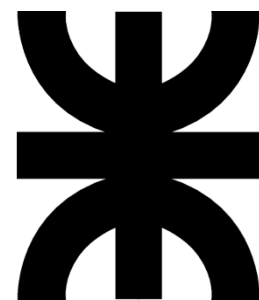


Universidad Tecnológica Nacional
Facultad Regional Córdoba



Técnicas DIGITALES I

Prácticos de aula

NIVEL: 3^{er} año

PLAN: 2025

BLOQUE : Tecnologías Aplicadas

JTP: Olmedo Sergio

Mercado Jorge

Índice

Unidad N° 1: Algebra de Boole.	4
Contenido.....	4
Resultado de aprendizaje.....	4
Objetivo.....	4
Observación:	4
Problemas	5
Problemas en clases.	5
Problemas adicionales.....	6
Problemas del libro de cátedra adicionales.	6
Unidad N° 2: Funciones lógicas y su minimización.	7
Contenido.....	7
Resultado de aprendizaje.....	7
Observación	7
Problemas	8
Problemas en clases.	8
Problemas adicionales.....	11
Unidad N° 3: Tecnología de las compuertas lógicas.	15
Contenido.....	15
Resultado de aprendizaje.....	15
Objetivo.....	15
Observación:	15
Problemas	15
Problemas en clases.	15
Problemas adicionales.....	18
Unidad N° 4: Circuitos combinacionales básicos.	19
Contenido.....	19

Resultado de aprendizaje.....	19
Objetivo.....	19
Observación:	19
Problemas	19
Problemas en clases.	19
Problemas adicionales.....	23
Problemas resueltos.	25

Unidad N° 5: Circuitos aritméticos. 26

Contenido.....	26
Resultado de aprendizaje.....	26
Objetivo.....	26
Observación:	26
Problemas	26
Problemas en clases.	26
Problemas adicionales.....	27

Unidad N° 6. Lógica secuencial síncrona 28

Contenido.....	28
Resultado de aprendizaje.....	28
Objetivo.....	28
Observación:	28
Problemas	28
Problemas en clases.	28
Problemas adicionales.....	34

Unidad N° 7. Registros y contadores 37

Contenido.....	37
Resultado de aprendizaje.....	37
Objetivo.....	37
Observación:	37
Problemas	37
Problemas en clases.	37

Problemas adicionales.....	38
----------------------------	----

Unidad N° xx: Xx.	40
--------------------------	-----------

Contenido.....	40
----------------	----

Resultado de aprendizaje.....	40
-------------------------------	----

Objetivo.....	40
---------------	----

Observación:	40
--------------------	----

Problemas	40
-----------------	----

Problemas en clases.	40
---------------------------	----

Problemas adicionales.....	40
----------------------------	----

Bibliografía	41
---------------------	-----------

Unidad N° 1: Álgebra de Boole.

Contenido.

Definición. Postulados. Principio de dualidad. Operaciones lógicas y su analogía mecánica con llaves. Compuertas básicas: AND, OR, NOT, y XOR. Tablas de verdad y compuertas derivadas (NAND, NOR). Teoremas del Álgebra de Boole. Teorema de De Morgan. Teorema del consenso. Concepto del lenguaje de descripción de Hardware Verilog. Elementos del lenguaje. Declaración de un módulo. Instanciación. Sentencia de asignación continua: ASSIGN. Operadores bitwise y de reducción. Variables internas. Representación de números.

Carga horaria por Unidad: 12hs

Resultado de aprendizaje.

RA1: Desarrollar el álgebra de conmutación para alcanzar un conocimiento básico de circuitos digitales y su función lógica considerando que un Sistema Digital es una interconexión de módulos digitales.

RA2: Emplear las herramientas de diseño adecuadas para la resolución de problemas a partir de una consigna dada.

RA3: Producir la minimización a nivel de compuertas de las funciones booleanas para alcanzar el menor costo posible teniendo en cuenta la eficiencia de la implementación.

RA7: Utilizar un lenguaje de descripción de hardware como Verilog con el propósito de implementar diferentes configuraciones de hardware utilizando dispositivos lógicos programables.

Objetivo.

Que el alumno pueda resolver ejercicios de los temas vistos en la clase teórica.

Observación:

Se resolverán ejercicios que figuran con la numeración indicada al final de cada capítulo del libro propuesto por la cátedra **Digital Design Morris-Mano** más ejercicios adicionales que no son del libro propuesto.

En la carpeta de TP de Aula deberán incluirse los ejercicios vistos y resueltos en clase, tanto los del libro como cualquier otro que se resuelva o se dé como consigna.

Problemas**Problemas en clases.****Problema 1.1.**

Listar la tabla de verdad de las siguientes funciones:

$$(a) \quad F = xy + xy' + y'z$$

$$(b) \quad F = bc + a'c'$$

Problema 1.2.

Podemos realizar operaciones lógicas sobre cadenas de bits, considerando por separado cada par de bits (operación bit a bit). Dadas dos cadenas de 8 bits $A=10110001$ y $B=10101100$. Evaluar el resultado de los 8 bits después de realizar las siguientes operaciones lógicas.

$$(a) \quad \text{AND} \quad (b) \quad \text{OR} \quad (c) \quad \text{XOR} \quad (d) \quad \text{NOT } A \quad (e) \quad \text{NOT } B$$

Problema 1.3.

Dada la función Booleana:

$$F = xy + x'y' + yz'$$

- a) Implementar con compuertas AND, OR, e inversores
- b) Implementar con compuertas OR, e inversores
- c) Implementar con compuertas AND, e inversores

Problema 1.4.

Convierta las siguientes expresiones a suma de productos y productos de suma.

$$a) \quad (AB+C)(B+C'D)$$

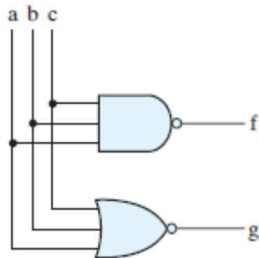
$$b) \quad x' + x(x+y')(y+z')$$

Problema 1.5.

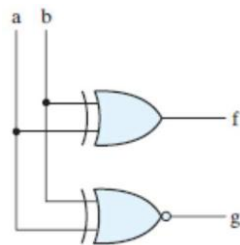
Realizar los puntos 1.1 y 1.4 en un simulador para verificar si las diferentes funciones tiene la misma salida.

Problemas adicionales.**Problema 1.A.1**

Mostrar las señales de salida f y g de la figura como función de las tres entradas a, b y c. Utilizar las ocho posibles combinaciones de las entradas a, b y c

**Problema 1.A.2**

Mostrar las señales de salida f y g de la figura como función de las dos entradas a y b. Utilizar las cuatro posibles combinaciones de las entradas a y b.

**Problemas del libro de cátedra adicionales.**

Resolver del libro de la cátedra:

M.Morris Mano & Michael D. Ciletti *DIGITAL DESIGN With an Introduction to de Verilog HDL – Fifth Edition*- Person.

Los siguientes ejercicios

Pag 69 problema 2.1; problema 2.4

pag 70 problema 2.14; problema 2.15; problema 2.17

pag. 71 problema 2.23; problema 2.27;

Unidad N° 2: Funciones lógicas y su minimización.

Contenido

Función del Álgebra de Boole. Concepto de Minitérminos y Maxitérminos. Forma canónica. Obtención de funciones desde la tabla de verdad. Expresión matemática de la Función. Función complemento. Expansión a la forma canónica, métodos tabulares y algebraicos. Conceptos básicos de minimización. Diagramas de Karnaugh para 2, 3, 4 y 5 variables. Condiciones no importa/no sucede. Aplicación de las funciones lógicas. Referencia introductoria a la estructura de un Dispositivo lógico programable. Descripción de funciones lógicas en Verilog. Introducción a las herramientas de CAD. Retardos en las compuertas. Introducción al concepto de Banco de pruebas (Testbench). Simulador ModelSim. Simulación de comportamiento y temporal. Demoras y respuestas transitorias. Glitches. Síntesis.

Carga horaria por Unidad: 12hs

Resultado de aprendizaje.

RA2: Emplear las herramientas de diseño adecuadas para la resolución de problemas a partir de una consigna dada.

RA3: Producir la minimización a nivel de compuertas de las funciones booleanas para alcanzar el menor costo posible teniendo en cuenta la eficiencia de la implementación.

RA7: Utilizar un lenguaje de descripción de hardware como Verilog con el propósito de implementar diferentes configuraciones de hardware utilizando dispositivos lógicos programables.

Observación

Se resolverán ejercicios que figuran con la numeración indicada al final de cada capítulo del libro propuesto por la cátedra Digital **Design Morris-Mano** más ejercicios adicionales que no son del libro propuesto.

En la carpeta de TP de Aula deberán incluirse los ejercicios vistos y resueltos en clase, tanto los del libro como cualquier otro que se resuelva o se dé como consigna.

Problemas**Problemas en clases.****Problema 2.1.**

Expandir las siguientes funciones booleanas a su forma canónica.

$$F1 = \overline{A}B + CA$$

$$F2 = (AB + A\overline{B}C)(\overline{B}D)$$

$$F3 = BD + A\overline{D}$$

Problema 2.2.

Expresar las siguientes ecuaciones en su forma canónica de suma de productos. (miniterminos).

$$F5_{(ABCD)} = (A + B + \overline{C}D)(A + \overline{D})$$

Problema 2.3.

Expresar las siguientes ecuaciones en su forma canónica de productos de suma (Maxiterminos).

$$F8 = \overline{A}\overline{B}C + (B + C)D + \overline{D}A$$

Problema 2.4.

Simplificar las siguientes funciones booleanas a su mínima expresión de literales.

$$(a)* xy + xy'$$

$$(c)* xyz + x'y + xyz'$$

$$(e) (a + b + c')(a' b' + c)$$

$$(b)* (x + y)(x + y')$$

$$(d)* (A + B)'(A' + B')'$$

$$(f) a'bc + abc' + abc + a'bc'$$

Problema 2.5.

