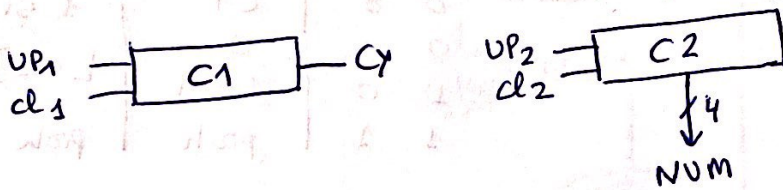
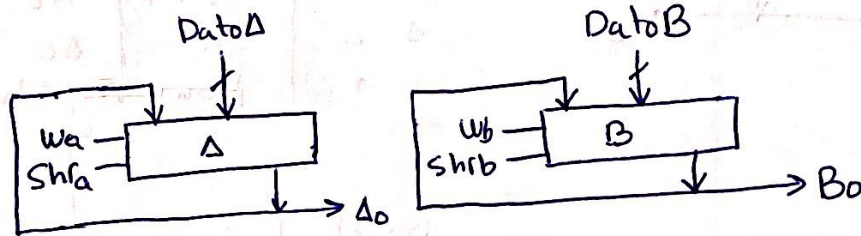


Problema 1

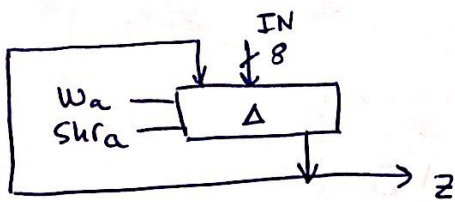
1) Unidad de datos



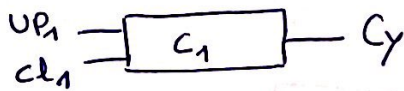
He añadido un contador módulo 8 para contar los 8 desplazamientos que habrá que hacer a Δ y a B para acceder a todos sus bits, dicho contador es C1 y tiene salida de Cy.

He añadido un contador módulo 16 para contar el nº de bits que coinciden en los dos registros, este número puede oscilar entre 0 y 8, por tanto no bastará con un contador módulo 8. Este contador es C2 y no necesita salida de Cy pero si necesita un bus de salida para mostrar el número almacenado: este bus es NUM y será salida de la Unidad de Datos.

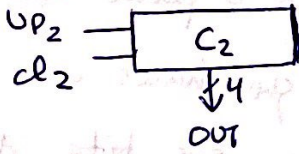
2) Descripción RT



Wa	Shra	$\Delta \leftarrow$	$Z =$
0	0	A	Δ_0
0	1	$\text{SHR}(A, i_P)$	Δ_0
1	0	IN	Δ_0
1	1	proh	proh

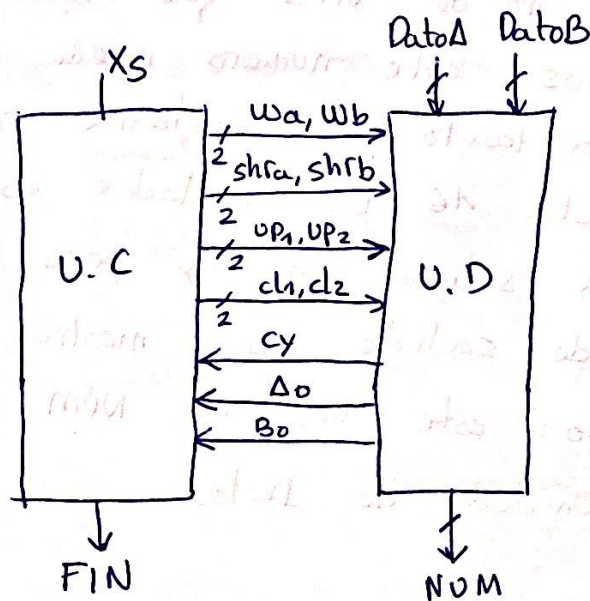


UP ₁	cl ₁	$C1 \leftarrow$	$Cy =$
0	0	C1	1 si
0	1	0	$[C1] = 7$
1	0	$C1 + 1$	
1	1	proh	proh

