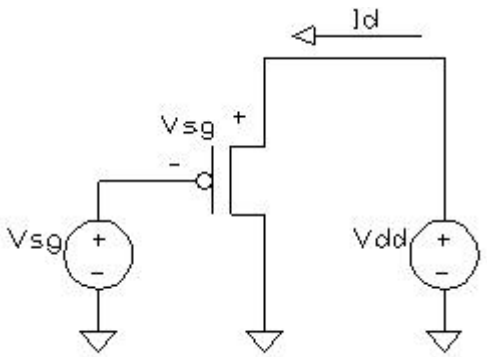
Problem 14.1 **Justin Wood**

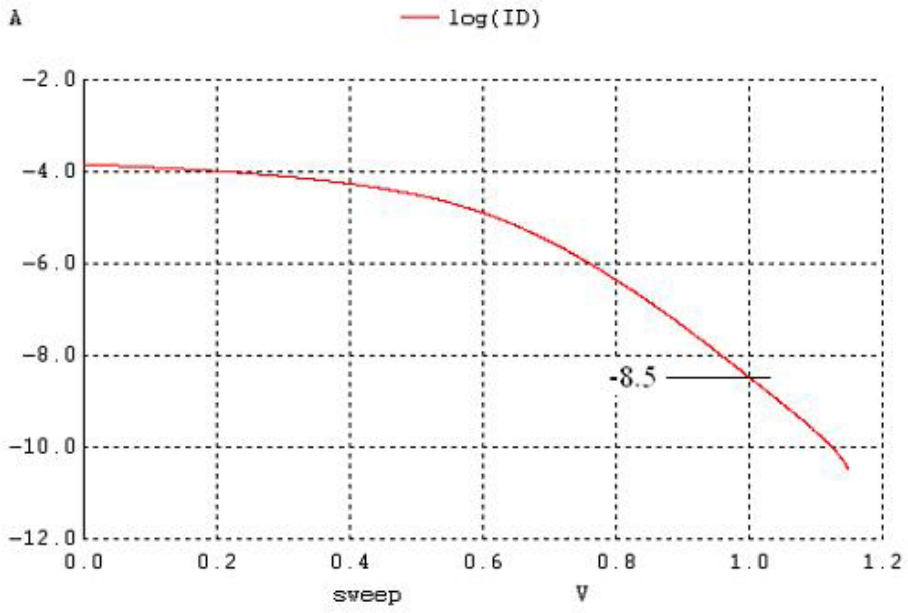
Regenerate Fig. 14.2 for the PMOS device. From the results determine the PMOS's I_{off}.

Solution: The following circuit was implemented in spice:



The netlist and plot are shown below.

```
.control
destroy all
run
let ID=-i(vdd)
plot log(ID)
.endc

.option scale=50n
.dc     vsg     0 1.15 1m

vdd     vdd     0       DC      1
vsg     vsg     0       DC      0

M1      vdd     vsg     0       vdd     PMOS L=1 W=10
```



To estimate I_{off} From the graph:

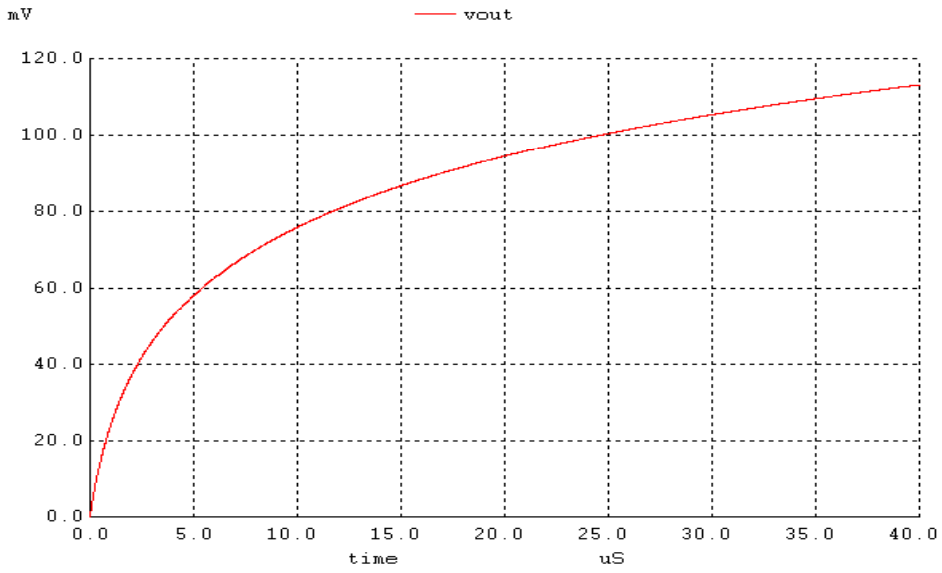
Log I_D = –8.5
$I_D = 10^{-8.5}$A
I_D=3.16nA

$I_D=I_{off}$ * W * Scale
$I_{off}=I_D/(W*Scale)$
I_{off}=3.16nA/(10*0.05μm
I_{off}=6.32nA/ μm

KRISHNAMRAJU KURRA

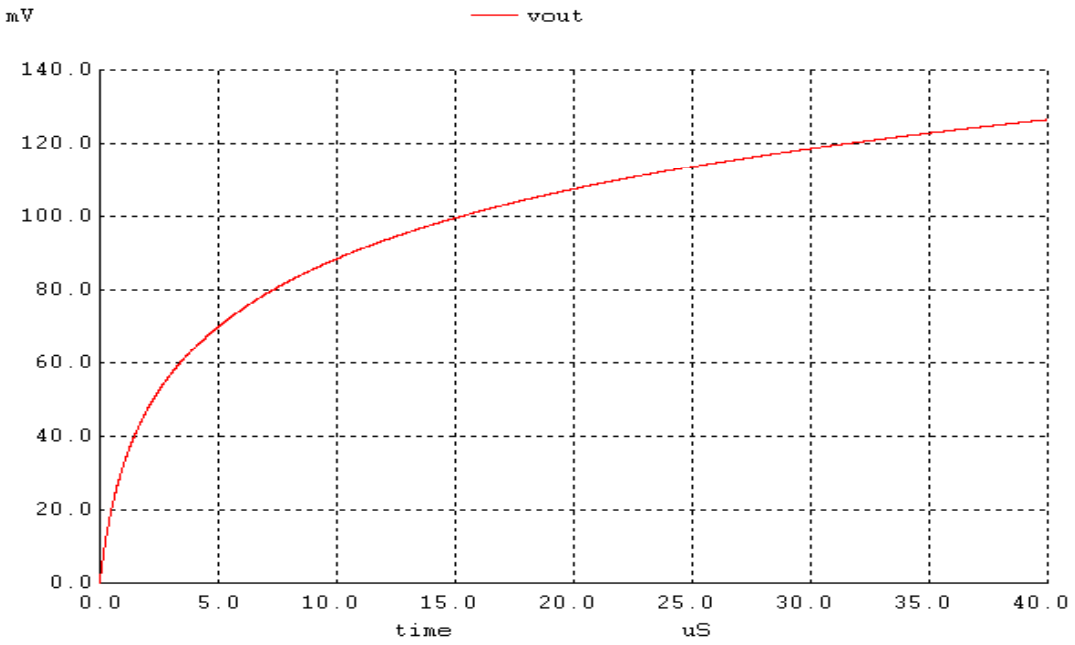
Problem: 14.2:
Repeat Ex. 14.2 if the storage node is a logic 0 (ground). Explain why the charge storage node charges up. What would happen if the PG's input were held at VDD instead of VDD/2.

```
*** Q:14.2 ***
.control
destroy all
run
plot vout
.endc
.option scale=50n
.tran 10n 40u UIC
vddby2          vddby2          0       DC      500m
M1      vout1   0       vddby2          0       NMOS L=1 W=10
vmeas   vout1   vout    DC      0
Cl      vout    0       50f     IC=0

.model  nmos  nmos  level = 14
.end
```



The voltage between the drain and source of the NMOS is equal to VDD/2. This results in flow of off current from the drain to source across the channel and the node is charge.
Here we can see that at 40usec the node charged up to ~110mV. (If we increase the off time or the simulation time (to 2msec) we can see that the node charges to ~185mv.)

