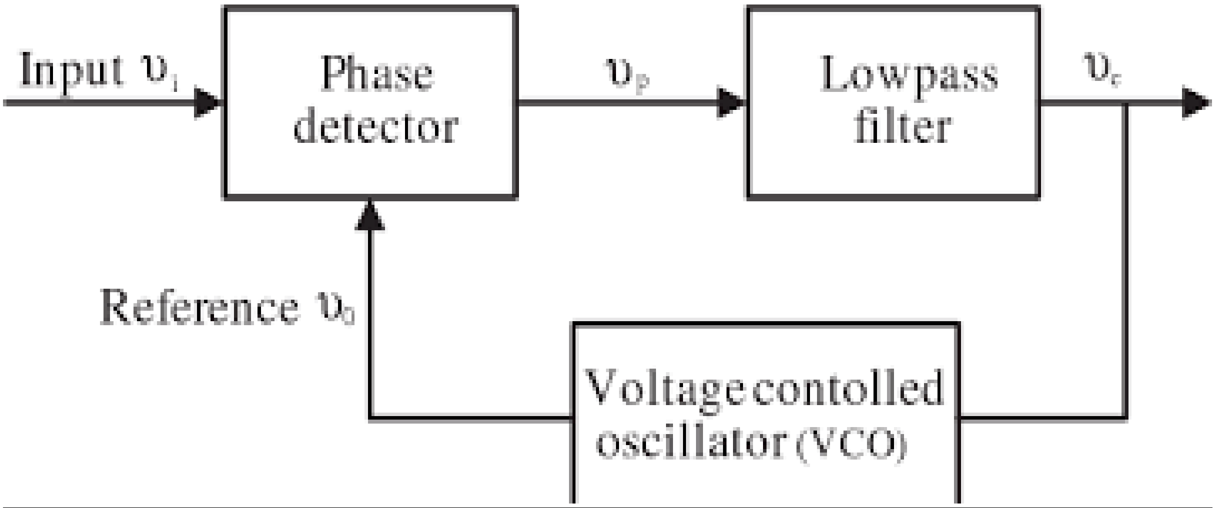
Problem Statement:

- Given a coded tech file and a basic magic layout file of a Cmos logic inverter.
- Go through the tech file, understand the meaning of each coded elements.
- Understand the DRC rules included in tech file.
- Observe the magic layout design of Cmos logic inverter, understand it by correlating with given tech file.
- Design the PLL (phase locked loop) circuit using the VCO circuit.
- Extract the spice from designed layout magic file (. mag file).
- Do post layout simulation using ng spice tool and verify the functionality of PLL circuit.

Solution:

A phase-locked loop (PLL) is an electronic circuit with a voltage or voltage-driven oscillator that constantly adjusts to match the frequency of an input signal.



The PLL compares the voltage-controlled oscillator signal with the input/reference signal. Because the PLL is both frequency- and phase-sensitive, it can detect both frequency and phase differences between the two signals.

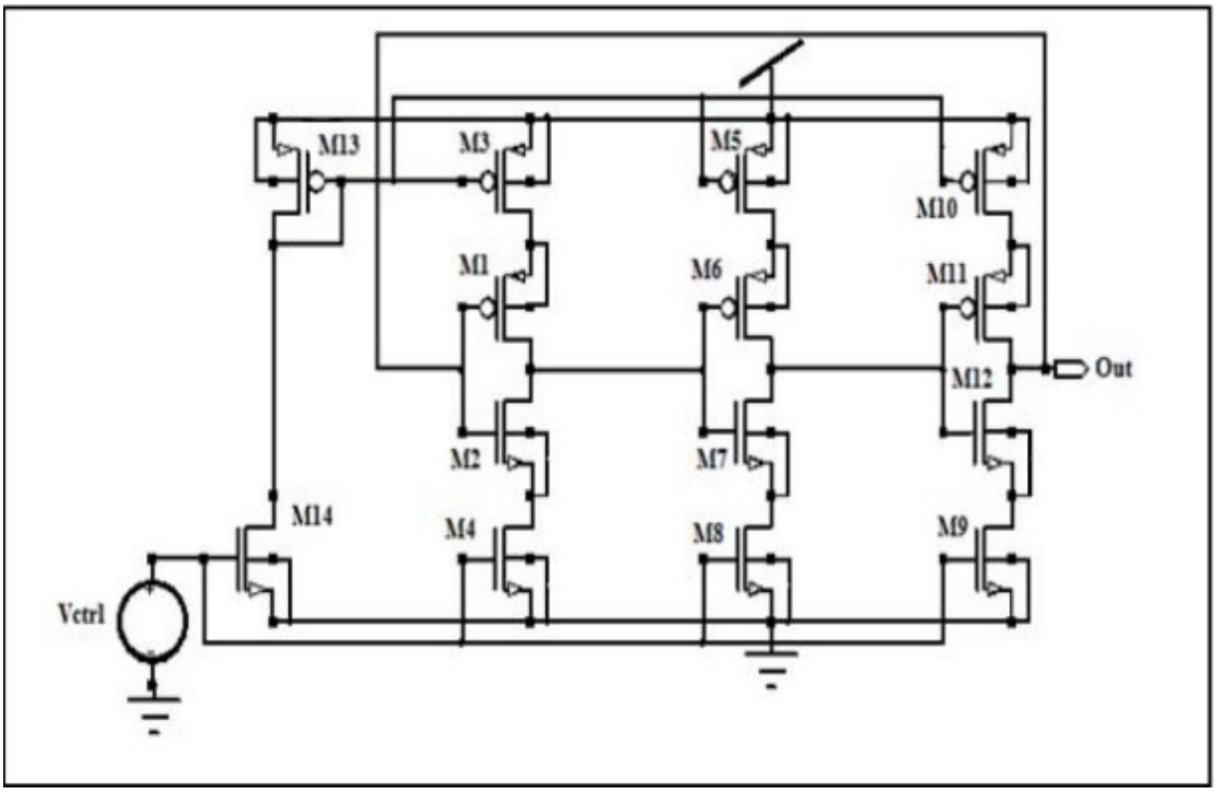
It generates an error signal that corresponds to the phase difference between the signals. This difference is passed on to the low-pass filter that removes any high-frequency elements, and filters the error signal into a varying direct current (DC) level. This "feedback signal" is then applied back to the voltage-controlled oscillator to control its frequency.

