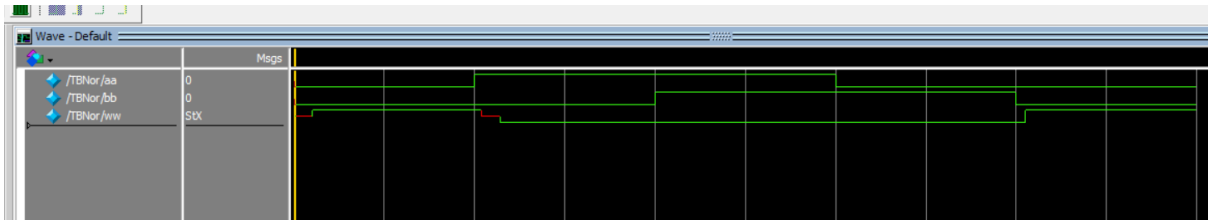
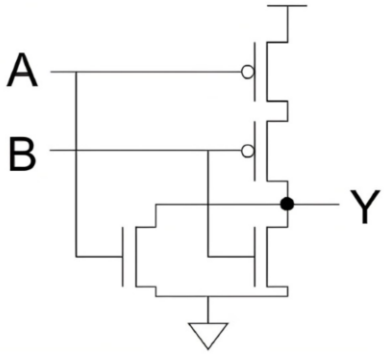
AmirAbbas MoumeniZadeh – SID:810101529 – CA1 Report – Digital System ECE 894

1) 2-Input Nor Implimentation





According to Calculation, worst-case dalay of this Circuit is 10ns, Which can be shown in the waveform diagram.

Calculation : pMOS Delay: (5,6,7) , nMOS Delay: (3,4,5)

$$TO1 : 2 \times (pMOS\ to1)\ , (nMOS\ to\ z) = 10\ , 5$$
$$\rightarrow WC\ Delay : 10$$

$$To0 :\ 2 \times (pMOS\ toZ)\ , (nMOS\ to\ 0) = 14\ , 4$$
$$\rightarrow WC\ Delay : 14$$

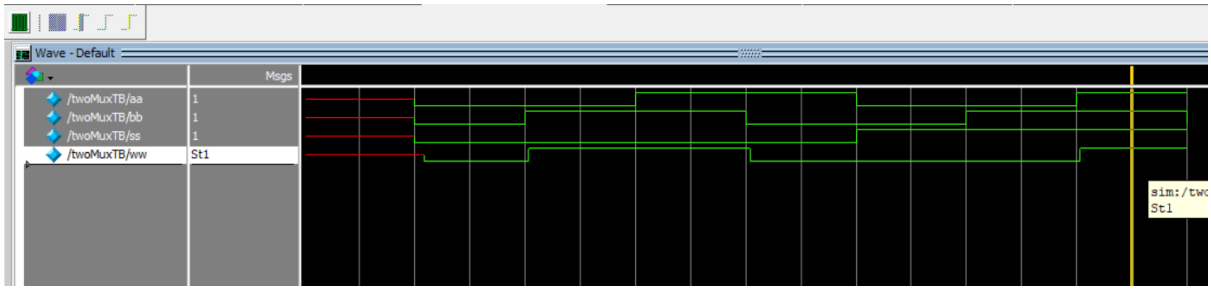
Implimentation Code :

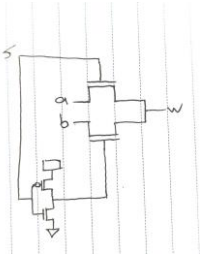
```
`timescale 1ns/1ns
module mynor(input a,b , output w);
wire j;
supply0 gnd;
supply1 vdd;
pmos #(5,6,7)(j,vdd,a);
pmos #(5,6,7)(w,j,b);
nmos #(3,4,5)(w,gnd,a);
nmos #(3,4,5)(w,gnd,b);
endmodule
```

Testbench Code :

```
`timescale 1ns/1ns
module TBNor();
reg aa,bb;
wire ww;
mynor UUT(.a(aa),.b(bb),.w(ww));
initial begin
aa=0; bb=0;
#100 aa=1 ; bb=0;
#100 aa=1 ; bb=1;
#100 aa=0 ; bb=1;
#100 aa=0 ; bb=0;
#100 $stop;
end
endmodule
```

2) 2-to-1 Mux implimentation:





We can see through the waveform that in first half of wave form which *S* is 0, output just looks at *a* and follows it and in the second half the opposite. Which means when *S* is 1, output follows *b*.

```
`timescale 1ns/1ns
module twoOneMUX(input a,b,s , output w);
wire j;
supply0 gnd;
supply1 vdd;
nmos #(3,4,5)(w,a,s);
pmos #(5,6,7)(j,vdd,s);
nmos #(3,4,5)(j,gnd,s);
nmos #(3,4,5)(w,b,j);
endmodule
```