Increasing the input voltage to VDD results in the following plot.



The node charged to 126mV in this case.
(If we increase the off time or the simulation time (to 2msec) we can see that the node charges to nearly 196mv.)

Problem 14.3 **Krishna Duvvada**

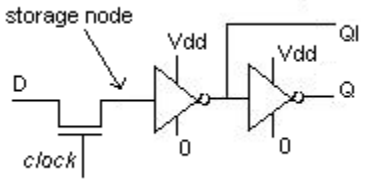
Comment on the usefulness of dynamic logic in our 50 nm CMOS process based on the results given in Ex. 14.3 with a clock frequency of 10MHz.

Solution: - From the example 14.3, for a clock frequency of 10MHz, the voltage at the storage node drops to 0.7V. At this voltage, there is a chance that the NMOS and PMOS in the inverter to turn on which is not desirable because of invalid logic. 50nm process is not useful for dynamic logic due to larger off current and smaller parasitic capacitance and it has to refresh very fast and hence the refresh rate is also faster.

14.4) **Surendranath C Eruvuru**

Using the circuit in Fig. 14.6 and the SPICE simulation show how the current drawn from VDD, by the inverters, changes with time. Do you see any concerns? If so, what?

Following is the schematic from fig. 14.6.



In the above circuit, NMOS turns on when the clock is high and the output follows D input. In this circuit, the current is drawn only when the inverters are in transition period. Assuming clock is high and D value is low, PMOS of the first inverters turns on and the NMOS of the second transistor turns on. The output capacitance of the first inverter and the input capacitance of the second inverter are charged through R_p of the first inverter. Simultaneously, R_n of the second inverter discharges the output capacitance of the second inverter. If clock is high and D value is high, then the NMOS of first inverter discharges the output capacitance of the first inverter and the input capacitance of the second inverter. Simultaneously, PMOS of the second inverter charges the output capacitance of the second inverter.

The current drawn from the VDD is only in the transition period. In transition period both NMOSs and PMOSs in both the inverters turn on. So, there is current flow directly from VDD to ground. If the simulation is observed, the current flow is only in the transition period.

Concerns: First most concern will be the amount of current flowing in transition period because this will decide the DC power consumption in this circuit. In the simulation we can see that more than 200μA of current is consumed when PMOS of the first inverter turns on.

Netlist:

```
.control
destroy all
run
plot V(CLK) V(D)+1.25 Vin+2.5
plot vdd#branch ylimit -250e-6 20e-6
.endc

.options scale=50n
.tran 20p 30n UIC
```

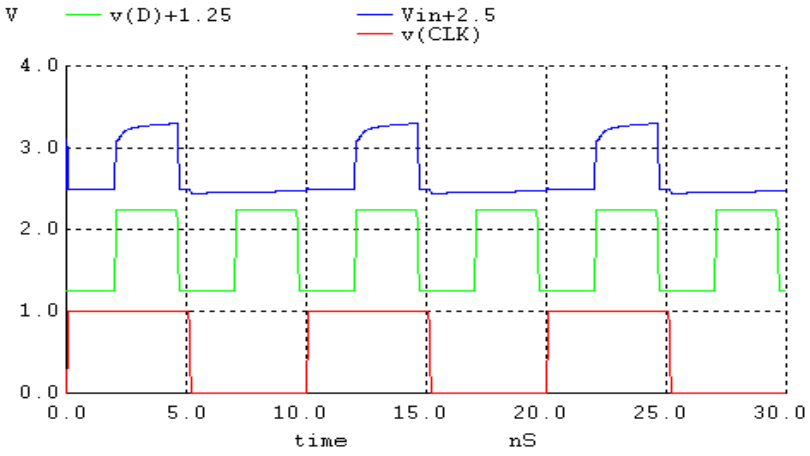
VCLK CLK 0 DC 0 pulse 0 1 0 100p 100p 5n 10n
VD D 0 DC 0 pulse 0 1 2n 100p 100p 2.5n 5n
VDD Vdd 0 DC 1

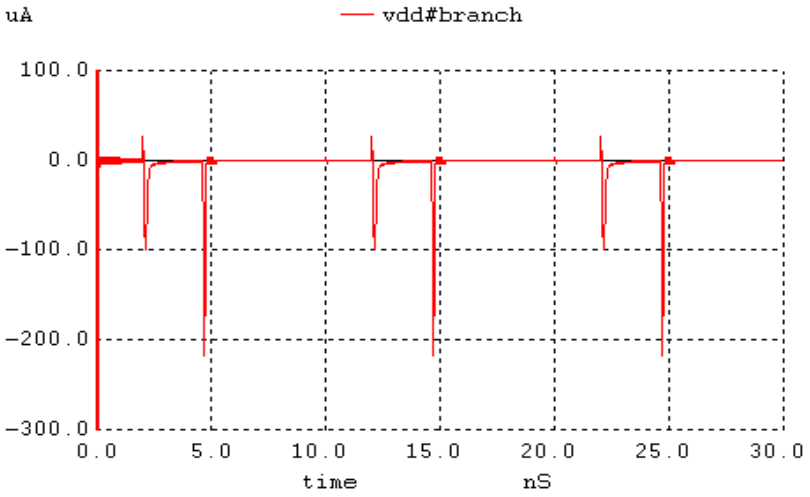
M5 vin CLK D 0 NMOS L=1 W=10

M1 QI Vin Vdd Vdd PMOS L=1 W=20
M2 QI Vin 0 0 NMOS L=1 W=10

M3 Q QI Vdd Vdd PMOS L=1 W=20
M4 Q QI 0 0 NMOS L=1 W=10

Simulation results:





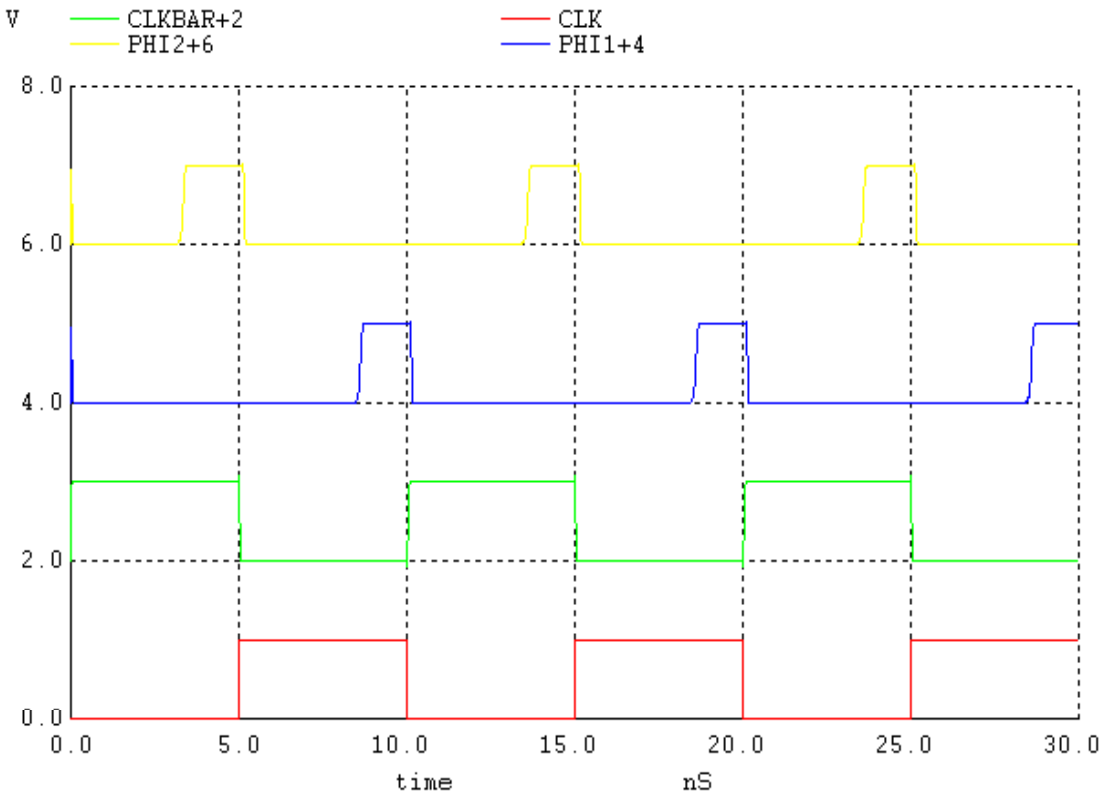
Problem 14.5 **Satish Dulam**

Simulate the operation of the non-overlapping clock generator circuit in Fig14.9. Assume that the input clock signal is running at 100MHz. Show how both PHI1 and PHI2 are non-overlapping.

