

# Práctica 4

## Sistemas Digitales

### Objetivos particulares

Durante el desarrollo de esta práctica, el estudiante aplicará un método para obtener las diferentes representaciones de los sistemas digitales binarios, tales como: Función Booleana, Tabla de Verdad, Diagrama Esquemático y la Simulación en un programa de aplicación.

También implementará en un prototipo las ecuaciones propuestas comprobando los resultados.

### Elementos de competencia.

Obtener los circuitos partiendo de las ecuaciones F1 SOP (And/Or) y F2 POS (Or/And) proporcionadas por el instructor. **Ecuación → Circuito**

Obtener la tabla de verdad para F1 y F2 por el método analítico explicado en este capítulo. **Circuito → Tabla de verdad**

Crear un nuevo proyecto utilizando el programa de ispLEVER Starter o el ispLEVER Clasic. **Circuito → Captura esquemática**

Crear una nueva fuente (New Source) implementando las funciones F1 y F2, por medio de la captura esquemática (Schematic).

Comprobar la tabla de verdad por medio de la creación de una nueva fuente ABEL Test Vector para obtener la simulación de las funciones F1 y F2. **Captura esquemática → Simulación**

Obtener los archivos de reporte (RPT) y Programación (JED) del PLD. **Captura esquemática → JED y RPT**

Programar el PLD utilizando el archivo JEDEC generado anteriormente. **JED → Dispositivo**

Programar el dispositivo GAL16V8, Gal20V10, GAL22V10 o GAL26V12, utilizando el programador universal y el archivo JEDEC generado.

Utilizando el dispositivo programado, construir un prototipo efectuando las conexiones físicas necesarias en una tablilla de conexiones, para generar las señales de entrada y salida del PLD y comprobar físicamente las tablas de verdad de las funciones F1 y F2. **Dispositivo→Prototipo**

Comunicar el procedimiento y los resultados obtenidos por medio de un reporte escrito. **RPT→Reporte**

Para el procedimiento de la práctica e instalación del software se recomienda ver los videos de la página <http://jagarza.fime.uanl.mx/Agosto2012/Videos.htm>

## **Fundamento Teórico.**

### **Sistema digital binario**

Un sistema digital binario es un conjunto de dispositivos destinados a la generación, transmisión, procesamiento o almacenamiento de señales digitales. Para el análisis, síntesis e implementación de los circuitos digitales, se utiliza como herramienta el álgebra de Boole con sus operadores lógicos (AND, OR y NOT).

Los sistemas digitales se pueden clasificar en:

**Sistemas digitales combinacionales:** Aquellos cuyas salidas solo dependen del estado de sus entradas en un momento dado. Por lo tanto, no necesitan módulos de memoria, ya que las salidas no dependen de los estados previos de las entradas.

**Sistemas digitales secuenciales:** Aquellos cuyas salidas dependen además del estado de sus entradas en un momento dado, de estados previos. Esta clase de sistemas necesitan elementos de memoria que recojan la información de la 'historia pasada' del sistema.

Los elementos con los que contamos para describir un Sistema Digital binario son:

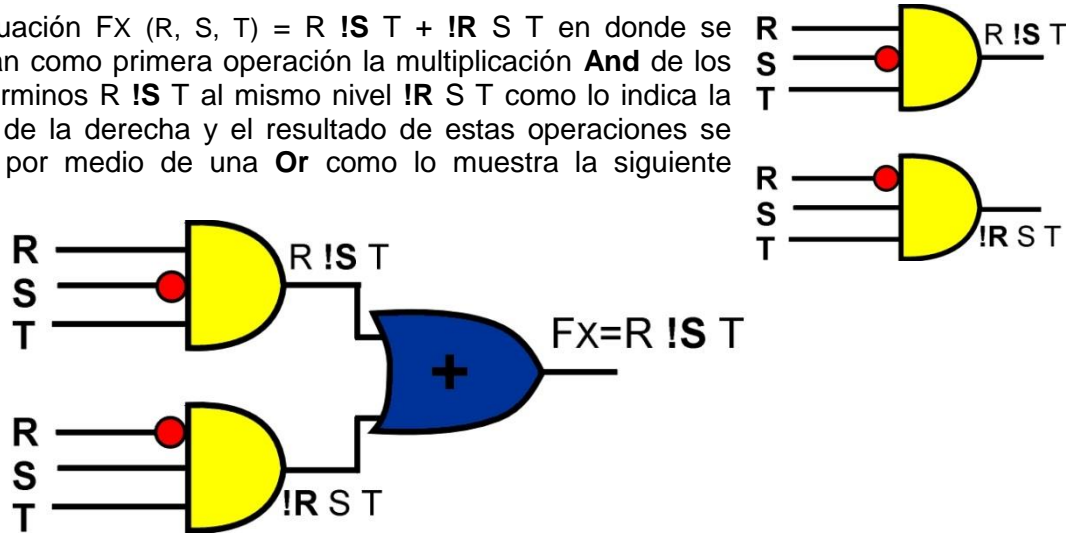
Descripción verbal	Descripción con palabras del comportamiento de un sistema, Utilizando alternativas (o) y/o Condiciones (y)
Circuito o Diagrama Esquemático.	Representación gráfica de una expresión Booleana a través de la interconexión de símbolos que corresponden a los operadores lógicos.
Ecuación.	Representación matemática de una función booleana.
Tabla de verdad.	Representación tabular de los valores de salida del sistema, para cada una de las posibles combinaciones de entrada.
Diagrama de Tiempos	Representación gráfica de los valores de salida del sistema, para todas las combinaciones de entrada en un tiempo dado.

