Circuito

1. Una o más terminales de entrada
2. Una o más terminales de salida
3. especificación funcional
4. Especificación de timing - delay

Nodos
- inputs
- internos
- outputs

Tipos

Lógica combinacional

· No tiene memoria - no recuerda datos de entradas pasadas

· Las salidas están determinadas por los valores actuales

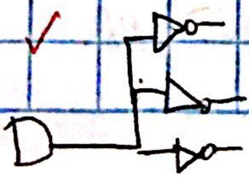
Lógica Secuencial

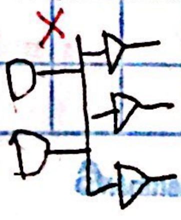
· Si tiene memoria - recuerda datos pasados sin haber pasado mucho tiempo

· Las salidas están determinadas por valores actuales y pasados de las entradas

Reglas combinacionales

1. Cada sub-circuito es combinacional
2. Cada nodo del circuito es
   · Una entrada al circuito
   · Está conectado a solamente una Terminal de salida, puede estar conectado a muchas entradas, pero una única salida





Ecuación booleana

Dan una especificación funcional de la salidas en enterminos de las entradas

Algebra booleana

Set de axiomas y teoremas para simplificar ecuaciones booleanas

Terminos

- literal: variable o su complemento
- Complemento: variable con una barra
- Implicante: producto de literales
- Mintérmino: producto que incluye toda las literales de entrada
- Maxtérmino: suma que incluye todas las literales de entrada

* Producto = AND
Suma = OR

Axiomas

A_1	$B=0$ if $B\neq 1$	A_1'	$B=1$ if $B\neq 0$	Binary field
A_2	$\overline{0}=1$	A_2'	$\overline{1}=0$	not
A_3	$0\cdot 0=0$	A_3'	$1+1=1$	AND/OR
A_4	$1\cdot 1=1$	A_4'	$0+0=0$	AND/OR
A_5	$0\cdot 1=1\cdot 0=0$	A_5'	$1+0=0+1=1$	AND/OR

Teoremas

T_1	$B \cdot 1 = B$	$T1'$	$B + 0 = B$	Identity
T_2	$B \cdot 0 = 0$	T_2'	$B + 1 = 1$	Null Element
T_3	$B \cdot B = 4$	T_3'	$B + B = B$	Indepotency
T_4	$\overline{\overline{B}} = B$	1		Involution
T_5	$B \cdot \overline{B} = 0$	T_5'	$B + \overline{B} = 1$	complements

Teoremas de multiples variables

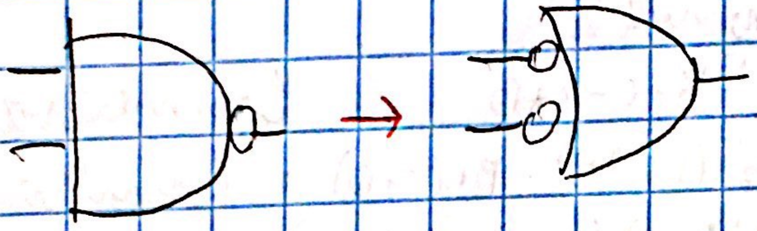
T_6	$B \cdot C = C \cdot B$	T_6'	$B + C = C + B$	Commutativity
T_7	$(B \cdot C) \cdot D = B \cdot (C \cdot D)$	T_7'	$(B+C)+D = B+(C+D)$	Associativity
T_8	$(B \cdot C) + (B \cdot D) = B \cdot (C+D)$	T_8'	$(B+C) \cdot (B+D) = B+(C \cdot D)$	Distributivity
T_9	$B \cdot (B+C) = B$	T_9'	$B + (B \cdot C) = B$	Covering
T_{10}	$(B \cdot C) + (B \cdot \overline{C}) = B$	T_{10}'	$(B+C) \cdot (B+\overline{C}) = B$	Combining
T_{11}	$(B \cdot C) + (\overline{B} \cdot D) + (C \cdot D) =$ $B \cdot C + \overline{B} \cdot D$	T_{11}'	$(B+C) \cdot (\overline{B}+D) \cdot (C+D)$ $= (B+C) \cdot (\overline{B}+D)$	Consensus
T_{12}	$\overline{B_0 \cdot B_1 \cdot B_2 \ldots}$ $= (\overline{B}_0 + \overline{B}_1 + \overline{B}_2)$	T_{12}'	$\overline{B_0 + B_1 + B_2 \ldots}$ $= (\overline{B}_0 \cdot \overline{B}_1 \cdot \overline{B}_2$	De Morgan's Theorem

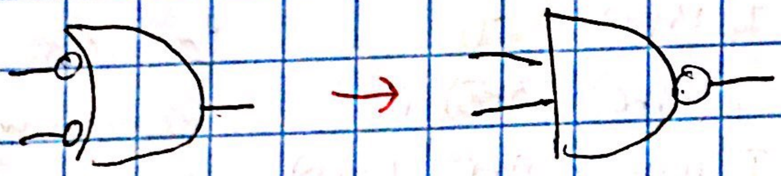
Jerarquía operacional

1. NOT
2. AND
3. OR

Teorema de de Morgan
Bubble pushing

Empujar las burbujas (NOTs) hacia atrás o hacia adelante, cambia el cuerpo de la compuerta de AND a OR y vice versa





* Tener AND en cascada es lo mismo que un solo AND de más entradas

HDL

Definir la función lógica y un CAD para que entregue la forma más optimizada