Solution:
Spice simulation of fig 14.9:

```
*Problem 14.5
.control
destroy all
run
plot CLK CLKBAR+2 PHI1+4 PHI2+6
.endc

.option scale=50n
.tran 10p 30n 0n UIC

vdd       vdd  0    DC   1
VCLK      CLK  0    DC   PULSE 0 1 5n 0 0 5n 10n

XINV      CLK       CLKBAR    vdd             INV
XUPNAND1  CLK       downtemp3 uptemp1   Vdd  NAND
XUPINV1   uptemp1   uptemp2   Vdd             INV
XUPINV2   uptemp2   uptemp3   Vdd             INV
XUPINV3   uptemp3   PHI1      Vdd             INV
XDOWNNAND2     CLKBAR    uptemp3   downtemp1 Vdd  NAND
XDOWNINV4 downtemp1 downtemp2 Vdd             INV
XDOWNINV2 downtemp2 downtemp3 Vdd             INV
XDOWNINV3 downtemp3 PHI2      Vdd             INV

CLoad          Q    0    50f  IC=0

.subckt INV    INV_IN    INV_OUT   VDD
MINV1     INV_OUT   INV_IN    VDD  VDD  PMOS L=1 W=20
MINV2     INV_OUT   INV_IN    0    0    NMOS L=1 W=10
.ends

.subckt NAND NAND_INA NAND_INB NAND_OUT VDD
MNAND1    NAND_OUT  NAND_INA  VDD  VDD  PMOS L=1 W=10
MNAND2    NAND_OUT  NAND_INB  VDD  VDD  PMOS L=1 W=10
MNAND3    NAND_OUT  NAN_INA         Ntemp     0    NMOS L=1 W=10
MNAND4    Ntemp          NAND_INB  0    0    NMOS L=1 W=10
.ends
```

```
* 50nm BSIM4 models
* Don't forget the .options scale=50nm if using an Lmin of 1
* 1<Ldrawn<200   10<Wdrawn<10000 Vdd=1V
* Change to level=54 when using HSPICE
```



Vinay Dindi

14.6) Simulate the operation of clocked CMOS latch shown in Fig. 14.11.

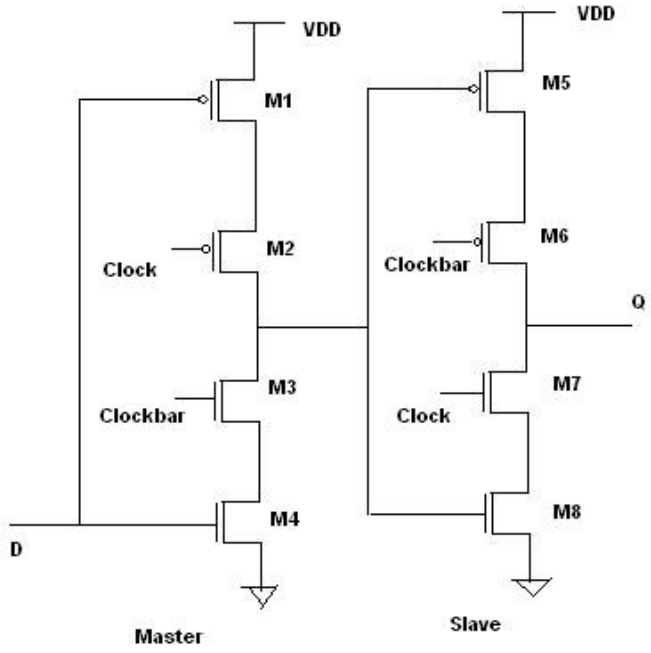
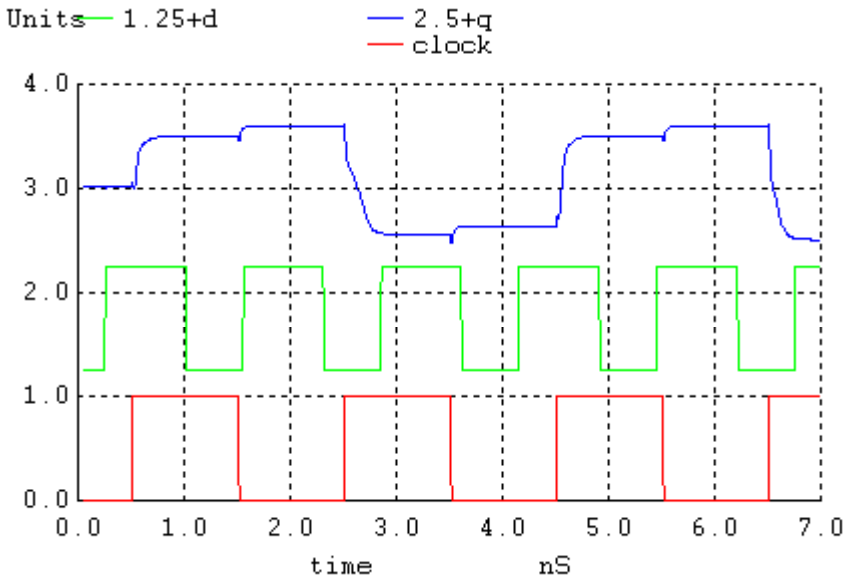


Fig 14.11

The netlist for the above circuit is

```
x1      clock   clockb vdd     INV
M1      v1      d      vdd     vdd     PMOS L=1 W=20
M2      v1out   clock  v1      vdd     PMOS L=1 W=20
M3      v1out   clockb v2      0       NMOS L=1 W=10
M4      v2      d      0       0       NMOS L=1 W=10
c1      v1out   0      10f
M5      v3      v1out  vdd     vdd     PMOS L=1 W=20
M6      q       clockb v3      vdd     PMOS L=1 W=20
M7      q       clock  v4      0       NMOS L=1 W=10
M8      v4      v1out  0       0       NMOS L=1 W=10
c2      v1out   0      10f

.subckt INV     A      out     VDD
M1      out     A      0       0       NMOS L=1 W=10
M2      out     A      VDD     VDD     PMOS L=1 W=20
.ends
```

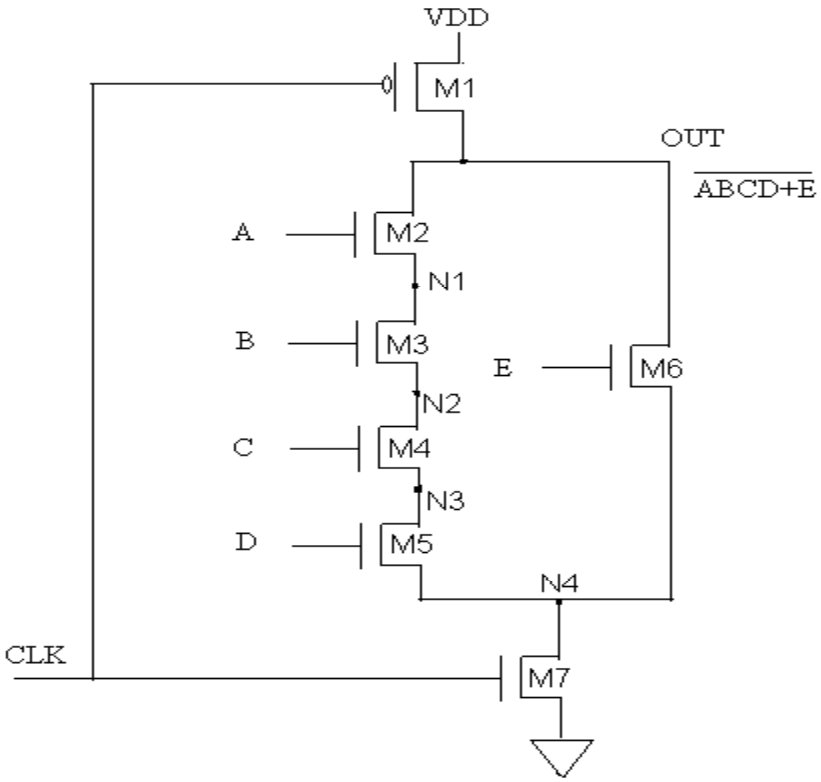
The simulation output for figure 14.11 is as follows



As we can see from the output waveform, the clock samples the input when it is low and transfers the data to the output "q" when the clock goes high. This behavior is similar to the edge triggered D-FF.

