

- **Contesteu les preguntes als fulls de respostes (podeu fer servir llapis).**
- **No es pot fer servir calculadora.**
- **Les notes es publicaran a Atenea en 7 dies i s'indicarà com demanar revisió.**

**1.- (1 punt)** Donada la taula de veritat d'un CLC amb 3 entrades ( $a$ ,  $b$  i  $c$ ) i dues sortides ( $r$  i  $s$ ):

- Sintetitzeu  $r$  i  $s$  descomposant-les en suma de *minterms*
- Sintetitzeu  $r$  i  $s$  mitjançant un decodificador
- Sintetitzeu  $r$  i  $s$  mitjançant una memòria ROM
- Optimitzeu mitjançant un mapa Karnaugh l'expressió lògica de  $r$  com a suma de productes i sintetitzeu-la.

En cada cas, indiqueu el nombre i tipus de components necessaris així com el contingut de la ROM.

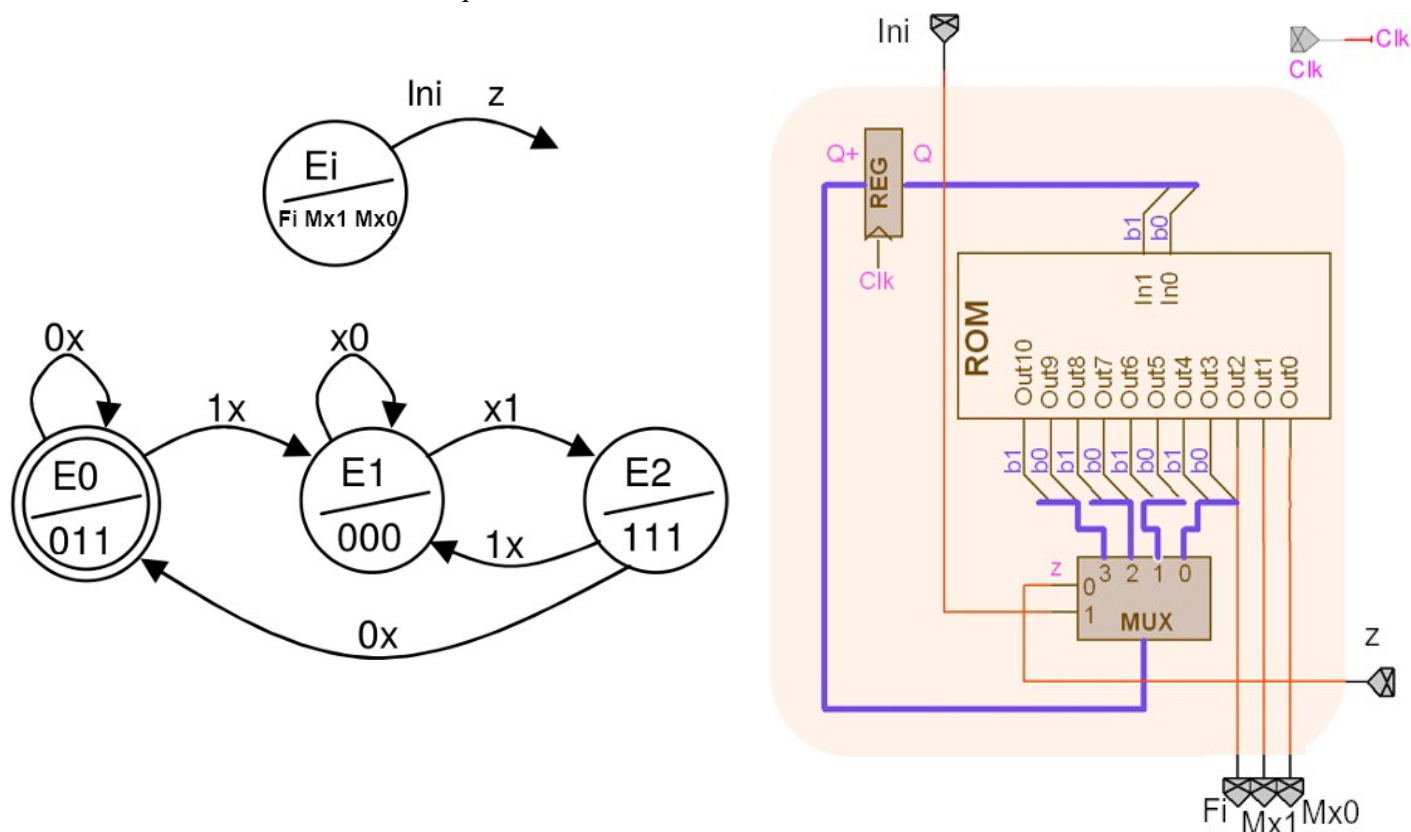
a	b	c	r	s
0	0	0	1	0
0	0	1	1	1
0	1	0	x	x
0	1	1	0	0
1	0	0	x	0
1	0	1	0	x
1	1	0	1	0
1	1	1	x	1

