

Práctica 3

Núcleo del camino de datos de un procesador y buffer circular

Carlota Catot
Miguel Antunez

Grupo 7

Quatrimestre primavera 2020-2021

Índice

1. Pregunta 1	2
1.1. Ventana temporal	2
1.2. Explicación ventana temporal	2
2. Pregunta 2	3
2.1. Esquema RTL	3
2.2. Identificación del esquema	3
2.2.1. Autómata principal	3
2.2.2. Autómatas secundarios	3
3. Pregunta 3	4
3.1. Cronograma	4
3.2. Tiempo de ciclo	4
4. Pregunta 4	5
4.1. Cronograma	5
4.2. Tiempo de ciclo	5
5. Pregunta 5	6
6. Pregunta 6	7
7. Pregunta 7	9

1. Pregunta 1

Considere el proyecto del diseño del banco de registros. Entregue una copia de la ventana temporal de la simulación que muestre el retardo de los componentes del Banco de registros involucrados en una operación de lectura por el puerto 1. Muestre en el mismo diagrama los retardos de los componentes involucrados en una operación de escritura. Para ello, modifique el programa de prueba si es necesario. Utilice los cursores para mostrar los retardos.

1.1. Ventana temporal

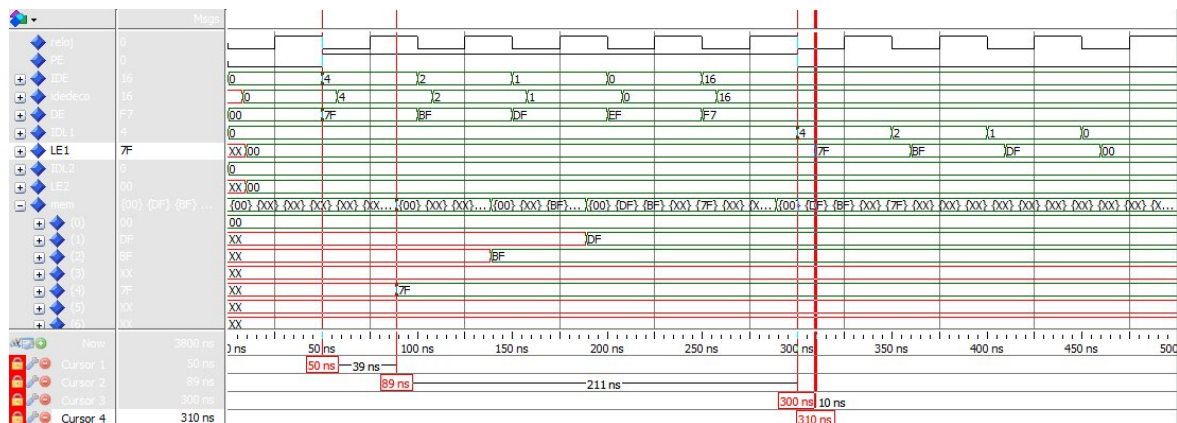


Figura 1: Ventana temporal ejecución Banco Registros

1.2. Explicación ventana temporal

En la figura 1 podemos ver los tiempos de retardo entre la escritura y la lectura de los registros, como podemos ver en los diferentes cursores:

- El cursor 1, que se encuentra en los 50 ns, se lee el registro 4 con el valor 7F.
- El cursor 2, que se encuentra en los 89 ns, podemos ver como en el registro 4 aparece el valor 7F.
- El cursor 3, que se encuentra en los 300 ns, se accede en al registro 4 para escritura.
- El cursor 4, que se encuentra en los 310 ns, se escribe el valor 7F.

2. Pregunta 2

Considere el módulo de control que genera la secuencia de operaciones con dependencias. Entregue una copia del esquema RTL de la unidad de control efectuada por Quartus. Identifique en el esquema la lógica de próximo estado del autómata principal y de los cuatro autómatas subordinados.