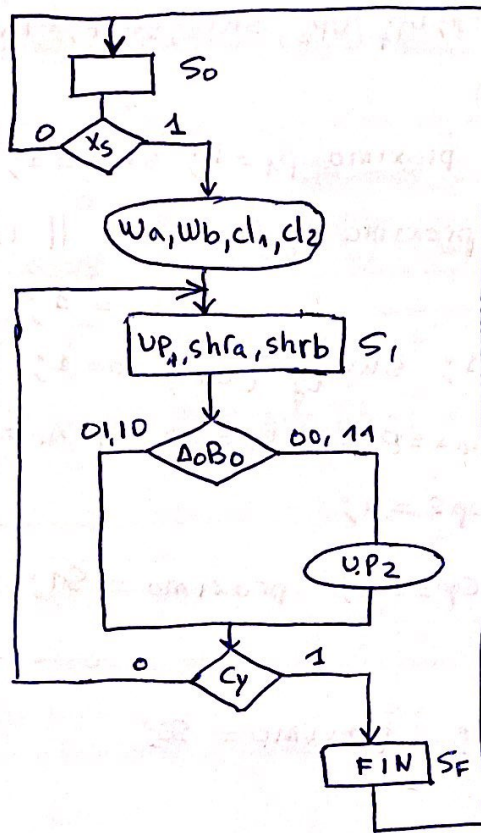


UP ₂	cl ₂	$C2 \leftarrow$	OUT =
0	0	C2	$[C2]$
0	1	0	$[C2]$
1	0	$C2 + 1$	$[C2]$
1	1	proh	proh

3) Diagrama de bloque



4) Carte ASM



module ucontrol (input xs, reset, ck, Δ0, B0, Cy,
output wa, wb, cl1, cl2, up1, shra, shrb,
up2, FIN);

parameter S0 = 2'b00, S1 = 2'b01, SF = 2'b10;

reg [1:0] presente, proximo;

always @ (posedge ck)

if (reset) presente = S0;

else presente = proximo;

always @ (xs, A0, B0, Cy, presente)

begin

{ wa, wb, d1, d2, up1, up2, shra, shrb, FIN } = 0;

case (presente)

S0: if (xs) begin wa=1; wb=1; d1=1; d2=1; proximo=S1; end

else proximo = S0;

→ se puede no poner

S1: begin

up1=1; shra=1; shrb=1;

if (A0==0 && B0==0 || A0==1 && B0==1)

up2=1;

if (Cy==1) proximo = SF;

end

SF: begin

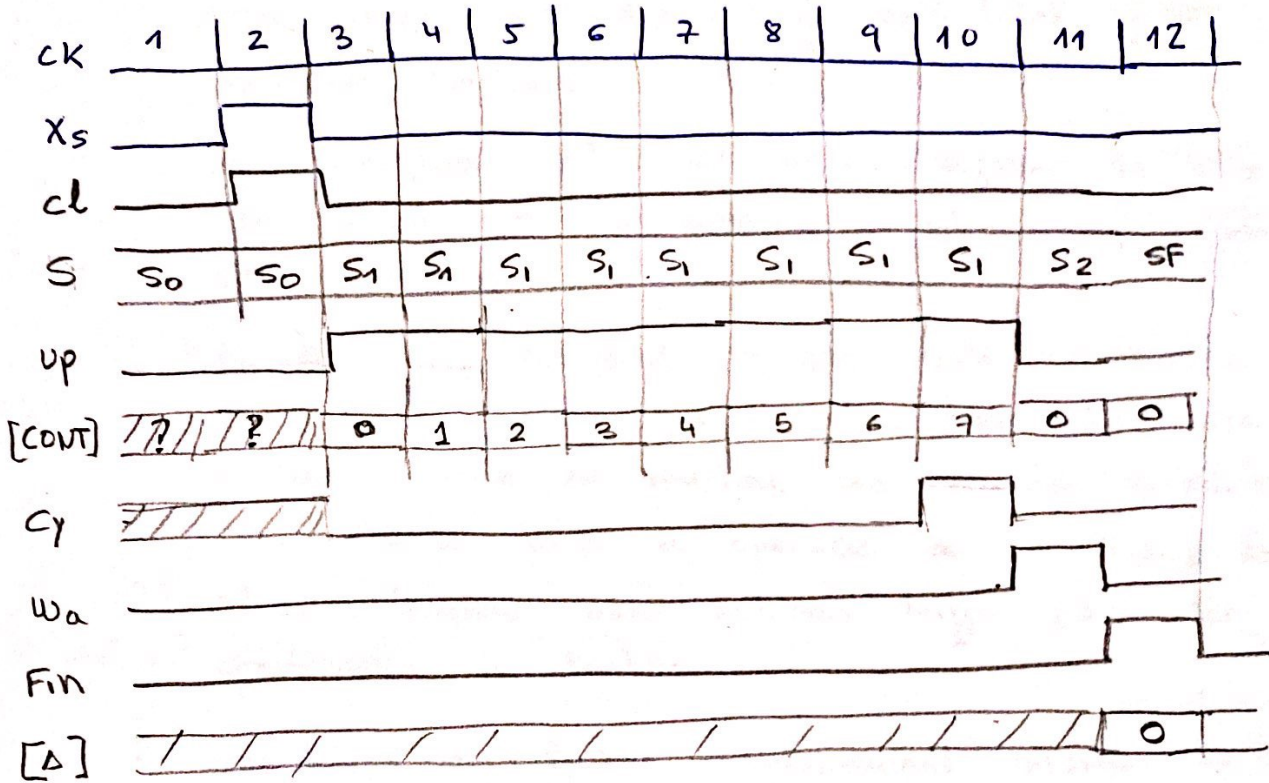
FIN=1; proximo = S0;