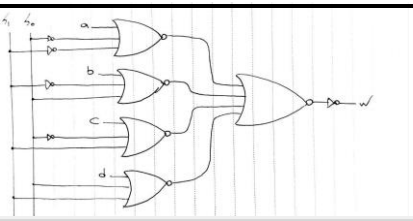
$$worst\ case\ delay:$$

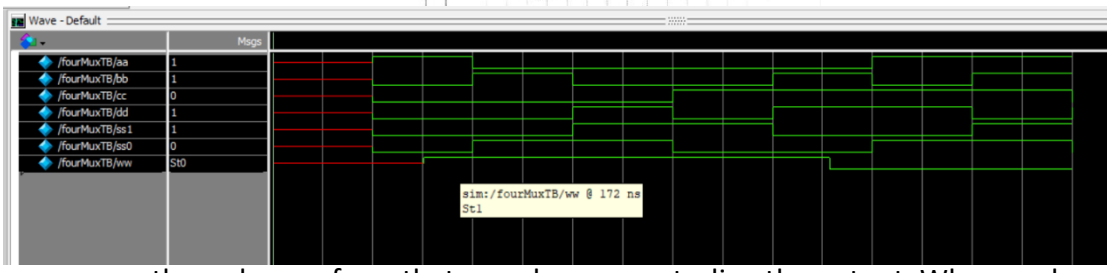
$$to1: nMOS\ toZ + pMOS\ toZ = 5 + 7 = 12$$

$$to0: nMOS\ toZ + pMOS\ toZ = 5 + 7 = 12$$

```
`timescale 1ns/1ns
module twoMuxTB();
reg aa,bb,ss;
wire ww;
twoOneMUX UUT1(.a(aa),.b(bb),.s(ss),.w(ww));
initial begin
#100 ss=0; aa=0; bb=0;
#100 ss=0; aa=0; bb=1;
#100 ss=0; aa=1; bb=1;
#100 ss=0; aa=1; bb=0;
#100 ss=1; aa=0; bb=0;
#100 ss=1; aa=0; bb=1;
#100 ss=1; aa=1; bb=1;
#100 ss=1; aa=1; bb=0;
end
endmodule
```

3) 4-to-1 Mux





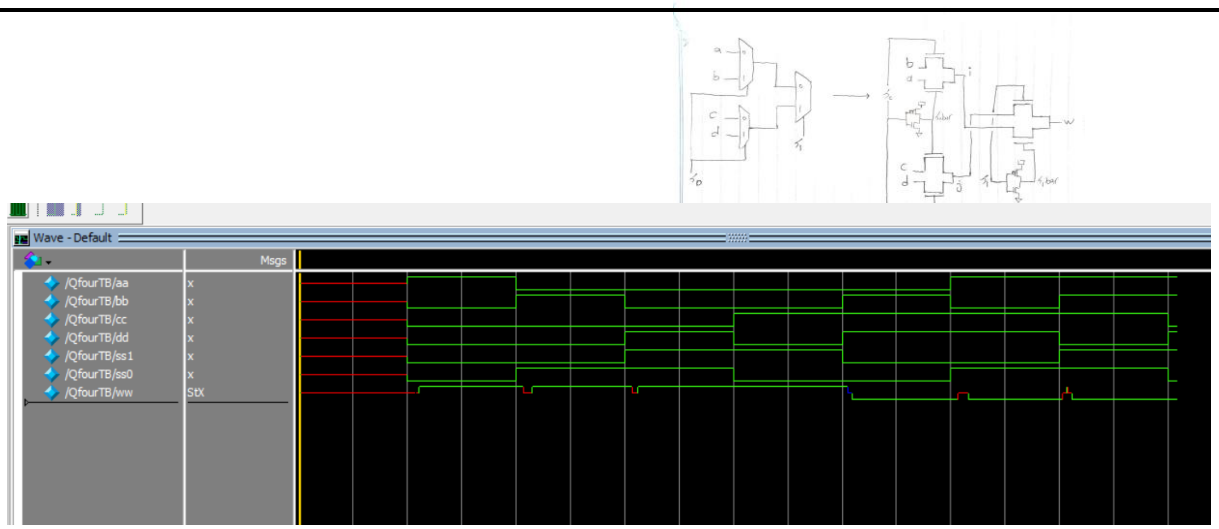
we can see through waveform that s_1 and s_0 are controling the output. When we have 00 on control we see *a* on output. 01 for *b*, 10 for *c* and 11 for *d*. (the testbench is made in a situation that in every step the value that we expect to have it on output is different from the others so easily can be undrestood that circuit Is working correct or not).

```
`timescale 1ns/1ns

module fourOneMUX(input a,b,c,d,s1,s0 ,
output w);

wire i,j,k,l,p;

nor #(15,21)(i,~s1,~s0,a);

nor #(15,21)(j,~s1,s0,b);

nor #(15,21)(k,s1,~s0,c);

nor #(15,21)(l,s1,s0,d);

nor #(20,28)(p,i,j,k,l);

nor #(10,14)(w,p,p);

endmodule
```

