

Organización y Arquitectura de Computadoras

2022-1

Práctica 2: Circuitos Combinacionales

Profesor: José de Jesús Galaviz Casas
Ayudante de laboratorio: Luis Soto Martínez

1. Objetivos

Generales:

- El alumno aprenderá a diseñar y simular circuitos combinacionales.

Particulares:

Al finalizar la práctica el alumno tendrá la capacidad de:

- Llevar a cabo el proceso de diseño de funciones de conmutación para resolver problemas lógicos simples.
- Utilizar las funciones básicas de *Logisim* para diseñar circuitos combinacionales.
- Realizar simulaciones de circuitos combinacionales con *Logisim*.

2. Requisitos

■ Conocimientos previos:

- Funciones de conmutación.
- Minimización de funciones de conmutación por medio de manipulación algebraica y mapas de Karnaugh.
- Los componentes básicos del diseño de circuitos combinacionales: transistores y compuertas **AND**, **OR** y **NOT**.

Se puede consultar los temas en [Mano] y [Patterson].

- **Tiempo de realización sugerido:**
5 horas.

- **Número de colaboradores:**
En parejas
- **Software a utilizar:**
 - *Java Runtime Environment* versión 5 o superior.
 - El paquete *Logisim* [**Logisim**].

3. Planteamiento

Para entender el funcionamiento de las computadoras modernas a un nivel general y antes de ser capaces de desarrollar nuevas arquitecturas, es necesario comprender el funcionamiento en el nivel de abstracción más bajo: los circuitos electrónicos. En la actualidad los circuitos electrónicos de una computadora se encuentran constituidos principalmente por transistores, por lo que es necesario conocer sus capacidades y limitaciones. El primer paso, entonces, es comenzar a diseñar circuitos combinatoriales con los componentes esenciales, además, antes de dar por finalizado el diseño, es de gran utilidad realizar pruebas por medio de simulaciones para detectar posibles errores, evitando realizar gastos económicos y de tiempo manufacturando componentes defectuosos.

4. Desarrollo

Para diseñar soluciones con circuitos a problemas que involucran lógica combinatorial, debes seguir una serie de pasos que te ayudarán a obtener el circuito combinatorial con la solución del problema, evitando errores y reduciendo el número de compuertas lógicas en el circuito.

Primero debes analizar el problema encontrando cuales serán las variables *booleanas* de entrada, las necesarias para codificar los datos y cuales serán las funciones de conmutación de salida, una vez más, considerando cuantas son necesarias para codificar la salida.

Realiza una tabla de verdad en la que a cada posible estado de las variables de entrada, asignes un estado a las funciones de conmutación. A partir de esta tabla, obtén la regla de correspondencia, ya sea reduciendo los minterminos o maxtérminos directamente con álgebra booleana o con ayuda de mapas de Karnaugh.

Finalmente realiza una simulación del circuito combinatorial con *Logisim* [**Burch**]. En el menú *ayuda* del programa podrás encontrar el manual de usuario con un tutorial para comenzar a usar el simulador, también se encuentra disponible en línea [**LogisimGuide**]. Se debe consultar las siguientes secciones, incluyen el manejo básico de programa y las funciones necesarias para el desarrollo de la práctica:

- *Beginner's tutorial.*
- *Libraries and attributes.*

- *Subcircuits.*
- *Wire bundles.*

5. Entrada

Dependiendo del ejercicio, se deberá elegir la codificación apropiada al dato de entrada.

6. Salida

Dependiendo del ejercicio, se deberá elegir la codificación apropiada al dato de salida.

7. Variables libres

No hay variables libres para el desarrollo de esta práctica.

8. Procedimiento

Deberás entregar un solo archivo de *Logisim* con las soluciones de los ejercicios, un subcircuito por cada uno y un documento con las respuesta a las preguntas planteadas y el diseño de la solución de cada ejercicio: tabla de verdad, la regla de correspondencia y los pasos de álgebra booleana o los mapas de Karnaugh utilizados para reducirla.

Recuerda etiquetar las entradas y salidas de cada uno de los subcircuitos.

9. Ejercicios

1. Construye las 3 compuertas básicas: Not, Or y And. Para corroborar la funcionalidad de sus compuertas, deberán construir circuitos para simular las siguientes funciones. Sólo puedes hacer uso de fuentes de alimentación power y ground, transistores, resistencias y pines de entrada y salida.

a) $A \iff B$

b) $A \oplus B$

2. Construye un circuito que resuelva las situaciones que se piden. Debes hacer uso de las compuertas que construiste en el ejercicio anterior.

a) Indicar si un número n es primo, con $n \in \{0, \dots, 15\}$

b) Simular la siguiente función y reduce mediante minterminos y máxterminos. Agrega ambos circuitos

E3	E2	E1	S
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

- c) En una planta de manejo de residuos tóxicos cuentan con la mejor tecnología para tratar desechos peligrosos y mantener a sus trabajadores seguros. Una parte fundamental de su sistema de protección consta de tres filtros que mantienen la toxicidad del área prácticamente nula. Sin embargo estos filtros pueden fallar, lo que volvería nocivo permanecer en la planta. Con la finalidad de monitorear el estado de los filtros, éstos tienen un sensor que indica si es que el filtro está fallando o no. Si uno de los filtros falla es posible trabajar con normalidad pero es necesario notificar al servicio técnico para que lo reparen a la brevedad. En caso de que fallen dos, la planta puede seguir trabajando pero los empleados se deben retirar mas temprano. Finalmente, si es que fallan los 3, será necesario activar el protocolo de alerta y evacuar inmediatamente la planta. Se necesita un mecanismo que indique la cantidad de filtros que fallan para que puedan tomar las acciones pertinentes segun sea el caso.

10. Preguntas

1. ¿Cuál es el procedimiento a seguir para desarrollar un circuito que resuelva un problema que involucre lógica combinacional?
2. Si una función de conmutación se evalúa a más ceros que unos ¿es conveniente usar mintérminos o maxtérminos? ¿En el caso que se evalúe a más unos que ceros?
3. Analizando el trabajo realizado, ¿cuáles son los inconvenientes de desarrollar los circuitos de forma manual?