

Verilog

4-Value Logic

- **Four values:**

- 0, 1, x, z

- x: unknown value
- This x does NOT mean “don’t care”
- z: high impedance

input	not
0	1
1	0
x	x
z	x

and	0	1	x	z
0	0	0	0	0
1	0	1	x	x
x	0	x	x	x
z	0	x	x	x

or	0	1	x	z
0	0	1	x	x
1	1	1	1	1
x	x	1	x	x
z	x	1	x	x

Vectors

```
output [0:3] D;
```

```
wire [7:0] SUM;
```

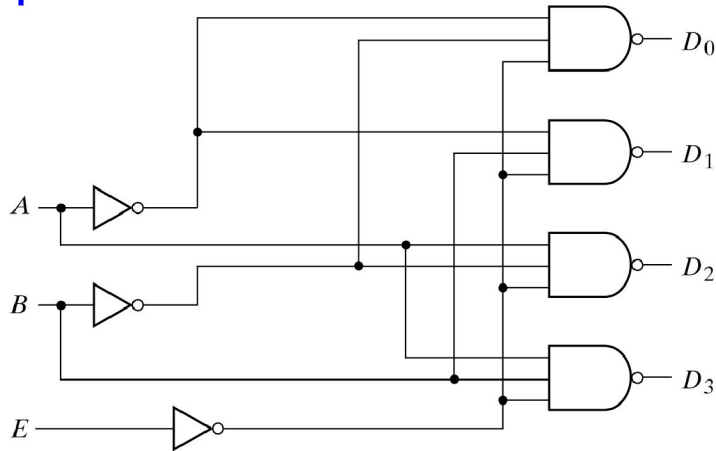
➤ **indexing:**

- D[3]

➤ **slicing:**

- SUM[5:2]

Decoder



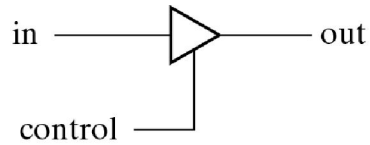
(a) Logic diagram

Fig. 4-19 2-to-4-Line Decoder with Enable

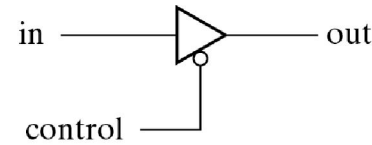
```
//HDL Example 4-1
//-----
//Gate-level description of a 2-to-4-line decoder
//Figure 4-19
module decoder_g1 (A,B,E,D);
    input A,B,E;
    output [0:3] D;
    wire Anot,Bnot,Enot;
    not
        n1 (Anot,A) ,
        n2 (Bnot,B) ,
        n3 (Enot,E);
    nand
        n4 (D[0],Anot,Bnot,Enot) ,
        n5 (D[1],Anot,B,Enot) ,
        n6 (D[2],A,Bnot,Enot) ,
        n7 (D[3],A,B,Enot);
endmodule
```

- Gate-level description
- NANDs and NOTs are instantiated in a group

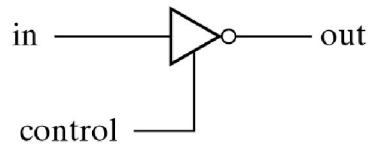
Tri-State Buffers and Inverters



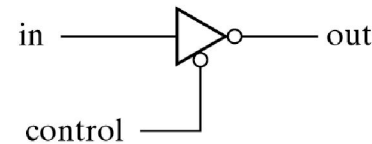
bufif1



bufif0



notif1



notif0

Fig. 4-31 Three-State Gates

gateName (output, input, control);

```
bufif1 (OUT, A, control);  
notif0 (Y, B, enable);
```

Multiplexer at Gate Level

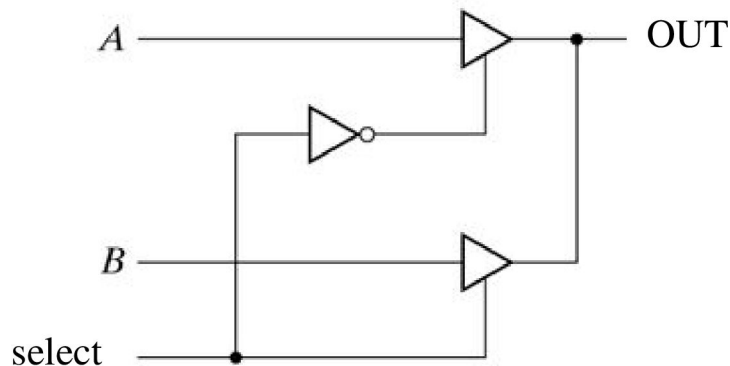
- **tri:**

- A net which has the capability of being driven by more than one driver

- **Types of nets:**

- wire, tri, supply1, supply0, wand (wired AND), wor (wired OR)

- Example:



(a) 2-to-1- line mux

```
module muxtri (A, B, select, OUT);  
  input A, B, select;  
  output OUT;  
  tri OUT;  
  bufif0 (OUT, A, select);  
  bufif1 (OUT, B, select);  
endmodule;
```