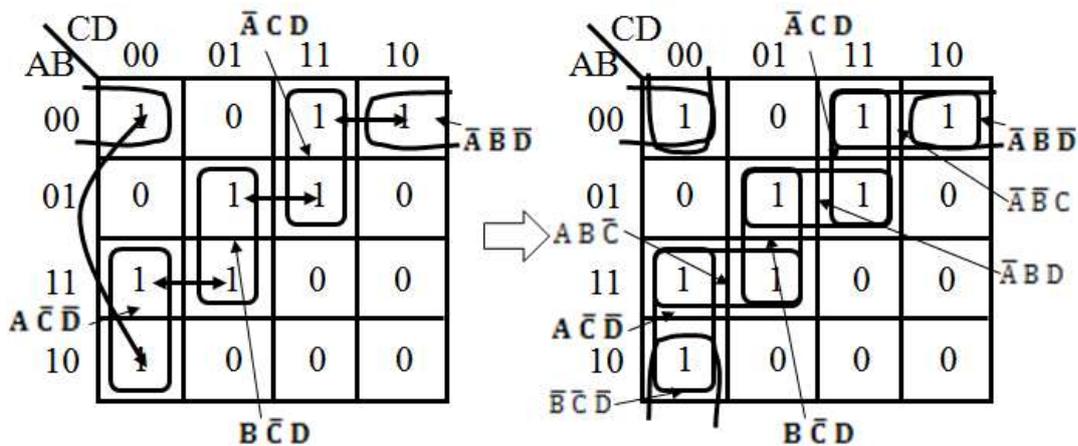
Página 7. Obtener una implementación libre de peligros de las siguientes funciones lógicas

Se debe encontrar la función lógica en dos niveles SOP, por ejemplo, en las que todas las casillas a 1, adyacentes del mapa, estén cubiertas por, al menos implicante primo común. Se puede partir de una función SOP mínima, y luego añadir los implicantes primos para que la función esté libre de peligros lógicos. Puede que incluso alguno de los implicantes primos de la función SOP mínima original se pueda eliminar.

a) $F(A, B, C, D) = \sum(0, 2, 3, 5, 7, 8, 12, 13)$

Esta función aparece en el problema 2e del tema IIc. La función lógica SOP mínima encontrada es $F(A, B, C, D) = \bar{A}\bar{B}\bar{D} + A\bar{C}\bar{D} + B\bar{C}D + \bar{A}CD$. La función original tiene 4 peligros lógicos: pares de casillas (0, 8), (2, 3), (5, 7), (12, 13). Para cubrir los peligros añado cuatro implicantes primos.

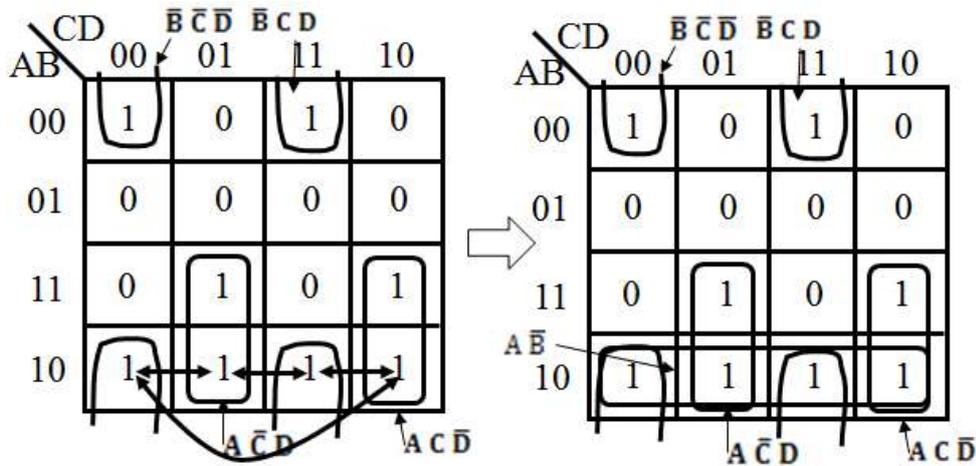
$$F = \bar{A}\bar{B}\bar{D} + A\bar{C}\bar{D} + B\bar{C}D + \bar{A}CD + \bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}C + \bar{A}BD + ABC$$



b) $F(A, B, C, D) = \prod(1, 2, 4, 5, 6, 7, 12, 15)$

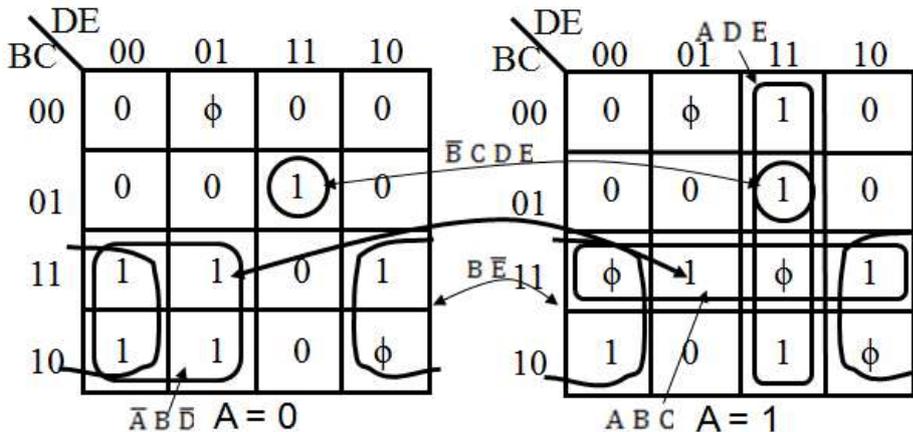
Esta función también aparece en el problema 2 del tema IIa como apartado f). La función SOP mínima es $F(A, B, C, D) = \bar{B}\bar{C}\bar{D} + A\bar{C}D + \bar{B}CD + ACD$. Esta función presenta peligros

lógicos en (8, 9), (9, 11), (10, 11) y (8, 10). Todos estos peligros se pueden eliminar añadiendo un implicante primo a la función $F = \bar{B} \bar{C} \bar{D} + A \bar{C} D + \bar{B} C D + A C \bar{D} + A \bar{B}$

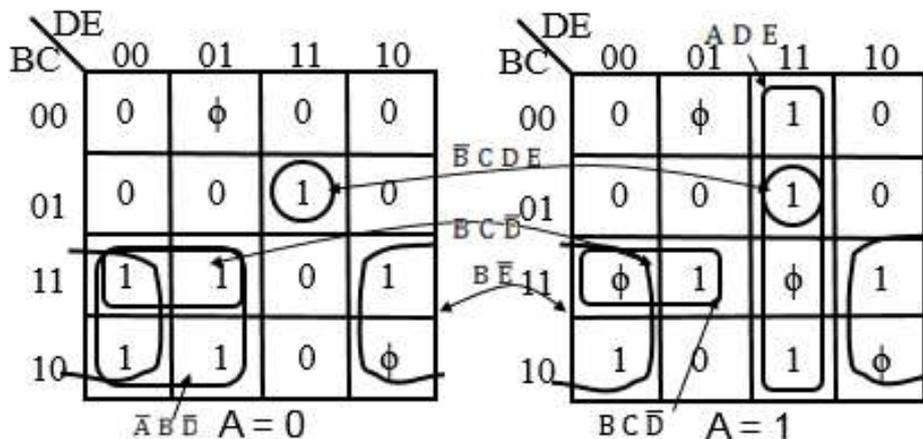


c) $F(A, B, C, D, E) = \sum(7,8,9,12,13,14,19,23,24,27,29,30) + \sum_{\phi}(1,10,17,26,28,31)$

Esta función se ha minimizado en el apartado c) del problema 6 del tema IIa. La función lógica mínima calculada era $F(A, B, C, D, E) = \bar{B} C D E + B \bar{E} + \bar{A} B \bar{D} + A D E + A B C$. Solo hay que comprobar las adyacencias entre casillas con 1s, entre las casillas con “don’t cares”, o entre 1 y “don’t cares” no hace falta comprobar el posible peligro.



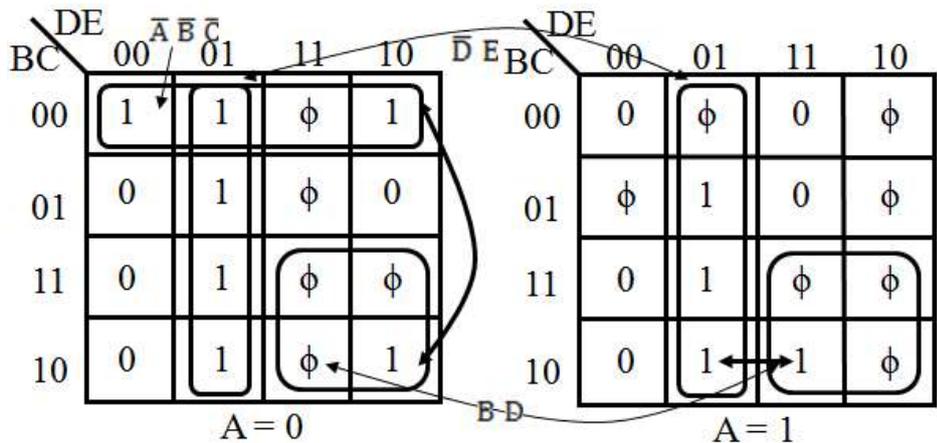
En la función minimizada hay un posible peligro no cubierto entre las casillas 13 y 29. Este peligro se puede eliminar usando el término $B C \bar{D}$. Al añadir ese término producto extra, sobra el término producto $A B C$, que se puede eliminar de la función y se siguen cubriendo todos los 1s, y sin crear nuevos peligros.



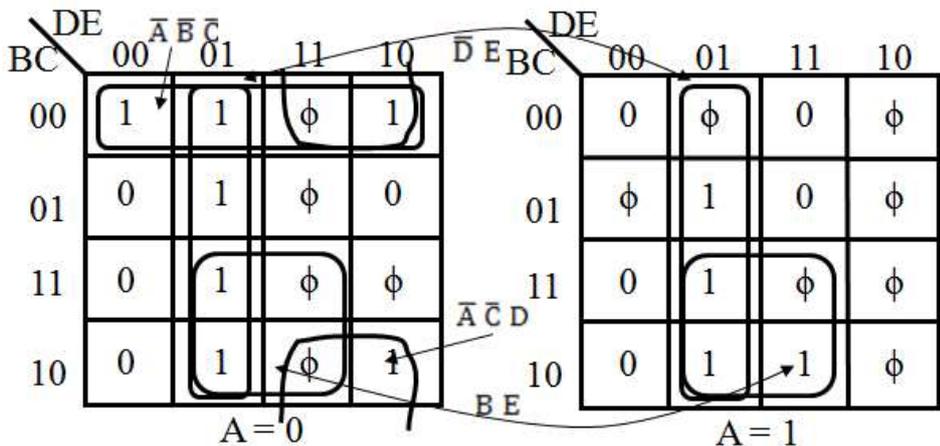
Cuando se pueden usar varios implicantes primos del mismo tamaño, suele ser mejor usar los que cubren más 1s que los que cubren más "don't cares". La función resultante es mínima y sin peligros. $F = \bar{B} C D E + B \bar{E} + \bar{A} B \bar{D} + A D E + B C \bar{D}$

d) $F(A,B,C,D,E) = \sum(0,1,2,5,9,10,13,21,25,27,29) + \sum_{\phi}(3,7,11,14,15,17,18,20,22,26,30,31)$

Esta función no ha aparecido en problemas anteriores. Encuentro una función mínima y luego elimino los peligros lógicos. La función mínima es $F = \bar{D} E + B D + \bar{A} \bar{B} \bar{C}$, y hay dos posibles peligros lógicos entre las casillas (2, 10) y (25, 27).



Para eliminarlos se pueden añadir dos implicantes primos más $B E$ y $\bar{A} \bar{C} D$. Pero, al hacerlo, el término $B D$ sobra, y se puede eliminar. La función libre de peligros más pequeña es: $F = \bar{D} E + \bar{A} \bar{B} \bar{C} + B E + \bar{A} \bar{C} D$.