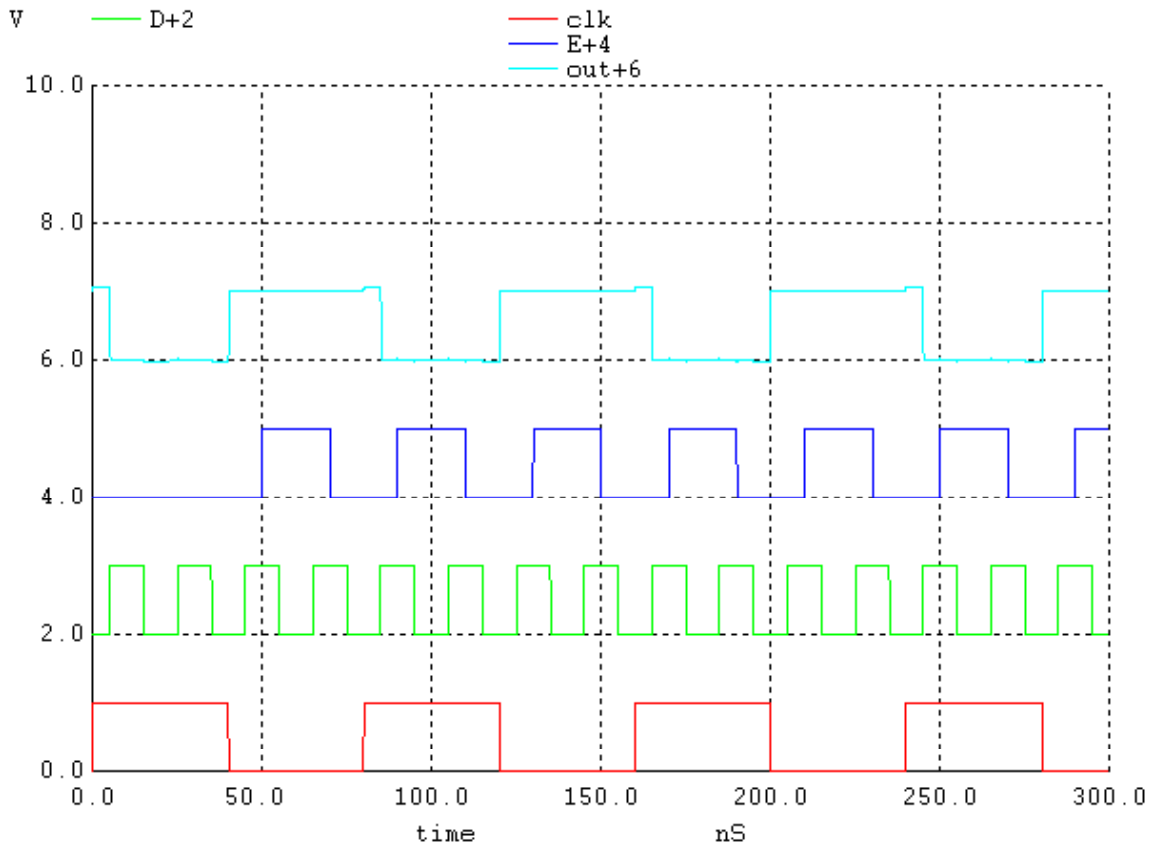
Problem 14.7:- ***Design and simulate the operation of a PE gate that will implement the logical function*** $F = \overline{A.B.C.D + E}$.

Solution:- Circuit for the PE logic $F = \overline{A.B.C.D + E}$




```
*** netlist for the circuit ***
.control
destroy all
run
plot clk D+2 E+4 out+6  ylimit 0 10
.endc

.option scale=50n
.tran 100p 300n

vdd     vdd     0       DC      1
Vclk    clk     0       DC      0 pulse 0 1 0 0 0 40n 80n

VA      A       0       DC      1
VB      B       0       DC      1
VC      C       0       DC      1

VD      D       0       DC      0 pulse 0 1 5n 0 0 10n 20n
```

```
VE      E       0       DC      0 pulse 0 1 50n 0 0 20n 40n

M1      out  CLk   vdd   vdd    PMOS L=1 W=20
M2      out  A     N1    0      NMOS L=1 W=10
M3      n1   B     n2    0      NMOS L=1 W=10
M4      n2   C     n3    0      NMOS L=1 W=20
M5      n3   D     n4    0      NMOS L=1 W=10
M6      out  E     n4    0      NMOS L=1 W=10
M7      n4   clk   0     0      NMOS L=1 W=10
*50nm BSIM4 models should be added
```



When the Clk is zero, the output is pulled to VDD by the PMOS.
When the Clk is one, the output is evaluated by the above logic. Once the output goes zero, it stays there until the clk goes zero again.

The circuit can also be simulated by interchanging the pulses of the voltages A, B, C, D and E in the netlist.

Problem 14.8 **RAHUL MHATRE**

If the PE gate shown in Fig. 14.9 drives a 50fF capacitor, estimate the worst-case t_{phl}.
Use a 20/1 PMOS and a 10/1 NMOS.

Solution:

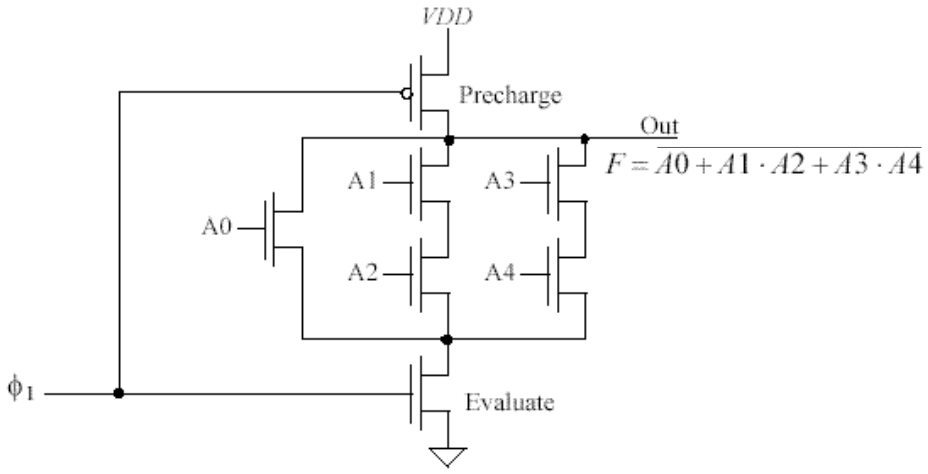


Figure 14.9 A complex PE gate.

For worst case $t_{phl,}$ when ϕ or ϕ_{bar}, go from low to low, the propogattion delay comprises of 3 10/1 NMOS transistors in the bottom path. The equivalent model when ϕ or ϕ_{bar} go from low to high is as shown below.

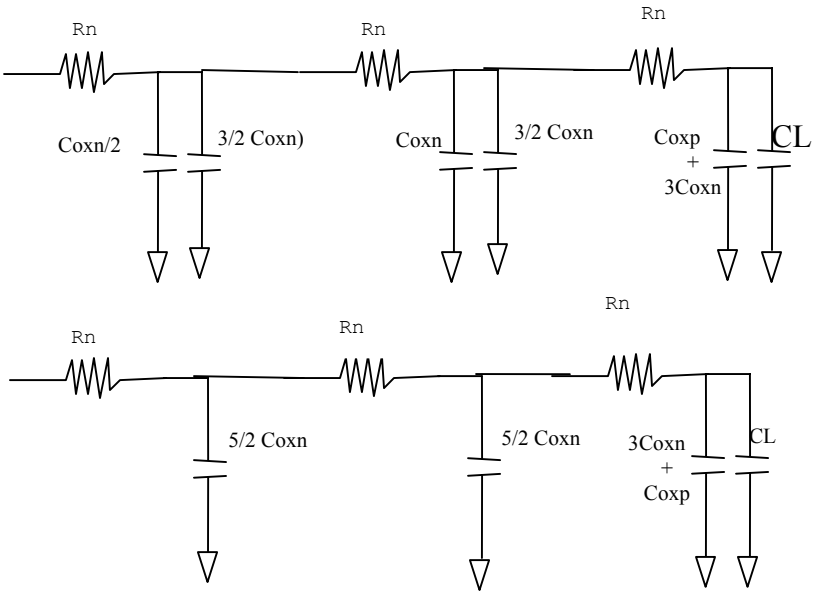


Figure 14.8.1 Circuit model for the worst-case t_{phl}.

