

UNIVERSIDAD DE GRANADA.

**ESCUELA TECNICA SUPERIOR DE
INGENIERIAS INFORMATICA Y DE
TELECOMUNICACIÓN.**



**Departamento de Arquitectura y
Tecnología de Computadores.**

**TECNOLOGÍA Y ORGANIZACIÓN DE
COMPUTADORES.**

**PRÁCTICA 3.
ANÁLISIS Y REALIZACIÓN DE UNA UNIDAD
ARITMÉTICO-LÓGICA (ALU) DE 4 BITS.**

1º GRADO EN INGENIERÍA INFORMÁTICA.

PRÁCTICA 3.

ANÁLISIS Y REALIZACIÓN DE UNA UNIDAD ARITMÉTICO-LÓGICA (ALU) DE 4 BITS.

Objetivos:

- Analizar teóricamente una Unidad Aritmético-Lógica de 4 bits a partir de los esquemáticos de los circuitos que la implementa. Montar y verificar su funcionamiento en un simulador lógico.

Material necesario para el desarrollo de la práctica:

- Guión de prácticas disponible en SWAD en el apartado ARCHIVOS>DOCUMENTOS>03.-PRACTICAS>PRACTICA_3>PRACTICA_3_TOC.PDF.
- Material del Tema 3º disponible en SWAD en el apartado ARCHIVOS>DOCUMENTOS>01.-TEORIA y PROBLEMAS>TEMA_3>03.-TEMA_3_TOC_SISTEMAS_COMBINACIONALES.PDF. Apartado 3.4.2. ALU y 3.4.4.- Multiplexores/Demultiplexores.
- Videoclase del Tema 3. Clase 1 (minutos 34 a 38) ubicada en:
<https://drive.google.com/open?id=1LYnJsHz1OUzPIDzAcr6vlaYK1XLKCTeb>
- Videoclase del Tema 3. Clase 2 (minutos 9 a 16) ubicada en:
<https://drive.google.com/open?id=1WJY2ZxF5grFL4WfDU2vcmlVtuPiZXT7O>
- Seminario 5. Guía de Trabajo Autónomo. PARTE 1: INTRODUCCIÓN AL MANEJO DE UN SIMULADOR LÓGICO, páginas 1-3 a 1-10 (ambas inclusive) disponible en SWAD en el apartado ARCHIVOS>DOCUMENTOS>02.-SEMINARIOS>SEMINARIO_5>05.-SEMINARIO_5_TOC_SIMULADOR_ENTRENADOR_LOGICO_GUIA.
- Videoclases de Simulador Lógico disponibles en las direcciones siguientes:
https://drive.google.com/open?id=1OsHIQ51fdcfGDe0p_6b7LtTTbwpovXop
<https://drive.google.com/open?id=1gvQlrsORnpHDrgPwKSyvEi2vprk6pul->
https://drive.google.com/open?id=16hp2vr4GTzM4j0_wm1KFXIB0FjWfnmFn
<https://drive.google.com/open?id=1tIFeDH0nthSq09fc75XL1NSP22aDifh5>
- Software Simulador Lógico LogicWorks.

3.1. Estudio preliminar:

Se pretende analizar, realizar y verificar, utilizando un simulador lógico, una Unidad Aritmético-Lógica (ALU) que sea capaz de operar con 2 datos A y B de 4 bits. Esta ALU proporcionará las operaciones aritméticas y lógicas indicadas en la *Figura 3.1*. Cada operación se seleccionará según el valor que tomen (1 ó 0) unas señales de control S_2, S_1, S_0 y un acarreo de entrada C_{in} .

