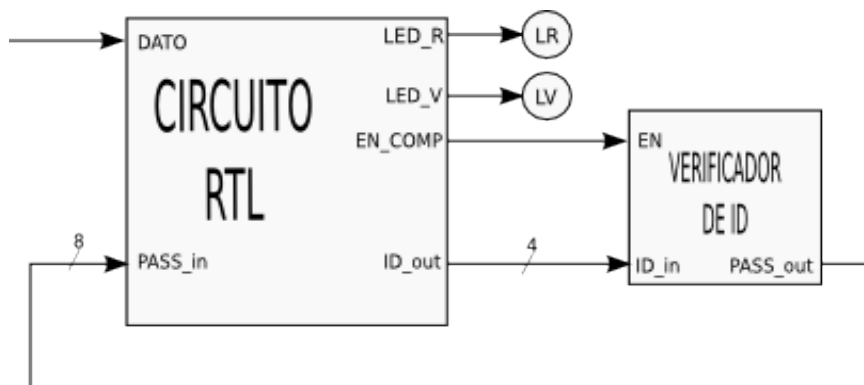


- Mínimo de aprobación: un problema y un ejercicio
- Cada hoja debe tener Nombre y CI.
- Utilice solo un lado de las hojas
- Deben estar numeradas y la primer hoja debe decir el total de hojas
- Incluya un solo problema por hoja
- Sea prolijo

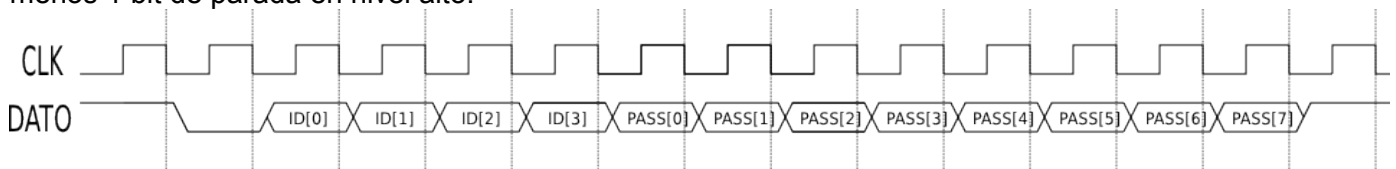
Problema 1



Se desea diseñar un circuito RTL que reciba un dato serie de 12 bits, que contiene un identificador de usuario (ID) de 4 bits y la correspondiente contraseña de 8 bits, y verifique si el usuario existe y si la contraseña es la correcta. Para lograrlo, se tiene un bloque comparador que es capaz de verificar si un ID es válido o no y obtener la contraseña correspondiente a dicho ID.

El bloque “Verificador de ID” funciona de la siguiente manera. En el flanco de bajada del reloj del sistema, si su entrada EN se encuentra en nivel alto, el bloque coloca en su salida PASS_out la contraseña correspondiente al usuario indicado en la entrada ID_in si éste es un usuario válido, y 00h si el usuario no es válido. La salida del bloque se mantendrá invariante hasta detectar nuevamente EN en nivel alto en un flanco de bajada del reloj.

Los datos serie llegan por una entrada DATO sincronizada con el reloj del sistema. La señal DATO se encuentra en 1 cuando el sistema se encuentra en reposo. Un dato válido comienza con un bit de arranque en 0 y luego los 12 bits de datos que contienen; los primeros 4 bits el ID y los siguientes 8 bits corresponden a la contraseña (comenzando siempre por el bit menos significativo), finalizando con al menos 1 bit de parada en nivel alto.



El circuito a diseñar debe contar con dos salidas, Led_R y Led_V, activas por nivel alto, para encender un led rojo y uno verde respectivamente. Si el usuario o la contraseña no son válidos se debe mantener prendido el LED rojo y apagado el LED verde. Si el usuario y la contraseña son correctos, se debe mantener prendido el LED verde y apagado el rojo. El estado de los LEDS solo se modifican al evaluar un nuevo dato.