end

endcase

end module

Problema 2



Teoría

- a. Debe hacerse una descripción estructural y una descripción funcional.

La descripción estructural es un diagrama de bloque que muestra todas las entradas y salidas del módulo con su nombre.

La descripción funcional es una tabla que muestra el modo de operar del circuito. Debe incluir una columna donde se muestran las señales de control, otra columna para la operación de escritura y tantas otras columnas como salidas haya para las operaciones de lectura.

- b. Los buses de lectura incondicional muestran en todo momento su valor, en cambio, los de lectura condicional solo lo hacen si una señal de control se activa.

Cuando un bus ha de conectarse a un bus compartido debe ser de lectura condicional con el objetivo de poder desconectarse cuando haya algún ^{otro} dispositivo volcando sus datos al bus.

- c. bus dedicado → solo un dispositivo puede escribir en este bus

bus compartido → diferentes dispositivos pueden escribir en el bus, por tanto, deben turnarse.

- bus triestado \rightarrow puede tomar los valores 0, 1 y HI
- bus estándar \rightarrow solo puede tomar los valores 0, 1
- bus unidireccional \rightarrow se puede identificar unívocamente al dispositivo fuente y al destino
- bus bidireccional \rightarrow los dispositivos fuente y destino pueden ser intercambiables

d, 1. No es verdadera, no siempre una macrooperación ha de constar de varias microoperaciones

2. Falso, una microoperación es una operación que se hace en un ciclo de reloj.

3. Verdadera, si las microoperaciones no escriben en el mismo bus, pueden ser simultáneas.

4. Falso, hay macrooperaciones que se pueden ejecutar en un ciclo de reloj (equivalente al caso 1)

5. No siempre, a veces es el incremento de un contador, o el desplazamiento en un registro.

e. La caja de estados es la única que debe aparecer obligatoriamente y la que da inicio al bloque ASM. Puede contener acciones a realizar.

La caja de condición permite diversificar caminos en función del cumplimiento de una condición que es evaluada.

La caja de acción condicional aparece siempre tras una caja de condición en caso de que haya que realizar acciones, estas aparecen en el interior de la caja.

Las formas de las cajas son también diferentes, la primera rectangular, la segunda rombo y la tercera ovalada.

f. Una unidad de datos es una colección de registros y elementos combinatoriales conectados mediante buses y líneas con el fin de realizar alguna tarea de procesamiento de complejidad media.

Una unidad de control es una máquina secuencial que genera una secuencia de órdenes mediante activación de señales que son recibidas por una unidad de datos para realizar una tarea concreta.

La unidad de datos puede realizar diferentes tareas, es la unidad de control la que establece cuál es la tarea realizada.

g. Para realizar $R0 \leftarrow 2R4 + R5 + R7$ haremos:

$R0 \leftarrow R4$	01 000 100
$R0 \leftarrow R0 + R4$	00 000 100
$R0 \leftarrow R0 + R5$	00 000 101
$R0 \leftarrow R0 + R7$	00 000 111

h. La secuencia en binario es esta ↵