Again at the output, we have the capacitance loading of the PMOS and other 2 NMOS transistors that are not conducting for the worst case (only A3=A4=1 when clock goes high)

t_{phl} = 0.7 * [Rn * (5/2 Coxn) + 2(5/2 * Coxn)*2Rn + 3Rn*(Coxp+3Coxn+C_L)]

For a 50nm process with a 20/1 PMOS and 10/1 NMOS, the above parameters are as given below.
Rn=3.4K
Coxp=1.25fF
Coxn=0.625aF

Load capacitance is given as 50fF.

Plugging in these values, we get
t_{phl} =137.59 pS.

The SPICE simulation is as given below. From simulations, the delay is found to be around **500pS**.

*****Problem 14.8 CMOS Circuit Design, Layout and Simulation*****

```
.control
destroy all
run
plot clk A3+1.5 A4+3 out+4.5
.endc

.option scale=50n
.tran 5p 10n 0n UIC

vdd         vdd         0       DC    1
VCLK        clk         0       DC    PULSE 0 1 1n 0 0 1n 2n
VA1         A1          0       DC    0
VA0         A0          0       DC    0
VA2         A2          0       DC    0
VA3         A3          0       DC    PULSE 0     1 1.2n 0 0 1.9n 3.9n
VA4         A4          0       DC    PULSE 0     1 1.2n 0 0 1.9n 3.9n

M1    n1    clk   0     0     NMOS L=1 W=10
M2    n3    A4    n1    0     NMOS L=1 W=10
M3    out   A3    n3    0     NMOS L=1 W=10
M4    n2    A2    n1    0     NMOS L=1 W=10
M5    out   A1    n1    0     NMOS L=1 W=10
M6    out   A0    n1    0     NMOS L=1 W=10
M7    out   clk   vdd   vdd   PMOS L=1 W=20
```

```
Cout    out      0        50f      IC=0

* 50nm BSIM4 models
*
* Don't forget the .options scale=50nm if using an Lmin of 1
* 1<Ldrawn<200   10<Wdrawn<10000 Vdd=1V
* Change to level=54 when using HSPICE
.end
```

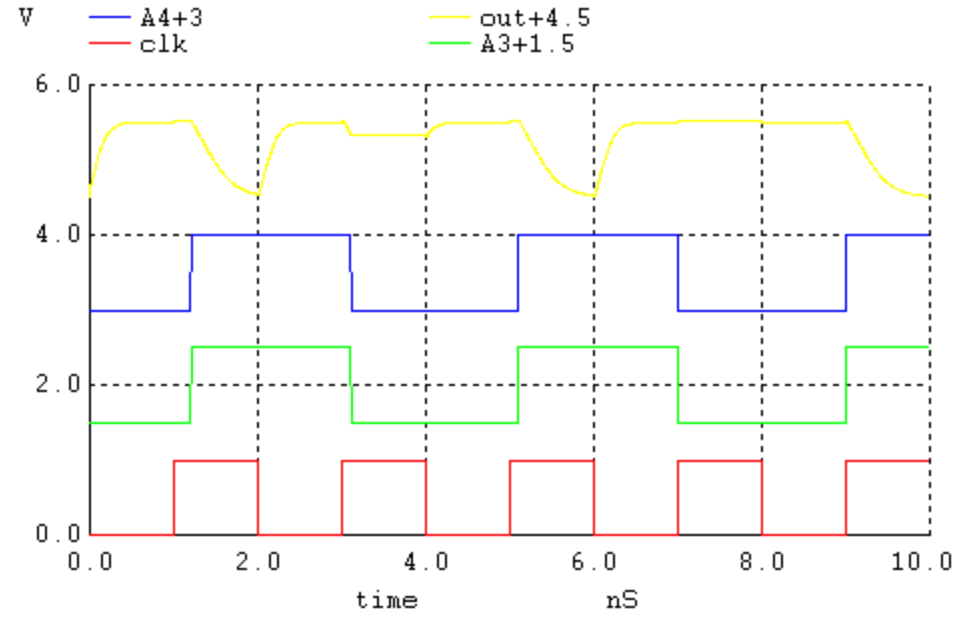
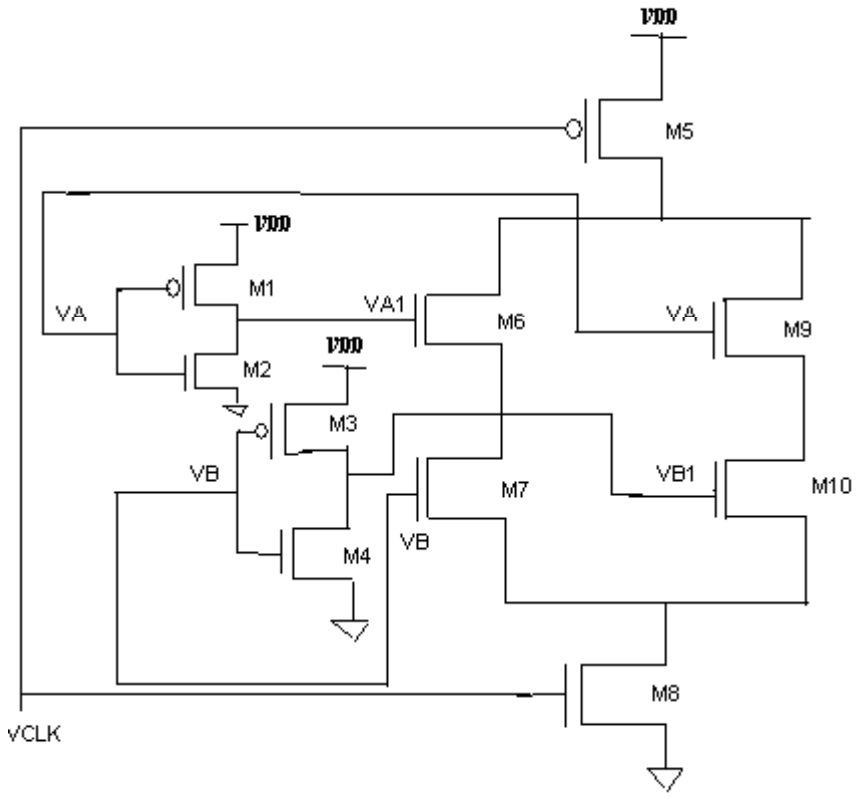


Figure 14.8.2 SPICE simulation for estimating the worst-case t_{phl}.

PROBLEM 14.9 **Indira**

Implement an XOR gate using Domino logic. Simulate the operation of the resulting implementation?

