Annex 1
How to Utilize the Data available under 'extras.springer.com'

The data provided under 'extras.springer.com' consist of look-up tables listing the semi-empirical data and E.K.V. parameters of the N- and P-channel devices considered throughout the book. These are ***global variables*** that must be declared before undertaking any other action.

A1.1 Global Variables

Thz ***Glob.m*** file residing in the ***0 start*** directory must be run before any other file in order to declare the ***global variables*** (this must be done once when starting). The ***global variables*** encompass the arrays listed hereunder:

Semi-empirical Global Variables – Courtesy of IMEC

1. Drain currents (W = 10 μm)		
IDRAINn/p	Drain currents[1]	I_D
2. Transconductance (W = 10 μm)		
GMn/p	The gate transconductance	g_m
GMBn/p	The back-gate transconductance	g_{mb}
GDSn/p	The drain conductance	g_{ds}
3. Intrinsic capacitances (W = 10 μm)		
CGGn/p	The gate capacitance	C_{gg}
CGSn/p	The gate-to-source capacitance	C_{gs}
CGDn/p	The gate-to-drain capacitance	C_{gd}
CGBn/p	The gate-to-substrate capacitance	C_{gb}
CSSn/p	The source capacitance (com-gate)	C_{ss}
4. Gate lengths (μm)		
LL	Available gate lengths	

and

[1] The lower case letter ending each array refers to N- or P-channel transistors.

Compact Model Global Parameters (W/L = 1)

nN/P	Slope factor	n
VToN/P	Threshold voltage	V_{To}
ISuoN/P	Unary specific current	I_{Suo}
ThN/P	Mobility degradation factor	
PolyN/P	Theta polynomial	

A1.2 An Example Making Use of the 'Semi-empirical' Data: The Evaluation of Drain Currents and g_m/I_D Ratio Matrices (MATLAB A12.m)

Once Glob.m run, files can access ***global variables***. The name of the ***global variables*** put to use must be listed on top of the files. For instance, a file making use of N-channel drain currents must begin with:

$$\textbf{\textit{global}}\ \text{IDRAINn}\ldots \tag{A1.1}$$

IDRAINn (like any other ***global variable***, transconductance or capacitance) consists of a '9 by 49 by 49 by 9 4D array that can be accessed by means of subscripts specifying addresses: lg for the 9 available gate lengths, vgs for the 49 gate-to-source voltages, vds for the 49 drain-to-source voltages and vs for the 9 source-to-substrate voltages.

The available gate lengths are listed under the ***global variable*** *LL*:

$$LL = [0.10 \quad 0.11 \quad 0.12 \quad 0.13 \quad 0.14 \quad 0.16 \quad 0.50 \quad 1.00 \quad 4.00]\ \mu\text{m} \tag{A1.2}$$

The available gate-to-source V_{GS}, drain-to-source V_{DS} and source-to-substrate voltages V_S are:

Gate-to-source voltages	0 : 0.025 : 1.200(V)
Drain-to-source voltages	0 : 0.025 : 1.200(V)
Substrate voltages	0 : 0.100 : 0.800(V)

(A1.3)

Voltages can be translated into addresses[2]:

$$vgs = round\,(40 * VGS + 1) \tag{A1.4}$$

$$vds = round\,(40 * VDS + 1) \tag{A1.5}$$

$$vs = round\,(10 * VS + 1) \tag{A1.6}$$

To go from addresses to voltages, we make use of:

[2] The optional ***round*** instruction is recommended to avoid eventual non-integer subscripts resulting from arithmetic calculations.

$$VGS = 0.025 * (vgs - 1) \tag{A1.7}$$

$$VDS = 0.025 * (vds - 1) \tag{A1.8}$$

$$VS = 0.1 * (vs - 1) \tag{A1.9}$$

Consider an example: suppose we want to construct the drain current matrix of a 100 nm (lg = 1) grounded source transistor (vs = 1) whose V_{GS} is swept across the full range of gate voltages while the drain voltage varies from 0.6 to 1.2 V in steps 0.2 V wide (VDS = 0.6: 0.2: 1.2). For vgs, a colon suffices since we consider all possible V_{GS}'s. For vds, according to A1.5:

$$\text{vds} = \text{round}(40 * \text{VDS} + 1);$$

The size of the resulting drain currents array, named ID, is [1 49 4 1].

