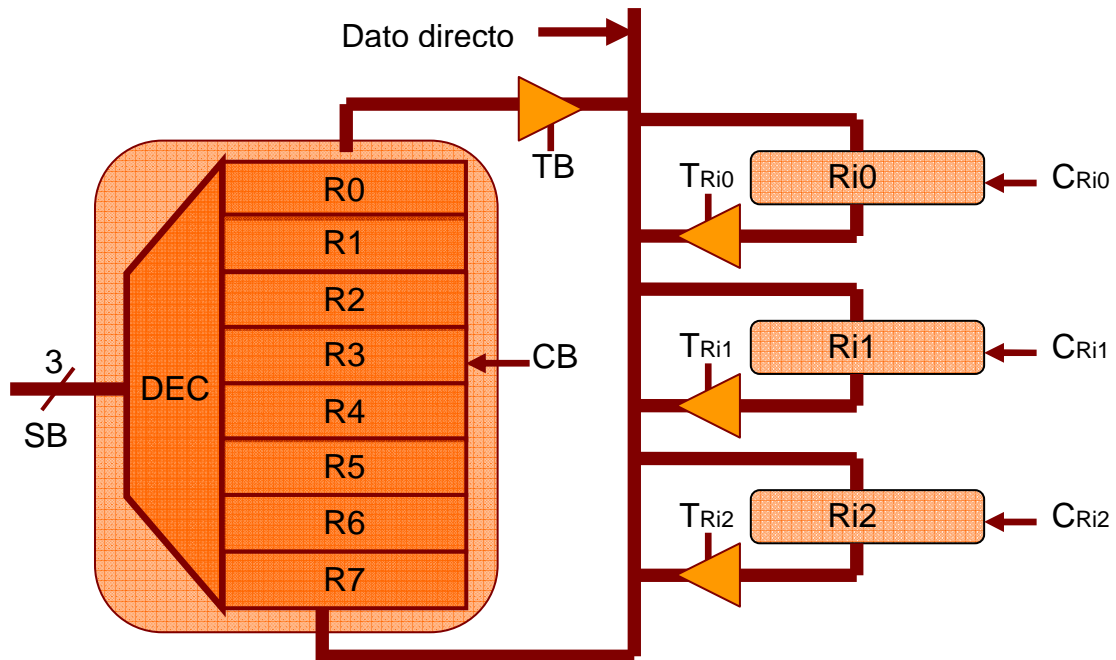


1. Realizar el cronograma correspondiente a las operaciones de transferencia de registros (8 bits) a realizar en el siguiente camino de datos.

