

12. Unidad aritmético lógica (UAL)

Es un circuito digital que realiza operaciones lógicas y aritméticas entre dos operandos de n bits.

12.1. Estructura de una computadora digital

En una computadora digital, la unidad aritmético lógica se localiza en la CPU (Unidad Central de Proceso) y realiza operaciones con los datos que provienen de la memoria o de dispositivos externos tal como se representa en la Figura 138.

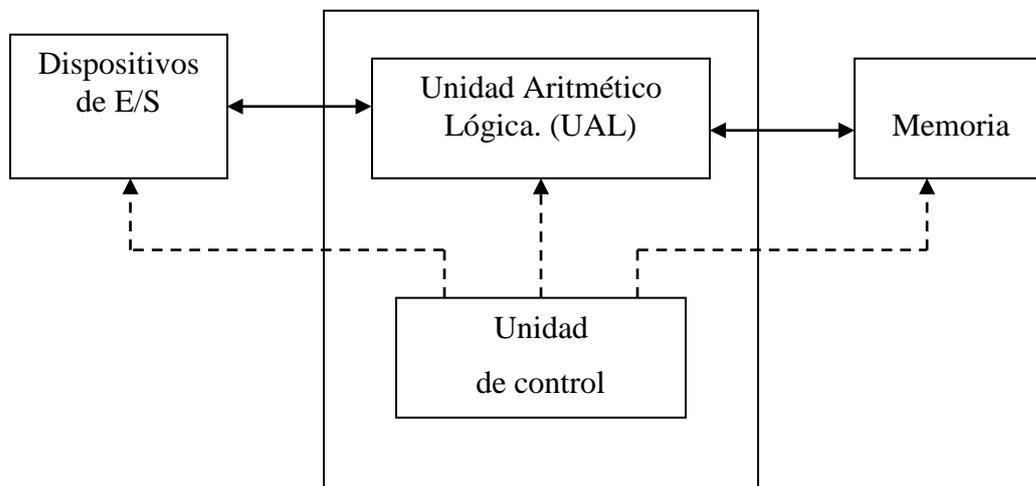


Figura 138

12.2. Diseño de una Unidad Aritmético Lógica: ALU

Como se ha definido anteriormente, una ALU realiza operaciones aritméticas y lógicas y por lo tanto requerirá de entradas que permitan la selección de una u otra operación. Así pues, como entradas, a parte de los operandos, dispondrá de k líneas de selección que permitirán seleccionar hasta $2k$ operaciones diferentes. Normalmente, se destina una de las k líneas a la distinción entre las operaciones lógicas de las aritméticas. El símbolo esquemático para una ALU es el representado en la Figura 139.

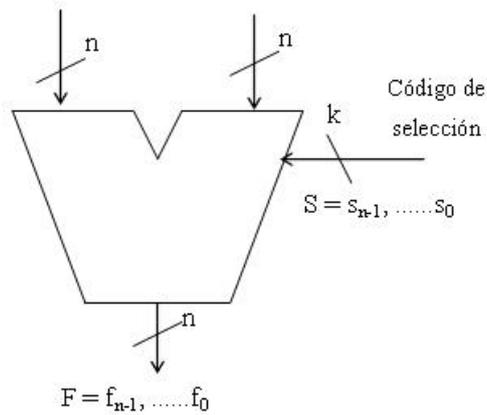


Figura 139

El procedimiento de diseño conlleva los siguientes pasos: asignación de una combinación de las k líneas a las operaciones a realizar, diseño de la unidad lógica (o aritmética), diseño de la unidad aritmética (o lógica) y finalmente, la fusión de las dos subunidades para completar la ALU. Apliquemos este procedimiento para diseñar una ALU que realice las siguientes operaciones sobre dos números (A y B) de n bits:

Operaciones Aritméticas	Operaciones Lógicas
A+B	AND
A-B	OR
A+1	XOR
A-1	NOT A

La primera decisión a tomar es cuánto debe valer k . En este caso 3, ya que son 8 el número de operaciones a seleccionar. $S_2 S_1 S_0$. Tal y como se ha expuesto en el párrafo anterior se designa uno de estos bits para diferenciar entre operaciones aritméticas o lógicas. Sea éste el bit de mayor peso S_2 . Así $S_1 S_0$ se utilizarán para diferenciar las cuatro operaciones de cada grupo.