2.1. Esquema RTL

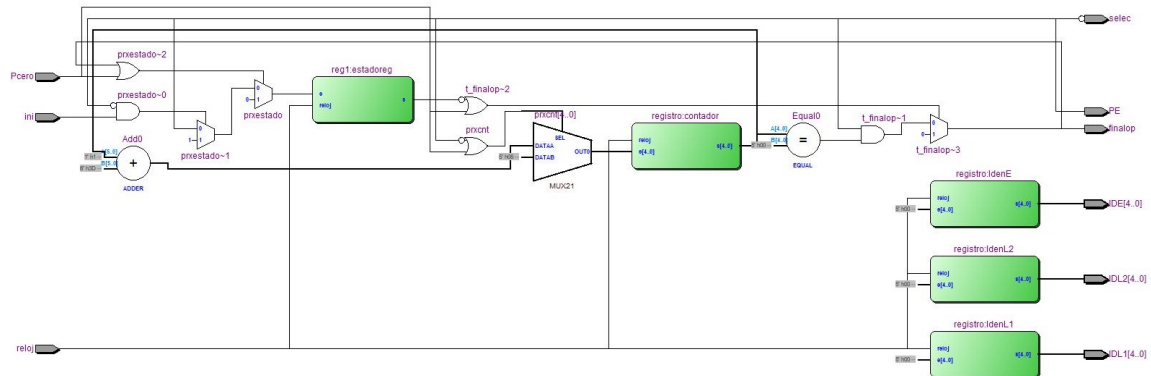


Figura 2: Esquema RTL del proyecto camino_control.qpf

2.2. Identificación del esquema

2.2.1. Autómata principal

```
prxestado <= '0' when pCero = '1' or t_finalop = '1'
           else '1' when (ini = '1' and estado = '0')
           else '1' when estado = '1' else '0';
```

2.2.2. Autómatas secundarios

```
IF PE = 1 AND contador < 9 THEN IDL1 = contador
IF PE = 1 AND contador < 18 THEN IDL2 = contador
IF PE = 1 AND contador < 27 THEN IDL3 = contador
```

3. Pregunta 3

Suponga la secuencia de operaciones independientes. Marque, oscureciendo las casillas, los retardos de cada componente para el segundo ciclo de la secuencia de operaciones ($IDL1 = 1$, $IDL2 = 10$, $IDE = 19$). El retardo de un componente se empieza a marcar a partir del instante de tiempo en el cual todas las señales de entrada, que determinan el valor de su salida, son válidas. En un ciclo de la señal Reloj, entendemos que una señal es válida a partir del instante de tiempo en que no se modifica.

3.1. Cronograma

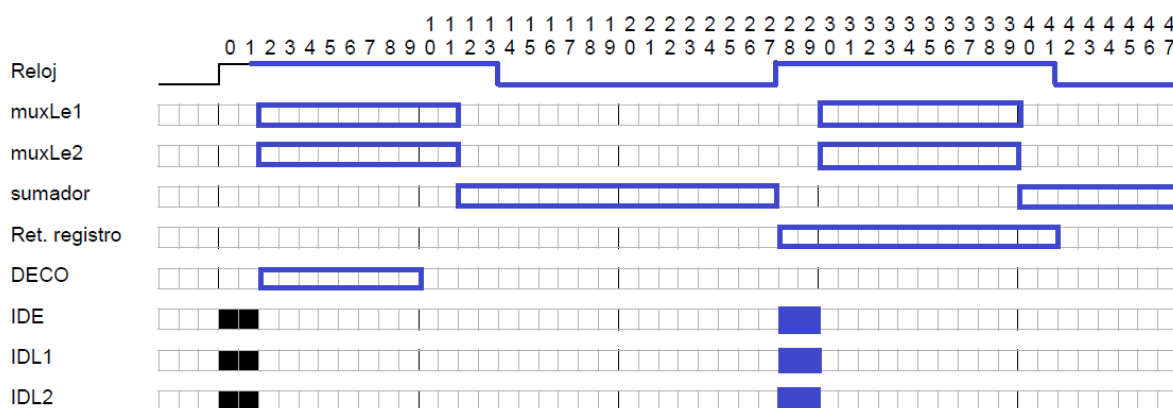


Figura 3: Cronograma pregunta 3

3.2. Tiempo de ciclo

Determine el periodo mínimo de la señal Reloj suponiendo que es cuadrada. Muestre este periodo dibujando en la fila etiquetada como Reloj los intervalos de tiempo en los cuales la señal Reloj está en el nivel lógico 0 y en el nivel lógico 1.

El tiempo de ciclo mínimo es de **28 ns**. En la figura 3 podemos ver en el cronograma en la señal de reloj que el semiperiodo corresponde a los 14 ns y el periodo completo corresponde a 28 ns. Podemos ver que los 28 ns acaban con el final de la operación del sumador, durante los 28 ns podemos ver que los 2 primeros ns tenemos el retardo del flanco de subida, en los 10 siguientes corresponden al retardo del multiplexor, y por último en los siguientes 16 ns. Por tanto $2 + 10 + 16 = 28$ ns que corresponde al tiempo de ciclo.

