

ALUMNO: _____

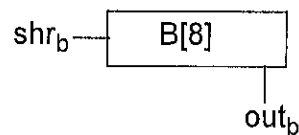
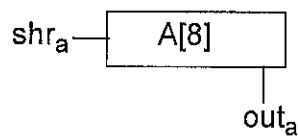
Problema 1

- Indique una clasificación de los computadores según las características de su repertorio de instrucciones (1 punto)
- Defina hardware, software y firmware (1 punto)

Problema 2

Diseñe un sistema digital que cuente el número total de unos que contienen dos de sus registros (A y B, de 8 bits). Para ello:

- Especifique totalmente la unidad de datos, añadiendo los elementos que crea necesario (2 puntos)
- Describa a nivel RT todos los elementos de la unidad de datos (1 punto)
- Represente mediante un diagrama de bloques las entradas y salidas de las unidades de datos y de control (1 punto)
- Obtenga la carta ASM de la unidad de control (2.5 puntos)
- Describa la carta ASM en Verilog (1.5 puntos)



Problema 1

(a) Se clasifican en RISC y CISC

RISC - Reduced Instruction Set Computer

pocas instrucciones, simples, de longitud fija y pocos formatos

CISC - Complex instruction Set Computer

muchas instrucciones, complejas, de longitud variable y muchos formatos

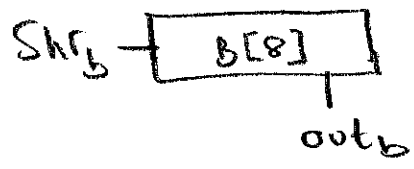
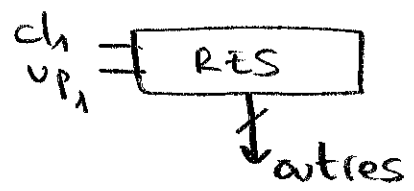
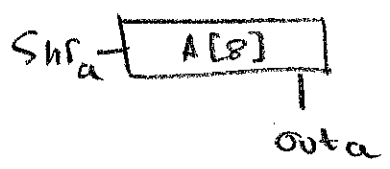
(b) hardware: conjunto de componentes o sistemas electrónicos o mecánicos que componen el ordenador o sus periféricos

software: programas destinados a ser ejecutados por la CPU del ordenador, son cargados en la memoria principal para su ejecución

firmware: programas grabados en memoria de solo lectura, suelen ir incluidos con el hardware 'de serie'.

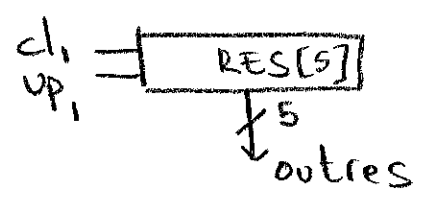
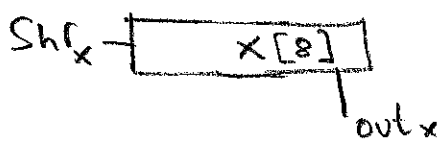
Problema 2

a)



Añado dos contadores; uno de ellos: CONT destinado a contar los desplazamientos realizados sobre los registros A y B, será de módulo 8, el otro: RES con el objeto de contar el número de unos totales en los dos registros ha de ser de módulo 17 como mínimo ya que ha distinguir los casos 0, 1, 2 ... 15 y 16.