$Worst\ Case\ Delay: 15 + 20 + 2 \times 10 = 55$

Which makes sence bacause we have a 3-input and a four input Nor and two invertors which have been made by conecting Nors inputs together SO we expect to have a lot of Timing delay

```
`timescale 1ns/1ns

module fourMuxTB();

reg aa,bb,cc,dd,ss1,ss0;

wire ww;

fourOneMUX
UUT1(.a(aa),.b(bb),.c(cc),.d(dd),.s1(ss1),.s0(ss0),.w(ww));

initial begin

#100 ss1=0; ss0=0; aa=1; bb=0; cc=0; dd=0;

#100 ss1=0; ss0=1; aa=0; bb=1; cc=0; dd=0;

#100 ss1=1; ss0=1; aa=0; bb=0; cc=0; dd=1;

#100 ss1=1; ss0=0; aa=0; bb=0; cc=1; dd=0;

#100 ss1=0; ss0=0; aa=0; bb=1; cc=1; dd=1;

#100 ss1=0; ss0=1; aa=1; bb=0; cc=1; dd=1;

#100 ss1=1; ss0=1; aa=1; bb=1; cc=1; dd=0;

#100 ss1=1; ss0=0; aa=1; bb=1; cc=0; dd=1;

end

endmodule
```



4) 4-to-1 Mux Implimenting with 2-to-1 Mux

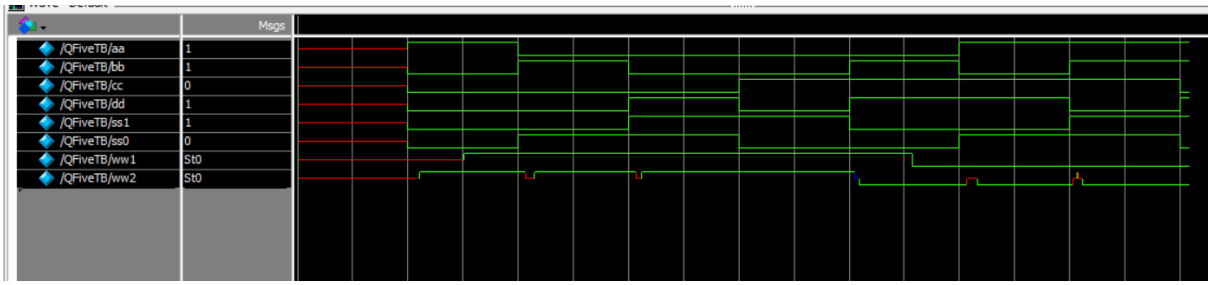
Test bench is just like the previous question. we can see through waveform that s_1 and s_0 are controling the output. When we have *00* on control we see *a* on output. *01* for *b*, 10 for *c* and *11* for *d*.

We can see that there is a lot of glitch on output comparing to the previous question but the timing delay is lower than previous one. We can examine this in next question in next question more accurate with seeing both outputs together.

```
`timescale 1ns/1ns

module QuesFourMUX(input a,b,c,d,s1,s0 , output w);

wire i,j,s0bar , s1bar;

supply0 gnd;

supply1 vdd;

//Mux1 a,b

nmos #(3,4,5)(i,b,s0);

nmos #(3,4,5)(i,a,s0bar);

//s0 inverter

nmos #(3,4,5)(s0bar,gnd,s0);

pmos #(5,6,7)(s0bar,vdd,s0);

//Mux2 c,d

nmos #(3,4,5)(j,c,s0bar);

nmos #(3,4,5)(j,d,s0);

//Mux3

nmos #(3,4,5)(w,j,s1);

nmos #(3,4,5)(w,i,s1bar);

//s1 inverter

nmos #(3,4,5)(s1bar,gnd,s1);

pmos #(5,6,7)(s1bar,vdd,s1);

endmodule
```

```
`timescale 1ns/1ns

module fourMuxTB();

reg aa,bb,cc,dd,ss1,ss0;

wire ww;

fourOneMUX UUT1(.a(aa),.b(bb),.c(cc),.d(dd),.s1(ss1),.s0(ss0),.w(ww));

initial begin

#100 ss1=0; ss0=0; aa=1; bb=0; cc=0; dd=0;

#100 ss1=0; ss0=1; aa=0; bb=1; cc=0; dd=0;

#100 ss1=1; ss0=1; aa=0; bb=0; cc=0; dd=1;

#100 ss1=1; ss0=0; aa=0; bb=0; cc=1; dd=0;


#100 ss1=0; ss0=0; aa=0; bb=1; cc=1; dd=1;

#100 ss1=0; ss0=1; aa=1; bb=0; cc=1; dd=1;

#100 ss1=1; ss0=1; aa=1; bb=1; cc=1; dd=0;

#100 ss1=1; ss0=0; aa=1; bb=1; cc=0; dd=1;

end

endmodule
```

5) Examine both 4-to-1 Muxes together



ww1 is output for Mux made with Nor Gates and *ww2* is output for 4-1Mux made with 3 2-1 Muxes. As we can see delay for *ww1* is much more than *ww2* which is bacause in first circuit there are much more transistors in use ($\cong 32\ switches$) but in second circuit this number is much lower ($10\ Switches$)

Hand drawn waveform for Problem 1.