

# EE 618: CMOS Analog VLSI Design

## ASSIGNMENT 1



**Student Name:** Bhuvansh Goyal

**Roll Number:** 22B3908

**Instructor:** Prof. Rajesh Zele

**Submission Date:** 20 Aug 2025

# Contents

<b>1 Characterizing the MOS devices</b>	<b>2</b>
1.1 $I_{sd}$ vs $V_{sg}$ for PMOS	2
1.2 $I_{sd}$ vs $V_{sd}$ for PMOS	2
1.3 $I_{ds}$ vs $V_{gs}$ for NMOS	3
1.4 $I_{ds}$ vs $V_{ds}$ for NMOS	3
1.5 $g_{mp}$ Calculation	4
1.6 $g_{mn}$ Calculation	4
1.7 $r_{outp}$ Calculation	4
1.8 $r_{outn}$ Calculation	4
1.9 $V_{thp}$ Calculation	5
1.10 $V_{thn}$ Calculation	5
1.11 $\lambda_p$ Calculation	6
1.12 $\lambda_n$ Calculation	6
1.13 $\mu_p C_{ox}$ Calculation	6
1.14 $\mu_n C_{ox}$ Calculation	6
<b>2 DC Analysis</b>	<b>7</b>
2.1 DC Transfer Curve of Inverter	7
2.2 Gain vs $V_{in}$ Plot of Inverter	7
<b>3 Transient Analysis</b>	<b>8</b>
3.1 Output and Input Waveforms	8
<b>4 Code</b>	<b>9</b>
4.1 NMOS	9
4.2 PMOS	11
4.3 Inverter	14

