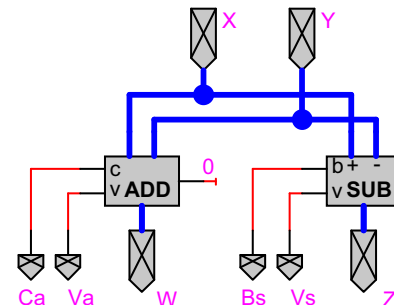


**Examen 2** (temas 4, 5, 6 y 7)

Duración: 1 hora 45 minutos. La solución de cada ejercicio se tiene que escribir en el espacio reservado para ello en el propio enunciado. No podéis utilizar calculadora, apuntes, etc. La solución del examen se publicará en Atenea mañana por la tarde y las notas antes de una semana.

**Ejercicio 1** (Objetivos 2.4, 5.6, 4.9 y 5.11) (2 puntos)

Dado el esquema del circuito combinacional a bloques, donde los buses son de 4 bits, completad las columnas que faltan, para cada una de las tres filas (casos concretos de valores de las entradas). Los vectores de bits X, Y, W y Z se indican en hexadecimal. Las interpretaciones de los vectores de bits de las salidas W y Z (con subíndices u y s según se trate de números naturales o enteros) se deben expresar en decimal sin y con signo respectivamente).



	X	Y	Ca	Va	W	Wu	Ws	Bs	Vs	Z	Zu	Zs
a	1	D										
b	F	A										
c	5	A										

**Ejercicio 2** (Objetivos: 6.11) (1,5 puntos)

Completad el grafo de estados que describe el funcionamiento del siguiente circuito lógico secuencial de Moore. Primero definimos qué es un pulso y luego la funcionalidad del circuito.

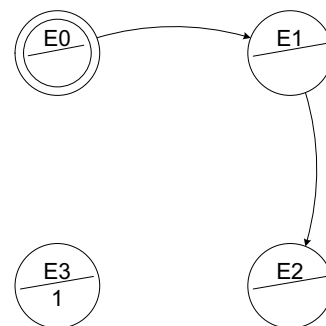
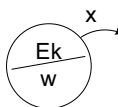
Un pulso, en una señal binaria síncrona (en una secuencia de bits), es cada secuencia de unos consecutivos. La anchura del pulso es el número de unos consecutivos que tiene el pulso. Por ello, una señal binaria síncrona consiste en una secuencia de pulsos (cada uno con una anchura determinada) separados entre sí por uno o más ceros.

El circuito tiene una entrada x y una salida w. La secuencia de bits de salida es igual a la de entrada después de eliminar los pulsos con anchura superior a la unidad (los pulsos de entrada con anchura p para  $p > 1$ , son sustituidos por p ceros consecutivos en la salida). La salida aparece retardada dos ciclos respecto a la entrada.

De cara a interpretar el enunciado considerad que en el estado inicial es como si en los ciclos anteriores (que no han existido) la entrada x hubiera valido siempre cero. El cronograma simplificado muestra un ejemplo del comportamiento del circuito.

Grafo:

Leyenda:



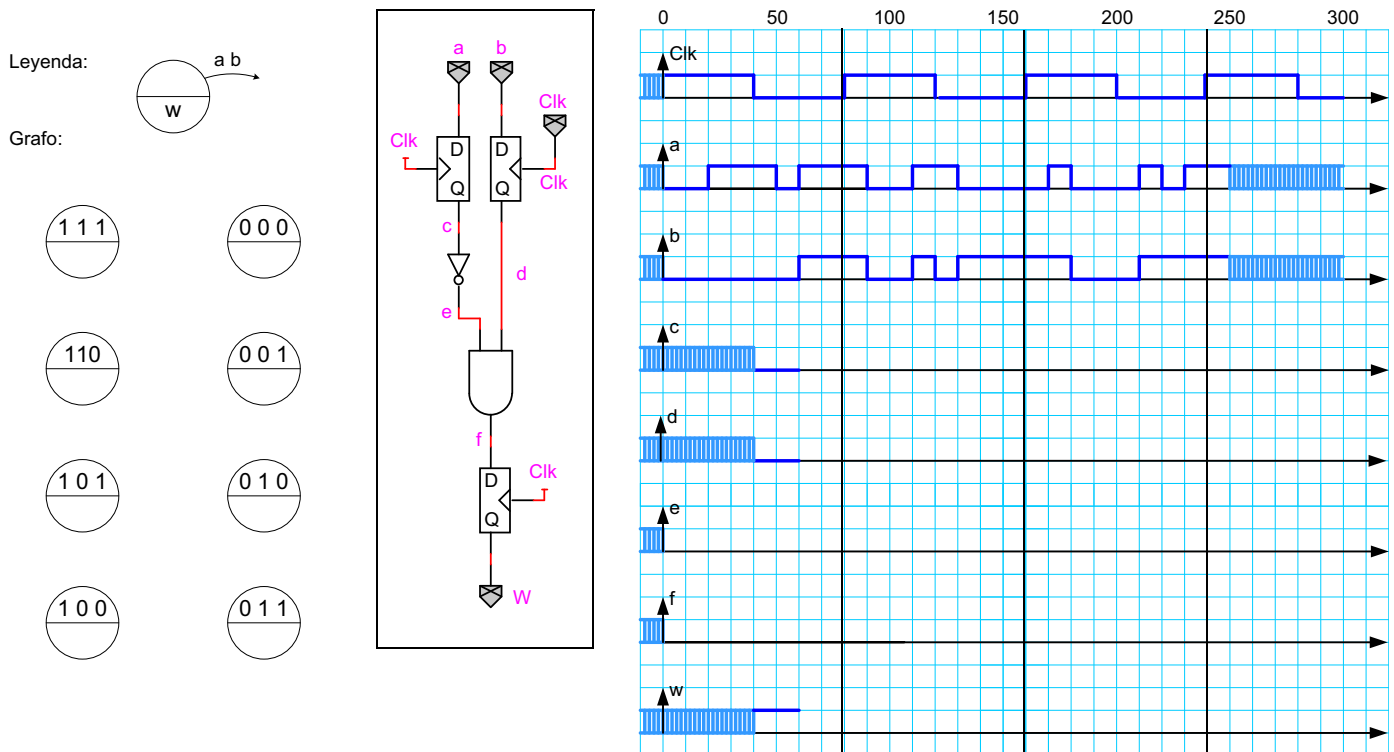
Ciclo:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
x	1	0	0	1	0	1	1	0	0	1	1	1	0	1	0	1	0	0	1	1	0	0
w	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0