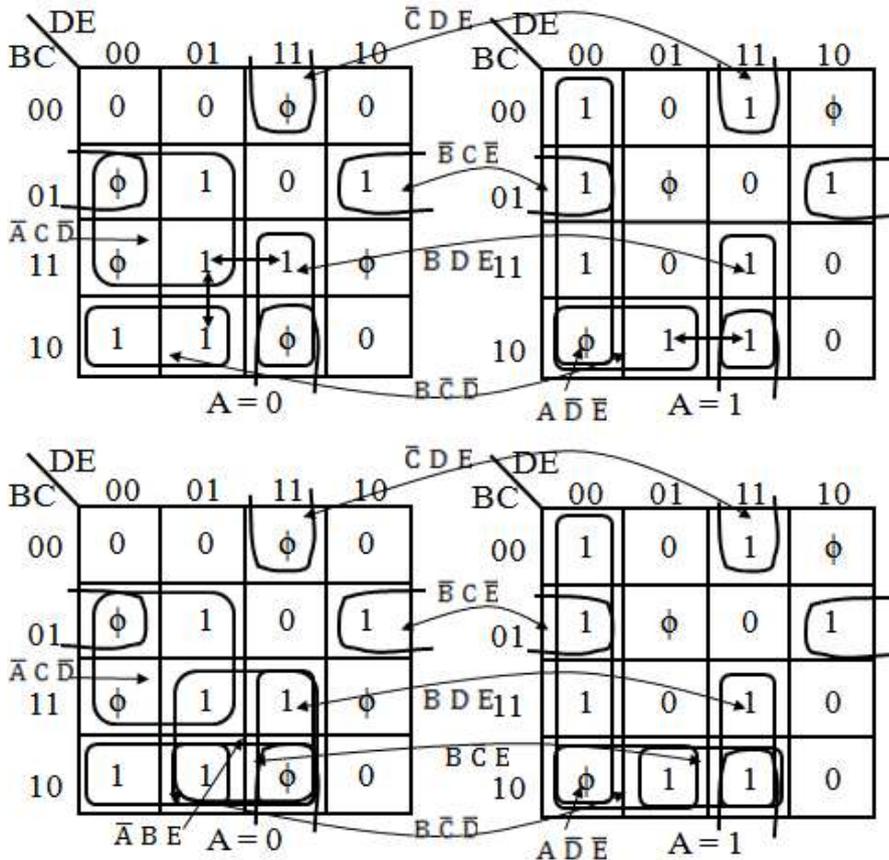
e) $F(A,B,C,D,E) = \sum(5,6,8,9,13,15,16,19,20,22,25,27,28,31) + \sum_{\phi}(3,4,11,12,14,18,21,24)$

Se puede encontrar una función lógica mínima. La casilla 31 tiene implicante primo esencial, y hay mejor implicante para las casillas 6, 19, y ahora también para las casillas 16 y 25. Las dos casillas restantes con 1s (5 y 13) pueden ser cubiertas por un único implicante primo. La función mínima queda:

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

Esta función tiene peligros en las casillas (9, 13), (12, 13) y (25, 27). Para eliminarlos hay que añadir dos términos producto más: $\bar{A} B E$, y $B \bar{C} E$. No se puede eliminar ningún término producto de la función mínima.

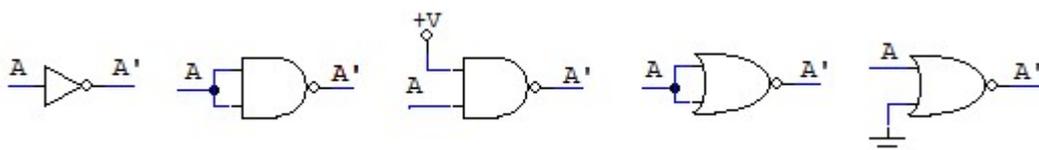
$$F(A, B, C, D, E) = B D E + \bar{B} C \bar{E} + \bar{C} D E + A \bar{D} \bar{E} + B \bar{C} \bar{D} + \bar{A} C \bar{D} + \bar{A} B E + B \bar{C} E$$

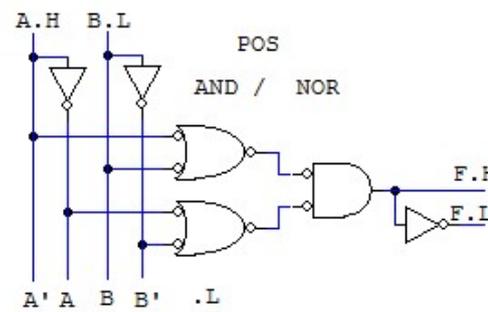
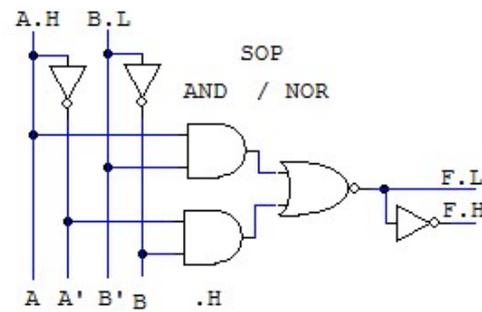
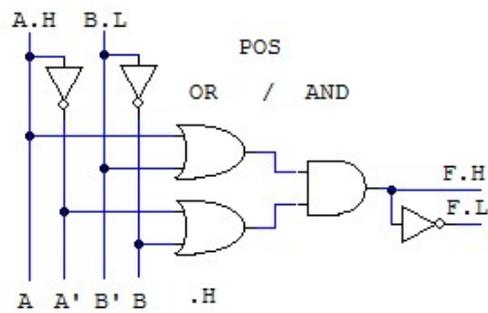
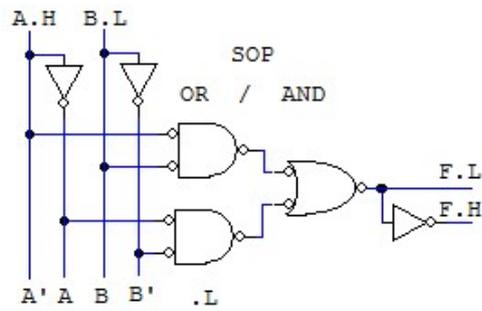
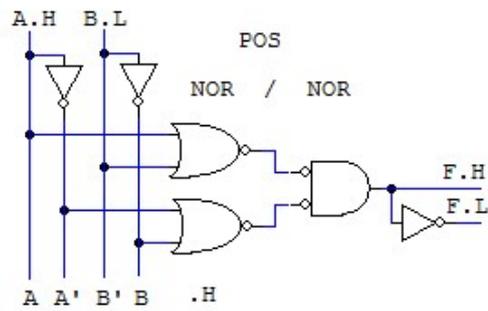
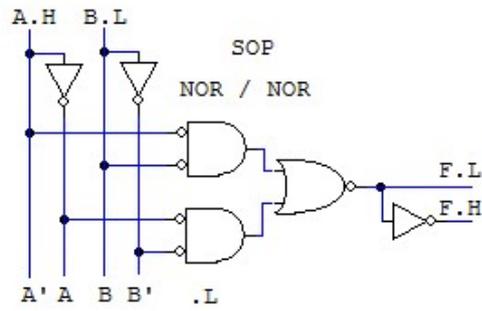
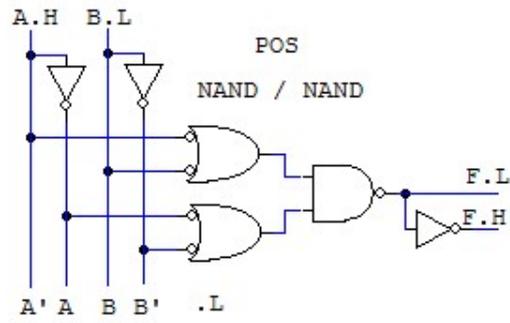
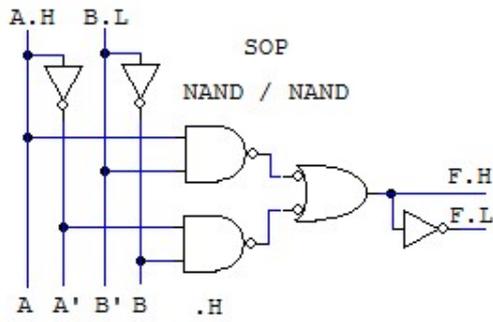


Página 8. Realizar una implementación en dos niveles de las siguientes funciones, utilizando en notación (Primer Nivel/Segundo Nivel) las puertas lógicas indicadas, y puertas lógicas NOT solo en las implementaciones que sea absolutamente necesario: a) NAND/NAND, b) NOR/NOR; c) OR/AND; d) AND/NOR.

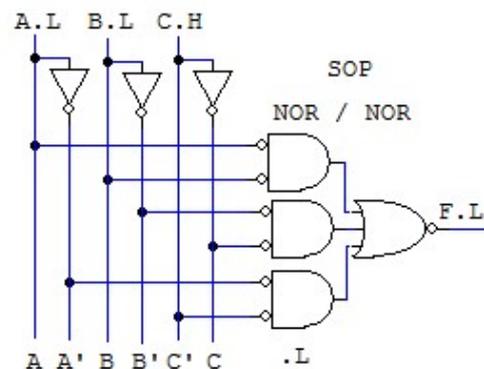
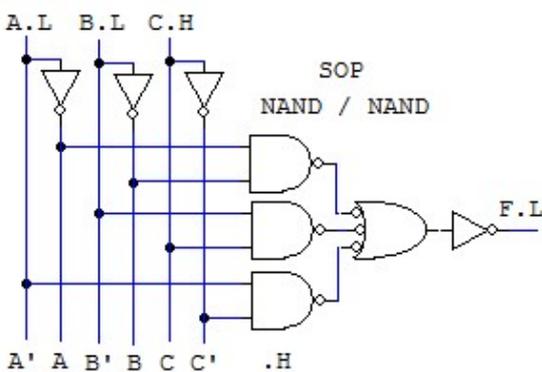
Con estas configuraciones primer nivel/segundo nivel, se pueden implementar funciones lógicas SOP y POS, situando las puertas de forma que coincida la polaridad entra las salidas del primer nivel y las entradas del segundo; se adaptan las polaridades de las entradas a las entradas del primer nivel, y se adapta la polaridad de la salida a la polaridad de salida del segundo nivel usando o sin usar un inversor. Los términos productos o suma los dibujos en orden desde arriba (el primero) hacia abajo (el último).

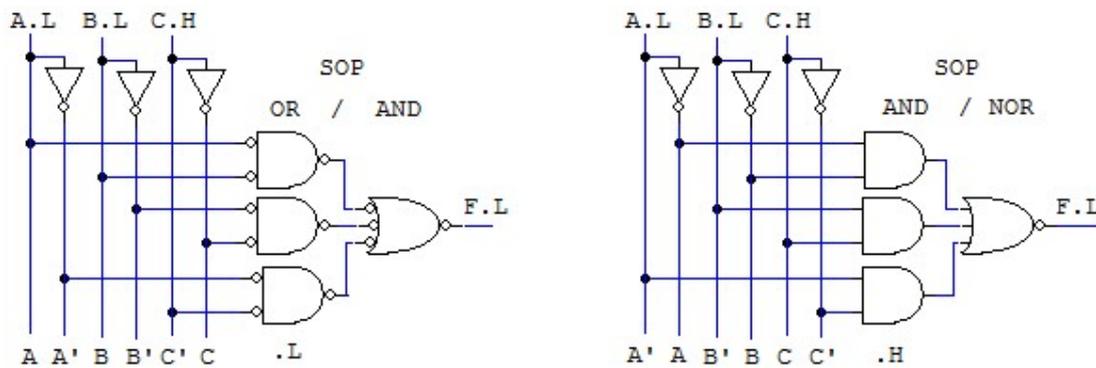
Aunque el enunciado pide que no se usen puertas NOT, en los diseños se usan por comodidad. Con puertas NAND o NOR de dos entradas se pueden hacer inversores. Solo deberían usar NOT en la configuración OR/AND. En los diseños elimino los inversores que quedan sin conectar. En las figuras la complementación la indico con '.