### ***Obtención del circuito a partir de la ecuación.***

Partiendo de una ecuación Booleana es posible obtener su circuito o diagrama esquemático por el orden de sus operaciones.

Ejemplo 1:

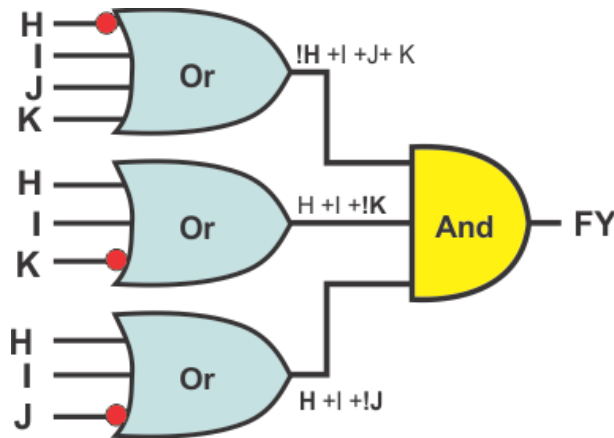
La ecuación  $F_X(R, S, T) = R \bar{S} T + \bar{R} S T$  en donde se realizan como primera operación la multiplicación **And** de los dos términos  $R \bar{S} T$  al mismo nivel  $\bar{R} S T$  como lo indica la figura de la derecha y el resultado de estas operaciones se suma por medio de una **Or** como lo muestra la siguiente figura.



A este formato se le conoce como la suma de los productos **SOP** o And/Or

Ejemplo 2 :

La ecuación  $F_Y(H, I, J, K) = (\bar{H} + I + J + K)(H + \bar{I} + K)(H + I + \bar{J})$ , en donde se realiza como primera operación la Or que está dentro del paréntesis representada por tres términos  $(\bar{H} + I + J + K)$ , al mismo nivel  $(H + \bar{I} + K)$  y  $(H + I + \bar{J})$  y el resultado de estas operaciones se multiplica por medio de una **And** como lo muestra la siguiente figura.

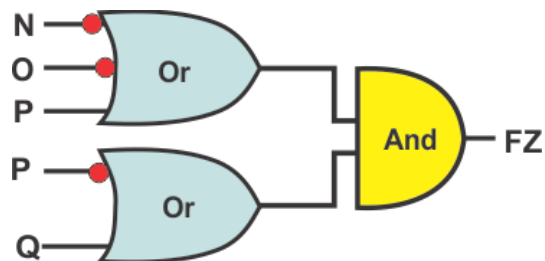


A este formato se le conoce como el producto de las sumas **POS** o Or/And.

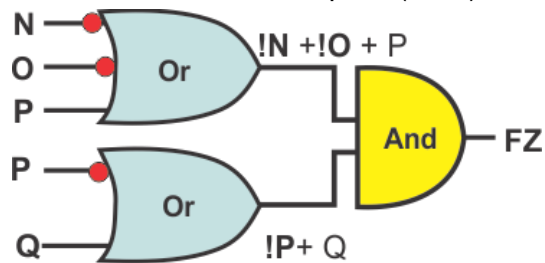
**Obtención de la ecuación a partir del circuito.**

Para obtener la ecuación a partir del circuito es recomendable, describir cada una de las operaciones del circuito en el orden de izquierda a derecha.

Ejemplo1:

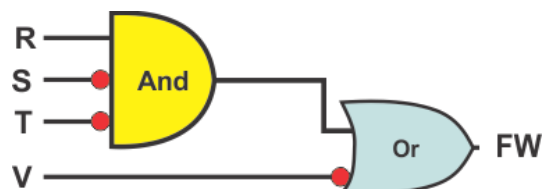


La primera operación que se realiza en este circuito son las OR que están al mismo nivel y el resultado de estas se multiplica (AND).

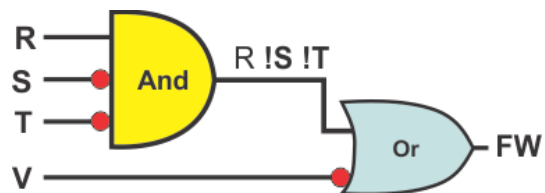


$$FZ = (!N + !O + P)(!P + Q)$$

Ejemplo 2:



La primera operación que se efectúa es la And de R y S y T y después la suma (OR) con V:

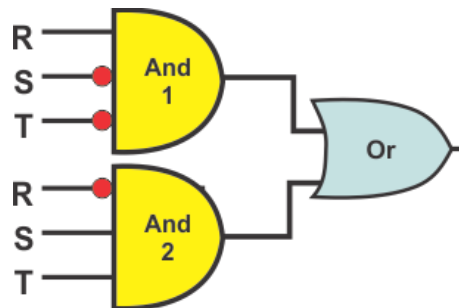


$$FW = R !S !T + !V$$

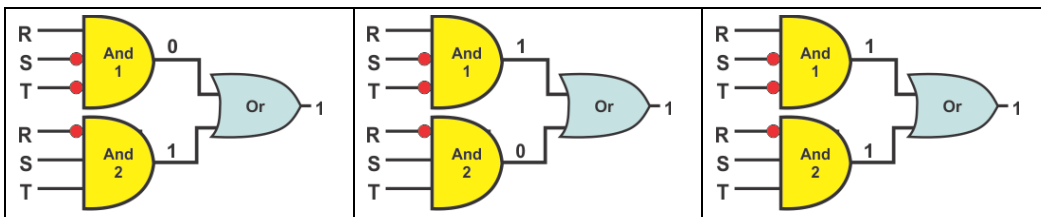
## Tabla de Verdad

Para obtener la tabla de verdad partiendo de un circuito, se proponen dos opciones, la primera consiste en dar valores a las entradas probando una a una todas las posibles combinaciones y obtener el valor de salida para cada una de ellas lo cual sería un método muy largo para el caso de múltiples entradas.