Si S_2 es 0 se realizarán operaciones aritméticas y si su valor es 1, se realizaran lógicas.

El resto de asignaciones se hará según la siguiente tabla:

S_2	S_1	S_0	Función	Operaciones a realizar
0	0	0	A + B	$F_A = a_{n-1}, \dots, a_0 + b_{n-1}, \dots, b_0 + 00\dots 0$
0	0	1	A - B	$F_A = a_{n-1}, \dots, a_0 + \overline{b_{n-1}}, \dots, \overline{b_0} + 0\dots 01$
0	1	0	A + 1	$F_A = a_{n-1}, \dots, a_0 + 00\dots 01 = a_{n-1}, \dots, a_0 + 00 \dots 0 + 0\dots 1$
0	1	1	A - 1	$F_A = a_{n-1}, \dots, a_0 + 11\dots 1 + 00\dots 0$
1	0	0	AND	$F_L = a_i \cdot b_i ; i = 0, \dots, n-1$
1	0	1	OR	$F_L = a_i + b_i ; i = 0, \dots, n-1$
1	1	0	NOT A	$F_L = \overline{a_i}$
1	1	1	XOR	$F_L = a_i \oplus b_i ; i = 0, \dots, n-1$

Tabla 35

Los siguientes pasos serán diseñar la unidad aritmética y la lógica.

Unidad Aritmética

Si nos fijamos en las operacines aritméticas a realizar, se observa que siempre debe realizarse la misma operación: una suma, donde uno de los operandos siempre es el valor de A, el otro operando es variable y el acarreo inicial que varía de una operación a otra . Es decir,

$$F_A = A + ALGO = A + Y + C_0$$

El desarrollo de esta operación es:

$$\begin{array}{ccccccc}
 c_{n-1} & \dots & c_i & \dots & c_0 \\
 a_{n-1} & \dots & a_i & \dots & a_0 \\
 \hline
 c_n & y_{n-1} & \dots & y_i & \dots & y_0 \\
 \hline
 f_{An} & f_{An-1} & \dots & f_{Ai} & \dots & f_{A0}
 \end{array}$$

Ecuación 8

Donde se ha representado mediante un recuadro rojo los circutios básicos a diseñar. Así pues, se diseña el bloque i-ésimo ($f_{Ai} = a_i + y_i + c_i$) y la suma total consistirá en conectar en serie n bloques. Puesto que el valor de Y depende de la combinación de $S_1 S_0$, deberá representarse en el diagrama de bloques tal y como se muestra en la Figura 140. En ella pueden distinguirse dos subcircuitos; el donominado ALGO y el Full Adder. Sin embargo, unicamante habrá que diseñar el primero de ellos, ya que el diseño del Full Adder se realizó en el apartado de Sumadores.

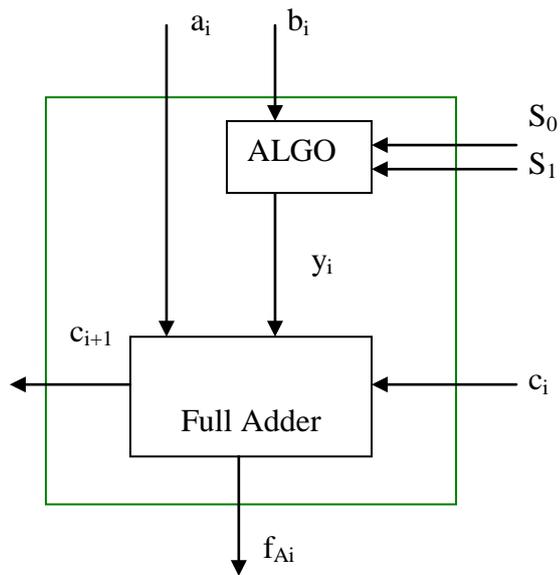


Figura 140

La tabla de verdad y circuito de ALGO serán:

S_1	S_0	b_i	y_i
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

$$y_i = S_0 \oplus \overline{S_1} b_i$$

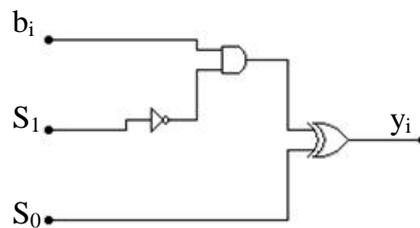


Figura 141

Si el número de bits de los operandos es n , entonces se cablearán en serie n circuitos tal y como se ilustra en la Figura 142.