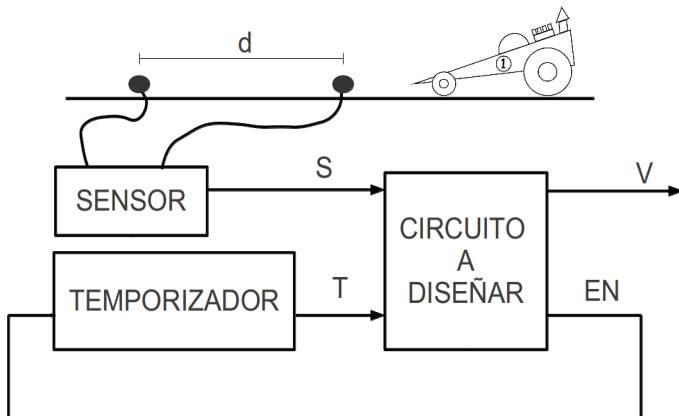
Adicionalmente, se debe implementar un sistema de seguridad tal que, si se ingresa de manera consecutiva 4 veces el mismo ID no válido, o 4 veces el mismo ID válido pero con contraseña incorrecta, el sistema debe permanecer bloqueado con ambos LEDS encendidos hasta el siguiente reset.

Se pide diseñar el circuito RTL detallado arriba (secuencia, bloque de control y bloque de datos) sensible al flanco de bajada.

Nota: Una Pass válida siempre es distinta de 00h y se puede suponer que los usuarios nunca ingresarán esta contraseña al identificarse.

Problema 2

Se desea diseñar un circuito secuencial modo nivel que permita indicar si un automóvil se encuentra circulando a una velocidad superior a la permitida. Para ello, se cuenta con un temporizador y dos sensores conectados como se indica en la figura.



El bloque SENSOR tiene una única salida la cual se encuentra normalmente en 0 y sube a 1 cuando por alguno de los dos sensores está pasando un automóvil. El temporizador tiene una entrada EN y una salida T y funciona de la siguiente manera. Cuando detecta un flanco de subida en su entrada EN (que normalmente se encuentra en nivel bajo), éste sube la señal T (que también se encuentra normalmente en nivel bajo) durante un tiempo t_{MIN} .

La distancia entre los sensores es tal que si el automóvil se encuentra circulando a una velocidad mayor a la permitida, este recorrerá la distancia d en un tiempo menor a t_{MIN} .

El circuito a diseñar debe funcionar de la siguiente manera. Al detectar un automóvil, debe subir la señal EN del temporizador. En consecuencia, el temporizador subirá la señal T durante el tiempo t_{MIN} . El circuito debe activar la salida V ($V=1$) si el automóvil se encuentra circulando a una velocidad mayor a la permitida. En este caso, la señal V debe permanecer activa durante el período de tiempo en el cual el automóvil se encuentra circulando por el segundo sensor. En el caso contrario la señal V se mantendrá en 0. La señal EN debe volver a 0 cuando el automóvil abandone el segundo sensor en el caso de la infracción, o cuando baje la señal T en el caso normal.

Diseñar el circuito secuencial modo nivel sin carreras, azares ni espurios.

NOTA: Se supone que:

- nunca habrá dos automóviles en el sistema y la separación entre vehículos consecutivos siempre será mucho mayor a t_{MIN} .
- el tiempo durante el cual se encuentra el automóvil en alguno de los sensores es mucho menor al tiempo necesario para recorrer la distancia d .
- el retardo introducido por el temporizador, entre la subida de EN y la subida de T es mucho menor que el tiempo durante el cual el automóvil se encuentra frente a alguno de los sensores.
- se trabajará en modo fundamental, o sea que las entradas del sistema cambian solo una a la vez.