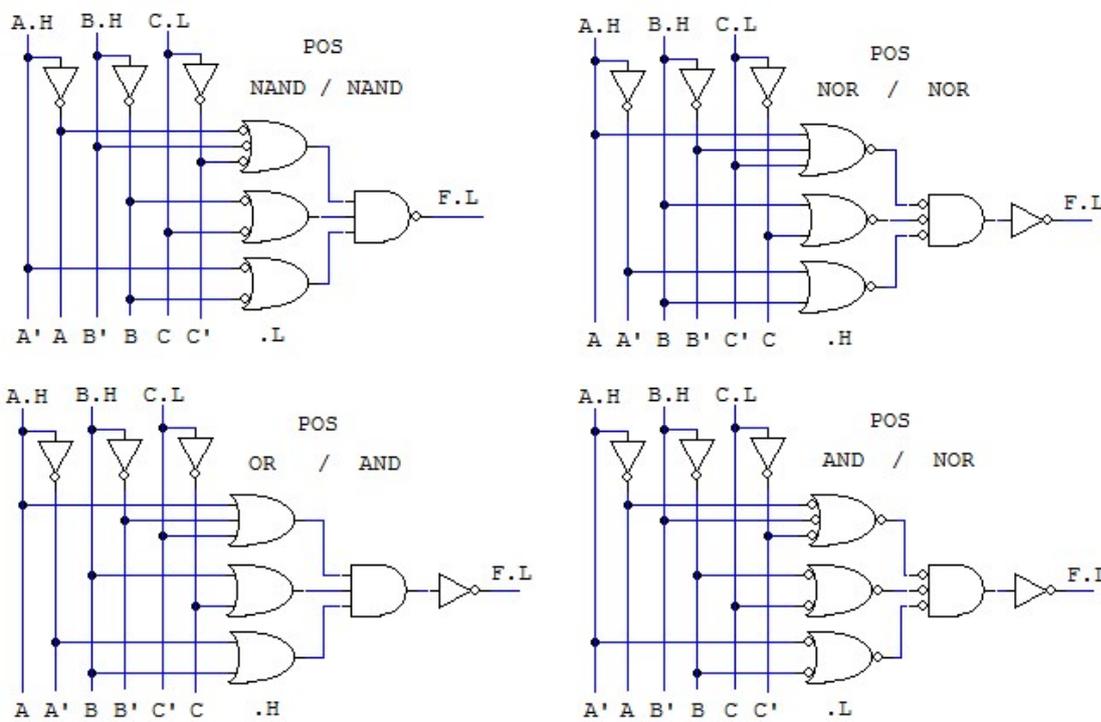


a) $F.L = [F(A, B, C)].L = [A B + \bar{B} C + \bar{A} \bar{C}].L$, con A.L, B.L y C.H.



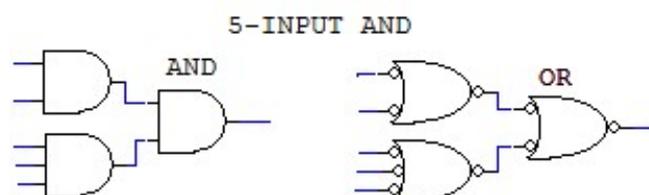


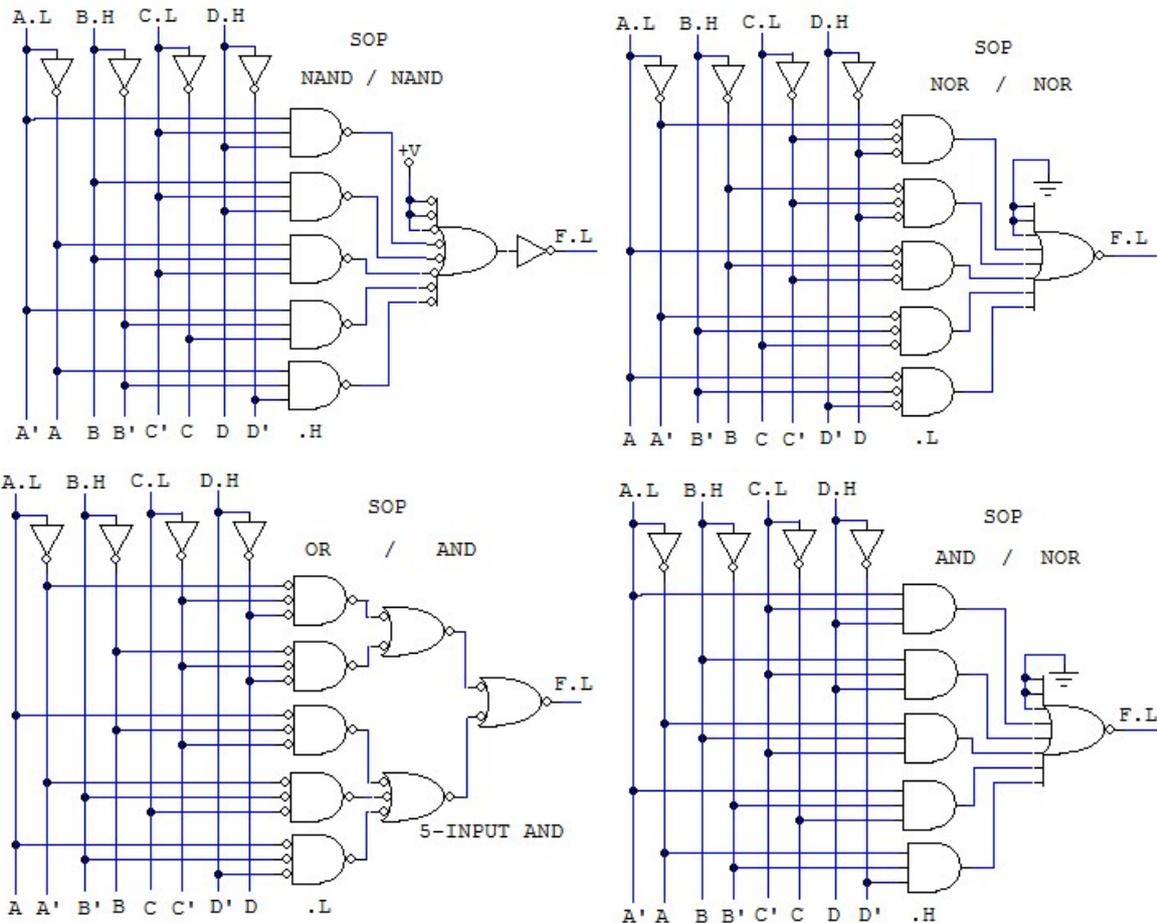
b) $F.L = [F(A, B, C)].L = [(A + \bar{B} + \bar{C})(B + C)(\bar{A} + B)].L$, con A.H, B.H y C.L.



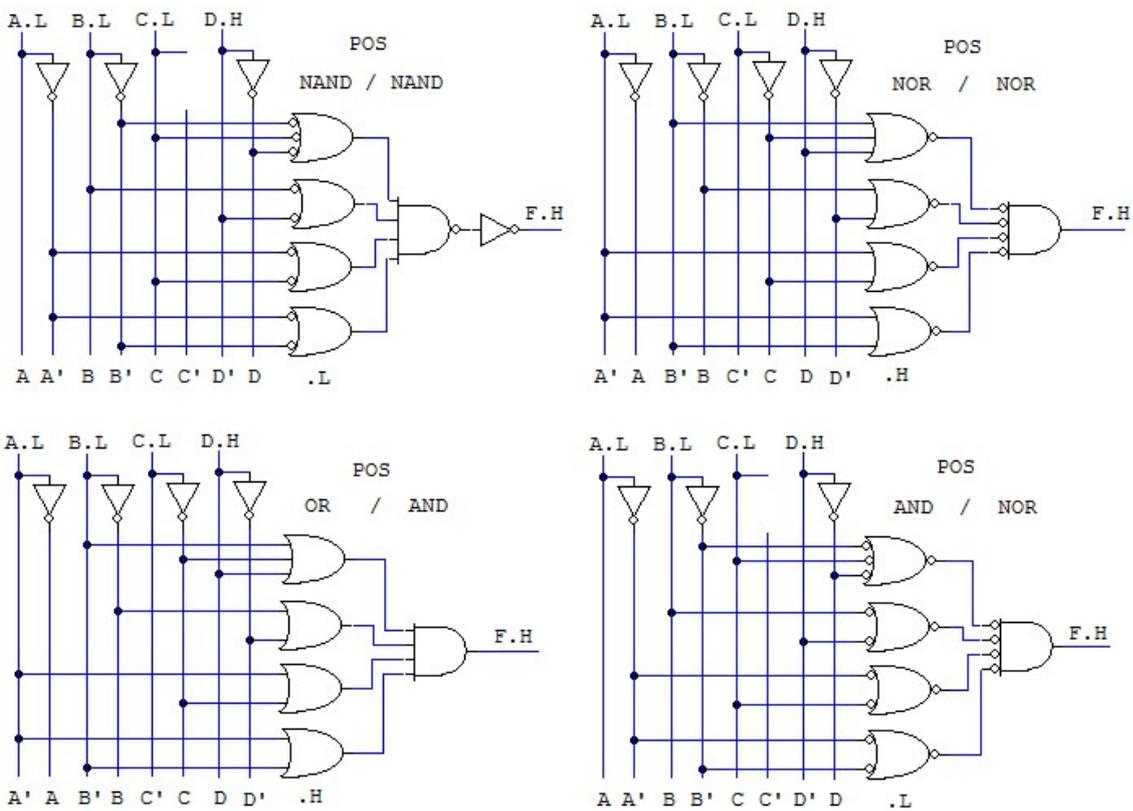
c) $F.L = [F(A, B, C, D)].L = [\bar{A}\bar{C}D + B\bar{C}D + AB\bar{C} + \bar{A}\bar{B}C + A\bar{B}\bar{D}].L$, con A.L, B.H, C.L y D.H

En esta función aparece una AND lógica de 5 entradas, que se implementará con NAND, NOR y AND. El editor de esquemáticos no contiene puertas con ese número de entradas, así que uso puertas NAND de 8 entradas con las entradas no usadas a 1, puertas NOR de 8 entradas con las entradas no usadas a 0, puertas AND de 5 entradas construidas como un árbol de ANDs, por ejemplo, de esta forma:

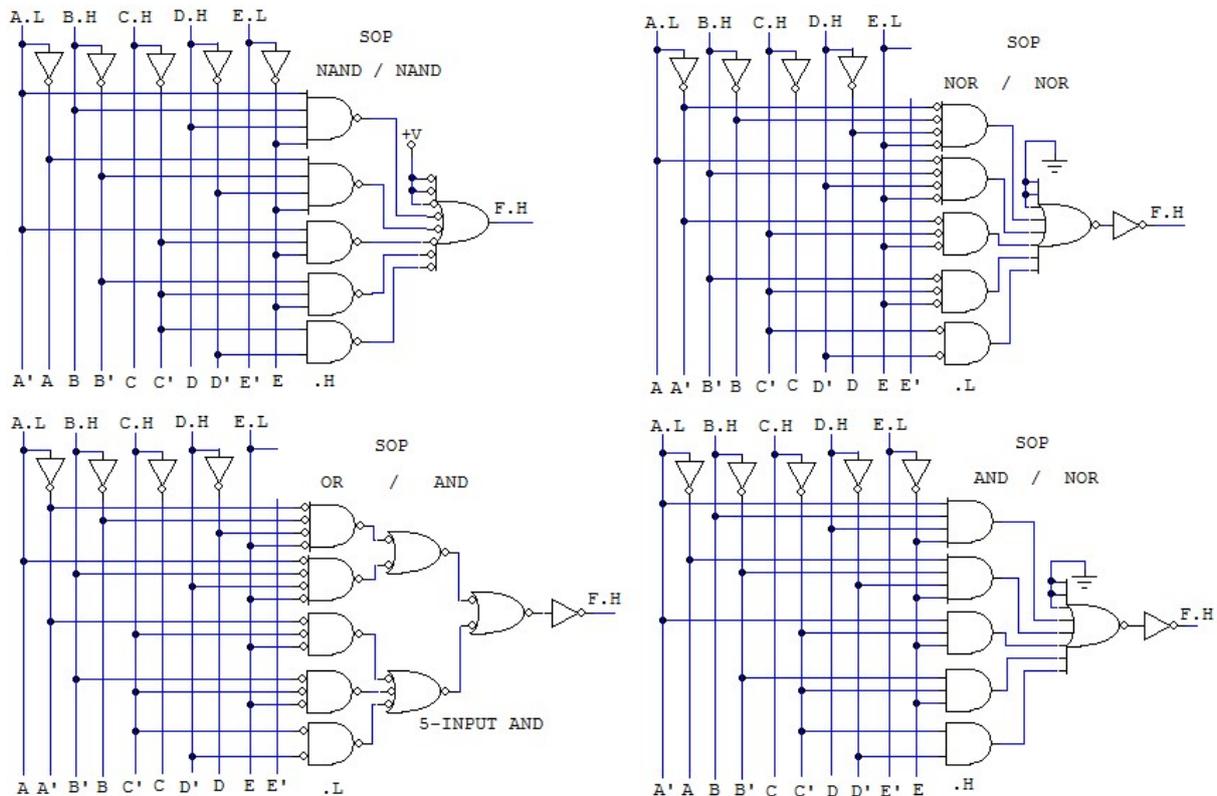




d) $F.H = [F(A, B, C, D)].H = [(\bar{B} + C + D)(B + \bar{D})(\bar{A} + C)(\bar{A} + \bar{B})].H$,
 con A.L, B.L, C.L y D.H.