La segunda opción es un método que consiste en suponer un valor conveniente de salida y verificar que combinaciones de entrada cumplen con el valor propuesto.



Por ejemplo en este circuito mostrado en la figura anterior, que termina en OR conviene suponer un valor de **uno** a la salida de la **Or** esto nos da una alternativa ya que cualquier entrada igual a uno en la operación Or produce una salida uno (una, otra o ambas), eso nos permite analizar por separado cada una de las entradas de la Or y determinar las combinaciones de R, S y T para las And 1 y la And 2 de que cumplan con los valores para generar un uno.



Analizada la salida de la And de arriba, la salida es uno solo cuando todas sus entradas son uno entonces **R=1, S=0 y T=1**, esto se presenta en la combinación 5 de la tabla de verdad (m=5).

En la And de abajo la salida es uno cuando todas sus entradas son uno **R=0, S=1 y T=1**, esto se presenta en la combinación 3 de la tabla de verdad (m=3), todas las demás combinaciones serán igual a cero.

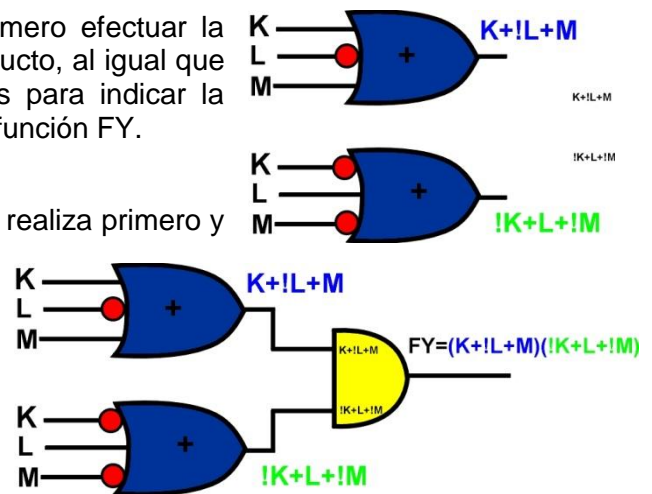
Tabla de Verdad de la Función Fx

M	R S T	Fx
0	0 0 0	0
1	0 0 1	0
2	0 1 0	0
3	0 1 1	1
4	1 0 0	0
5	1 0 1	1
6	1 1 0	0
7	1 1 1	0
15	1 1 1	0

Ejemplo: En el caso que se necesite primero efectuar la operación **Or** suma antes que la **And** producto, al igual que en el álgebra se hace uso de paréntesis para indicar la prioridad como se muestra en la siguiente función FY.

$$FY ( K,L,M ) = (K + !L + M) (!K + L + !M)$$

En donde la operación OR ( $K + !L + M$ ) se realiza primero y al mismo nivel la operación Or ( $!K + L + !M$ ) como lo muestra la figura a la derecha, posteriormente con la salida de estas dos se realiza la operación And. Como lo muestra la siguiente figura:

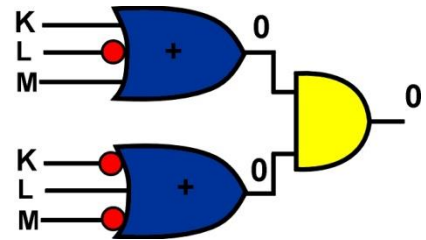


### Tabla de Verdad

Para obtener la tabla de verdad de este circuito se puede suponer un valor de **cero** a la salida de la **And** esto nos da una alternativa ya que cualquier entrada cero en la operación And produce una salida cero (una, otra o ambas).

Analizada la salida de la Or de arriba la salida es cero solo cuando todas sus entradas son cero entonces **K=0, L=1 y M=0**, esto se presenta en la combinación 2 de la tabla de verdad (m=2).

En la la Or de abajo la salida es cero solo cuando todas sus entradas son cero entonces **K=1, L=0 y M=1**, esto se presenta en la combinación 5 de la tabla de verdad (m=5), en la tabla de verdad, todas las salidas para las demás combinaciones serán igual a uno.



**Tabla de verdad para la función Fy**

M	K L M	Fy
0	0 0 0	1
1	0 0 1	1
2	0 1 0	0
3	0 1 1	1
4	1 0 0	1
5	1 0 1	0
6	1 1 0	1
7	1 1 1	1

## **Actividad de aprendizaje**

**Con las Funciones asignadas por el maestro o instructor efectúe el siguiente procedimiento**

- 1.- Dibuje el circuito correspondiente para las funciones F1 y F2.
- 2.- Obtenga la Tabla de Verdad en forma analítica partiendo del circuito.
- 3.- Con el programa de aplicación IspStarter en el módulo de captura esquemática dibuje las funciones F1 y F2 Asignadas
- 4.- Obtenga el diagrama de tiempos por medio del archivo Test\_Vectors y compare los resultados obtenidos con la tabla de verdad del paso 2
- 5.- Si la tabla de verdad coincide con el diagrama de tiempos, programe un Dispositivo Lógico programable y construya un prototipo.
- 6.- Compruebe físicamente todas las combinaciones con la tabla de verdad y el diagrama de tiempos.
- 7.- Elabore su reporte

### **Material a utilizar**

Tablilla de conexiones

Circuito Integrado PLD GAL16V8 o GAL22V10.

