k= nand(a,b);

v\_and= nand(k,k); %AND via NAND

a\_bar = nand(a,a); %Inverter using NAND

b\_bar = nand(b,b);

x\_nand= nand(a\_bar,b\_bar); % OR via NAND

l=nor(a,b);

x\_or= nor(l,l); % OR via NOR

a\_bar = nor(a,a); %Inverter using NOR

b\_bar = nor(b,b);

x\_and= nor(a\_bar,b\_bar); % AND via NOR

FULL ADDER

X1 = xor(A,B); % Intermediate signal

S = xor(X1,Cin);

Y1= A & B; % Intermediate signal

Z1= X1 & Cin; % Intermediate signal

Cout = Z1 | Y1;

FULL SUBTRACTOR

X1 = xor(A,B); % Intermediate signal

S = xor(X1,Bin);

Y=~A & B;

Z= ~X1 & Bin;

Bout=Z | Y;

MUX