



Universidad Nacional Autónoma de México

Facultad de Ciencias

Organización y Arquitectura de Computadoras

Práctica 03 | Circuitos combinacionales

López Molina Andrés Daniel - 319117026

Reyna Méndez Cristian Ignacio

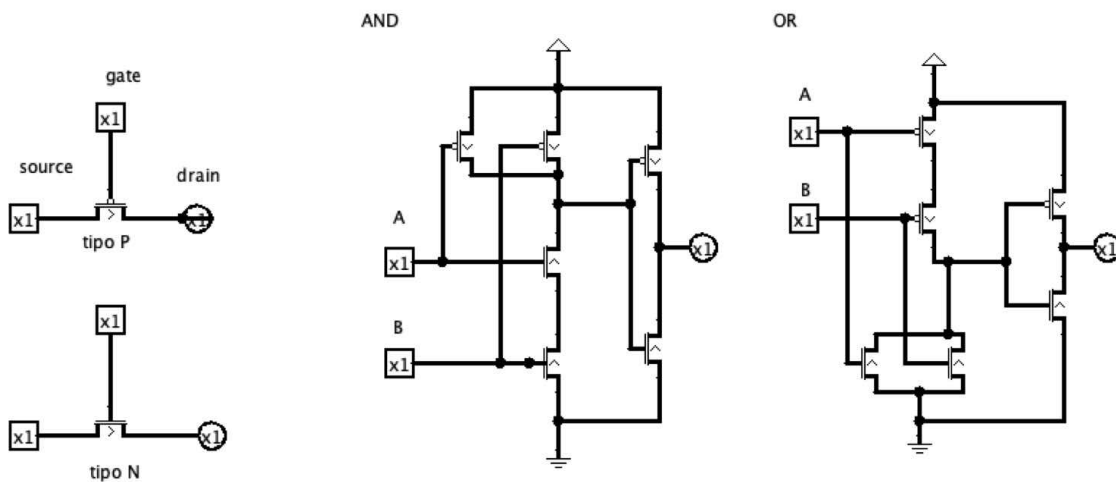
8 de septiembre de 2024



1. Ejercicios (Tablas de verdad, regla de correspondencia y mapas de Karnaugh):

2. Preguntas:

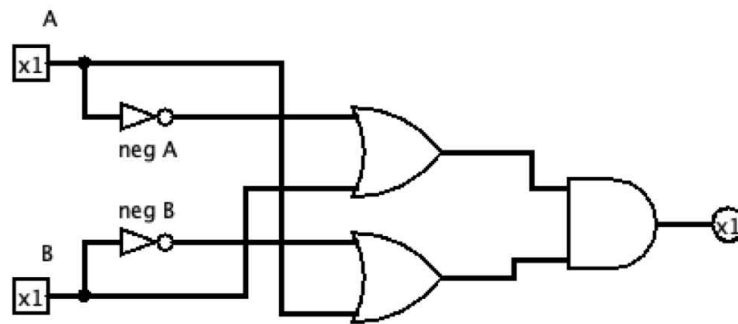
- Realiza una compuerta lógica que simule el comportamiento del AND y el OR usando Transistores de Tipo P y Tipo N.



A	B	$A \vee B$
0	0	0
0	1	1
1	0	1
1	1	1

A	B	$A \wedge B$
0	0	0
0	1	0
1	0	0
1	1	1

- Desarrolla un circuito que simule el comportamiento de la tabla de verdad del Si y solo si $P \iff Q$. Para este ejercicio solo puedes usar compuertas lógicas.



A	B	$A \leftrightarrow B$
0	0	1
0	1	0
1	0	0
1	1	1

- Sean $x, y \in \{0, 1, 2, 3\}$, desarrolla un comparador electrónico de 2 bits, las salidas del comparador deben ser: $x \geq y$ Para este ejercicio solo puedes usar compuertas lógicas.

x		y		$x \geq y$
D	C	B	A	
0	0	0	0	1
1	0	0	0	1
0	1	0	0	1
1	1	0	0	1
0	0	1	0	0
1	0	1	0	1
0	1	1	0	0
1	1	1	0	1
0	0	0	1	0
1	0	0	1	1
0	1	0	1	1
1	1	0	1	1
0	0	1	1	0
1	0	1	1	0
0	1	1	1	0
1	1	1	1	1