To start, this loop will be out of lock. The error signal will pull the voltage-controlled oscillator frequency toward the reference frequency, and continue to do so until it cannot reduce the error any further. At one point, however, the phase difference between the two signals will become zero (i.e., they will both be on exactly the same frequency).

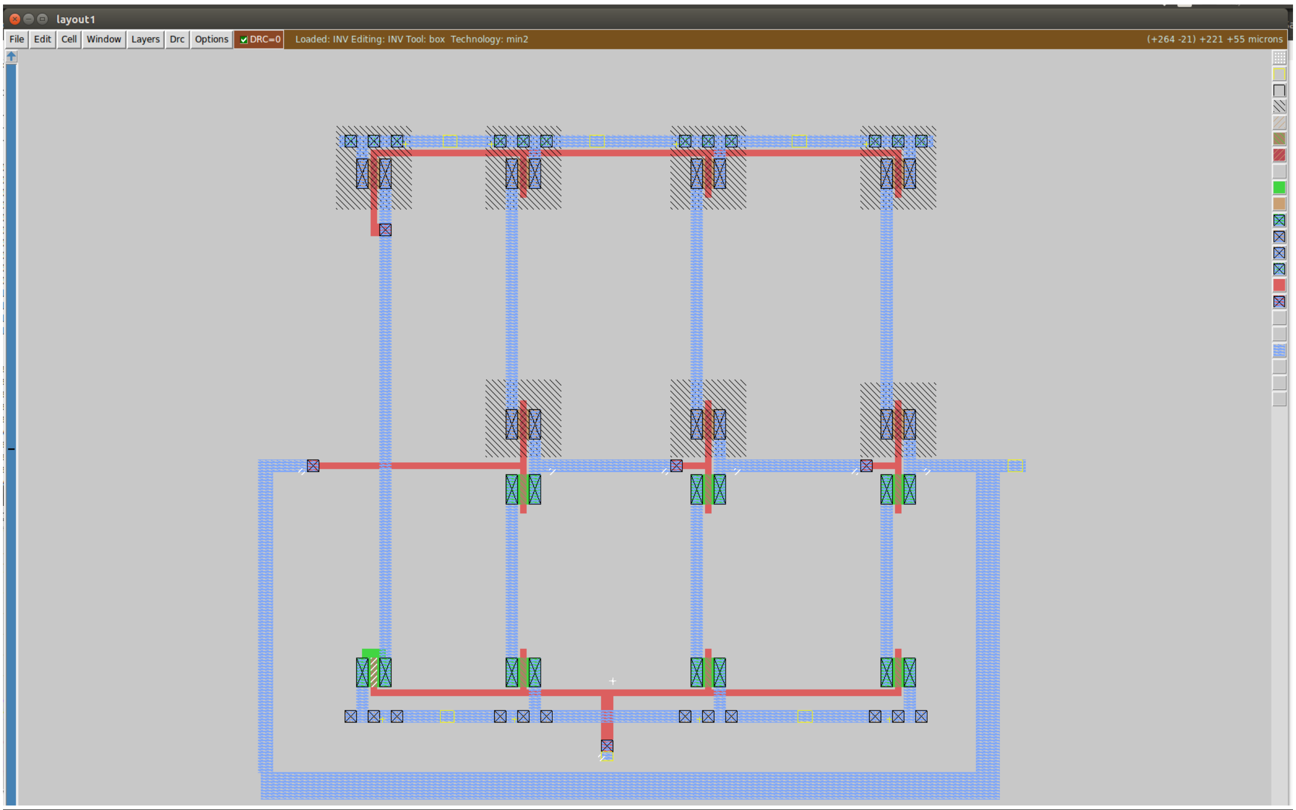
This is when the loop is said to be locked, and a steady-state error voltage is produced.

