

**Ej. 1** — (2.0 puntos) Diseñar la ruta de datos y la unidad de control (diagrama ASM y tabla de salidas) de un sistema algorítmico que calcule hasta el  $n$ -ésimo término,  $f_n$ , de la sucesión de Fibonacci. Las ecuaciones (1-3) definen la relación de recurrencia que genera la sucesión de Fibonacci y el pseudo-código se incluye a continuación:

$$\begin{array}{ll}
 f_i &= f_{i-1} + f_{i-2} & (1) \\
 f_{-1} &= 0 & (2) \\
 f_0 &= 1 & (3)
 \end{array}$$

```

 $f_{-1} \leftarrow 0;$ 
 $f_0 \leftarrow 1;$ 
for  $i=1, i \leq n, i++$  do
     $f_i \leftarrow f_{i-1} + f_{i-2};$ 
     $\text{mem}(i) \leftarrow f_i;$ 
end
 $\text{fib} \leftarrow f_n;$ 

```

El sistema tiene cuatro entradas y dos salidas. Las entradas son: `clk`, `rst`, `ini` y `n`. Supóngase que el valor de `n` está codificado en binario puro y que siempre será mayor que cero. Las salidas son: `fib` y `fin`, de forma que la salida `fib` sea el  $n$ -ésimo elemento de la sucesión. El sistema comienza a funcionar cuando la señal `ini` se pone a 1. En ese instante se almacena el valor de `n` en un registro interno del camino de datos. Cuando se complete el cálculo el sistema volverá al estado inicial, en el que la señal `fin` es igual a 1. Los términos de la sucesión de Fibonacci se irán almacenando en una memoria SRAM síncrona de 64x48 bits con un único puerto de lectura y escritura y modo de funcionamiento `READ_FIRST`.

```

entity asm is
    port (clk    : in  std_logic;
          rst    : in  std_logic;
          ini    : in  std_logic;
          n      : in  std_logic_vector(5 downto 0);
          fib    : out std_logic_vector(47 downto 0);
          fin    : out std_logic
        );
end asm;

```

```

entity ram is
    port (clk    : in  std_logic;
          dina    : in  std_logic_vector(47 downto 0);
          addra   : in  std_logic_vector(5 downto 0);
          wea     : in  std_logic;
          ena     : in  std_logic;
          douta   : out std_logic_vector(47 downto 0)
        );
end ram;

```

En la ruta de datos se puede usar una memoria RAM<sup>(1)</sup> síncrona de un solo puerto, registros, un único sumador, y los elementos combinatoriales adicionales que se consideren necesarios.

<sup>(1)</sup>: memoria SRAM síncrona de 64x48 bits, single-port y modo de escritura `READ_FIRST`.



TEC. COMPUT. // TEC. Y ORGA. COMPUT.

2º

GII — GIC

08/02/2016

**Ej. 2** — (1.0 puntos) Diseñar utilizando puertas lógicas de dos entradas:

1. Una red iterativa 1D que dado un vector  $x$  de  $n$  bits que sólo contiene dos 1 genere una salida,  $z$ , de forma que  $z = 1$  si los dos bits que contienen el valor 1 están separados por un número par de bits y  $z = 0$  en caso contrario. Si los dos 1 están consecutivos en el vector  $x$  entonces se entiende que están separados por cero bits y por tanto la salida  $z = 1$ .
2. Suponiendo que todas las puertas de dos entradas tienen un retardo  $t$ , calcular el retardo del camino crítico en el diseño del apartado anterior para  $n = 16$ . En el caso de haber usado inversores, suponer que su retardo es despreciable.
3. Indicar qué modificaciones habría que realizar en el diseño del apartado 1 en el caso de que el patrón a detectar al comienzo y al final fuese dos 1 consecutivos.

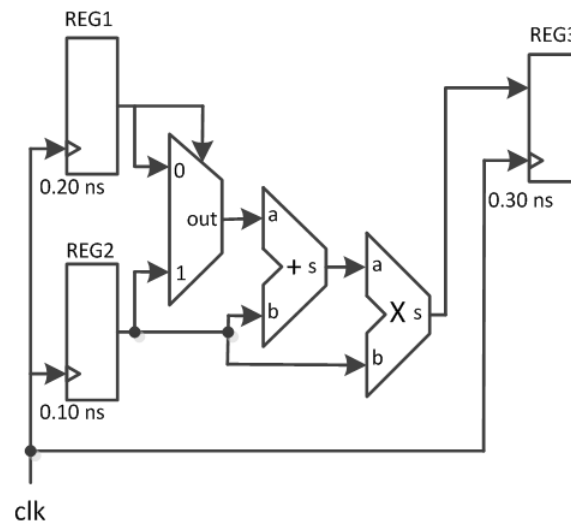
TEC. COMPUT. // TEC. Y ORGA. COMPUT.