# 1 Characterizing the MOS devices

## 1.1 $I_{sd}$ vs $V_{sg}$ for PMOS

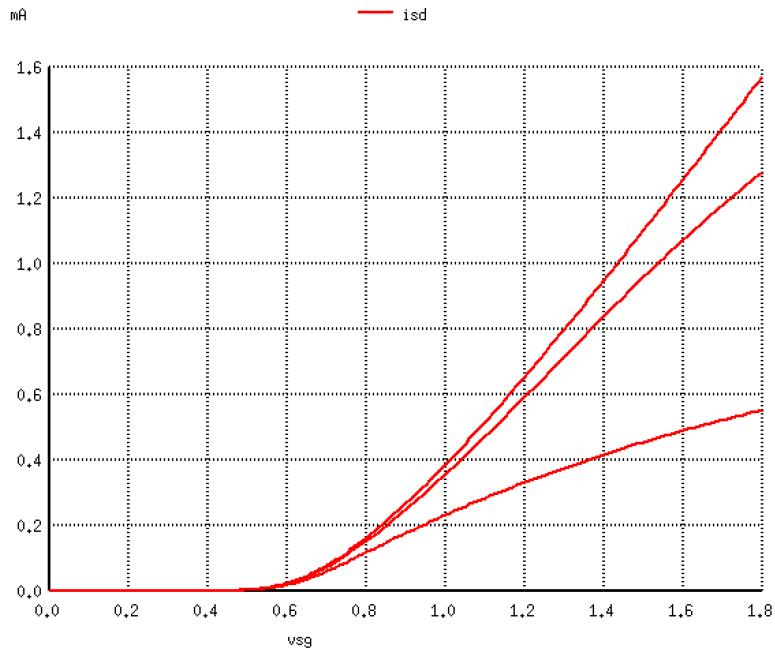


Figure 1:  $I_{sd}$  vs  $V_{sg}$  for  $V_{sd} = 0.2, 0.6, 1.0$  V

## 1.2 $I_{sd}$ vs $V_{sd}$ for PMOS

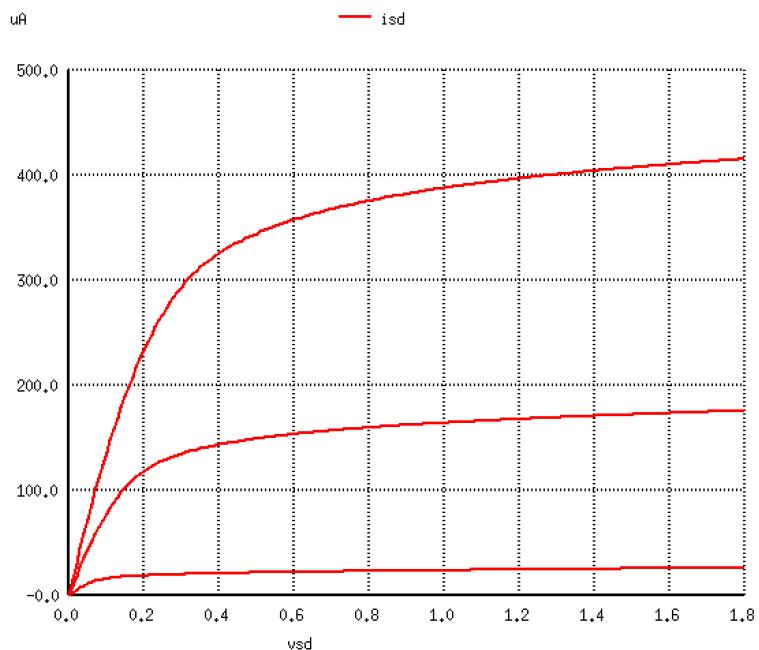


Figure 2:  $I_{sd}$  vs  $V_{sd}$  for  $V_{sg} = 0.6, 0.8, 1.0$  V

### 1.3 $I_{ds}$ vs $V_{gs}$ for NMOS

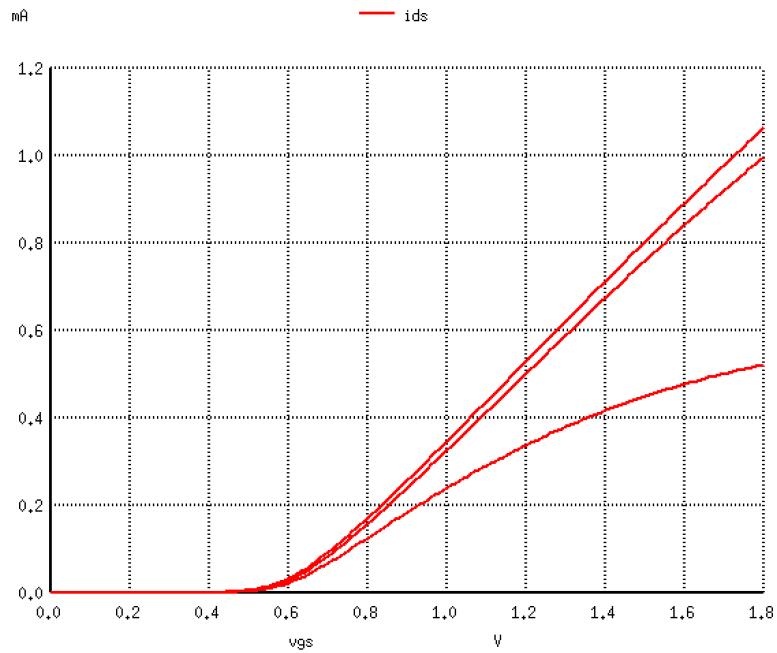


Figure 3:  $I_{ds}$  vs  $V_{gs}$  for  $V_{ds} = 0.2, 0.6, 1.0$  V

### 1.4 $I_{ds}$ vs $V_{ds}$ for NMOS

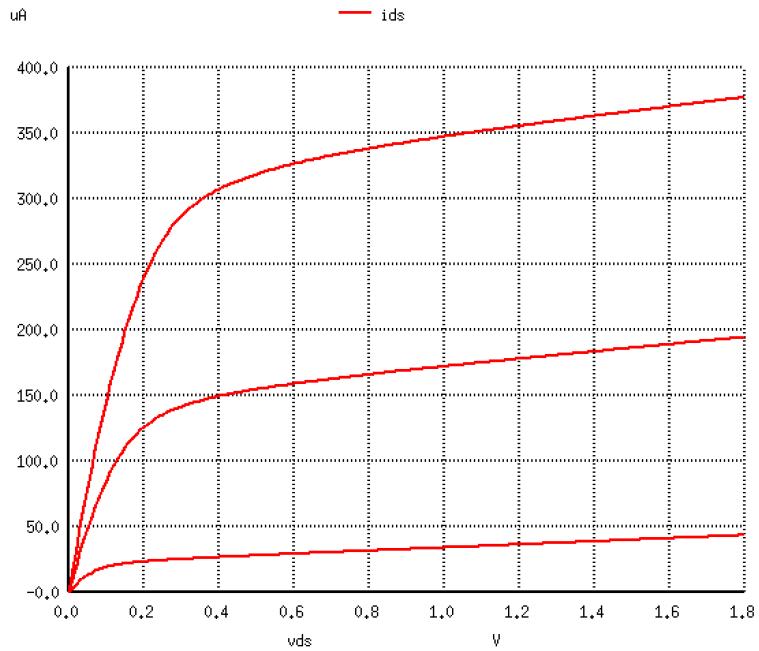


Figure 4:  $I_{ds}$  vs  $V_{ds}$  for  $V_{gs} = 0.6, 0.8, 1.0$  V

## 1.5 $g_{mp}$ Calculation

$g_{mp}$  is calculated at  $V_{sg} = 0.9\text{ V}$  and  $V_{sd} = 1.0\text{ V}$ . By definition,

$$g_{mp} = \left. \frac{\partial I_{sd}}{\partial V_{sg}} \right|_{V_{sd} \text{ constant}}$$

The derivative is taken from the  $I_{sd}$ - $V_{sg}$  curve at  $V_{sd} = 1.0\text{ V}$ , using the `deriv` function in ngspice. The result obtained is:

$$g_{mp} = 1.125 \text{ mS}$$

## 1.6 $g_{mn}$ Calculation

$g_{mn}$  is calculated at  $V_{gs} = 0.9\text{ V}$  and  $V_{ds} = 1.0\text{ V}$ . By definition,

$$g_{mn} = \left. \frac{\partial I_{ds}}{\partial V_{gs}} \right|_{V_{ds} \text{ constant}}$$

The derivative is taken from the  $I_{ds}$ - $V_{gs}$  curve at  $V_{ds} = 1.0\text{ V}$ , using the `deriv` function in ngspice. The result obtained is:

$$g_{mn} = 0.880 \text{ mS}$$

## 1.7 $r_{outp}$ Calculation

$r_{outp}$  is calculated at  $I_{sd} = 400\text{ }\mu\text{A}$ . By definition,

$$r_{outp} = \frac{1}{\left. \frac{\partial I_{sd}}{\partial V_{sd}} \right|}$$

The derivative is taken from the  $I_{sd}$ - $V_{sd}$  curve at  $V_{sg} = 1.0\text{ V}$  and  $I_{sd} = 400\text{ }\mu\text{A}$ , using the `deriv` function in ngspice. The result obtained is:

$$r_{outp} = 27.132 \text{ k}\Omega$$

## 1.8 $r_{outn}$ Calculation

$r_{outn}$  is calculated at  $I_{ds} = 350\text{ }\mu\text{A}$ . By definition,

$$r_{outn} = \frac{1}{\left. \frac{\partial I_{ds}}{\partial V_{ds}} \right|}$$

The derivative is taken from the  $I_{ds}$ - $V_{ds}$  curve at  $V_{gs} = 1.0\text{ V}$  and  $I_{ds} = 350\text{ }\mu\text{A}$ , using the `deriv` function in ngspice. The result obtained is:

$$r_{outn} = 24.348 \text{ k}\Omega$$

## 1.9 $V_{thp}$ Calculation

$V_{thp}$  is obtained by extrapolating the slope of the  $\sqrt{I_{sd}}$ - $V_{sg}$  curve at  $V_{sg} = 0.7$  V, where  $V_{sd} = 1.0$  V. After this point, the curve bends down as the device enters the linear/triode region. The point where the extrapolated line meets the  $V_{sg}$  axis is taken as  $V_{thp}$ .

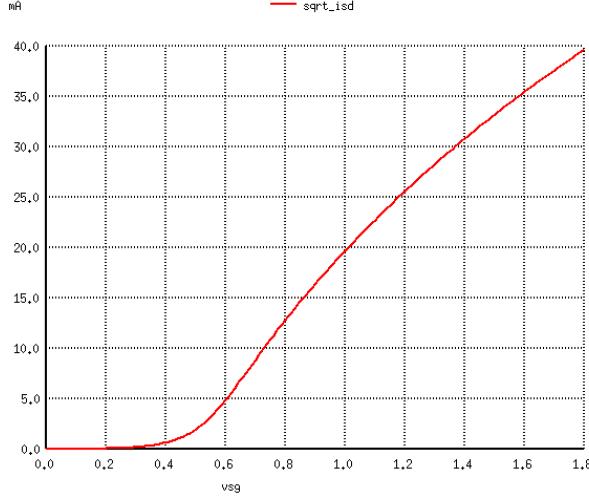


Figure 5:  $\sqrt{I_{sd}}$  vs  $V_{sg}$  for  $V_{sd} = 1.0$  V

The value obtained is:

$$V_{thp} = 0.482 \text{ V}$$

## 1.10 $V_{thn}$ Calculation

$V_{thn}$  is obtained by extrapolating the slope of the  $\sqrt{I_{ds}}$ - $V_{gs}$  curve at  $V_{gs} = 0.7$  V, where  $V_{ds} = 1.0$  V. After this point, the curve bends down as the device enters the linear/triode region. The point where the extrapolated line meets the  $V_{gs}$  axis is taken as  $V_{thn}$ .

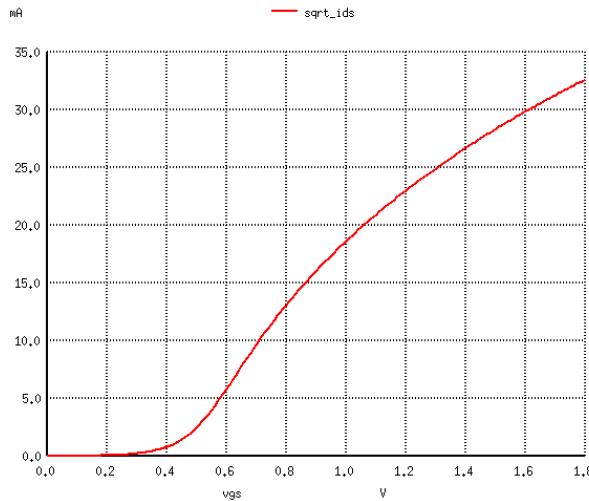


Figure 6:  $\sqrt{I_{ds}}$  vs  $V_{gs}$  for  $V_{ds} = 1.0$  V

The value obtained is:

$$V_{thn} = 0.438 \text{ V}$$

## 1.11 $\lambda_p$ Calculation

$\lambda_p$  is calculated using the values of  $r_{outp}$ ,  $I_{sd}$ , and  $V_{sd}$  at the same operating point where  $r_{outp}$  was obtained. The formula used is:

$$\lambda_p = \frac{1}{(I_{sd} \cdot r_{outp}) - V_{sd}}$$

Substituting the values from the  $r_{outp}$  calculation, the result obtained is:

$$\lambda_p = 0.104 \text{ V}^{-1}$$

## 1.12 $\lambda_n$ Calculation

$\lambda_n$  is calculated using the values of  $r_{outn}$ ,  $I_{ds}$ , and  $V_{ds}$  at the same operating point where  $r_{outn}$  was obtained. The formula used is:

$$\lambda_n = \frac{1}{(I_{ds} \cdot r_{outn}) - V_{ds}}$$

Substituting the values from the  $r_{outn}$  calculation, the result obtained is:

$$\lambda_n = 0.134 \text{ V}^{-1}$$

## 1.13 $\mu_p C_{ox}$ Calculation

$\mu_p C_{ox}$  is calculated under the same conditions as  $g_{mp}$ , i.e., at  $V_{sg} = 0.9 \text{ V}$  and  $V_{sd} = 1.0 \text{ V}$ . The formula used is:

$$\mu_p C_{ox} = \frac{g_{mp} \cdot L}{W \cdot (V_{sg} - V_{thp}) \cdot (1 + \lambda_p V_{sd})}.$$

After substituting the values, the result obtained is:

$$\mu_p C_{ox} = 62.712 \text{ } \mu\text{A/V}^2$$

## 1.14 $\mu_n C_{ox}$ Calculation

$\mu_n C_{ox}$  is calculated under the same conditions as  $g_{mn}$ , i.e., at  $V_{gs} = 0.9 \text{ V}$  and  $V_{ds} = 1.0 \text{ V}$ . The formula used is:

$$\mu_n C_{ox} = \frac{g_{mn} \cdot L}{W \cdot (V_{gs} - V_{thn}) \cdot (1 + \lambda_n V_{ds})}.$$