6 resistores de 330  $\Omega$ .

Fuente de 5 V de corriente directa.

1 Dip switch de 4 o 8 interruptores o 4 micro Push Boton NA .

6 Leds

- Alambre para conexiones.

### **Programas de aplicación (software):**

IspLEVER

LogicAid

Microsoft Word (reporte)

Recortes (Windows XP o Windows 7)







A	1	24	V <sub>CC</sub>	m	A	B	C	D	F1	F2
D	2	23	F1	0	0	0	0	0		
C	3	22		1	0	0	0	1		
B	4	21		2	0	0	1	0		
	5	20		3	0	0	1	1		
	6	19		4	0	1	0	0		
	7	18		5	0	1	0	1		
	8	17		6	0	1	1	0		
	9	16		7	0	1	1	1		
	10	15		8	1	0	0	0		
	11	14	F2	9	1	0	0	1		
GND	12	13		10	1	0	1	0		
				11	1	0	1	1		
				12	1	1	0	0		
				13	1	1	0	1		
				14	1	1	1	0		
				15	1	1	1	1		

Es muy probable que la asignación de terminales realizada por el programa IspStarter no estén en el mismo orden de la tabla de verdad como lo muestra la figura, por lo que se recomienda al efectuar las conexiones en el proto alambrar los interruptores o dip switch en el mismo orden de la tabla de verdad, de modo que al comprobar la tabla con el circuito las combinaciones de entrada coincidan y así observar los valores de salida

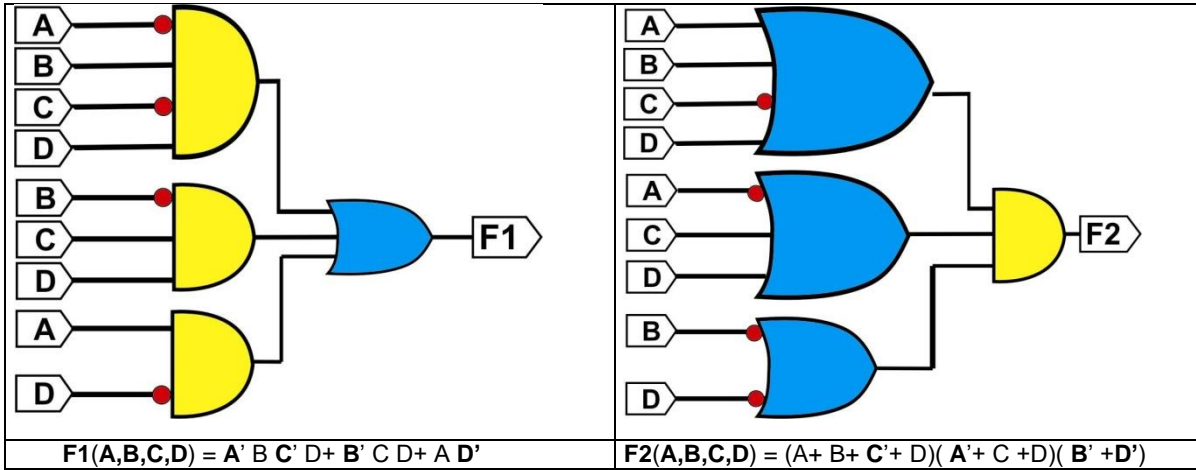
Ejemplo, obtenga para F1 y F2:

- La Tabla de Verdad en forma analítica.
- La Tabla de Verdad de la implementación del circuito.
- El diagrama de tiempos usando el archivo TEST\_VECTORS.

Funciones:  $F1(A, B, C, D) = A' B C' D + B' C D + A D'$

$F2(A, B, C, D) = (A + B + C' + D)(A' + C + D)(B' + D')$

Dibuje el circuito de F1 y F2



Obtenga la Tabla de Verdad mediante el análisis de F1 y F2

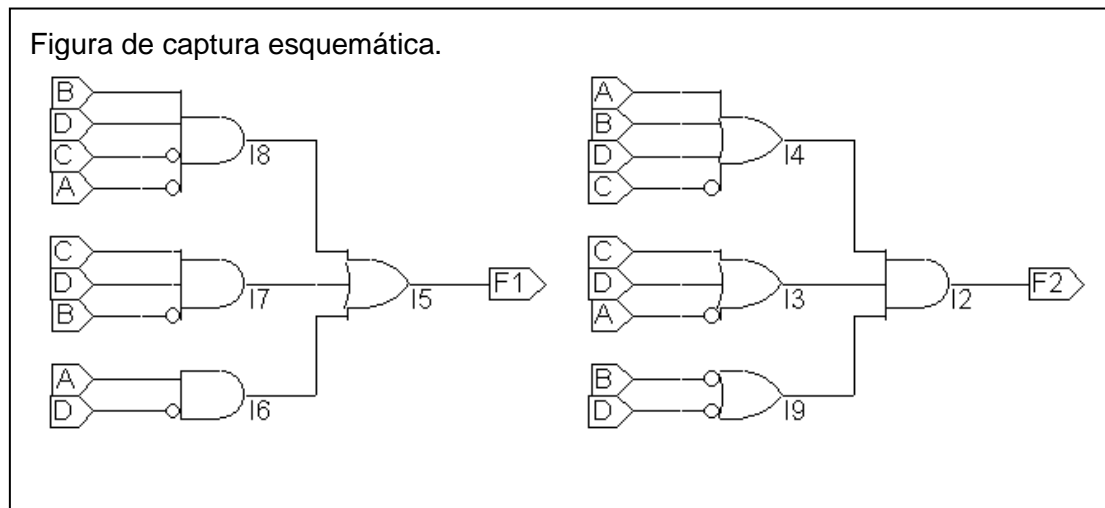
m	A B C D	F1		F2	
0	0 0 0 0	0		1	
1	0 0 0 1	0		1	
2	0 0 1 0	0		0	$A + B + C' + D$
3	0 0 1 1	1	$B' C D$	1	
4	0 1 0 0	0			
5	0 1 0 1	1	$A' B C' D$	0	$B' + D'$
6	0 1 1 0	0		1	
7	0 1 1 1	0		0	$B' + D'$
8	1 0 0 0	1	$A D'$	0	$A' + C + D$
9	1 0 0 1	0		1	
10	1 0 1 0	1	$A D'$	1	
11	1 0 1 1	1	$B' C D$	1	
12	1 1 0 0	1	$A D'$	0	$A' + C + D$
13	1 1 0 1	0		0	$B' + D'$
14	1 1 1 0	1	$A D'$	1	
15	1 1 1 1	0		0	$B' + D'$