4. Pregunta 4

Suponga la secuencia de operaciones dependientes (página 186). Marque, oscureciendo las casillas, los retardos de cada componente para el segundo ciclo de la secuencia de operaciones ($IDL1 = 1$, $IDL2 = 10$, $IDE = 10$).

4.1. Cronograma

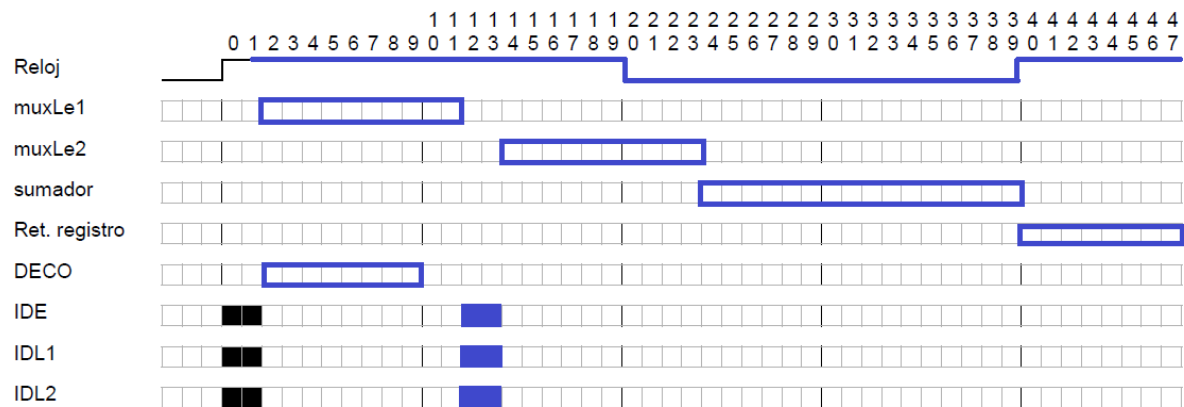


Figura 4: Cronograma pregunta 4

4.2. Tiempo de ciclo

Determine el periodo mínimo de la señal Reloj suponiendo que es cuadrada. Muestre este periodo dibujando en la fila etiquetada como Reloj los intervalos de tiempo en los cuales la señal Reloj está en el nivel lógico 0 y en el nivel lógico 1.

El tiempo de ciclo es de **40 ns**. En la figura 4 podemos ver en el cronograma la señal de reloj que el semiperiodo corresponde a los 20 ns y el periodo completo corresponde a 40 ns. Podemos ver que, como en el ejercicio anterior, a los 40 ns acaban con el final del sumador.

5. Pregunta 5

Entregue una copia del contenido de la ventana temporal que se obtiene con Modelsim cuando el camino de datos procesa la secuencia de operaciones dependientes. El objetivo es mostrar que el periodo del reloj es correcto. Por tanto, en la ventana de tiempo se debe poder identificar claramente el periodo de la señal de reloj. Identifique en el cronograma el ciclo correspondiente a la segunda operación y muestre los retardos de los componentes mediante cursores.