$$\text{ID} = \text{IDRAINn}(\text{lg}, :, \text{vds}, \text{vs})$$

To turn ID into a 49 rows and 4 columns matrix, one makes use of ***global variable the squeeze*** instruction:

$$\text{ID} = \textbf{\textit{squeeze}}(\text{ID});$$

The file below computes the derivative of $\textbf{\textit{log}}(I_D)$ with respect to V_G to generate the g_m/I_D matrix and plot the result versus the gate voltage. The derivative takes advantage of the ***diff*** instruction. Since ***diff*** instructions are carried out vertically, the drain current matrix must be organized along gate controlled rows and drain controlled columns.

```
1% test
2 clear
3 clf
4
5 global IDRAINn
6
7 lg = 1;
8 vs = 1;
9 VGS = (0:.025: 1.2)'; z = length(VGS);
10 VDS =.6:.2: 1.2; vds = round(40*VDS + 1);
11 ID = squeeze(IDRAINn(lg,:,vds,vs)); size(ID)
12
13 gmID1 = diff(log(ID))/.025;
14 U =.5* (VGS(1:z-1) + VGS(2:z));
15 [X,Y] = meshgrid(VDS,U);
16 gmID = interp2(X,Y,gmID1,VDS,VGS);
17
18 plot(VGS,gmID,'k');
19 xlabel('V_G_S (V)'); ylabel('gm/ID (1/V)');
```

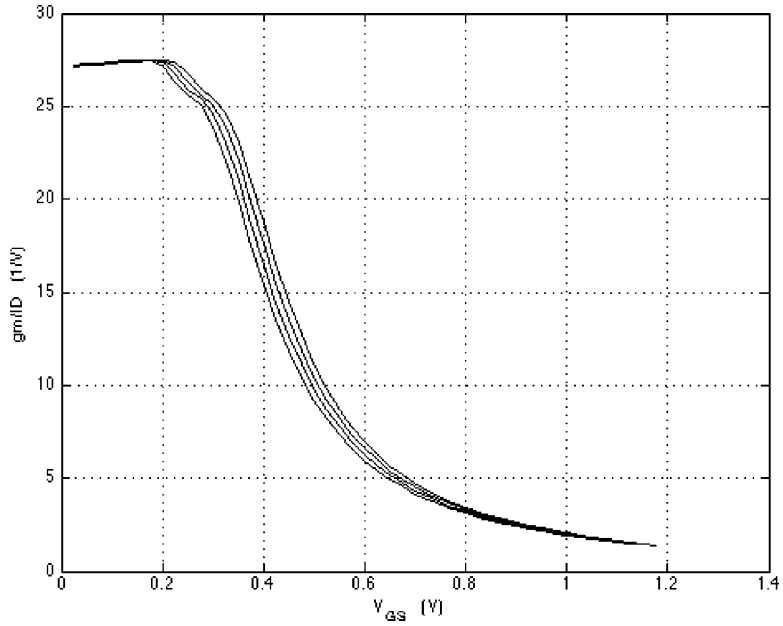

Fig. A1.1 The gm/ID curves obtained after running the file above

Care is needed regarding the size of gmID1. Owing to the differentiation, the number of rows of gmID1 is one step shorter than those of ID matrix. To get a g_m/I_D matrix with the same dimensions, the number of rows must be incremented by one unit. Resizing gmID1 in the vertical direction is done by means of the ***interp2*** instruction of line 16. We calculate therefore the X and Y matrix-coordinates of gmID1. This is done by means of the ***meshgrid*** instruction of line 15, which requires the pseudo-gate voltage U of gmID1 first. Figure A1.1 shows the final g_m/I_D curves. Notice that the same method can be put to use in order to calculate g_d/I_D when the drain current matrix is transposed before the ***diff*** instruction is performed.