After substituting the values, the result obtained is:

$$\mu_n C_{ox} = 151.292 \text{ } \mu\text{A/V}^2$$

## 2 DC Analysis

### 2.1 DC Transfer Curve of Inverter

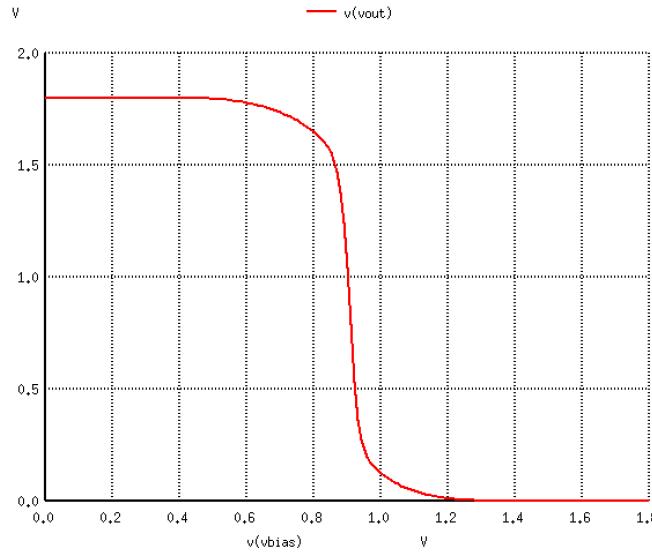


Figure 7:  $V_{out}$  vs  $V_{in}$  for inverter

### 2.2 Gain vs $V_{in}$ Plot of Inverter

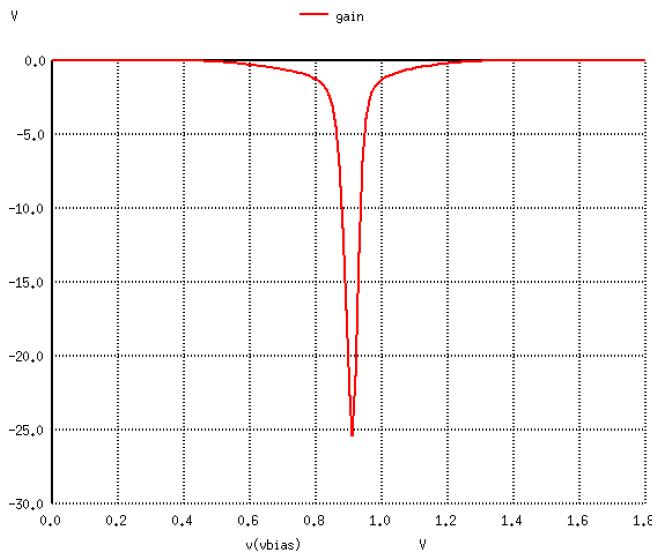


Figure 8: Gain vs  $V_{in}$  plot for inverter

From the above plot, here is the gain at  $V_{in} = 0.9 V$ :