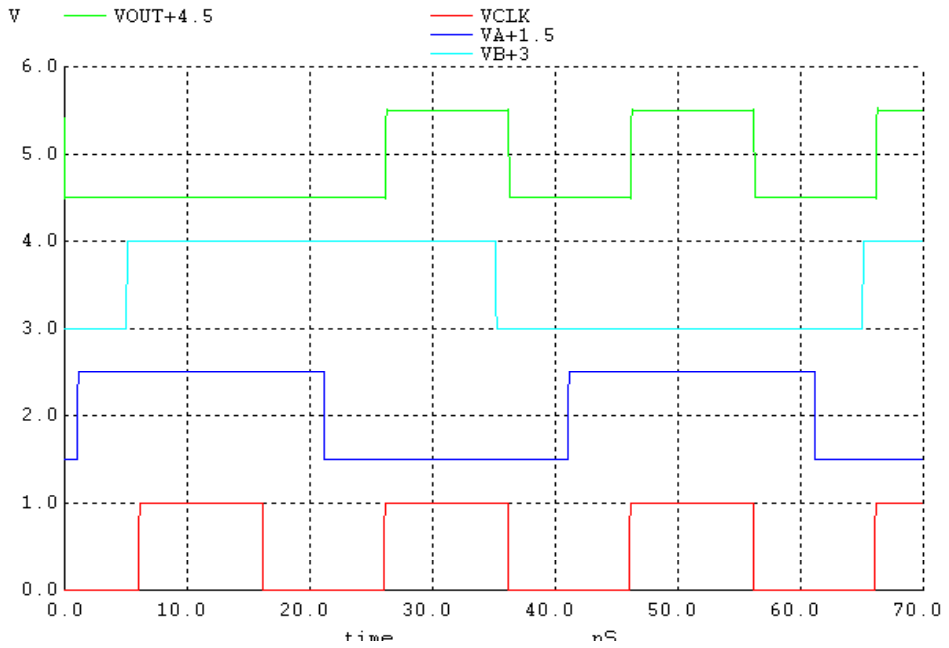
Solution:



Truth Table:

VA	VB	VOUT
0	0	0
0	1	1
1	0	1
1	1	0

```
***problem 14.9******
.control
destroy all
run
plot VCLK VOUT+4.5 VA+1.5 VB+3
.endc
.option scale=50nm
.tran .1n 70n UIC

VDD   VDD   0      DC    1
VA    VA    0      DC    0      PULSE  0 1 1n 0 0 20n 40n
VB    VB    0      DC    0      PULSE  0 1 5n 0 0 30n 60n
VCLK  VCLK  0      DC    0      PULSE  0 1 6n 0 0 10n 20n

M1    VA1     VA     VDD    VDD   PMOS L=1 W=20
M2    VA1     VA     0      0     NMOS L=1 W=10
M3    VB1     VB     VDD    VDD   PMOS L=1 W=20
M4    VB1     VB     0      0     NMOS L=1 W=10
M5    VOUT1   VCLK   VDD    VDD   PMOS L=1 W=20
M6    VOUT1   VA1    N1     0     NMOS L=1 W=10
M7    N1      VB     N2     0     NMOS L=1 W=10
M8    N2      VCLK   0      0     NMOS L=1 W=10
M9    VOUT1   VA     N3     0     NMOS L=1 W=10
M10   N3      VB1    N2     0     NMOS L=1 W=10
M11   VOUT    VOUT1  VDD    VDD   PMOS L=1 W=20
M12   VOUT    VOUT1  0      0     NMOS L=1 W=10
*50nm models included
```

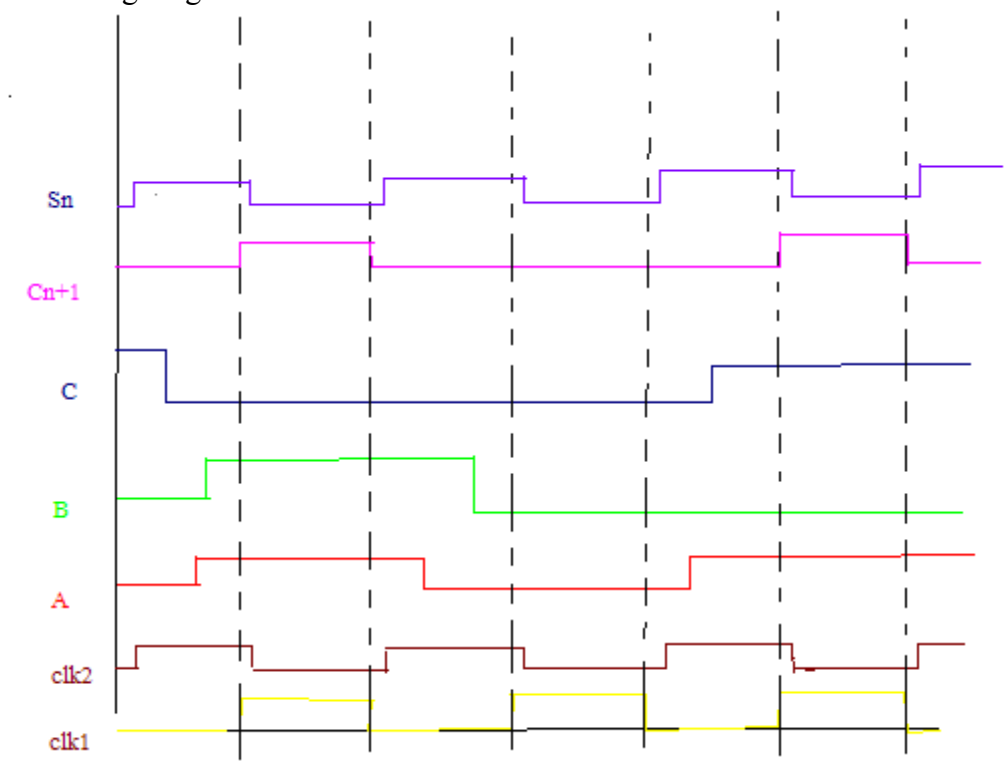


Problem 14.10 **Rupa Balan**

The circuit shown in fig 14.19 is the implementation of a high-speed adder cell (1-bit). What type of logic was used to implement the circuit? Using timing diagrams, describe the operation of the circuit.

Solution: NP logic was used to implement the circuit.

The timing diagrams for the circuit are shown:



Clk1 is same as phi and clk2 is same as complement of phi. A,B and C are the inputs.

C_{n+1} is the carry output and S_n is the sum output.

When clk1 (phi) is low or clk2 (phi complement) is high, C_{n+1} and S_n are in precharge phase.

Hence complement of C_{n+1} is a logic 1 which shuts the PMOS is the second stage off.

From the timing diagrams, during the precharge phase when clk1 is low or clk2 is a high C_{n+1} is logic 0 and S_n is logic 1.

When clk1 is high or clk2 is low, both stages are in the evaluation phase and both S_n and C_{n+1} change according to the inputs A, B and C.

The truth table is shown:

A	B	C	S_n	C_{n+1}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Problem 14.11 **Harish Reddy Singidi**

Discuss the design of a two-bit adder using adder cell of Fig. 14.19 if a clock running at 200 MHZ, is used with the two-bit adder, how long will it take to add two words? How long will it take if the word size is increased to 32 bits?

Fig. 14.19 is implemented by NP logic. It is used to add two one-bits words where as to add two two-bit words we can use pipelining. In pipelining bits of the word are delayed both on the input and output of the adder, so that all bits of the sum reach the output at the same time.

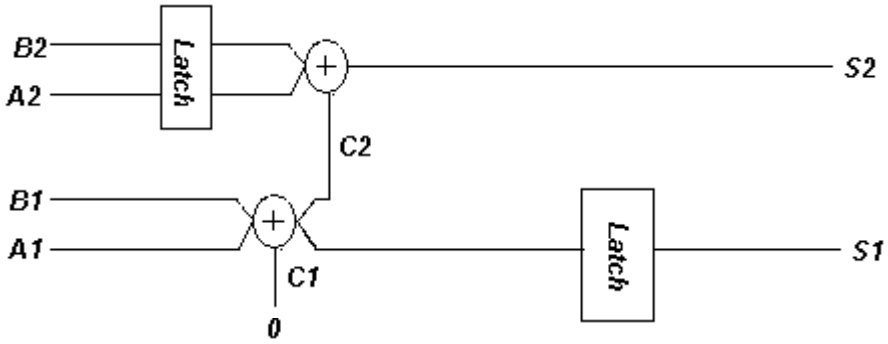


Fig: A Pipelined Adder

Clock is running at f = 200MHz

The time it takes to add two two-bit words is t = 2T = 2/f = 0.01us

The time it takes to add two 32-bit words is t = 32T = 32/f = 0.16us.

