

Verilog Transistor Modeling

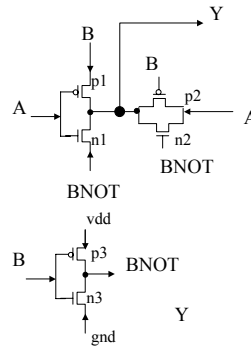
- Verilog has both unidirectional and bidirectional transistor models for switch-level CMOS modeling
- Unidirectional
 - *nmos*, *pmos* are unidirectional (three terminals)
 - *cmos* is a pass transistor – an *nmos*, *pmos* connected back to back (four terminals).
 - Strength reduction versions are *rnmos*, *rpmos*, *rcmos*
- Bidirectional, non-strength reducing
 - *tranif0*, *tranif1* are bidirectional (three terminals)
 - strength reducing versions are *rtranif0*, *rtranif1*

3/26/2002

BR

1

$$Y = A \text{ xor } B$$



module xorgate (Y, A, B);

input A, B;
output Y;
supply0 gnd;
supply1 vdd;

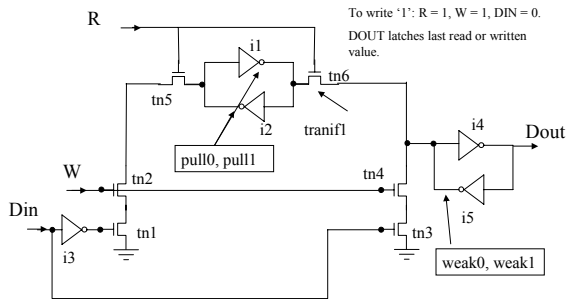
```
// Inverter for internal Bnot
pmos p3 (BNOT, vdd, B);
nmos n3 (BNOT, gnd, B);
pmos p1 (Y, B, A);
nmos n1 (Y, BNOT, A);
pmos p2 (Y, A, B);
nmos n2 (Y, A, BNOT);
endmodule
```

3/26/2002

BR

2

Ramcell



To read, R = 1, W = 0, DIN = X.
To write '0': R = 1, W = 1, DIN = 1.
To write '1': R = 1, W = 1, DIN = 0.
DOUT latches last read or written value.

3/26/2002

BR

3

Ramcell Comments

- Transistor tn6 needs to be bidirectional
 - during read operation, drive is from gate I2
 - during write operation, drive is from transistor tn4
- Drive strength of gates I1, I2 needs to be 'pull' because write operations must overpower these gates
- Drive strength of gate I5 must be weak because
 - needs to be overpowered by drive of gate I1 during read
 - needs to be overpowered by drive of transistor tn4 during write

3/26/2002

BR

4

module ramcell (dout, din, r, w);

input din, r, w;
output dout;
supply0 gnd;

Note: Cannot specify drive strength on transistors!

not i3 (dinnot, din);
nmos tn1 (n5, gnd, dinnot);
nmos tn2 (n1, n5, w);
nmos tn3 (n6, gnd, din);
nmos tn4 (n4, n6, w);

nmos tn5 (n2, n1, r);
tranif1 tn6 (n4, n3, r);

pull0, pull1

not (pull0, pull1) i1(n3, n2);
not (pull0, pull1) i2(n2, n3);

weak0, weak1

not i4(dout, n4);
not (weak0, weak1) i5(n4, dout);

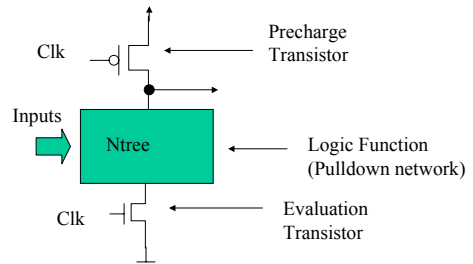
endmodule

3/26/2002

BR

5

Dynamic Gates



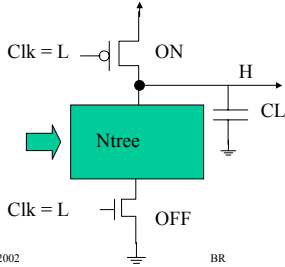
3/26/2002

BR

6

Operation of a Dynamic Gate

- Precharge phase: Clock Low, Precharge transistor on, output pull high. Evaluation transistor off.



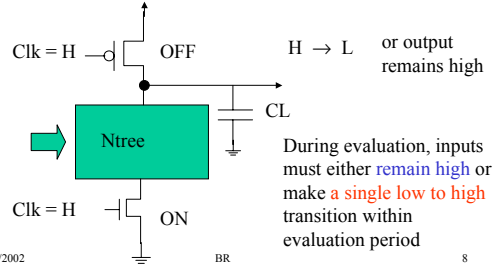
3/26/2002

BR

7

Operation of a Dynamic Gate (cont)

- Evaluation phase: Clock High, Precharge transistor off, Evaluation transistor on. Output is pulled low if there is a path in Ntree to ground.

