**A) EXOR using NOR:**

**B) EXOR using NAND:**

**C) ENOR using NOR:**

**D) ENOR using NAND:**

**Even Parity Generator:** A ⊕ B ⊕ C

**Even Parity Checker:** (A ⊕ B) ⊕ (C ⊕ P)

**Odd Parity Generator:** A ⊕ B ⊕ C

**Odd Parity Checker :** ((A ⊕ B) ⊕ (C ⊕ P))’

**Binary to Grey Code:**

G3 = B3

G2 = B3 ⊕ B2

G1 = B2 ⊕ B1

G0 = B1 ⊕ B0

**Grey Code to Binary:**

B3 = G3

B2 = B3 ⊕ G2

B1 = B2 ⊕ G1

B0 = B1 ⊕ G0

**Half Adder: Sum =** A ⊕ B, **Carry =** AB

**Full Adder: Sum =** A ⊕ B ⊕ C, **Carry =** AB + AC + BC

**Full Adder Using Two Half Adder:** **Sum =** A ⊕ B ⊕ C, **Carry = AB + C(A** ⊕ B)  
Proving carry equation in Full Adder Using Two Half Adder Carry   
C = AB + C( A⊕B)

= AB + AB’C + A’BC

= A(B + B’C) + A’BC

= AB + AC + A’BC

= AB + C(A + A’B)

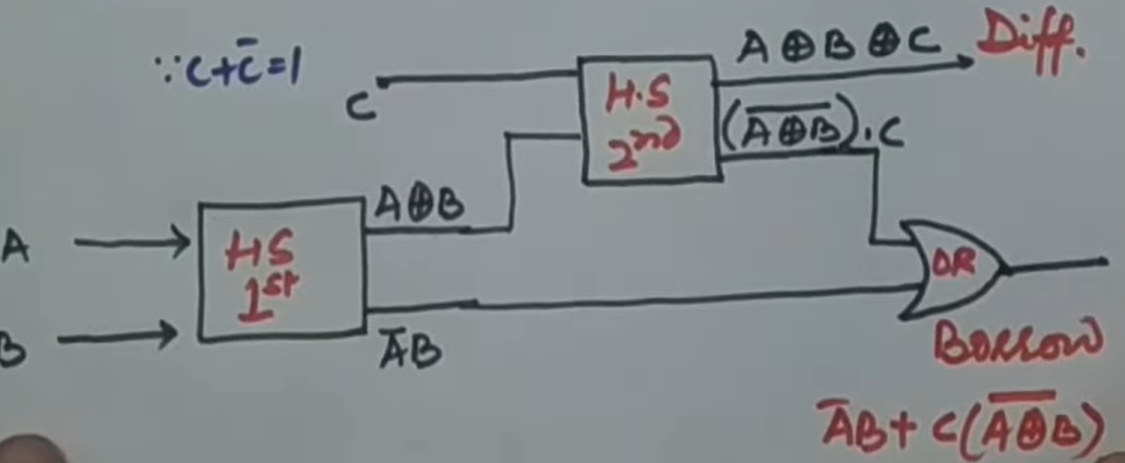
= AB + C(A + B)

= AB + AC + BC

**Half Subtractor:** **Diff =** A ⊕ B, **Carry =** A’B

**Full Subtractor:** **Diff =** A ⊕ B ⊕ C, **Carry =** A’B + A’C + BC

**Full Subtractor Using Two Half Subtractor:** **Diff =** A ⊕ B ⊕ C, **Carry =** A’B + C(A ⊕ B)’



Proving borrow equation: B = A’B + A’C + AB

= A’B(C + C’) + A’C(B + B’) + AB(C + C’)

= A’BC + A’BC’ + A’B’C + ABC

= A’B(C + C’) + C(A’B + AB)

= A’B + C(A ⊕ B)’

**One Bit Comparator:**

A>B = AB’ A<B = A’B A=B = A’B’ + AB

**Two Bit Comparator:**

A>B = A0B1’B0 + A1A0B0’+ A1B1

A<B = A1’A0’B0 + A1’B1 + A0’B1B0

A=B = (A1 ⊙ B1)(A0 ⊙ B0)

SR Flip Flop

D Flip Flop

JK Flip Flop

T Flip Flop