Tabla de Verdad obtenida en forma analítica.

m	A B C D	F1	F2
---	---------	----	----

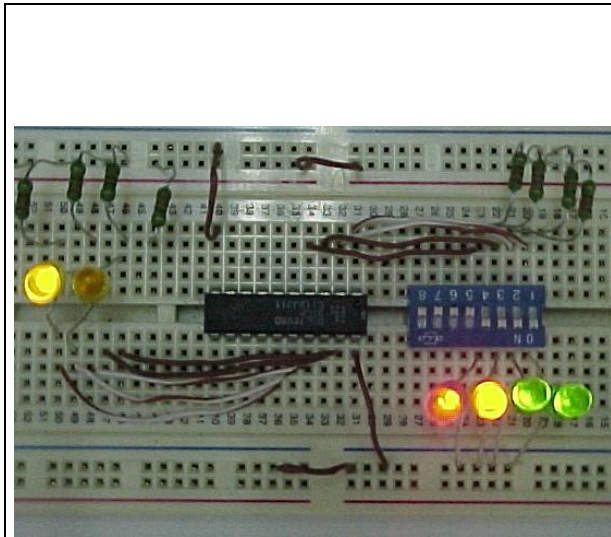
0	0 0 0 0	0	1
1	0 0 0 1	0	1
2	0 0 1 0	0	0
3	0 0 1 1	1	1
4	0 1 0 0	0	
5	0 1 0 1	1	0
6	0 1 1 0	0	1
7	0 1 1 1	0	0
8	1 0 0 0	1	0
9	1 0 0 1	0	1
10	1 0 1 0	1	1
11	1 0 1 1	1	1
12	1 1 0 0	1	0
13	1 1 0 1	0	0
14	1 1 1 0	1	1
15	1 1 1 1	0	0

**Programa las funciones F1 y F2 en el circuito integrado GAL16V8D por medio de capture esquemática.**



<p style="text-align: center;">P16V8AS</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>D</td><td>1</td><td>20</td><td>Vcc</td></tr> <tr><td>C</td><td>2</td><td>19</td><td></td></tr> <tr><td>B</td><td>3</td><td>18</td><td></td></tr> <tr><td>A</td><td>4</td><td>17</td><td></td></tr> <tr><td></td><td>5</td><td>16</td><td>!F1</td></tr> <tr><td></td><td>6</td><td>15</td><td>!F2</td></tr> <tr><td></td><td>7</td><td>14</td><td></td></tr> <tr><td></td><td>8</td><td>13</td><td></td></tr> <tr><td></td><td>9</td><td>12</td><td></td></tr> <tr><td>GND</td><td>10</td><td>11</td><td></td></tr> </table> <p style="text-align: center;">SIGNATURE: N/A</p>	D	1	20	Vcc	C	2	19		B	3	18		A	4	17			5	16	!F1		6	15	!F2		7	14			8	13			9	12		GND	10	11		<p>Archivo JEDEC necesario para programar el GAL16V8</p>
D	1	20	Vcc																																						
C	2	19																																							
B	3	18																																							
A	4	17																																							
	5	16	!F1																																						
	6	15	!F2																																						
	7	14																																							
	8	13																																							
	9	12																																							
GND	10	11																																							
<p>Archivo Reporte que indica la distribución de terminales del circuito integrado (PIN OUT).</p>																																									
	<pre> Report Viewer - [prac4.jed] File Edit View Options Window Help  LispDesignEXPERT 8.1 Lattice Semiconductor Corp. JEDEC file for: P16V8AS V9.0 Created on: Tue Dec 12 20:42:02 2000  prac4.blc * QF20* QF2194* QV0* F0* Y0* NOTE Table of pin names and numbers* NOTE PINS F2:15 F1:16 D:1 B:3 C:2 A:4* L0768 01010111111111111111111111111111* L0800 11010111011111111111111111111111* L1024 10010111011111111111111111111111* L1056 10101011011111111111111111111111* L1088 10011011101111111111111111111111* L2048 00010000* L2128 11* L2192 1* C1B59* I7ASD </pre>																																								

Programar el GAL16V8, asegúrese de cargar el archivo JEDEC y definir el dispositivo a programar.