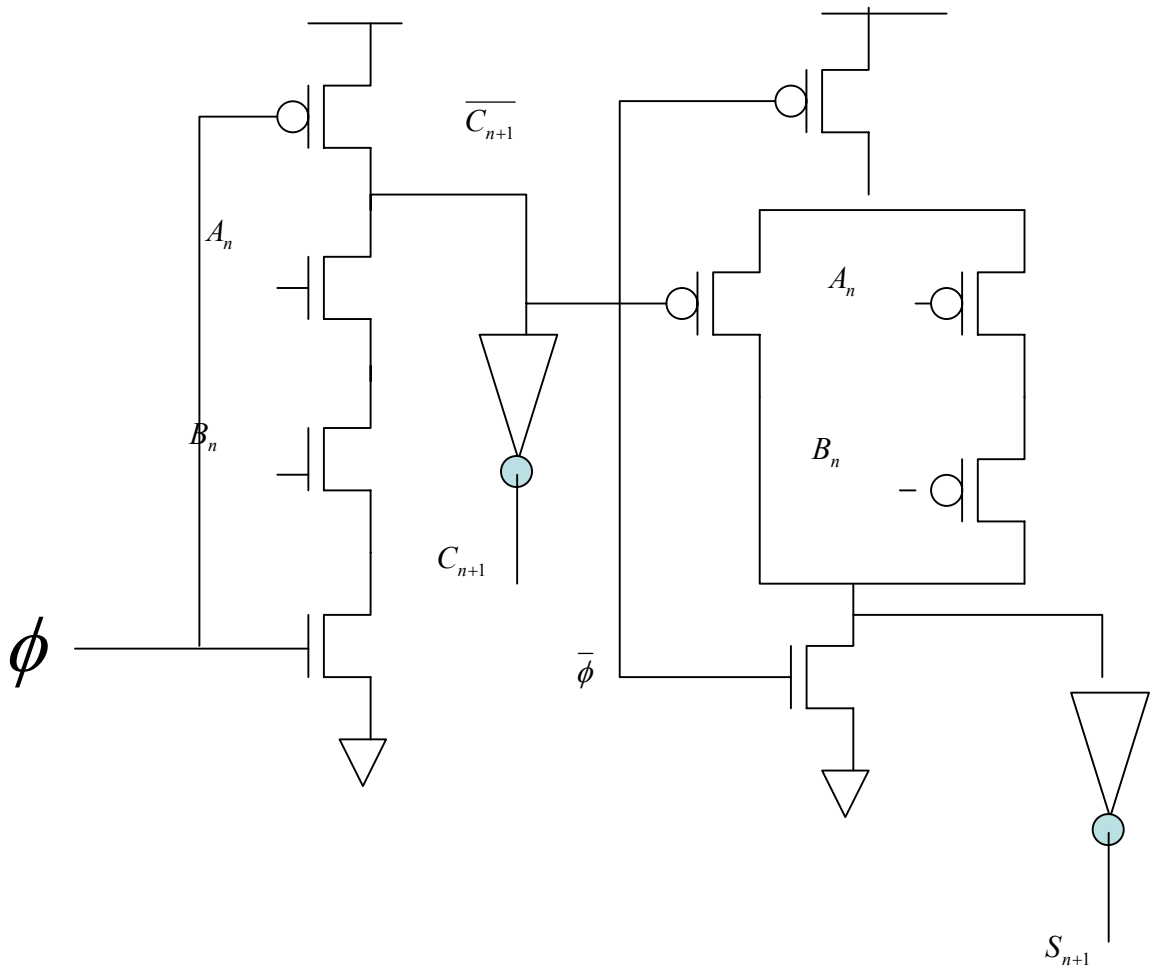
PROBLEM 14.12 **Shambhu Roy**

Sketch the implementation of an NP logic half adder cell.

Solution: For half adder

$S_n = A_n \oplus B_n = (A_n + B_n).(\overline{A_n} + \overline{B_n}) = (A_n + B_n).C_{n+1}$

$C_{n+1} = A_n B_n$

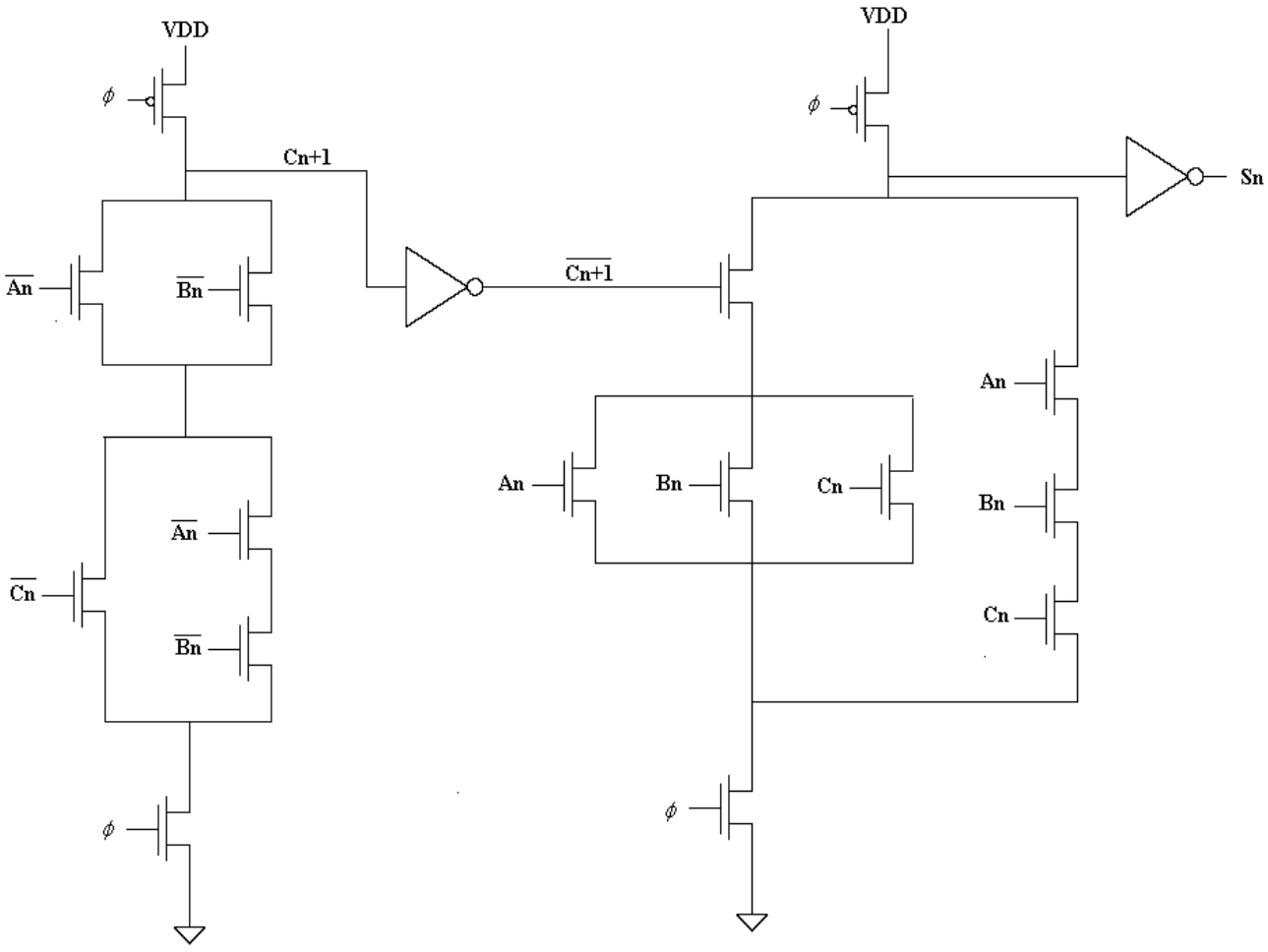


Problem 14.13 **Steve Bard**

Design (sketch the schematic of) a full adder circuit using PE logic.