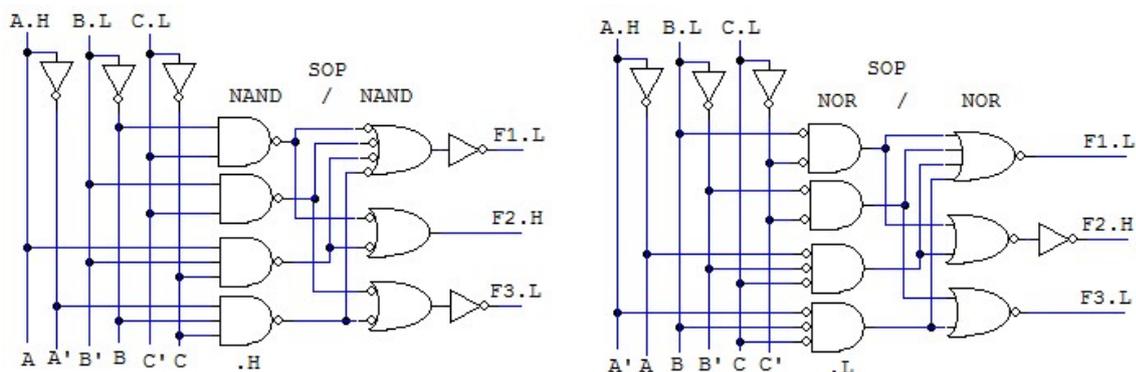
e) $F.H = [F(A,B,C,D,E)].H = [\bar{A} B D E + A \bar{B} \bar{D} E + \bar{A} \bar{C} E + \bar{B} \bar{C} E + \bar{C} \bar{D}].H,$
 con A.L, B.H, C.H, D.H y E.L

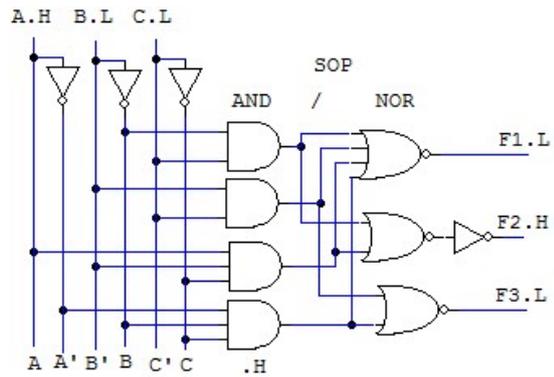
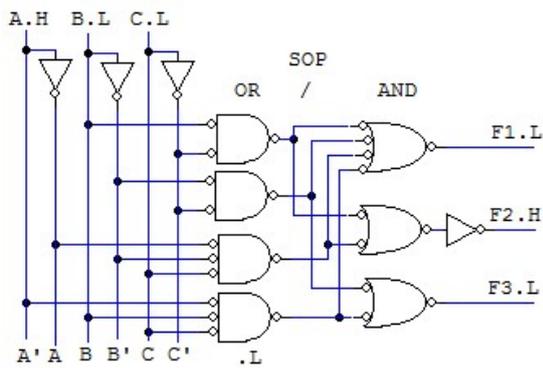


Página 9. Realizar una implementación en dos niveles de las siguientes funciones, utilizando en notación (Primer Nivel/Segundo Nivel) las puertas lógicas indicadas, y puertas lógicas NOT sólo en las implementaciones que sea absolutamente necesario: a) NAND/NAND, b) NOR/NOR; c) OR/AND; d) AND/NOR.

Este problema es similar al anterior con la diferencia de que hay varias salidas que comparten términos producto o suma: la salida de una puerta del primer nivel se conecta a entradas de varias puertas del segundo nivel. Sitúo los términos producto o suma, en el orden en que aparecen en las funciones, primero los de F1, y luego los nuevos de F2 y de F3.

- a) $F1(A, B, C) = B \bar{C} + \bar{B} \bar{C} + A \bar{B} C + \bar{A} B C$
- $F2(A, B, C) = B \bar{C} + A \bar{B} C$
- $F3(A, B, C) = \bar{B} \bar{C} + \bar{A} B C$
- Para A.H, B.L, C.L, F1.L, F2.H y F3.L



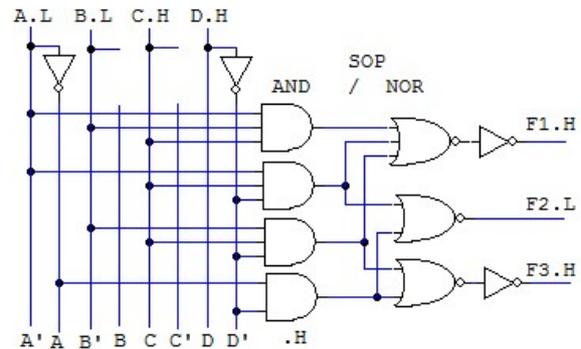
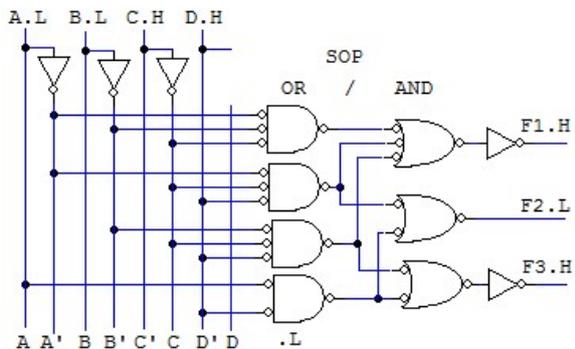
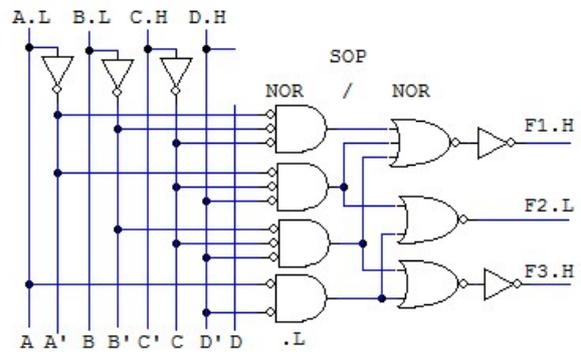
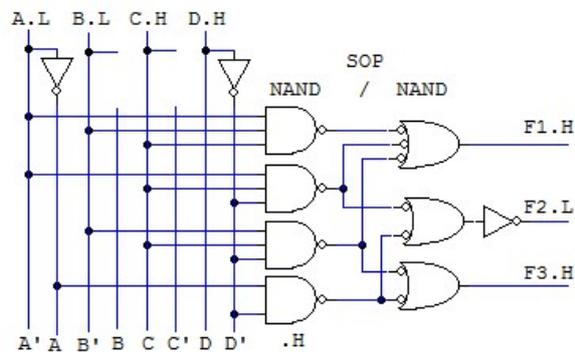


b) $F1(A, B, C, D) = \bar{A}\bar{B}C + \bar{A}C\bar{D} + \bar{B}C\bar{D}$

$F2(A, B, C, D) = \bar{A}C\bar{D} + A\bar{D}$

$F3(A, B, C, D) = \bar{B}C\bar{D} + A\bar{D}$

Para A.L, B.L, C.H, D.H, F1.H, F2.L y F3.H

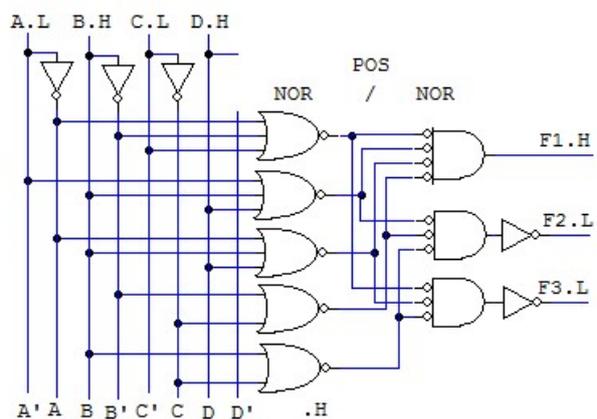
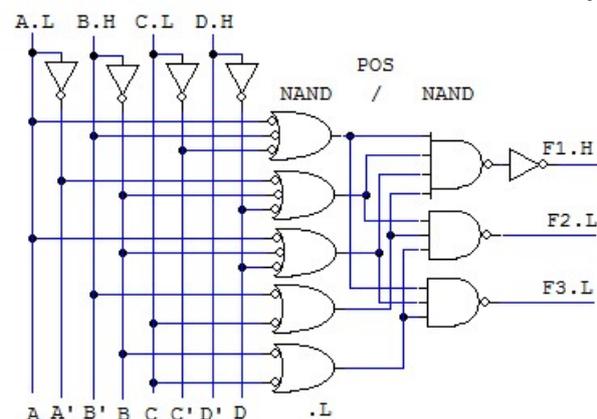


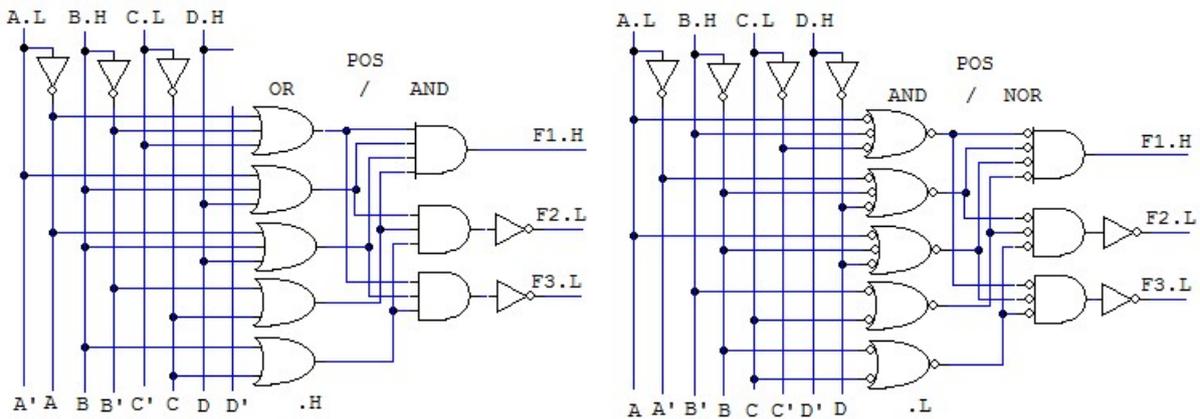
c) $F1(A, B, C, D) = (A + \bar{B} + \bar{C})(\bar{A} + B + D)(A + B + D)(\bar{B} + C)$

$F2(A, B, C, D) = (\bar{A} + B + D)(\bar{B} + C)(B + C)$

$F3(A, B, C, D) = (A + \bar{B} + \bar{C})(A + B + D)(B + C)$

Para A.L, B.H, C.L, D.H, F1.H, F2.L y F3.L

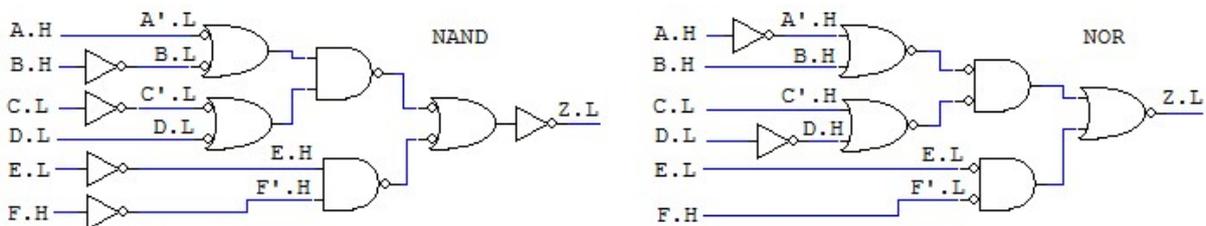




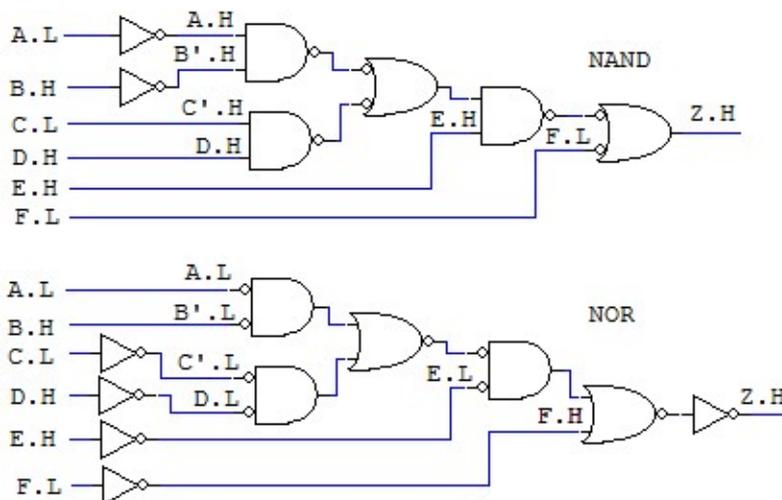
Página 10_1. Diseñar usando solamente puertas NAND y solamente puertas NOR el circuito correspondiente a las siguientes expresiones lógicas factorizadas.