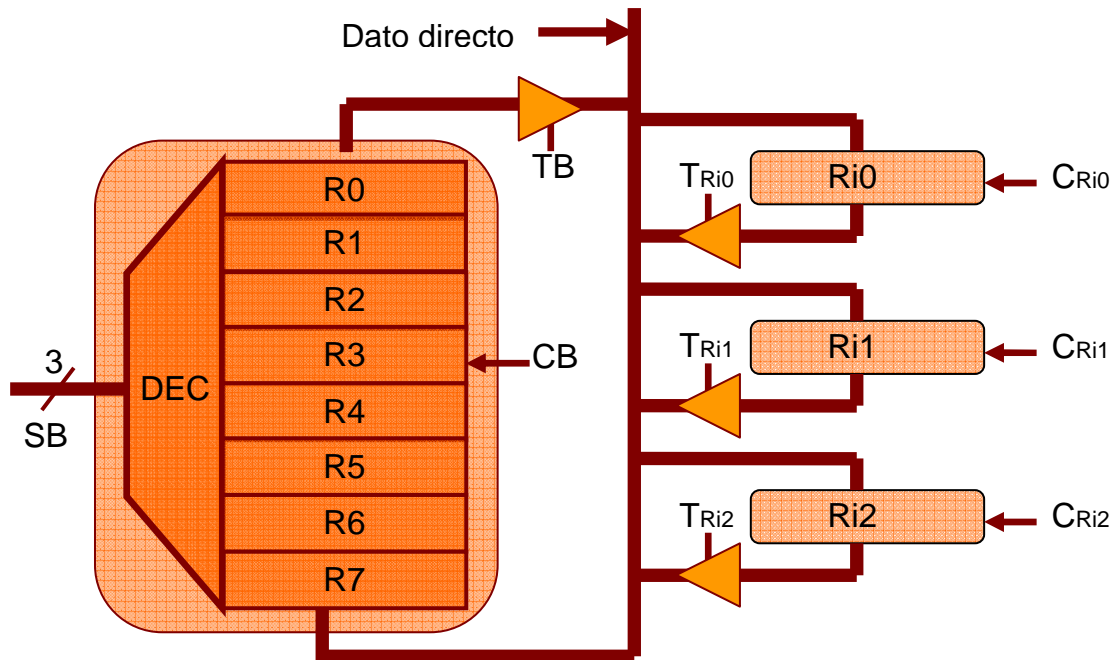


- a) $R0 \leftarrow 30h$ | $Ri2 \leftarrow 30h$ | $Ri1 \leftarrow R0$
b) $R4 \leftarrow 4Bh$ | $Ri1 \leftarrow R4$ | $Ri0 \leftarrow Ri1$

CLK									
Datos D.									
SB									
CB									
TB									
CRi0									
TRi0									
CRi1									
TRi1									
CRi2									
TRi2									

CLK									
Datos D.									
SB									
CB									
TB									
CRi0									
TRi0									
CRi1									
TRi1									
CRi2									
TRi2									

2. Realizar el cronograma correspondiente a las operaciones de transferencia de registros (8 bits) a realizar en el siguiente camino de datos.



c) $Ri1 \leftarrow FAh$ | $Ri2 \leftarrow Ri1$ | $R7 \leftarrow Ri2$

d) $R1 \leftarrow 4Bh$ | $R2 \leftarrow 5Bh$ | $Ri1 \leftarrow R1$

CLK									
Datos D.									
SB									
CB									
TB									
CRi0									
TRi0									
CRi1									
TRi1									
CRi2									
TRi2									

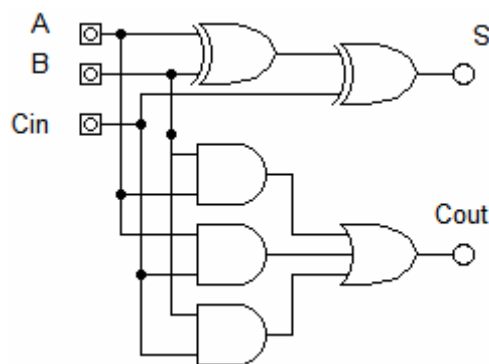
CLK									
Datos D.									
SB									
CB									
TB									
CRi0									
TRi0									
CRi1									
TRi1									
CRi2									
TRi2									

3. Diseñar una ALU que trabaja con operandos de 1 bit capaz de realizar las operaciones lógicas: AND, OR, XOR, NOT. Construye la tabla que indique la funcionalidad de la ALU.

4. Diseñar un operador lógico de inversión para operandos de 8 bits. Indicar el resultado del operador cuando se introduce la información 1111 0000.

5. Diseñar una ALU para dos operandos A y B de 4 bits cada uno que sea capaz de realizar las operaciones: AND, OR, NOR y NAND. Detallar cada operador, la ALU puede dibujarla usando diagramas de bloques de los operadores ya diseñados y detallados.

7. En el siguiente circuito, si el retardo de cada puerta del sumador completo es de 8ns.



a) Averigüe el tiempo que debe dejar transcurrir para leer en sus dos salidas un resultado correcto desde que se aplican a sus entradas los valores lógicos correspondientes.

b) Calcular tiempo para los siguientes retardos XOR= 12ns, AND= 7ns y NOR = 9ns

8. Dibujar el esquema de una ALU con AND, OR, XOR y suma binaria completa de dos bits de entrada A y B y acarreo previo C. Detallar el circuito del sumador binario serie.

9. Se dispone de un sumador paralelo de 4 bits compuesto por cuatro sumadores completos de 25ns de retardo cada uno. Si se introducen al sumador como operandos los valores 1010 y 0111:

a) Tiempo total para obtener la suma

b) Resultado de la suma

c) Si solo se deja transcurrir 50ns, ¿qué bits serán correctos y cuáles erróneos?

10. Realizar las operaciones de suma y resta siguientes, indique junto a cada operando su valor equivalente en decimal, sabiendo que el formato de los operandos es binario puro (sin signo).

a) 00011110 + 10000000

b) 0011 + 1101

c) 101011 + 000111

d) 1110 - 0001

e) Determina el rango de representación de cada uno de los operandos trabajado antes (n=8, n=4, n=6).

h) En función de los rangos obtenidos indica en qué casos se ha producido desbordamiento al obtenerse como resultado un valor fuera del rango válido de representación.