A1.3 An Example Making Use of the E.K.V Global Variables: The Elaboration of an ID(VGS) Characteristic (Matlab A13.m)

The ***global variables*** nN/P, VToN/P and ISuoN/P, respectively the compact model slope factor, threshold voltage and unary specific current, of the compact model consist of '49 by 9 by 9' 3D arrays. These can be accessed by means of subscripts specifying vds, vs and lg, the same as with the 'semi-empirical' data. The model ignore V_{GS} by definition.

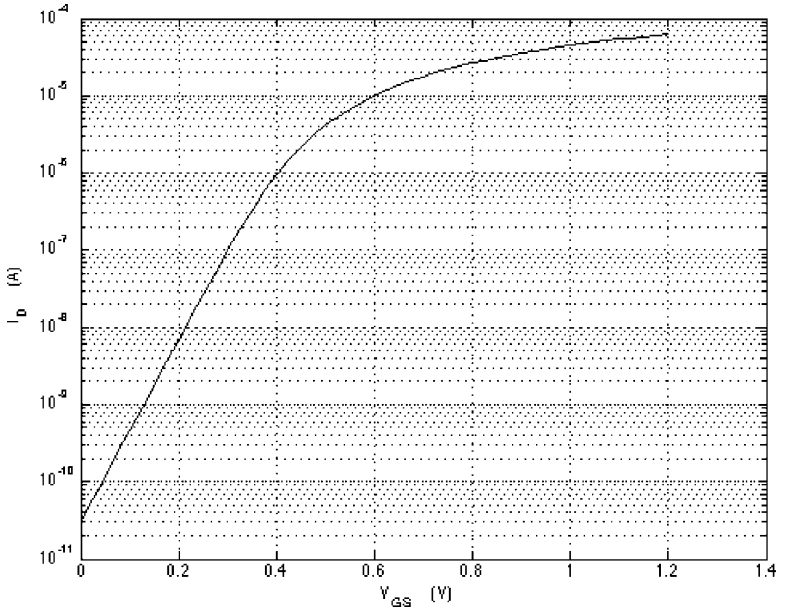

Fig. A1.2 The semilog representation the drain current versus the gate-to-source voltage resulting from the file above

The ***global variables*** PolyN/P and ThN/P are 4D arrays allowing to calculate the mobility degradation factor. The three first subscripts of both variables are the same as above. The fourth subscript of PolyN/P is always a colon. PolyN/P displays the coefficients ordered in descending powers of the mobility degradation polynomial. The number of coefficients is 5 (order 4 polynomial) and the argument of the polynomial the normalized drain current. The second global variable ThN/P makes use vds, vs, lg while the fourth subscript vgs calculates the degradation factor along the same lines as the polynomial representation.

The file below shows an example. The transistor is the same as above but the drain voltage is now constant and equal to 0.6 V. The slope factor n, the threshold voltage VTo and the unary specific current ISuo are scalars. A squeeze instruction is needed in order to turn the coefficients of the mobility degradation polynomial in to a vector. The calculation of the drain current is straightforward and follows the steps presented in Chapter 5. The result is shown in Fig. A1.2.

```
global nN VToN ISuoN PolyN
% data -----------------
lg = 1;
vs = 1;
VDS = 0.6;
% compute ----------------
UT =.026;
```

```
vds = round(40*VDS + 1);
n = nN(vds,vs,lg);
VTo = VToN(vds,vs,lg);
ISuo = ISuoN(vds,vs,lg);
P = squeeze(PolyN(vds,vs,lg,:));
VGS = 0:.025: 1.2;
VP = (VGS - VTo)./n;
qF = invq(VP/UT);
qR = invq((VP - VDS)/UT);
i = qF.^2 + qF - qR.^2 - qR;
ID = i.*ISuo./polyval(P,i);
% plot ------------------
semilogy(VGS,ID); grid
xlabel('V_G_S (V)'); ylabel('I_D (A)');
```

Annex 2
The 'MATLAB' Toolbox

A series of dedicated functions enabling to run MATLAB files referenced throughout the book are accessible in the toolbox. *It is strongly recommended to make use of the set path facility before running any file that makes use of functions of the toolbox*. If not done, the functions will not be accessed.