$G = A - 1$	Decrementar A
$G = A + B$	Sumar: A más B
$G = A$	Transferir A
$G = \bar{A}$	Transferir complemento de A
$G = A \wedge B$	Operación A AND B
$G = A + B + 1$	Sumar A más B más 1
$G = A \oplus B$	Operación: A EXOR B
$G = A + \bar{B}$	Sumar A con el complemento a 1 de B
$G = A + 1$	Incrementar A
$G = A \vee B$	Operación: A OR B
$G = A + \bar{B} + 1$	Restar: $A - B$ (en complemento a 2)

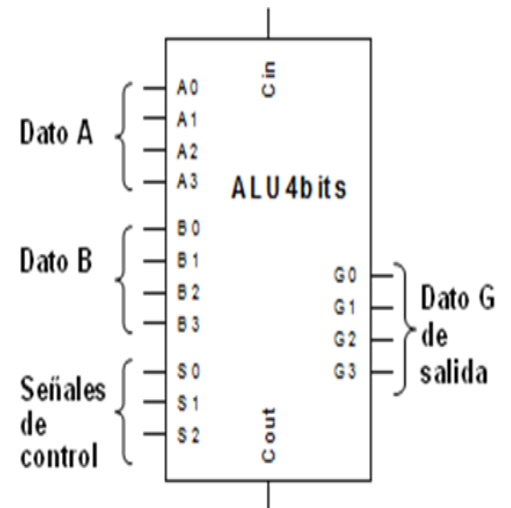
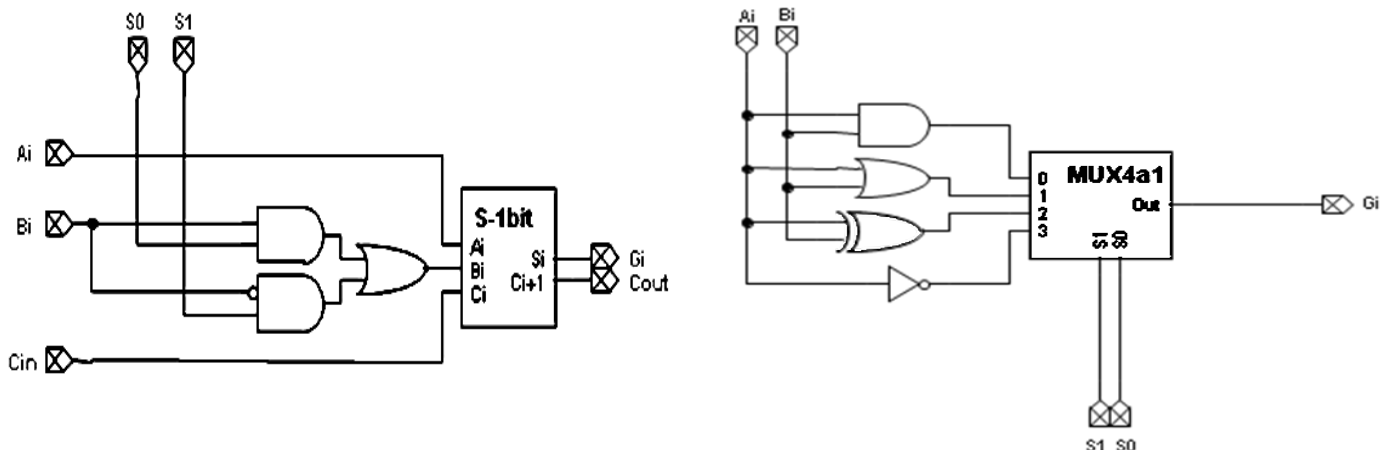


Figura 3.1: Operaciones que realiza la ALU. Esquema general de entradas y salidas.

La estructura jerárquica modular mediante Logic Works de la ALU se establece en las *Figuras 3.2, 3.3 y 3.4*. Se parte de los circuitos de la *Figura 3.2(a) y 3.2(b)* que corresponden a las etapas aritmética y lógica, respectivamente. Ambas operan sobre datos de un bit. La etapa aritmética incluye a un sumador completo de un bit (S-1bit) y la etapa lógica un multiplexor 4:1. Estos módulos se “encapsulan” dando lugar a los bloques indicados en la parte inferior de las *Figura 3.2(a) y (b)*.



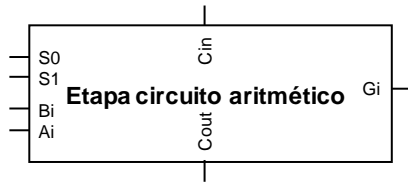


Figura 3.2(a)

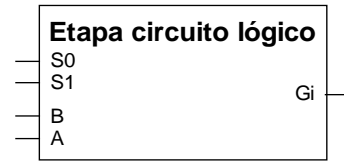


Figura 3.2(b)

Figura 3.2. Estructura modular de cada elemento aritmético ó lógico de 1 bit.