En la diapositiva 62 de teoría se indica cómo se puede implementar una función lógica factorizada únicamente con puertas NAND, o únicamente con puertas NOR. Al situar las puertas NAND como forma AND o como forma OR se generan niveles consecutivos AND/OR/AND/OR..., donde la polaridad de la salida de un nivel coincide con la de las entradas de la puerta del siguiente, por lo que las operaciones AND u OR se realizan correctamente. La polaridad de las entradas del problema se ajusta, usando o no inversores, a las entradas de las puertas lógicas de los últimos niveles, utilizando las relaciones $A.H = \bar{A}.L$, y $A.L = \bar{A}.H$. La polaridad de la salida se ajusta usando un inversor, si no coincide la requerida por el problema con la generada por el circuito. En las figuras uso ' para indicar complementación.

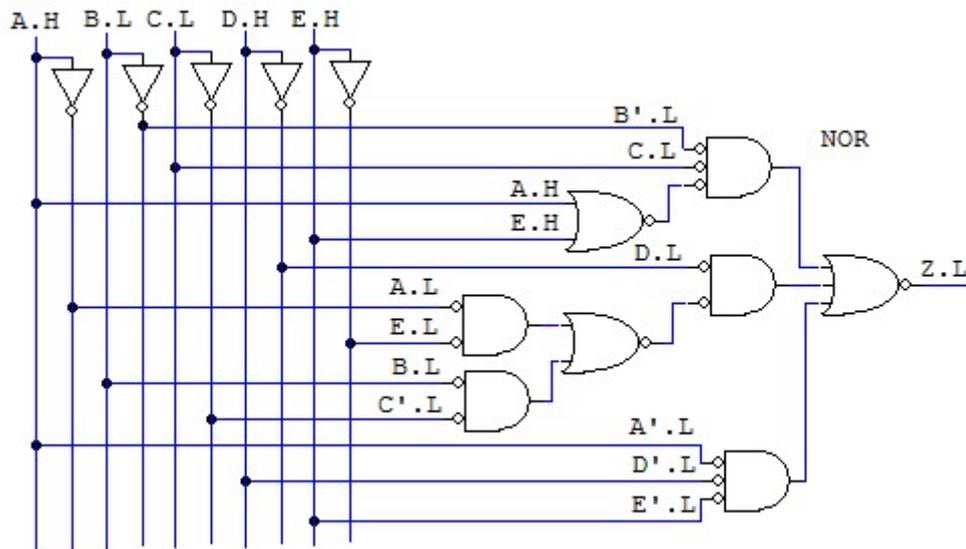
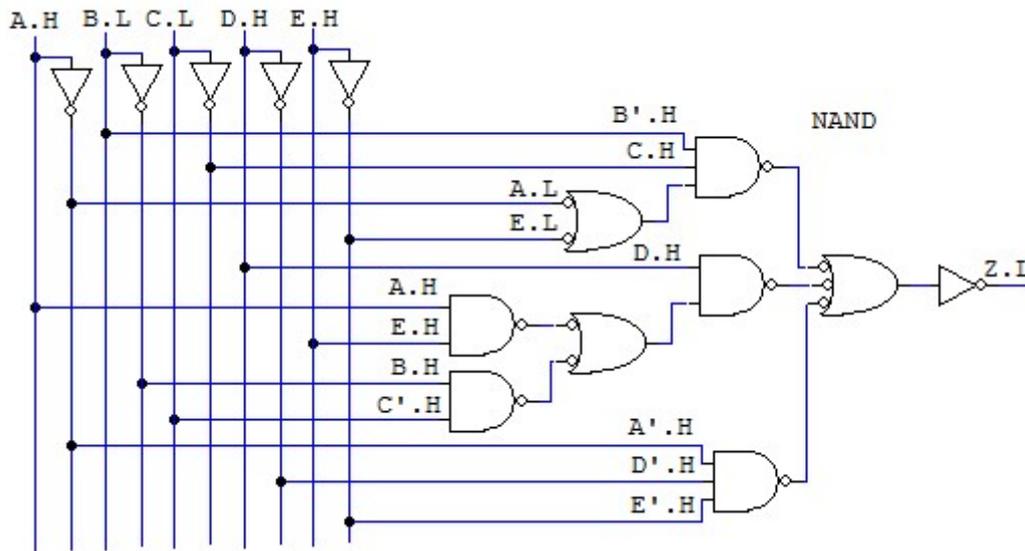
a) $Z.L = [(\bar{A} + B)(\bar{C} + D) + E\bar{F}].L$ para A.H, B.H, C.L, D.L, E.L, F.H



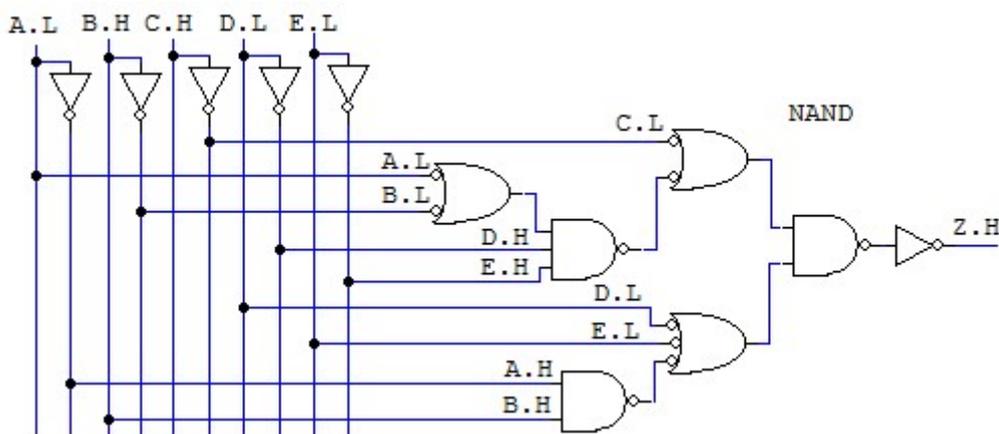
b) $Z.H = [(A\bar{B} + \bar{C}D)E + F]$, para A.L, B.H, C.L, D.H, E.H, F.L

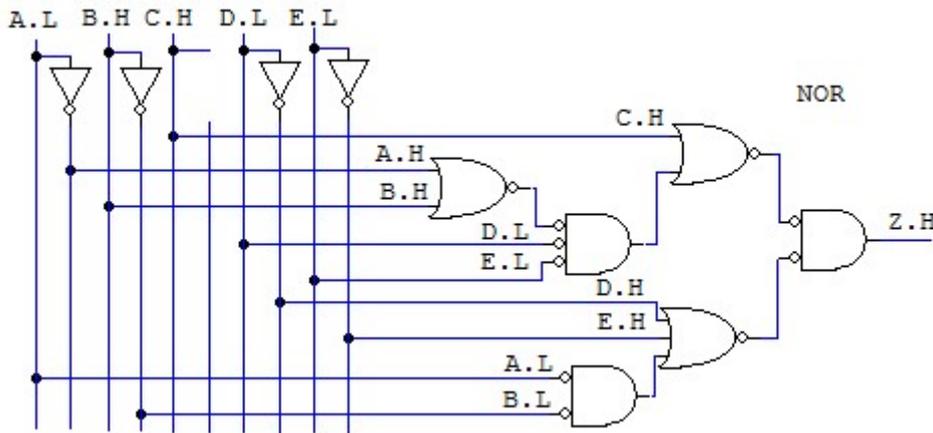


c) $Z.L = \bar{B} C (A + E) + D (A E + B \bar{C}) + \bar{A} \bar{D} \bar{E}$, para A.H, B.L, C.L, D.H, E.H

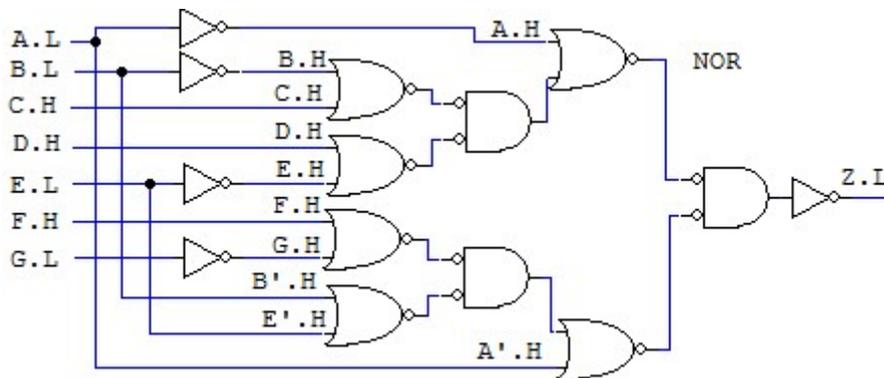
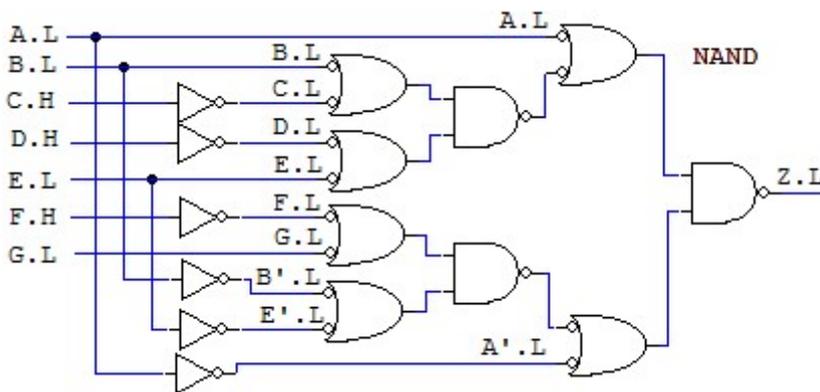


d) $Z.H = [C + (A + B) D E] (D + E + A B)$, para A.L, B.H, C.H, D.L, E.L





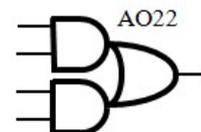
e) $Z.L = [A + (B + C) (D + E)] \cdot [(F + G) (\bar{B} + \bar{E}) + \bar{A}]$
 para A.L, B.L, C.H, D.H, E.L, F.H, G.L



Página 10_2. Como resultado de una minimización multinivel se han generado las siguientes expresiones lógicas:

$$F1 = [C + (\bar{A} + B) (\bar{D} + E)] [\bar{C} + (B + \bar{E}) (A + D)]$$

$$F2 = [\bar{C} + (\bar{A} + B) (\bar{D} + E)] [C + \bar{B} (A + D)]$$



Implementar las expresiones lógicas dadas para F1 y F2 utilizando el menor número de elementos lógicos del tipo AO22 e inversores, para F1.H, F2.L, A.H, B.H, C.L, D.L, E.H.

