

Suma de Bits

Héctor M. Solís Villodas
Jorge L. Morales Ortiz

Abstracto

Uno de los requisitos claves en los ordenadores digitales es usar funciones lógicas para llevar a cabo operaciones aritméticas. La operación básica es la suma. Si podemos hacer sumas, también podemos hacer resta, multiplicación y división. En este experimento utilizaremos circuitos lógicos para sumar bits.

I. Introducción

El 74LS83A es un “4 Bit Binary Full Adder” de alta velocidad, con un carry interno. Este acepta dos palabras de 4 bits binario (A_1 - A_4 , B_1 - B_4) y un carry input (C_0). Este genera la suma binaria por los outputs (Σ_1 - Σ_4) y un carry output (C_4) desde el most significant bit. El 74LS83A opera con cualquiera: active high o active low (logica positiva o negativa). El 74LS83 es recomendado para nuevos diseños.

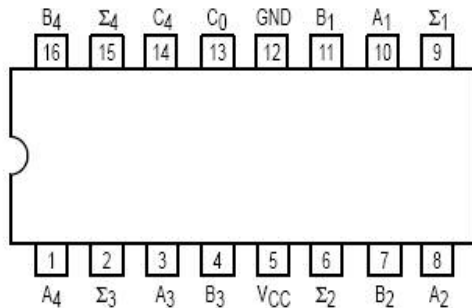


Figura 1 – Diagrama del 74LS83A

Descripción de la función del dispositivo:

El 74LS83A suma dos palabras de 4 bits binario ($A + B$) añadiendo el carry sobrante. La suma binaria aparece en la suma de outputs (Σ_1 - Σ_4) y sacando el carry sobrante por el output (C_4).

II. Experimento

Descripción del problema:

Diseñar un programa que sume dos números de dos bits cada uno.

III. Análisis de Datos

Tabla 1 – Tabla de verdad representando el problema planteado

A1	A0	B1	B0	C	S1	S0
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	0	1	0
0	0	1	1	0	1	1
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	0	1	1
0	1	1	1	1	0	0
1	0	0	0	0	1	0
1	0	0	1	0	1	1
1	0	1	0	1	0	0
1	0	1	1	1	0	1
1	1	0	0	0	1	1
1	1	0	1	1	0	0
1	1	1	0	1	0	1
1	1	1	1	1	1	0

Funciones Resultantes –

$$S0 = (\overline{A1} \cdot \overline{A0} \cdot \overline{B1} \cdot B0) + (\overline{A1} \cdot \overline{A0} \cdot B1 \cdot B0) + (\overline{A1} \cdot A0 \cdot \overline{B1} \cdot \overline{B0}) + (\overline{A1} \cdot A0 \cdot B1 \cdot \overline{B0}) + (A1 \cdot \overline{A0} \cdot \overline{B1} \cdot B0) + (A1 \cdot \overline{A0} \cdot B1 \cdot B0) + (A1 \cdot A0 \cdot \overline{B1} \cdot \overline{B0}) + (A1 \cdot A0 \cdot B1 \cdot \overline{B0})$$

$$S1 = (\overline{A1} \cdot \overline{A0} \cdot B1 \cdot \overline{B0}) + (\overline{A1} \cdot \overline{A0} \cdot B1 \cdot B0) + (\overline{A1} \cdot A0 \cdot \overline{B1} \cdot B0) + (\overline{A1} \cdot A0 \cdot B1 \cdot \overline{B0}) + (A1 \cdot \overline{A0} \cdot \overline{B1} \cdot \overline{B0}) + (A1 \cdot \overline{A0} \cdot \overline{B1} \cdot B0) + (A1 \cdot A0 \cdot \overline{B1} \cdot \overline{B0}) + (A1 \cdot A0 \cdot B1 \cdot B0)$$

$$C = (\overline{A1} \cdot A0 \cdot B1 \cdot B0) + (A1 \cdot \overline{A0} \cdot B1 \cdot \overline{B0}) + (A1 \cdot \overline{A0} \cdot B1 \cdot B0) + (A1 \cdot A0 \cdot \overline{B1} \cdot B0) + (A1 \cdot A0 \cdot B1 \cdot \overline{B0}) + (A1 \cdot A0 \cdot B1 \cdot B0)$$

K-maps de las funciones resultantes –

S0 →

		B1B0			
		$\overline{B1}\overline{B0}$	$\overline{B1}B0$	$B1\overline{B0}$	$B1B0$
$\overline{A1}A0$	$\overline{A1}\overline{A0}$		1	1	
	$\overline{A1}A0$	1			1
	$A1\overline{A0}$				
	$A1A0$		1	1	

S1 →

		B1B0			
		$\overline{B1}\overline{B0}$	$\overline{B1}B0$	$B1\overline{B0}$	$B1B0$
$\overline{A1}A0$	$\overline{A1}\overline{A0}$			1	1
	$\overline{A1}A0$		1		1
	$A1\overline{A0}$	1			
	$A1A0$	1	1		

C →

		B1B0			
		$\overline{B1}\overline{B0}$	$\overline{B1}B0$	$B1\overline{B0}$	$B1B0$
$\overline{A1}A0$	$\overline{A1}\overline{A0}$				
	$\overline{A1}A0$			1	
	$A1\overline{A0}$		1	1	1
	$A1A0$			1	1

Funciones Simplificadas –

$$S0 = (A0 \oplus B0)$$

$$S1 = \overline{A0}(A1 \oplus B1) + \overline{B0}(A1 \oplus B1) + A0 \cdot B0(\overline{A1} \oplus \overline{B1})$$

$$C = (A1 \cdot A0 \cdot B0) + (A0 \cdot B1 \cdot B0) + (A1 \cdot B1)$$

IV. Conclusión

En este experimento trabajamos con el circuito integrado 74LS83A para sumar 4 bits y 8 bits. También diseñamos un programa en Visual Basic para sumar dos bits. Las funciones utilizadas en el programa fueron obtenidas a través del álgebra booleana y los k-maps. El uso de chips, como el 74LS83A, minimiza la complejidad de los circuitos, reduciendo así el diseño y análisis de los mismos.

V. Referencias

(1)http://descartes.cnice.mecd.es/taller_de_matematicas/logica_digital/Logica%20Digital_5.htm

(2)<http://www.cs.ucla.edu/classes/spring02/cs152A/l3/74ls83.pdf>