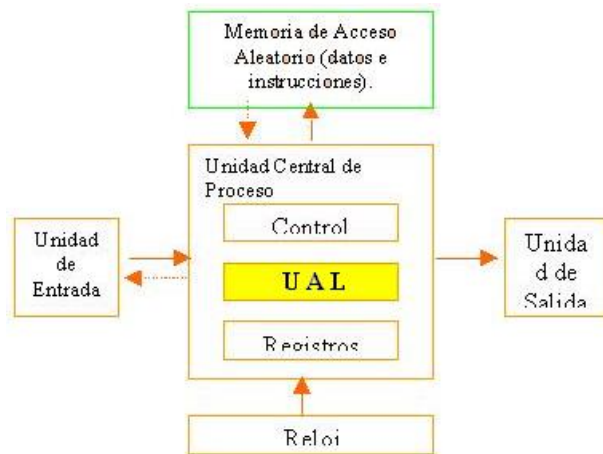


✠ ALU (Unidad Aritmético Lógica).



La Unidad Aritmética Lógica es normalmente un circuito combinacional encargado de realizar el conjunto de operaciones, como su nombre indica Aritméticas como sumas, resta, incrementos, etc. Y lógicas como and or not, etc. Y de la capacidad de procesamiento de todo el microcontrolador, depende en gran medida de la ALU integrada, de hecho, no se puede entender un microprocesador sin una ALU.

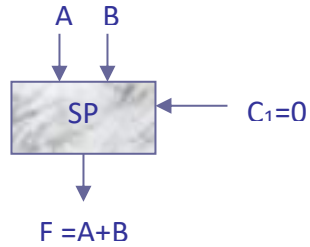
En lugar de tener registros individuales ejecutando las microoperaciones en forma directa, los sistemas computacionales emplean cierta cantidad de registros de almacenamiento conectados a todas las unidades operacionales comunes, la cual se denomina unidad aritmética-lógica, y se abrevia ALU (*arithmetic logic unit*).

Para ejecutar una microoperación, el contenido de los registros especificados se coloca en las entradas de la ALU común. La ALU ejecuta una operación y el resultado se transfiere al registro destino. La ALU es un circuito combinatorio, por lo que toda la operación de transferencia de registros desde los registros fuente por la ALU y hasta dentro del registro destino, puede ejecutarse durante un período de un pulso de reloj. Las microoperaciones de corrimiento con frecuencia se ejecutan en una unidad separada, pero en ocasiones la unidad de corrimiento forma parte de la ALU general. En este capítulo se plantea la función y el diseño de una ALU genérica.

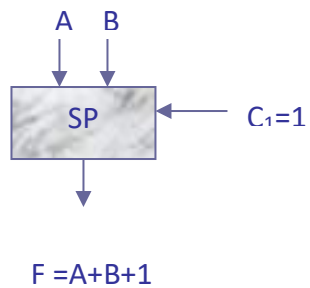
A continuación se presenta el conjunto de microoperaciones que una Unidad Aritmético Lógica puede efectuar.

☒ Conjunto de microoperaciones de una ALU.