11. Considerar datos de 8 bits almacenado en complemento a 2
- a) Indicar las representaciones de los siguientes valores: (+15), (15), (+98), (-98), (+113), (-113)
 - b) Realizar las siguientes operaciones: (+15)+(+98), (+15)-(-98), (15)-(-113), (-15)+(-98), (-113)+(+15), (+113)-(+98).

14. Considerar datos de 8 bits representados en complemento a dos. Indicar el valor que tomarán los biestables de acarreo y desbordamiento al realizar las siguientes sumas.

- a) (+7)+(+2)
- b) -4)+(+3)
- c) (-7)+(-13)
- d) (+96)+(+80)
- e) (-71)+(-72)

15. Dados A=1001 y B=0110 en formato complemento a 1, realizar la operación A-B utilizando el sumador de complemento a 1. Dibuje el esquema con los valores dados y los resultados obtenidos.

16. Dados A=101 y B=011 en formato complemento a 2, realizar la operación A-B utilizando el sumador de complemento a 2. Dibuje el esquema con los valores dados y los resultados obtenidos.

17. Diseñar una ALU de 16 bits que realice las siguientes operaciones:

- a) Suma/resta en complemento a 2;
- b) Funciones lógicas AND y OR.

El operando A se almacena en el registro acumulador, mientras que el operando B procede de la memoria a través del bus de datos. La ALU dispone de dos biestables de estado: acarreo y cero.

18. Diseñar el esquema lógico de un operador de desplazamiento que permita desplazar los bits contenidos en un registro A de 4 bits a la derecha o a la izquierda, según las señales de control apropiadas y dejando el resultado en el registro destino B.

19. Indicar el contenido del registro destino al aplicar el operador de desplazamiento de dos posiciones a la derecha sobre el contenido de un registro de 8 bits cuyo contenido es 69h.

20. Dibujar el contenido de un registro destino al aplicarle un operador lógico de desplazamiento a la izquierda al contenido de un registro origen de 8 bits cuyo contenido es 8Dh.

21. Dado el contenido de un registro origen, averiguar el del registro destino cuando sobre el primero se aplica un operador de desplazamiento aritmético que multiplica por 4 el operando inicial.

22. Aplicar a un registro origen A con valor CBh un operador de desplazamiento circular de dos posiciones a la derecha.

25. El registro A contiene el valor 0001 0010 que representa una cantidad en binario con signo. Para dividir por 2, se realiza un desplazamiento aritmético a la derecha ¿cuál es el valor final que contiene A?

26. Diseñar un operador combinacional de cuatro bits para cada una de las operaciones siguientes:

- a) Desplazamiento lógico a derecha e izquierda de 1 bit
- b) Desplazamiento lógico a derecha e izquierda de 2 bits

27. Diseñar un operador combinacional de cuatro bits (formato complemento a 1) para el desplazamiento aritmético a derecha e izquierda de 1 bit (dividir o multiplicar por dos, respectivamente)

28. Diseñar un operador combinacional de cuatro bits para un desplazamiento circular a derecha e izquierda de 1 bit.

29. Diseñar un operador combinacional de cuatro bits para cada una de las operaciones siguientes:

- a) Desplazamiento lógico a derecha e izquierda de 1 bit concatenado con el bit de acarreo
- b) Desplazamiento lógico a derecha e izquierda de 2 bits concatenado con el bit de acarreo

30. Diseñar una ALU de 8 bits que realice las siguientes operaciones:

- a) Suma/resta en ca1;
- b) Desplazamiento lógico a izquierda y derecha de 1 bit.

La ALU dispone de cuatro biestables de estado: C, S, Z y V.

31. Se desea dotar a la ALU de una máquina de un operador que compare dos operandos de 8 bits y tenga una línea de salida que tome el valor lógico 1 cuando ambos operandos sean iguales. Diseñar dicho operador.

32. Dibuje el esquema de un sumador-restador de 16 bits en complemento a 2. Complete el esquema añadiendo las puertas necesarias para activar los biestables de Cero (Z), Negativo(N), Desbordamiento (V) y Acarreo (C).