For generating PLL's Layout we first need VCO (voltage-controlled oscillator)'s layout.

<u>Below is the circuit used to generate the Layout in MAGIC for VCO:</u>

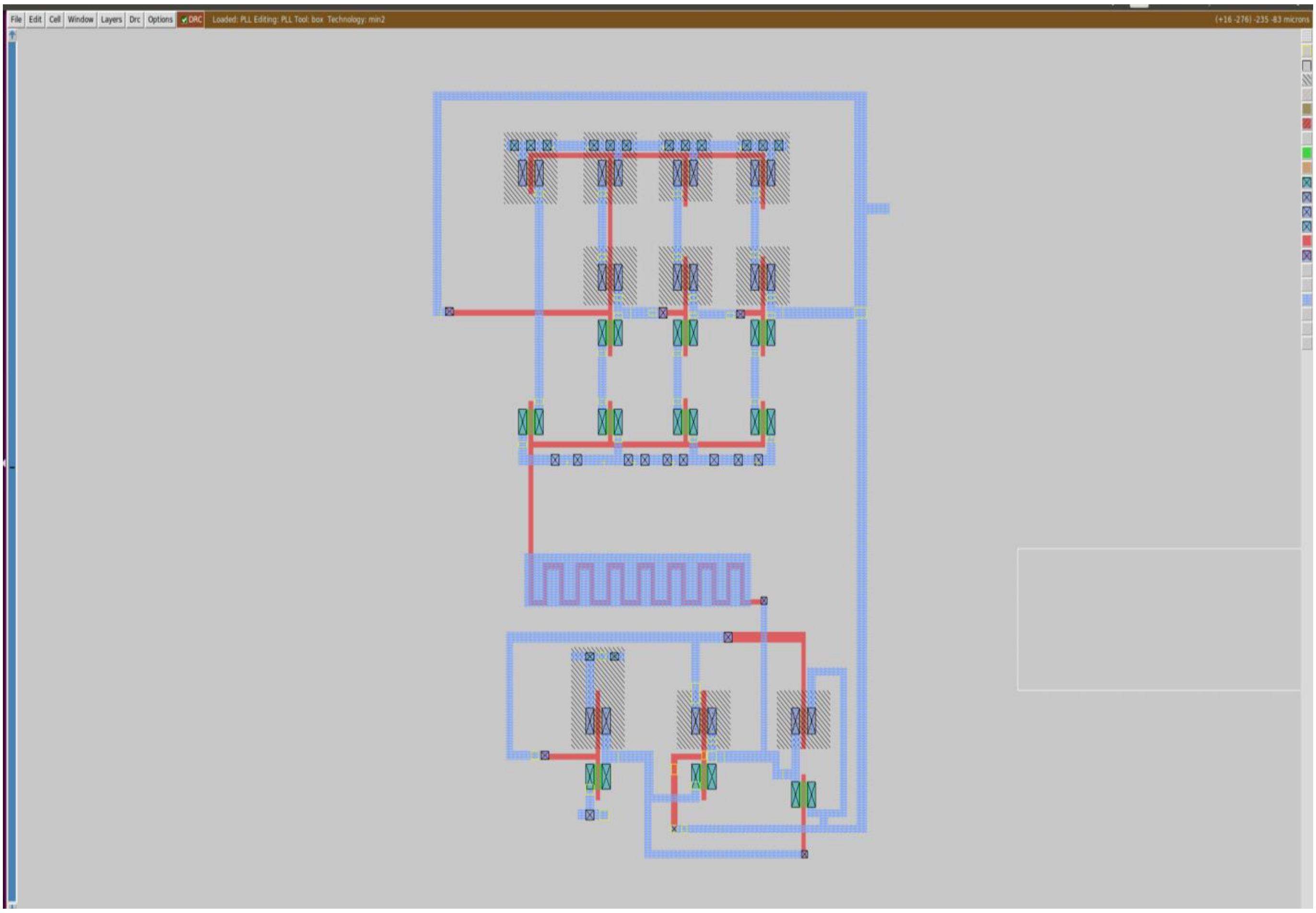


<u>Generated MAGIC Layout for VCO:</u>



Now using the above generated VCO we will generate a magic layout for PLL.