A2.1 Charge Sheet Model Files

The files hereafter are intended to reproduce figures of Chapters 2 and 3 and to carry out 'software experiments'.

A2.1.1 The pMat(T,N,tox) *Function*

The **pMat** function puts together the ***technology vector*** p (or ***matrix***) needed to run C.S.M. functions like the **IDsh** function. The input data are scalars and/or equal lengths row vectors representing: T (the temperature in K), N (the doping concentration expressed in at.cm^{-3}) and t_{ox}(the oxide thickness in nm). A sign is associated to the doping concentration N to differentiate semiconductor types, positive for N-type substrates, negative for P-type. Else, the sign is ignored. The output of the **pMat** function consists of (a) column vector(s). The three first rows list ϕ_B, γ, and U_T, which are utilized by the ***IDsh*** instruction described further. The fourth row yields K, the product of the mobility μ times the oxide capacitance per unit area C'_{ox} derived from the oxide thickness t_{ox}. The default value of μ is 500 cm^2/Vs for N-type and 190 cm^2/Vs for P-type transistors (open the **pMat** file to change these). The fifth row represents the gate oxide capacitance per unit area C'_{ox}. Every item can be accessed separately by means of its row index. For instance, p(3) reads U_T or *kT/q*.

Consider a N-channel transistor with a doping concentration equal to 10^{17} at.cm^{-3} and an oxide thickness of 5 nm. T is equal to 300 °K:

$$p = p\text{Mat}(300, 1e17, 5) \tag{A2.1}$$

pMat outputs one column, the interpretation of which is:

$$\begin{array}{llll} p(1) = 0.4078 & V & \Leftarrow & \Phi_B \\ p(2) = 0.2646 & V\text{^}(0.5) & \Leftarrow & \gamma \\ p(3) = 0.0259 & V & \Leftarrow & U_T \\ p(4) = 3,45e-4 & A/V\text{^}2 & \Leftarrow & K = \mu\, C'_{ox} \\ p(5) = 6,90e-7 & F/cm\text{^}2 & \Leftarrow & C'_{ox} \end{array} \tag{A2.2}$$

The parameters are updated automatically when the temperature changes owing to appropriate expressions stored in the **pMat** file. Consider for instance three temperatures 250, 300 and 350°K (MATLAB A13.m):

$$p = \text{pMat}(250 : 50 : 350, 1e17, 5); \tag{A2.3}$$

The output consists now of a 5 rows and 3 columns matrix. Each column corresponds to a temperature, 250 first, etc (γ and C'_{ox} are constants of course):

$$\begin{array}{llll} p = 0.4832 & 0.4462 & 0.4078 & \\ 0.2646 & 0.2646 & 0.2646 & \\ 0.0173 & 0.0216 & 0.0259 & \\ 0.7762 & 0.4968 & 0.3450 & *\text{e-3} \\ 0.6900 & 0.6900 & 0.6900 & *\text{e-6} \end{array} \tag{A2.4}$$

A2.1.2 The surfpot(p,V,VG) *Function*

The **surfpot** function calculates the surface potential by solving the non-linear implicit function listed under Eq. 2.20. The input data are the ***Technology vector*** p, the non-equilibrium voltage V and the gate voltage V_G. These may be scalars and/or equal length column-vectors.

Consider the same example as above with T equal to 300 °K. The gate-to-substrate voltage is constant and equal to 2 V while the non-equilibrium voltage V varies from 0 to 2 V. Two lines suffice in order to evaluate the surface potential, the ***Technology vector*** given by Eq. A2.1 and the **surfpot** function:

$$psiS = \text{surfpot}(p, linspace(0, 2, 100)', 2); \tag{A2.5}$$

The resulting surface potential is shown in Fig. 3.1. Knowing the surface potential, we can evaluate the threshold voltage V_T given by Eq. 3.6. All what is needed is to add the line below where $p(2)$ stands for γ.

