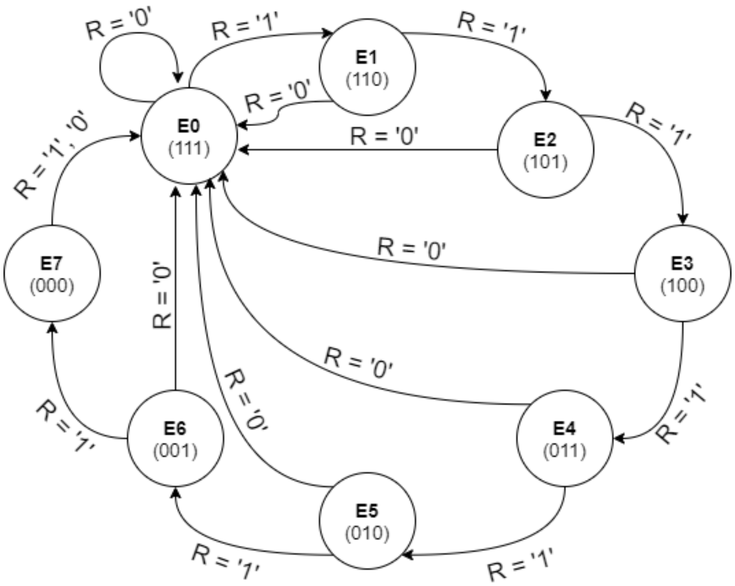
PRÁCTICO 5 - Circuitos Secuenciales
Ejercicio 6

Implementar un contador de 3 bits de cuenta regresiva (“111” -> “110” -> “101” -> … “000”), con una entrada **R** (reinicio), que lleve el contador al estado “111” en el siguiente ciclo de reloj, si su valor es igual a ‘0’. Utilizar Flip-flops tipo D y las compuertas lógicas necesarias. Tener en cuenta que el contador es cíclico, es decir, que pasa del estado “000” al “111”.

Dado que se trata de un contador de 3 bits, serán necesarios 8 estados en la implementación del circuito (uno para cada valor de cuenta). El diagrama de estados se puede observar en la figura siguiente:



La salida correspondiente a cada estado está expresada entre paréntesis (xxx), comenzando por el estado inicial E0, cuyo valor de salida es 111, y avanza al próximo valor de cuenta mientras la entrada de Reset (R) se encuentra inactiva en ‘1’. Como se observa en el diagrama, si la señal de R toma el valor “0”, el contador pasa al estado E0 nuevamente.
Debido a que la cantidad de FF-D necesarios para representar los estados del contador coincide con la cantidad de salidas necesarias del circuito, este problema presenta dos alternativas de implementación posibles, descritas a continuación.

Implementación: opción 1
La primera opción es codificar los estados de forma convencional (E0 = ‘000’ ; E1 = ‘001’... E7 = ‘111’), y armar las tablas de verdad del combinacional de estados y la del combinacional de salida en base al diagrama de estados. A modo de ejemplo, a continuación se muestra la tabla de codificación de estados y de salida del contador para esta forma de implementación:

Codificación de estados				Salidas del circuito		
Estados	**Q2**	**Q1**	**Q0**	**Out0**	**Out1**	**Out0**
E0	0	0	0	1	1	1
E1	0	0	1	1	1	0
E2	0	1	0	1	0	1
E3	0	1	1	1	0	0
E4	1	0	0	0	1	1
E5	1	0	1	0	1	0
E6	1	1	0	0	0	1
E7	1	1	1	0	0	0

Luego de elaborar la tabla de verdad para el combinacional de estados, se deben simplificar las funciones lógicas y proceder a su implementación mediante compuertas lógicas.
(A completar por el alumno).

