**Trabajo Práctico N° 1:**

**Operaciones y Circuitos Lógicos.**

**Ejercicio 1.**

*Realizar las siguientes operaciones lógicas:*

*(Nota: Se opera lógicamente con los bits ubicados en la misma posición del o de los operandos.)*

00010001 AND 01011100 = 00010000

01010101 AND 01010101 = 01010101

01010101 AND 10101010 = 00000000

11110000 AND 11111111 = 11110000

01010101 OR 01010101 = 01010101

01010101 OR 10101010 = 11111111

11110001 OR 11110010 = 11110011

01010101 XOR 01010101 = 00000000

01010101 XOR 10101010 = 11111111

00001111 XOR 00000000 = 00001111

NOT 11111111 = 00000000

NOT 01000000 = 10111111

NOT 00001110 = 11110001

**Ejercicio 2.**

*Si DATO “operación\_lógica” MASK = RESULTADO, determinar la operación lógica y el valor de MASK tal que RESULTADO sea el indicado:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DATO** | **Op. lógica** | **MASK** |  | **RESULTADO** |
|  | OR | 11100111 | = | 111111 |
|  | OR | 00001000 | = | 1 |
|  | AND | 01111111 | = |  |
|  | XOR  XNOR | 01010000  10101111 | = |  |

**Ejercicio 3.**

*Analizar los siguientes esquemas y determinar los valores de las salidas C y D para todas las combinaciones de entrada (A y B o A, B y IN). ¿Se puede asociar los resultados obtenidos con una operación aritmética?*

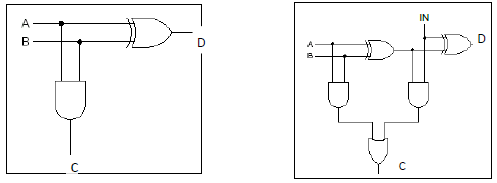


Figura 1:

C= AND (A, B)

C= A AND B

C= A \* B.

D= XOR (A, B)

D= A XOR B

D= A B.

|  |  |  |  |
| --- | --- | --- | --- |
| **A** | **B** | **C** | **D** |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 |

Figura 2:

C= OR (AND (A, B), AND (XOR (A, B), IN))

C= (A AND B) OR ((A XOR B) AND IN)

C= A \* B + (A B) \* IN.

D= XOR (XOR (A, B), IN)

D= (A XOR B) XOR IN

D= (A B) IN.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **A** | **B** | **IN** | **A \* B** | **A** | **(A B) \* IN** | **C** | **D** |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |

**Ejercicio 4.**

*Si sólo se poseen puertas lógicas NAND:*

**(a)** *¿Será posible obtener las funciones AND, OR y NOT?*

Sí, es posible obtener las funciones AND, OR y NOT si sólo se poseen puertas lógicas NAND.

**(b)** *¿Cómo se implementarían?*

AND: = = A \* B.

OR: = = = A + B.

NOT: = .