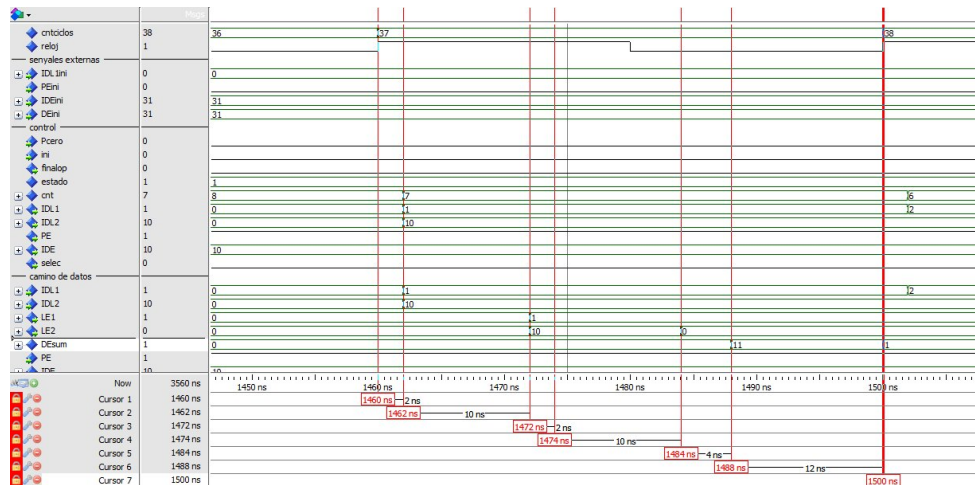


Figura 5: Ventana temporal con los retardos del ejercicio 4 marcados

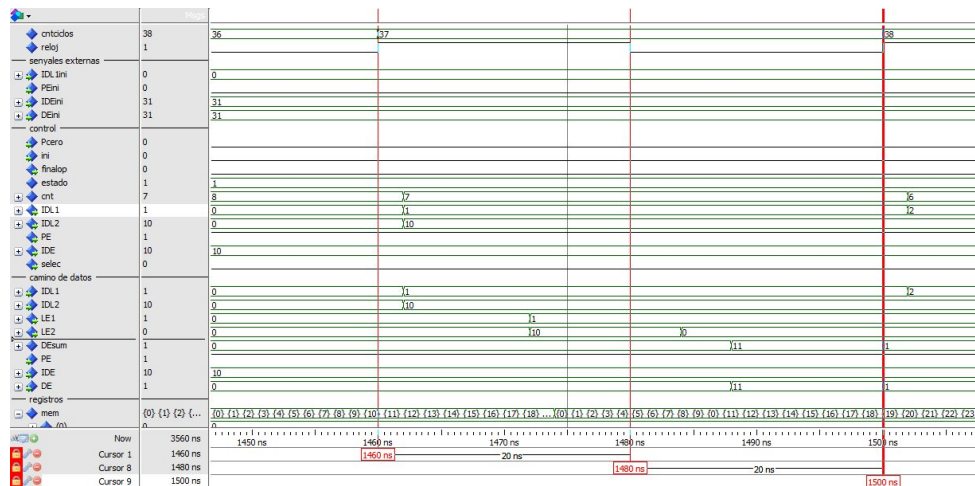


Figura 6: Ventana temporal con el semiperiodo y el periodo marcados

En las ventanas temporales de las figuras 5 i 6 podemos ver de una manera gráfica el cronograma de la pregunta hecho de manera manual en la pregunta anterior. Para que nos pueda salir esto en la ventana temporal se ha tenido que modificar el código de *camino_control.vhd* para que haya dependencia entre las operaciones.

6. Pregunta 6

En el programa de prueba asociado al diseño del elemento de acceso al banco de registros del buffer circular se planifican 4 acciones de escritura. Sin embargo, al final de la simulación se observa que el valor del puntero cola es 5, por tanto, se efectúan 5 escrituras. Justifique este comportamiento. Analice los procesos *estimulos* y *gen_reloj* del programa de prueba y los procedimientos involucrados.

El código que se usa para hacer la simulación tiene los procesos *estimulos* y *gen_reloj* como se indica en el enunciado, pero para poder justificar el porqué aparece la 5a escritura, nos centraremos únicamente en el código de *estimulos* presentado a continuación:

```
estimulos: process
begin
    pzero <= '1';
    escritura <= '0';
    lectura <= '0';

    wait_until_falling_edges(reloj,2);
    pzero <= '0';

    escribir (escritura, lectura, reloj);
    escribir (escritura, lectura, reloj);
    desactivar (escritura, lectura, reloj);
    leer (escritura,lectura, reloj);
    desactivar (escritura, lectura, reloj);
    conclesclec (escritura, lectura, reloj);
    conclesclec (escritura, lectura, reloj);

    wait_until_falling_edges(reloj,1);

    report "Prueba finalizada.";
    final := true;
    wait;
end process;
```

Este código genera la ventana temporal que podemos observar en la figura 7 se puede ver como el valor de cola llega a 5 al final de la ejecución, esto es debido a que en ningún momento se desactiva la escritura y estamos dando tiempo al programa con la sentencia "*wait_until_falling_edges(reloj,1);*" de seguir procesando dichas lecturas durante un flanco extra.