$$\text{Gain} = \frac{dV_{out}}{dV_{in}} = -22.636$$

### 3 Transient Analysis

#### 3.1 Output and Input Waveforms

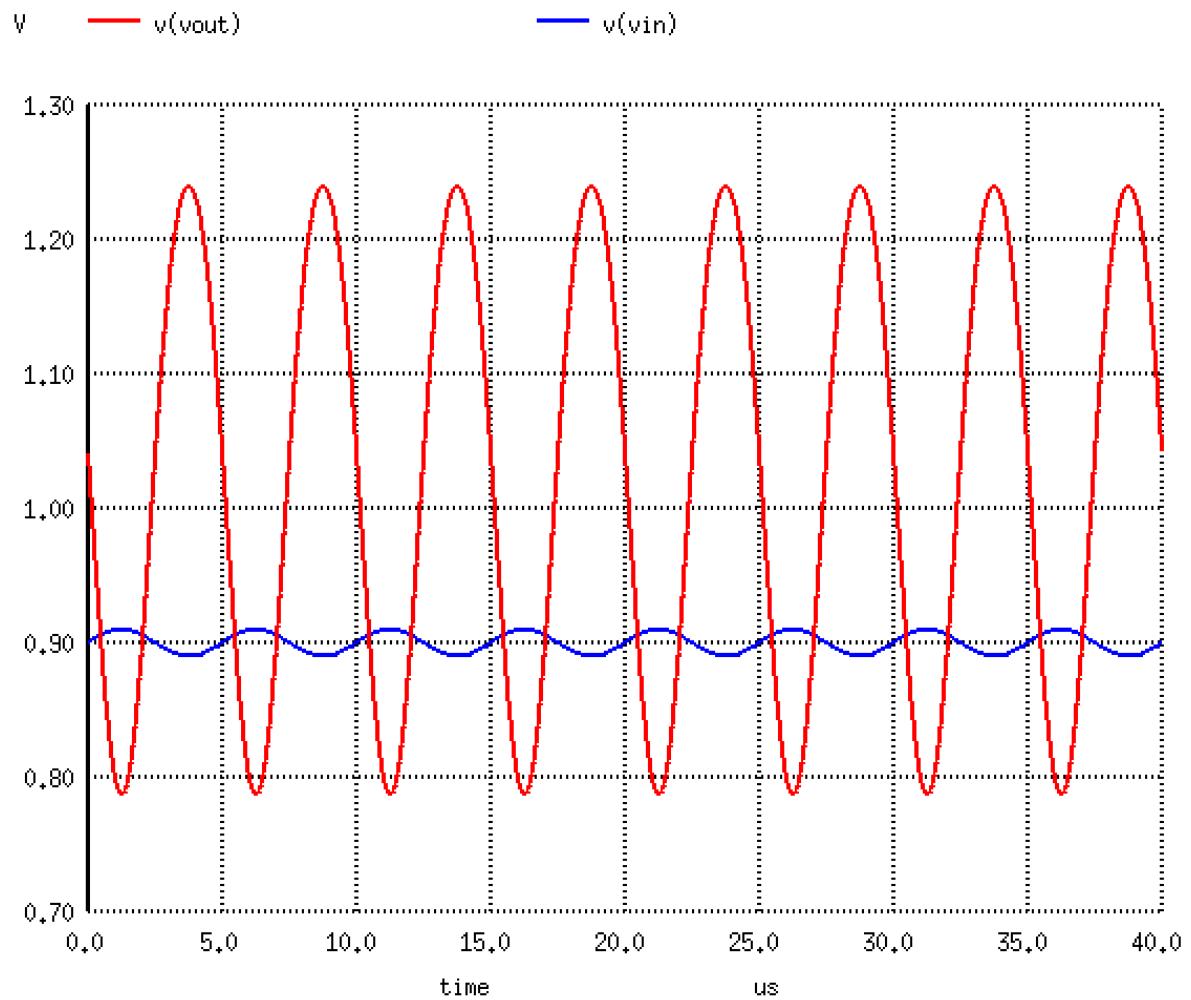


Figure 9: Transient waveforms of inverter showing  $V_{out}$  and  $V_{in}$

The measured peak-to-peak output voltage is:

$$V_{out,pp} = 0.453 \text{ V}$$

The measured peak-to-peak input voltage is:

$$V_{in,pp} = 0.020 \text{ V}$$

Therefore, the gain observed from the transient analysis is:

$$\text{Gain} = \frac{V_{out,pp}}{V_{in,pp}} = -22.647$$

## 4 Code

### 4.1 NMOS

Listing 1: Ngspice netlist for NMOS characterization

```
1 .title NMOS Characterization
2
3 .include models-180nm
4
5 vds d 0 dc 0
6 vgs g 0 dc 0
7
8 mn d g 0 0 CMOSN W=2u L=0.18u
9
10 .control
11
12 run
13
14 set color0 = white
15 set color1 = black
16 set color2 = red
17
18 let gm_probe_vgs = 0.9
19 let gout_probe_ids = 350u
20 let vth_probe_vgs = 0.7
21
22 let k = 0
23 let w = 2u
24 let l = 0.18u
25
26 * ----- Ids vs Vgs @ Vds = 0.2V -----
27
28 alter vds dc 0.2
29 dc vgs 0 1.8 0.01
30
31 let ids = -i(vds)
32 let vgs = v(g)
33
34 plot ids vs vgs
35
36 * ----- Ids vs Vgs @ Vds = 0.6V -----
37
38 alter vds dc 0.6
39 dc vgs 0 1.8 0.01
40
41 let ids = -i(vds)
42 let vgs = v(g)
43
44 plot ids vs vgs
45
46 * ----- Ids vs Vgs @ Vds = 1V -----
47
48 alter vds dc 1
49 dc vgs 0 1.8 0.01
50
51 let ids = -i(vds)
52 let vgs = v(g)
```

```

53
54 let sqrt_ids = sqrt(ids)
55 let sqrt_slope = deriv(sqrt_ids)
56
57 plot sqrt_ids vs vgs
58
59 meas dc vth_probe_sqrt_slope find sqrt_slope when vgs = vth_probe_vgs
60 meas dc vth_probe_sqrt_ids find sqrt_ids when vgs = vth_probe_vgs
61
62 let vth = vth_probe_vgs - (vth_probe_sqrt_ids / vth_probe_sqrt_slope)
63 print vth
64
65 plot ids vs vgs
66
67 let slope = deriv(ids)
68 meas dc gm find slope when vgs = gm_probe_vgs
69
70 let k = (gm * 1) / (w * (gm_probe_vgs - vth))
71
72 * ----- Ids vs Vds @ Vgs = 0.6V -----
73
74 alter vgs dc 0.6
75 dc vds 0 1.8 0.01
76
77 let ids = -i(vds)
78 let vds = v(d)
79
80 plot ids vs vds
81
82 * ----- Ids vs Vds @ Vgs = 0.8V -----
83
84 alter vgs dc 0.8
85 dc vds 0 1.8 0.01
86
87 let ids = -i(vds)
88 let vds = v(d)
89
90 plot ids vs vds
91
92 * ----- Ids vs Vds @ Vgs = 1V -----
93
94 alter vgs dc 1
95 dc vds 0 1.8 0.01
96
97 let ids = -i(vds)
98 let vds = v(d)
99
100 plot ids vs vds
101
102 let slope = deriv(ids)
103
104 meas dc gout find slope when ids = gout_probe_ids
105 meas dc gout_probe_vds find vds when ids = gout_probe_ids
106
107 let rout = 1 / gout
108 print rout
109
110 let lambda = 1 / ((rout * gout_probe_ids) - gout_probe_vds)

```

```

111 print lambda
112
113 let k = k / (1 + lambda)
114 print k
115
116 * ----- Ids vs Vgs @ Vds = 0.2V, 0.6V, 1V -----
117
118 dc vgs 0 1.8 0.01 vds 0.2 1 0.4
119
120 let ids = -i(vds)
121 let vgs = v(g)
122
123 plot ids vs vgs
124
125 * ----- Ids vs Vds @ Vgs = 0.6V, 0.8V, 1V -----
126
127 dc vds 0 1.8 0.01 vgs 0.6 1 0.2
128
129 let ids = -i(vds)
130 let vds = v(d)
131
132 plot ids vs vds
133
134 .endc
135
136 .end

```

## 4.2 PMOS

Listing 2: Ngspice netlist for PMOS characterization

```

1 .title PMOS Characterization
2
3 .include models-180nm
4
5 vdd vdd 0 dc 1.8
6 vsd vdd d dc 0.2
7 vsg vdd g dc 0
8
9 mp d g vdd vdd CMOSP W=7u L=0.18u
10
11 .control
12
13 run
14
15 set color0 = white
16 set color1 = black
17 set color2 = red
18
19 let gm_probe_vsg = 0.9
20 let gout_probe_isd = 400u
21 let vth_probe_vsg = 0.7
22
23 let k = 0
24 let w = 7u
25 let l = 0.18u
26