Simplificar las siguientes funciones booleanas a su mínima expresión de literales.

$$(a)* \overline{A}BC + A'B + \overline{A}BC'$$

$$(c)* (x + y)'(x' + y')$$

$$(e)* (BC' + A'D)(AB' + CD')$$

$$(b)* x'yz + xz$$

$$(d)* xy + x(wz + wz')$$

$$(f) (a' + c')(a + b' + c')$$

Problema 2.6.

Reducir la siguiente expresión booleana al número de literales indicado:

<p>(a) $A'C' + ABC + AC'$</p> <p>(b) $(x'y' + z)' + z + xy + wz$</p> <p>(c) $A'B(D' + C'D) + B(A + A'CD)$</p> <p>(d) $(A' + C)(A' + C')(A + B + C'D)$</p> <p>(e) $ABC'D + A'BD + ABCD$</p>	<p>a) Tres literales.</p> <p>b) Tres literales.</p> <p>c) Un literal.</p> <p>d) Cuatro literales</p> <p>e) Dos literales</p>
---	--

Problema 2.7.

Realizar del ejercicios anterior (los puntos a, b y e) describiendo en Verilog la función inicial y la reducida. Verificar el RTL que es correcta la descripción. Simular y verificar que son correctas la función sin reducir y la reducida.

Problema 2.8.

Del ejercicio del problema 1.4, se solicita describir en Verilog la función inicial y resultado. Verificar el RTL que es correcta la descripción. Simular y verificar que son correctas la función inicial y el resultado.

Problema 2.9.

Simplifique las siguientes funciones con mapas de Karnaugh de 3 variables:

(a) $F(x, y, z) = \Sigma(0, 2, 4, 5)$

(b) $\bar{F}(x, y, z) = \Sigma(0, 2, 4, 5, 6)$

(c) $F(x, y, z) = \Sigma(0, 1, 2, 3, 5)$

(d) $F(x, y, z) = \Sigma(1, 2, 3, 7)$

Problema 2.10.

Simplifique las siguientes funciones con mapas de Karnaugh de 3 variables:

(a)* $F(x, y, z) = \Sigma(0, 1, 5, 7)$

(b)* $\bar{F}(x, y, z) = \Sigma(1, 2, 3, 6, 7)$

(c) $F(x, y, z) = \Sigma(2, 3, 4, 5)$

(d) $F(x, y, z) = \Sigma(1, 2, 3, 5, 6, 7)$

(e) $F(x, y, z) = \Sigma(0, 2, 4, 6)$

(f) $F(x, y, z) = \Sigma(3, 4, 5, 6, 7)$

Problema 2.11.

Simplifique las siguientes funciones con mapas de Karnaugh:

(a)* $F(x, y, z) = \Sigma(2, 3, 6, 7)$

(b)* $\bar{F}(A, B, C, D) = \Sigma(4, 6, 7, 15)$

(c)* $F(A, B, C, D) = \Sigma(3, 7, 11, 13, 14, 15)$

(d)* $F(w, x, y, z) = \Sigma(2, 3, 12, 13, 14, 15)$

(e) $F(w, x, y, z) = \Sigma(11, 12, 13, 14, 15)$

(f) $F(w, x, y, z) = \Sigma(8, 10, 12, 13, 14)$

Problema 2.12.

Simplificar las siguientes funciones booleanas con mapas de Karnaugh:

(a)* $F(\bar{A}, B, C, D) = \Pi(1, 3, 5, 7, 13, 15)$

(b) $F(A, B, C, D) = \Pi(1, 3, 6, 9, 11, 12, 14)$

Problema 2.13.

Simplifique las siguientes funciones con mapas de Karnaugh con el agregado de condiciones X “no importan”:

$af(A, B, C) = \sum(0,5)x = 1$	$df(A, B, C, D) = \sum(1,3,5)x = 2,6,9,11$
$bf(A, B, C) = \sum(0,2,5,6)x = 1,3,4$	$df(A, B, C, D) = \sum(0,1,3,11)x = 2,9,11$
$cf(x, y, z) = \prod(2,7)x = 0,1,3$	$ef(w, x, y, z) = \prod(0,1,3,4,5)x = 2,7,9$

Problema 2.14.

Dibuje el diagrama lógico NAND que implemente el complemento de la siguiente función:

$$F(A, B, C, D) = \sum(0, 1, 2, 3, 6, 10, 11, 14)$$

Problema 2.15.

Implementar el diagrama lógico utilizando solamente compuertas NOR de dos entradas para la siguiente función:

$$F(A, B, C, D) = (A \oplus B)'(C \oplus D)$$

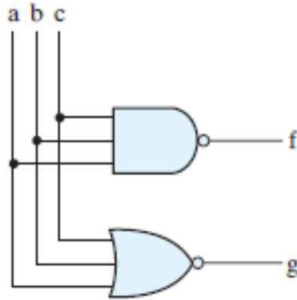
Problema 2.16.

Dibuje el circuito multinivel NOR para la siguiente expresión:

$$CD(B + C)A + (BC' + DE')$$

Problemas adicionales.**Problema 2.A.1.****Problema 2.A.2.**

Describir en Verilog la figura del esquemático; mostrar las señales de salida f y g de la figura como función de las entradas a ,b y c. Utilizar las ocho posible combinaciones de las entradas verificando la simulación. Verificar el RTL dado por la herramienta y analizarlo.

**Problema 2.A.2.**

Aplicar las propiedades del algebra y teoremas adecuados para simplificar las siguientes expresiones lógicas:

$$F1_{(ABCD)} = \overline{\overline{A}BC + B\overline{C}D}$$

$$F2_{(ABC)} = ABC + B(\overline{A}C + \overline{A}C)$$

$$F3_{(ABC)} = A.B.C + \overline{A}BC + \overline{A}\overline{B}C + \overline{A}\overline{B}\overline{C}$$

Problema 2.A.3.

Expresar las siguientes ecuaciones en su forma canónica de suma de productos (miniterminos).

$$F6_{(ABCD)} = A + \overline{(A \oplus B \oplus C \oplus D)}$$

$$F7_{(ABCD)} = (A + B)(B + C)(\overline{B}\overline{D})$$

Problema 2.A.4.

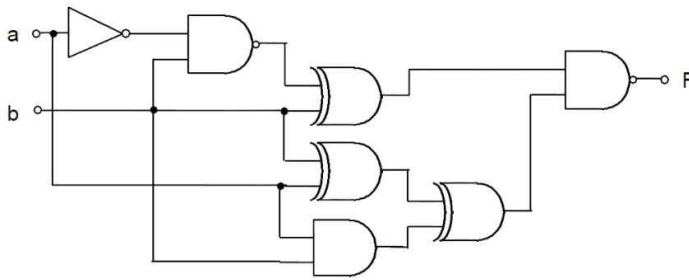
Expresar las siguientes ecuaciones en su forma canónica de productos de suma (Maxiterminos).

$$F9 = \overline{A}\overline{B}C + A\overline{B}\overline{D} + ABCD$$

$$F10 = A \oplus C + \overline{A}\overline{B}C + \overline{A}C\overline{D}$$

Problema 2.A.5.

Del siguiente circuito lógico escribir la función lógica, simplificarla aplicando las propiedades y teoremas del algebra de Boole e implementar el circuito resultante.



Problema 2.A.3.

Dibujar el diagrama lógico para implementar las siguientes funciones booleanas:

(a) $y = [(u + x') (y' + z)]$

(b) $y = (u \oplus y)' + x$

(c) $y = (u' + x') (y + z')$

(d) $y = u(x \oplus z) + y'$

(e) $y = u + yz + uxy$

(f) $y = u + x + x'(u + y')$

Problema 2.A.4.

Implementar la siguiente función Booleana:

$$F = xy + x'y' + y'z$$

- a) Con AND/OR e inversores
- b) Con OR e inversores
- c) Con AND e inversores
- d) Con NAND e inversores
- e) Con NOR e inversores

Problema 2.A.5.

Se quiere realizar un circuito que reciba un número BCD-XS3 (4 bits) y devuelva '1' si el número recibido es primo, y devuelva '0' si no lo es.

Se considerará el número 1 como número primo. El cero no es un número primo.

En ningún caso el circuito recibirá números que no estén codificados en BCD-XS3.

Se pide:

- a) Realizar la tabla de verdad de la señal de salida.
- b) Obtener la expresión reducida en suma de productos, y producto de sumas
- c) Dibujar el esquema en puertas de estas expresiones

Problema 2.A.6.

Se quiere realizar un circuito para activar la alarma de incendios (**A**) para la evacuación de un edificio. Para ello se tiene un sensor de gases (**G**), un sensor de humos (**H**), y dos señales procedentes de un termómetro que indican si la temperatura es mayor de 45°C (**T45**) y si la temperatura es mayor de 60°C (**T60**).