Con tal de evitar este comportamiento, después de el bloque de lecturas y escrituras podríamos añadir la siguiente instrucción "*desactivar (escritura, lectura, reloj);*", de esta manera el contador se pararía en 4 y finalizaría la ejecución. Podemos ver la ventana temporal correspondiente a la solución planteada para que no pase en la figura 8

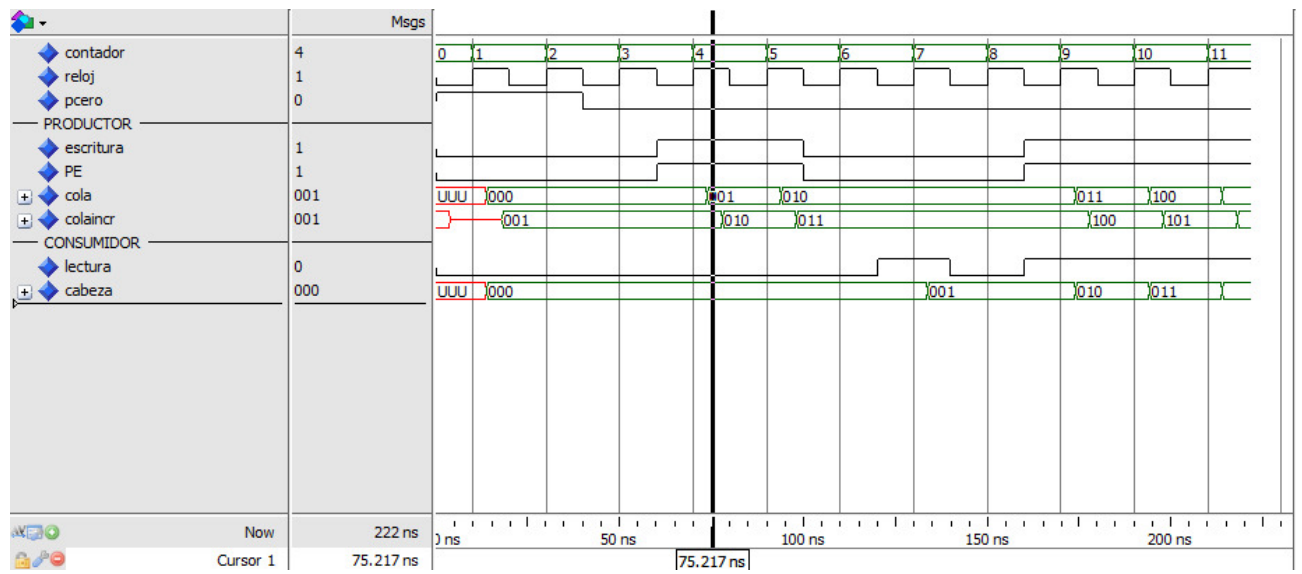


Figura 7: Ventana temporal original

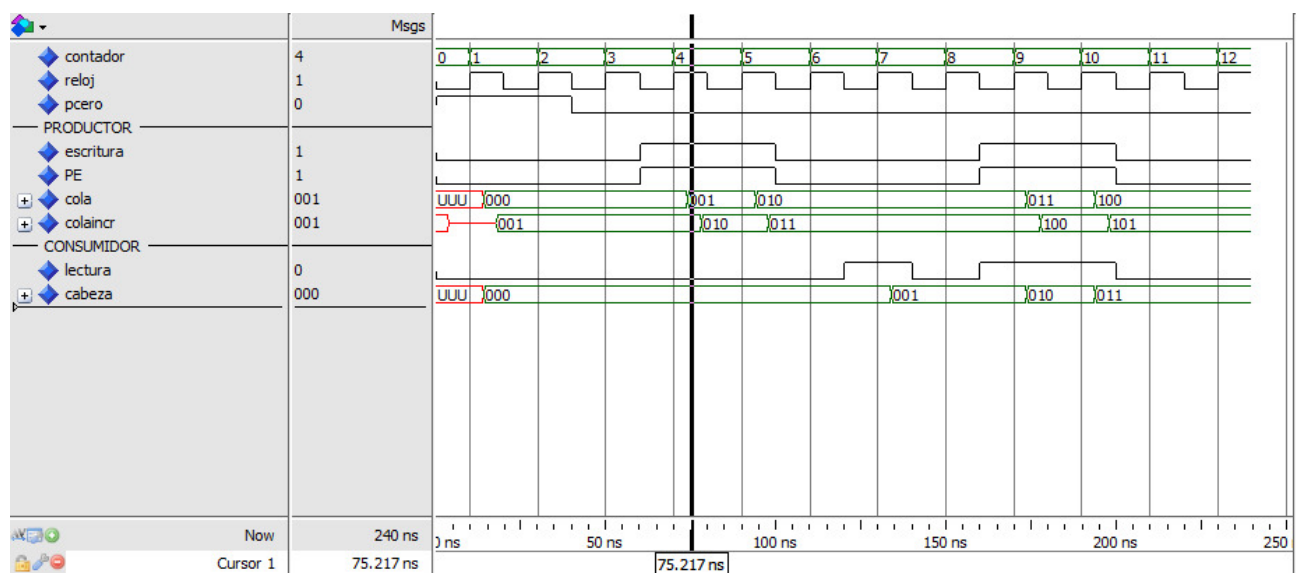


Figura 8: Ventana temporal desactivando escrituras al final de la ejecución

7. Pregunta 7

Indique el retardo necesario en una acción de actualización del buffer circular. Considere el diseño que no utiliza todas las entradas.