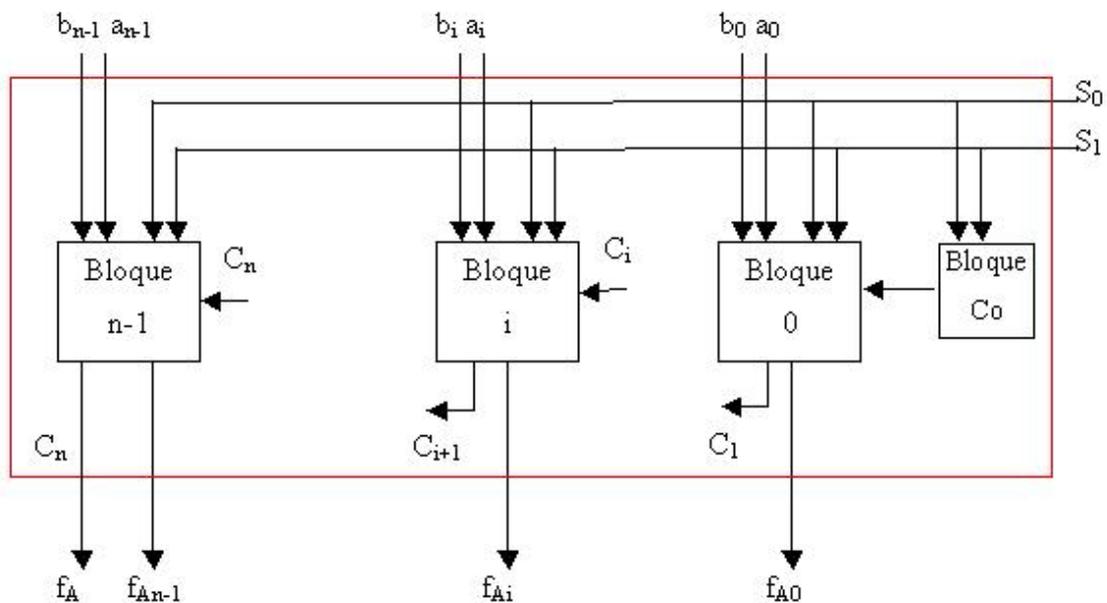


Figura 142

Resulta evidente la inclusión de un bloque extra denominado Bloque C_0 . Dicho bloque es necesario para la generación del acarreo inicial, que evidentemente, depende, también, de S_1 S_0 . Así, la tabla de verdad su diseño y la función lógica obtenida a partir de la misma será el siguiente:

S_0	S_1	C_0
0	0	0
0	1	1
1	0	1
1	1	0

$$C_0 = S_0 \oplus S_1$$

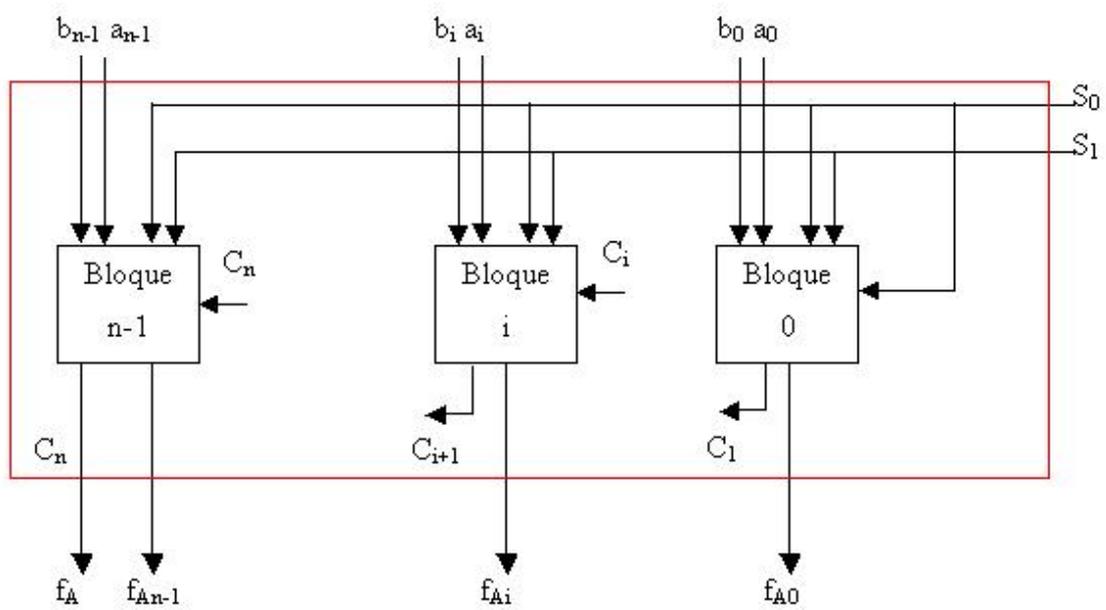
S_0	S_1
↓	↓
<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> Generador de acarreo inicial </div>	
C_0 ←	