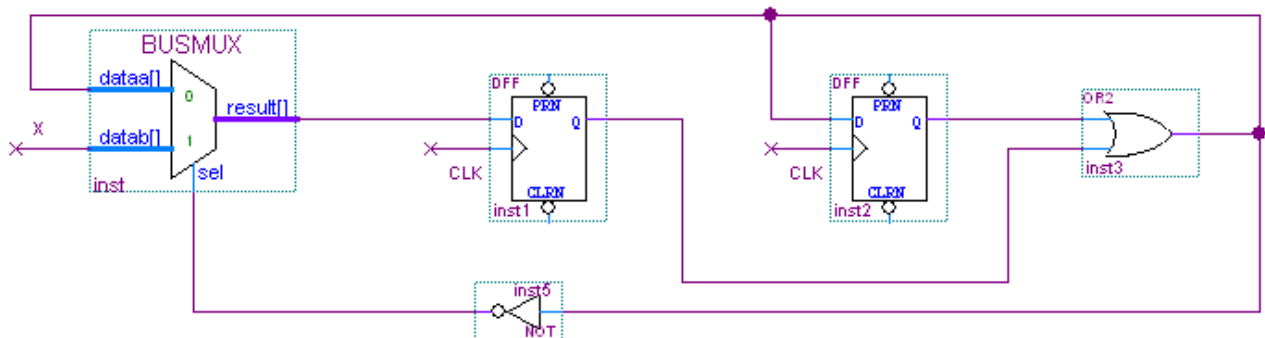
Ejercicio 1

a) Codificar en Hamming de 7bits: $m_3 m_2 m_1 m_0 = 0011$

1	2	3	4	5	6	7
P1	P2	m3	P3	m2	m1	m0

b) La siguiente es una palabra de 4 bits codificada en código Hamming de 7 bits: 0001001
Indicar cual era la palabra de 4 bits original.

Ejercicio 2



a) Determinar la frecuencia máxima de funcionamiento del circuito. Suponer la entrada X constante. Justificar claramente con un diagrama de tiempo, indicando a qué FF corresponden los parámetros utilizados.

b) Indicar cuál es el intervalo de tiempo durante el cual debe permanecer constante X para asegurar un correcto funcionamiento. Tomar como referencia el flanco de subida de clock.

Datos:

Tiempo de setup de FF: t_s

Tiempo de hold de FF: t_h

Tiempo de propagación de FF: $t_{pmin} < t_p < t_{pmax}$

Tiempo de prop. en compuertas and: $t_{andmin} < t_{and} < t_{andmax}$

Tiempo de prop. en compuertas not: $t_{notmin} < t_{not} < t_{notmax}$

Tiempo de prop. en mux:

Selector-salida: $t_{ssmin} < t_{ss} < t_{ssmax}$

Entrada-salida: $t_{esmin} < t_{es} < t_{esmax}$

SOLUCIÓN

Problema 1

Module: Solución

Input: Dato, Pass_in[7..0]

Output: Led_R, Led_V, En_Comp, ID_out[3..0]

Memory: Cont[3..0] , ID_M[3..0], ID_Ant[3..0], Pass_M[7..0], LR, LV,
Repeticion[1..0]