33. Diseñar una ALU de 8 bits que realice las siguientes funciones:

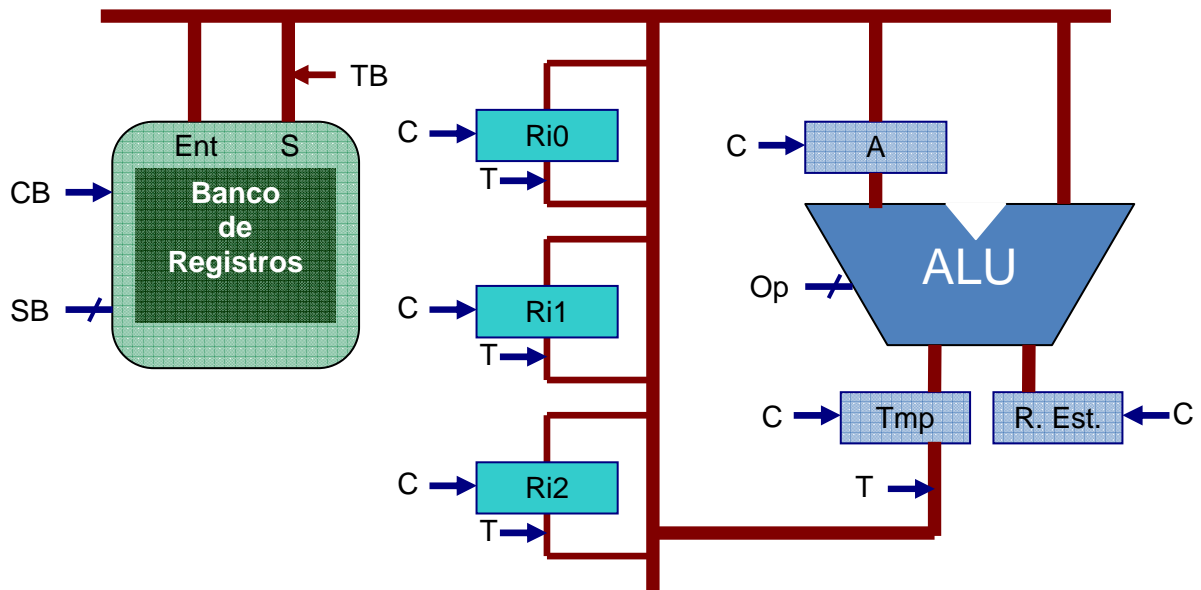
S1	S0	F
0	0	Rotar B a la izquierda
0	1	A más B
1	0	Rotar B a la derecha
1	1	A menos B

34. Diseñar el esquema de la ALU de una máquina de 16 bits. Los operandos proceden de la memoria principal (bus de datos), por lo que se necesitan dos registros, A y B. La ALU dispone de cuatro operaciones:

- a. Suma/resta en complemento a 2
- b. Función AND
- c. Función XOR
- d. Paso transparente del operando A

Así como de dos biestables: Acarreo (C) y Cero (Z)

Partiendo del siguiente camino de datos:




Con las siguientes características:

- Bus de datos de 8 bits
- Selectores de Banco de Registros 2 bits
- Formato del registro de estado de 4 bits: P-C-V-Z
- Las operaciones se indican directamente.
- Los operadores monádicos se operan a través del registro A
- El registro de estado se actualiza en toda operación que pueda producir cambios en los bits que lo componen.

Realizar los cronogramas correspondientes a las siguientes operaciones:


- $Ri2 = R2 + R3$
- $R3 = Ri2 + Ri0$
- $Ri0 = R4 \text{ OR } R1$
- $Ri0 = \text{NOT}(Ri0)$
- $R1 = \text{desplazamiento lógico a la derecha con acarreo del dato F2h contenido en R2.}$
- $Ri3 = \text{desplazamiento aritmético a la izquierda de dato 4Dh, contenido en R0}$

La plantilla aparece en la página siguiente.

Reloj									
Bus									
CB									
TB									
SB									
CRi0									
Tri0									
CRi1									
Tri1									
CRi2									
Tri2									
CA									
CTmp									
TTmp									
C R.Est									
OP									

[illegible]

Reloj									
Bus									
CB									
TB									
SB									
CRi0									
Tri0									
CRi1									
Tri1									
CRi2									
Tri2									
CA									
CTmp									
TTmp									
C R.Est									
OP									

Reloj									
Bus									
CB									
TB									
SB									
CRi0									
Tri0									
CRi1									
Tri1									
CRi2									
Tri2									
CA									
CTmp									
TTmp									
C R.Est									
OP									