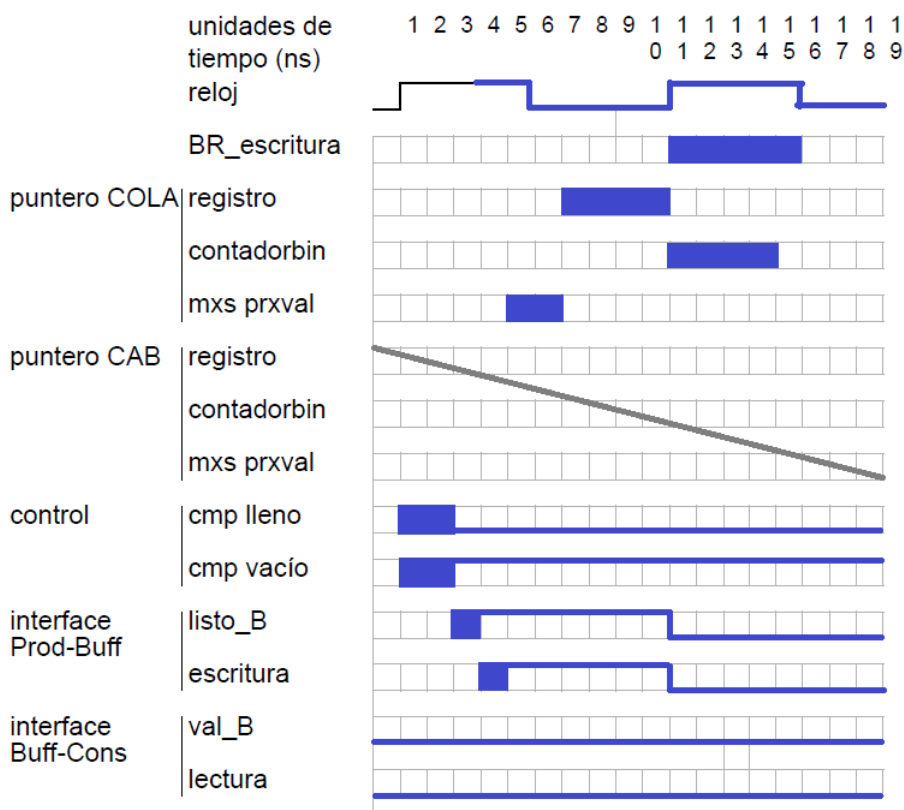


Figura 9: Cronograma pregunta 7

Para rellenar el cronograma hemos seguido el esquema de la figura 10 para simular la ejecución del código. El retardo de actualización de buffer circular, como podemos ver en el cronograma es de **10 ns**.

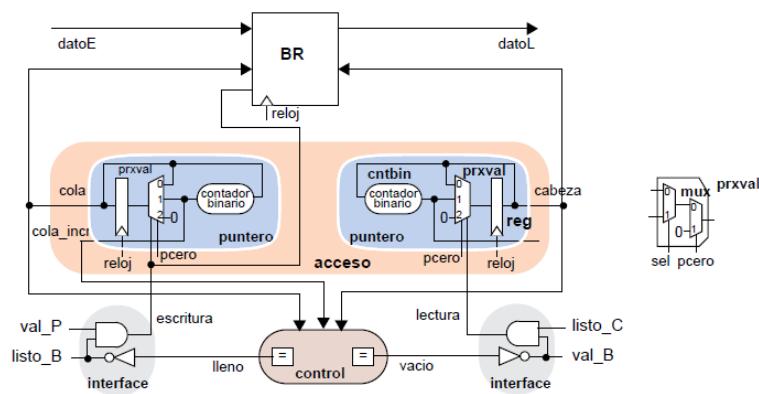


Figura 3.37 Implementación del buffer circular. Elementos constituyentes.

Figura 10: Cronograma pregunta 7