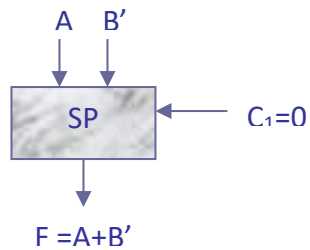
a) Sumador Paralelo



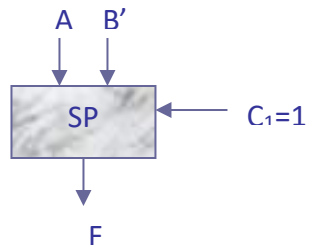
b) Suma con acarreo



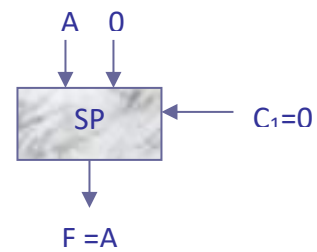
c) A más el complemento a uno de B



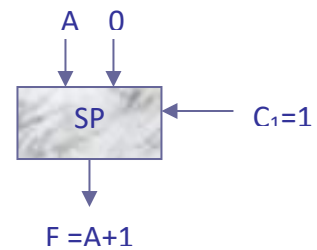
d) Resta



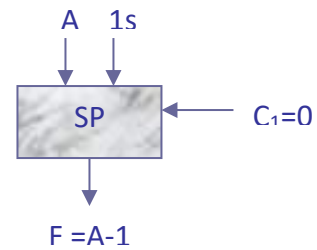
e) Transferir A



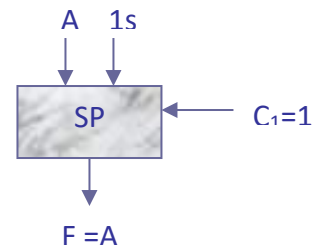
f) Incrementar A



g) Decrementar A



h) Transferir A

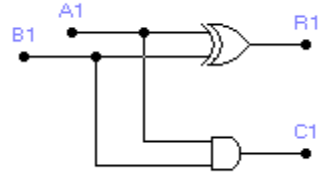




Unidad Aritmética

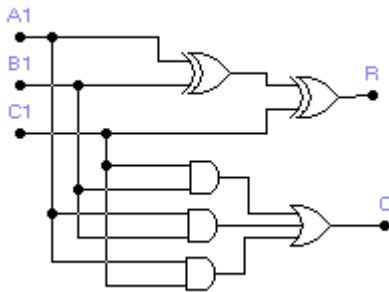
Para iniciar con el diseño de una UA partimos del diseño de un circuito mas basico

Diseño de un medio sumador (para circuito combinacional).



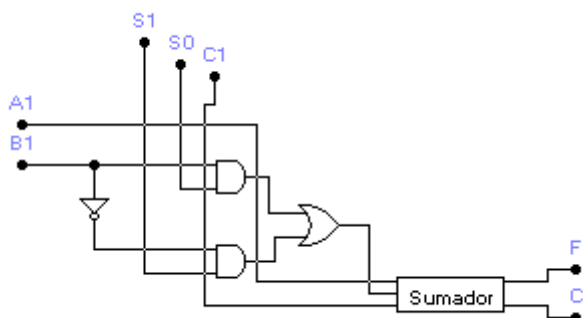
A1	B1	R1	C1
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Diseño de un sumador completo (circuito combinacional)



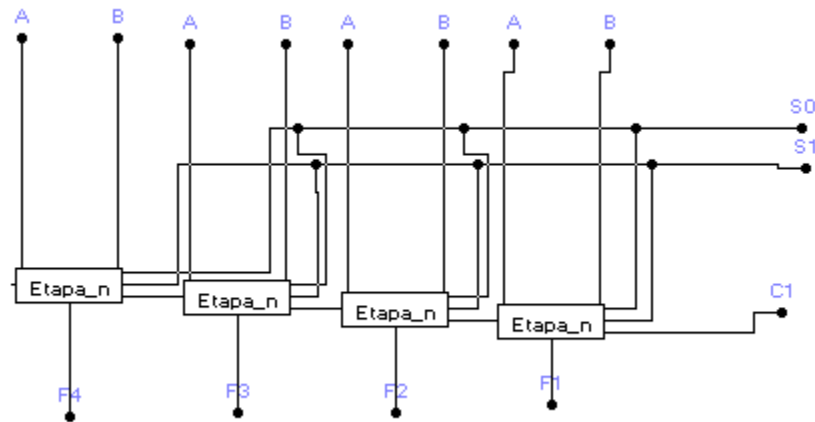
C1	A1	B1	R1	C1
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Ahora con un pequeño arreglo de compuertas obtenemos la unidad aritmética de un 1 bit