Espacio en blanco de libre uso

**Ejercicio 3** (Objetivo 6.11, 6.16, 6.13 y 6.10) (3 puntos)

Para el circuito secuencial de la figura:

- a) Dibujad todos los arcos que salen de los nodos 000 y 111 indicando claramente las etiquetas de cada arco, el nodo origen y el destino. Para ello podéis ordenar las variables lógicas del estado actual como mejor os parezca, pero no olvidéis de incluir sus nombres ordenados en la leyenda del grafo, que se da incompleta. Escribid el valor de la salida  $w$  para todos los nodos. (1 punto)
- b) Completad el cronograma de las señales del esquema lógico sabiendo que los tiempos de propagación de las puertas son  $T_p(\text{Not}) = 10 \text{ ut}$  y  $T_p(\text{And-2}) = 20 \text{ ut}$  y el de los biestables es  $T_p(\text{FF}) = 40 \text{ ut}$ . (1 punto)



- c) Si se implementa un circuito con la misma funcionalidad que el de la figura con solamente tres biestables, una ROM y un multiplexor de buses se necesita una ROM de  $p$  palabras de  $n$  bits por palabra y un MUX de  $k$  buses de entrada y 1 de salida siendo los buses de  $b$  bits. ¿Cuánto valen  $p$ ,  $n$ ,  $k$  y  $b$ ? (0,5 puntos)

$p =$	$n =$	$k =$	$b =$
-------	-------	-------	-------

- d) Completad el cronograma simplificado que muestre la misma información (considerando una señal síncrona como una secuencia de bits) que el cronograma detallado anterior. Escribid en cada casilla/ciclo los tres bits del estado siguiendo el orden que hayáis usado en la leyenda del grafo y mantened este orden en el estado siguiente (0,5 puntos)

Ciclo:	0	1	2
Entradas: $ab$			
Estado: ???			
Estado <sup>+</sup> : ???			
Salida: $w$			

**Ejercicio 4** (Objetivos 7.6, 4.6.2 y 7.8) (3,5 puntos)

Completad el diseño del PPE cuya UP completa y cuyo grafo de la UC incompleto se muestran a continuación. El PPE forma parte de un circuito más complejo de una máquina de cobro automático. El precio a cobrar siempre es un número natural de euros (una cantidad de euros exacta, sin céntimos) en el rango de 1 a 2.000 euros. La máquina devuelve el cambio, cuando haya que devolverlo, entregando el número mínimo de billetes de 10 euros y de monedas de un euro (no devuelve ningún otro tipo de billetes ni de monedas). El PPE, para cada cobro que requiere dar cambio, recibe la cantidad de euros a devolver (del subsistema que alimenta al PPE) y se encarga calcular y de dar la orden al subsistema dispensador de cambio (alimentado por el PPE) de cada billete y de cada moneda a devolver.

