Advanced Compact MOSFET Model 2: Bridging design and simulation

03/07/2025

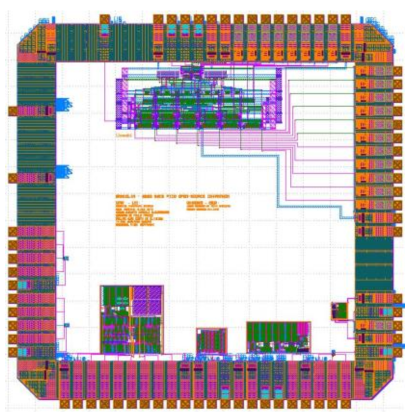
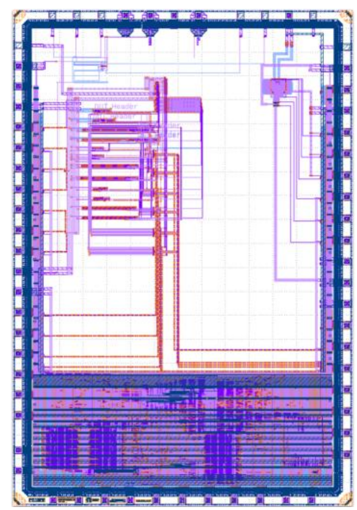
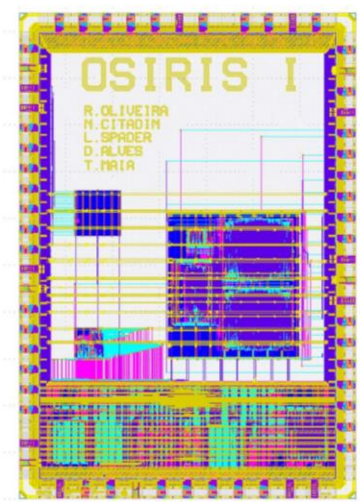
Deni Germano Alves Neto

denialves77@gmail.com

https://github.com/ACMmodel/MOSFET_model

About me

- Universidade Federal de Santa Catarina – UFSC - Brazil
- Undergrad and Masters in IC design - 2022
  - Subject : Ultra-Low-Voltage IC circuits $V_{DD} < 100\ mV$
    - Dissertation: Ultra-Low-Voltage Standard Cell Library
      - ACM for low voltage circuits
- Contact with open-source IC design :
  - Chipathon - SSCS 2021 : Analog-front-end for Biosignals – AFEbio
  - Chipathon-SSCS & UNIC-CASS 2023/2024 – Analog/Digital IC design
- Start PhD in 2023 : MOSFET Modeling – ACM2
- Joint PhD between UFSC and
- UGA - Grenoble INP
  - TIMA (Currently based)
  - RFIC design

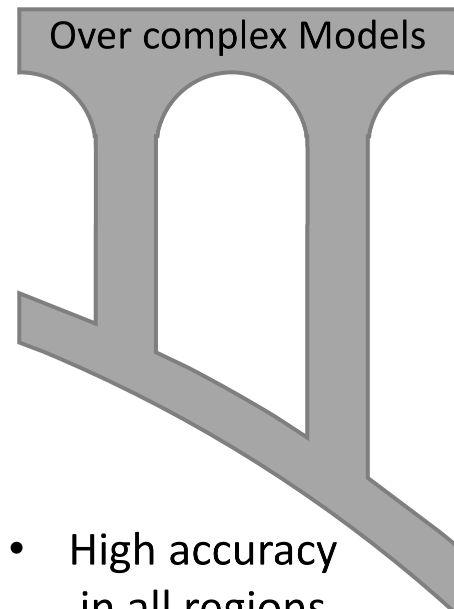
Outline

- Compact models
- ACM2 model
- Parameter Extraction
- DC characteristics
- Circuit examples

Why the need for a design-oriented MOSFET model ?

- Compact Model is the **medium** of information exchange between foundry and designer.
- Provides **detailed information** about device operation & characteristics.
- **Simple** enough to be incorporated in circuit simulators.
- **Accurate** enough to predict behavior of circuits.

- Poor accuracy, only in one region
- 2/3 DC parameters

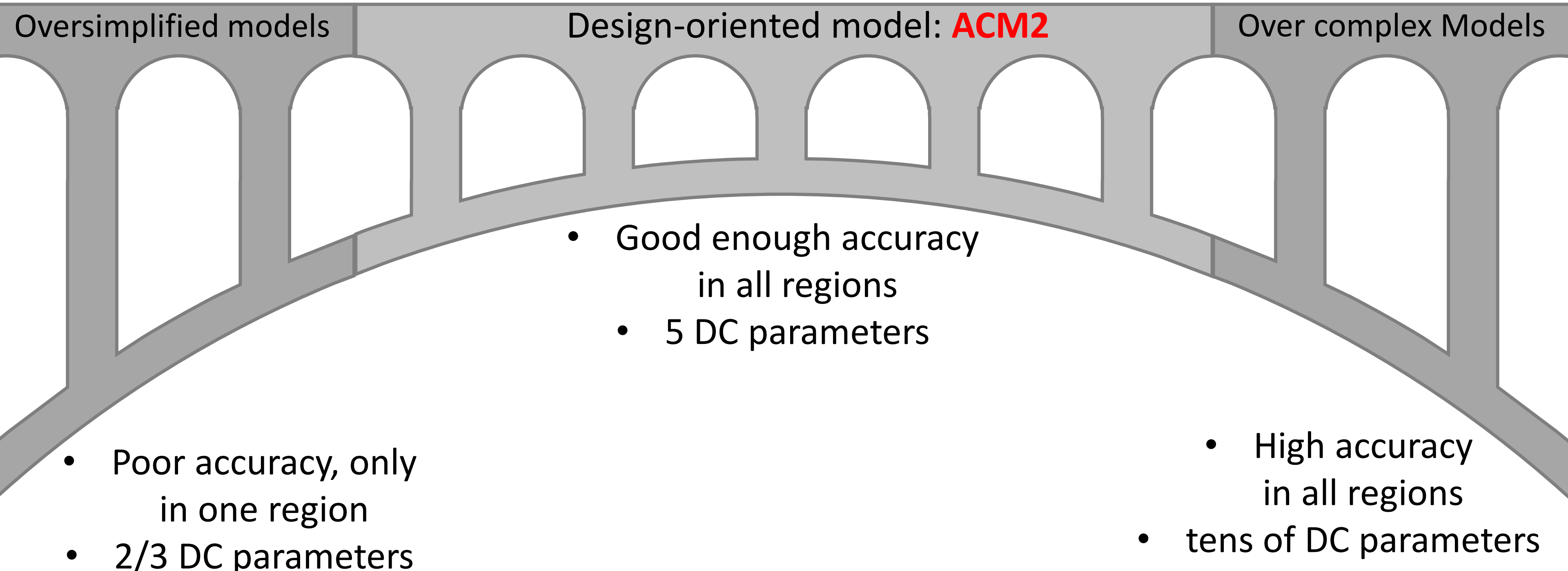


- High accuracy in all regions
- tens of DC parameters

Why the need for a design-oriented MOSFET model ?

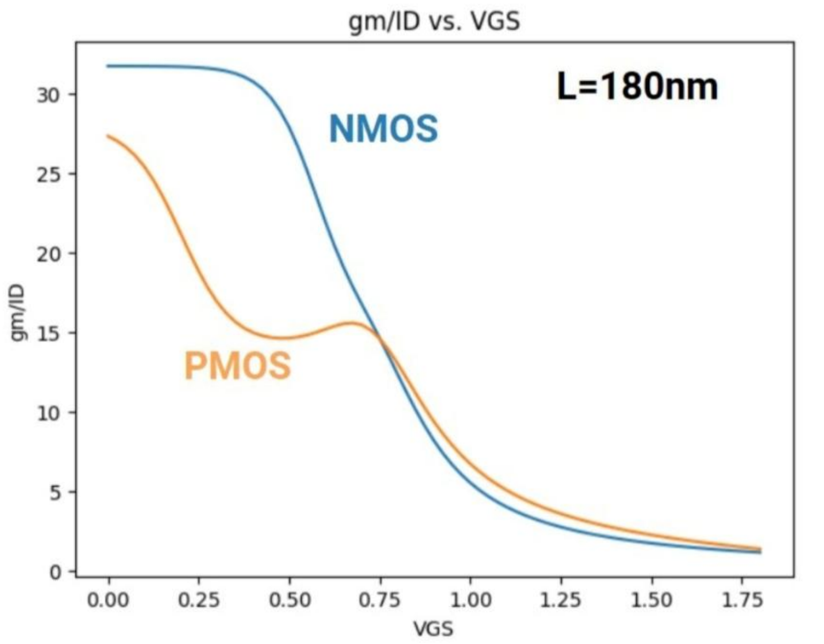
- Compact Model is the **medium** of information exchange between foundry and designer.
- Provides **detailed information** about device operation & characteristics.
- **Simple** enough to be incorporated in circuit simulators.
- **Accurate** enough to predict behavior of circuits.
- **Bridge the gap between design and simulation!**

IC designers bridge



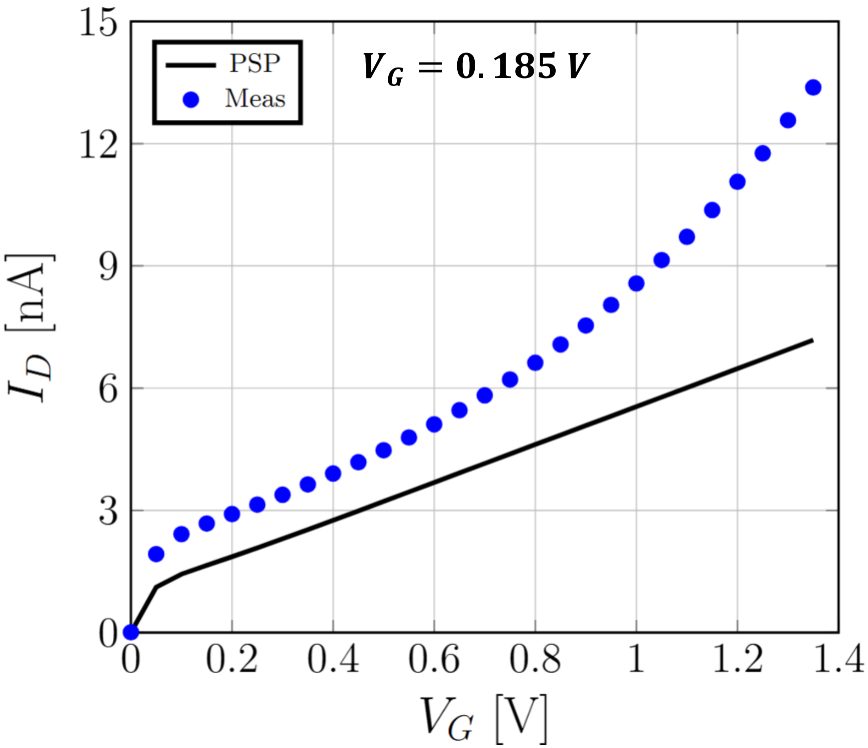
Problems in current PDK models

BSIM4 issues in sky130



Subthreshold region

PSP103 issues in IHP open PDK



Modeling problem in weak inversion : no DIBL effect

Open-Source IC design

Open-source PDK

Open-source EDA tools

Few open-source IP & libraries

Available compact MOSFET models

We propose:

ACM2 : A Simple 5-DC-parameter MOSFET model

ACM2: A simple 5-DC-parameter MOSFET model

Complete Continuous All-region charge-based MOSFET model

$$V_P = \frac{V_{GB} - V_{T0} + \sigma(V_{DB} + V_{SB})}{n}$$

$$\frac{V_P - V_{SB}}{\phi_t} = q_s - 1 + \ln(q_s)$$

- Used to calculate q_s
- Bridge between **WI** and **SI** regions

$$q_{dsat} = q_s + 1 + \frac{1}{\zeta} - \sqrt{\left(1 + \frac{1}{\zeta}\right)^2 + \frac{2q_s}{\zeta}}$$

$$\frac{V_{DS}}{\phi_t} = q_s - q_d + \ln\left(\frac{q_s - q_{dsat}}{q_d - q_{dsat}}\right)$$

- Used to calculate q_d
- Bridge between **Triode** and **Saturation** regions

$$I_D = I_S \frac{(q_s + q_d + 2)}{1 + \zeta(q_s - q_d)}(q_s - q_d)$$

Specific current I_S ***(W,L)***	Threshold voltage V_{T0} ***(W,L)***	Slope factor n ***(W,L)***	DIBL factor σ ***(W,L)***	V_{sat} effect ζ ***(W,L)***

Oversimplified model vs ACM model @ Saturation

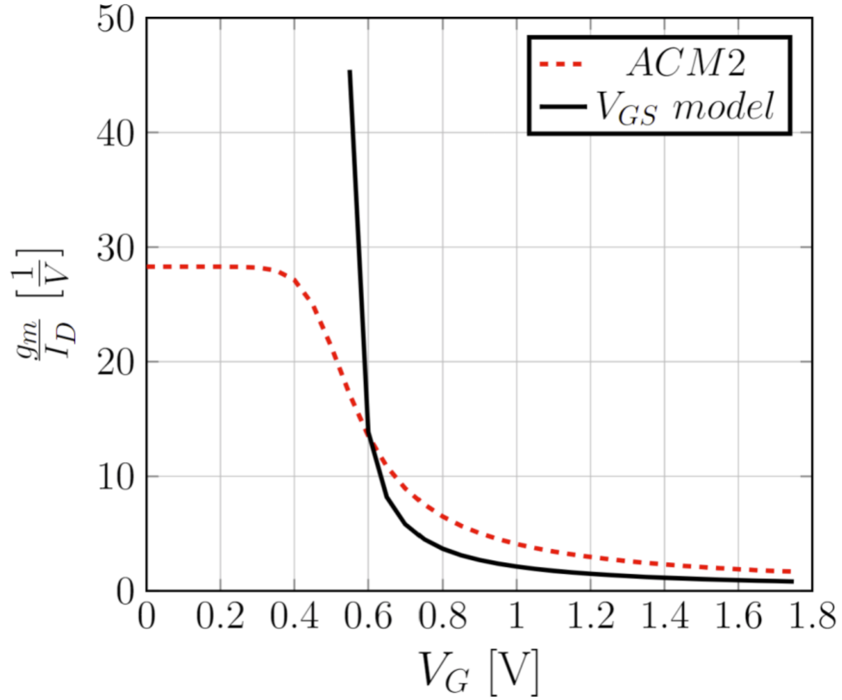
Unified Charge Control Model

$$I_D = \frac{\beta}{2}(V_{GS} - V_T)^2$$

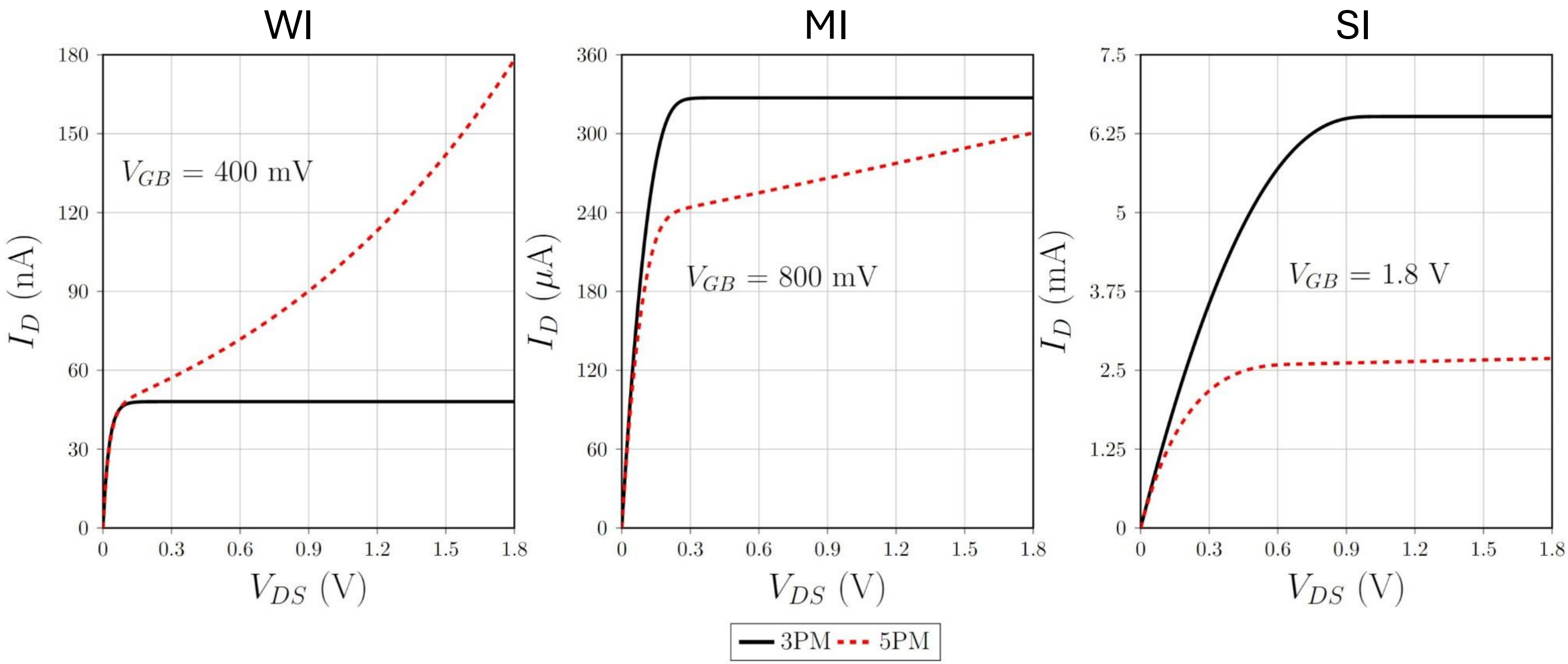
$$V_P = \frac{V_{GB} - \boldsymbol{V_{T0}}}{\boldsymbol{n}} \qquad \frac{V_P - V_{SB}}{\phi_t} = q_s - 1 + \ln q_s$$

$$\frac{g_m}{I_D} = \frac{1}{V_{GS} - V_T}$$

$$I_D = \boldsymbol{I_S}(q_s^2 + 2q_s) \qquad \frac{g_m}{I_D} = \frac{2}{n\phi_t(1 + q_s)}$$



Output characteristics including DIBL and v_{sat}



$$I_D = \boldsymbol{I_S} \frac{(q_s + q_d + 2)}{1 + \boldsymbol{\zeta}(q_s - q_d)}(q_s - q_d)$$

DIBL model: $V_T = V_{T0} - \sigma(V_{SB} + V_{DB})$

$$\zeta = \frac{(\mu_s \phi_t / L)}{v_{sat}}$$

Transistor	$W/L\ [\mu m]$	$V_{T0}\ [mV]$	$I_S\ [\mu A]$	n	σ	ζ
NMOS	**5/0.18**	528	5.52	1.37	0.025	0.056

Outline

- Compact models
- ACM2 model
- Parameter Extraction
- DC characteristics
- Circuit examples

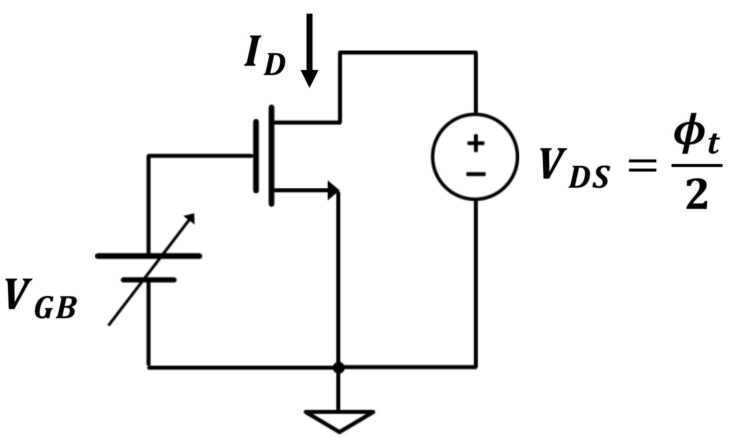
I_S, V_{T0} and n extraction

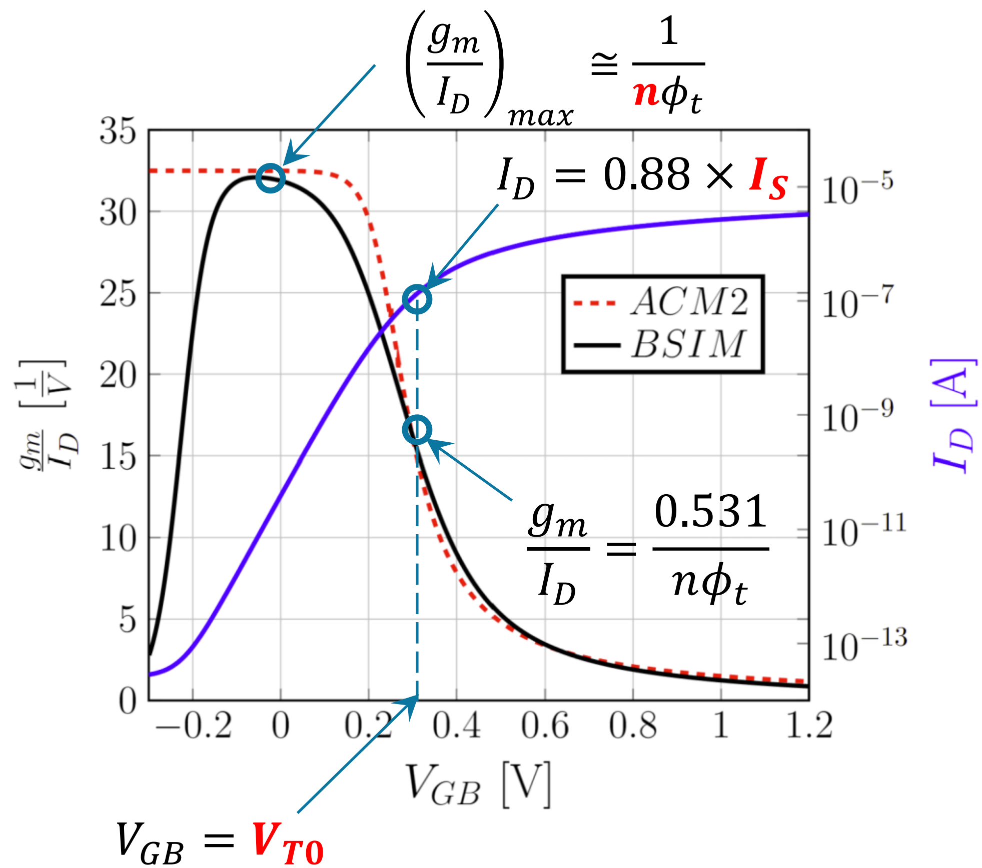
The g_m/I_D method

For $q_S = 1$, $V_{GB} = V_{T0}$ and g_m/I_D is at ½ of its maximum value

$$\frac{V_{GB} - V_{T0}}{n} = q_s - 1 + \ln q_s$$

$$\left.\frac{g_m}{I_D}\right|_{V_{DS}\to 0} = \frac{1}{n\phi_t(1+q_S)}$$

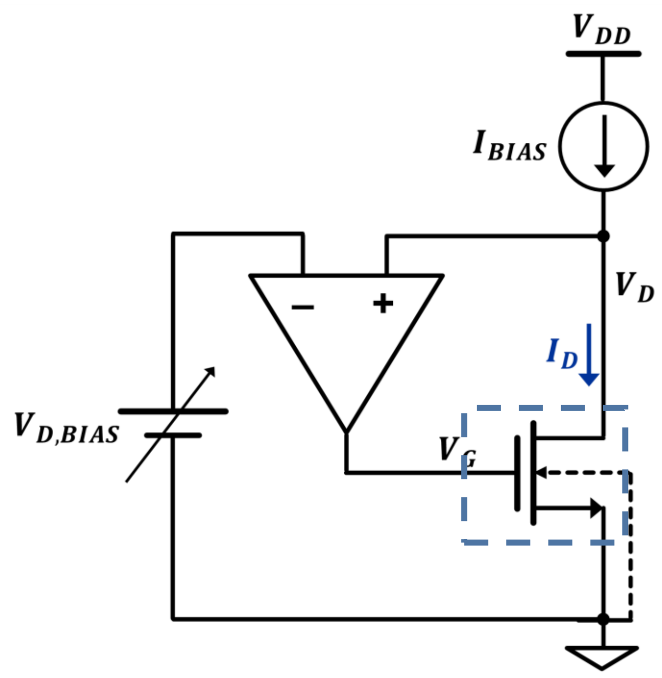


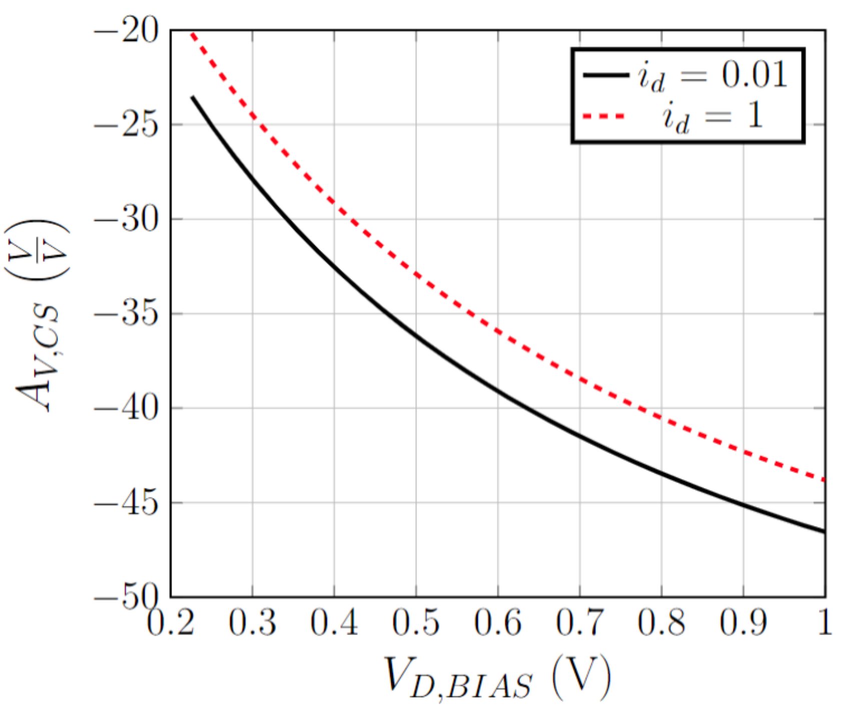


Extraction of σ

Common-Source Intrinsic-Gain method

$$A_{V,CS} = -\frac{g_{msat}}{g_{dssat}} = -\frac{1}{\sigma}$$

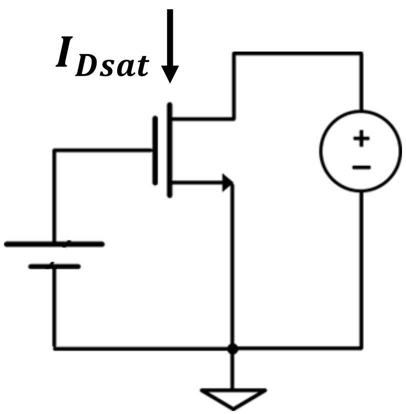


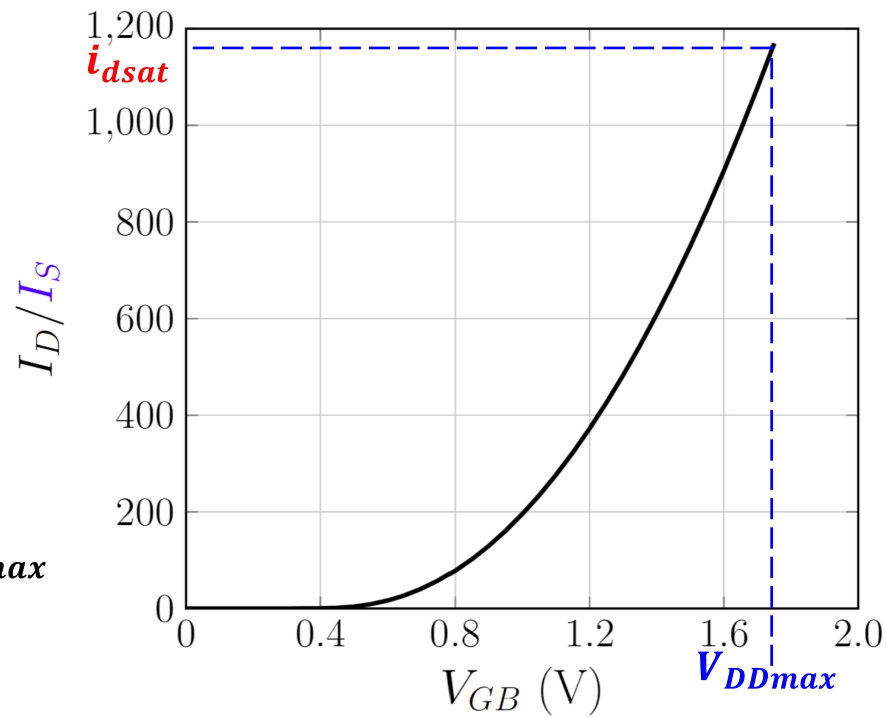


ζ extraction

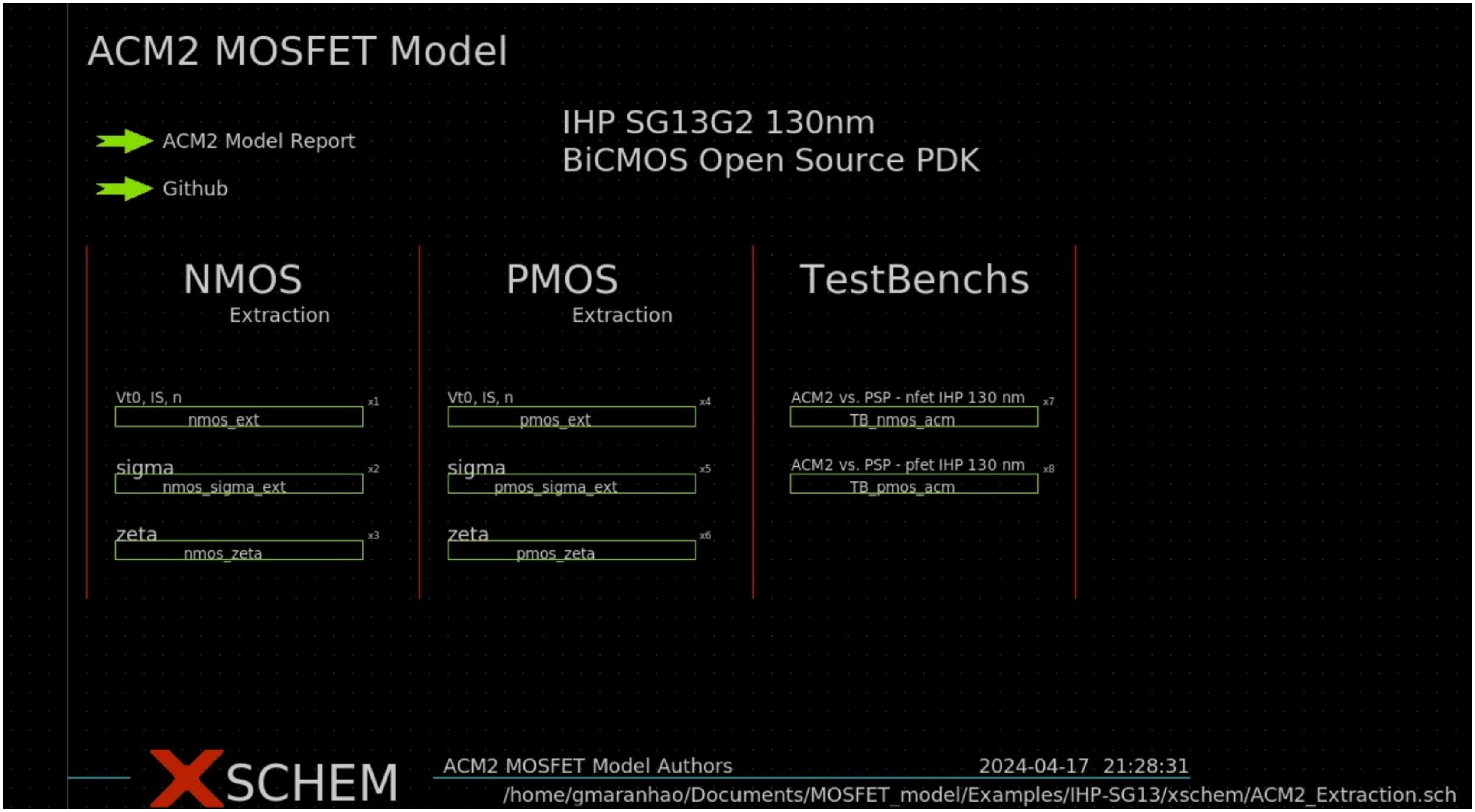
$$\zeta = \frac{2\left(q_s + 1 - \sqrt{1 + i_{dsat}}\right)}{i_{dsat}}$$

- q_s calculated from UCCM(V_{T0}, n, σ).
- Measure $i_{dsat} = I_{Dsat}/I_S$.





Automatic parameter extraction – IHP @ Xschem

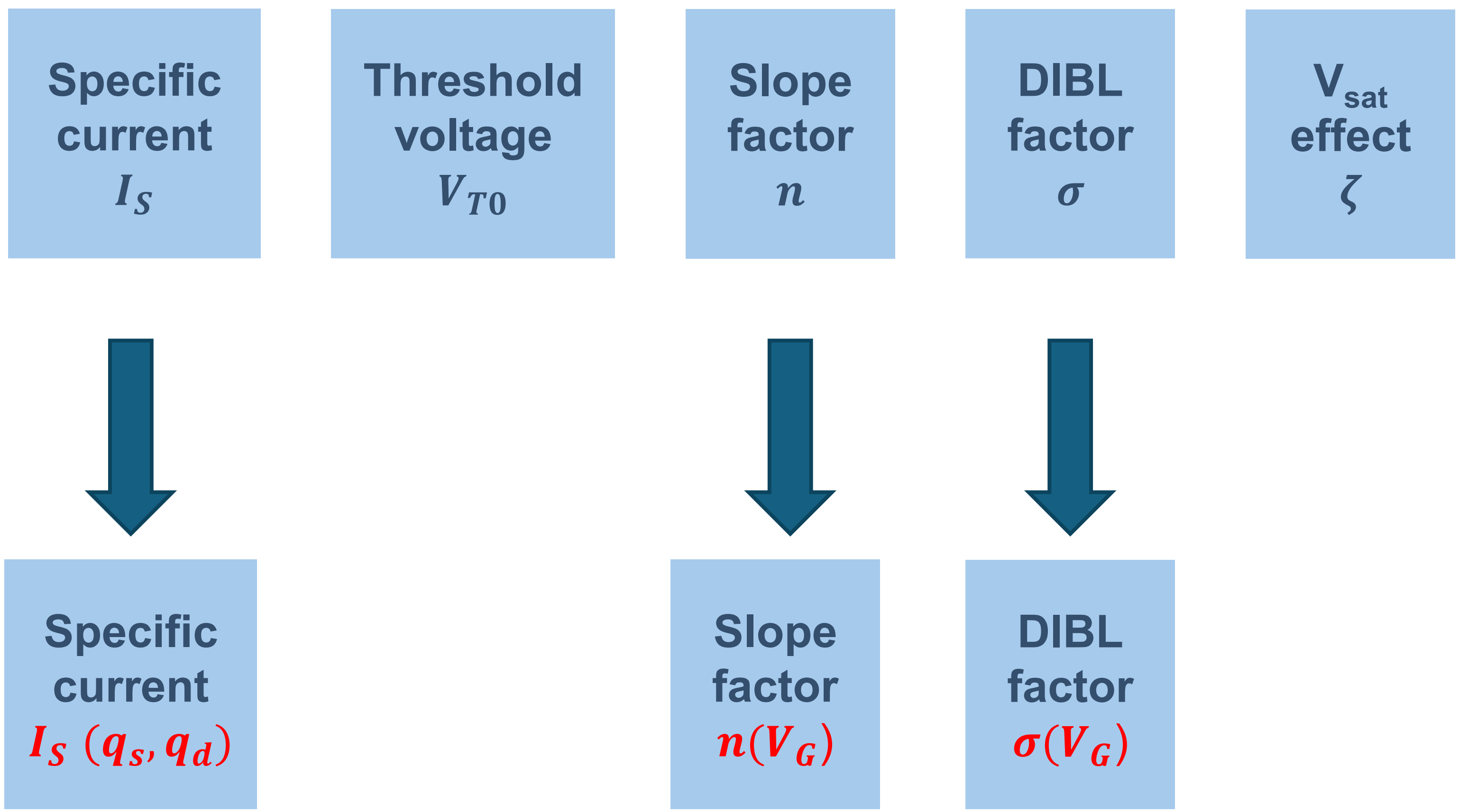


Also, available for GF180 and SKY130

Outline

- Compact models
- ACM2 model
- Parameter Extraction
- DC characteristics
- Circuit examples

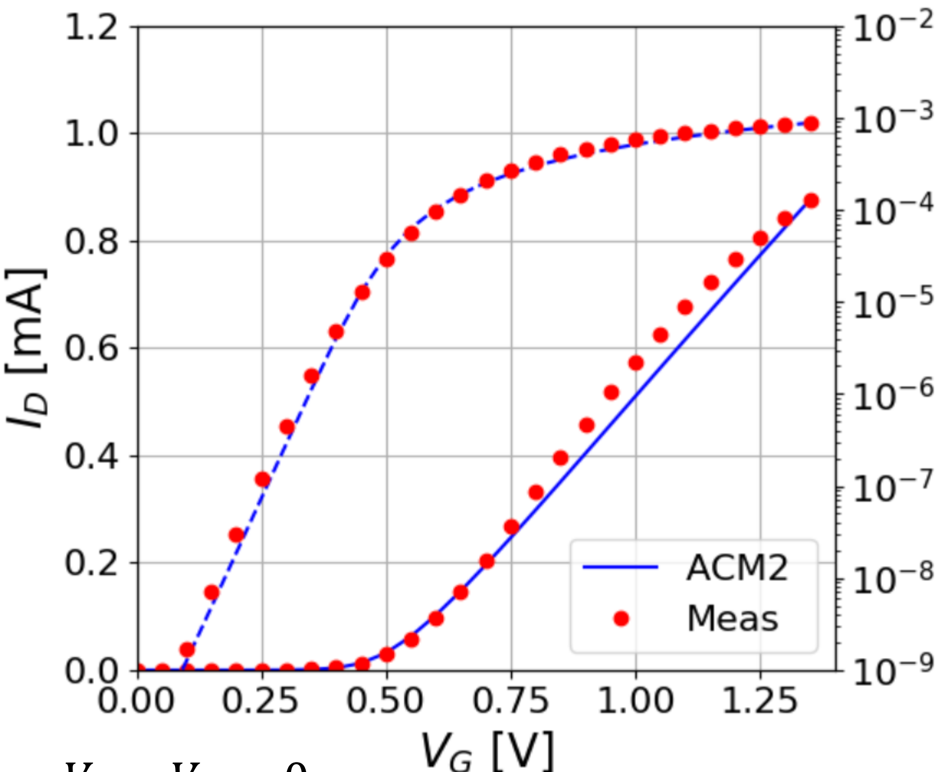
ACM2: A simple 5-DC-parameter MOSFET model



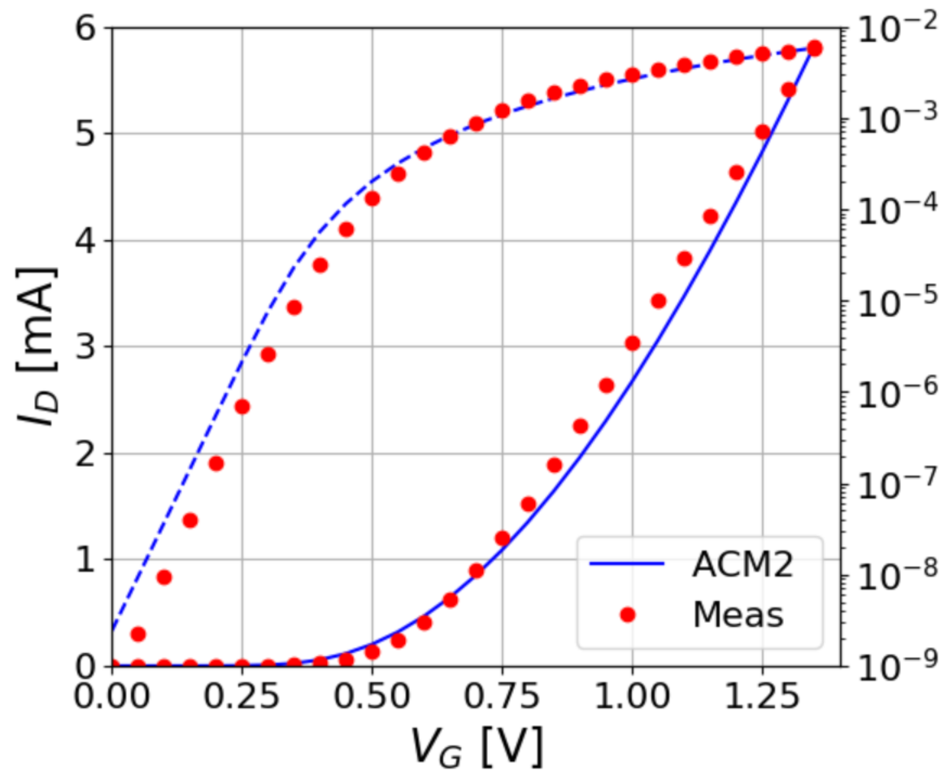
ACM2[1] vs meas – 130 nm SiGe IHP[2]

I_D vs V_{GB}

$V_{DS} = 50\ \text{mV}$



$V_{DS} = 1.2\ \text{V}$



$V_S = V_B = 0$

Characteristics of a LVT NMOS bulk transistor with W /L = 10µm/ 120 nm.

[1] ACM2 : implemented in verilog-A, compiled by OPENVAF, simulated in Ngspice

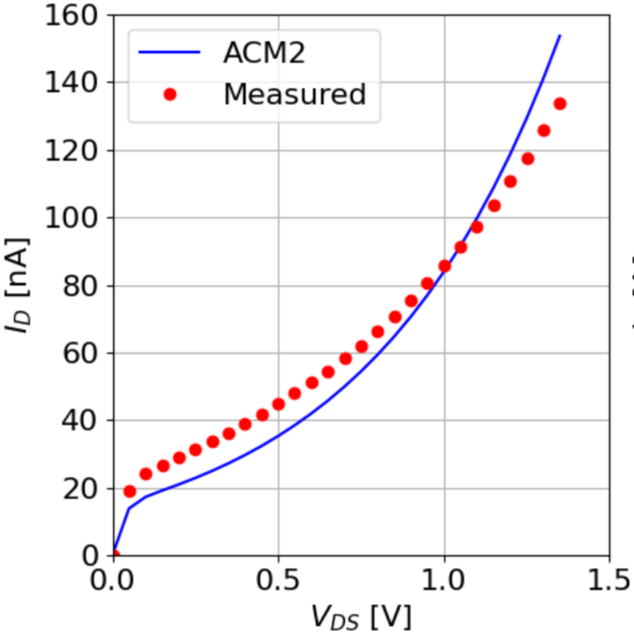
[2] Institut for High-Performance Microelectronics (IHP) open-source PDK

ACM2[1] vs meas – 130 nm SiGe IHP[2]

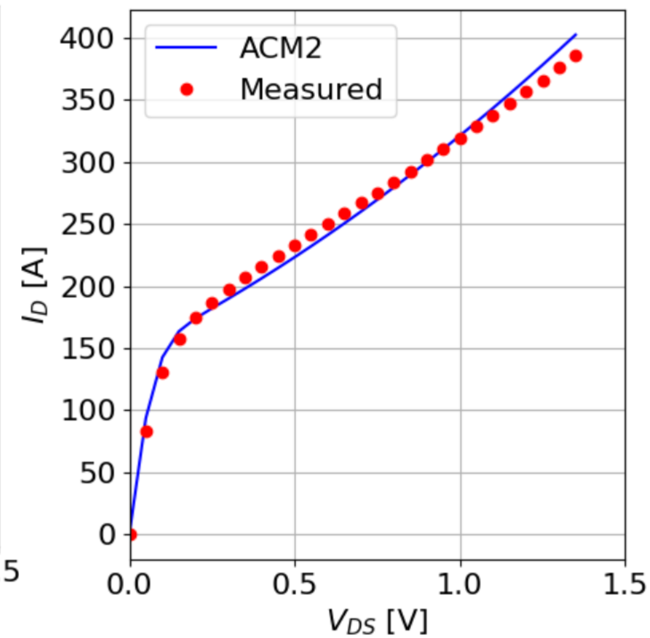
$V_S = V_B = 0$

I_D vs V_{DS}

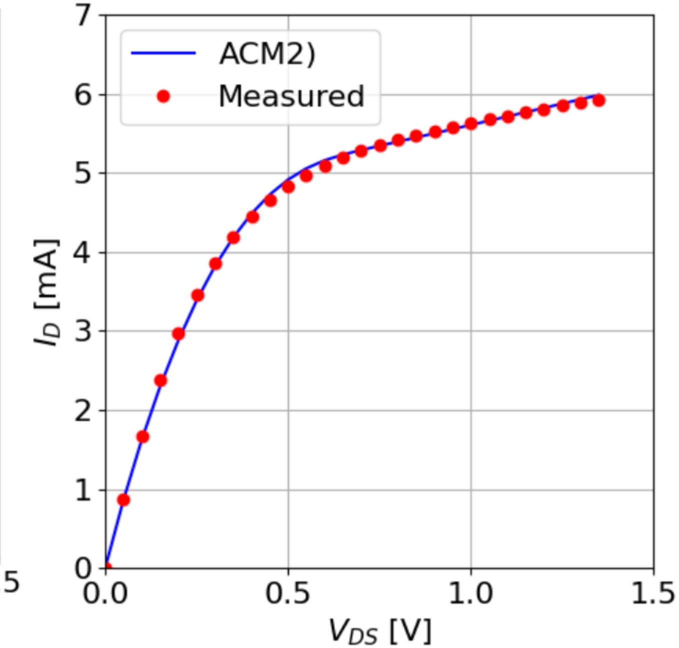
V_{GB} =185mV



V_{GB} = 585 mV



V_{GB} = 1.35 V



Characteristics of a LVT NMOS bulk transistor with W /L = 10µm/ 120 nm.

	PSP103	BSIM4	EKV2.6	ACM2
OpenVAF	3.48s	6.7s	0.22s	0.15s

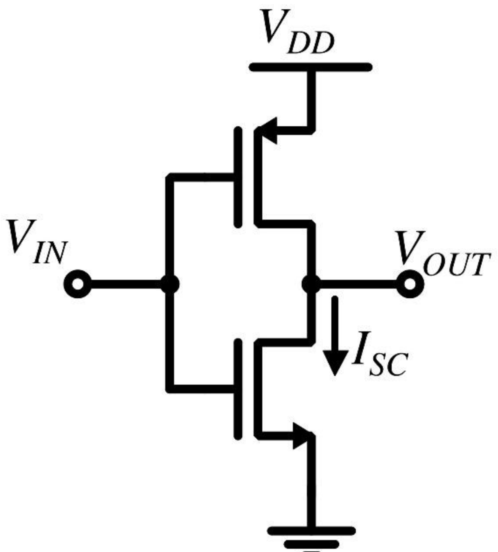
[1] ACM2 : implemented in verilog-A, compiled by OPENVAF, simulated in Ngspice

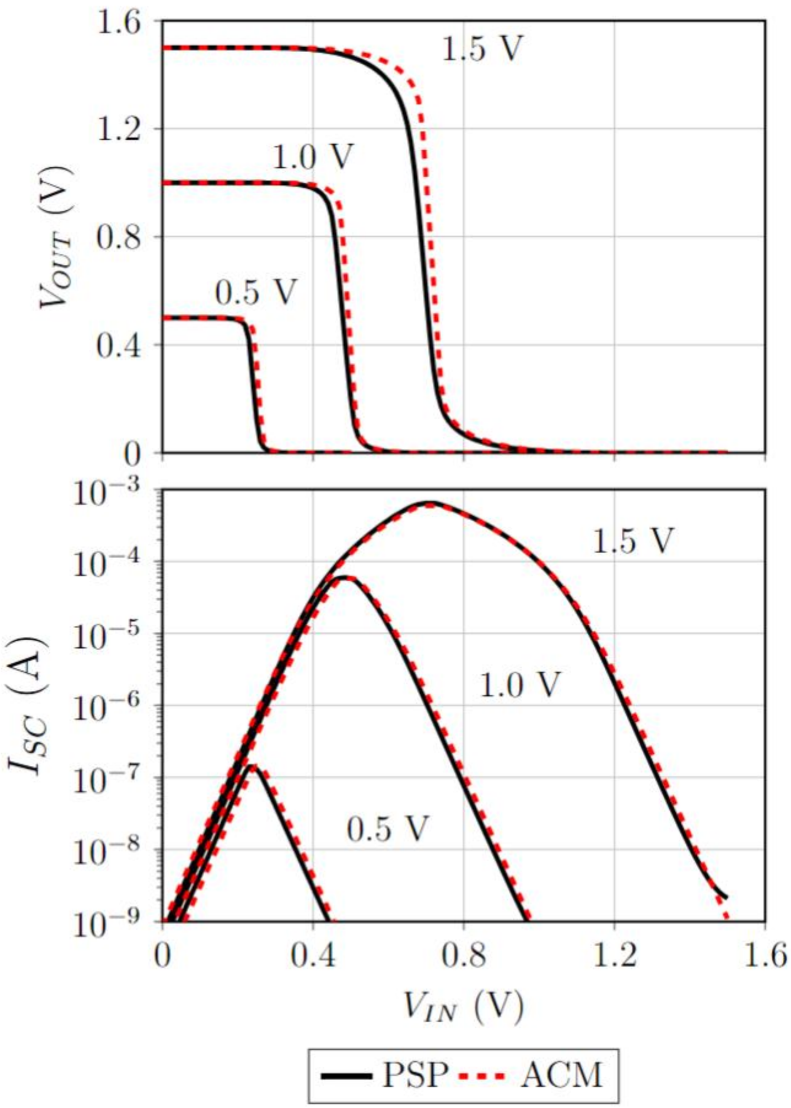
[2] Institut for High-Performance Microelectronics (IHP) open-source PDK

Outline

- **Compact models**
- **ACM2 model**
- **Parameter Extraction**
- **DC characteristics**
- **Circuit examples**

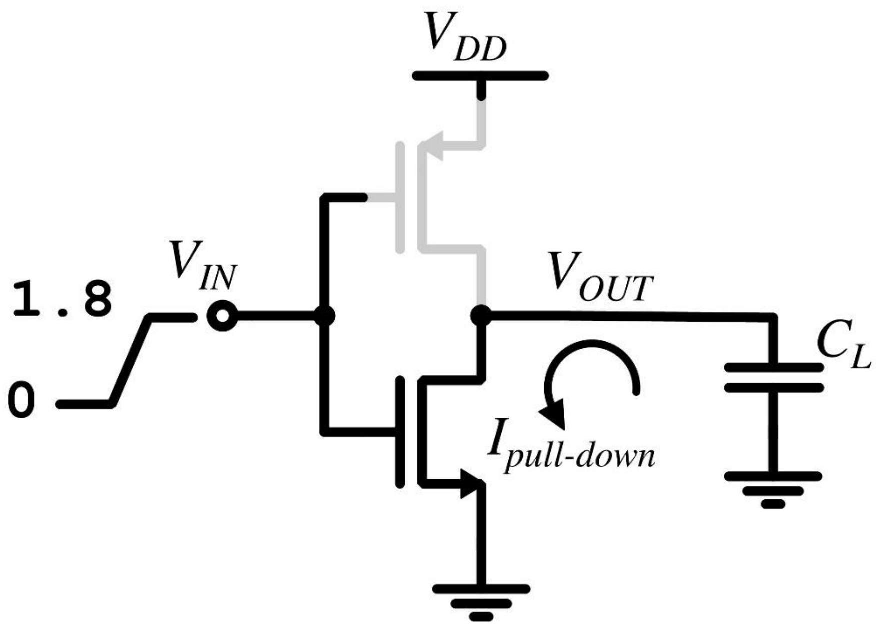
CMOS Inverter in 130 nm bulk
VTC and short-circuit current

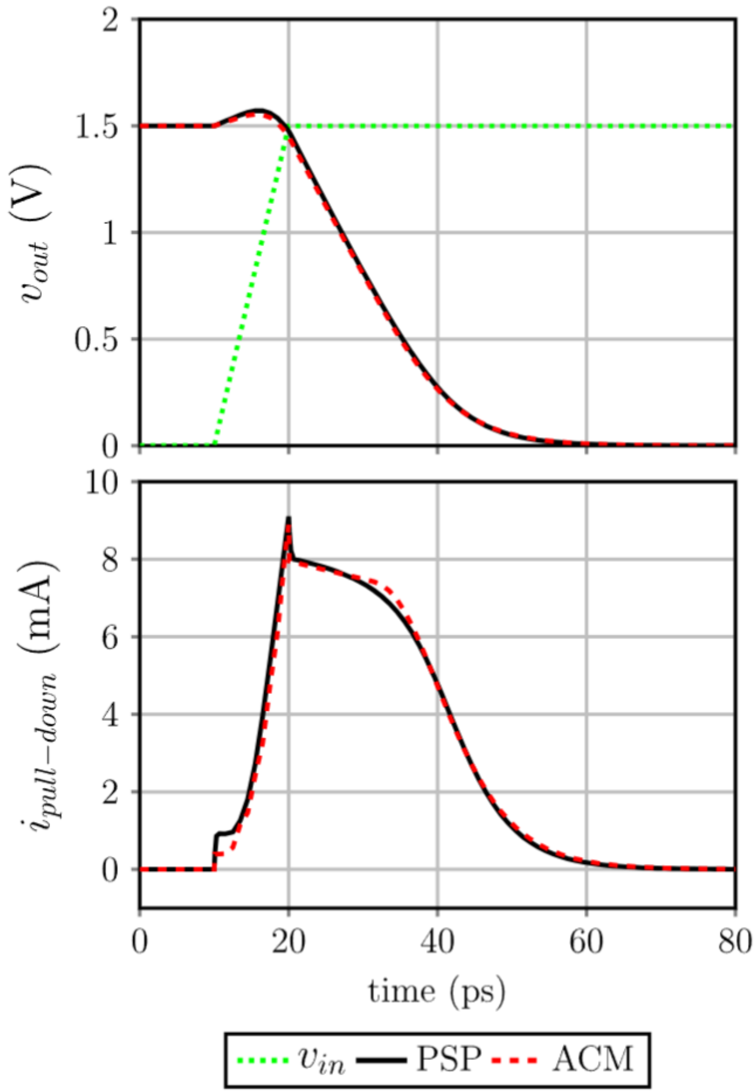




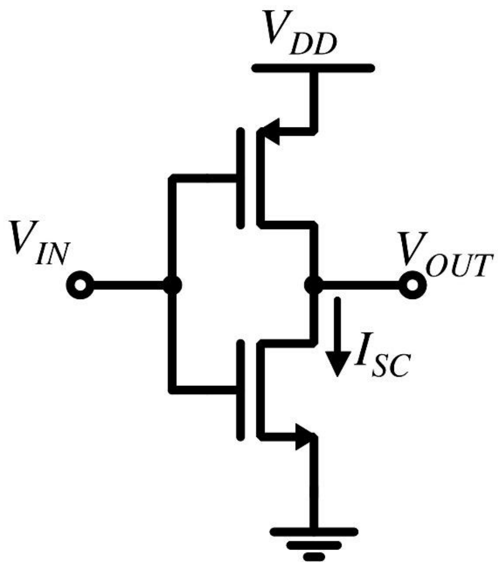
CMOS Inverter in 130 nm bulk

Output Voltage and pull-down current



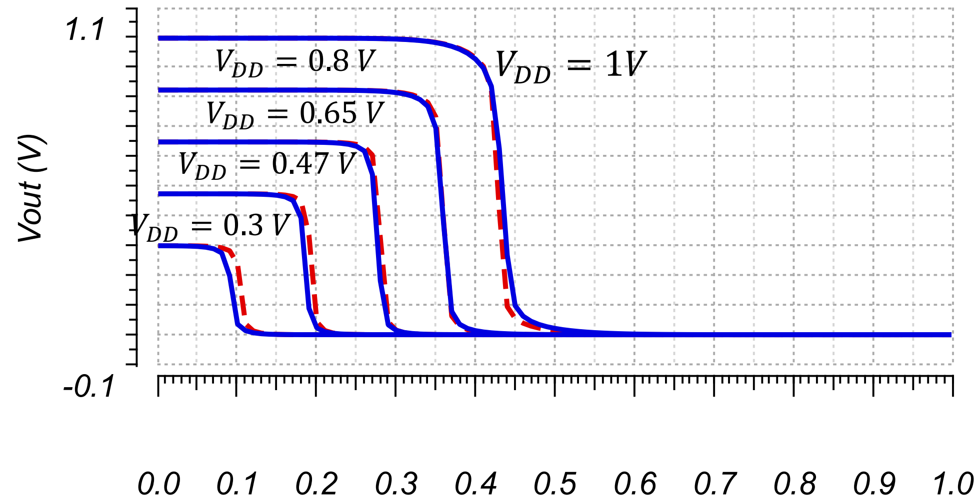


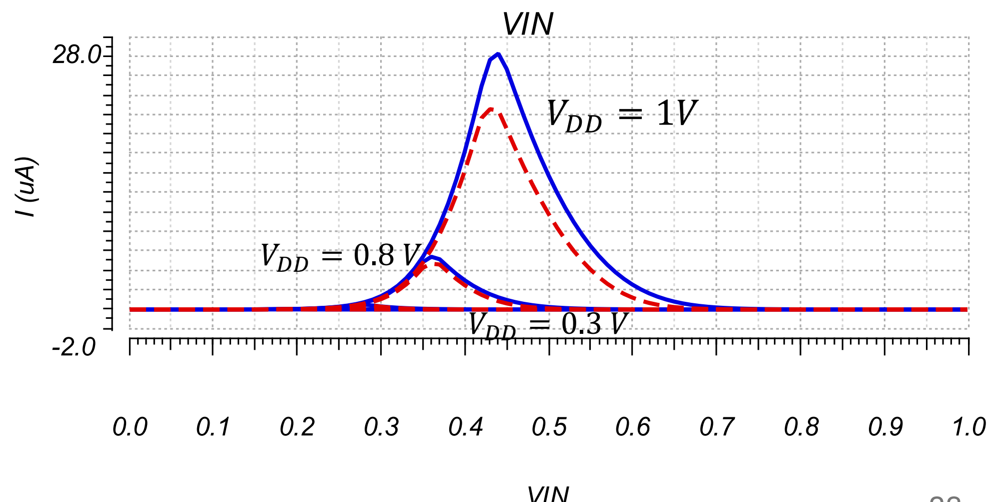
CMOS Inverter
VTC and short-circuit current in 28 nm FD-SOI



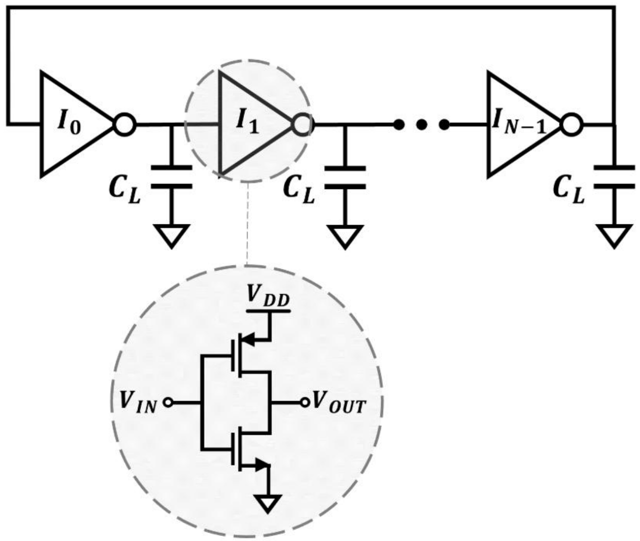
$$\boldsymbol{W_n = W_p = 1\ \mu m}$$
$$\boldsymbol{L_n = L_p = 60\ nm}$$

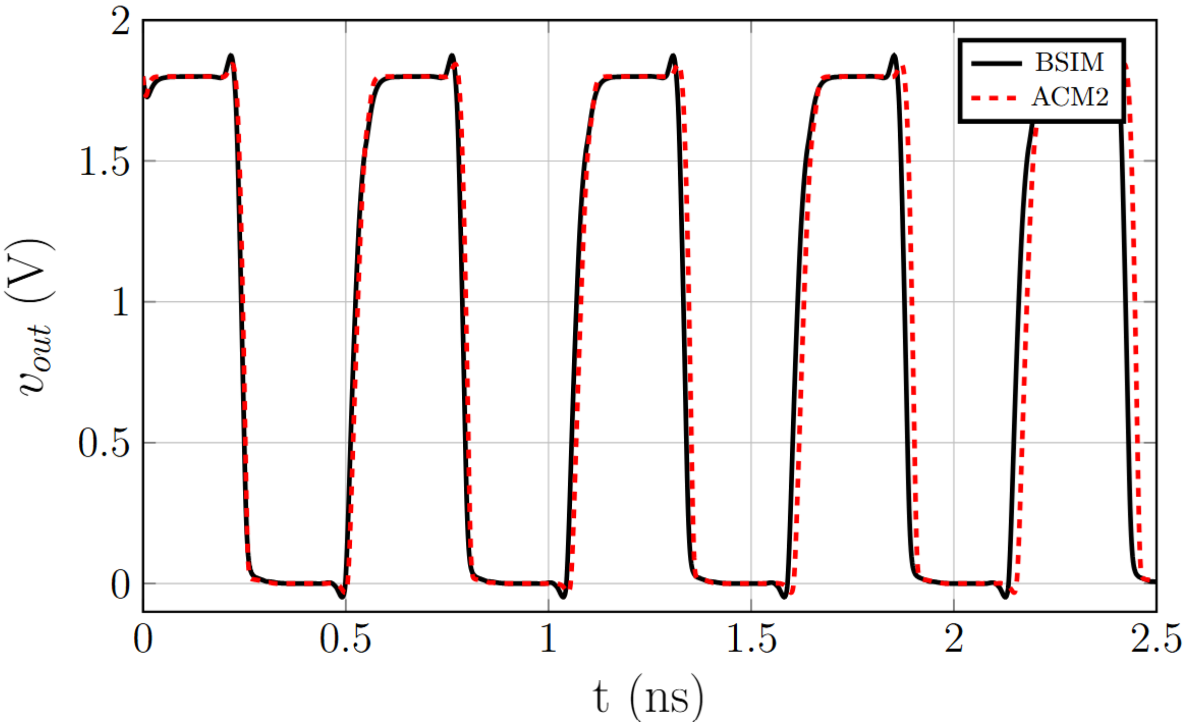
ACM
UTSOI2



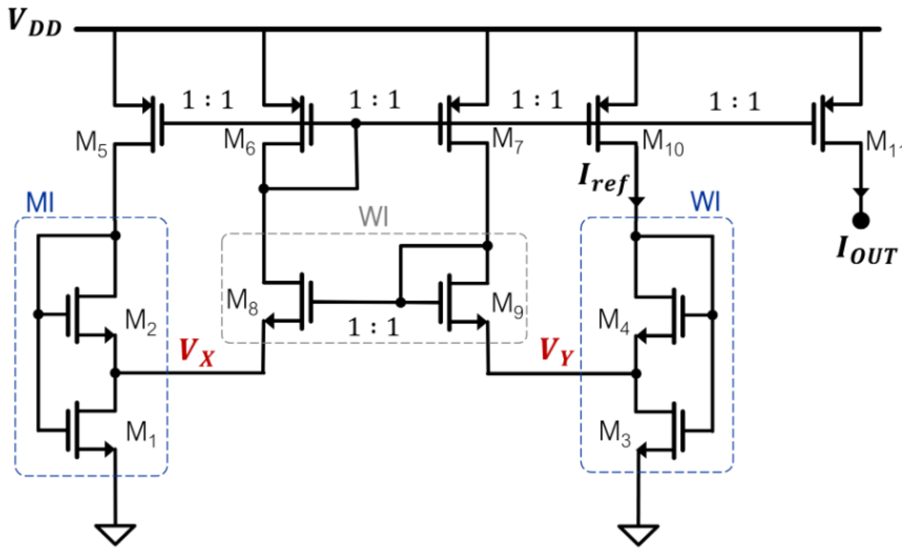


11-Stages Ring Oscillator 180 nm bulk
Output Voltage

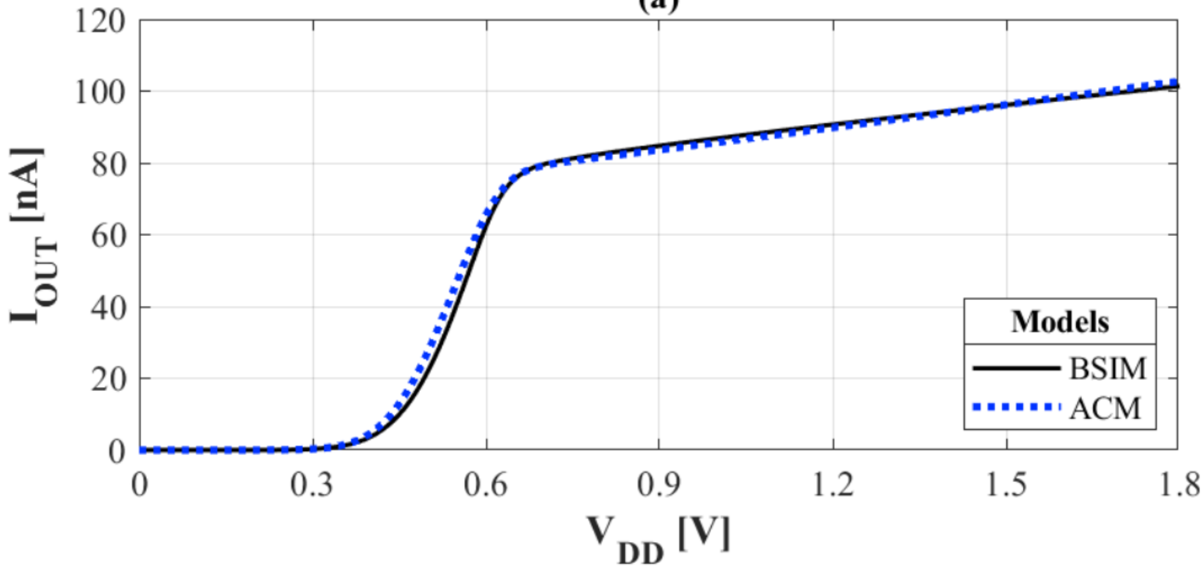




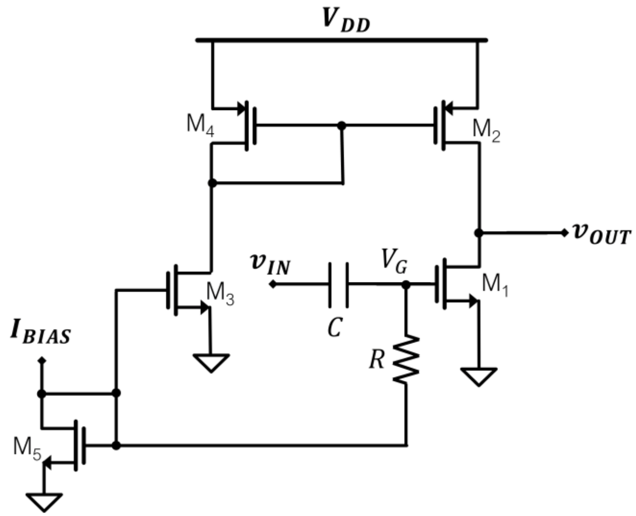
Self-biased current source in 180 nm bulk

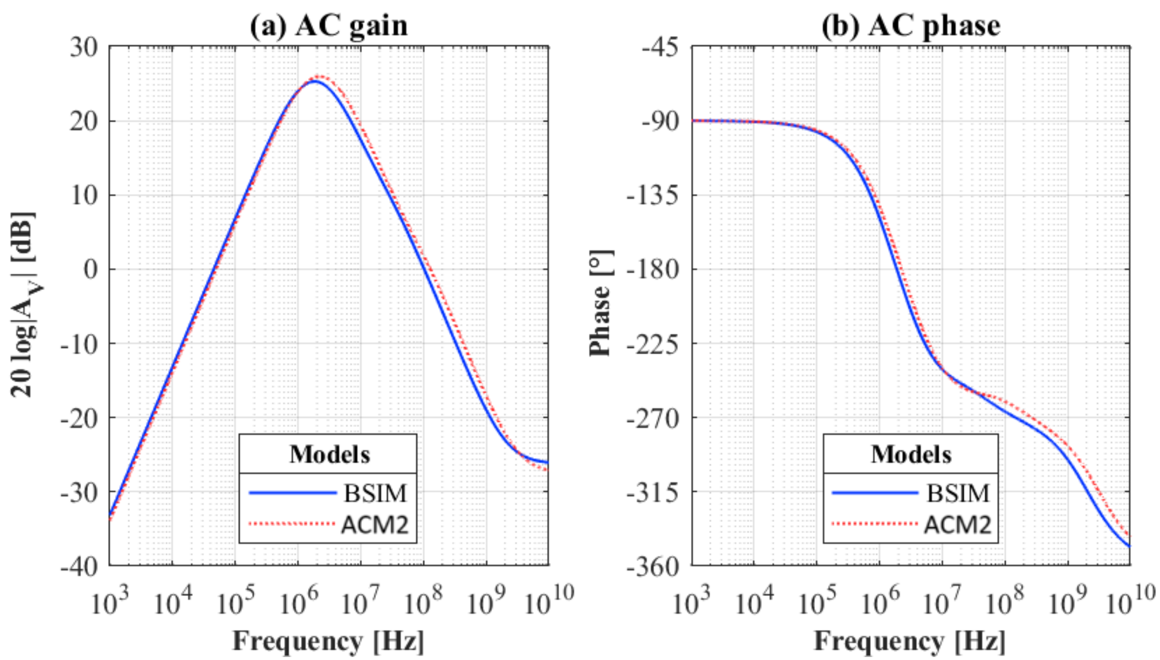


(a)

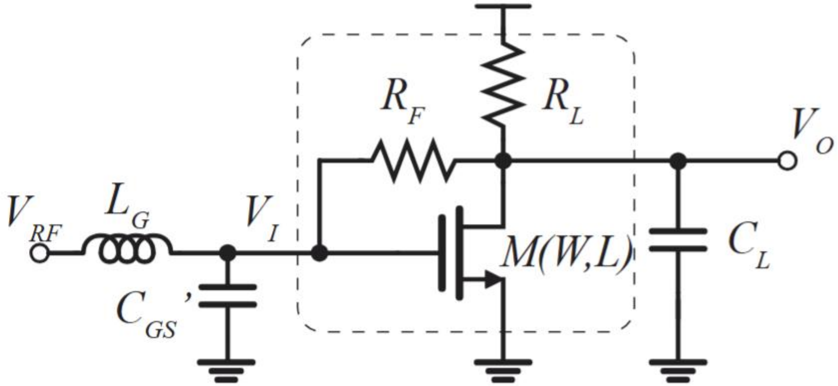


Common-source Amplifier in 180 nm bulk

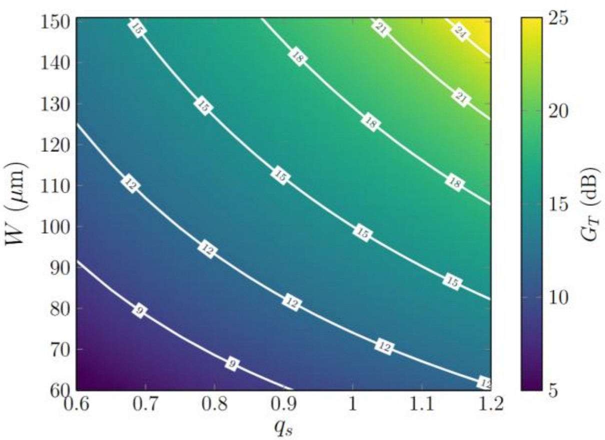




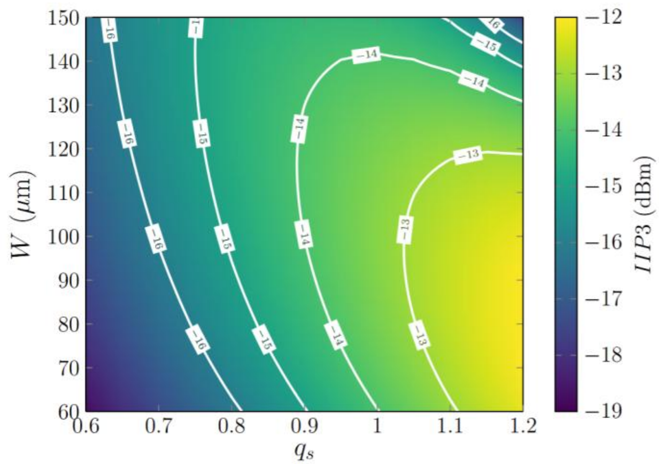
Use the model to explore the design space: R-feedback LNA case



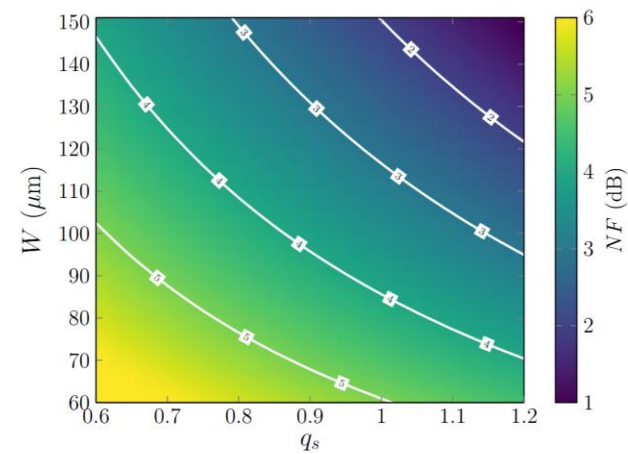
$GT \propto g_m(q_s, W)$



$IIP3 \propto g_m(q_s, W)$ & $g_{m3}(q_s, W)$



$NF \propto g_m(q_s, W)$



Summary – The ACM2 model

- *A truly compact MOSFET model with single-piece functions*
- *Implemented in Verilog-A for simulation*
- *Interchangeable between simulators (SPICE or SPECTRE)*
- *Verify in all three open-source PDKs (Sky130, GF180, IHP-SG13G2)*
- *Helpful to designers (only 5-DC-parameters)*
- *Simplified parameter extraction procedure*
    - *extracted from simulations or chip measurements*

Acknowledgments

CAPES

CNPq
Conselho Nacional de Desenvolvimento Científico e Tecnológico

SOIL
SOLIDIFY THE EUROPEAN FDSOI ECOSYSTEM ACCELERATING ITS INDUSTRIAL DEPLOYMENT

This project has received funding from the European Union's Horizon Europe research and innovation programme under the HORIZON-KDT-JU-2023-1-IA grant agreement No 101139785

Chips JU

UFSC

TIMA

cnrs

GRENOBLE INP UGA

UGA Université Grenoble Alpes

ihp

efabless.com

References

- D. Germano Alves Neto et al., "Design-Oriented Single-Piece 5-DC-Parameter MOSFET Model," in IEEE Access, vol. 12, pp. 87420-87437, 2024.
- Adornes, C.M.; Alves Neto, D.G.; Schneider, M.C.; Galup-Montoro, C. Bridging the Gap between Design and Simulation of Low-Voltage CMOS Circuits. J. Low Power Electron. Appl. 2022, 12, 34.
- ACM2 Github: https://github.com/ACMmodel/MOSFET_model
- IHP Github : https://github.com/IHP-GmbH/IHP-Open-PDK

"Scan me"

- **Available in Github:**
  - **DC model**
  - **Small-signal model**
  - **Dynamic model**
  - **Thermal &**

  Flicker noise models (1/f)

Github – ACM2

“Scan me”

Github - Content

ACMmodel Merge pull request #26 from gabrielmaranhao/main ··· 71ee7cd · 6 months ago

Examples	Update SKY130 and GF180 using ACM examples on xschem
Verilog-A	Update PMOS_ACM_2V0.va
docs	Delete 5PM_NewCAS.pdf
LICENSE	Update LICENSE
README.md	Update README.md

README ECL-2.0 license

Advanced Compact MOSFET model (ACM)

ACM is a simple MOSFET model to design and simulate Analog, Mixed-Signal, and RF circuits

Examples of PDKs and circuit simulators using the ACM model

NGSPICE & XSCHEM

- GF180MCU
- IHP-SG13
- SkyWater130
- Spectre

Verilog-A code Available!

MOSFET_model / Verilog-A / NMOS_ACM_2V0.va

ACMmodel Update NMOS_ACM_2V0.va

Code Blame 291 lines (245 loc) · 12.3 KB

```
//*****************************************************************************
// *  ACM NMOS model (Verilog-A)                                              *
// *  07/2023    V2.0.0                                                       *
//*****************************************************************************

// ****************************************************************************
// * Copyright under the ECL-2.0 license                                      *
// * Universidade Federal de Santa Catarina                                   *
// *                                                                          *
// * Current developers:  Deni Germano Alves Neto (Doctoral student, UFSC)    *
// *                      Cristina Missel Adornes (Doctoral student, UFSC)    *
// *                      Gabriel Maranhao        (Doctoral student, UFSC)    *
// *                                                                          *
// * Project Supervisors:  Prof. Carlos Galup-Montoro                         *
// *                       Prof. Marcio Cherem Schneider                      *
// ****************************************************************************

`include "constants.vams"
`include "disciplines.vams"

// function of the algorithm 443 to calculate de normalize charge densities
`define algo_443(Z,qn) \
        if(Z < 0.7385) begin \
                numeratorD = Z + (4.0/3.0)*Z*Z; \
                denominatorD = 1.0 + (7.0/3.0)*Z+(5.0/6.0)*Z*Z; \
                WnD = numeratorD/denominatorD; \
        end     else begin \
                numeratorD = ln(Z)*ln(Z)+2.0*ln(Z)-3.0; \
                denominatorD = 7.0*ln(Z)*ln(Z) + 58.0*ln(Z) +127.0; \
                WnD = ln(Z) - 24.0*(numeratorD/denominatorD); \
```