

Laboratorio #04

Este laboratorio será trabajado de forma individual y se entregará de forma digital de acuerdo a la fecha de entrega en Canvas. Haga los ejercicios **SIN** usar una calculadora (a menos que se le indique lo contrario). Deberá identificar su entrega con su nombre, carné y sección.

Ejercicio 01

Encuentre la ecuación minimizada de las siguientes tablas utilizando mapas de Karnaugh. Deje evidencia de su procedimiento completo.

Tabla 01

A	B	C	Y
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

Tabla 02

A	B	C	Y
0	0	0	1
0	0	1	X
0	1	0	0
0	1	1	0
1	0	0	X
1	0	1	1
1	1	0	0
1	1	1	0

Tabla 03

A	B	C	D	Y
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

Tabla 04

A	B	C	D	Y
0	0	0	0	X
0	0	0	1	X
0	0	1	0	X
0	0	1	1	0
0	1	0	0	0
0	1	0	1	X
0	1	1	0	0
0	1	1	1	X
1	0	0	0	1
1	0	0	1	0
1	0	1	0	X
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	X
1	1	1	1	1

Ejercicio 02

Construya tablas de verdad a partir de las siguientes ecuaciones. Luego encuentre la ecuación minimizada utilizando mapas de Karnaugh y construya una tabla de verdad compacta utilizando *don't cares* en las entradas (lo más compacta que pueda).

1. $Y = A \cdot B \cdot C \cdot \overline{D} + A \cdot \overline{B} \cdot C \cdot \overline{D} + (\overline{A} + \overline{B} + \overline{C} + \overline{D})$
2. $Y = \overline{A} \cdot B \cdot C + \overline{B} \cdot \overline{C} + B \cdot C$
3. $Y = (\overline{A} + \overline{B} + \overline{C} \cdot D) + A \cdot D + B$
4. $Y = B \cdot C + \overline{A} \cdot \overline{B} \cdot \overline{C} + B \cdot \overline{C}$

Ejercicio 03

Implemente todas las ecuaciones booleanas minimizadas de los Ejercicios 01 y 02 en el *assignment* en CircuitVerse. Tome una captura de pantalla a cada ejercicio que implemente y agregue esta captura a su entrega de laboratorio en Canvas.

Ejercicio 04

Implemente **TODAS** las ecuaciones booleanas minimizadas de los ejercicios 01 y 02 en verilog. Deberá implementar 4 con *Gate Level Modelling* y otras 4 utilizando operadores lógicos. Deberá escribir únicamente 2 archivos **.v**:

- Uno de sus archivos tendrá **módulos** por cada una de las ecuaciones booleanas de los ejercicios 01 y 02.
- El segundo será el *testbench* de **todos** los ejercicios. Deberá utilizar múltiples bloques **initial** para probar todos los circuitos con todas sus entradas.
- Deberá desplegar en la terminal las pruebas de cada una de las ecuaciones (con todas las variaciones de las entradas).
- Deberá crear también el diagrama de **timing** combinado de todas las ecuaciones booleanas.
- Recuerde que deberá organizar el **timing** exacto para que se desplieguen correctamente sus soluciones.

En la entrega de Canvas deberá adjuntar un link a su repositorio (su repositorio debe ser público) y también deberá entregar un archivo .zip con todos sus códigos. De igual forma en su entrega deberá incluir **screenshots** de su código así como del diagrama de **timing**.

Ejercicio 05

Usted trabaja en una empresa donde diseñan e instalan sistemas de alarmas. Diseñe la lógica que cumpla con las siguientes condiciones:

- Si el sistema de alarma está armado entonces estar pendiente de los sensores
- Si el sensor de la ventana / puerta está roto (i.e. se apaga) entonces sonar la alarma y encender las luces
- Si el sensor de movimiento se dispara (detecta un intruso) entonces sonar la alarma y encender las luces
- ¿Cuáles son entradas y cuáles son salidas?
- Estados de las variables
 - Alarma y luces encendidas → 1 = encendido
 - Sensor de ventana / puerta → 1 = sensor OK
 - Sensor de movimiento → 0 = sensor **NO** detecta movimiento
 - El sistema de alarma está *armado* → 1 = sistema armado

Deberá construir la tabla de verdad, ecuaciones SOP y POS, y el mapa de Karnaugh para encontrar la ecuación simplificada. Implemente la solución en Circuitverse y también en Verilog (tanto SOP, POS como la solución con el Mapa de Karnaugh. Implemente *Gate Level Modelling* así como *Behavioral Modelling*).

En su entrega deberá incluir **screenshots** de su procedimiento completo, diagramas de Circuitverse, código de Verilog y los resultados del código de Verilog.