

1. Describir en VHDL el funcionamiento del dispositivo que responde al cronograma de la **Error! No se encuentra el origen de la referencia.. (1pto)**

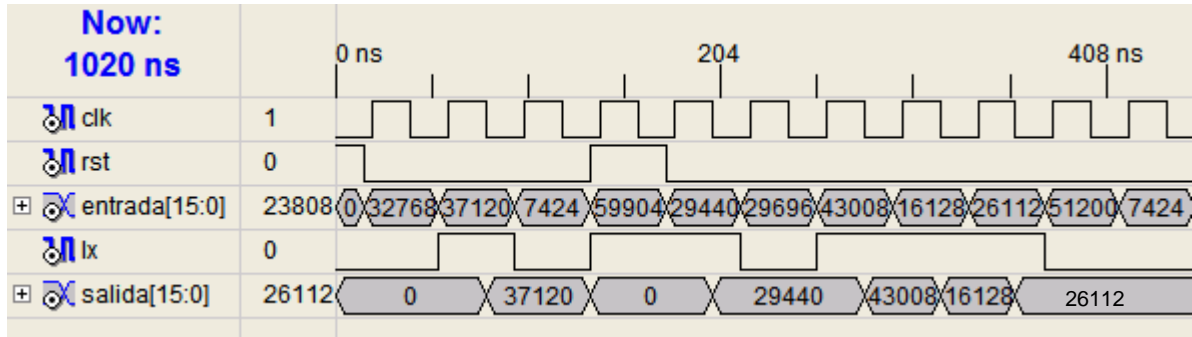


fig. 1

2. Disponemos de un procesador que genera una señal de lectura escritura definida como R/\overline{W} , cuyo bus de datos es de 16bits y el de direcciones de 32bits. Disponemos de memorias tipo SRAM de 32Mx16 y 64Mx16 cuya tabla de funcionamiento se muestra al lado. Diseñar un mapa de memoria con todas las señales para conseguir direccionar 256Mx16 que empiecen en la dirección A0000000H.

SRAM			
\overline{CS}	\overline{OE}	\overline{WE}	FUNCIÓN
1	X	X	Z/PowerDown
0	0	1	LECTURA
0	X	0	ESCRITURA
0	1	1	Z

Se pide:

- a) Tabla de direcciones **(1pto)**
b) Mapa de conexiones con todas las señales **(1,5ptos)**

NOTA:

Utiliza el menor número posible de memorias.

Utiliza el decodificador apropiado para no repetir direcciones

3. Describir en VHDL el registro HL de la fig. 2, teniendo en cuenta que la señal **clear** es asíncrona y la señal **LH**, carga en la parte **alta** el contenido de la entrada y que la señal **LL** carga en la parte **baja** el contenido de la entrada, las señales **LH** y **LL** son síncronas con el flanco de subida del reloj. **(2ptos)**

NOTA:

Observa que la entrada es de 8 bits y la salida de 16bits.

La salida está presente desde el momento que se cumple las condiciones de carga.

4. Disponemos de la siguiente descripción VHDL. Dibuja el hardware que describe, se pueden utilizar puertas, multiplexores, decodificadores, demultiplexores, biestables, registros, etc **(1pto)**

```
library ieee;
use ieee.std_logic_1164.all;
```

```
entity examen is
```

```
port (Clk : in std_logic;
      A : in std_logic_vector(3 downto 0);
      B : in std_logic_vector(3 downto 0);
      S : in std_logic;
      Y : out std_logic);
```

```
end examen;
```

.../...

```
architecture RTL of examen is
```

```
begin
```

```
process(Clk)
```

```
begin
```

```
if rising_edge(Clk) then
```

```
if S = '1' then
```

```
Y <= A(0) and A(1) and A(2) and A(3);
```

```
else
```

```
Y <= B(0) or B(1) or B(2) or B(3);
```

```
end if;
```

```
end if;
```

```
end process;
```

```
end RTL;
```