**2.- (1,5 punts)** Completeu els espais en blanc de cada fila.  $X$  i  $Y$  son vectors de 8 bits,  $X+Y$  és la suma en aritmètica modular en 8 bits, on  $c_n$  i  $v1_n$  són les sortides que indiquen la no representabilitat per a naturals/enters respectivament.  $X-Y$  és la resta en aritmètica modular en 8 bits, on  $b_n$  i  $v2_n$  són les sortides que indiquen no representabilitat per a naturals/enters respectivament.

X	$X_u$	$X_s$	Y	X+Y	$c_n$	$v1_n$	X-Y	$b_n$	$v2_n$
0001 0000			1000 1000						
	240			0111 1111					
		-45					1000 0000		

**3.- (0,5 punts)** Sintetitzeu un multiplexor de bits 4-1 a partir de multiplexors de bits 2-1.

**4.- (0,5 punts)** Volem sintetitzar el CLS corresponent al següent graf d'estats mitjançant una única ROM de mida mínima i un multiplexor de busos. Indiqueu el contingut en hexadecimal de les files 0 i 1 de la ROM, assumint que l'estat  $Ei$  està codificat amb el nombre natural  $i$ .



5.- (2,5 punts) La conjectura de Collatz sosté que, per a tot natural  $n$ , la seqüència de nombres  $\{a_i\}$  on  $a_i = \begin{cases} n & \text{per } i = 0 \\ f(a_{i-1}) & \text{per } i > 0. \end{cases}$  i  $f(n) = \begin{cases} n/2 & \text{si } n \equiv 0 \pmod{2} \\ 3n+1 & \text{si } n \equiv 1 \pmod{2} \end{cases}$  conté el valor 1