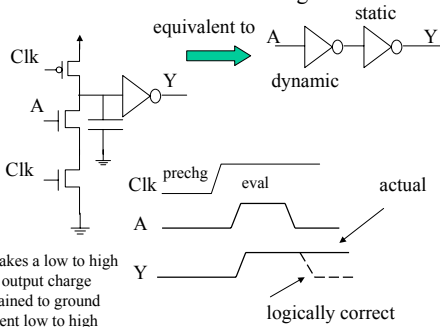


3/26/2002

BR

8

Why must gate input only make a single low to high transition or remain high?



Input A makes a low to high transition, output charge already drained to ground when A went low to high

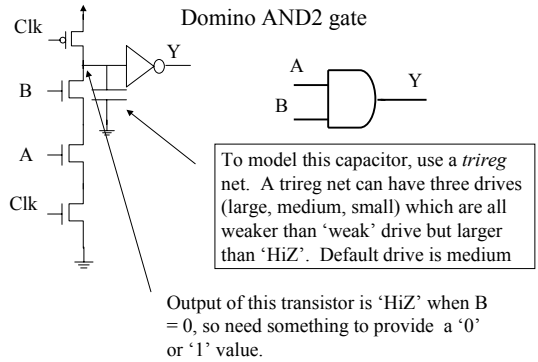
logically correct

3/26/2002

BR

9

Domino AND2 gate



To model this capacitor, use a trireg net. A trireg net can have three drives (large, medium, small) which are all weaker than 'weak' drive but larger than 'HiZ'. Default drive is medium

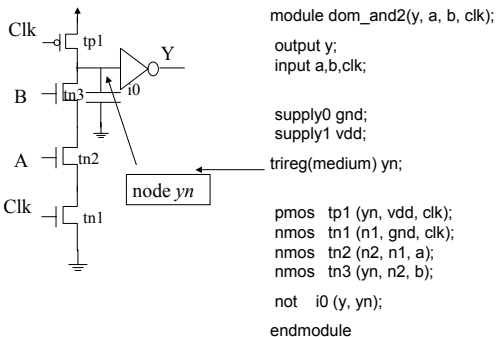
Output of this transistor is 'HiZ' when B = 0, so need something to provide a '0' or '1' value.

3/26/2002

BR

10

Domino AND2 gate (cont)

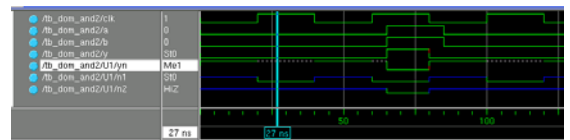


3/26/2002

BR

11

Simulation of Domino AND2 gate



drive strength of yn internal node is "me1"

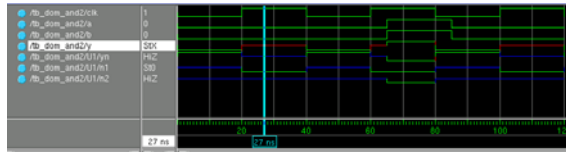
node yn gets pulled to ST0, so output is ST1

3/26/2002

BR

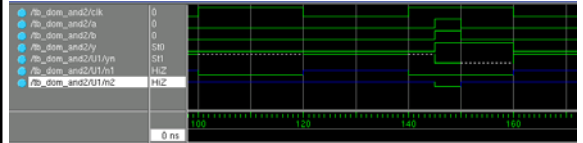
12

Simulation of Domino AND2 gate without *trireg* node



drive strength of yn internal node is now 'HiZ', which makes output a 'StX'.

Simulation of illegal transition on A, B inputs

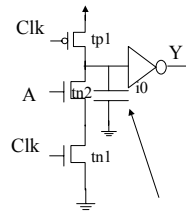


A, B go high, then drop low. Output goes to '1' when both 'A', 'B' = 1, but then internal charge is drained so output remains at '1' even when A, B return to '0'.

Subthreshold Leakage

- If the clock stays high for an extended period, the output charge will leak off due to subthreshold currents
- Dynamic circuits typically have a minimum clock frequency requirement because of this.
- Subthreshold leakage gets worse as you scale down in technology.
- A *trireg* net can have three delay values
 - (rise_delay, fall_delay, charge_decay_time)
 - charge_decay_time is time it takes for charge to decay to an 'X' value.

Domino Buffer with Charge Delay

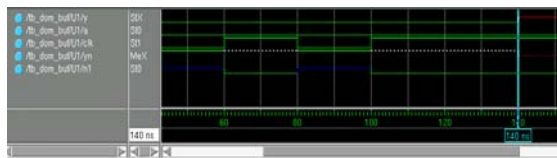


```

module dom_buf(y, a, clk);
    output y;
    input a, clk;
    supply0 gnd;
    supply1 vdd;
    trireg (medium) #(0,0, 40) yn;
    pmos tp1 (yn, vdd, clk);
    nmos tn1 (n1, gnd, clk);
    nmos tn2 (yn, n1, a);
    not i0 (y, yn);
endmodule
    
```

node y/n, charge delay of 40

Simulation with Charge Delay



From 100 ns, Clk remains high.
 After 40 ns, charge (Me1) decays to X value (MeX)
 Output goes to StX