

Basic Boolean Laws

Idempotent Law

$$A * A = A$$

$$A + A = A$$

Associative Law

$$(A * B) * C = A * (B * C)$$

$$(A + B) + C = A + (B + C)$$

Commutative Law

$$A * B = B * A$$

$$A + B = B + A$$

Distributive Law

$$A * (B + C) = A * B + A * C$$

$$A + (B * C) = (A + B) * (A + C)$$

Identity Law

$$A * 0 = 0 \quad A * 1 = A$$

$$A + 1 = 1 \quad A + 0 = A$$

Complement Law

$$A * \sim A = 0$$

$$A + \sim A = 1$$

Involution Law

$$\sim(\sim A) = A$$

DeMorgan's Law

$$\sim(A * B) = \sim A + \sim B$$

$$\sim(A + B) = \sim A * \sim B$$

Redundancy Laws

Absorption

$$A + (A * B) = A$$

$$A * (A + B) = A$$

$$(A * B) + (A * \sim B) = A$$

$$(A + B) * (A + \sim B) = A$$

$$A + (\sim A * B) = A + B$$

$$A * (\sim A + B) = A * B$$

And

a	b	out
1	1	1
0	1	0
1	0	0
0	0	0

Nand(a = a, b = b, out = c)

Nand(a = c, b = c, out = out)

Or

a	b	out
1	1	1
0	1	1
1	0	1
0	0	0

Nand(a = a, b = a, out = c)

Nand(a = b, b = b, out = d)

Nand(a = c, b = d, out = out)

Not

a	out
1	0
0	1

Nand(a = in, b = in, out = out)

Xor

//A · Not(B) + Not(A) · B

a	b	out
1	1	0
1	0	1
0	1	1
0	0	0

//A · Not(B)

Nand(a = b, b = b, out = Not-B)

Nand(a = a, b = Not-B, out = c1)

Nand(a = c1, b = c1, out = A-NotB)

//Not(A) · B

Nand(a = a, b = a, out = Not-A)

Nand(a = Not-A, b = b, out = c2)

Nand(a = c2, b = c2, out = NotA-B)

//A · Not(B) + Not(A) · B

Nand(a = A-NotB, b = A-NotB, out = c3)

Nand(a = NotA-B, b = NotA-B, out = c4)

Nand(a = c3, b = c4, out = out)

Multiplexor

//out = A · Not(S) + B · S

a	b	sel	out
0	0	0	0
0	1	0	0
1	0	0	1
1	1	0	1
0	0	1	0
0	1	1	1
1	0	1	0
1	1	1	1

Demultiplexor

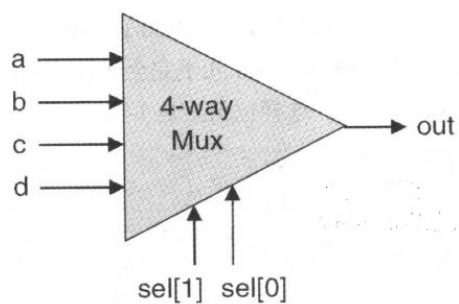
//a = in · Not(sel)

//b = in · sel

in	sel	a	b
0	0	0	0
0	1	0	0
1	0	1	0
1	1	0	1

Mux4Way16

sel[1]	sel[0]	out
0	0	a
0	1	b
1	0	c
1	1	d



DMux4Way

sel[1]	sel[0]	a	b	c	d
0	0	in	0	0	0
0	1	0	in	0	0
1	0	0	0	in	0
1	1	0	0	0	in

