

Laboratorio 3: Lógica combinacional y Aritmética I

Fiorella Delgado León Jonathan Guzmán Araya, Gerald Valverde McKenzie

Instituto Tecnológico de Costa Rica

Área Académica Ingeniería en Computadores

fio@gmail.com jonathana1196@gmail.com gerald@gmail.com

Abstract—

Index Terms—ALU, compuertas lógicas, flip-flops, microprocesadores

I. INTRODUCCIÓN

Los circuitos combinacionales son aquellos en los que las salidas solo dependen solamente de las entradas, y no de ningún tipo de sincronización con señales de reloj, esto hace que los sistemas combinacionales sean generalmente rápidos. En sistemas digitales complejos, como los microprocesadores, los circuitos de lógica combinacional desempeñan un papel fundamental. La arquitectura de un microprocesador es como la que se observa en la Figura 1.

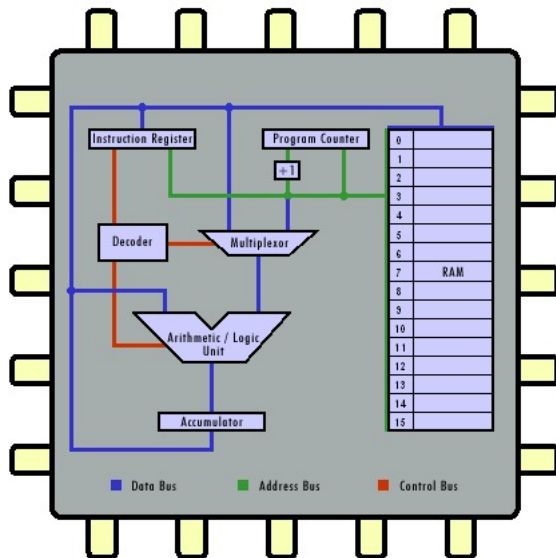


Fig. 1. Arquitectura básica de un microprocesador [1]

Una función esencial de muchas computadoras (microprocesadores) y calculadoras es la realización de operaciones lógicas y aritméticas. Estas operaciones se efectúan en la unidad aritmética-lógica de una computadora, donde se combinan compuertas lógicas y flip-flops de manera que puedan sumar, restar, multiplicar y dividir números binarios [2]. El símbolo de una ALU es como el de la Figura 2.

Además de los operadores lógicos, la ALU cuenta con una serie de registros para almacenar los datos y bits de información sobre los resultados, también llamados banderas [3], las banderas más comunes en una ALU son: Z, C, V, N.

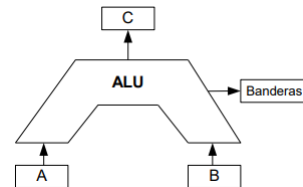


Fig. 2. Símbolo ALU [3]



Fig. 3. Compuerta NOT y su representación en álgebra de Boole [2]

- Zero flag: usada para determinar si dos valores son iguales, generalmente se usa una resta, y se revisa si el resultado es cero, si es así, esta bandera se enciende.
- Overflow flag: usada para determinar si el resultado de una operación (usualmente suma o multiplicación), tiene un tamaño superior al máximo representable, o inferior al mínimo representable en la ALU, es posible que existan como banderas separadas.
- Negative flag: usada para determinar si el resultado de una resta es menor a cero
- Carry/Borrow flag: Esta bandera indica cuando una operación resulta en un valor más largo del cual el acumulador puede representar o menor del cual el acumulador puede representar.

La ALU es simplemente un operador, es decir solo realiza operaciones, esta no toma decisiones, acepta datos binarios que están almacenados en la memoria y ejecuta operaciones con estos datos, de acuerdo a las instrucciones que vienen de la unidad de control.

Para realizar operaciones lógicas la ALU utiliza compuertas lógicas como:

- NOT: La Figura 3 muestra el símbolo para un circuito NOT, al cual se le conoce más comúnmente como INVERTER, el inversor afecta a una señal de entrada. Invierte (complementa) la señal de entrada en todos los puntos de la forma de onda, por lo que siempre que la *entrada* = 0, la *salida* = 1, y viceversa [2].
- AND: La Figura 4 muestra el símbolo lógico para una compuerta AND de dos entradas. La salida de la compuerta AND es igual al producto AND de las entradas

Q	Q'
0	1
1	0

TABLE I

TABLA DE VERDAD DE LA COMPUERTA NOT

lógicas; es decir, $x = AB$. En otras palabras, la compuerta AND es un circuito que opera de manera que su salida esté en ALTO sólo cuando todas sus entradas se encuentren en ALTO. Para todos los demás casos, la salida de la compuerta AND estará en BAJO [2].

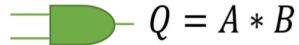


Fig. 4. Compuerta AND y su representación en álgebra de Boole [2]

A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

TABLE II

TABLA DE VERDAD DE LA COMPUERTA AND

- OR: La Figura 5 muestra el símbolo lógico para una compuerta OR de dos entradas. Las entradas A y B son niveles lógicos de voltaje y la salida x es un nivel lógico de voltaje cuyo valor es el resultado de la operación OR sobre A y B; es decir, $x = A + B$. En otras palabras, la compuerta OR opera de manera que su salida esté en ALTO, 1 lógico, si cualquiera de las entradas A o B o ambas se encuentran en el nivel 1 lógico. La salida de la compuerta OR estará en BAJO, 0 lógico, sólo si todas sus entradas están en 0 lógico [2].



Fig. 5. Compuerta OR y su representación en álgebra de Boole [2]

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

TABLE III

TABLA DE VERDAD DE LA COMPUERTA OR

- XOR: También llamada OR exclusiva, esta actúa como una suma binaria de un dígito cada uno y el resultado de la suma sería la salida. Otra manera de verlo es que con valores de entrada igual el estado de salida es 0 y con valores de entrada diferente, la salida será 1.

II. CONCLUSIONES

III. BIBLIOGRAFÍA

REFERENCES

- [1] A. J. F. González. (2017, Jul.) Arquitectura de microprocesadores desde el principio. beBee. [Online]. Available: <https://www.bebec.com/producer/@http-alfredo-j-feijoo-webnode-es-contacto/arquitectura-de-microprocesadores-desde-el-principio>



Fig. 6. Compuerta XOR y su representación en álgebra de Boole [2]

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

TABLE IV

TABLA DE VERDAD DE LA COMPUERTA XOR

- //www.bebec.com/producer/@http-alfredo-j-feijoo-webnode-es-contacto/arquitectura-de-microprocesadores-desde-el-principio
- [2] G. M. Tocci, Ronald J. Neal S. Widmer, *Sistemas digitales: principios y aplicaciones*, L. M. C. Castillo, Ed. Pearson Educación, 2007.
- [3] D. A. G. García. Procesadores. Departamento de Mecatrónica. [Online]. Available: <http://homepage.cem.itesm.mx/garcia.andres/PDF201411/ArquitecturaComputacional.pdf>
- [4] D. M. H. Sarah L. Harris, *Digital Design and Computer Architecture*. Morgan Kaufmann, 2010.