Debido a que a veces los sensores detectan humos y gases que no siempre proceden de incendios (por ejemplo de los cigarrillos o las cocinas), para evitar falsas alarmas, la señal **A** se activará cuando se cumplan las siguientes condiciones:

- Si la temperatura es mayor de 60°C siempre se activará la alarma
- Si la temperatura está entre 45°C y 60°C se activará la alarma sólo si han detectado gases o humos (o ambos).
- Si la temperatura es menor de 45°C se activará la alarma sólo si se detectan gases y humos

Resumiendo, las 4 señales binarias de entrada y la salida:

- **G**: vale '1' si se detecta **GAS** resultante de la combustión.
- **H**: vale '1' si se detecta **HUMO**.
- **T45**: vale '1' si la temperatura es superior a 45°C
- **T60**: vale '1' si la temperatura es superior a 60°C

La señal de salida **A** (alarma) se activará a nivel alto

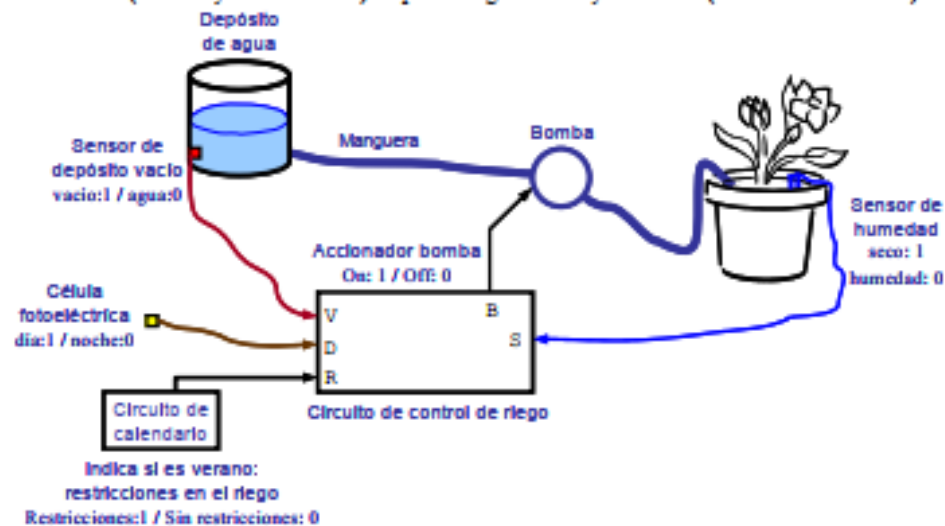
Se pide

- a) Realizar la tabla de verdad de la señal de alarma (**A**) a partir de las señales de entrada (**G**, **H**, **T45**, **T60**) . Explicarla brevemente.
- b) Obtener la expresión reducida en suma de productos y producto de sumas
- c) Dibujar el esquema en puertas de estas expresiones

Problema 2.24.

Se desea hacer un circuito de riego automático como el mostrado en la figura. El circuito deberá accionar la bomba en las siguientes condiciones

- El circuito accionará la bomba solamente cuando la tierra esté seca, pero antes debe comprobar las siguientes condiciones:
- Para evitar que la bomba se estropee por funcionar en vacío, nunca se accionará la bomba cuando el depósito de agua esté vacío.
- Si hay restricciones en el riego (época de verano), sólo se podrá regar de noche.
- En el resto del año (si no hay restricciones) se podrá regar de día y de noche (si la tierra está seca).



Para la implementación del circuito se dispone de las siguientes entradas:

- S: Señal que indica si la tierra está seca.
Tierra seca: S=1 ; Tierra húmeda: S=0
- R: Señal que indica si hay restricciones en el riego (es verano):
Hay restricciones: R=1 No hay restricciones: R=0
- D: Señal que indica si es de día o de noche:
Día: D=1 ; Noche: D=0
- V: Señal que indica si el depósito de agua está vacío:
Vacío: V=1 ; Hay agua: V=0

Y la salida B, que accionará la bomba para regar: *Bomba funcionando: B=1 ; Bomba apagada B=0*

Con esta información se debe:

- Elaborar la tabla de verdad del circuito
- Obtener la ecuación en la primera forma normal
- Hacer el mapa de Karnaugh
- Obtener la ecuación simplificada en suma de productos y productos de suma
- Representar ambas ecuaciones simplificadas (SOP y POS) en puertas lógicas

Unidad N° 3: Tecnología de las compuertas lógicas.

Contenido.

Contenidos: Estructura básica de un transistor MOS. Familia lógica CMOS. Características: especificaciones de tensión y de corriente de entrada y de salida, margen de ruido, retardo de propagación, consumo estático y dinámico. Familia TTL: Serie 74/54 Standard como referencia histórica. Interfase C-MOS/TTL y TTL/C-MOS. Puerta de transmisión CMOS. Lógica de tres estados. Tipos de salida. Valores flotantes en Verilog (Zs) y desconocidos o ilegales (Xs). Buffer.

Carga horaria por Unidad: 8hs

Resultado de aprendizaje.

RA4: Reconocer una forma de relacionar el valor continuo al valor discreto para definir los niveles lógicos y márgenes de ruido de los circuitos digitales considerando que las variables están representadas por cantidades físicas continuas tal como el voltaje en un cable...

Objetivo.

Que el alumno pueda resolver ejercicios de los temas vistos en la clase teórica.

Observación:

Se resolverán ejercicios que figuran con la numeración indicada al final de cada capítulo del libro propuesto por la cátedra Digital **Design Morris-Mano** más ejercicios adicionales que no son del libro propuesto.

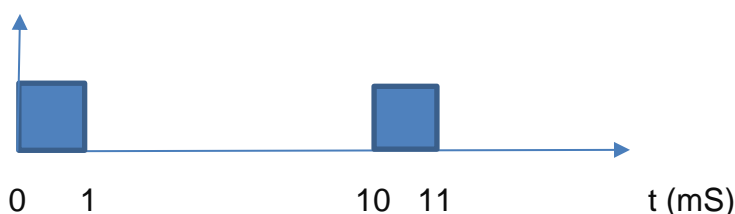
En la carpeta de TP de Aula deberán incluirse los ejercicios vistos y resueltos en clase, tanto los del libro como cualquier otro que se resuelva o se dé como consigna.

Problemas

Problemas en clases.

Problema 3.1.

Se muestra una parte de una señal digital periódica. Determine: a) periodo. b) frecuencia. c) Ciclo de trabajo.

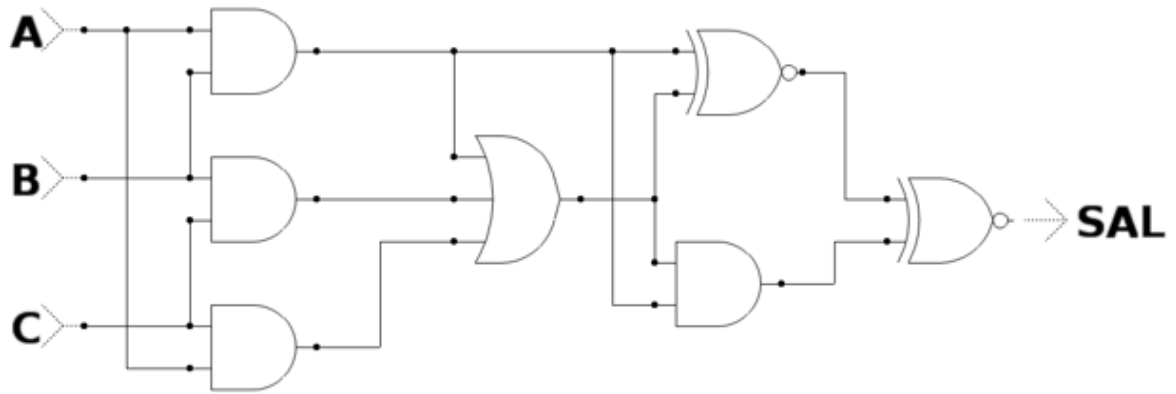


Problema 3.1.

Por una compuerta circulan $2\mu\text{A}$ cuando su salida está en nivel alto y $3,6\mu\text{A}$ cuando está en nivel bajo.Cuál es la disipación de potencia media para $V_{cc} = 5\text{V}$ y la compuerta funciona con un ciclo de trabajo del 50%.

Problema 3.2.

Suponga que la compuerta OR EXC tiene un retardo de propagación de 20 nS y que las compuertas AND y OR tienen un retardo de 10 nS. Calcule el retardo de propagación total del siguiente circuito.



Problema 3.3.

Un celular determinado tiene una batería de 6 W/H y opera a 1,2 V. Suponga que cuando está en uso el teléfono opera a 300 MHz. Y el valor promedio de la capacidad en conmutación del chip en un momento dado es de 10 nF. Cuando está en uso también emite 3W de potencia de su antena. Cuando no está en uso la potencia dinámica cae casi a 0, debido a que el procesamiento de la señal se apaga. Sin embargo el teléfono también drena 40mA de corriente de reposo si está en uso o no. Determine la vida de la batería del teléfono si: a) No está siendo Usado. b) Si está siendo usado continuamente.

Problema 3.4.

La hoja de datos de una compuerta 7400 indican los siguientes valores. Podrá indicar el Fan-Out de una compuerta?

$$I_{ih(max)} = 40 \mu\text{A}$$

$$I_{il(max)} = 1,6 \text{ mA}$$

$$I_{oh(max)} = 400 \mu\text{A}$$

$$I_{ol(max)} = 16 \text{ mA}$$

Problema 3.5.