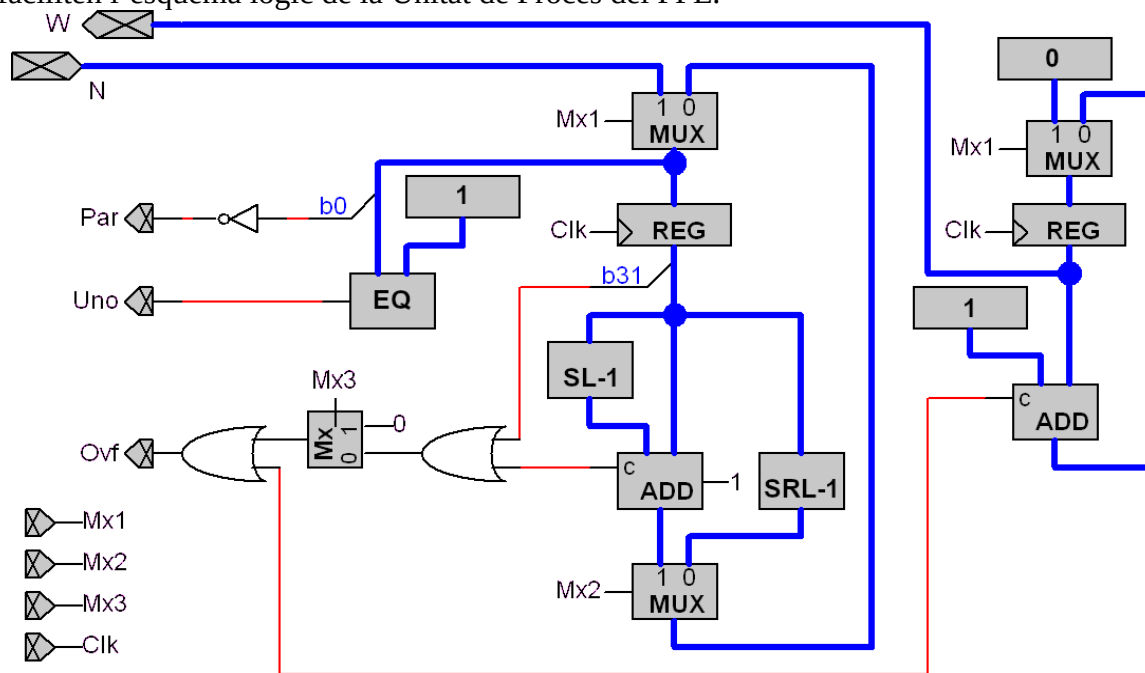
El primer terme de la seqüència és  $n$ , el següent és  $f(n)$  on si  $n$  és parell  $f$  retorna  $n/2$ , si no retorna  $3 \cdot n + 1$ ; en general, cada terme és el resultat d'aplicar la funció  $f$  al terme anterior. Per exemple, per a  $n=4$ , la seqüència és  $\{4, 2, 1, 4, 2, 1, \dots\}$  i per a  $n=6$  és  $\{6, 3, 10, 5, 16, 8, 4, 2, 1, \dots\}$ . Al primer cas calen 2 iteracions per arribar al valor 1 mentre que al segon cas en calen 8. Per a  $n=1$  en calen 0.

Construirem un PPE tal que donat un natural  $N \geq 1$  codificat en 32 bits ens indiqui quantes iteracions són necessàries per a arribar al valor 1. El PPE utilitzarà aritmètica modular de 32 bits. Les entrades del PPE seran un senyal de control d'un bit,  $Ini$ , i un bus de dades,  $N$ , de 32 bits. El mateix cicle que  $Ini$  valgui 1, al bus  $N$  tindrem el nombre natural que s'ha de processar.

Les sortides del PPE són dos senyals binaris d'un bit,  $End$  i  $Err$ , i un bus  $W$  de 32 bits. Mentre el PPE estigui esperant un nombre a processar o estigui processant un nombre,  $End$  valdrà 0. Quan  $End$  valgui 1, indicarà que el PPE dona per finalitzat el processament. En aquest punt hi ha dues opcions: si  $Err=1$  vol dir que s'ha produït un *overflow* que influeix al resultat i el contingut de  $W$  serà irrellevant; si  $Err=0$ ,  $W$  contindrà el nombre d'iteracions.

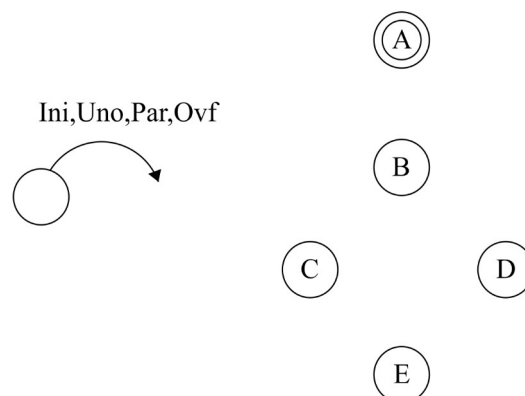
Un cop iniciat un processament, s'ignorarà el senyal  $Ini$  fins el cicle posterior a mostrar  $End=1$ .