5. Disponemos de la ruta de datos, fig. 2, Se pide:

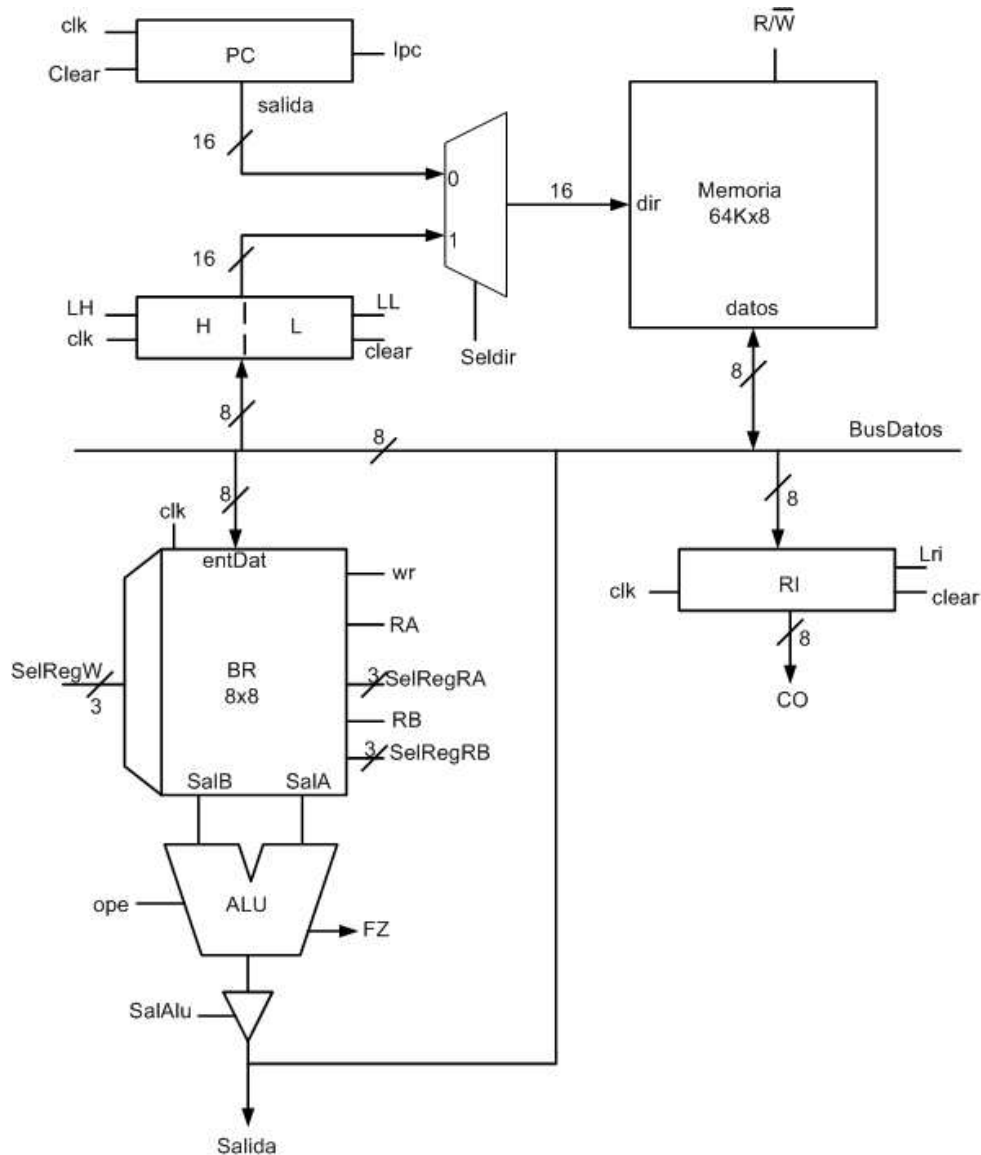


fig. 2

a) Cronograma, con el menor número posible de periodos de reloj, de la instrucción **Store dir**, definida como $M(dir) \leftarrow R5$ (1,5ptos)

NOTA: dir tiene 16 bits y se guarda en memoria primero el byte bajo

b) ¿Se puede ejecutar en esta ruta de datos la instrucción **JZ dir?**, definida como
Si $Z=1$ entonces $PC \leftarrow dir$
si no $PC \leftarrow PC+2$

Poner una cruz donde corresponda

(0,25ptos)

Sí ☐

NO ☐

c) En caso negativo, realiza los cambios hardware para que se pueda ejecutar, (en el propio dibujo del examen, hay que entregar esta hoja) (0,25ptos)

d) Cronograma, con el menor número posible de periodos de reloj, de la instrucción **JZ dir** (1,5ptos)