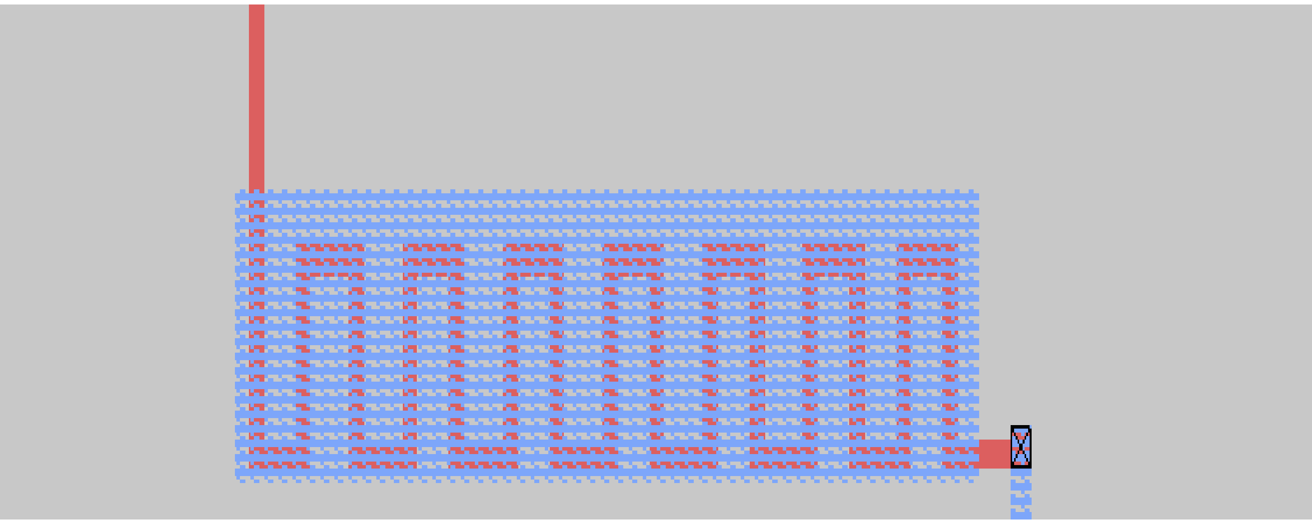
Below is the generated MAGIC Layout for PLL:



We can see that DRC is checked and there are no errors.

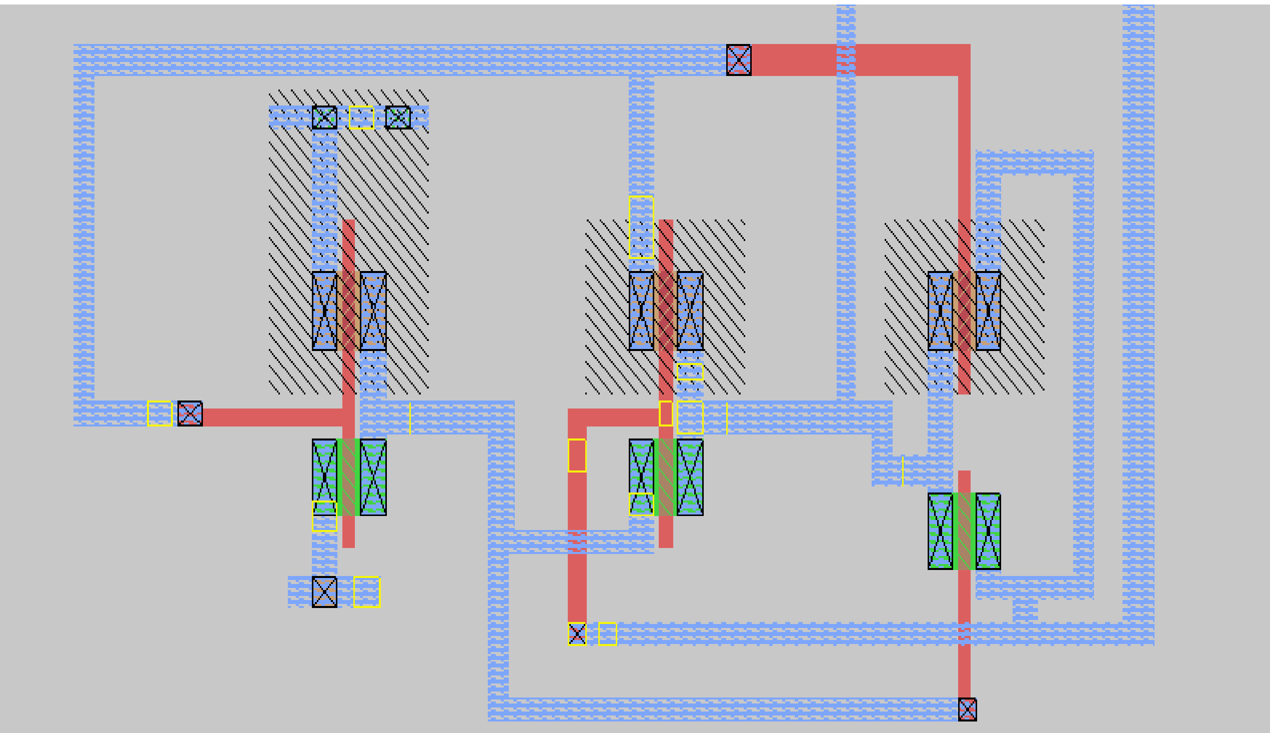
There are 2 additions to the previously obtained VCO:

1st is this RC filter made using a long-folded layer of polysilicon and N-well all over the polysilicon to make it behave like RC low pass filter.