- Sean $x, y \in \{0, 1, 2, 3\}$, desarrolla un comparador electrónico de 2 bits, las salidas del comparador deben ser:

- $x < y$
- $x > y$
- $x = y$

Para este ejercicio solo puedes usar compuertas lógicas.



x		y		$x < y$	$x > y$	$x = y$
D	C	B	A			
0	0	0	0	0	0	1
1	0	0	0	0	1	0
0	1	0	0	0	1	0
1	1	0	0	0	1	0
0	0	1	0	1	0	0
1	0	1	0	0	0	1
0	1	1	0	1	0	0
1	1	1	0	0	1	0
0	0	0	1	1	0	0
1	0	0	1	0	1	0
0	1	0	1	0	0	1
1	1	0	1	0	1	0
0	0	1	1	1	0	0
1	0	1	1	1	0	0
0	1	1	1	1	0	0
1	1	1	1	0	0	1

$$\cdot X > Y$$

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0	1	1	1
$\bar{A}B$	0	0	1	0
AB	0	0	0	0
$A\bar{B}$	0	1	1	0

$$\bar{B}D + \bar{A}\bar{B}C + \bar{A}CB$$

$$\cdot X = Y$$

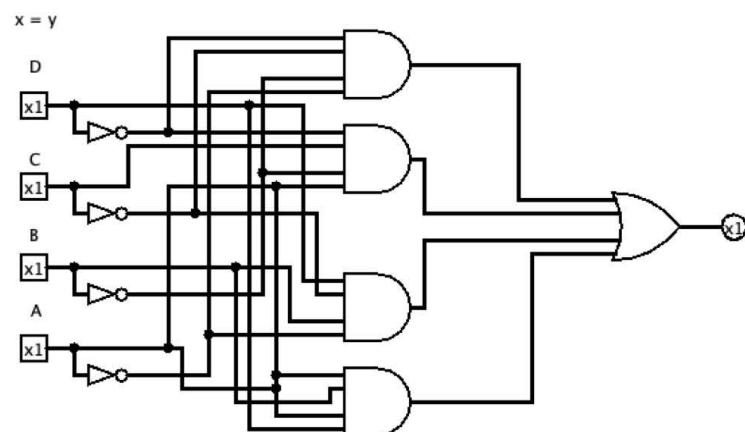
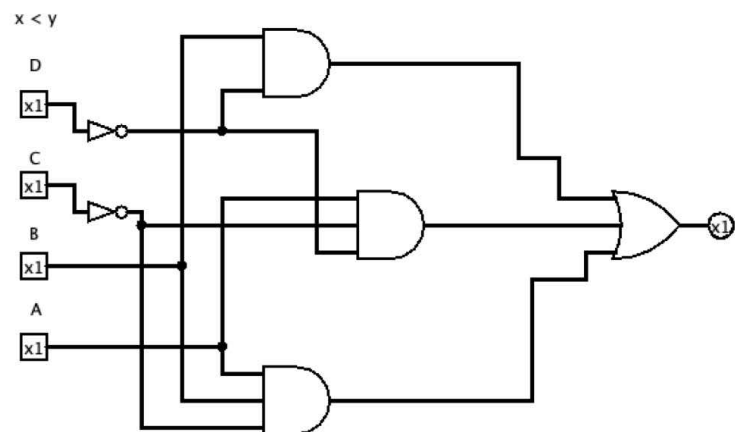
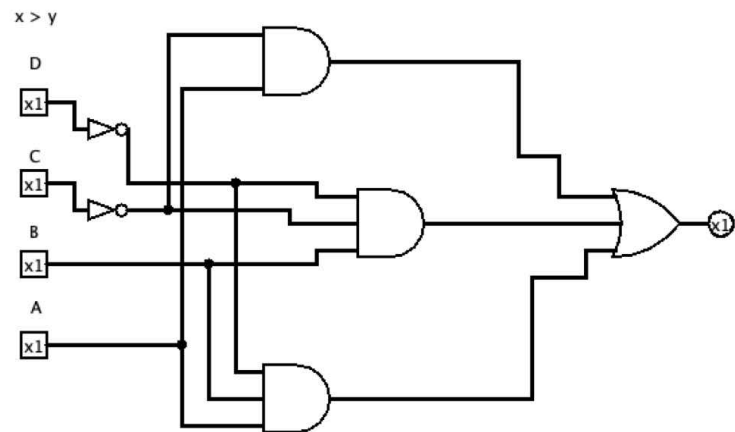
	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	1	0	0	0
$\bar{A}B$	0	1	0	0
AB	0	0	1	0
$A\bar{B}$	0	0	0	1