S1	S0	C1	SALIDA	DESCRIPCIÓN
0	0	0	$F = A$	Transfiere A
0	0	1	$F = A + 1$	Incrementa A
0	1	0	$F = A + B$	Suma A+B
0	1	1	$F = A + B + 1$	Suma acarreo
1	0	0	$F = A + B'$	A más B'
1	0	1	$F = A + B' + 1$	Resta
1	1	0	$F = A - 1$	Decrementa A
1	1	1	$F = A - 1 + 1 = A$	Transfiere A

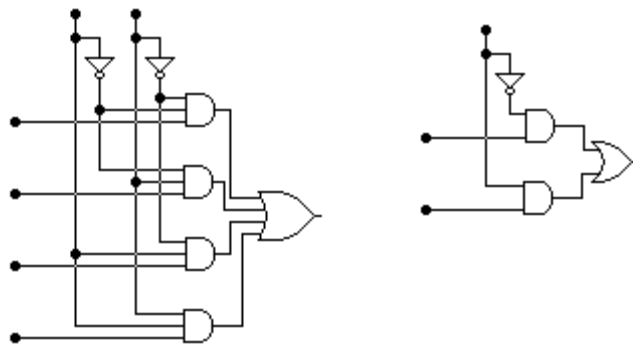
UNA ETAPA DEL CIRCUITO ARITMÉTICO.



Circuito aritmético compuesto de 4 etapas.

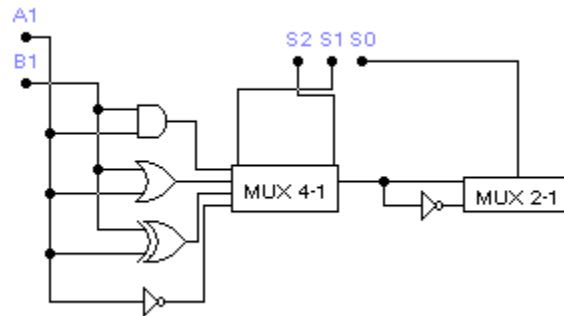
✠ **Unidad lógica:** Es un circuito combinacional cuyo propósito es realizar las operaciones lógicas como son AND, OR, XOR, NEGACIÓN, NAND, NOR, XNOR de una unidad aritmética.

Primero se utilizó un multiplexor de 4 a 1 para hacer una de las partes de la unidad lógica de 1 bit:



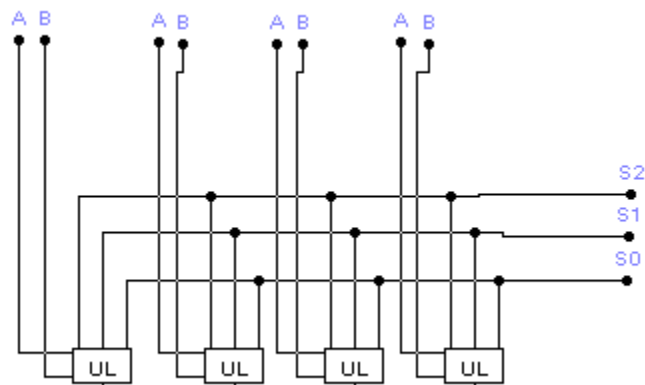
Multiplexores 4x1 y 2x1

Ahora con la ayuda del multiplexor 4 a 1 obtenemos la unidad lógica de 1 bit que es la sig.