El circuito AO22 realiza la operación $X1 X2 + X3 X4$. Sin embargo, las funciones lógicas F1 y F2 se pueden generar a base a operaciones del tipo $(Y1 + Y2) (Y3 + Y4)$. Por ejemplo:

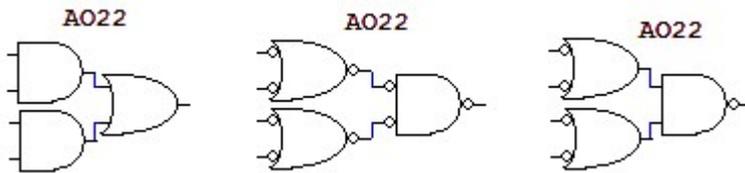
$$F1 = (C + X)(\bar{C} + Y) \quad F2 = (\bar{C} + X)(C + Z)$$

$$X = (\bar{A} + B) (\bar{D} + E)$$

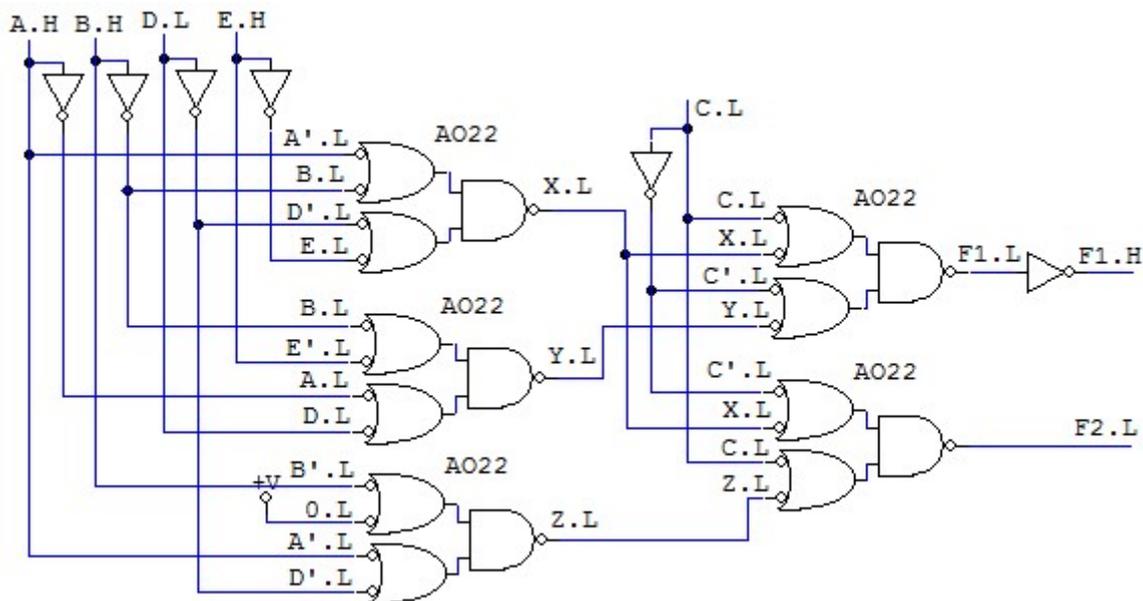
$$Y = (B + \bar{E}) (A + D)$$

$$Z = \bar{B} (\bar{A} + \bar{D}) = (\bar{B} + 0) (\bar{A} + \bar{D})$$

La puerta AO22 también puede hacer una operación del tipo $(Y1 + Y2) (Y3 + Y4)$, si se transforman adecuadamente las puertas internas, y se consideran las entradas y las salidas en lógica negativa.



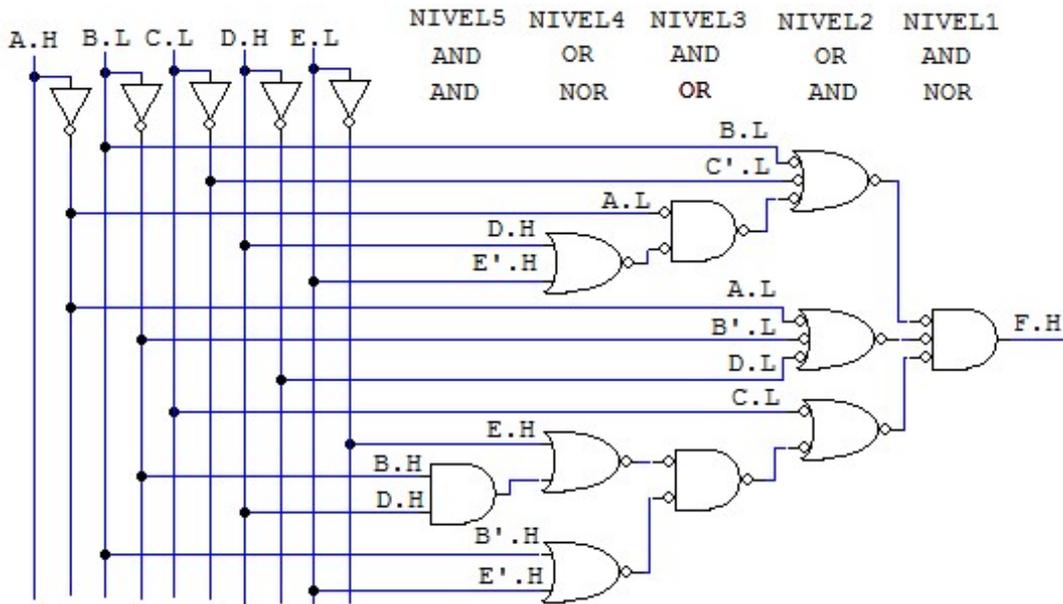
Con esto ya puedo plantear el circuito con 5 elementos AO22 y puertas NOT. La complementación se indica en la figura mediante ‘.



Página 10_3. Implementar la siguiente función lógica en cinco niveles, que desde la salida (1º nivel) hacia las entradas (5º nivel) usan puertas NOR (1º nivel), AND (2º nivel), OR (3º nivel), NOR (4º nivel) y AND (5º nivel) para entradas A.H, B.L, C.L, D.H, E.L y salida F.H

$$F(A, B, C, D, E) = [B + \bar{C} + A(D + \bar{E})](A + \bar{B} + D)[C + (E + BD)(\bar{B} + \bar{E})]$$

Al examinar la expresión lógica veo que desde la salida hacia las entradas el 1º nivel realiza una operación AND, el 2º nivel OR, el 3º AND, el 4º OR y el 5º AND. Entonces en el primer nivel tengo que hacer la operación AND con una puerta NOR, en el segundo nivel las operaciones OR con puertas AND, en el tercer nivel las operaciones AND con puertas OR, en el cuarto nivel las operaciones OR con puertas NOR, y en el quinto nivel las operaciones AND con puertas AND. Utilizo las puertas directamente o transformadas, y compruebo que en los nudos internos las polaridades son las mismas entre la salida de las puertas de un nivel y las entradas de las puertas del nivel anterior. Si eso se cumple, ajusto las conexiones entre las entradas del circuito y las entradas de las puertas, y compruebo que la polaridad de la salida F es correcta, usando un inversor si fuese necesario. Muestro el circuito que cumple las especificaciones, donde la complementación se indica mediante ‘.

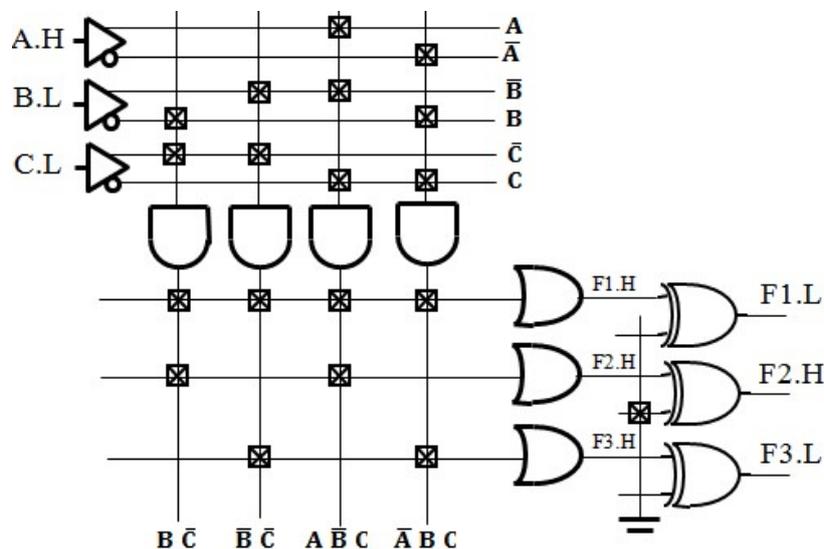


Página 11_a. Realizar una implementación en dos niveles de las siguientes funciones usando: a) PLA, b) PAL; c) ROM.

En este problema se deben implementar los mismos sistemas multifunción del problema 9, pero esta vez con circuitos programables PLA, PAL y ROM. La implementación con circuitos PLAs y PALs es similar a lo que se ha hecho en dos niveles. Hay un buffer que genera la señal de entrada y su complemento, líneas AND y OR que funcionan como puertas lógicas, aunque se podrían usar líneas de otro tipo o en otro orden en los dos niveles, como con las puertas lógicas. Se puede “jugar” con la polaridad de las entradas y las salidas, para esto último se puede incluir en la salida de los dispositivos una EXOR programable.

En la implementación con ROM se carga en el dispositivo directamente la tabla de verdad del problema, la cual hay que calcular.

- a) $F1(A, B, C) = B\bar{C} + \bar{B}\bar{C} + A\bar{B}C + \bar{A}BC$
- $F2(A, B, C) = B\bar{C} + A\bar{B}C$
- $F3(A, B, C) = \bar{B}\bar{C} + \bar{A}BC$
- Para A.H, B.L, C.L, F1.L, F2.H y F3.L



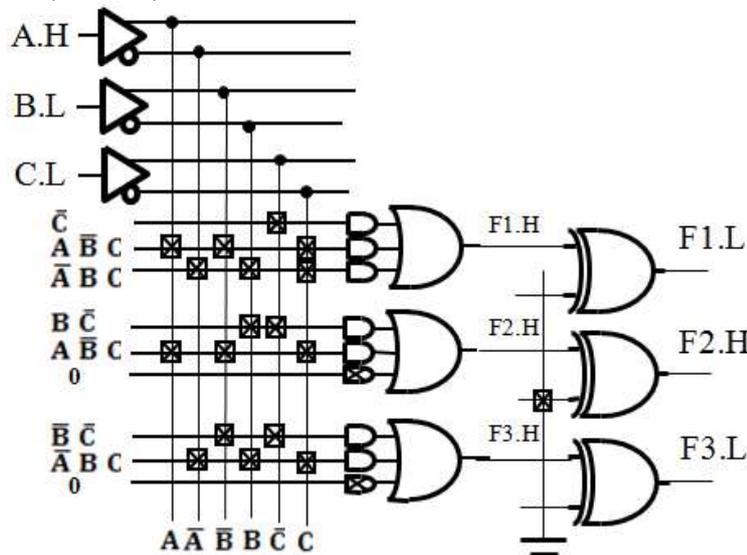
Con PLA los dos planos o niveles son programables. Hay tres entradas, luego se usan tres buffers de entrada, hay tres salidas, luego hay que usar tres líneas OR, y cuatro términos productos luego hay que usar cuatro líneas AND. Las salidas del plano OR son todas .H: uso las puertas EXOR: con fusible, entrada a GND => $X \oplus 0 = X$ (mantiene la polaridad); sin fusible, entrada a "1" => $X \oplus 1 = \bar{X}$ (cambio de polaridad).