2nd is the Phase Detector, for detection we will be using an X-OR gate.

MAGIC layout for X-OR is given below:



The generated MAGIC Layout for PLL is then converted into the .spice file and then that file is converted into .cir to run over NG-SPICE.

Below are the different code snippets for various files like .spice, .cir, .ext files.

PLL.ext →

PLL.ext - Notepad

File Edit Format View Help

```
timestamp 1660333099
version 8.3
tech min2
style lambda=0.09
scale 1000 1000 9
resistclasses 8000 8000 900000 900000 7400 80
node "m1_9_n160#" 0 1.05718 9 -160 m1 0 0 0 0 0 0 0 0 0 0 2331 264
node "a_39_n244#" 25 2.45618 39 -244 pdc 50 30 16 16 0 0 0 0 0 0 132 74
node "out" 199 5.54017 46 -232 ndiff 100 60 50 30 0 0 0 0 69 72 690 414
node "a_17_n220#" 488 3.09289 17 -220 pc 0 0 50 30 0 0 0 0 388 318 780 430
node "out" 2165 9.84027 11 -159 p 100 60 100 60 0 0 0 0 1111 1114 636 360
node "a_120_n92#" 56 3.29256 120 -92 ndc 100 60 0 0 0 0 0 0 0 0 180 98
node "a_82_n92#" 56 3.29256 82 -92 ndc 100 60 0 0 0 0 0 0 0 0 180 98
node "a_45_n92#" 56 3.29256 45 -92 ndc 100 60 0 0 0 0 0 0 0 0 180 98
node "a_89_n57#" 222 3.26269 89 -57 ndiff 50 30 50 30 0 0 0 0 106 108 197 118
node "out" 221 3.27095 52 -57 ndiff 50 30 50 30 0 0 0 0 110 110 216 116
node "a_120_n35#" 56 2.87892 120 -35 pdc 0 0 100 60 0 0 0 0 0 0 204 110
node "a_82_n35#" 56 2.8843 82 -35 pdc 0 0 100 60 0 0 0 0 0 0 204 110
node "a_45_n35#" 56 2.87892 45 -35 pdc 0 0 100 60 0 0 0 0 0 0 204 110
node "a_13_n92#" 34 3.39732 13 -92 ndiff 50 30 50 30 0 0 0 0 0 0 432 224
node "out" 1622 11.7047 -30 -45 pc 100 60 100 60 0 0 0 0 861 838 3550 1792
node "w_133_n216#" 1017 0.300196 133 -216 nw 0 0 0 0 598 98 0 0 0 0 0 0
node "w_84_n216#" 1017 0.300196 84 -216 nw 0 0 0 0 598 98 0 0 0 0 0 0
node "w_32_n216#" 1450 3.40101 32 -216 nw 24 28 50 30 1040 132 0 0 0 0 194 116
node "w_113_n41#" 1017 0.300196 113 -41 nw 0 0 0 0 598 98 0 0 0 0 0 0
node "w_75_n41#" 1017 0.300196 75 -41 nw 0 0 0 0 598 98 0 0 0 0 0 0
node "w_38_n41#" 1017 0.300196 38 -41 nw 0 0 0 0 598 98 0 0 0 0 0 0
node "vdd!" 13050 17.7319 -1 -1 nw 192 192 200 120 2886 430 0 0 0 0 776 396
substrate "gnd!" 0 0 6 -92 ndc 200 120 144 144 0 0 0 0 0 0 792 404
cap "vdd!" "a_120_n35#" 0.066584
cap "vdd!" "a_45_n35#" 0.066584
cap "w_113_n41#" "a_89_n57#" 0.060468
cap "w_133_n216#" "out" 0.061212
cap "w_32_n216#" "out" 0.061212
cap "w_84_n216#" "a_17_n220#" 0.066584
cap "vdd!" "out" 0.717756
cap "w_113_n41#" "a_120_n35#" 0.066584
cap "w_75_n41#" "out" 0.060468
cap "w_75_n41#" "a_82_n35#" 0.066584
cap "w_84_n216#" "out" 0.080652
cap "out" "a_17_n220#" 0.05652
cap "out" "out" 0.07065
```