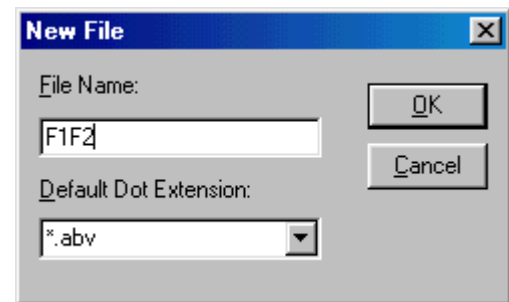
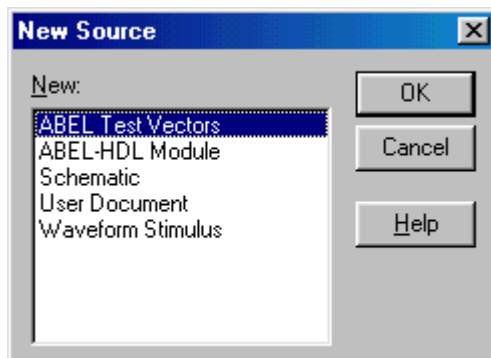
m	A B C D	F1	F2
0	0 0 0 0		
1	0 0 0 1		
2	0 0 1 0		
3	0 0 1 1		
4	0 1 0 0		
5	0 1 0 1		
6	0 1 1 0		
7	0 1 1 1		
8	1 0 0 0		
9	1 0 0 1		
10	1 0 1 0		
11	1 0 1 1		
12	1 1 0 0		
13	1 1 0 1		
14	1 1 1 0		
15	1 1 1 1		

## Pasos para la Simulación

Capture el programa ABEL Test Vectors.

Pasos para generar el Archivo Test Vectors

- 1.-En el menú de Source seleccione New y después ABEL Test Vectors.
- 2.-Teclee el nombre del archivo



- 3.-En la ventana del **Text Editor** teclee el siguiente archivo.

**Module F1**

**" ENTRADAS**

**A,B,C,D PIN;**

**"SALIDAS**

**F1, F2 PIN ISTYPE'COM';**

**TEST\_VECTORS**

**([A,B,C, D]-> [F1,F2])**

**[0,0,0,0]->[.x.,.x.];**

**[0,0,0,1]->[.x.,.x.];**

**[0,0,1,0]->[.x.,.x.];**

**[0,0,1,1]->[.x.,.x.];**

**[0,1,0,0]->[.x.,.x.];**

**[0,1,0,1]->[.x.,.x.];**

**[0,1,1,0]->[.x.,.x.];**

**[0,1,1,1]->[.x.,.x.];**

**[1,0,0,0]->[.x.,.x.];**

**[1,0,0,1]->[.x.,.x.];**

**[1,0,1,0]->[.x.,.x.];**

**[1,0,1,1]->[.x.,.x.];**

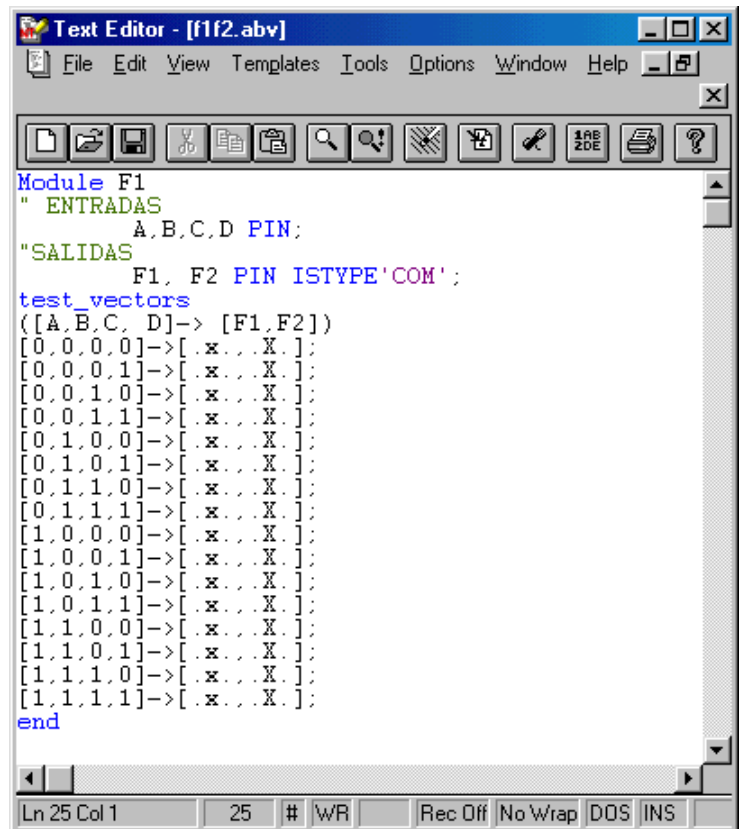
**[1,1,0,0]->[.x.,.x.];**

**[1,1,0,1]->[.x.,.x.];**

**[1,1,1,0]->[.x.,.x.];**

**[1,1,1,1]->[.x.,.x.];**

**End**





4.-Compile el archivo Test Vectors.

Regrese al programa **Project Navigator** en donde aparecerá incluido el archivo **F1F2.ABV**

Efectué la compilacion Compile Test Vectors.

Ejecute Simulation JEDEC File.

Ejecute JEDEC simulation Waveform.

En el programa Waveform Viewer aparecerá una nueva pantalla, **seleccione Edit y posteriormente SHOW.**

Seleccione las variables A y oprima Show, posteriormente la variable B y de nuevo Show y así con las demás variables C, D, F1 y F2

Cierre la ventana Show Waveform y compare la gráfica con la tabla de verdad.