Determine los márgenes de ruido de los niveles alto y bajo para TTL y CMOS. Utilizando los siguientes datos.

$$V_{dd} = 5\text{V}$$

$$V_{cc} = 5\text{V}$$

$$V_{ih(min)} = 3,5 \text{ V}$$

$$V_{ih(min)} = 2\text{V}$$

$$V_{il(max)} = 1,5 \text{ V}$$

$$V_{il(max)} = 0,8 \text{ V}$$

$$V_{oh(min)} = 4,4 \text{ V}$$

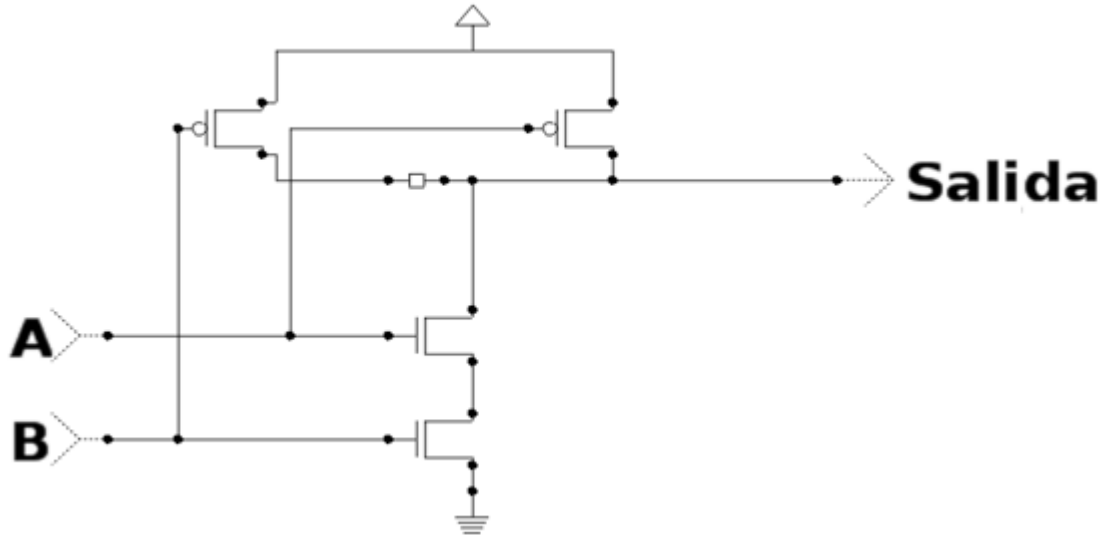
$$V_{ol(max)} = 0,33 \text{ V}$$

$$V_{oh(min)} = 2,4 \text{ V}$$

$$V_{ol(max)} = 0,4 \text{ V}$$

Problema 3.6.

Dado el siguiente circuito realice la tabla de verdad e indique que compuerta es.



Problema 3.7.

Implemente con transistores CMOS un inversor.

Problema 3.8.

Diseñe el circuito de una compuerta NAND de 4 entradas con transistores CMOS.

Problema 3.9.

Diseñe el circuito de una compuerta OR de 3 entradas con transistores CMOS.

Problema 3.10.

Las siguientes son especificaciones de compuertas NAND cuádruples de dos entradas Schottky TTL 74S00 . Calcule: a) el Fan-Out. b) La disipación de Potencia. c) el retardo de propagación. d) el margen de ruido.

$V_{cc} = 5 \text{ V}$		
$I_{cch} = 10 \text{ mA}$ (de 4 comp)	$V_{ih(min)} = 2\text{V}$	$I_{ih(max)} = 0,05\text{mA}$
$I_{ccl} = 20 \text{ mA}$ (de 4 comp)	$V_{il(max)} = 0,8\text{V}$	$I_{il(max)} = 2\text{mA}$
$V_{oh(min)} = 2,7\text{V}$	$I_{oh(max)} = 1\text{mA}$	$t_{plh} = 3 \text{ nS}$
$V_{ol(max)} = 0,5\text{V}$	$I_{ol(max)} = 20\text{mA}$	$t_{phl} = 3 \text{ nS}$

Problemas adicionales.**Problema 3.11.**

De las siguientes funciones, se pide realizar:

A) Implemente con transistores CMOS.

B) Realizar la simulación con el software en línea “Falstad”

C) Describir en Verilog, y verificar la simulación de la herramienta de Xilinx con el de Falstad”

$$f1 = \overline{A} \cdot B$$

$$f2 = \overline{A} + B$$

$$f2 = (A + \overline{B}) * C$$

$$f2 = (A \cdot \overline{B}) + (\overline{A} \cdot B)$$

Unidad N° 4: Circuitos combinacionales básicos.

Contenido.

Multiplexores, Demultiplexores, Decodificadores, Codificadores con y sin prioridad, Circuito conversor de binario a Gray y de Gray a binario, circuitos detectores y correctores de error: paridad, código de Hamming. Decodificador BCD a 7 segmentos. Desplazadores. Uso del decodificador y del multiplexor como generador de funciones. El operador ternario en Verilog para implementación de multiplexores. Modelado estructural. Diferencia entre una variable y una signal. Código concurrente versus secuencial.

Carga horaria por Unidad: 16hs

Resultado de aprendizaje.

RA2: Emplear las herramientas de diseño adecuadas para la resolución de problemas a partir de una consigna dada.

RA5: Explicar la especificación funcional de un circuito combinacional a fin de enfatizar la función del bloque constructivo considerando que la lógica combinatoria a menudo se agrupa en bloques constructivos más grandes para construir sistemas más complejos.

RA7: Utilizar un lenguaje de descripción de hardware como Verilog con el propósito de implementar diferentes configuraciones de hardware utilizando dispositivos lógicos programables

Objetivo.

Que el alumno pueda resolver ejercicios de los temas vistos en la clase teórica.

Observación:

Se resolverán ejercicios que figuran con la numeración indicada al final de cada capítulo del libro propuesto por la cátedra Digital **Design Morris-Mano** más ejercicios adicionales que no son del libro propuesto.

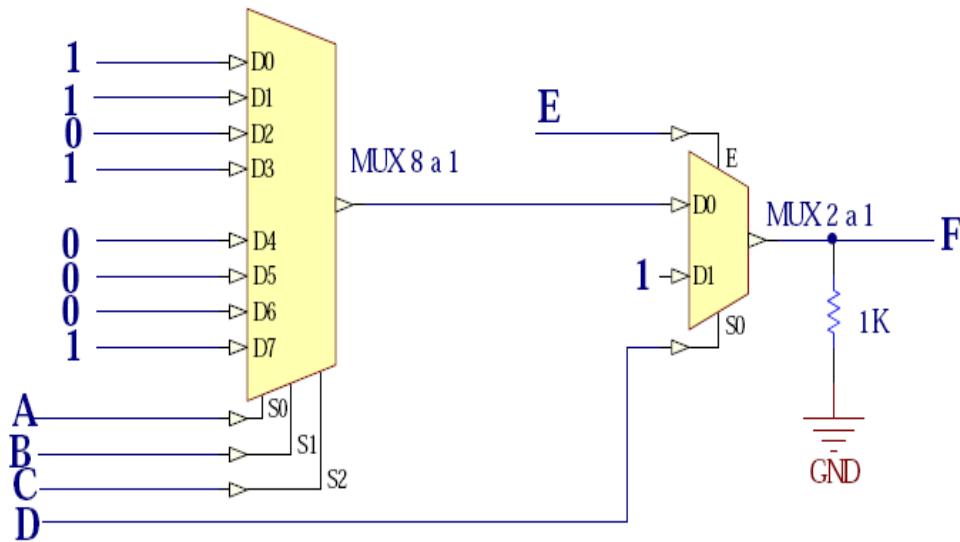
En la carpeta de TP de Aula deberán incluirse los ejercicios vistos y resueltos en clase, tanto los del libro como cualquier otro que se resuelva o se dé como consigna.

Problemas

Problemas en clases.

Problema 4.1.

Determine las funciones lógicas correspondiente al siguiente esquemas con multiplexores. Exprese la función como suma de mini términos.

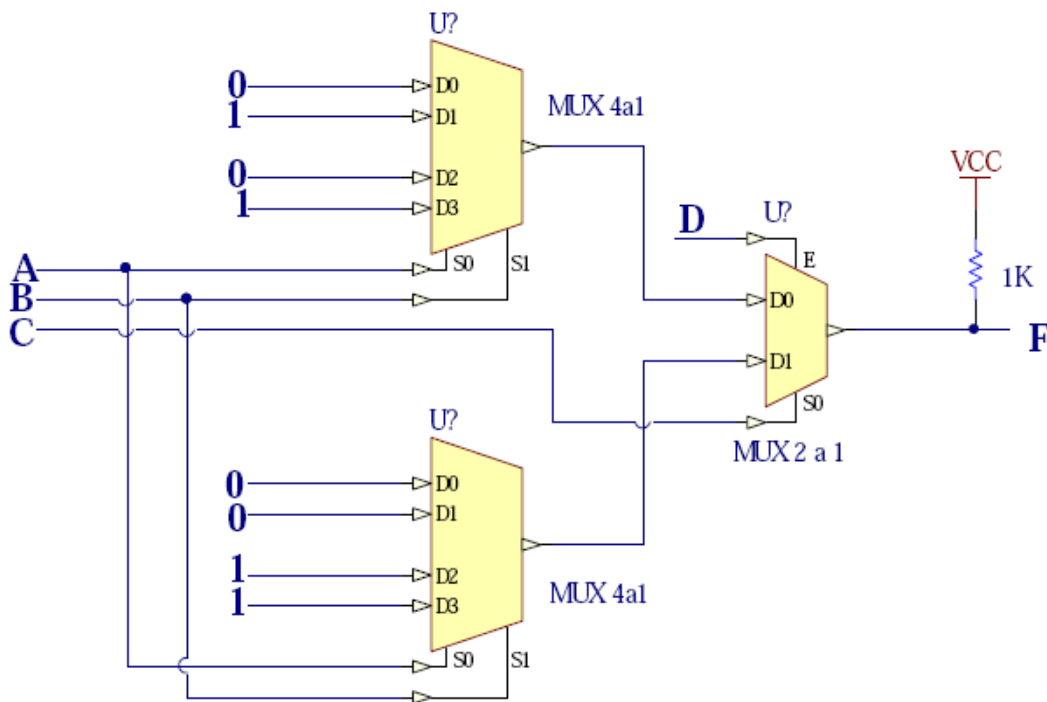


Resultado:

<http://falstad.com/circuit/circuitjs.html?ctz=CQAgjCAMB0I3BWcMBMcUHYMGZIA4UA2ATmxAUgpABZsKBTAWjD ACgw8aRjCe+wNbryghsbAEpiUIPFWwyaeUVRpUkVTdARsA5rPmFlc2nBVsa7v2sLrkS9JsyT9qyLApljtj8teOfjj+7sQy3qFQDu FhMgHBfAryMZpRyf4RfibYRgYgnr4OJvnWxZIUxUUIfGV5VSV1NSh1RSjca0lHe3cJmCEKVa9-dZ9KQAYokLKTEq0s5ogAGYAhgA2AM4M1PYTqoTcM8o0+ypLa5vbbLu0Joc3C1QrG1saDqqzo3MFVp9T4MPYLiRKx7bhgD B8Y5tBzgyEnMARKHA8Cl+ERQHqQwoDoYvldJQDPHcAmTFwclHZZQiP7U8CSaRUWEMvL4U5qbYqbQOJp8ESUuyPRkQ5 lxQXgTxihKiiZ9CgeEB3BDyhZPC6vH58JWKWxa5F0GS6-UomQ7MQIGSteTmvJ4XyiVUvZHYa2W0zldl+NTu1RmBEmtg0BE2o6y3GBvhoWbiLG2+5EuN+Yp-JMnROxpF+2ip7np2U82iyxN1cO1SHzbnFj7DP72IA>

Problema 4.2.