Ens faciliten l'esquema lògic de la Unitat de Procés del PPE.



Es demana que construïu el graf d'estats i la taula de sortides de la Unitat de Control (ens diuen que el graf té 5 estats, ens facilitem la llegenda del graf i alguns valors de les sortides). Al graf d'estats no cal indicar les transicions impossibles. A la taula de sortides, cal posar x als senyals irrellevants.

		Sortides UC				
		End	Err	Mx1	Mx2	Mx3
Estats UC	A (inicial)					
	B	1				
	C					
	D				0	
	E		1			



**6.- (0,5 punts)** Volem sintetitzar la UC del PPE de l'exercici 5 mitjançant una ROM de mida mínima i un multiplexor de busos. Indiqueu quina serà la mida de la ROM en termes de nombre de paraules (files) i nombre de bits per paraula (columnes). Observació: no cal haver fet l'exercici 5 per a respondre aquesta pregunta.

**7.- (1 punt)** Determineu el camí(ns) crític(s) i el seu temps de propagació a la UP de l'exercici 5. Assumiu els següents temps de propagació (en unitats de temps):  $T_p(\text{Not}) = 10$ ,  $T_p(\text{OR-2}) = 20$ ,  $T_p(\text{Mx2-1}) = T_p(\text{MUX2-1}) = 50$ ,  $T_p(\text{SL-1}) = T_p(\text{SRL-1}) = T_p(1) = T_p(0) = 0$ ,  $T_p(\text{ADD}) = 200$ ,  $T_p(\text{EQ}) = 330$ ,  $T_p(\text{REG}) = 50$ . Ens diuen que **tots** els senyals d'entrada (incloent-hi els senyals de control generats per la UC) triguen 150 u.t en establir-se i que **tots** els senyals de sortida (incloent-hi els de la paraula de condició) han de mantenir-se estables 80 u.t.

Cal especificar el camí crític indicant els components que el componen correlativament. Per exemple,  $\text{REG} \rightarrow \text{SRL-1} \rightarrow \text{MUX} \rightarrow \text{MUX} \rightarrow \text{REG}$ ; al càlcul del temps de propagació cal indicar explícitament tots els valors que intervenen al càlcul. Per exemple,  $100 + 20 + 20 = 140$  u.t.

Observació: no cal haver fet l'exercici 5 per a respondre aquesta pregunta.

**8.- (0,5 punts)** Indicad el valor que debe tener cada uno de los bits de la palabra de control de la UPG básica para que realice, durante un ciclo, la acción concreta especificada mediante el mnemotécnico. Indicad con x las casillas cuyo valor no importe para la ejecución de la instrucción. En caso que no se pueda realizar la acción, tachad toda la línea de señales.

Mnemotécnico	@A	@B	Rb/N	OP	F	In/Alu	@D	WrD	N (hexa)
CMPEQ -, R2, R7									
SUBI R2, R3, -1									

**9. (0,5 puntos)** Indicad el mnemotécnico que corresponde a cada una de las siguientes palabras de control de la UPG básica.

Mnemotécnico	@A	@B	Rb/N	OP	F	In/Alu	@D	WrD	N (hexa)
	001	xxx	x	00	011	0	110	1	XXXX
	111	xxx	x	10	000	0	000	1	XXXX

**10.- (1,5 puntos)** Dado el siguiente fragmento de código en C donde todos los datos son naturales, completad el fragmento de grafo de estados de la UC de propósito específico para que junto con la UPG formen un procesador que realice la funcionalidad descrita en el fragmento de código. Indicad los arcos que faltan, las etiquetas de los arcos (z, !z, o nada) y completad cada palabra de control con mnemotécnicos a la derecha de cada nodo del grafo. Tenéis que utilizar R7 como registro temporal.

```
R0 = 0;
while (R1 > 1) {
    if (R1 % 2 == 0) // R1 par
        R1 = R1 / 2;
    else
        R1 = 3*R1 + 1;
    R0 = R0 + 1;
}
...
```