$$V_T = p(2)^* sqrt(psiS) + psiS; \tag{A2.6}$$

A2.1.3 The IDsh(p,VS,VD,VG) *Function*

The ***IDsh*** function evaluates the drain current of 'unary' transistors according to the C.S.M model ('unary' means that the W over L ratio is equal to one). The input data consist of the ***Technology vector*** p and the terminal voltages with respect to the substrate: V_S, V_D and V_G. These may be scalars, equal length vectors or combinations. The function makes use of the MATLAB ***polyval*** instruction:

$$\frac{\mathrm{I_D}}{\beta} = \mathrm{polyval}\left(\mathrm{P}, \sqrt{\psi_{\mathrm{SD}}}\right) - \mathrm{polyval}\left(\mathrm{P}, \sqrt{\psi_{\mathrm{SS}}}\right) \tag{A2.7}$$

The surface potentials ψ_{SD} and ψ_{SS} are derived from the **surfpot** function, V being equal to V_D and V_S. P is a row vector consisting of the coefficients ranked from highest to lowest order of the polynomial listed under Eq. 2.19:

$$\mathrm{P} = \left[-\frac{1}{2} - \frac{2}{3}\gamma\,(V_G + U_T) \quad \gamma\,U_T \quad 0\right] \tag{A2.8}$$

It is recommended to add a realistic flat band voltage V_{FB} to V_G to take into consideration interface charges and the gate work function. V_{FB} is a separate variable that makes the gate voltage look more realistic. It does not reside in the ***Technology vector*** and is chosen freely. The flat-band voltage of N-channel transistors lies generally in the range 0.6–0.9 V. It depends on the physical treatments the transistor has been subjected to during fabrication, such as oxidation temperature, ion implantation, etc.

An example illustrating the use of the **IDsh** function is given in Annex 3.

A2.2 Compact Model Files

The files hereafter relate to the compact model of Chapters 4 and 5.

A2.2.1 The Identif 3(Nb,tox,VFB,T) *Function*

The **Identif3** function bridges the C.S.M. to the E.K.V. compact model. The function extracts n, V_{To} and I_{Suo} from C.S.M. drain currents. The input data are the substrate impurity concentration, oxide thickness, flat-band voltage plus the temperature. The parameter extraction is done by means of the algorithm described under Section 4.5.1.

A2.2.2 The invq(z) *Function*

The **invq** function inverts Eq. 4.2.3d:

$$V_P - V = U_T\,(2\,(q-1) + \log\,(q)) \tag{A2.9}$$

and computes the normalized mobile charge density q

$$q = invq\left(\frac{V_P - V}{U_T}\right) \tag{A2.10}$$

The pinch-off voltage V_P and non-equilibrium or channel voltage V may be scalars, equal size vectors or matrices.

A2.2.3 The ComS(VGS,VDS,VS,lg) *Function*

The function **ComS** evaluates the drain current I_D and the output conductance g_d versus V_{DS} of the variable parameters compact model. The gate-to-source voltage V_{GS} must be a scalar, the drain-to-source voltage V_{DS} a row vector (or a scalar) and the source voltage a scalar. Both, V_{GS} and V_{DS}, can take any value between 0 and 1.2 V whereas V_S should be one of the nine equally spaced source-to-substrate voltages comprised between 0 and 0.8 V.

The function evaluates n, V_{To}, I_{Suo} and the *Theta* function considering for the drain voltage two V_{DS} vectors separated by ± 1 mV. The output conductance g_d is derived from the ***diff*** of the drain current vectors divided by the 2 mV difference separating the drain voltages. The drain current I_D is the mean of the two drain currents. The output of the **ComS** function consists of a two columns matrix y, the first column represents the drain current I_D, the second the output conductance g_d.