$$\bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}B\bar{C}D + A\bar{B}C\bar{D} + ABCD$$

$$\cdot X < Y$$

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0	0	0	0
$\bar{A}B$	1	0	0	1
AB	1	1	0	1
$A\bar{B}$	1	0	0	0

$$B\bar{D} + A\bar{C}\bar{D} + AB\bar{C}$$



1. ¿Cuál es la diferencia entre un transistor P y uno N?

Los transistores de tipo P y N se refieren a transistores bipolares de unión (BJT) o transistores de efecto de campo (FET). Un transistor NPN (tipo N) tiene dos capas de material tipo N separadas por una capa tipo P. Los electrones son los portadores mayoritarios de carga, y el transistor

se activa cuando una corriente fluye desde la base hacia el emisor. Un transistor PNP (tipo P) tiene dos capas de material tipo P separadas por una capa tipo N. Los huecos son los portadores mayoritarios de carga, y el transistor se activa cuando una corriente fluye desde el emisor hacia la base.

2. ¿Cuáles son las partes de un transistor?

Un transistor está compuesto por tres partes principales: Emisor: Es la terminal que emite los portadores de carga. En un transistor NPN, el emisor emite electrones, y en un transistor PNP, emite huecos. Base: Es la región intermedia que controla el flujo de portadores entre el emisor y el colector. Una pequeña corriente o voltaje en la base puede controlar una corriente más grande entre el emisor y el colector. Colector: Es la terminal que recoge los portadores de carga que fluyen a través de la base. En un NPN recoge electrones, mientras que en un PNP recoge huecos.

3. ¿Por qué se dice que los mapas de Karnaugh **no** nos dan una garantía de que **siempre** nos van a devolver la expresión mínima de una función?

Los mapas de Karnaugh permiten simplificar funciones booleanas, pero no garantizan siempre obtener la mínima expresión posible debido a que la simplificación puede depender de la elección de los grupos de términos a combinar. En algunos casos, es posible que exista una combinación alternativa que nos dé una expresión más simplificada, pero el proceso de agrupación de términos adyacentes en el mapa puede no llevar a esa combinación óptima, especialmente en funciones complejas o de muchas variables.

4. ¿Cuál es el procedimiento a seguir para desarrollar un circuito que resuelva un problema que involucre lógica combinacional?

El procedimiento para diseñar un circuito que resuelva un problema de lógica combinacional implica varios pasos: Definición clara del problema y la función que debe realizar el circuito. Especificación de la tabla de verdad, que describa las salidas del circuito en función de todas las combinaciones de entradas posibles. Simplificación de la función utilizando métodos como mapas de Karnaugh o álgebra booleana para reducir la cantidad de compuertas lógicas necesarias. Dibujo del diagrama lógico utilizando las compuertas necesarias (AND, OR, NOT, etc.) según la expresión simplificada. Implementación física del circuito o simularlo para verificar su correcto funcionamiento.

5. Si una función de conmutación se evalúa a más ceros que unos, ¿es conveniente usar minterminos o maxterminos?, ¿en el caso que se evalué a más unos que ceros?

- Si una función de conmutación se evalúa a más ceros que unos, es conveniente usar maxterminos, ya que estos cubren los ceros en la función, lo que puede resultar en una expresión más sencilla.
- Si una función se evalúa a más unos que ceros, es conveniente usar minterminos, ya que estos representan los términos en los cuales la función es igual a uno y puede generar una expresión más compacta.

6. Analizando el trabajo realizado, ¿cuáles son los inconvenientes de desarrollar los circuitos de forma física?

- Complejidad y errores
- Limitaciones de espacio y costo
- Dificultad de modificación
- Problemas de entorno