Determine las funciones lógicas correspondiente al siguiente esquemas con multiplexores. Exprese la función como suma de mini términos.



Problema 4.3.

Resolver las siguientes funciones lógicas con dos multiplexores de 4:1. Junto a cada función a resolver se encuentra la referencia a la combinación de

multiplexores a utilizar. Con el tipo de MUX ver la tabla de verdad para la combinación de la salida.

- a) $f_{a(ABC)} = \sum 0, 2, 3, 5, 7$ [Mux1,Mux1], [Mux2,Mux2], [Mux3,Mux3]
 b) $f_{b(ABC)} = \sum 1, 3, 4, 6, 7$ [Mux2,Mux2], [Mux3,Mux3], [Mux1,Mux2]
 c) $f_{c(ABC)} = \sum 0, 2, 4, 5, 6, 7$ [Mux3,Mux3], [Mux1,Mux2], [Mux3,Mux1]
 d) $f_{d(ABCD)} = \sum 1, 3, 4, 7, 10, 12, 14$ [Mux3,Mux1], [Mux3,Mux2], [Mux1,Mux2]
 e) $f_{d(ABCD)} = \sum 2, 4, 8, 7, 10, 12, 14, 15$ [Mux1,Mux1], [Mux3,Mux3], [Mux2,Mux2]

Mux1			
CE	A	B	out
1	x	x	1
0	0	0	In0
0	0	1	In1
0	1	0	In2
0	1	1	In3

Mux2			
CE	A	B	out
0	x	x	0
1	0	0	In0
1	0	1	In1
1	1	0	In2
1	1	1	In3

Mux3			
CE	A	B	out
1	x	x	Z
0	0	0	In0
0	0	1	In1
0	1	0	In2
0	1	1	In3

Problema 4.4.

Implementar las siguientes funciones utilizando un multiplexor de 8 a 1.

- a) $f_{a(ABCD)} = \prod 0, 2, 3, 5, 7, 12, 10, 14$
 b) $f_{b(ABCD)} = \sum 1, 3, 4, 6, 7$
 c) $f_{c(ABCD)} = \prod 0, 2, 4, 5, 6, 7, 10, 11, 12, 15$
 d) $f_{d(ABCD)} = \sum 1, 2, 3, 4, 8, 9, 10, 12, 14$
 e) $f_{e(ABCD)} = \prod 0, 2, 3, 5, 7, 8, 9, 12, 14, 15$
 f) $f_{f(ABCD)} = \sum 1, 3, 4, 6, 7$
 g) $f_{g(ABCD)} = \prod 0, 2, 4, 5, 6, 7, 14, 15$
 h) $f_{h(ABCD)} = \sum 0, 1, 4, 7, 10, 12, 14$

Problema 4.5.

Implementar las siguientes funciones utilizando un multiplexor de 4 a 1.

- a) $fa_{(ABCD)} = \sum 0, 2, 3, 5, 7, 12, 10, 14$
 b) $fb_{(ABCD)} = \sum 1, 3, 4, 6, 7$
 c) $fc_{(ABCD)} = \sum 0, 2, 4, 5, 6, 7, 10, 11, 12, 15$
 d) $fd_{(ABCD)} = \sum 1, 2, 3, 4, 8, 9, 10, 12, 14$
 e) $fe_{(ABCD)} = \sum 0, 2, 3, 5, 7, 8, 9, 12, 14, 15$
 f) $ff_{(ABCD)} = \prod 1, 3, 4, 6, 7$

Problema 4.6.

Diseñar un decodificador BCD a 7 segmentos utilizando 1 multiplexor de 4 a 1 por cada segmento, tipo CD4052. En caso de que se presente a la entrada un dato erróneo inhibir la salida utilizando la señal de selección del chip.

Problema 4.7.

Describir un multiplexor 8:1 y otro de 4:1 con control de salida para tercer estado en VHDL utilizando "WHEN ELSE", utilizar una señal de habilitación que deje la salida en '0' cuando la misma es '1'.

*<name> <= <expression> when <condition> else
 <expression> when <condition> else
 <expression>;*

Problema 4.8.

Obtener el código de Hamming correspondiente a las siguientes palabras binarias y expresar la palabra completa a transmitir.

A) 10110110	B) 10101010	C) 10100101
D) 1101011000	E) 0010110011	F) 10110111

Problema 4.9.

Determinar si las siguientes palabras codificadas en Hamming contienen algún error y corregirlas. El código responde a Hamming ponderado, siendo el bit de la derecha K1. Puede usar cualquiera de los métodos desarrollados en clase.

A) 10101010011	B) 01001111001	C) 10010001110
D) 11001111101	E) 01111010100	F) 000000000111

Problema 4.10.

Describir un codificador y un decodificador para un código de Hamming de 7 bits de datos en VHDL, el decodificador debe corregir la palabra y mostrar en un display de 7 segmentos la ubicación del error.

Problemas adicionales.

Problema 4.A.1.

Implementar el decodificador de errores para un código de Hamming de 9 bits de información utilizando decodificadores de 3 a 8.

Problema 4.A.2.

Obtenga la palabra de código Hamming de 15 bits para la palabra de datos 11001001010 de 11 bits.

Problema 4.A.3.

Diseñar un sistema combinacional con dos entradas A y B de 8 bits cada una, las cuales representan un número entero sin signo, y una entrada de control extra MIN/MAX. La salida del sistema Z también es de 8 bits. La salida $Z=0$ si $A=B$,

$Z=\min(A,B)$ si $\text{MIN/MAX}=1$, $Z=\max(A,B)$ si $\text{MIN/MAX}=0$.

- Diseñar el sistema utilizando multiplexores.
- Describir el sistema en VHDL.
- Adjuntar la descripción.
- Adjuntar la pantalla de simulación.

Problema 4.A.4.

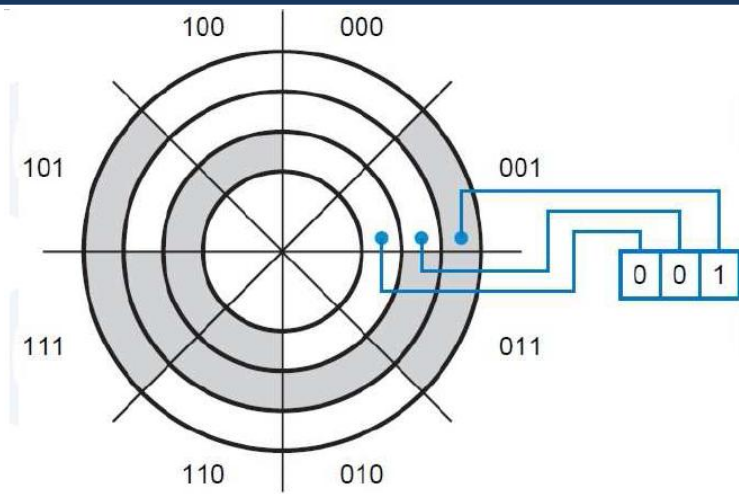
Desarrollar un circuito para convertir binario de 8 BIT en código Gray y viceversa.

- Dibujar la implementación con compuertas.
- Describir los circuitos en VHDL.

Problema 4.A.5.

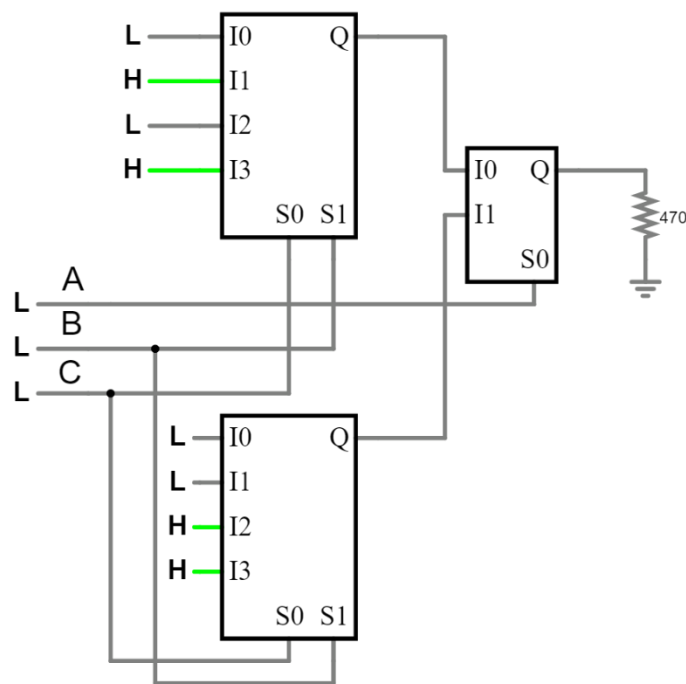
El siguiente grafico representa un encoder óptico, utilizado para determinar la posición del eje de un motor según la combinación binaria que presenta a su salida. Se considera por ejemplo 000 -> 0° y 011 -> 90° etc.