2º

GII — GIC

08/02/2016

**Ej. 3** — (1.5 puntos) Sea el circuito de la figura, y los valores de propagación de sus componentes los siguientes:  $ADD(a \rightarrow s) = 2,75$  ns,  $ADD(b \rightarrow s) = 2,60$  ns,  $MUL(a \rightarrow s) = 2,75$  ns,  $MUL(b \rightarrow s) = 2,60$  ns,  $MUX(sel \rightarrow out) = 0,50$  ns,  $MUX(0 \rightarrow out) = 0,50$  ns y  $MUX(1 \rightarrow out) = 0,40$  ns. Los valores en las líneas de reloj son el retardo de propagación desde la fuente de reloj. Los parámetros de los registros son:  $t_{clk-2-q} = 0,10$  ns,  $t_{setup} = 0,15$  ns,  $t_{hold} = 0,20$  ns.



1. ¿Podría este circuito funcionar correctamente a 250 MHz? Justificar la respuesta.
2. Supóngase que el retardo  $MUL(a \rightarrow s)$  es variable. Indicar el rango de valores de este parámetro para los que el circuito pueda funcionar correctamente a 250 MHz.
3. Asumiendo que el retardo de  $MUL(a \rightarrow s)$  es 2,75 ns, para que el circuito pudiera funcionar a una frecuencia de 250 MHz, habría que segmentarlo. Indicar dónde hay que añadir el/los registro(s) de segmentación para que fuera posible; y demostrar si, efectivamente, se consigue hacer funcionar el circuito a 250 MHz o no. ¿Se introducen nuevas violaciones de hold con esta modificación? Si es así, ¿cómo se podrían resolver? Suponer que el retardo de reloj de el/los nuevo(s) registro(s) es de 0,30 ns.

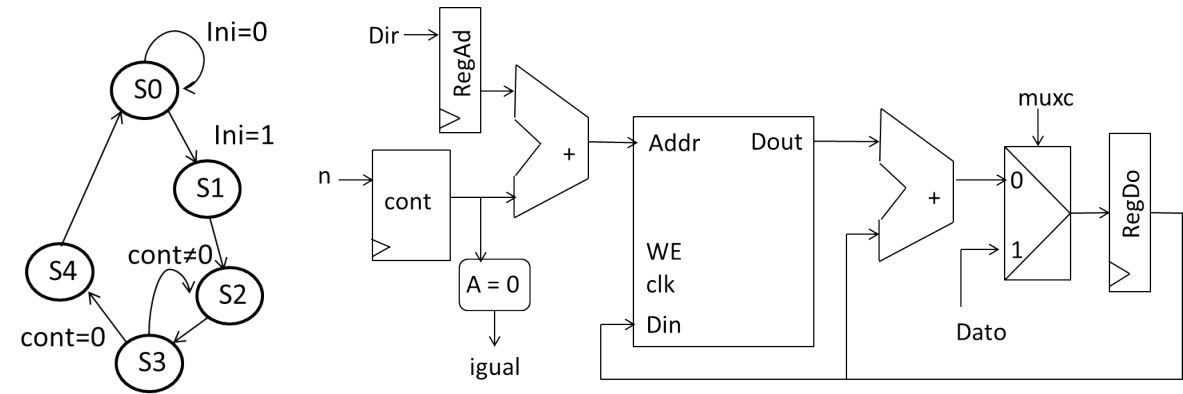
TEC. COMPUT. // TEC. Y ORGA. COMPUT.