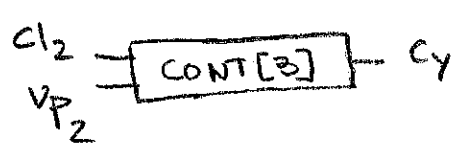
b) Descripción RT



Shrx	X ←	outx =
0	X	X ₀
1	SHR(X,-)	X ₀

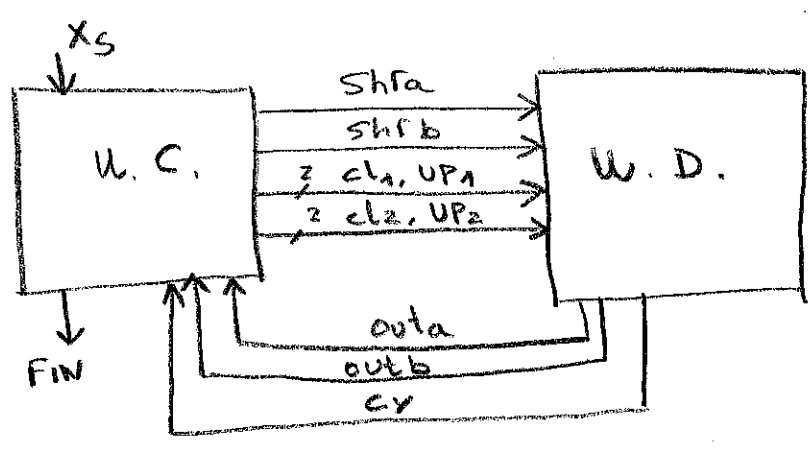
X es A o B

cl1	up1	RES ←	outres =
0	0	RES	[RES]
0	1	RES + 1	[RES]
1	0	0	[RES]
1	1	proh	proh

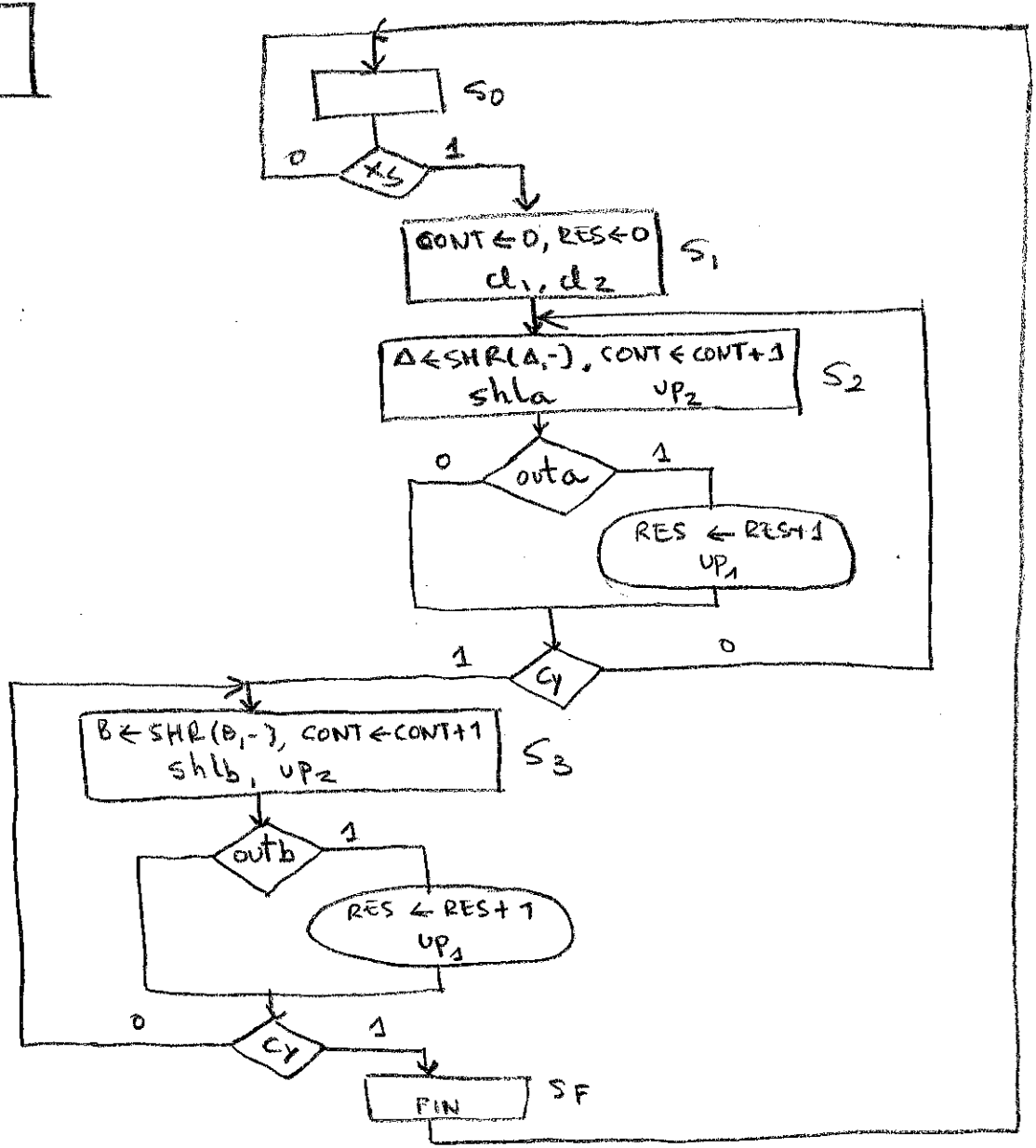


cl ₂	VP ₂	CONT ←	Cy
0	0	CONT	1 si [CONT]=7
0	1	CONT+1	"
1	0	0	"
1	1	proh	proh

c)



d)



e)

```

module mi_carta_asm (input reset, xs, outa, outb, cy,
                    output reg shra, shrb, cl1, up1, cl2, up2);

```

```

parameter S0 = 3'b000,
          S1 = 3'b001,
          S2 = 3'b010,
          S3 = 3'b011,
          SF = 3'b100;

```

```

reg [2:0] current_state, next_state;

```

```

always @(posedge cl or posedge reset)

```

```

    if (reset)
        current_state <= S0;
    else
        current_state <= next_state;

```

```

always @(current_state, xs, outa, outb, cy)

```

```

begin
    shra = 0;
    shrb = 0;
    cl1 = 0;
    cl2 = 0;
    up1 = 0;
    up2 = 0;
    case (current_state)
        S0: if (xs) next_state = S1;
            else next_state = S0;

```

S1: begin cl₁ = 1; cl₂ = 1; next_state = S2; end

S2: begin

shla = 1; up₂ = 1;

if (outa) up₁ = 1;

if (cy) next_state = S2;

else next_state = S3;

end

S3: begin

shlb = 1; up₂ = 1;

if (outb) up₁ = 1;

if (cy) next_state = S3;

else next_state = SF;

end

SF: begin

FIN = 1;

next_state = S0;

end