PLL.mag→

PLL.mag - Notepad

File Edit Format View Help

```
magic
tech min2
timestamp 1660333099
<< nwell >>
rect -1 -1 25 27
rect 38 -1 64 27
rect 75 0 101 27
rect 113 -1 139 27
rect 38 -41 64 -18
rect 75 -41 101 -18
rect 113 -41 139 -18
rect 32 -216 58 -176
rect 84 -216 110 -193
rect 133 -216 159 -193
<< ntransistor >>
rect 50 -57 52 -47
rect 87 -57 89 -47
rect 125 -57 127 -47
rect 11 -92 13 -82
rect 50 -92 52 -82
rect 87 -92 89 -82
rect 125 -92 127 -82
rect 44 -232 46 -222
rect 96 -232 98 -222
rect 145 -239 147 -229
<< ptransistor >>
rect 11 6 13 16
rect 50 6 52 16
rect 87 6 89 16
rect 125 6 127 16
rect 50 -35 52 -25
rect 87 -35 89 -25
rect 125 -35 127 -25
rect 44 -210 46 -200
rect 96 -210 98 -200
rect 145 -210 147 -200
<< ndiffusion >>
rect 49 -57 50 -47
rect 52 -57 53 -47
rect 86 -57 87 -47
rect 89 -57 90 -47
rect 124 -57 125 -47
rect 127 -57 128 -47
rect 10 -92 11 -82
```

PLL.spice→

PLL.spice - Notepad

File Edit Format View Help

```
* SPICE3 file created from PLL_shivam.ext - technology: min2

.option scale=0.09u

M1000 ptd0 ptd0 vdd vdd pmos w=10 l=2
+  ad=50 pd=30 as=200 ps=120
M1001 pd3 i3 ns3 gnd nmos w=10 l=2
+  ad=100 pd=60 as=100 ps=60
M1002 gnd a_xor_b ns1 gnd nmos w=10 l=2
+  ad=250 pd=150 as=100 ps=60
M1003 gnd a_xor_b ns2 gnd nmos w=10 l=2
+  ad=0 pd=0 as=100 ps=60
M1004 inp_a_bar inpA vdd vdd pmos w=10 l=2
+  ad=50 pd=30 as=50 ps=30
M1005 pd3 inpA a_xor_b w_108_n209# pmos w=10 l=2
+  ad=100 pd=60 as=100 ps=60
M1006 ptd0 a_xor_b gnd gnd nmos w=10 l=2
+  ad=50 pd=30 as=0 ps=0
M1007 inp_a_bar inpA gnd gnd nmos w=10 l=2
+  ad=100 pd=60 as=0 ps=0
M1008 vdd ptd0 ps1 vdd pmos w=10 l=2
+  ad=0 pd=0 as=100 ps=60
M1009 i3 i2 ps2 w_47_1# pmos w=10 l=2
+  ad=50 pd=30 as=100 ps=60
M1010 pd3 i3 ps3 w_98_1# pmos w=10 l=2
+  ad=0 pd=0 as=100 ps=60
M1011 a_xor_b pd3 inpA w_70_n213# pmos w=10 l=2
+  ad=0 pd=0 as=50 ps=30
M1012 i3 i2 ns2 gnd nmos w=10 l=2
+  ad=50 pd=30 as=0 ps=0
M1013 a_xor_b pd3 inp_a_bar gnd nmos w=10 l=2
+  ad=100 pd=60 as=0 ps=0
M1014 vdd ptd0 ps3 vdd pmos w=10 l=2
+  ad=0 pd=0 as=0 ps=0
M1015 pd3 inp_a_bar a_xor_b gnd nmos w=10 l=2
+  ad=0 pd=0 as=0 ps=0
M1016 i2 pd3 ns1 gnd nmos w=10 l=2
+  ad=50 pd=30 as=0 ps=0
M1017 gnd a_xor_b ns3 gnd nmos w=10 l=2
+  ad=0 pd=0 as=0 ps=0
M1018 vdd ptd0 ps2 vdd pmos w=10 l=2
+  ad=0 pd=0 as=0 ps=0
M1019 i2 pd3 ps1 w_n1_1# pmos w=10 l=2
+  ad=50 pd=30 as=0 ps=0
```

