

Universidad Autónoma de Baja California
Facultad de Ciencias Químicas e Ingeniería



ORGANIZACIÓN DE LAS COMPUTADORAS Y LENGUAJE ENSAMBLADOR

Practica 3

Diseño de una Unidad Aritmética Lógica de 8 bits

Docente: Sanchez Herrera Mauricio Alonso

Alumno: Gómez Cárdenas Emmanuel Alberto

Matricula: 1261509

Contenido

TEORIA.....	3
Unidad Aritmética Lógica.....	3
Unidad de Coma Flotante.....	3
Condiciones de Sobreflujo (Overflow)	4
Regla de sobreflujo para la suma.....	4
Regla de sobreflujo para la resta	4
Entradas y Salidas de una ALU	4
Entradas.....	4
Salidas	4
Funciones Básicas.....	5
DESARROLLO.....	6
Circuito ALU implementado	6
Implementación de la ALU como circuito integrado.....	6
CONCLUSIONES.....	7
ENLACE AL CIRCUITO	7
REFERENCIAS.....	7

TEORIA

Unidad Aritmética Lógica

La unidad aritmética lógica (arithmetic logic unit, ALU) es la parte de la unidad de central de procesamiento (central processing unit, CPU) que realiza las operaciones lógicas y aritméticas. En algunas implementaciones, esta se divide en unidad aritmética (AU) y unidad lógica (LU). Algunos procesadores cuentan con más de una AU, una para tratar enteros y la otra para las operaciones de coma flotante (FPU).

Las entradas de la ALU en las que se hacen las operaciones, son conocidas como operandos. En muchos diseños la ALU cuenta con entradas y, o salidas de estado, que transmiten información sobre una operación anterior o actual, entre la ALU y los registros de estado externos.

Comúnmente, la ALU tiene acceso de entrada y salida directos al controlador del procesador, la memoria principal (RAM) y los dispositivos de entrada y salida. En general, la ALU incluye almacenamiento para los operandos, el resultado acumulado y los resultados desplazados.

Unidad de Coma Flotante

La unidad de coma flotante (Floating-point unit, FPU) también conocida como coprocesador matemático es parte de un sistema diseñado especialmente para realizar operaciones con números de coma flotante. Las operaciones típicas de esta unidad son: suma, resta, multiplicación, división y raíz cuadrada. Algunas implementaciones incluyen funciones trascendentales como lo son los cálculos exponenciales o trigonométricos. Aunque debido a que la precisión puede ser muy baja, algunos sistemas prefieren calcular estas funciones en software.

En algunas arquitecturas, las operaciones de coma flotante se tratan de forma completamente diferente a las operaciones con valores enteros, con registros dedicados y tiempos de ciclo diferentes. En algunos casos, las FPU's pueden ser especializados y divididas en operaciones de coma flotante sencillas (como lo son la suma y la multiplicación) y las complejas (como la división). También hay ocasiones en las que solo las operaciones básicas se implementan en hardware y las operaciones complejas solo se implementan mediante software.

Condiciones de Sobreflujo (Overflow)

El sobreflujo ocurre cuando más información de la que el hardware puede manejar es transmitida, en el caso de la ALU un overflow de enteros ocurre cuando una operación produce un resultado que está fuera del rango de lo que puede ser representado.

Regla de sobreflujo para la suma

Si dos números de complemento a 2 son sumados y ambos tienen el mismo signo (ambos son positivos o negativos), entonces ocurre el sobreflujo si y solo si el resultado tiene un signo diferente.

Sumar dos números positivos debe dar positivo, sumar dos números negativos debe dar negativo.

El sobreflujo ocurre si:

$$(+A) + (+B) = -C \quad \text{ó} \quad (-A) + (-B) = +C$$

Regla de sobreflujo para la resta

Si se restan dos números de complemento a 2 y sus signos son diferentes, entonces el sobreflujo ocurre si y solo si el resultado tiene el mismo signo que el substraendo (el número que se está restando)

El sobreflujo ocurre si:

$$(+A) - (-B) = -C \quad \text{ó} \quad (-A) - (+B) = +C$$

Entradas y Salidas de una ALU

Una ALU cuenta generalmente con 4 entradas y 4 salidas estas son:

Entradas

- A y B: Son los operandos, en este caso de 8 bits cada uno.
- Cin: Es el bit carry de entrada, este es utilizado solo en ciertas operaciones.
- Función: Esta entrada indica la operación a realizar, al ser de 4 bits podemos tener como máximo 16 operaciones diferentes.

