**Algebra de Boole:**

Compuertas lógicas, NOT AND y OR.

OPERACION NOT: Cambia de un nivel lógico al nivel lógico opuesto, cambia un 1 por un 0.Expresión booleana: si a la entrada tenemos la variable, a la salida tenemos, (A invertido o Negado, o bien, complemento de A).

|  |  |
| --- | --- |
| Entrada | Salida |
| 0 | 1 |
| 1 | 0 |

OPERACION AND: Realiza la multiplicación lógica, genera un nivel ALTo cuando todas las entradas están a nivel ALTO. Tendremos un 1 a la salida cuando todas las entradas estén en 1.

|  |  |  |
| --- | --- | --- |
| Entradas | | Salida |
| **A** | **B** | **X** |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

OPERACIÓN OR: Se conoce como suma lógica booleana. Se genera un nivel alto cuando una o más entradas están a nivel alto.

|  |  |  |
| --- | --- | --- |
| Entradas | | Salida |
| **A** | **B** | **X** |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

**Compuertas NAND, NOR, XOR:**

**NAND:** Se puede utilizar como una puerta universal, se pueden combinar para implementar las operaciones de las puertas AND, OR y del inversor.

**NOR:** Al igual que la NAND, se puede emplear como una puerta universal.

**OR-Exclusiva:** Es de alto nivel solo cuando dos entradas están a niveles lógicos opuestos. La salida XOR equivale a = AB+AB.

**Leyes del Algebra de BOOLE:**

Ley CONMUTATIVA: A+B = B+A AB=BA

Ley ASOCIATIVA: (A+B) + C=A (B+C) (AB) C = A (BC)

Ley DISTRIBUTIVA: A (B+C) = AB+ AC

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | A+0= A | 7 | A.A=A |
| 2 | A+1 =1 | 8 | A.A=0 |
| 3 | A.0=0 | 9 | A = A |
| 4 | A.1 = A | 10 | A+AB=A |
| 5 | A+A=A | 11 | A+AB=A+B |
| 6 | A+A=1 | 12 | (A+B)(A+C)=A+BC |

Demostración de la regla 10: A+AB = A

A+AB=A (1+B) Factor común

A+AB=A.1 Regla 2

A+AB=A Regla 4