Organización y Arquitectura de Computadoras Practica 3

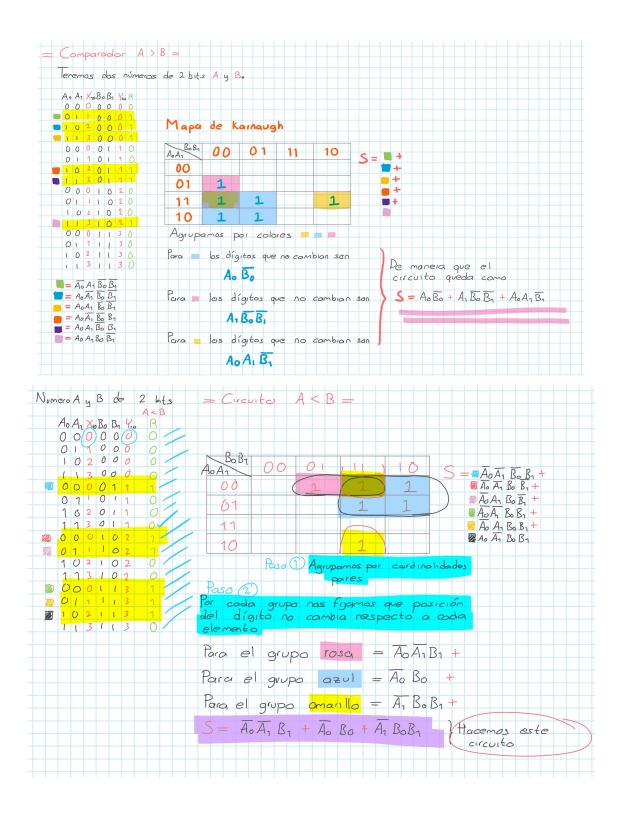
Arrieta Mancera Luis Sebastian

1. Preguntas

- 1.- ¿Cuál es el procedimiento a seguir para desarrollar un circuito que resuelva un problema que involucre lógica combinacional?
 - 1.- Analizar el problema para determinar las variables booleanas de entrada.
 - 2.- Realizar una tabla de verdad con las variables para asignar un estado a las funciones de conmutación que sean verdaderas.
 - 3.- Obtener la regla de correspondencia, ya sea reduciendo mintérminos o maxtérminos directamente usando álgebra booleana o con ayuda de mapas de Karnaugh.

Para resolver la actividad 9.1 me ayudé de los conocimientos de lógica. Se sabe que $P \Rightarrow Q$ es lógicamente equivalente a $\neg P \lor Q$ por lo que el circuito se construye usando una compuerta **NOT** y una compuerta **OR** ya que como tál no existe una compuerta de implicación \Rightarrow .

Para la actividad 9.2 se siguió el procedimiento descrito anteriormente con las lógicas A > B y A < B. Para la lógica A = B se tenía comparar que los números sean iguales bit a bit. Por lo que solo era necesario el uso de la compuerta **XNOR** y la compuerta **AND**.



2.- Si una función de conmutación se evalúa a más ceros que unos ¿es conveniente usar min-términos o max-términos? ¿En el caso que se evalúe a más unos que ceros?

Recordemos que

Mintérmino: Es una suma de productos donde solo se consideran los casos en los que el resultado fue positivo (1).Se niegan los 0's.

Maxtérminos: Es un producto de sumas donde solo se consideran los casos en los que el resultado fue negativo (0). Se niegan los 1's.

Entonces si una función de conmutación se evalúa a más ceros que unos entonces conviene usar mintérminos y para el caso en el que se evalúa más unos que ceros entonces conviene usar maxtérminos.

3.- Analizando el trabajo realizado, ¿Cuáles son los inconvenientes de desarrollar los circuitos de forma manual?

Me parece que los aspectos principales a tomar en cuenta al diseñar circuitos manualmente es lo tardado e inhumano que puede llegar a ser. Puesto que para n cantidad de variables se tiene 2^n combinaciones posibles. En este caso fue relativamente sencillo hacer un comparador para números de 2 bits.