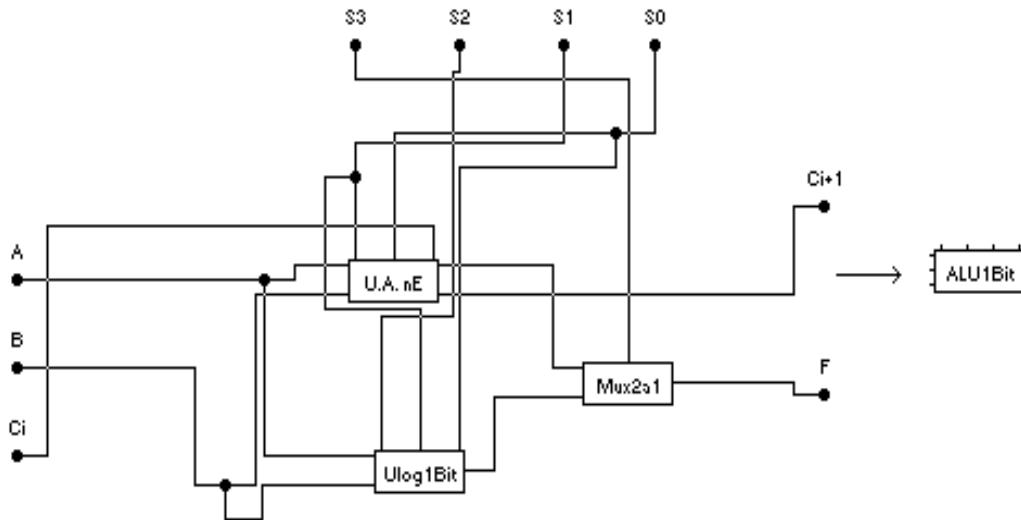
UNIDAD LÓGICA

Unidad Lógica de 4 bits

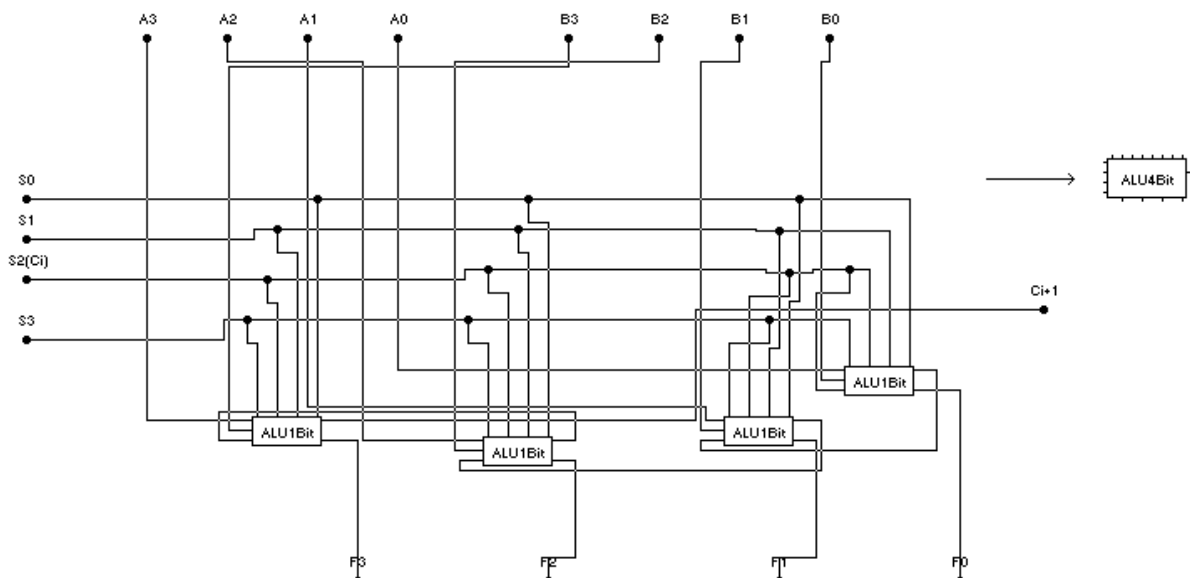


S2	S1	S0	FUNCIÓN
0	0	0	A and B
0	0	1	A or B
0	1	0	A xor B
0	1	1	A negada
1	0	0	A nand B
1	0	1	A nor B
1	1	0	A xnor B
1	1	1	Transfire A

Con un multiplexor se junta la unidad aritmética y la unidad lógica
ALU de 1 Bit:

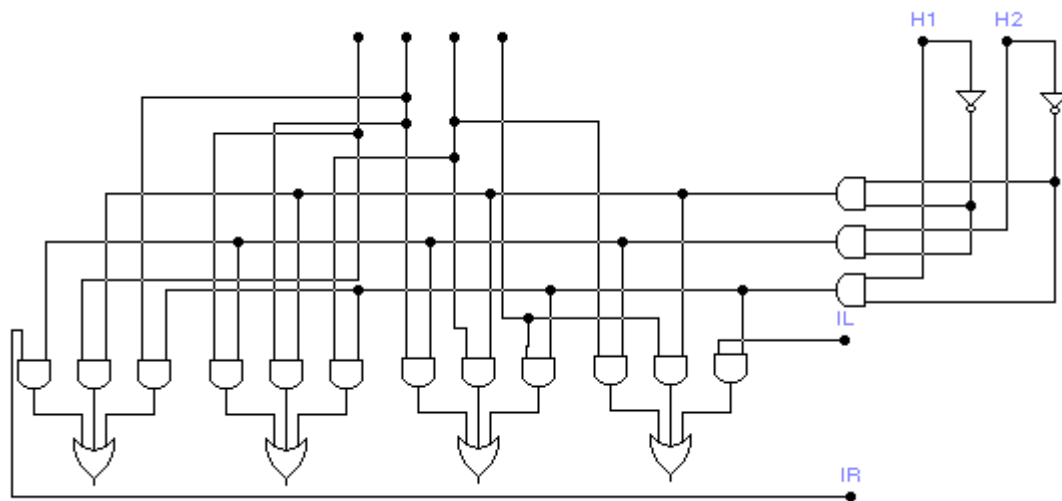


Agregando 3 unidades mas aritméticas se obtiene la ALU de 4 bits.
ALU de 4 Bits:

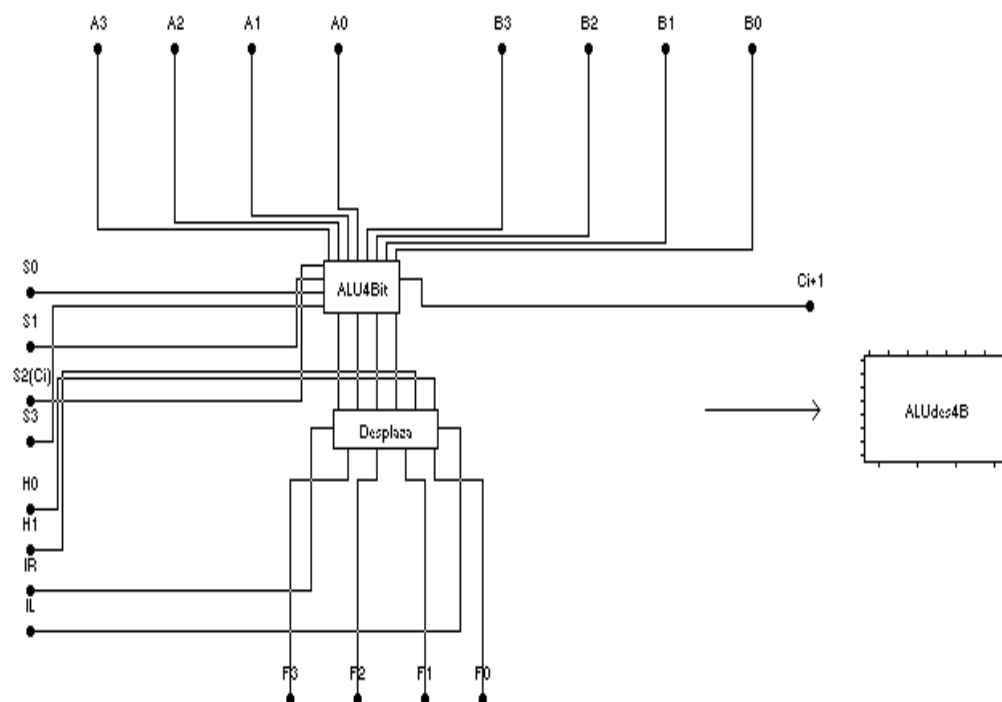


Para que quede mas completa la unidad aritmética lógica se le agrega un desplazador que hace los corrimientos y las rotaciones de los datos.