PLL.cir→

PLL.cir - Notepad

File Edit Format View Help

```
M1001 pd3 i3 ns3 gnd cmosn W=Wn L=Lmin AD=(2*Wn*Lmin) PD=(2*Wn+4*Lmin) AS=(2*Wn*Lmin) PS=(2*Wn+4*Lmin)
M1002 gnd a_xor_b ns1 gnd cmosn W=Wn L=Lmin AD=(2*Wn*Lmin) PD=(2*Wn+4*Lmin) AS=(2*Wn*Lmin) PS=(2*Wn+4*Lmin)
M1003 gnd a_xor_b ns2 gnd cmosn W=Wn L=Lmin AD=(2*Wn*Lmin) PD=(2*Wn+4*Lmin) AS=(2*Wn*Lmin) PS=(2*Wn+4*Lmin)
M1004 inp_a_bar inpA vdd vdd cmosp W=Wp L=Lmin AD=(2*Wp*Lmin) PD=(2*Wp+4*Lmin) AS=(2*Wp*Lmin) PS=(2*Wp+4*Lmir
M1005 pd3 inpA a_xor_b w_108_n209# cmosp W=Wp L=Lmin AD=(2*Wp*Lmin) PD=(2*Wp+4*Lmin) AS=(2*Wp*Lmin) PS=(2*Wp+
M1006 ptd0 a_xor_b gnd gnd cmosn W=Wn L=Lmin AD=(2*Wn*Lmin) PD=(2*Wn+4*Lmin) AS=(2*Wn*Lmin) PS=(2*Wn+4*Lmin)
M1007 inp_a_bar inpA gnd gnd cmosn W=Wn L=Lmin AD=(2*Wn*Lmin) PD=(2*Wn+4*Lmin) AS=(2*Wn*Lmin) PS=(2*Wn+4*Lmir
M1008 vdd ptd0 ps1 vdd cmosp W=Wp L=Lmin AD=(2*Wp*Lmin) PD=(2*Wp+4*Lmin) AS=(2*Wp*Lmin) PS=(2*Wp+4*Lmin)
M1009 i3 i2 ps2 w_47_1# cmosp W=Wp L=Lmin AD=(2*Wp*Lmin) PD=(2*Wp+4*Lmin) AS=(2*Wp*Lmin) PS=(2*Wp+4*Lmin)
M1010 pd3 i3 ps3 w_98_1# cmosp W=Wp L=Lmin AD=(2*Wp*Lmin) PD=(2*Wp+4*Lmin) AS=(2*Wp*Lmin) PS=(2*Wp+4*Lmin)
M1011 a_xor_b pd3 inpA w_70_n213# cmosp W=Wp L=Lmin AD=(2*Wp*Lmin) PD=(2*Wp+4*Lmin) AS=(2*Wp*Lmin) PS=(2*Wp+4
M1012 i3 i2 ns2 gnd cmosn W=Wn L=Lmin AD=(2*Wn*Lmin) PD=(2*Wn+4*Lmin) AS=(2*Wn*Lmin) PS=(2*Wn+4*Lmin)
M1013 a_xor_b pd3 inp_a_bar gnd cmosn W=Wn L=Lmin AD=(2*Wn*Lmin) PD=(2*Wn+4*Lmin) AS=(2*Wn*Lmin) PS=(2*Wn+4*L
M1014 vdd ptd0 ps3 vdd cmosp W=Wp L=Lmin AD=(2*Wp*Lmin) PD=(2*Wp+4*Lmin) AS=(2*Wp*Lmin) PS=(2*Wp+4*Lmin)
M1015 pd3 inp_a_bar a_xor_b gnd cmosn W=Wn L=Lmin AD=(2*Wn*Lmin) PD=(2*Wn+4*Lmin) AS=(2*Wn*Lmin) PS=(2*Wn+4*L
M1016 i2 pd3 ns1 gnd cmosn W=Wn L=Lmin AD=(2*Wn*Lmin) PD=(2*Wn+4*Lmin) AS=(2*Wn*Lmin) PS=(2*Wn+4*Lmin)
M1017 gnd a_xor_b ns3 gnd cmosn W=Wn L=Lmin AD=(2*Wn*Lmin) PD=(2*Wn+4*Lmin) AS=(2*Wn*Lmin) PS=(2*Wn+4*Lmin)
M1018 vdd ptd0 ps2 vdd cmosp W=Wp L=Lmin AD=(2*Wp*Lmin) PD=(2*Wp+4*Lmin) AS=(2*Wp*Lmin) PS=(2*Wp+4*Lmin)
M1019 i2 pd3 ps1 w_n1_1# cmosp W=Wp L=Lmin AD=(2*Wp*Lmin) PD=(2*Wp+4*Lmin) AS=(2*Wp*Lmin) PS=(2*Wp+4*Lmin)
C0 a_xor_b gnd 11.89fF
C1 ns3 gnd 3.25fF
C2 ns2 gnd 3.25fF
C3 ns1 gnd 3.26fF
C4 i3 gnd 3.34fF
C5 i2 gnd 3.04fF
C6 pd3 gnd 12.30fF
C7 ptd0 gnd 3.86fF
C8 vdd gnd 2.71fF
C9 vdd gnd 17.81fF

v2 vdd 0 dc 3.3
v1 inpA gnd dc 0 PULSE(3.3 0  8nS 2pS 2pS 4nS 8nS)

.tran 0.5nS 40nS

.control
run
pLot V(inpA) 4+V(inpB)

.endc
.end
```