Si en lugar de realizar la asignación de la Tabla 35 se Realiza la asignación representada en la Tabla 36, el circuito correspondiente al acarreo de entrada podría reducirse a la línea S_0 , ya que en esta nueva asignación se ha hecho coincidir el valor de ese bit con el valor del acarreo en cada operación.

S_2	S_1	S_0	Función	Operaciones a realizar
0	0	0	A + B	$F_A = a_{n-1}, \dots, a_0 + b_{n-1}, \dots, b_0 + 00\dots0$
0	0	1	A - B	$F_A = a_{n-1}, \dots, a_0 + \overline{b_{n-1}}, \dots, \overline{b_0} + 0\dots01$
0	1	0	A + 1	$F_A = a_{n-1}, \dots, a_0 + 11\dots1 + 00\dots0$
0	1	1	A - 1	$F_A = a_{n-1}, \dots, a_0 + 00\dots01 = a_{n-1}, \dots, a_0 + 00\dots0 + 0\dots01$

Tabla 36

Por ello, el circuito completo sería:



Unidad Lógica

Las operaciones lógicas deben realizarse entre cada pareja de bit i -ésimo de los operandos A y B. Por lo tanto, se diseña el circuito para cada pareja de dichos bits y posteriormente se repite para cada uno de los bits de los operandos A y B. El diagrama de bloques correspondiente se presenta en la Figura 143.

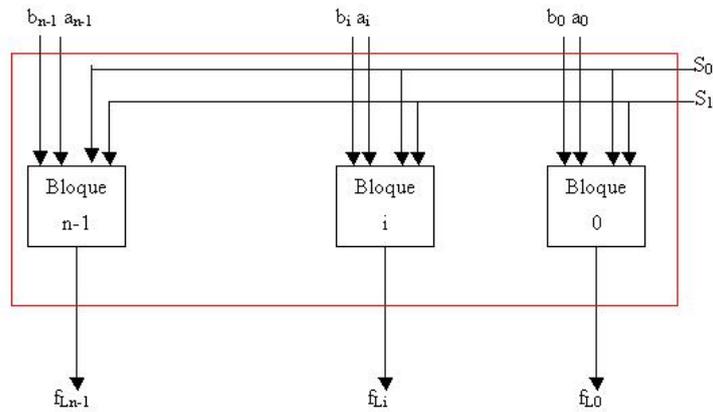


Figura 143

El diseño del bloque i -ésimo puede implementarse utilizando la función lógica obtenida a partir Tabla de Verdad (Tabla 37) ó utilizando un multiplexor de 4 a 1 (Figura 144), cuyas entradas de selección sean S_1S_0

S_1	S_0	a_i	b_i	$f_{L,i}$
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1

S_1	S_0	a_i	b_i	$f_{L,i}$
1	0	0	0	1
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

$$f_{L,i} = S_0 a_i \bar{b}_i + \bar{S}_1 a_i b_i + S_0 \bar{a}_i b_i + \bar{S}_0 S_1 \bar{a}_i$$

Tabla 37

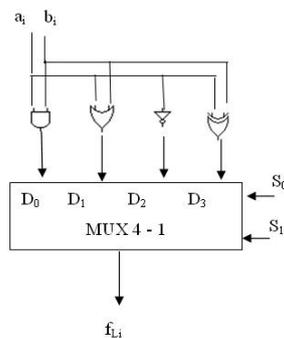


Figura 144

ALU Completa

Una vez diseñadas las unidades l3gica y aritm3tica, la salida de la ALU deber3 ser s3lo una de ellas. Para ello, basta con utilizar un multiplexor que seleccione en base a la entrada S_2 . El multiplexor ser3 tal que contenga n multiplexores de 2 a 1 l3neas.