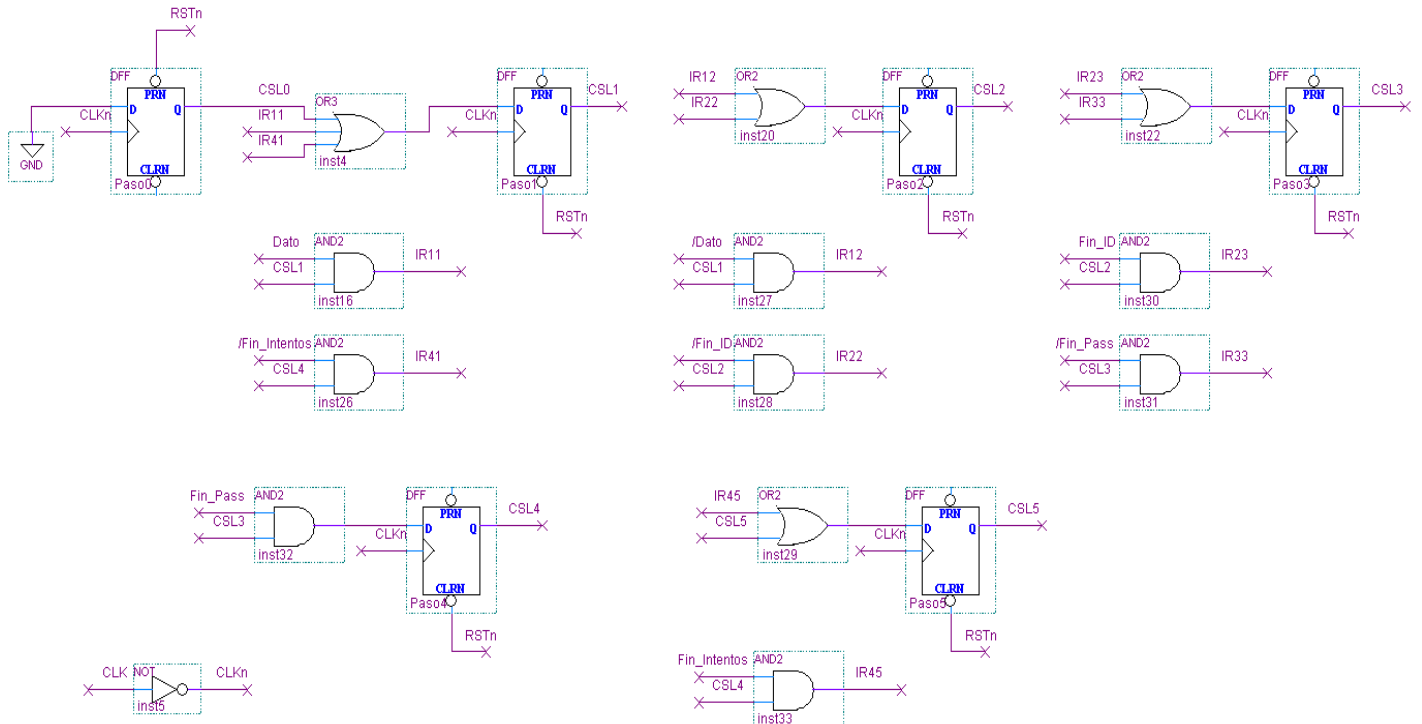
- a) LR \leftarrow 0
LV \leftarrow 0
Repeticion[] \leftarrow 00
ID_M[] \leftarrow 0000
- b) Cont[] \leftarrow 000
ID_Ant[] \leftarrow ID_M[]
 \rightarrow (Dato, /Dato) / (1,2)
- c) ID_M[] \leftarrow Dato, ID_M[3..1]
Cont[] \leftarrow INC(Cont[])
 \rightarrow (/Fin_ID, Fin_ID) / (2,3)
- d) Pass_M[] \leftarrow Dato, Pass_M[7..1]
Cont[] \leftarrow INC(Cont[])
En_Comp = 1
 \rightarrow (/Fin_Pass, Fin_Pass) / (3,4)
- e) LR \leftarrow /Pass_OK
LV \leftarrow Pass OK + Fin_Intentos
Repeticion[] \leftarrow INC(Repeticion[]).(Repite_ID. LR . /Pass_OK)
 \rightarrow (Fin_Intentos, /Fin_Intentos) / (5,1)
- f) \rightarrow (5)

End Sequence
Control Reset (0)

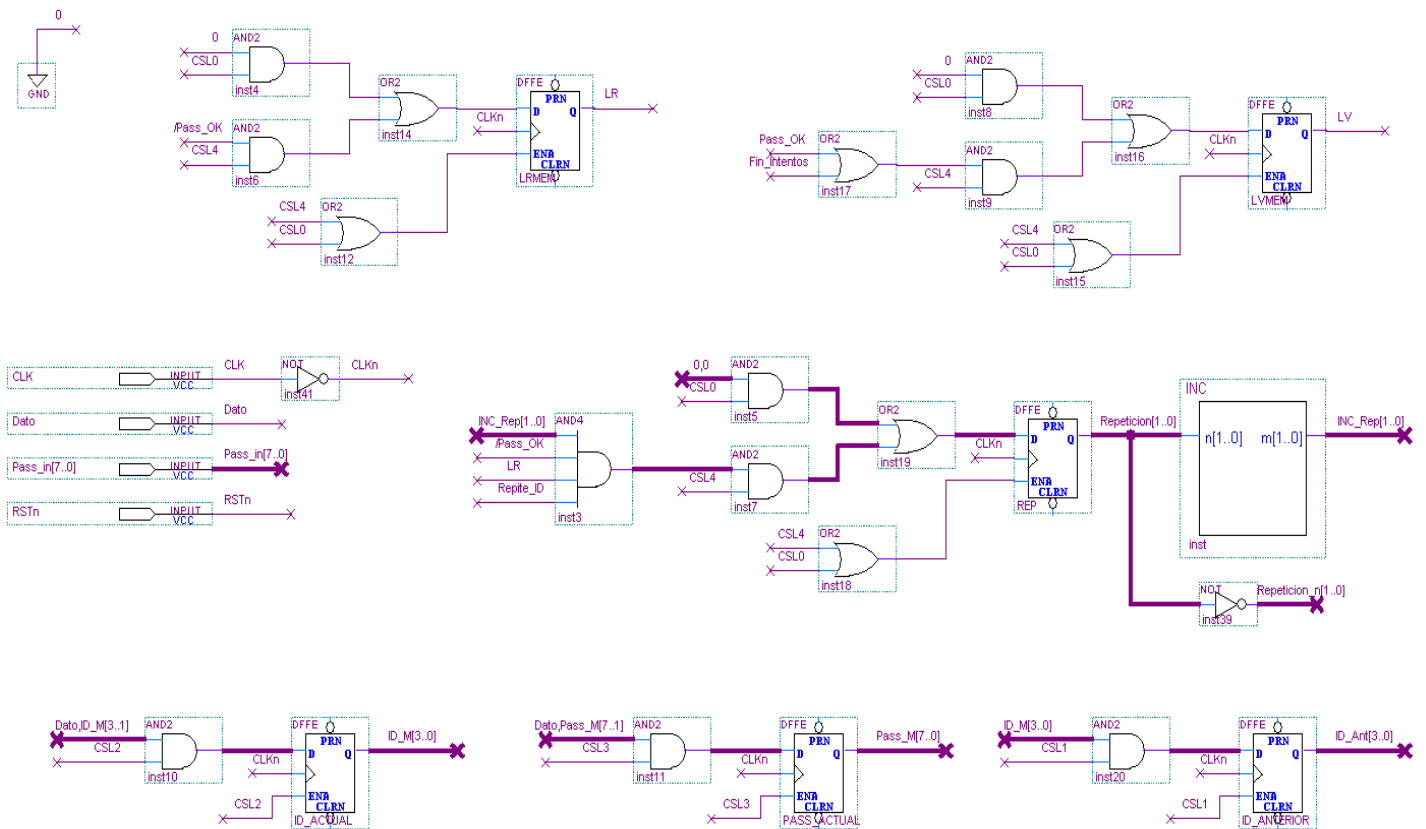
Fin_ID = (Cont []= 0011)
Fin_Pass = (Cont[]=1011)
Pass_OK = (Pass_M[] = Pass_in[])
Repite_ID = (ID_M[] = ID_Ant[])
Fin_Intentos = (Repeticion[]=10).(Repite_ID. LR . /Pass_OK)
ID_out[] = ID_M[]
Led_R = LR
Led_V = LV

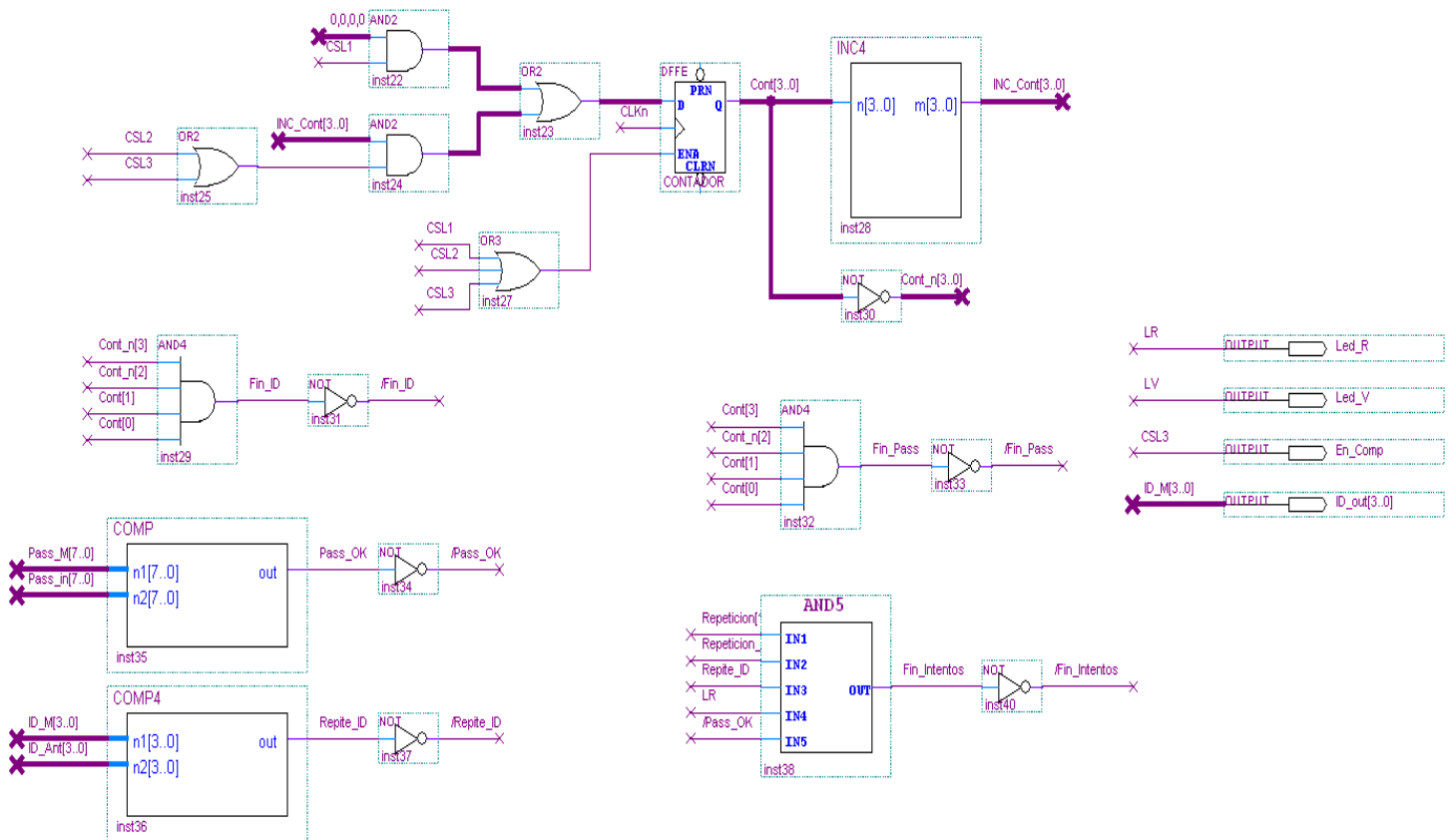
End

Bloque de Control:

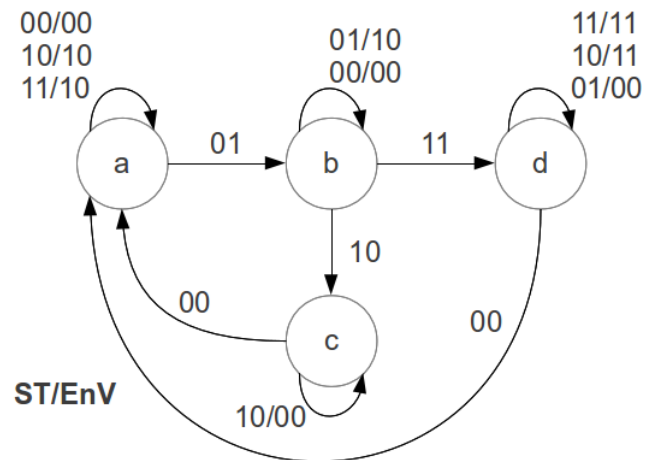


Bloque de Datos:



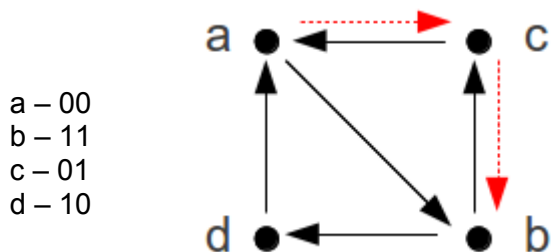


Problema 2



	00	01	11	10	00	01	11	10
a	a	b c	a	a	00	10	10	10
b	b	b	d	c	00	10	1x	00
c	a	b		c	00	10		00
d	a	d	d	d	00	00	11	11

El diagrama ya es mínimo,



y1y2/ST	00	01	11	10	00	01	11	10
00 (a)	00	01	00	00	00	10	10	10
01 (c)	00	11		01	00	10		00
11 (b)	11	11	10	01	00	10	1x	00
10 (d)	00	10	10	10	00	00	11	11

y1y2/ST	00	01	11	10
00	0	0	0	0
01	0	1	x	0
11	1	1	1	0
10	0	1	1	1

$$Y1 = y1.T + y2.T + y1.y2.!S + y1.!y2.S$$

y1y2/ST	00	01	11	10
00	0	1	0	0
01	0	1	x	1
11	1	1	0	1
10	0	0	0	0

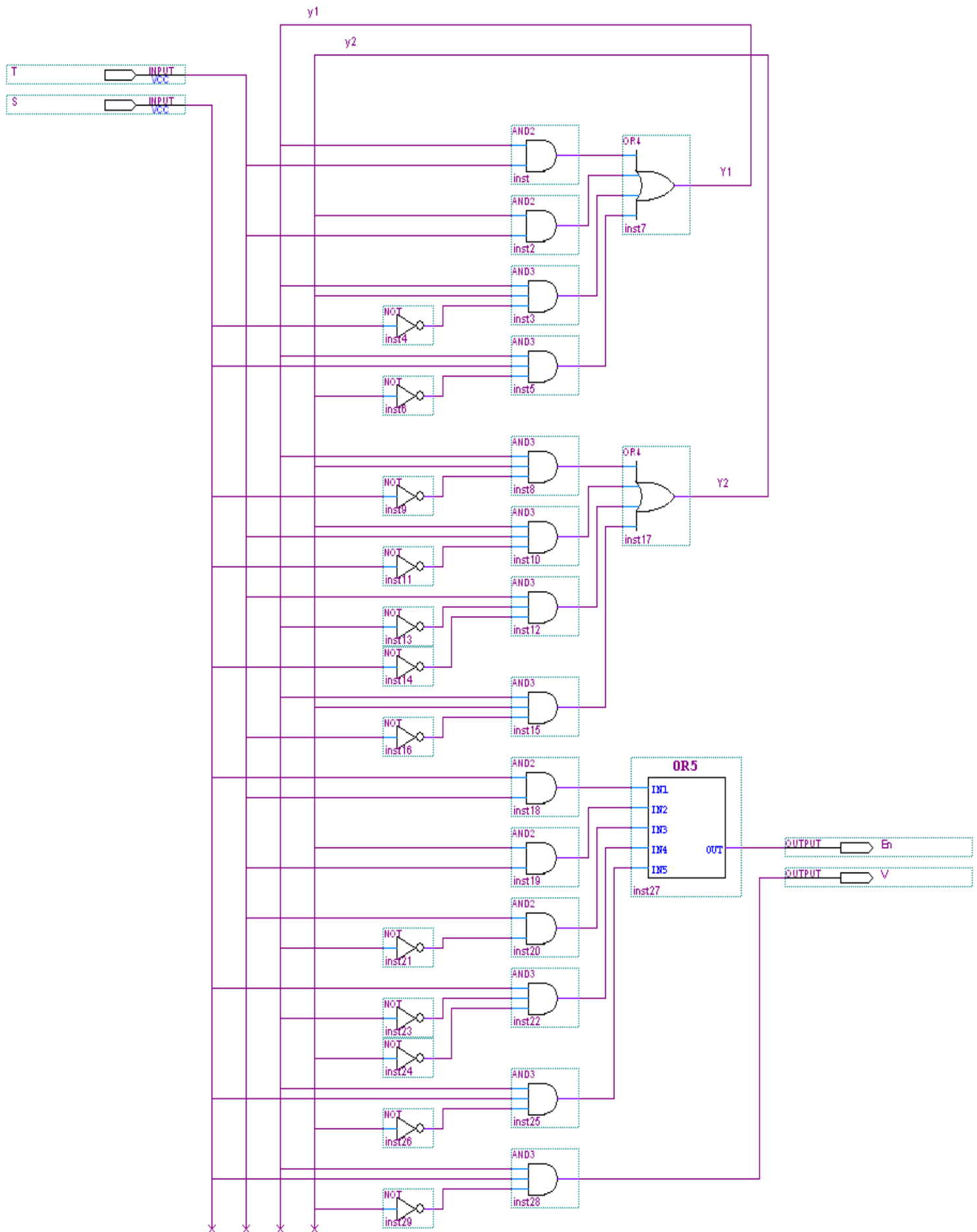
$$Y2 = y1.y2.!S + y2.!S.T + !y1.!S.T + y2.S.!T + y1.y2.!T$$