En la Figura 3.3 se ilustra el diseño y “encapsulado” de la Etapa ALU de un bit que integra los módulos anteriores de la Figura 3.2.

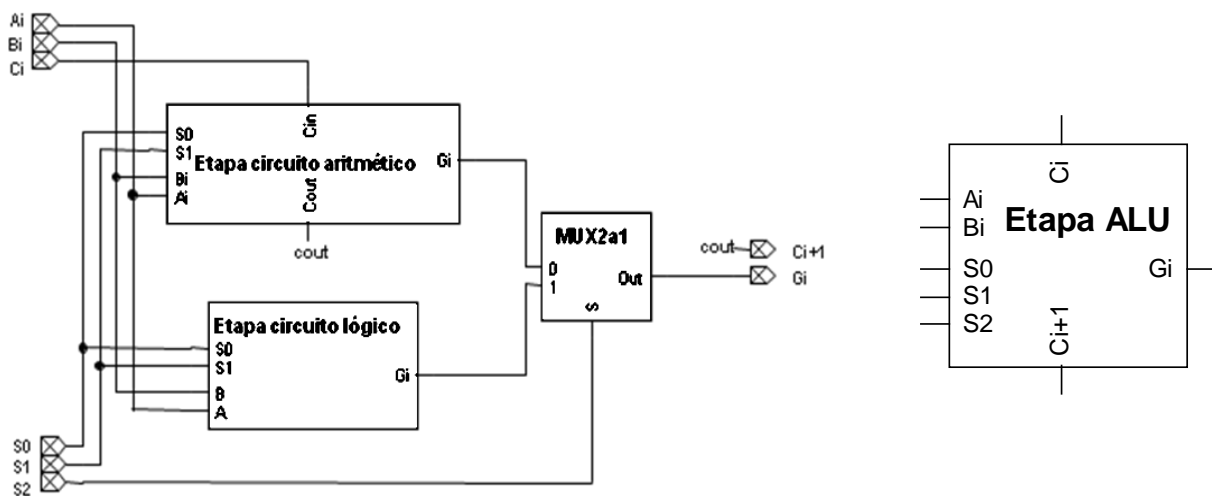


Figura 3.3. Estructura de la Etapa ALU de un bit.

En la Figura 3.4 se ilustra el diseño y “encapsulado” de una ALU de 4 bits, utilizando 4 Etapas ALU de un bit, como las diseñadas en la Figura 3.3.

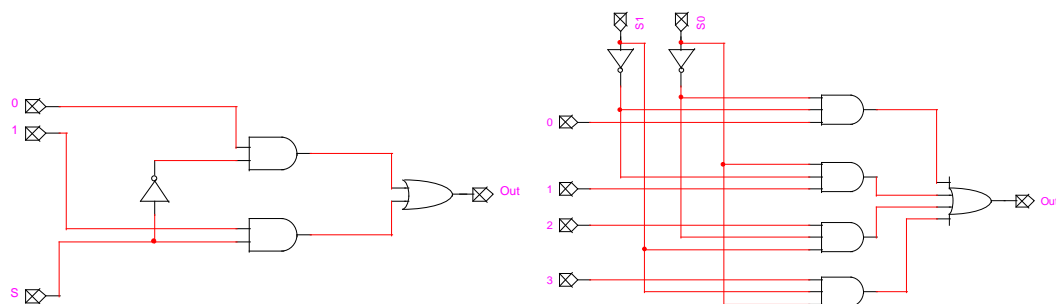


Figura 3.3 (bis). Estructura de un MUX 2 a 1 (izquierda) y de un MUX 4 a 1 (derecha).