Salidas

En el caso de las salidas la ALU cuenta con 4 salidas. R indica el resultado obtenido al realizar la operación indicada por la entrada Función. Las últimas 3 salidas son utilizadas como banderas, estas son:

- Cout: Carry-out o carry de salida, es el bit de “carry” resultante de una suma, o el “borrow” de una resta.
- Zero: indica que todos los bits de R son cero.
- Ov: Overflow, esta bandera es activada cuando la operación aritmética excede el rango numérico de R.

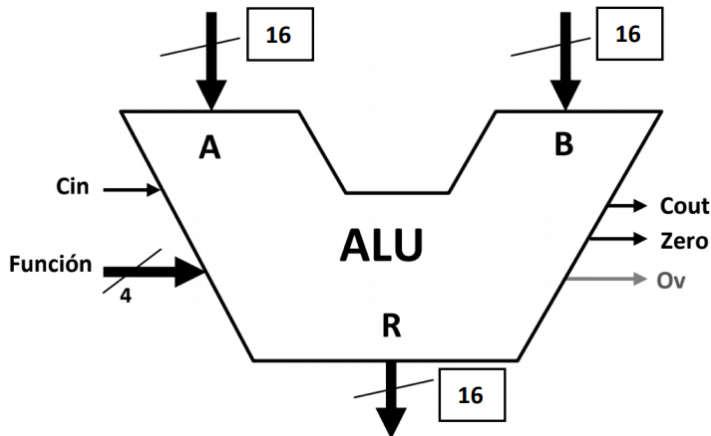


Tabla 1. Funciones básicas de un ALU

AND A,B	ADC A,B	SHL A
OR A,B	SUB B,A	SHR A
XOR A,B	SBB B,A	ROL A
NOT A	INC A	ROR A
ADD A,B	DEC A	

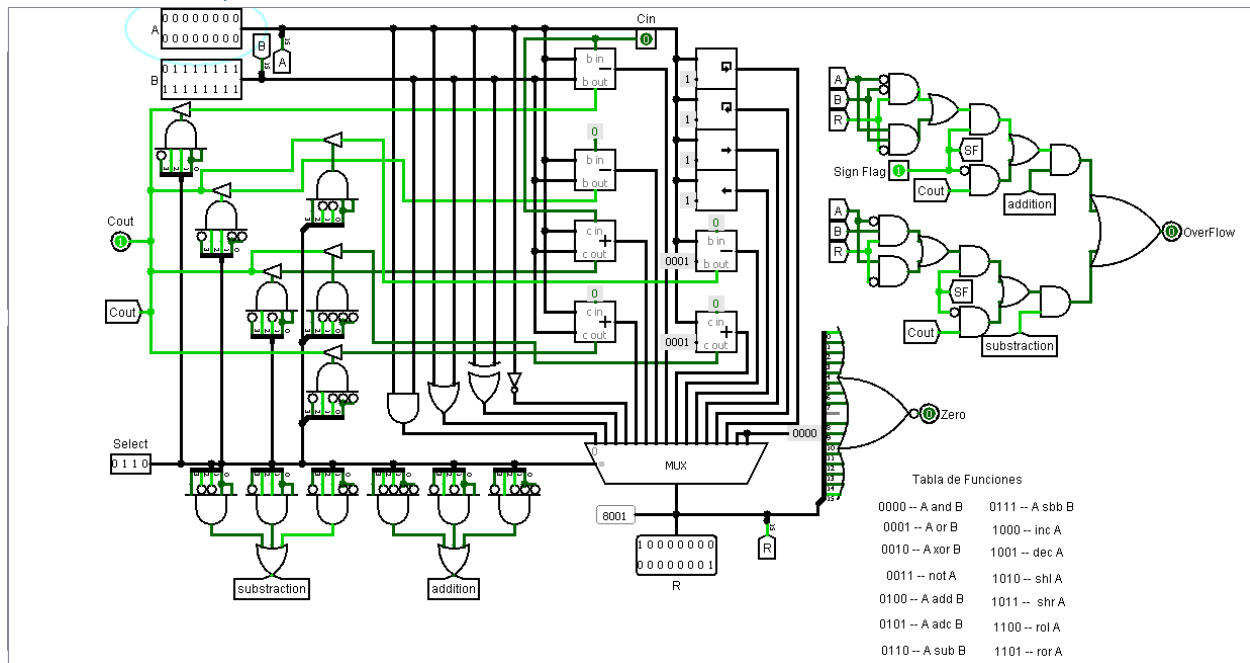
Funciones Básicas

Las funciones básicas comúnmente soportadas por cualquier implementación de una ALU son:

- AND A, B (A & B): Operación lógica AND bit a bit entre A y B.
- OR A, B (A | B): Operación lógica OR bit a bit entre A y B.
- XOR A, B (A \oplus B): Operación lógica XOR bit a bit entre A y B.
- NOT A (\neg A): Negación lógica (NOT) bit a bit entre A y B.
- ADD A, B (A+B): Suma aritmética entre A y B.
- ADC A, B (A+B+Cin): Suma aritmética entre A, B y el carry de entrada.
- SUB B, A (A-B): Resta aritmética a "A" de "B" (a "A" se le resta B).
- SBB B, A (A-B-Cin): Resta aritmética a "A" de "B" y "Cin". (a "A" se le restan "B" y "Cin").
- INC A (A+1): Incremento del valor de A por una unidad.
- DEC A (A-1): Decremento del valor de A por una unidad.
- SHL A (A<<1): Se le aplica un corrimiento a la izquierda al valor de A.
- SHR A (A>>1): Se le aplica un corrimiento a la izquierda al valor de A.
- SHR A (A>>1): Se le aplica una rotación a la izquierda al valor de A (en la rotación a la derecha, el bit menos significativo termina en la posición del bit más significativo).
- SHR A (A>>1): Se le aplica una rotación a la izquierda al valor de A (en la rotación a la derecha, el bit más significativo termina en la posición del bit menos significativo).

DESARROLLO

Circuito ALU implementado

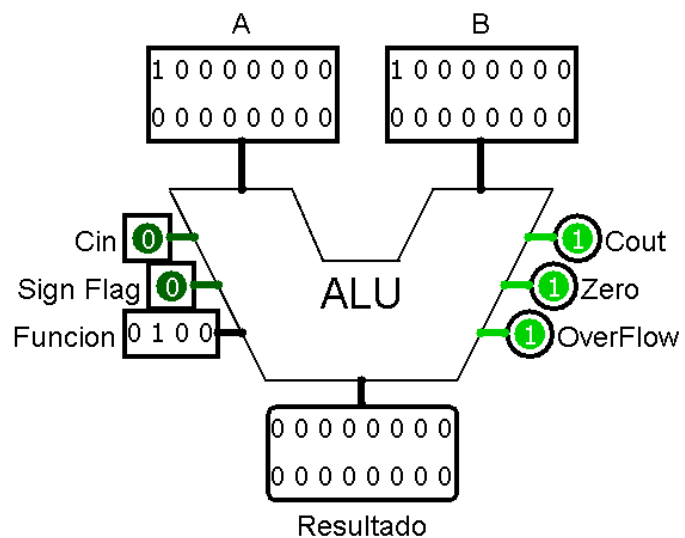


Para poder calcular la bandera Overflow de manera correcta al manipular valores con o sin bit, he agregado una entrada como bandera para indicarle a la ALU cuando está tratando con valores signed o unsigned.

Implementación de la ALU como circuito integrado

Tabla de Funciones

0000 -- A and B	0111 -- A sbb B
0001 -- A or B	1000 -- inc A
0010 -- A xor B	1001 -- dec A
0011 -- not A	1010 -- shl A
0100 -- A add B	1011 -- shr A
0101 -- A adc B	1100 -- rol A
0110 -- A sub B	1101 -- ror A



CONCLUSIONES

La ALU es una de las partes más importantes en un procesador, ya que, sin este, el procesador sería incapaz de realizar las operaciones mas básicas.

En esta práctica se aprendió lo complicado e interesante que fue obtener la bandera de Overflow y también lo interesante que es poder transformar señales tan básicas como encendido y apagado, a algo tan abstracto como lo son las matemáticas. Aunque en este caso solo hayan sido operaciones aritméticas básicas, es importante observar y analizar lo tanto que se ha logrado con este tipo de circuitos digitales.

ENLACE AL CIRCUITO

https://drive.google.com/file/d/1T6iL_fyZktiaSZ2vkJxORN3icAly6Fa5/view?usp=sharing

REFERENCIAS

Arithmetic logic unit. En.wikipedia.org.

https://en.wikipedia.org/wiki/Arithmetic_logic_unit.

What is arithmetic-logic unit (ALU)? - Definition from WhatIs.com.

<https://whatis.techtarget.com/definition/arithmetic-logic-unit-ALU>.

Floating-point unit. En.wikipedia.org. https://en.wikipedia.org/wiki/Floating-point_unit.

Arithmetic Operations on Binary Numbers - from doc.ic.ac.uk.

<https://www.doc.ic.ac.uk/~eedwards/compsys/arithmetic/>