OUTPUTS:

NG-SPICE TERMINAL:

ngspice 35

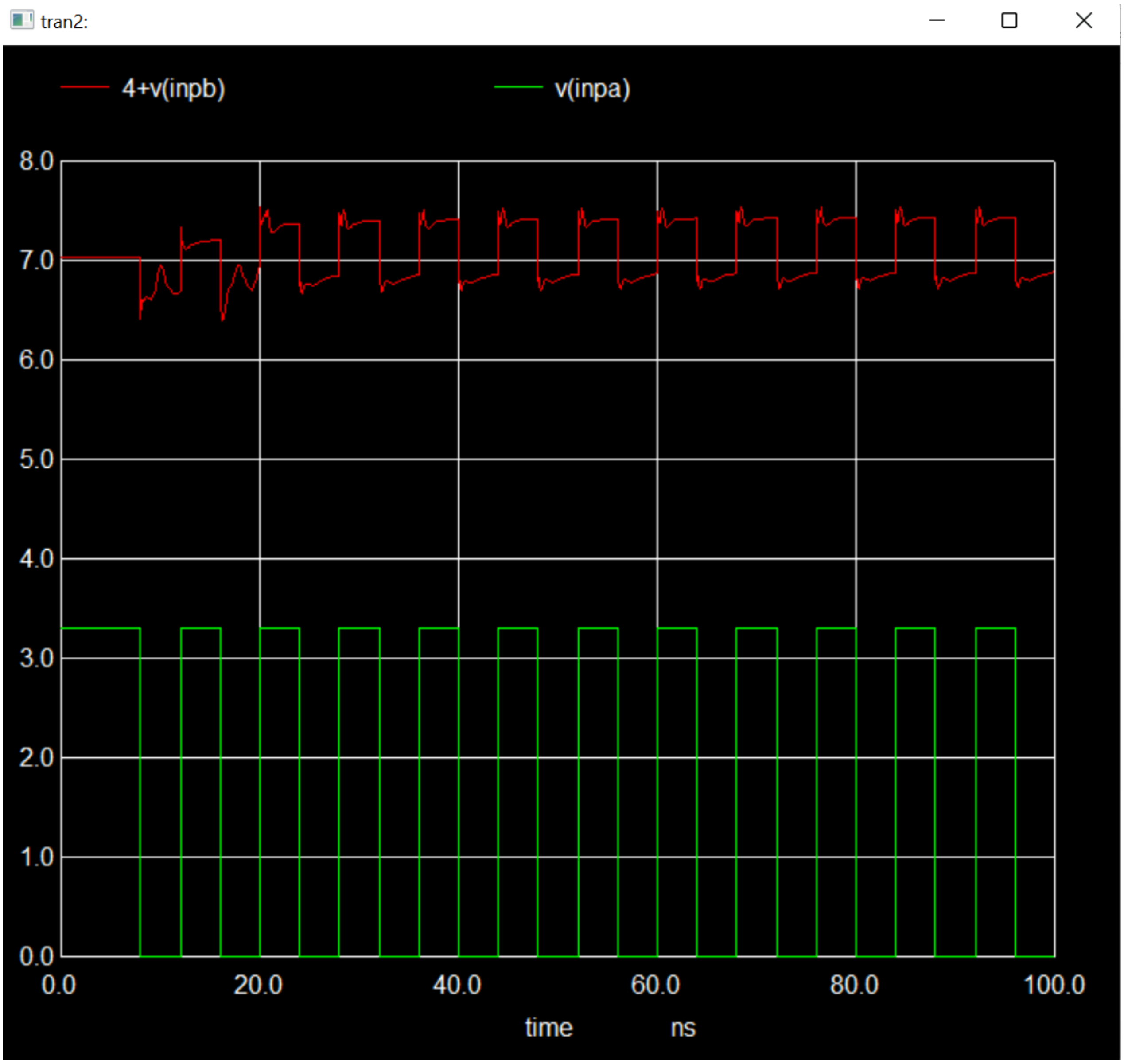
```
Initial Transient Solution
--------------------------

Node                                   Voltage
----                                   -------
ptd0                                    0.9328
vdd                                        3.3
pd3                                    1.66049
i3                                     1.66049
ns3                                   0.172332
a_xor_b                                2.88112
ns1                                   0.172332
ns2                                   0.172332
inp_a_bar                             0.167011
inpa                                       3.3
inpb                                   3.03149
ps1                                    3.05562
i2                                     1.66049
ps2                                    3.05562
ps3                                    3.05562
v1#branch                          -7.8279e-05
v2#branch                         -0.000493252

No. of Data Rows : 717
ngspice 3 ->
```

pll.cir -- ready -- Quit

PLL PLOT:



CONCLUSION:

From the above plot and Ng-spice terminal results we can conclude that after certain duration we get in phase locked output voltage wave with respect to input voltage waveform.