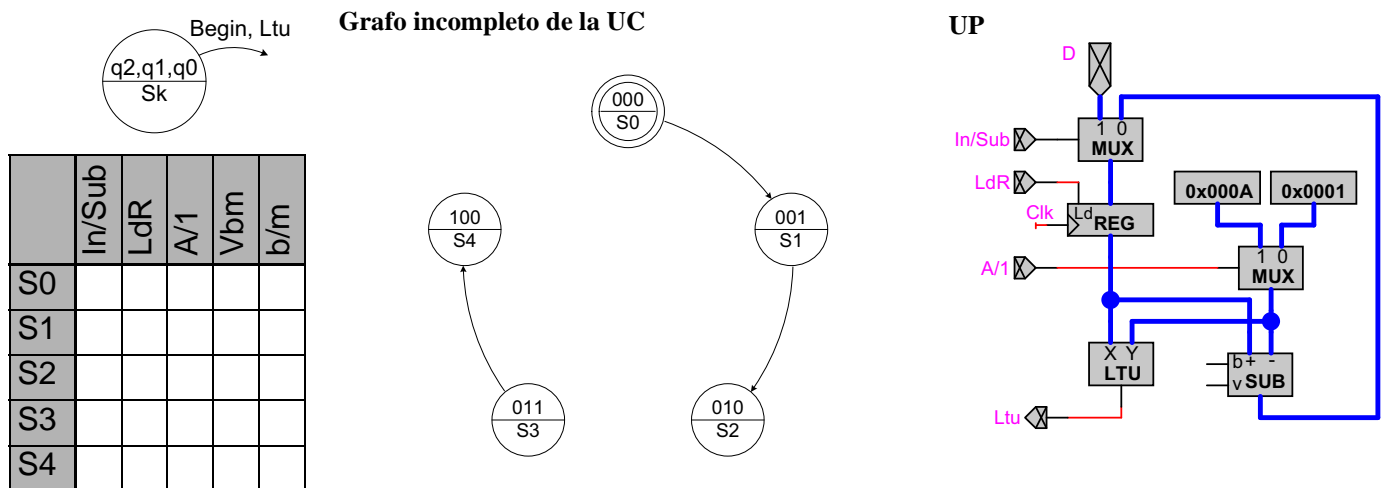
El PPE tiene un bus de entrada de datos,  $D$ , de 16 bits y una señal de entrada,  $\text{Begin}$  (de un bit). Recibe el número de euros a devolver codificado en binario por el bus  $D$  durante un ciclo, el mismo en el que la señal  $\text{Begin}$  vale 1. Esto es, el ciclo en el que  $\text{Begin}$  vale 1 le indica al PPE que debe iniciar el cálculo del número de billetes de diez euros y de monedas de un euro a devolver, que suman la cantidad de euros que indica el bus  $D$  en ese mismo ciclo.

Apellidos y Nombre: .....Grupo:.....DNI: .....

El PPE tiene dos señales de salida de un bit cada una, Vbm y b/m, utilizadas para enviar las órdenes de dar cambio al subsistema dispensador de cambio. El PPE pone a 1 la señal Vbm durante un ciclo para validar el valor que tiene la señal b/m durante ese mismo ciclo. Cuando b/m vale 1 significa que se debe devolver un billete y cuando vale 0 que debe devolverse una moneda. En los ciclos en los que Vbm vale 0 el valor de b/m carece de significado y en esos ciclos no se da orden de dar ni billete ni moneda. Si, por ejemplo, se deben devolver 23 euros el PPE enviará al subsistema dispensador una secuencia de cinco órdenes de dar cambio (cinco ciclos, que no tienen porque ser consecutivos, en los que Vbm valdrá 1). Los 23 euros del ejemplo se devolverán mediante una secuencia de dos órdenes de dar un billete y tres de dar una moneda.

Tal como está construida la máquina de cobro, se puede asegurar que el subsistema que alimenta al PPE nunca pondrá Begin a 1 hasta que el PPE haya terminado de dar todas las órdenes de devolver el cambio de un cobro: Begin valdrá 1 durante un ciclo en el que el PPE este en el estado inicial, esperando trabajo.

Nota: La salida del bloque LTU vale 1 cuando  $X_u < Y_u$ . El bus de salida de 16 bits del bloque 0x000A y del 0x0001 siempre codifica el valor constante que indica su nombre.



- Completad el grafo de estados de la UC, al que le faltan arcos y las etiquetas de los arcos así como el valor de las salidas en cada nodo, que se debe indicar en la tabla. (2 puntos)
- Indicad el camino crítico del PPE (o uno de ellos si hay varios) listando los dispositivos por los que pasa y el tiempo de este camino escribiendo los sumandos que lo forman. ¿Cuál es el tiempo de ciclo mínimo? Suponed que:
  - la UC se implementa con el número mínimo de biestables, con una única ROM (ROM\_UC) y con un Multiplexor de buses,
  - todas las entradas del PPE están estables pasadas 300 ut del inicio de ciclo y que todas las salidas deben estar estables al menos 600 ut antes del final de ciclo,
  - los tiempos de propagación de cualquiera de los biestables con los que está construido el PPE (incluidos los biestables de los registros) es de  $T_p(FF)=100$  ut y que
  - los tiempos de propagación de los combinacionales, desde cualquier entrada a cualquier salida, son:  $T_p(SUB)=T_p(LTU)=500$ ,  $T_p(MUX-2-1)=50$ ,  $T_p(MUX-4-1)=100$ ,  $T_p(MUX-8-1)=150$  y  $T_p(ROM\_UC)=60$  ut.

(Recordad que un registro con señal Ld tiene algo más que biestables en su implementación interna) (1 punto)

Camino crítico:

Sumandos:

= ; Tiempo\_ciclo\_mínimo =

- Dibujad el circuito que puede reemplazar al bloque original LTU para reducir el coste en hardware de la UP (podéis usar bloques, puertas...) (0,5 puntos).