2º

GII — GIC

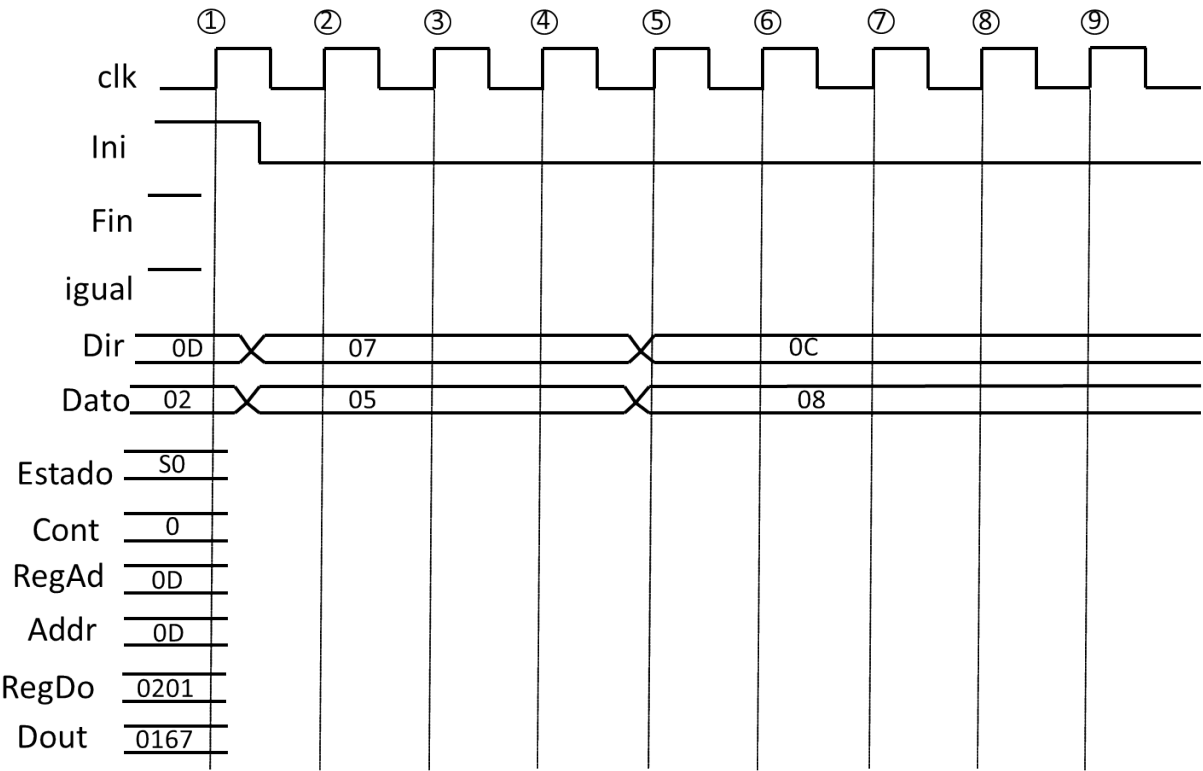
08/02/2016

Ej. 4 — (1.5 puntos) Completar el cronograma correspondiente al siguiente sistema e indicar el contenido final de la memoria. Las entradas al sistema son: Ini, Dato, Dir, n, clk y rst (=0 en el cronograma); la salida del sistema es Fin, el resto de señales son señales internas según tabla adjunta. Considerar que la memoria es síncrona tanto para lectura como para escritura, en la memoria se escribe cuando WE es '1'. Suponer que el contador es un contador módulo 8 descendente, la memoria es READ\_FIRST y que la señal n = 2.



	IdDo	IdAd	IdCont	CE	WE	muxc	fin
S0	0	0	0	0	0	0	1
S1	1	1	1	0	0	1	0
S2	1	0	0	0	0	0	0
S3	0	0	0	1	0	0	0
S4	0	0	0	0	1	0	0

Dirección	Dato
0x07	0x10DC
0x08	0x0102
0x09	0x000A
0x0A	0x1255
0x0B	0x0134
0x0C	0xF5F4
0X0D	0xFFED
0x0E	0X44CB





**Ej. 5 — (0.5 puntos)** Contestar brevemente las siguientes dos cuestiones:

1. Explicar los elementos que constituyen una celda DRAM, su función y enumerar las ventajas y desventajas que presentan respecto a las celdas de tipo SRAM.

2. Se desea implementar una memoria de tamaño 2Kx8 bits, a partir de una Block RAM de Xilinx de tamaño 512x32 bits. ¿Cuál sería el hardware adicional necesario para realizar correctamente las operaciones de lectura sobre dicha Block RAM? Realizar un esquema indicándolo.