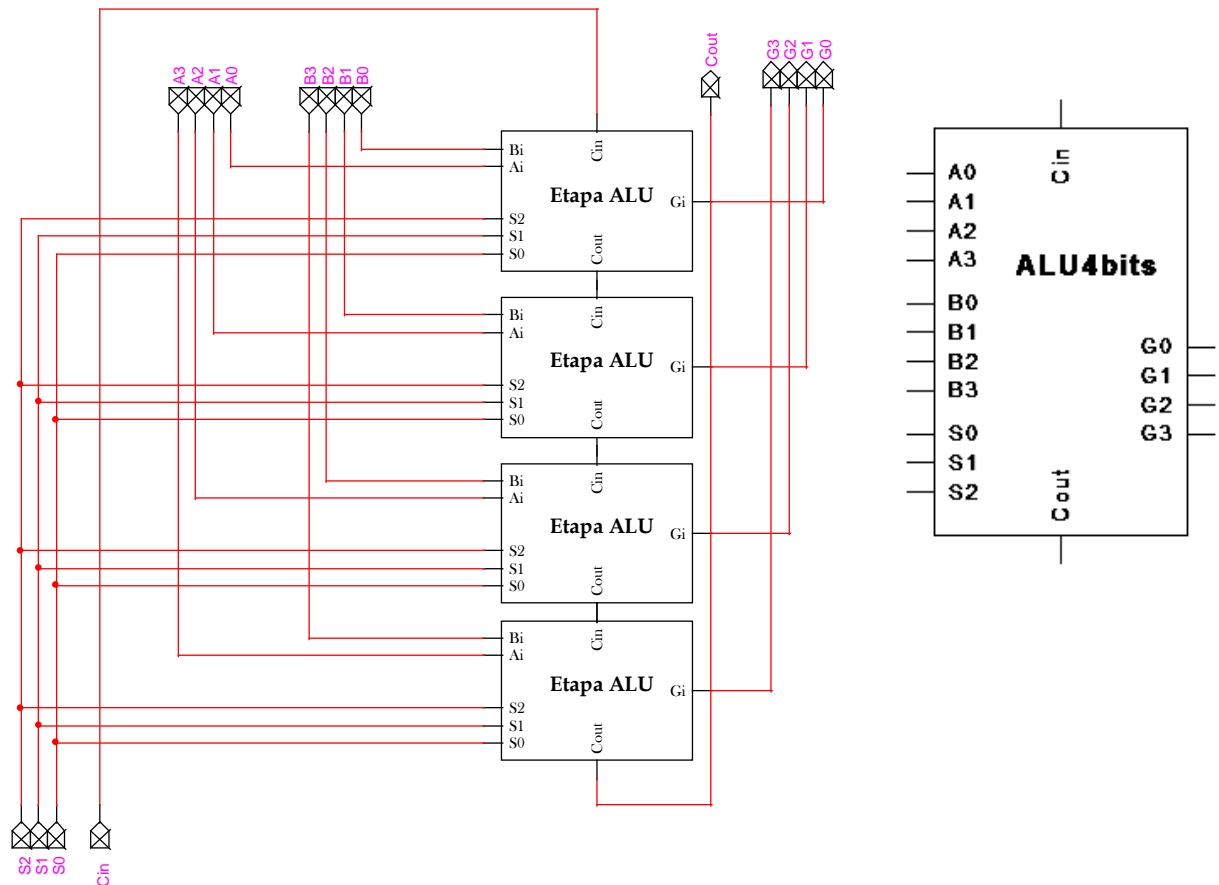


Figura 3.4. Estructura completa de la ALU para palabras de 4 bits.

3.2. Realización práctica:

1. Trabajo para realizar ANTES de la sesión de prácticas (debe venir preparado):
 - 1.1. **Analice teóricamente** los esquemáticos de los circuitos y etapas de la ALU mostradas en las Figuras 3.2, 3.3 y 3.4. Deduzca, razonadamente, para qué combinaciones de las señales de control S_2 , S_1 , S_0 y valor de C_{in} se realiza cada una de las operaciones indicadas en la Figura 3.1. El resultado indíquelo en la columna de la Tabla 3.1 correspondiente a “Operaciones de la ALU deducidas Teóricamente”.
 - 1.2. A la vista del apartado anterior, con las “Operaciones ALU deducidas Teóricamente” y tomando como datos A y B de entrada de 4 bits los siguientes $A = 1101$ y $B = 0111$, obtenga, para cada una de las operaciones obtenidas en el apartado anterior, el resultado (dato G) que debería obtenerse.