Solution: A full adder performs the logic functions

$$Sn = \overline{An}\cdot\overline{Bn}\cdot Cn + \overline{An}\cdot Bn\cdot\overline{Cn} + An\cdot\overline{Bn}\cdot\overline{Cn} + An\cdot Bn\cdot Cn$$
$$= (An + Bn + Cn)\cdot(\overline{Cn+1}) + An\cdot Bn\cdot Cn$$
$$Cn+1 = An\cdot Bn + Cn\cdot(An + Bn)$$



Schematic of Full Adder using PE logic

Problem 14.14

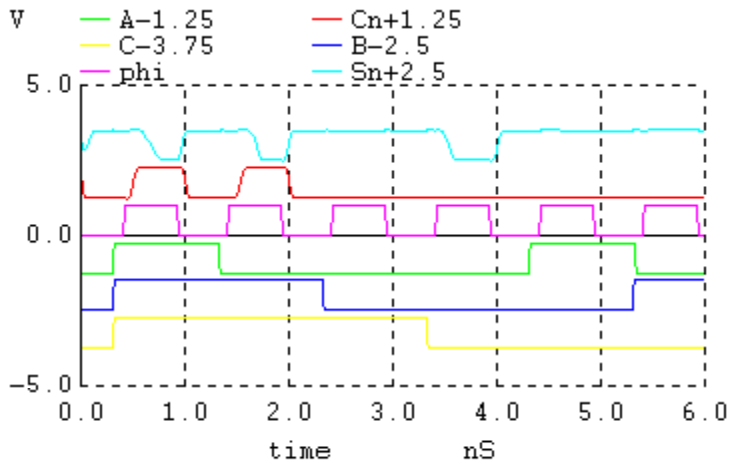
John Spratt

Simulate the operation of the circuit designed in Problem 14.10.

Netlist:

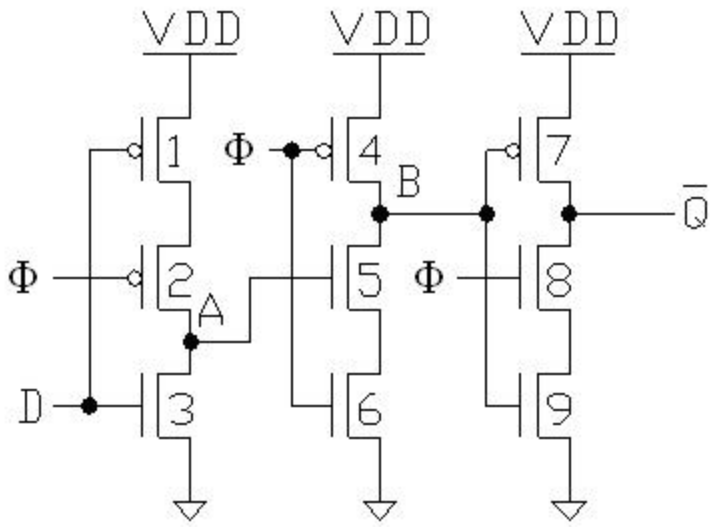
```
vdd       vdd       0         DC        1
VA        A         0         DC        0         PULSE 0 1 300p 0 0 1n 4n
VB        B         0         DC        0         PULSE 0 1 300p 0 0 2n 5n
VC        C         0         DC        0         PULSE 0 1 300p 0 0 3n 6n

Vphi      phi       0         DC        0         PULSE 0 1 400p 0 0 .5n 1n
Vphi_bar phi_bar 0  DC        0         PULSE 1 0 400p 0 0 .5n 1n

M1        Cn_bar phivdd       vdd       PMOS L=1 W=20
M2        Cn_bar A  n1        0         NMOS L=1 W=10
M3        n1        B         n2        0         NMOS L=1 W=10
M4        Cn_bar C  n3        0         NMOS L=1 W=10
M5        n3        A         n2        0         NMOS L=1 W=10
M6        n3        B         n2        0         NMOS L=1 W=10
M7        n2        phi       0         0         NMOS L=1 W=10

M8        Cn        Cn_bar    vdd       vdd       PMOS L=1 W=20
M9        Cn        Cn_bar    0         0         NMOS L=1 W=10

M10       n4        phi_bar   vdd       vdd       PMOS L=1 W=20
M11       n5        A         n4        vdd       PMOS L=1 W=20
M12       n5        B         n4        vdd       PMOS L=1 W=20
M13       n5        C         n4        vdd       PMOS L=1 W=20
M14       n6        Cn_bar n5 vdd       PMOS L=1 W=20
M15       n7        A         n5        vdd       PMOS L=1 W=20
M16       n8        B         n7        vdd       PMOS L=1 W=20
M17       n6        C         n8        vdd       PMOS L=1 W=20
M18       n6        phi_bar   0         0         NMOS L=1 W=10

M19       Sn        n6        vdd       vdd       PMOS L=1 W=20
M20       Sn        n6        0         0         NMOS L=1 W=10
```

Simulation:



Problem 14.15 **Justin Wood**

Show that the dynamic circuit shown in Fig 14.20 is an edge-triggered flip-flop [5]. Note that a single-phase clock signal is used.



Solution:

Case 1: Φ=0, D=0 or 1

When Φ=0, M4 is on, M6 is off, and node B is charged to D. This causes M7 to turn off and M9 to turn on. The output does not change regardless of the D input.

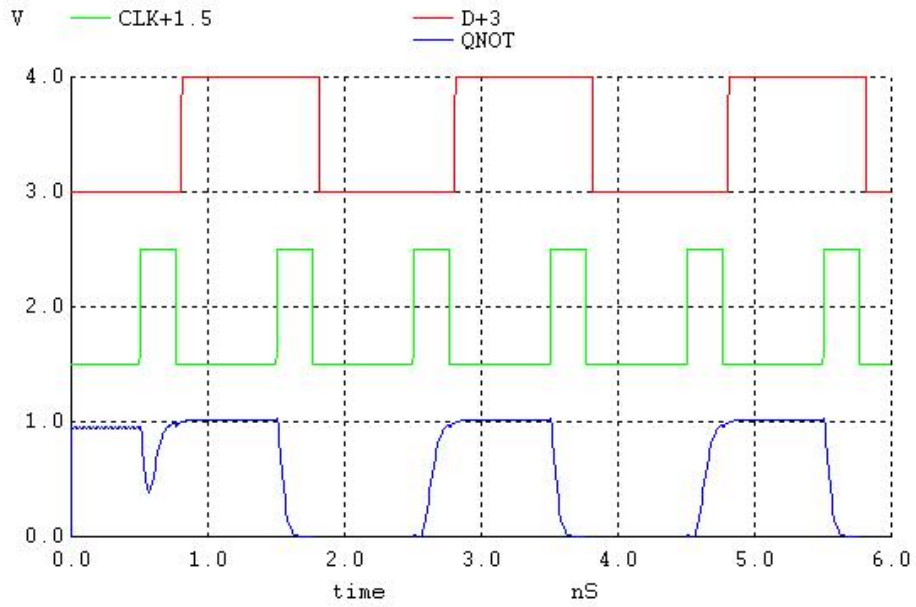
Case 2: D=0 and Φ transitions to 1

In this case, node A was charged to Vdd before Φ goes to 1. M5 is then on. When Φ goes to 1, M4 is off, M6 is on, and node B is pulled to ground. This causes M7 to turn on and M9 to turn off. Qnot is then pulled to Vdd.

Case 3: D=1 and Φ transitions to 1

Before Φ transistions to 1, node A has been pulled to ground from D being 0, which turns off M5. From case 1, node B was charged to Vdd when Φ was 0 and M7 is off and M9 is on. Once Φ changes to 1, M8 turns on, and Qnot is then pulled to logic 0.

Below is a spice simulation showing the logical operation.



Below is the spice netlist.

```
.control
destroy all
run
plot D+3 CLK+1.5 QNOT ylimit 0 4
.endc

.option scale=50n
.tran 10p 6000p uic
.ic v(Qnot)=0
vdd     vdd     0       DC      1
vin     d       0       DC      0 pulse 0 1 800p 0 0 1000p 2000p
vin2    clk     0       DC      0 pulse 0 1 500p 0 0 250p 1000p

M1      vdd     D       vn1     vdd     PMOS L=1 W=10
M2      vn1     CLK     vn2     vdd     PMOS L=1 W=10
M3      vn2     D       0       0       NMOS L=1 W=10
M4      vdd     clk     vn3     vdd     PMOS L=1 W=10
M5      vn3     vn2     vn4     0       NMOS L=1 W=10
M6      vn4     clk     0       0       NMOS L=1 W=10
M7      vdd     vn3     Qnot    vdd     PMOS L=1 W=10
M8      Qnot    clk     vn5     0       NMOS L=1 W=10
M9      vn5     vn3     0       0       NMOS L=1 W=10
```