- Desarrollar la tabla de verdad que relacione los ángulos de rotación con el código de gray correspondiente.
- Implementar un decodificar que permita leer los ángulos de rotación en tres display de 7 segmentos.



Problemas resueltos.

Problema 4.x.



Repuesta: $\sum_{i=0}^7 m1, m3, m6m7$

Link

<https://www.falstad.com/circuit/circuitjs.html?ctz=CQAgjCAMB0l3BWcMBMcUHYMGZIA4UA2ATmIxAUgoqoQFMBaMMAKACcQAWPKsTzrxBo8ULhkgsA5hQyEKhWnNnyqkmdyojBWIALUsweAWEWDhe81sPHwZ7NnkpLD1cJsDX4flzjf94CwAMiBCmCgg2BHhYlQAZgCGADYAznTUKiFCuJ4RObEgianpSJmhWpxUUcKVBuVpGSWAHiCEECi4IFjCDlw1IACCzRSiKGRdSLhUApYAwsNmRBHKREgzAgBCLADuwnii3JGOVjtH8odefPqnl8peXpK798co+2dQpygl58RO3xRgCKPCwCTi-EDgsGqU7gvRUMK1YGw2qVCoGXaQ8Hg-Jl+T5bKQa4YvGEkGRUnA5yeUIU4T4D67MIYCKA0bMhngV7CdmsgFA068hCA4T-IX83a8mJfJzsyn-KXy+nAzjVK46fwfEKmaaiZgRQ5qQrJBqlYJ2eG8YVCXhG4qNLVmPXgYVOw31Eq a80ap1qm3u+1ey4+B5if2mh1VQi64NRurGj1lbW+Xg+VFiCBhz1JIUs1PVP3xxpAA>

Unidad N° 5: Circuitos aritméticos.

Contenido.

Concepto de complemento a la base y a la base-1. Operaciones aritméticas de números en CA2 (suma, resta, multiplicación, división). Implementación de circuitos aritméticos. Sumador completo de 1 bit, Sumador por propagación de acarreo. Sumadores rápidos. Circuitos Detectores de todos unos o ceros. Circuitos comparadores de magnitud y de igualdad. Evaluación de funciones matemáticas mediante tablas. Operaciones con un operando constante.

Resultado de aprendizaje.

RA2: Emplear las herramientas de diseño adecuadas para la resolución de problemas a partir de una consigna dada.

RA5: Explicar la especificación funcional de un circuito combinacional a fin de enfatizar la función del bloque constructivo considerando que la lógica combinatoria a menudo se agrupa en bloques constructivos más grandes para construir sistemas más complejos.

RA7: Utilizar un lenguaje de descripción de hardware como Verilog con el propósito de implementar diferentes configuraciones de hardware utilizando dispositivos lógicos programables.

Objetivo.

Que el alumno pueda resolver ejercicios de los temas vistos en la clase teórica.

Observación:

Se resolverán ejercicios que figuran con la numeración indicada al final de cada capítulo del libro propuesto por la cátedra Digital **Design Morris-Mano** más ejercicios adicionales que no son del libro propuesto.

En la carpeta de TP de Aula deberán incluirse los ejercicios vistos y resueltos en clase, tanto los del libro como cualquier otro que se resuelva o se dé como consigna.

Problemas

Problemas en clases.

Problema 5.1.

Realizar la resta de los números binarios sin signo usando el complemento a dos en el sustraendo, donde el resultado sea negativo, encontrar el complemento a dos y colocar el signo menos.

(a) $10011 - 10010$ (b) $100010 - 100110$

(c) $1001 - 110101$ (d) $101000 - 10101$

Problema 5.2.

Problemas adicionales.**Problema 5.A.1.**

Encontrar el complemento a 9 y complemento a 10 de los siguientes números:

(a) 25, 478,036 (b) 63, 325, 600

(c) 25, 000,000 (d) 00, 000,000.

Problema 5.A.2.

Decodificar el siguiente código ASCII:

1010011 1110100 1100101 1110110 1100101 0100000 1001010 1101111
11000101110011

Problema 5.A.3.

La siguiente es una cadena de caracteres ASCII cuyos patrones de bits han sido convertidos a hexadecimal: 73 F4 E5 76 E5 4A EF 62 73. De los ocho bits en cada par de dígitos, el MSB (bit más significativo) es el bit de paridad. Los restantes bits corresponden al carácter ASCII.

- a) Convertir la cadena a la forma de bits y decodificar el ASCII.
- b) Determinar la paridad usada: par o impar?

Unidad N° 6. Lógica secuencial síncrona

Contenido.

Sistemas secuenciales. Introducción de los conceptos de memoria y tiempo. Autómata de Mealy y de Moore. Latches SR y D. Flip-flop maestro-esclavo. Flip-flop activado por flanco. Procedimiento de análisis. Diagrama de estados. Asignación de estados. Diseño con flip-flop D. Reducción de estados. Descripción de máquinas de estado en Verilog, simulación. Problemas de aplicación.

Resultado de aprendizaje.

RA6: Expresar el procedimiento formal para el análisis y el diseño de circuitos secuenciales síncronos considerando la diferencia entre disparo por nivel y por flanco y el modelado comportamental en Verilog para los circuitos secuenciales.

RA7: Utilizar un lenguaje de descripción de hardware como Verilog con el propósito de implementar diferentes configuraciones de hardware utilizando dispositivos lógicos programables.

Objetivo.

Que el alumno pueda resolver ejercicios de los temas vistos en la clase teórica.

Observación:

Se resolverán ejercicios que figuran con la numeración indicada al final de cada capítulo del libro propuesto por la cátedra Digital **Design Morris-Mano** más ejercicios adicionales que no son del libro propuesto.

En la carpeta de TP de Aula deberán incluirse los ejercicios vistos y resueltos en clase, tanto los del libro como cualquier otro que se resuelva o se dé como consigna.

Problemas

Problemas en clases.

Problema 6.1

Construir un Flip Flop JK utilizando un Flip Flop D, un multiplexor de 2 a 1 y lógica combinacional.

Problema 6.2

Mostrar que la ecuación característica para la salida complementada de un Flip Flop JK es:

$$Q' (t+1) = J'.Q' + K.Q$$

Mostrar que la ecuación característica para la salida de un Flip Flop JK es:

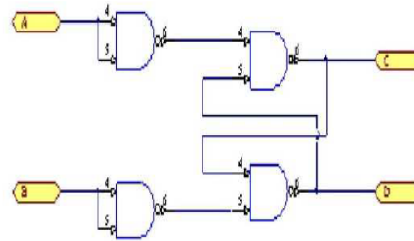
$$Q (t+1) = J.Q' + K'.Q$$

Problema 6.3

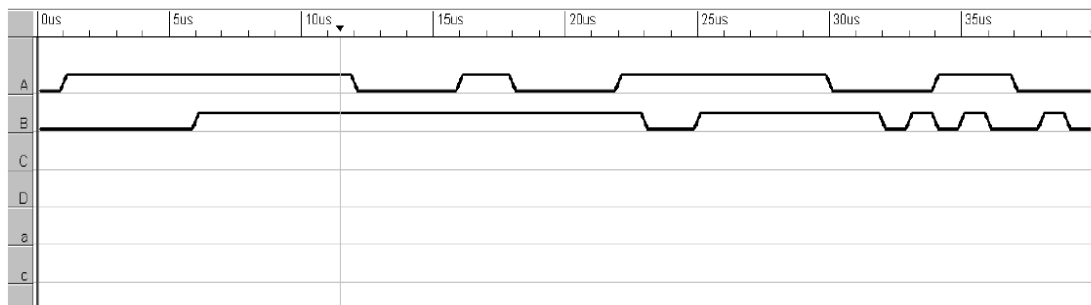
Para el circuito de la figura completar el diagrama temporal.

Determinar que función cumple A y B.

W

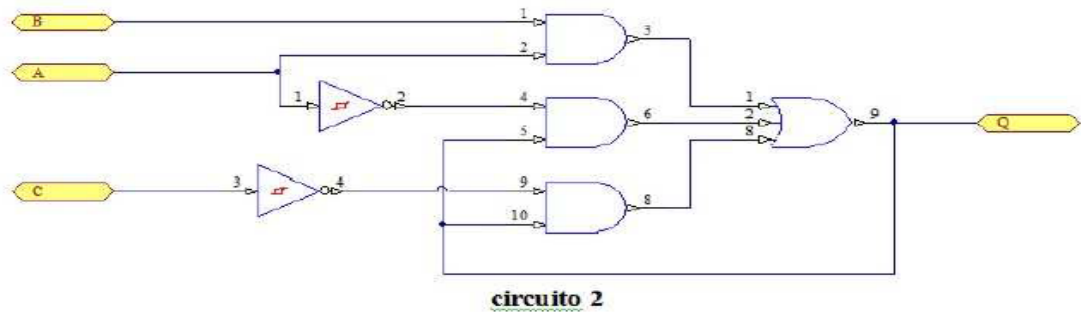


circuito 1



Problema 6.4

Completar la tabla de verdad del siguiente circuito, y determinar a qué tipo de biestable corresponde



A	B	C	Q(T)	Q(T+1)
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	0	
1	0	0	0	
1	0	1	0	
1	1	0	0	
1	1	1	0	
0	0	0	1	
0	0	1	1	
0	1	0	1	
0	1	1	1	
1	0	0	1	
1	0	1	1	
1	1	0	1	
1	1	1	1	

Problema 6.5

Describir en Verilog los siguientes Flip-Flop, compilar y simular. Pegar la descripción y la simulación de:

- J - K con reset asíncrono
- J - K con reset síncrono
- D con reset asíncrono LATH activado por nivel '1'

Problema 6.6

Diseñar un contador síncrono módulo 7 utilizando Flip Flop JK. Condición inicial 111.