SEÑALES $S_2S_1S_0C_{in}$	Operaciones de la ALU, deducidas Teóricamente	Operaciones de la ALU deducidas Experimentalmente
0 0 0 0 0 0 0 1 0 0 1 0 0 0 1 1		
0 1 0 0 0 1 0 1 0 1 1 0 0 1 1 1		
1 0 0 0 1 0 0 1 1 0 1 0 1 0 1 1		
1 1 0 0 1 1 0 1 1 1 1 0 1 1 1 1		

Tabla 3.1

- Realice en LogicWorks la ALU de la *Figura 3.4* y **el circuito de prueba de la Figura 3.5**. Para ello, use (de la biblioteca *Simulation IO.clf*) dos componentes HEX KEYBOARD wo/STB para generar los datos $A_3A_2A_1A_0$ y $B_3B_2B_1B_0$, un componente HEX_DISPLAY para visualizar las salidas $G_3G_2G_1G_0$, cuatro BINARY SWITCH para generar las señales $S_2S_1S_0C_{in}$, y un BINARY PROBE para visualizar la salida C_{out} y así realizar la prueba experimental del circuito.
- En el Laboratorio, introduzca los ejemplos, deducidos teóricamente en el apartado (1.2), en el circuito de prueba y **deduzca experimentalmente** los valores de las señales de control S_2, S_1, S_0 y valor de C_{in} correspondiente a cada una de las operaciones indicadas en la *Figura 3.1* y **rellene la última columna** de la *Tabla 3.1*. Compruebe que los resultados obtenidos experimentalmente coinciden con los obtenidos en el apartado 1.2.

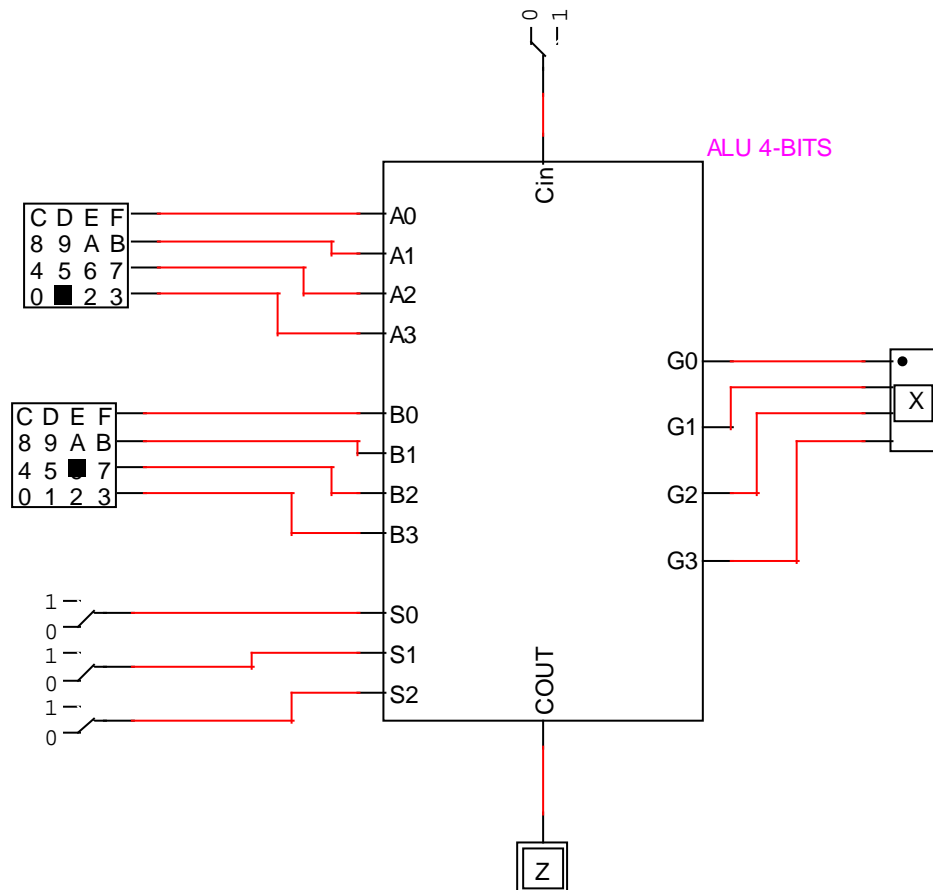


Figura 3.5. Circuito de prueba de la ALU de 4 bits.