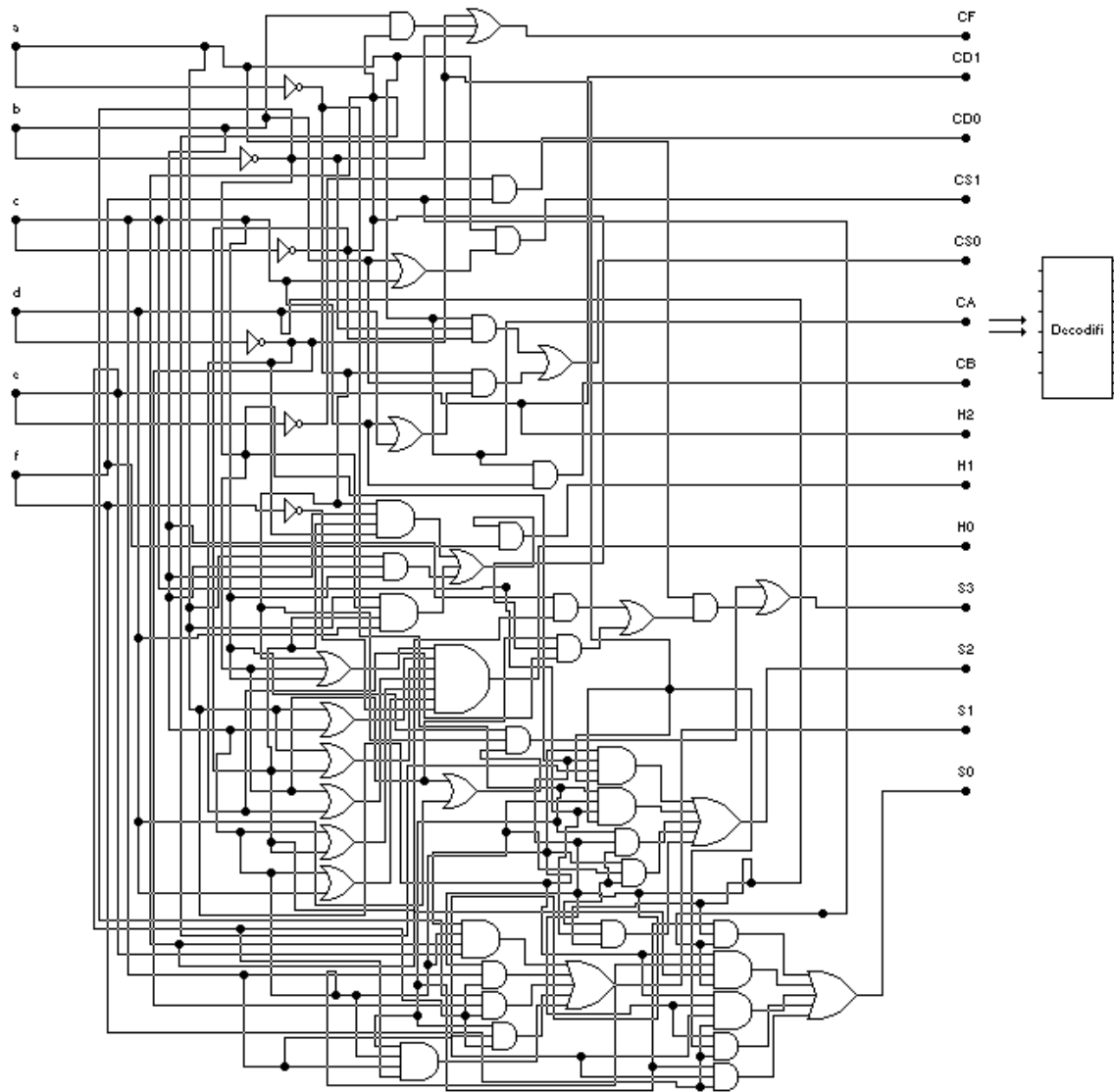
Desplazador:



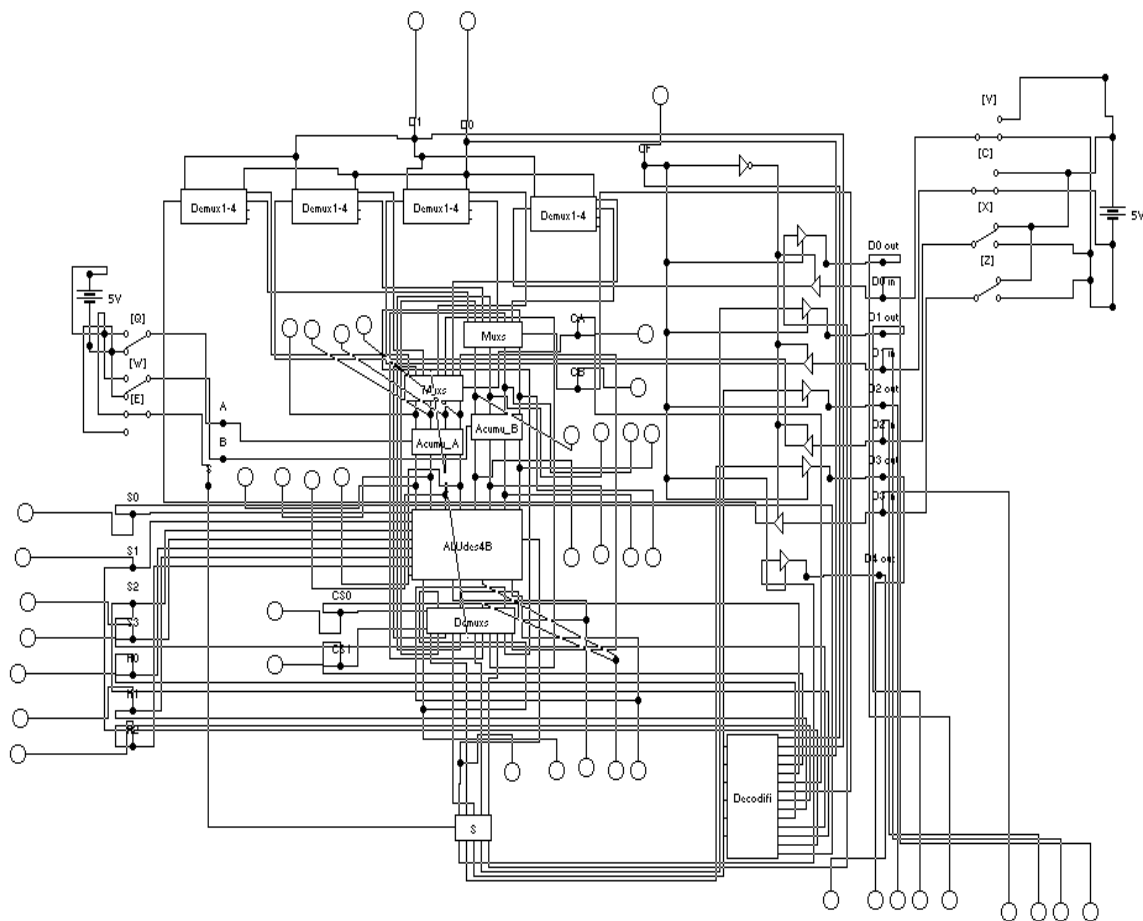
Entonces obtenemos ALU de 4 Bits con desplazador:



Y por último se le agrega a la ALU de 4 bits un decodificador
Decodificador:



Todo incorporado:



Flip-Flop utilizado para almacenamiento de datos:

