وَمَا أُوتِيتُمْ مِنَ الْعِلْمِ إِلَّا قَلِيلًا

Analog IC Design – Xschem/Ngspice
Lab 07
Gm/ID Design Methodology

Intended Learning Objectives

In this lab you will:

- Design and simulate a 5T OTA.
- Learn how to plot and use gm/ID design charts using ADT.
- Learn how to simulate open-loop/closed-loop characteristics of the 5T OTA using open-source tools.

PART 1: gm/ID design charts

Using ADT Device Xplore, plot the following design charts vs gm/ID for both PMOS and NMOS. Set VDS = VDD/3 and L = 0.33u,0.4u:0.4u:2u:

(1) Gm/gds
(2) ID/W
(3) Gm/cgg (use Y expression)
(4) VGS

Hints:

(1) Change the directory using the command **"cd /home/tare/ADT_Working_Directory"**
(2) Run ADT in the terminal using the command **"./start_adt.sh"**
(3) Load nmos_03v3.lut and pmos_03v3.lut from the directory **"/home/tare/LUTs"**
(4) Go to sizing assistant and set the LUT settings (VDD = 1.8), leave gm/ID empty to become the x-axis.

Part 2: OTA Design

Use gm/ID methodology to design a diff input SE output operational transconductance amplifier (OTA) that achieves the following specs. Use an ideal external 10uA DC current source in your test bench (not included in the OTA current consumption spec), but design your own current mirror.

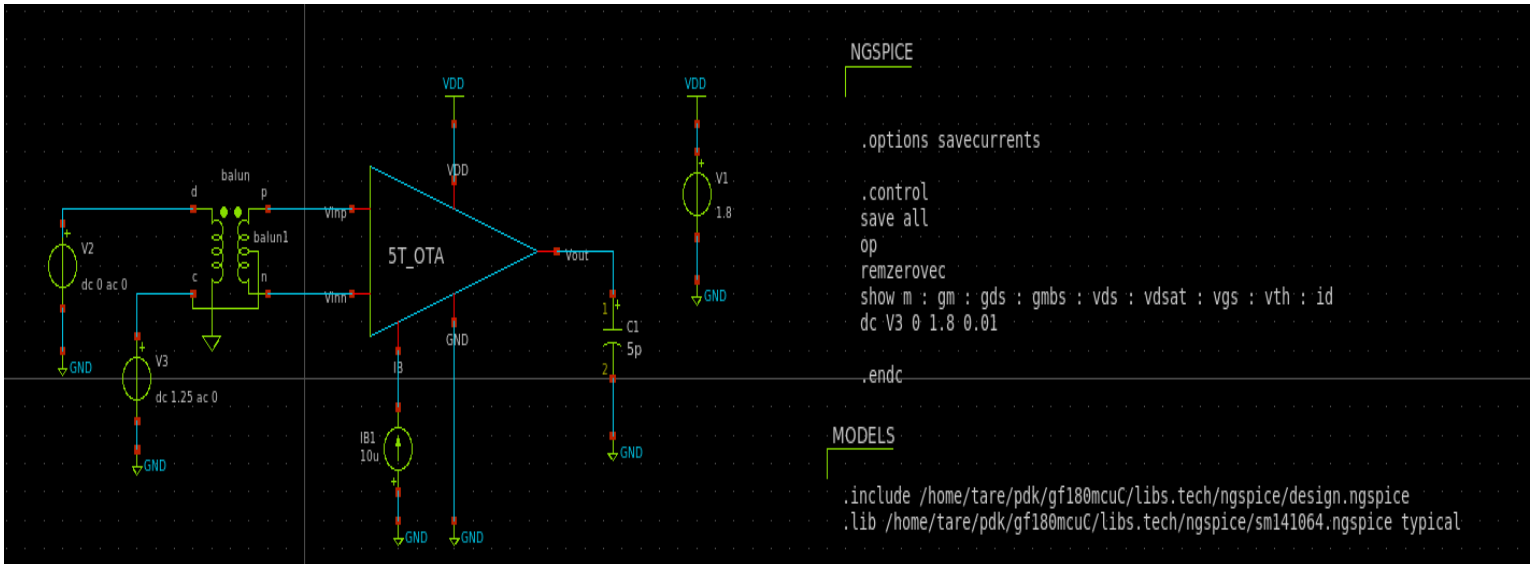
Technology	0.13 um CMOS	0.18 um CMOS
Supply Voltage	1.2V	1.8V
Load	5pF	5pF
Open Loop DC Voltage Gain	>= 34dB	>= 34dB
CMRR @ DC	>= 74dB	>= 74dB
Phase Margin	>= 70°	>= 70°
CM input range - low	<= 0.6V	<= 1V
CM input range - high	>= 1V	>= 1.5V
GBW	>= 10MHz	>= 10MHz

Report the following:

(1) Detailed design procedure and hand analysis. You need to make reasonable assumptions using g_m/I_d methodology. You need to explain why you chose the architecture that you implemented.

(2) A table showing W, L, g_m, I_d , g_m/I_d , V_{dsat}, V_{ov}= V_{gs}−V_{th} , and V^*=2*g_m/I_d of all transistors (as calculated from gm/ID curves).

Part 3: Open – loop OTA simulation[1]



Create a testbench as shown above. Note that IDC connection (sinking or sourcing) in the test bench may be different from the one shown above depending on the type of your input pair (PMOS/NMOS). Report the following:

(1) Schematic of the OTA showing sizing of the transistors:

- Use VICM at the middle of the CMIR and show the operating points.
- Is the current (and gm) in the input pair exactly equal?
- What is DC voltage at VOUT? Why?

(2) Diff small signal ccs:

- Use AC analysis (1Hz:10GHz, logarithmic, 10 points/decade).
- Set VIDAC = 1 and VICMAC = 0.
- Use VICM at the middle of the CMIR.

---

[1] To avoid the change in the operating points from run to another run,you need to disable Monte carlo simulation as follows:

(1) Change the directory in the terminal using the command
"cd /home/tare/pdk/gf180mcuC/libs.tech/ngspice"

(2) Open the file **"design.ngspice"**

(3) Make sure that you disabled mismatch simulation as shown in the figure

```
********** Default mc switches **********
**
.param
+  sw_stat_global = 1
+  sw_stat_mismatch = 0
**
********* Default mc skew value *********
```

➔ **Ngspice Hint:**

```
.options savecurrents
.control
  op
  save all
  ac dec 10 1 10G
  run
  meas ac peak MAX vmag(Avd) FROM=2 TO=1.5GHZ
  let f3db = peak/sqrt(2)
  meas ac f1 WHEN vmag(Avd)=f3db RISE=1
  meas ac f2 WHEN vmag(Avd)=f3db FALL=1
  let GBW = peak*f2
  print peak
  print f2
  print GBW
  remzerovec
  show m : gm : gmbs : gds : vds : vdsat : vgs : vth : id
  write OTA_TB.raw
.endc
```

- Plot diff gain (in dB) vs frequency.
- Compare simulation results with hand calculations in a table.

(3) CM small signal ccs:

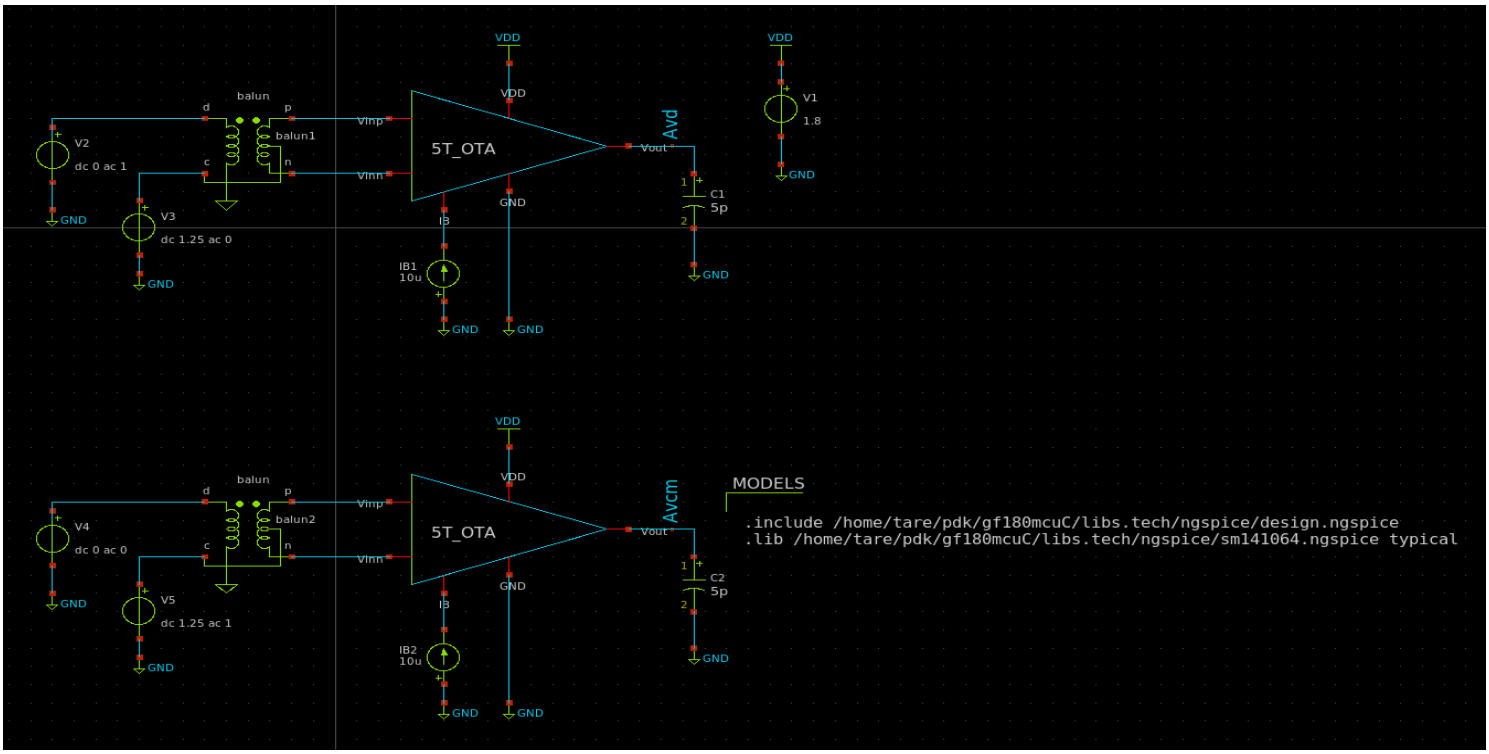
- Use AC analysis (1Hz:10GHz, logarithmic, 10 points/decade).
- Set VICMAC = 1 and VIDAC = 0.
- Use VICM at the middle of the CMIR.
- Plot CM gain in dB vs frequency.
- Compare simulation results with hand calculations in a table.

(4) CMRR:

- Use VICM at the middle of the CMIR.
- Plot CMRR in dB vs frequency at VICM at the middle of the CMIR.
- Compare simulation results with hand calculations in a table.

➔ Xschem Hint:

- You will need to repeat the schematic in the same testbench but output label should be different.



➔ Ngspice Hint:

```
.options savecurrents
.control
   save all
   ac dec 10 1 10G
   remzerovec
   let CMRR = Avd/Avcm
  plot db(CMRR)
  write OTA_TB.raw
.endc
```

(5) Diff large signal ccs:

- Use VICM at the middle of the CMIR.
- Use DC sweep (**not parametric sweep**) VID = -VDD:1m: VDD. You must use a small step (1mV) because the gain region is very small (steep slope).
- Plot VOUT vs VID.
- From the plot, what is the value of Vout at VID = 0? Why?
- Plot the derivative of VOUT vs VID. Compare the peak with Avd.

➔ **Ngspice Hint:**

- To Run dc sweep and print the derivative of the output signal, you can use the following code.
- V2 in the code is the name of the differential voltage source.

```
.options savecurrents
.control
   op
   save all
   dc V2 -1.8 1.8 1m
   remzerovec
   let derivative = abs(deriv(Avd))
   **plot abs(deriv(Avd))
  write OTA_TB.raw
.endc
```

(6) (Optional) CM large signal ccs (GBW vs VICM):

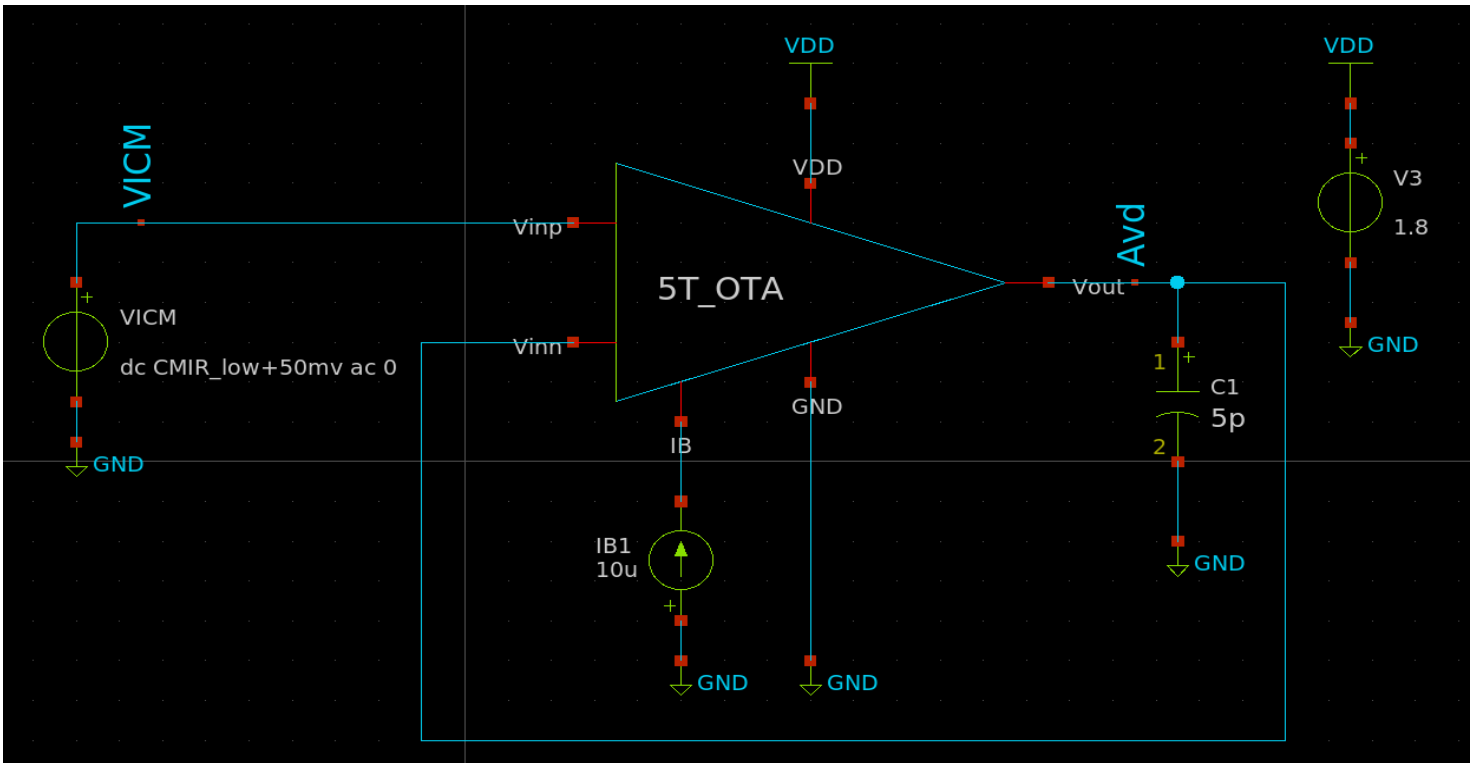
- Use AC analysis (1Hz, 1 point only).
- Set VIDAC = 1 and VICMAC = 0.
- Use **parametric** sweep (**not DC sweep**) VICM = 0:10m:VDD.

➔ **Ngspice Hint:**

```
.control
let Vstart=0
let Vstop=1.8
let Vstep=10m
let Vact=Vstart
let VICMmax=0
let VICMmin =0
let gbwmin = 0
save all
ac dec 10 1 100G
let vx = Avd
meas ac DiffGain MAX vmag(vx) FROM=1 TO=10G
let ff1=DiffGain*0.707
meas ac f3db WHEN vmag(vx) = ff1 FALL =1
let GBWmin = 0.9*DiffGain*f3db
save gbwmin
while Vact le Vstop
alter VICM Vact
ac dec 10 1 100G
let vx = Avd
meas ac DiffGain MAX vmag(vx) FROM=1 TO=100MEG
let ff1=DiffGain*0.707
meas ac f3db WHEN vmag(vx) = ff1 FALL =1
let GBW = DiffGain*f3db
save GBW
let vx = Vact
let VICMact = Vact + VICM
save VICMact
set appendwrite
remzerovec
write OTA.raw
if (GBW > GBWmin)
let VICMmax = Vact
endif
if (GBW < GBWmin)
if (Vact<1)
let VICMmin = Vact
endif
endif
let Vact=Vact+Vstep
end
print VICMmax
print VICMmin
plot GBW
.endc
```

PART 4: Closed-Loop OTA Simulation

(1) Create schematic of the OTA showing current and g_m operating point. Use VINCM = $CMIR_{low} + 50\ mv$.



- Is the current (and gm) in the input pair exactly equal? Why?
- Calculate the mismatch in I_d and g_m.

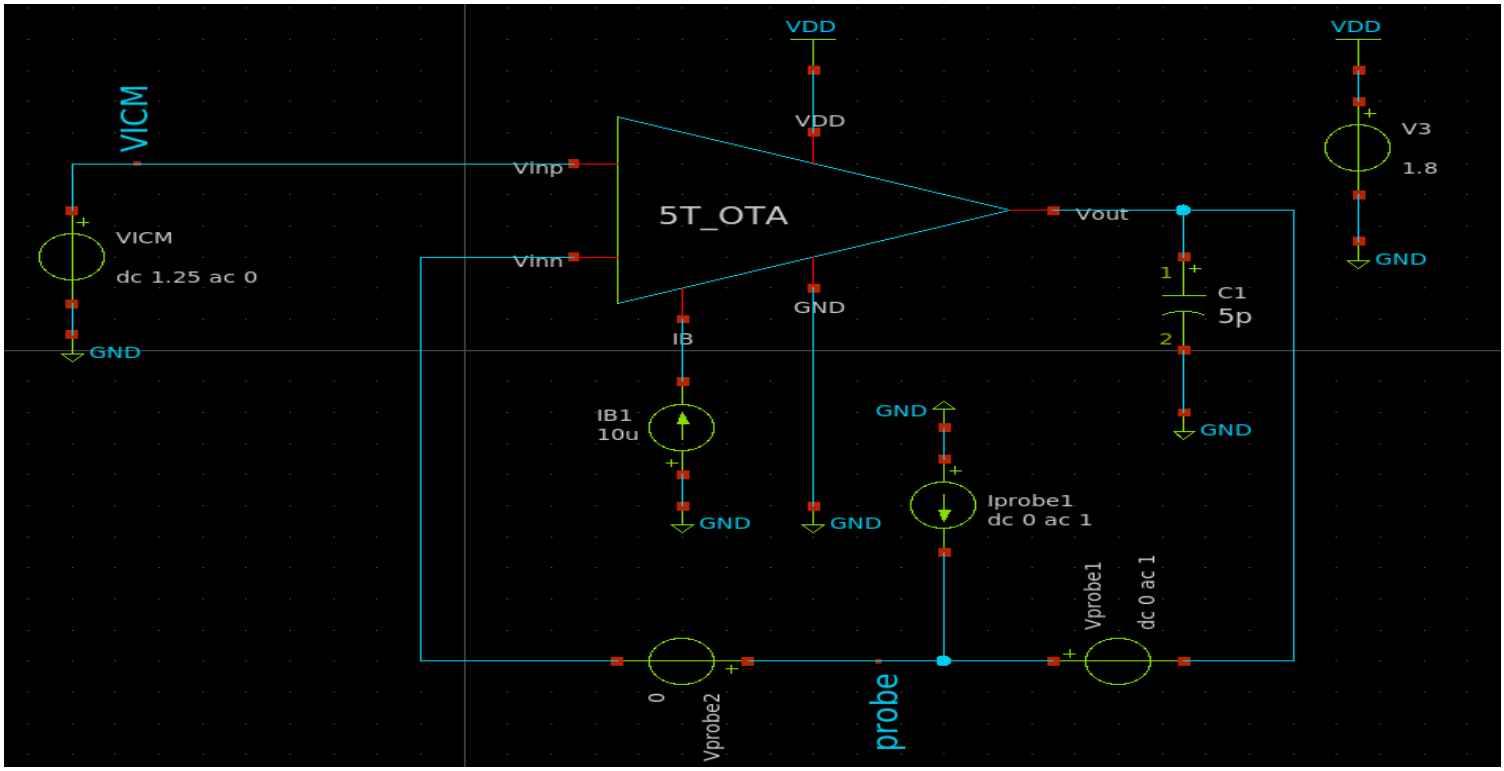
➔ **Ngspice Hint:**

```
.options savecurrents
.control
  op
  save all
  remzerovec
  show m : gm : gmbs : gds : vds : vdsat : vgs : vth : id
  let gm1 = @m.x1.xm7.m0[gm]
  let gm2 = @m.x1.xm8.m0[gm]
  let gm_mismatch = gm1 - gm2
  print gm_mismatch
  let Id1 = @m.x1.xm7.m0[id]
  let Id2 = @m.x1.xm8.m0[id]
  let Id_mismatch = Id1 - Id2
  print Id_mismatch
```

```
 write OTA_TB.raw
.endc
```

(2) Loop Gain:

- Put the OTA in a unity gain buffer configuration as shown in the schematic below.



- Use VICM at the middle of the CMIR.
- Plot loop gain in dB and phase vs frequency. Show the results in the console.
- Compare DC gain and GBW with those obtained from open-loop simulation. Comment.
- Show the operating point at VICM in the middle of the CMIR.
- Compare simulation results (DC gain and GBW) with hand calculations in a table.

➔ **Ngspice Hint:**

```
.options savecurrents
.control
  let runs=2
  let run=0

  alter @Vprobe1[acmag]=1
  alter @iprobe1[acmag]=0

  dowhile run < runs

  set run =”$&run”
  ac dec 20 0.01 10G
  write OTA_STB_TB_{$run}.raw

  all
  alter @Vprobe1[acmag]=0
  alter @iprobe1[acmag]=1
  let run = run + 1
  end

  let ip22 = ac2.i(Vprobe2)
  let vprb1 = ac1.probe
  let mb = 1/(vprb1+ip22)-1
  let phase_mb = 180/PI*vp(mb)
  plot vdb(mb)
  plot phase_mb
  plot vdb(mb) phase_mb

  echo "------------------------------"
  echo "------------------------------"

  meas ac peak MAX vmag(mb) FROM=2 TO=1.5GHZ
  let f3db = peak/sqrt(2)
  meas ac f1 WHEN vmag(mb)=f3db RISE=1
  meas ac BW WHEN vmag(mb)=f3db FALL=1
  let GBW = peak*BW
  meas ac pm_deg find phase_mb when vdb(mb)=0
  meas ac dominant_pole_f when vdb(mb)=0
  meas ac loop_gain find vdb(mb) at=0.01
  print GBW

  op
  show m : gm : gmbs : gds : vds : vdsat : vgs : vth : id
.endc
```