y1y2/ST	00	01	11	10
00	0	1	1	1
01	0	1	x	0
11	0	1	1	0
10	0	0	1	1

$$En = S.T + y2.T + !y1.T + !y1.!y2.S + y1.!y2.S$$

y1y2/ST	00	01	11	10
00	0	0	0	0
01	0	0	x	0
11	0	0	x	0
10	0	0	1	1

$$V = y1.!y2.S$$



Ejercicio 1

a) Palabra a codificar: 0011

$$P1 = m0 \oplus m2 \oplus m3 = 1$$

$$P2 = m0 \oplus m1 \oplus m3 = 0$$

$$P3 = m0 \oplus m1 \oplus m2 = 0$$

$$P1P2m3P3m2m1m0 = 1000011$$

b) Palabra recibida: 0001001

$$V1 = P1 \oplus m0 \oplus m2 \oplus m3 = 0 \oplus 1 \oplus 0 \oplus 0 = 1$$

$$V2 = P2 \oplus m0 \oplus m1 \oplus m3 = 0 \oplus 1 \oplus 0 \oplus 0 = 1$$

$$V3 = P3 \oplus m0 \oplus m1 \oplus m2 = 1 \oplus 1 \oplus 0 \oplus 0 = 0$$

$$V3V2V1=011 \text{ Error en el bit 3: } m3$$

La palabra codificada sin error es: 0011001 $\rightarrow m3m2m1m0 = 1001$

Ejercicio 2

a)

$$\text{Q-FF2 a D-FF1 por Or2} : T1 \geq tp_{2max} + tor_{max} + tes_{max} + ts_{1min}$$

$$\text{Q-FF1 a D-FF1 por Or2} : T2 \geq tp_{1max} + tor_{max} + tes_{max} + ts_{1min}$$

$$\text{Q-FF2 a D-FF1 por Or2 y NOT} : T3 \geq tp_{2max} + tor_{max} + tnot_{max} + tss_{max} + ts_{1min}$$

$$\text{Q-FF1 a D-FF1 por Or2 y NOT} : T4 \geq tp_{1max} + tor_{max} + tnot_{max} + tss_{max} + ts_{1min}$$

$$\text{Q-FF1 a D-FF2 por Or2} : T5 \geq tp_{1max} + tor_{max} + ts_{2min}$$

$$\text{Q-FF2 a D-FF2 por Or2} : T6 \geq tp_{2max} + tor_{max} + ts_{2min}$$

$$T = \max (T1 , T2 , T3 , T4 , T5 , T6)$$

$$f_{max} = 1/T$$

b)

$$txh > th - tes_{min}$$

$$txs > ts + tes_{max}$$