En una PAL las puertas OR son fijas, las líneas AND no se comparten entre puertas OR, y habría que repetir las por cada salida. Una mejor minimización de las funciones es por cada salida individual en vez de por múltiples salidas, reduciendo el número de términos productos (implicantes) de cada salida sin importar el número de literales (no hace falta que sean primos). En el problema puedo reducir un término producto en F1 (se podrían reducir más literales, pero no es significativo para la implementación) mediante el teorema de adyacencia.

$$F1(A, B, C) = \bar{C} + A\bar{B}C + \bar{A}BC$$

$$F2(A, B, C) = B\bar{C} + A\bar{B}C$$

$$F3(A, B, C) = \bar{B}\bar{C} + \bar{A}BC$$



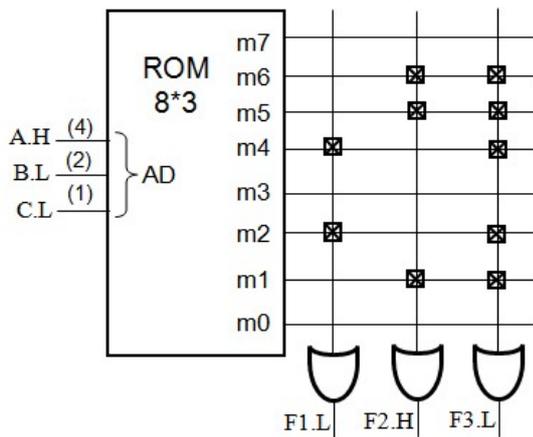
Para implementar el circuito con una ROM, tengo que extraer la tabla de verdad, que ajusto en función de las señales .L, que complemento. El circuito se implementa usando un circuito que genere todos los minterms posibles de la función (decodificador, se estudia en el tema IIIb) y, para cada salida, se hace el OR de sus 1s en DecF usando líneas OR programables.

Dec	A	B	C	F1	F2	F3	A.H	B.L	C.L	F1.L	F2.H	F3.L	DecF
0	0	0	0	1	0	1	0	1	1	0	0	0	3
1	0	0	1	0	0	0	0	1	0	1	0	1	2
2	0	1	0	1	1	0	0	0	1	0	1	1	1
3	0	1	1	1	0	1	0	0	0	0	0	0	0
4	1	0	0	1	0	1	1	1	1	0	0	0	7
5	1	0	1	1	1	0	1	1	0	0	1	1	6
6	1	1	0	1	1	0	1	0	1	0	1	1	5
7	1	1	1	0	0	0	1	0	0	1	0	1	4

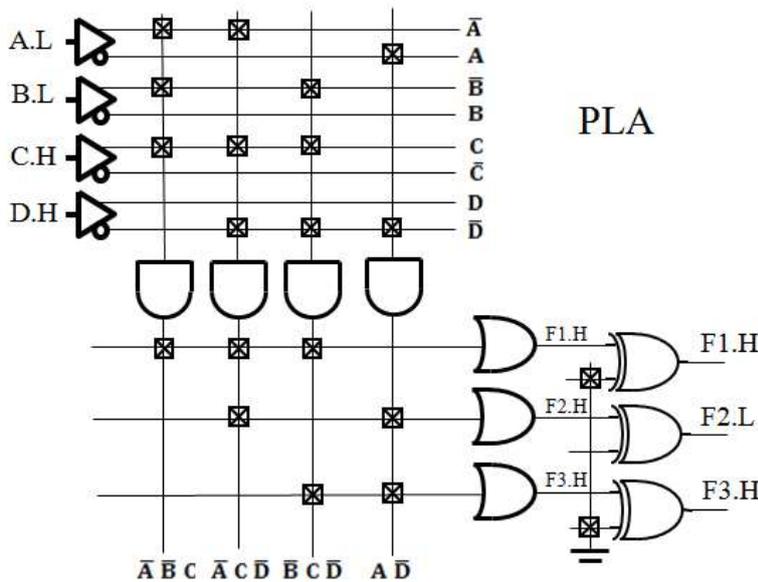
$$F1.L = F1(A, B, C) = \sum(2, 4)$$

$$F2.H = F2(A, B, C) = \sum(1, 5, 6)$$

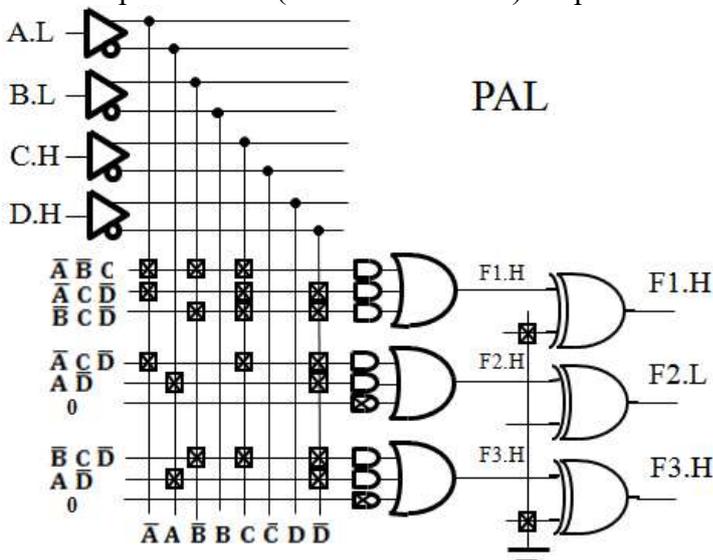
$$F3.L = F3(A, B, C) = \sum(1, 2, 4, 5, 6)$$



- b) $F1(A, B, C, D) = \bar{A}\bar{B}C + \bar{A}C\bar{D} + \bar{B}C\bar{D}$
 $F2(A, B, C, D) = \bar{A}C\bar{D} + A\bar{D}$
 $F3(A, B, C, D) = \bar{B}C\bar{D} + A\bar{D}$
 Para A.L, B.L, C.H, D.H, F1.H, F2.L y F3.H



Para la implementación mediante PAL compruebo que no se puede reducir el número de términos por función (sí un literal en F2). Implemento las funciones tal como están definidas.



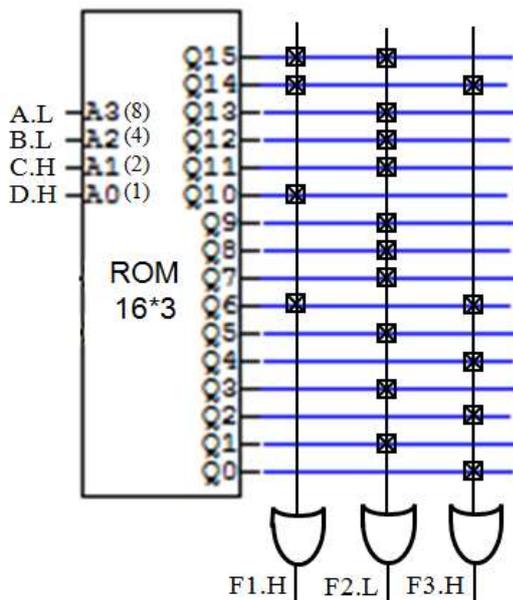
Para generar la implementación con una ROM genero la tabla de verdad, complemento las entradas y las salidas .L, y genero una tabla de verdad modificada, de la que obtengo las funciones lógicas en notación decimal usando la notación de DecF, y de ahí el circuito.

Dec	A	B	C	D	F1	F2	F3	A.L	B.L	C.H	D.H	F1.H	F2.L	F3.H	DecF
0	0	0	0	0	0	0	0	1	1	0	0	0	1	0	12
1	0	0	0	1	0	0	0	1	1	0	1	0	1	0	13
2	0	0	1	0	1	1	1	1	1	1	0	1	0	1	14
3	0	0	1	1	1	0	0	1	1	1	1	1	1	0	15
4	0	1	0	0	0	0	0	1	0	0	0	0	1	0	8
5	0	1	0	1	0	0	0	1	0	0	1	0	1	0	9
6	0	1	1	0	1	1	0	1	0	1	0	1	0	0	10
7	0	1	1	1	0	0	0	1	0	1	1	0	1	0	11
8	1	0	0	0	0	1	1	0	1	0	0	0	0	1	4
9	1	0	0	1	0	0	0	0	1	0	1	0	1	0	5
10	1	0	1	0	1	1	1	0	1	1	0	1	0	1	6
11	1	0	1	1	0	0	0	0	1	1	1	0	1	0	7
12	1	1	0	0	0	1	1	0	0	0	0	0	0	1	0
13	1	1	0	1	0	0	0	0	0	0	1	0	1	0	1
14	1	1	1	0	0	1	1	0	0	1	0	0	0	1	2
15	1	1	1	1	0	0	0	0	0	1	1	0	1	0	3

$$F1.H = F1(A, B, C, D) = \sum(6, 10, 14, 15)$$

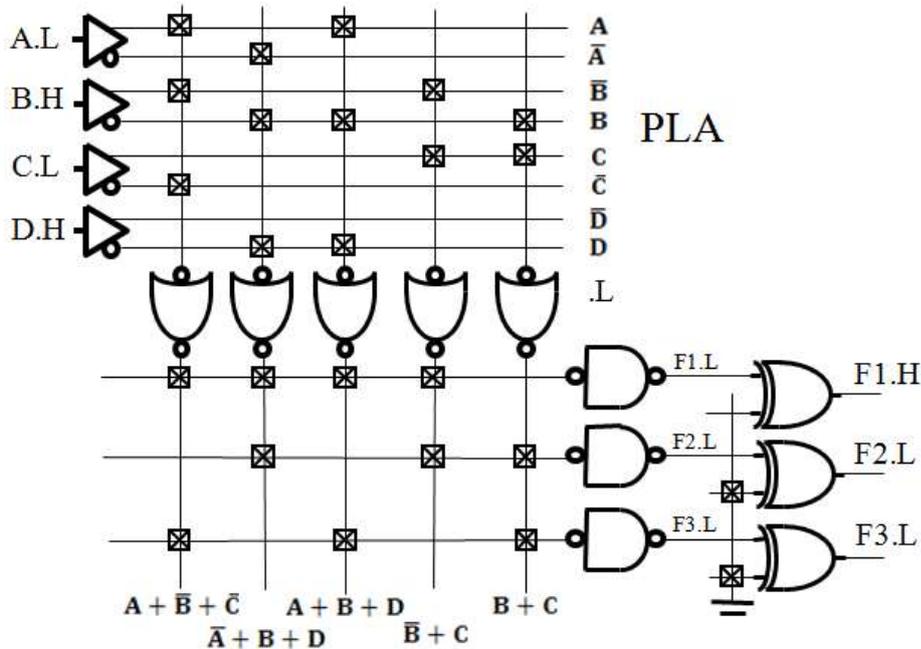
$$F2.L = F2(A, B, C, D) = \sum(1, 3, 5, 7, 8, 9, 11, 12, 13, 15)$$

$$F3.H = F3(A, B, C, D) = \sum(0, 2, 4, 6, 14)$$



- c) $F1(A, B, C, D) = (A + \bar{B} + \bar{C}) (\bar{A} + B + D) (A + B + D) (\bar{B} + C)$
 $F2(A, B, C, D) = (\bar{A} + B + D) (\bar{B} + C) (B + C)$
 $F3(A, B, C, D) = (A + \bar{B} + \bar{C}) (A + B + D) (B + C)$
 Para A.L, B.H, C.L, D.H, F1.H, F2.L y F3.L

Las formas POS pueden implementarse con PLAs de plano AND y plano OR programables, haciendo la misma transformación que con las puertas lógicas: se sustituyen las líneas AND en lógica positiva por líneas OR en lógica negativa, y líneas OR en lógica positiva por líneas AND, en lógica negativa. Igual que con puertas lógicas, también se podrían usar las leyes de De Morgan y hacer, por ejemplo, $F_{POS}.H = \overline{\overline{F_{POS}.H}} = \overline{FSOP.H} = FSOP.L$. Uso la transformación de puertas:

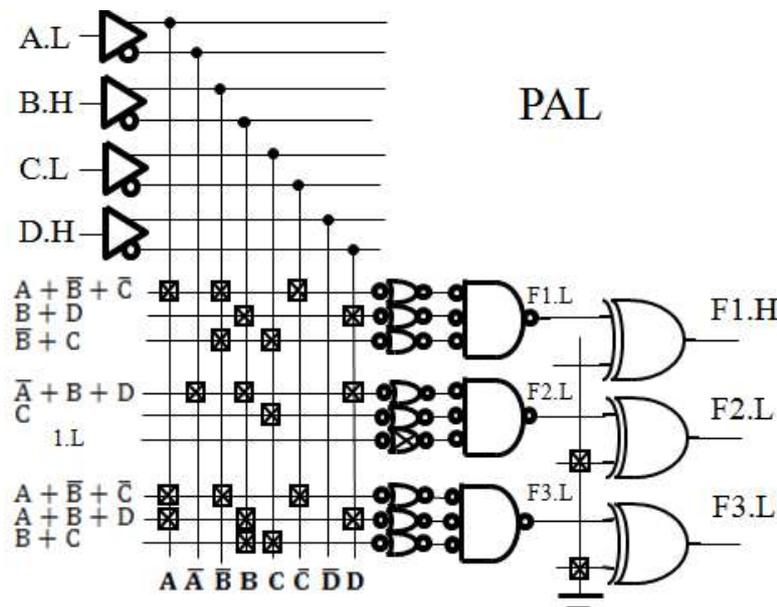


Para la implementación con PAL, primero reduzco términos suma en F1 y F2 con el teorema de adyacencia. F3 se mantiene igual. Se podría reducir un literal más por el teorema de simplificación en F1, pero no tiene importancia para la implementación en PAL.

$$F1(A, B, C, D) = (A + \bar{B} + \bar{C})(B + D)(\bar{B} + C)$$

$$F2(A, B, C, D) = (\bar{A} + B + D)C$$

$$F3(A, B, C, D) = (A + \bar{B} + \bar{C})(A + B + D)(B + C)$$



Para la implementación con ROM repito lo hecho en los apartados anteriores. Genero la tabla de verdad, complemento los literales .L, e implemento cada salida como una suma de los minterms según la notación de DecF añadiendo una línea OR por salida.