Problema 6.7

Diseñar un contador de código Gray de 4 bits utilizando Flip-Flop del tipo D. Obtener una salida auxiliar que presente el código BCD.

Problema 6.8

Diseñe un contador creciente $M=5$

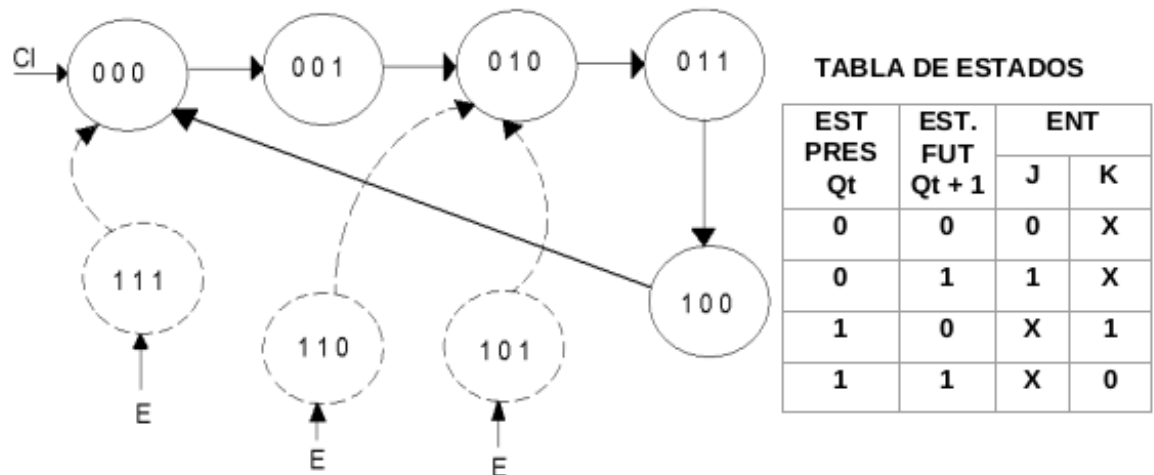
MODULO: N DE ESTADOS POR LOS QUE ATRAVIESA EL CONTADOR

n = CANTIDAD DE FLIP-FLOP NECESARIOS PARA EL DISEÑO

1) CANTIDAD DE FLIP-FLOP $n = 3$

2) DIAGRAMA DE ESTADOS

$2^{\exp(n)} > M$

**Problema 6.9**

Diseñar un circuito secuencial con dos Flip Flop D, A y B y una entrada x_{in} .

Cuando $x_{in} = 0$, el estado del circuito se mantiene igual. Cuando $x_{in} = 1$ el circuito transiciona del estado 00 al estado 01, al 11 al 10 al 00 y se repite el ciclo.

Problema 6.10

Diseñar un circuito secuencial que implemente la siguiente salida

000-101-111-010-110

000=Condición inicial.

En caso de detectar un estado no contemplado en la secuencia el sistema debe retornar inmediatamente a la condición inicial.

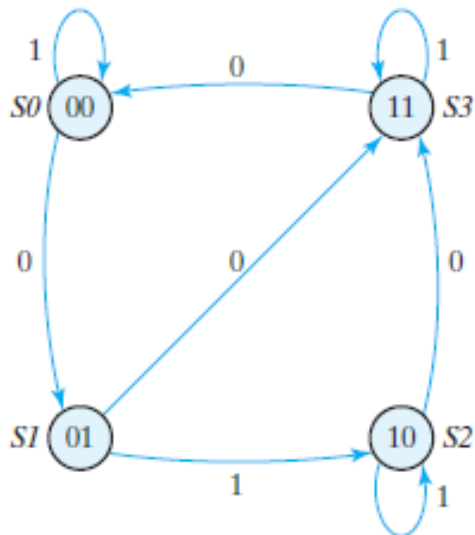
Problema 6.11

Realizar el ejercicio 6.6 en HDL y simularlo en un nivel post síntesis.

Problema 6.12

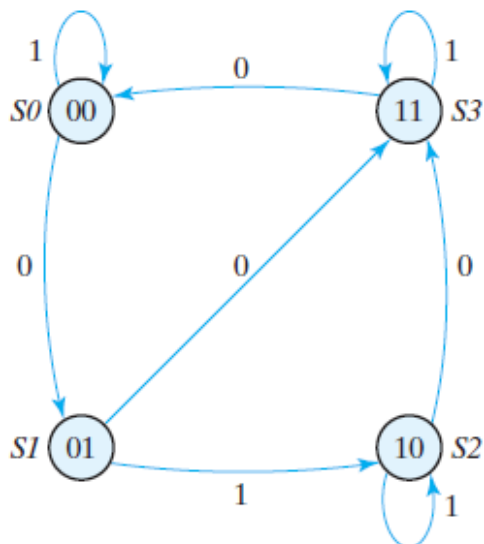
Un circuito secuencial con 2 FF y una entrada x_{in} y una salida y_{out} . Se pide hacer el diagrama de estado se muestra en la figura. El circuito se diseñará tomando los estados no usados como condiciones no importa. La salida deber ser “1” para el estado S1 y S2. Realizar los puntos a y b.

- a) Flip Flop D
- b) Flip Flop JK



Problema 6.14

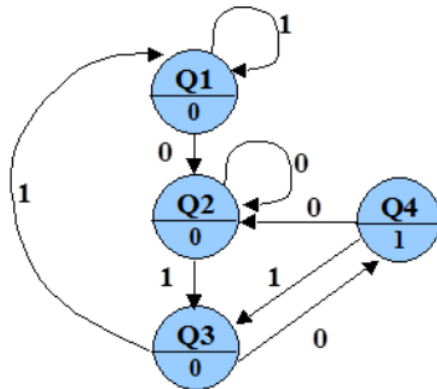
Realizar la descripción comportamental de la máquina de estado representada en la figura. Escribir el Testbench y verificar la funcionalidad de la descripción.



Problema 6.15

El diagrama de estado correspondiente al detector de secuencias bajo un modelo Moore es el representado en la figura. Realizar las tablas de transición; salida y realizarlo con FFD. Realizar la descripción y simulación en Verilog.

Ver: https://dsd.webs.upv.es/?page_id=630



Problema 6.16

Una puerta se abre al activar un pulsador P. La apertura se produce hasta que alcanza el tope de apertura detectado por el sensor A. A partir de ese momento se produce el cierre de la puerta, hasta alcanzar el tope de cierre detectado por el sensor C, y en momento se produce la parada.

Diseñar el circuito correspondiente. La puerta se controla por dos salidas, S1 Y S0 cuando S1=1 la puerta se abre, cuando S0=1 la puerta se cierra. Diseñar el autómata, dibujar el circuito y describir en verilog

Problemas adicionales.**Problema 6.A.1**

Diseñar un contador módulo 12 utilizando un registro de desplazamiento y lógica de realimentación. Desarrollar la lógica de decodificación necesaria para dar una salida decimal.

Problema 6.A.2

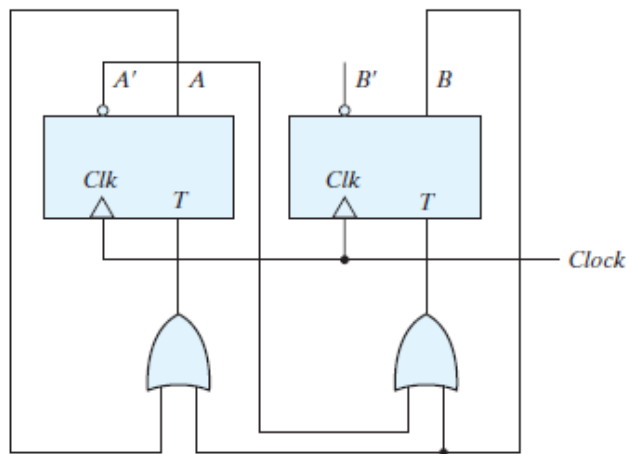
Diseñar Utilizando un registro de desplazamiento y lógica combinatoria, diseñar un generador de secuencia que genere la secuencia binaria 0-1-0-0-1-0-1-1.

Problema 6.A.3

Describirlo en HDL el problema 6.A.2

Problema 6.A.4

Derivar la tabla de estado y el diagrama de estado del circuito secuencial mostrado en la figura, explicar la función que el circuito realiza.

**Problema 6.A.5**

Diseñar con FFD para que generen las siguientes secuencias:

- a). 00 – 11- 10 ; b). 000-101-110-111-001

Problema 6.A.6

Diseñar con registro de desplazamiento sistemas que generen las siguientes secuencias:

- a). 00 – 11- 10 ; b). 000-101-110-111-001
c). 010-111-010-101-100-110-111

Problema 6.A.7

Diseñar un circuito secuencial con dos Flip Flop D, A y B y una entrada x_{in} .

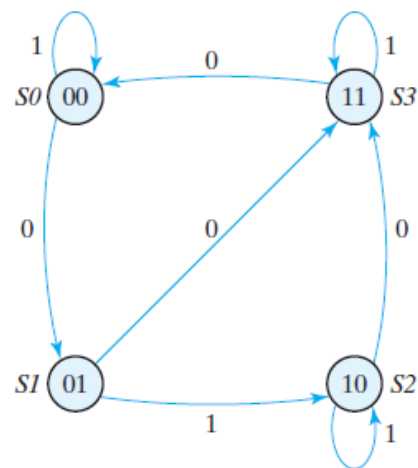
Cuando $x_{in} = 0$, el estado del circuito se mantiene igual. Cuando $x_{in} = 1$ el circuito transiciona del estado 00 al estado 11, al 00 al 10, al 01 y se repite el ciclo.