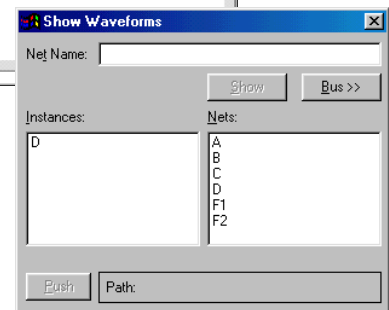
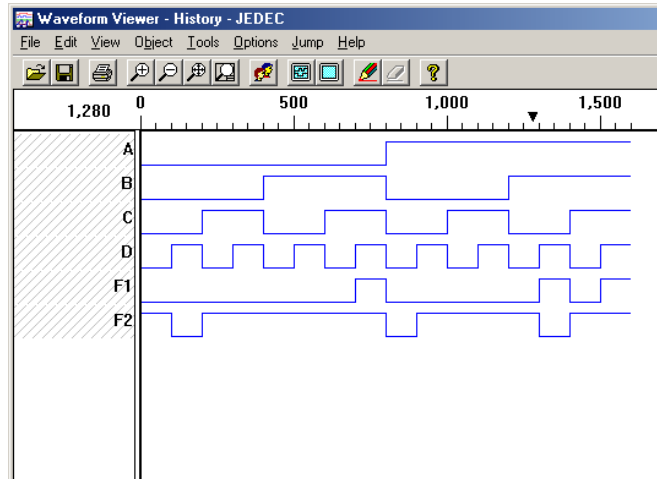
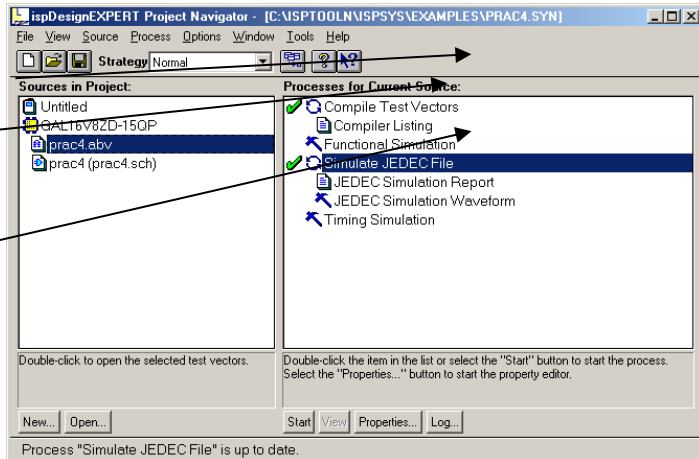
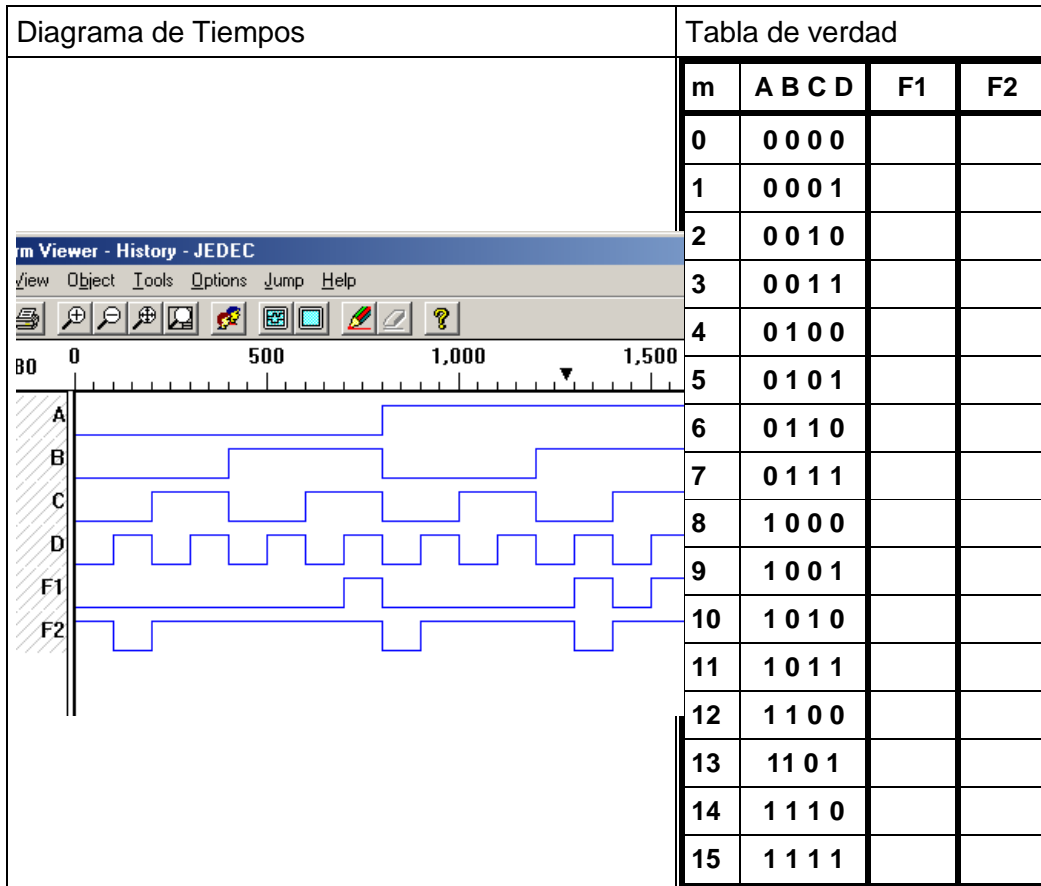


Diagrama de tiempos de las funciones F1 y F2.



**Trabajo solicitado, obtenga para uno de los ejercicios lo siguiente:**

- d) La tabla de verdad en forma analítica partiendo.
- e) La tabla de verdad de la implementación del circuito.
- f) El diagrama de tiempos usando el archivo TEST\_VECTORS.

## Ecuaciones propuestas

	F1 SOP	F2 POS
1	$A' B' C' D' + A' B' D' + A B C' D'$	$(A + B + C' + D) (A + B + D') (A' + B + C' + D')$
2	$A' B' C' D' + A' B' D' + C' D'$	$(A + B + C' + D) (A + B + D') (A' + C' + D')$
3	$A' B' C' D' + A B D' C + C' D'$	$(A' + B + C' + D') (A + B + D') (A' + C' + D')$
4	$A' B' C' D + A' C D' + C' D$	$(A' + B + C' + D) (A + C + D) (A' + B + C' + D')$
5	$A' B C' D + A' C D' + C' D$	$(A' + B' + C' + D) (A + C + D) (A' + B + C' + D')$
6	$A' B C' D + A' C D' + A' D$	$(A' + B' + C' + D) (A + C' + D) (A' + B + C' + D')$
7	$A' B C' D + A' C D' + A' B$	$(A' + B' + C' + D) (A + B + D) (A' + B + C' + D')$
8	$A' B C' D + A' D' + A' B D$	$(A + B' + C' + D) (A' + B + D) (A' + B + C' + D')$
9	$X' Z' W' + X' Y W' + X' Y$	$(X + Y' + Z' + W) (X' + Y + W) (Y' + Z' + W')$
10	$X' Z' Y W' + X' Y W' + X' Y$	$(X + Z' + W) (X' + Y + W) (Y' + Z' + W')$
11	$X' Z' Y W' + X' W' + X Y$	$(X' + Z' + W) (X' + Y + W) (Y + Z' + W')$
12	$X' Z' Y W + X' W + X' Y$	$(X' + Z' + Y + W) (X' + Y + W) (Y + W')$
13	$X' Z + Z' W' + X' Y W$	$(X + Z + W') (X' + Z' + W) (Y + W')$
14	$X' Y' W' + X' Y W + X' Z' W'$	$X (Y + Z + W') (Y' + W)$
15	$Y Z W' + X' Z W' + X' Y$	$(X + W) (X + Z') (X + Y') (Y' + W) (Y' + Z')$
16	$Y' W' + X' Z' W' + X' Y'$	$(X + Y) (X + W) (Y + Z) (Y + W)$
17	$A' B' C' D' + A' B' + A B C' D$	$(A + D') (A' + B) (A + B') (B' + C')$
18	$B + A D' + A C' + C' D'$	$(D') (A' + B) (A + B') (B' + C')$
19	$A D' + A C' + A' B + C' D'$	$(D') (B + C') (A + C')$
20	$D' + A C' + A' B$	$(C' + D') (C + D) (A' + D)$
21	$A C' + A' C + B C + A' D$	$(C' + D') (C + D) (A' + C')$
22	$B D + A C' + A' C + B' C D' + A' D$	$(A') (B + D') (B + C)$
23	$B D + C' + A B' D' + A' D$	$(D') (A' + B) (A + B') (A' + C')$
24	$B D + A B' D' + A' B + A' D + A C'$	$(A + D') (B + C') (A + C')$
25	$A B + C' D + A' B' + A' D + A' C'$	$(A' + B) (B + D') (B + C)$
26	$X Y W' + Y' W + X' Y' + Y Z'$	$(X') (Y + Z') (Y + W')$
27	$X Y W' + Y' W + X' Z' + Y Z'$	$(X' + Y') (X' + W') (Y' + Z') (Y' + W')$
28	$Y W + Y Z' + Z' W + X' W'$	$(X' + W') (X' + Z) (X' + Y) (Y + W') (Y + Z)$
29	$B D + A C' + A' C + B' C D' + A' D$	$(A' + B' + C' + D) (A' + B + C' + D') (A + B + C + D')$
30	$B D + C' + A' D$	$(A') (C' + D') (C + D) (B + C')$
31	$A D + C' + B' D$	$(A') (C' + D') (C + D) (B + C')$
32	$A' B C D' + A' C' D$	$(C' + D) (A + B' + C')$
33	$A B C + B' C + A B D' + A C D.$	$(A' + C' + D) (A' + B) (A + B + D').$

## **Reporte** (lista de Cotejo, Check List)

<b>1</b>	<b>Portada.</b>
<b>2</b>	<b>Ecuaciones asignadas F1 y F2.</b>
<b>3</b>	<b>Circuito de F1 y F2</b>
<b>4</b>	<b>Diagrama de Bloques (entradas y Salidas)</b>
<b>4</b>	<b>Tabla de verdad</b>
<b>5</b>	<b>Captura esquemática</b>
<b>6</b>	<b>Archivo Abel Test Vectors</b>
<b>7</b>	<b>Simulación y comparación con la tabla de Verdad</b>
<b>8</b>	<b>Ecuaciones mínimas del archivo reporte.</b>
<b>9</b>	<b>Distribución de terminales (Pin Out)</b>
<b>10</b>	<b>Foto del circuito y comprobación de su funcionamiento</b>
<b>11</b>	<b>Conclusiones y recomendaciones</b>