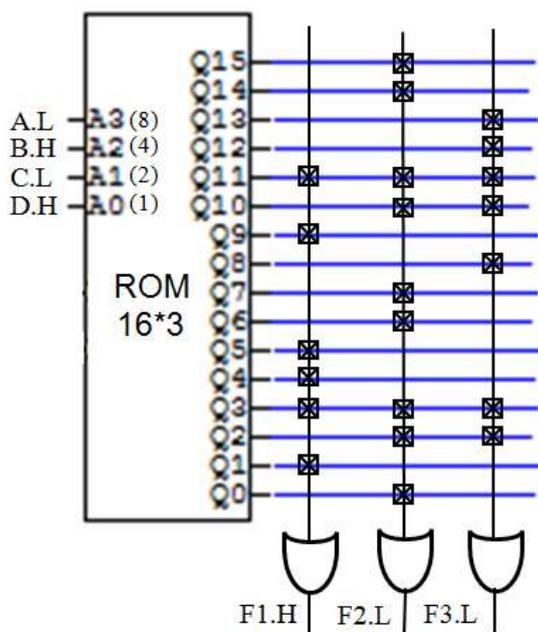
Dec	A	B	C	D	F1	F2	F3	A.L	B.H	C.L	D.H	F1.H	F2.L	F3.L	DecF
0	0	0	0	0	0	0	0	1	0	1	0	0	1	1	10
1	0	0	0	1	1	0	0	1	0	1	1	1	1	1	11
2	0	0	1	0	0	1	0	1	0	0	0	0	0	1	8
3	0	0	1	1	1	1	1	1	0	0	1	1	0	0	9
4	0	1	0	0	0	0	1	1	1	1	0	0	1	0	14
5	0	1	0	1	0	0	1	1	1	1	1	0	1	0	15
6	0	1	1	0	0	1	0	1	1	0	0	0	0	1	12
7	0	1	1	1	0	1	0	1	1	0	1	0	0	1	13
8	1	0	0	0	0	0	0	0	0	1	0	0	1	1	2
9	1	0	0	1	1	0	0	0	0	1	1	1	1	1	3
10	1	0	1	0	0	0	1	0	0	0	0	0	1	0	0
11	1	0	1	1	1	1	1	0	0	0	1	1	0	0	1
12	1	1	0	0	0	0	1	0	1	1	0	0	1	0	6
13	1	1	0	1	0	0	1	0	1	1	1	0	1	0	7
14	1	1	1	0	1	1	1	0	1	0	0	1	0	0	4
15	1	1	1	1	1	1	1	0	1	0	1	1	0	0	5

Según la tabla, las funciones quedan:

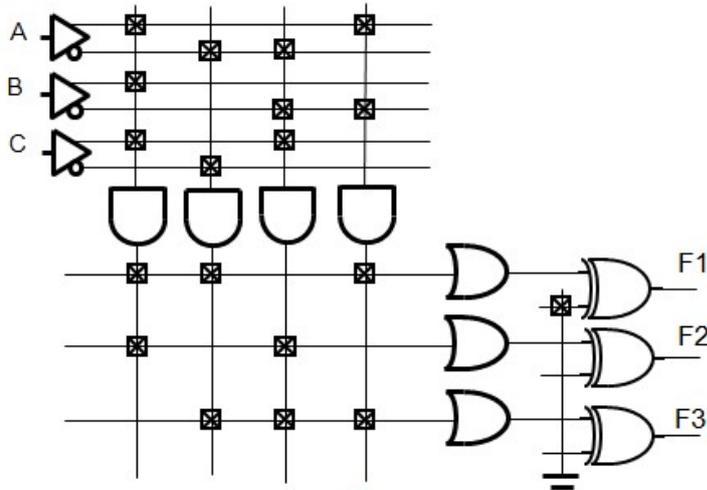
$$F1.H = F1(A, B, C, D) = \sum(1, 3, 4, 5, 9, 11)$$

$$F2.L = F2(A, B, C, D) = \sum(0, 2, 3, 6, 7, 10, 11, 14, 15)$$

$$F3.L = F3(A, B, C, D) = \sum(2, 3, 8, 10, 11, 12, 13)$$



Página 12_1. Dado el siguiente circuito PLA, indicar las funciones lógicas que realiza en notación decimal.



Supongo que todas las entradas y salidas están en lógica positiva (.H). Según esto, de izquierda (LA1) a derecha (LA4), las líneas AND realizan las funciones:

$$LA1 = A B C; LA2 = \bar{A} \bar{C}; LA3 = \bar{A} \bar{B} C, \text{ y } LA4 = A \bar{B}.$$

Usando las líneas OR para generar las funciones, las líneas OR de arriba abajo son:

$$LO1 = LA1 + LA2 + LA4; LO2 = LA1 + LA3; LO3 = LA2 + LA3 + LA4.$$

Según las conexiones de las puertas EXOR.

$$F1 = 0 \oplus LO1 = LO1; F2 = 1 \oplus LO2 = \overline{LO2}; F3 = 1 \oplus LO3 = \overline{LO3}.$$

Con tablas de verdad obtengo las funciones lógicas, de las que saco su notación decimal.

Dec	A	B	C	LA1	LA2	LA3	LA4	LO1	LO2	LO3	F1	F2	F3
0	0	0	0	0	1	0	0	1	0	1	1	1	0
1	0	0	1	0	0	1	0	0	1	1	0	0	0
2	0	1	0	0	1	0	0	1	0	1	1	1	0
3	0	1	1	0	0	0	0	0	0	0	0	1	1
4	1	0	0	0	0	0	1	1	0	1	1	1	0
5	1	0	1	0	0	0	1	1	0	1	1	1	0
6	1	1	0	0	0	0	0	0	0	0	0	1	1
7	1	1	1	1	0	0	0	1	1	0	1	0	1

$$F1(A, B, C) = \sum (0, 2, 4, 5, 7)$$

$$F2(A, B, C) = \sum (0, 2, 3, 4, 5, 6)$$

$$F3(A, B, C) = \sum (3, 6, 7)$$

Página 12_2. Se quieren implementar funciones lógicas usando un dispositivo PAL. ¿Cuál es el criterio de minimización lógica más adecuado para este dispositivo?

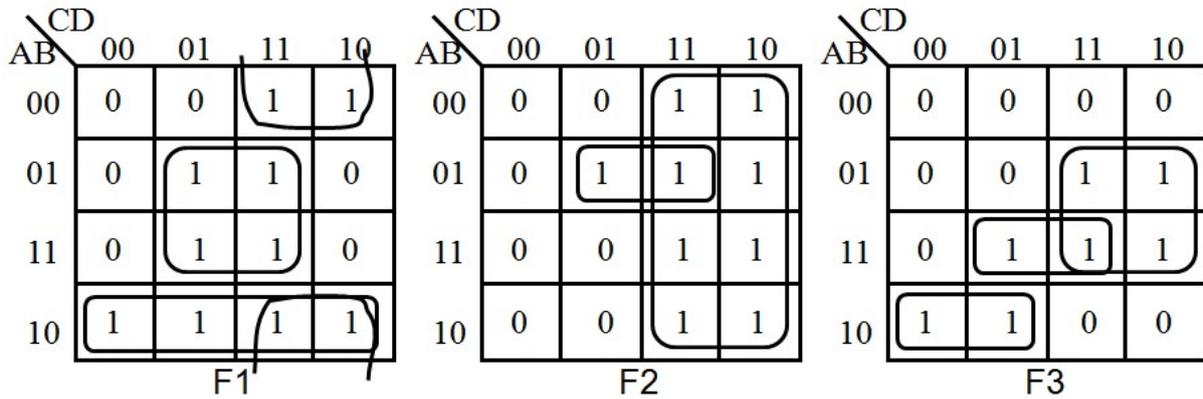
Implementar las siguientes funciones lógicas usando un dispositivo PAL con polaridad de salida programable, para A.L, B.L, C.H, D.H, F1.L, F2.H, F3.H.

$$F1(A,B,C,D) = \sum(2,3,5,7,8,9,10,11,13,15)$$

$$F2(A,B,C,D) = \sum(2,3,5,6,7,10,11,14,15)$$

$$F3(A,B,C,D) = \sum(6,7,8,9,13,14,15)$$

Debido a que en una PAL básica la implementación de las funciones lógicas de dos niveles no comparten términos producto, la minimización conjunta no tiene sentido. El criterio de minimización debería ser obtener el menor número de términos productos (o líneas AND) para cada función lógica por separado, sin importar el número de entradas de los términos producto, ya que las líneas AND pueden tener cualquier número de entradas. Si minimizo un problema de varias salidas con mapas de Karnaugh, minimizo cada salida por separado. Para las funciones del problema:

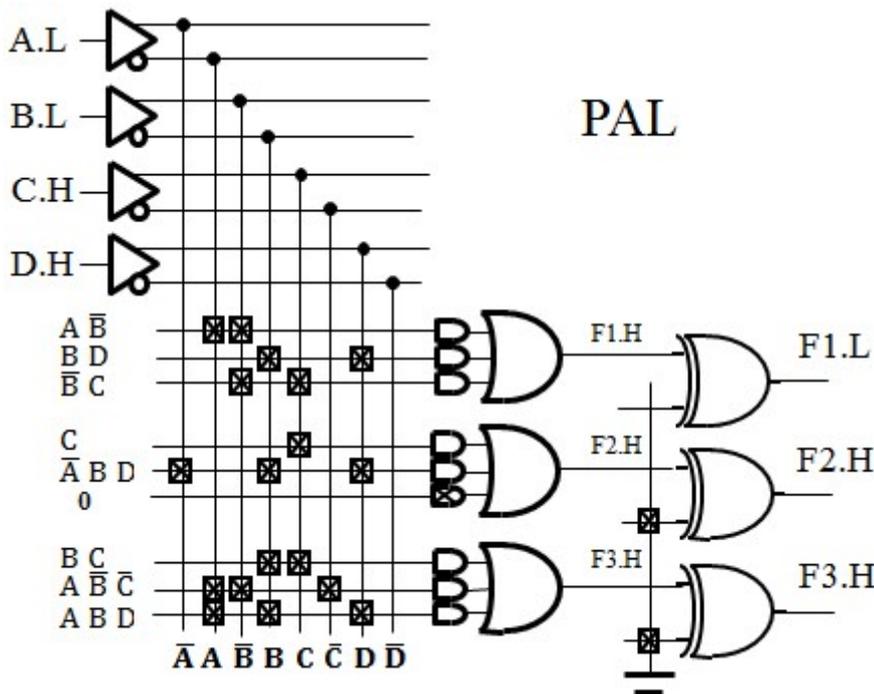


$$F1(A, B, C, D) = A\bar{B} + BD + \bar{B}C$$

$$F2(A, B, C, D) = C + \bar{A}BD$$

$$F3(A, B, C, D) = BC + A\bar{B}\bar{C} + ABD$$

Este resultado es similar al del problema 11c simplificado para PAL, con una forma SOP en lugar de una POS (en 11c y en F1, el término suma $A + \bar{B} + \bar{C}$ puede reducirse a $A + \bar{B}$). La implementación con una PAL queda:



La solución podría reducirse en F1, generando $\bar{F1} = B\bar{D} + \bar{A}\bar{B}\bar{C}$, usando sólo dos términos productos para hacer F1. En la puerta OR se genera $\bar{F1}.H = F1.L$. Queda propuesto.