```

```

27 * ----- Isd vs Vsg @ Vsd = 0.2V -----
28
29 alter vsd dc 0.2
30 dc vsg 0 1.8 0.01
31
32 let isd = -i(vsd)
33 let vsg = v(vdd) - v(g)
34
35 plot isd vs vsg
36
37 * ----- Isd vs Vsg @ Vsd = 0.6V -----
38
39 alter vsd dc 0.6
40 dc vsg 0 1.8 0.01
41
42 let isd = -i(vsd)
43 let vsg = v(vdd) - v(g)
44
45 plot isd vs vsg
46
47 * ----- Isd vs Vsg @ Vsd = 1V -----
48
49 alter vsd dc 1
50 dc vsg 0 1.8 0.01
51
52 let isd = -i(vsd)
53 let vsg = v(vdd) - v(g)
54
55 let sqrt_isd = sqrt(isd)
56 let sqrt_slope = deriv(sqrt_isd)
57
58 plot sqrt_isd vs vsg
59
60 meas dc vth_probe_sqrt_slope find sqrt_slope when vsg = vth_probe_vsg
61 meas dc vth_probe_sqrt_isd find sqrt_isd when vsg = vth_probe_vsg
62
63 let vth = vth_probe_vsg - (vth_probe_sqrt_isd / vth_probe_sqrt_slope)
64 print vth
65
66 plot isd vs vsg
67
68 let slope = deriv(isd)
69 meas dc gm find slope when vsg = gm_probe_vsg
70
71 let k = (gm * 1) / (w * (gm_probe_vsg - vth))
72
73 * ----- Isd vs Vsd @ Vsg = 0.6V -----
74
75 alter vsg dc 0.6
76 dc vsd 0 1.8 0.01
77
78 let isd = -i(vsd)
79 let vsd = v(vdd) - v(d)
80
81 plot isd vs vsd
82
83 * ----- Isd vs Vsd @ Vsg = 0.8V -----
84

```

```

85 alter vsg dc 0.8
86 dc vsd 0 1.8 0.01
87
88 let isd = -i(vsd)
89 let vsd = v(vdd) - v(d)
90
91 plot isd vs vsd
92
93 * ----- Isd vs Vsd @ Vsg = 1V -----
94
95 alter vsg dc 1
96 dc vsd 0 1.8 0.01
97
98 let isd = -i(vsd)
99 let vsd = v(vdd) - v(d)
100
101 plot isd vs vsd
102
103 let slope = deriv(isd)
104
105 meas dc gout find slope when isd = gout_probe_isd
106 meas dc gout_probe_vsd find vsd when isd = gout_probe_isd
107
108 let rout = 1 / gout
109 print rout
110
111 let lambda = 1 / ((rout * gout_probe_isd) - gout_probe_vsd)
112 print lambda
113
114 let k = k / (1 + lambda)
115 print k
116
117 * ----- Isd vs Vsg @ Vsd = 0.2V, 0.6V, 1V -----
118
119 dc vsg 0 1.8 0.01 vsd 0.2 1 0.4
120
121 let isd = -i(vsd)
122 let vsg = v(vdd) - v(g)
123
124 plot isd vs vsg
125
126 * ----- Isd vs Vsd @ Vsg = 0.6V, 0.8V, 1V -----
127
128 dc vsd 0 1.8 0.01 vsg 0.6 1 0.2
129
130 let isd = -i(vsd)
131 let vsd = v(vdd) - v(d)
132
133 plot isd vs vsd
134
135 .endc
136
137 .end

```

### 4.3 Inverter

Listing 3: Ngspice netlist for Inverter analysis

```
1 .title Inverter
2 .include models-180nm
3
4 vdd vdd 0 dc 1.8
5 vsin vin vbias sin(0 10m 200k)
6 vbias vbias 0 dc 0.9
7
8 mp vout vin vdd vdd CMOSP W=7u L=0.18u
9 mn vout vin 0 0 CMOSN W=2u L=0.18u
10
11 cl vout 0 250f
12
13 .control
14 run
15 set color0 = white
16 set color1 = black
17 set color2 = red
18
19 * ----- Transient Analysis -----
20
21 tran 10ns 40us
22
23 plot v(vout) v(vin)
24
25 meas tran vout_max max v(vout)
26 meas tran vout_min min v(vout)
27
28 let vpp_out = vout_max - vout_min
29 print vpp_out
30
31 meas tran vin_max max v(vin)
32 meas tran vin_min min v(vin)
33
34 let vpp_in = vin_max - vin_min
35 print vpp_in
36
37 let gain_transient = vpp_out / vpp_in
38 print gain_transient
39
40 * ----- DC Analysis -----
41
42 alter vsin dc 0
43 dc vbias 0 1.8 0.01
44
45 plot v(vout) vs v(vbias)
46
47 let gain = deriv(v(vout))
48 plot gain vs v(vbias)
49
50 meas dc gain_at_vbias_0v9 find gain when v(vbias) = 0.9
51
52 .endc
53
54 .end
```