Problema 6.A.8

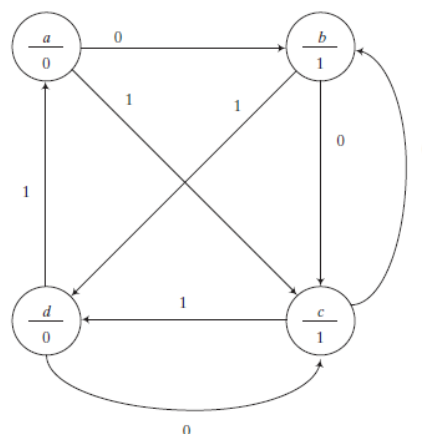
Explicar las diferencias entre Tabla de Verdad, Tabla de Estado, Tabla Característica y Tabla de Excitación. Explicar la diferencia entre una ecuación booleana, una ecuación de estado, una ecuación característica y la ecuación de entrada de un FF

Problema 6.A.9

Realizar la descripción comportamental de la máquina de estado representada en la figura. Escribir el Testbench y verificar la funcionalidad de la descripción.

**Problema 6.A.10**

Escribir el modelo en Verilog de la FSM de Moore descrita en el siguiente diagrama de estados. Realizar el Testbench y mostrar los gráficos.

**Problema 6.A.11**

Realizar en Verilog la descripción comportamental de un flip flop JK utilizando la sentencia IF-ELSE basada en el valor del estado presente.

Obtener la ecuación característica cuando $Q=0$ o $Q=1$.

Explicar cómo las entradas J y K afectan la salida del flip flop con cada pulso de reloj.

Problema 6.A.12

Un limpiaparabrisas tiene pulsadores de marcha y parada activa, si se pulsa P se para cuando se detecta el sensor de reposo R en uno. Esto hace que el sistema pare siempre en el mismo lugar. Diseñar el autómata de Mealy que resuelva el problema, dibujar el circuito y describir en Verilog. *Nota: se recomienda realizarlo como máquina de Moore y de Mealy. Resolverlos con FFD y FFJK. Simularlo con falstad*

Problema 6.A.13

Hay que controlar las puertas de cristal de un edificio. Disponemos de un detector de personas P, de un detector de puerta abierta A y de un detector de puerta cerrada C. Si viene una persona se abre la puerta. Si la puerta empieza a abrirse, completa su ciclo entero aunque desaparezca la persona. Si sigue apareciendo gente la puerta permanece abierta continuamente. Si cuando estaba cerrándose la puerta aparece una persona, la puerta vuelve a abrirse, completando un nuevo ciclo. Diseñar el autómata, dibujar el circuito y describir en Verilog. *Nota: se recomienda realizarlo como máquina de Moore y de Mealy. Resolverlos con FFD y FFJK. Simularlo con falstad.*

Problema 6.A.14

En un desvío de trenes se desea automatizar el cambio de vía. Si el tren es largo debe ir por la vía B ($B=1$) y si el tren es corto por la vía A ($A=1$). Para saber si el tren es largo o corto, se dispone de dos detectores ópticos E1 y E0 (E1 es el detector de la entrada y E0 es el detector de salida), separados entre sí 50 metros. Si en algún momento los dos detectores están en ON el tren es largo, siendo corto en caso contrario. Mientras se mide un tren no puede entrar otro a la zona de medición, el desvío se encuentra suficientemente lejos para que todo funcione bien, es decir el tren que se midió pueda acceder a la vía correcta. Diseñar el autómata, dibujar el circuito y describir en Verilog. Nota: se recomienda realizarlo como máquina de Moore y de Mealy. Resolverlos con FFD y FFJK. Simularlo con falstad.

Unidad N° 7. Registros y contadores

Contenido.

Introducción. Registros. Registro con carga en paralelo. Registro de desplazamiento. Registro de desplazamiento con carga en paralelo. Registro de desplazamiento bidireccional. Estudio de los registros de acuerdo a su capacidad de almacenamiento: registros individuales y bancos de registros. Contadores binarios sincrónicos up/down. Contadores basados en registros: Anillo, Johnson y LFSR (Linear Feedback Shift Register). Descripción en Verilog.

Resultado de aprendizaje.

. RA6:.Expresar el procedimiento formal para el análisis y el diseño de circuitos secuenciales síncronos considerando la diferencia entre disparo por nivel y por flanco y el modelado comportamental en Verilog para los circuitos secuenciales..

RA7: Utilizar un lenguaje de descripción de hardware como Verilog con el propósito de implementar diferentes configuraciones de hardware utilizando dispositivos lógicos programables.

Objetivo.

Que el alumno pueda resolver ejercicios de los temas vistos en la clase teórica.

Observación:

Se resolverán ejercicios que figuran con la numeración indicada al final de cada capítulo del libro propuesto por la cátedra Digital **Design Morris-Mano** más ejercicios adicionales que no son del libro propuesto.

En la carpeta de TP de Aula deberán incluirse los ejercicios vistos y resueltos en clase, tanto los del libro como cualquier otro que se resuelva o se dé como consigna.

Problemas

Problemas en clases.

Problema 7.1

Diseñar un contador síncrono módulo 7 utilizando Flip Flop JK. Condición inicial 101

Problema 7.2

Diseñar un dispositivo Flip-Flop del tipo D. Obtener una salida BCD la cuenta 0,2,4,6 cuando la entrada de control $x=0$ y 1,3,5,7 cuando $x=1$.

Problema 7.3

Describirlo en HDL el problema 6.1

Problema 7.4

Describirlo en HDL el problema 6.2

Problema 7.5

Diseñar un contador síncrono binario de cuatro bits con FF D.

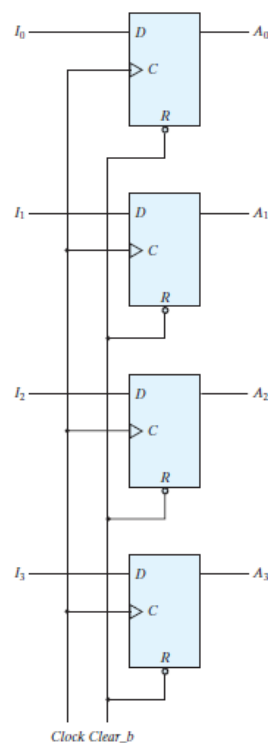
Problema 7.6

Usando FF JK :

- Diseñar un contador con la siguiente secuencia binaria que se repite: 0,1,2,3,4,5,6.
- Dibujar el diagrama lógico del contador.

Problema 7.7

Describir y verificar el HDL comportamental y estructural del siguiente registro de 4 bits.



Problemas adicionales.

Problema 7.A.1

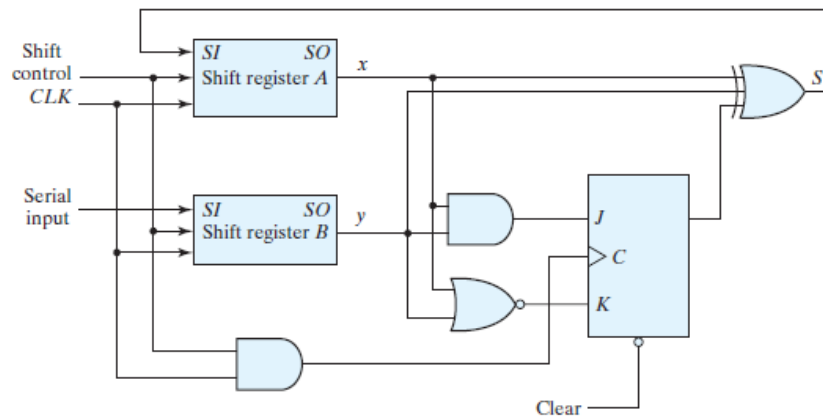
Diseñar un contador up-down módulo 6 que cuente en binario natural. Implementar con CD4013. Condición inicial 100. Detecte cuando el mismo entre en un estado no permitido y encienda un led en forma permanente.

Problema 7.A.2

Describir y simular los circuitos del ejercicio 7.A.1 en HDL.

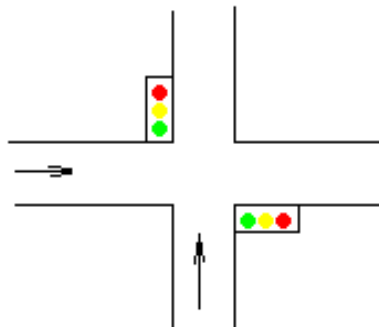
Problema 7.A.3

El sumador en serie de la siguiente figura usa dos registros de cuatro bits. El registro A contiene el número binario 0101 y el registro B contiene 0111. El flip-flop de acarreo se reinicia inicialmente a 0. Enumere los valores binarios en el registro A y el flip-flop de acarreo después de cada desplazamiento.



Problema 7.A.4

Diseñe una máquina de estado tipo Moore que implemente los semáforos del siguiente esquema. Considere que tiene una señal X que cuando vale '1', los semáforos pasan a intermitencia.



Unidad N° xx: Xx.

Contenido.

Resultado de aprendizaje.

Objetivo.

Que el alumno pueda resolver ejercicios de los temas vistos en la clase teórica.

Observación:

Se resolverán ejercicios que figuran con la numeración indicada al final de cada capítulo del libro propuesto por la cátedra Digital **Design Morris-Mano** más ejercicios adicionales que no son del libro propuesto.

En la carpeta de TP de Aula deberán incluirse los ejercicios vistos y resueltos en clase, tanto los del libro como cualquier otro que se resuelva o se dé como consigna.

Problemas

Problemas en clases.

Problemas adicionales.

Bibliografía

M.Morris Mano & Michael D. Ciletti *DIGITAL DESIGN With an Introductio to de Verilog HDL – Fifth Edition-* Person.



Atribución-NoComercial-SinDerivadas

Se permite descargar esta obra y compartirla, siempre y cuando no sea modificado y/o alterase su contenido, ni se comercializase. Referenciarlo de la siguiente manera:

Universidad Tecnológica Nacional Regional Córdoba. Material para la materia “Técnicas digitales I. Córdoba, Argentina.