Implementación: opción 2

La segunda alternativa es codificar los estados con el mismo valor numérico que el de la salida del circuito. La ventaja de esta forma es que no es necesaria la implementación del circuito combinacional de salida, ya que, como se dijo antes, los valores de salida de los FF-D constituyen directamente la salida del contador. La codificación de estados de esta implementación se muestra en la siguiente tabla:

Codificación de estados			
Estado	**Q2**	**Q1**	**Q0**
E0	1	1	1
E1	1	1	0
E2	1	0	1
E3	1	0	0
E4	0	1	1
E5	0	1	0
E6	0	0	1
E7	0	0	0

De esta forma, solo es necesario determinar la tabla de verdad de cambio de estados. Para este ejercicio quedaría conformada de la siguiente forma:

Entradas del combinacional de estados				Salidas del combinacional de estados		
Estado Actual			Entrada	Estado siguiente		
Q2	**Q1**	**Q0**	**R**	**D2**	**D1**	**D0**
0	0	0	0	1	1	1
0	0	0	1	1	1	1
0	0	1	0	1	1	1
0	0	1	1	0	0	0
0	1	0	0	1	1	1
0	1	0	1	0	0	1
0	1	1	0	1	1	1
0	1	1	1	0	1	0
1	0	0	0	1	1	1
1	0	0	1	0	1	1
1	0	1	0	1	1	1
1	0	1	1	1	0	0
1	1	0	0	1	1	1
1	1	0	1	1	0	1
1	1	1	0	1	1	1
1	1	1	1	1	1	0

En base a esta tabla de verdad, es posible determinar las funciones simplificadas a partir de los diagramas de Karnaugh

D0:

Q0 R \ Q2 Q1	00	01	11	10
00	1	1	1	1
01	1	1	1	1
11	0	0	0	0
10	1	1	1	1

D1:

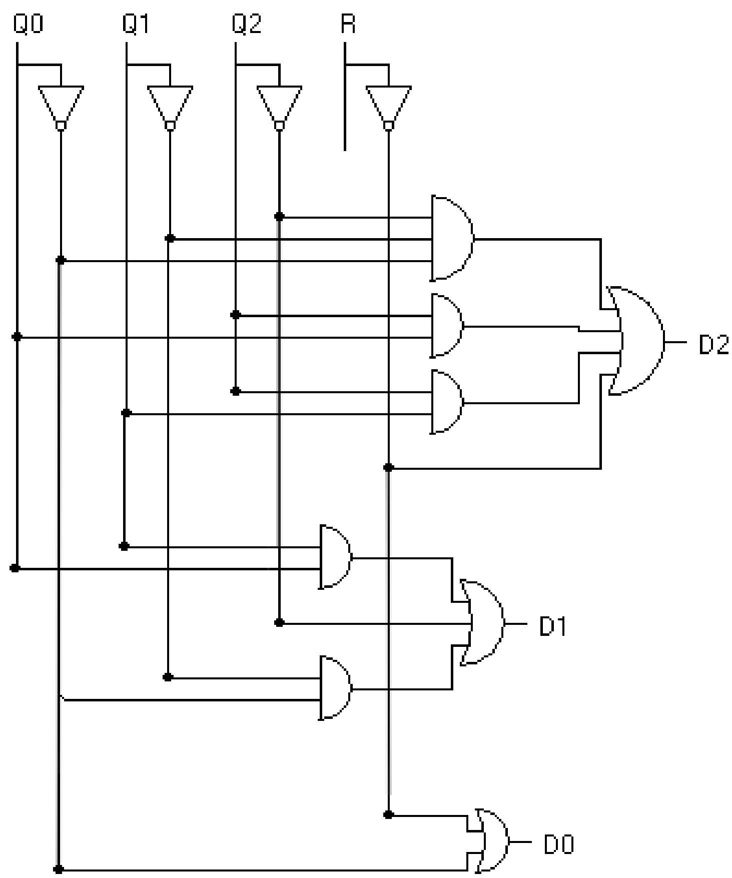
Q0 R \ Q2 Q1	00	01	11	10
00	1	1	1	1
01	1	0	0	1
11	0	1	1	0
10	1	1	1	1

D2:

Q0 R \ Q2 Q1	00	01	11	10
00	1	1	1	1
01	1	0	1	0
11	0	0	1	1
10	1	1	1	1

Expresiones simplificadas:
D0 = (R’)+(Q0’)
D1 = (Q1.Q0)+(R’)+(Q1’.Q0’)
D2 = (Q2’.Q1’.Q0’)+(Q2.Q0)+(Q2.Q1)+(R’)

La implementación circuital del combinacional de estados queda conformada de la siguiente manera:



Vale recordar que no es necesaria la implementación de un combinacional de salida para esta solución, ya que las salidas del contador (Out2, Out1, Out0) se corresponden de forma directa con las salidas de los FF’s (Q2, Q1, Q0).