A2.3 Other Functions

A2.3.1 The jctCap(L,W,R,V) *Function*

The **JctCap** function evaluates junction capacitances knowing the gate length $L(\mu m)$ and the gate width $W(\mu m)$ of N- and P-channel transistors (see Section 6.2.2). L and W may be scalars, vectors or matrices. The transistors are partitioned automatically in sub-transistors with identical widths comprised between maximal and minimal tolerated values fixed by R and $R/2$ (μm). Partitioning takes place as soon as W gets larger than R. The fourth variable V takes care of the reverse voltage applied to the junction. V is defined with respect to

the substrate for N-channel transistors and V_{DD} for P-channel. All capacitances are multiplied by the factor:

$$(1 + V/.5).\hat{}(-0.5)$$

Every capacitance combines a vertical junction capacitance CJ, two peripheral capacitances CJsw (outside periphery) and CJswg (inside periphery – the side capacitance between the junction and the channel) as illustrated in Fig. 6.9. The unit-capacitances are respectively equal to:

1e-15 F/μm^2 for CJ
1e-16 F/μm for CJsw
3e-16 F/μm for CJswg

The **JctCap** function outputs a 5D array y(:,:, 1 to 5) consisting of matrices having the same dimensions as L and W (the sizes of L and W determine the number of colons). These represent:

y(:,:,1) the drain junction cap. CJD
y(:,:,2) the source junction cap. CJS
y(:,:,3) the number of sub-transistors
y(:,:,4) the width of each sub-transistor
y(:,:,5) the total area of the transistor

Source capacitances are always larger than drain capacitances since the first surround the second.

A2.3.2 The Gss(x,H) *Function*

The **Gss** function calculates the Gaussian distribution of data listed in the column vector x. H is an optional variable representing the mean of x. The Gaussian distribution encompasses the 20 bins histogram of x (opening the **Gss** file allows changing the number of bins called M). The file outputs a graph representing the histogram, Gaussian distribution and lists the 3-sigma of the data in the command window.

Annex 3
Temperature and Mismatch, from C.S.M. to E.K.V.

The influence of temperature and mismatch on the drain current and g_m/I_D of the Charge Sheet Model is examined hereafter. It is extended to the E.K.V. model.

A3.1 The Influence of the Temperature on the Drain Current (MATLAB A31.m)

The influence of the temperature on I_D can be illustrated by means of the ***IDsh*** function of the MATLAB toolbox. The file below shows an example considering a grounded source transistor undergoing a temperature change from 250 to 350 K. The doping concentration of the substrate is supposed to be equal to 10^{17} at.cm^{-3}, the oxide thickness equal to 5 nm and the flat band voltage equal to 0.8 V. The drain voltage is large enough to keep the transistor saturated while the gate voltage varies from 0 to 2 V.

After inputting technological and electrical data, the ***pMAT*** function is called in order to set up the Technology Matrix required by the ***IDsh*** function.

```
% influence of T on ID(VG)
clear
clf
% technological data ------------------------
T = 250: 50: 350; % row vector
N = 1e17;
tox = 5;
VFB = 0.8;
% electrical data ---------------------------
VS = 0;
VD = 2;
VG = linspace(0,2,50)'; % column vector.
% compute -----------------------------------
p = pMat(T,N,tox);
for k = 1:length(T),
```

```
ID(:,k) = IDsh(p(:,k),VS,VD,VG + VFB);
end
% plot ---------------------------------
semilogy(VG,ID);
xlabel('V_G (V)'); ylabel('I_D (A)'); grid
```
(A3.1)

A number of well-known effects are illustrated in Fig. A3.1. When the temperature increases, the drain current grows rapidly in weak inversion while the opposite holds true in strong inversion. Conflicting effects explain the antagonist trends. The influence of the rising temperature on the factor A of Eq. 2.31 explains the increase in weak inversion. Mobility degradation explains the decrease in strong inversion. The first overrules the second in weak inversion while the opposite holds true in strong inversion. Around 0.8 and 1V, the two cancel out.

A3.2 The Influence of the Temperature on gm/ID (Matlab A32.m)

The evaluation the influence the temperature has on g_m/I_D is straightforward since the ratio boils down to the slope of the curves plotted in Fig. A3.1. The result is

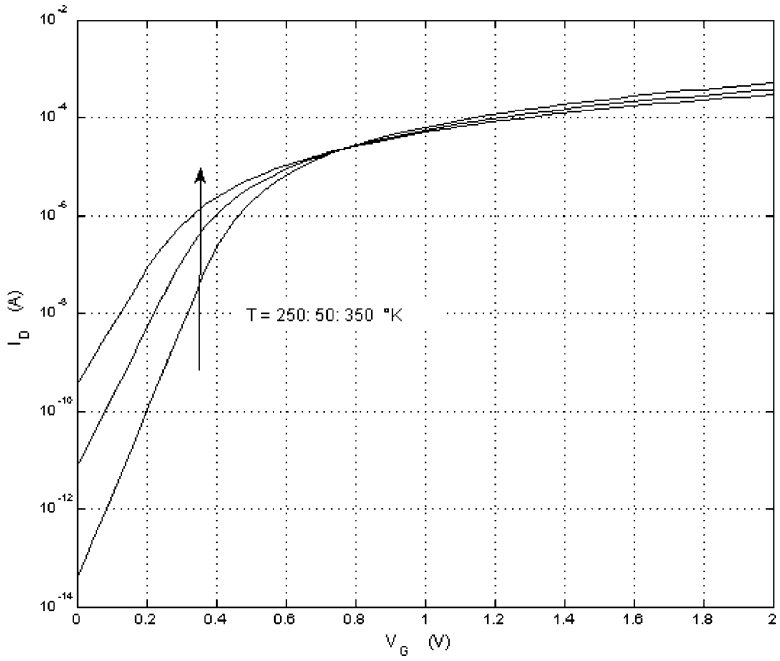

Fig. A3.1 C.S.M. drain current for temperatures of 250, 300 and 350 K

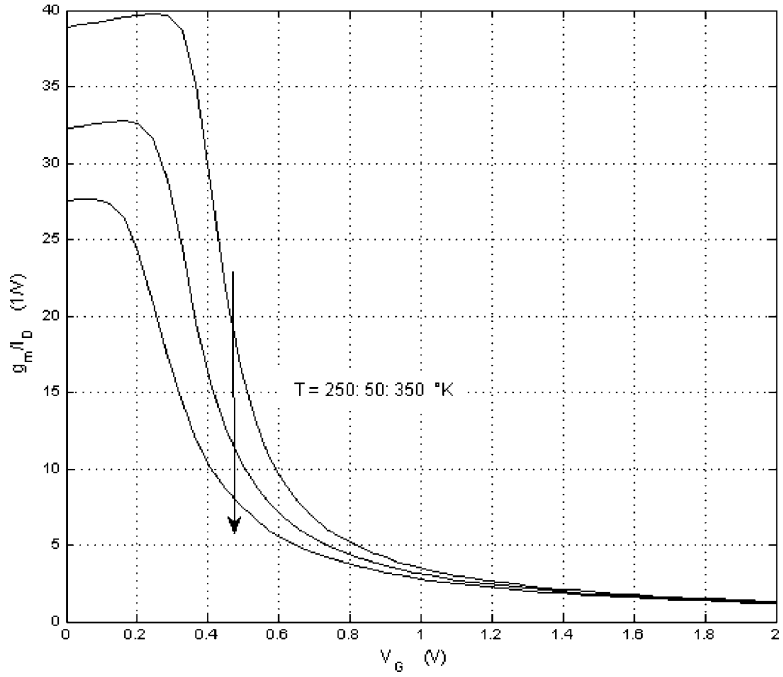

Fig. A3.2 g_m/I_D versus temperature of the transistor considered in the previous figure

Table A3.1 Comparison of temperature sensitivities of g_m/I_D's

T(K)	$n_{w.i.}$ (Eq. 2.38)	U_T 0.026.T/300	$1/nU_T(\mathrm{V}^{-1})$	max(g_m/I_D) (C.S.M.)
250	1.1628	0.0217	39.69	39.79
300	1.1749	0.0260	32.74	32.75
350	1.1892	0.0303	27.72	27.62

displayed in Fig. A3.2. The lessening of the subthreshold slope in weak inversion has a strong impact on the maximum g_m/I_D.

Table A3.1 compares the maximum of g_m/I_D predicted by the C.S.M. (most right column) to $1/nU_T$. The first is derived from the maximum of the derivative of $\log(I_D)$ whereas the second takes advantage of the analytic expression of the slope factor given by Eq. 2.38. The table shows that the latter is clearly a good approximation of the C.S.M. slope factor.

A3.3 Temperature Dependence of E.K.V Parameters (MATLAB A33.m)

We showed in Chapter 4 that the basic E.K.V model is an approximation of the C.S.M. The acquisition method enabling to extract E.K.V parameters from C.S.M drain currents described in Section 4.5 offers the possibility consequently to assess the impact of the temperature of n, V_{To} and I_{Suo}. The plots of Fig. A3.3 show the influence of the temperature of the slope factor n, the threshold voltage V_{To} and the unary specific current I_{Su} when the temperature goes from 250 to 350 K. The threshold voltage, which is equal to 0.3984 V at 300 K, drops by 1.31 mV/°C, the slope factor, equal to 1.1267, increases by 8.2×10^{-5} per°C, and the unary specific current, equal to 4.44 × 10–7 A, decreases by 62.3 pA per°C.

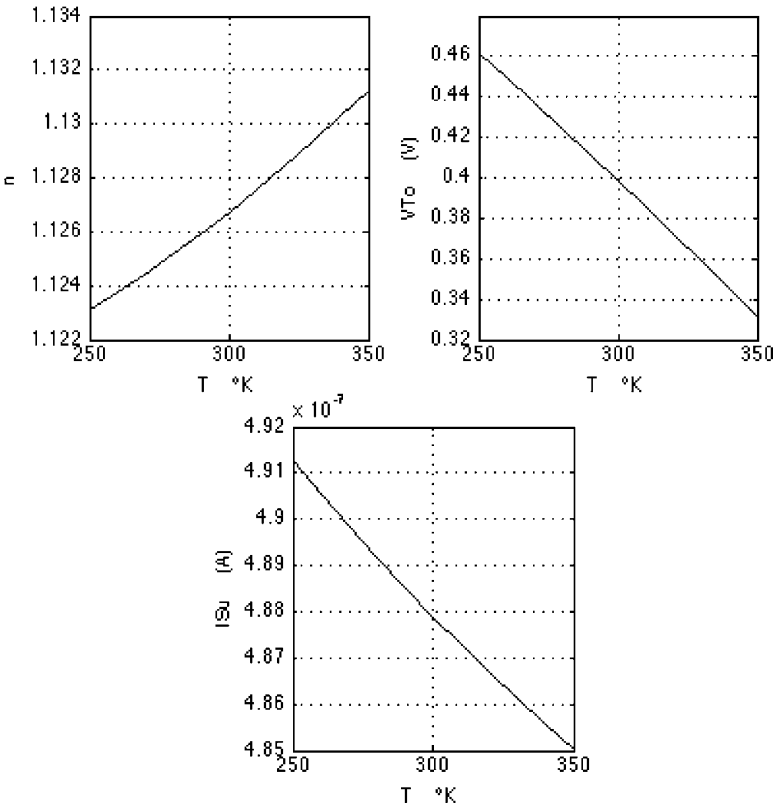

Fig. A3.3 influence of the temperature on the E.K.V parameters

A3.4 The Impact of Technological Mismatches on the Drain Current (Matlab A34.m)

The impact of substrate doping and oxide thickness mismatches on the drain current can be assessed easily with the C.S.M model. We consider the same transistor as above with T equal to 300 K and suppose that the doping concentration N and the oxide thickness *tox* obey Gaussian distributions, with sigmas respectively equal to 2.0% and 0.5%. We consider two constant gate voltages, one in weak and one in strong inversion. The two histograms of Fig. A3.4 give an idea of the spread of the drain current caused by the mismatches. Left, the gate voltage is equal to 0.2 V, right it is equal to 0.6 V. The mean unary drain currents are respectively 5.56 nA and 8.87 μA. The high mismatch sensitivity of MOS transistors in weak inversion is corroborated by a large spread.

```
% influence of N and tox mismatch on ID(VG)
clear
clf
% technological data ----------------
T = 300;
z = 1000; % number of samples
N = 1e17*(1 +.02*randn(1,z));
tox = 5*(1 +.005*randn(1,z));
VFB = 0.8;
% electrical data
VS = 0;
VD = 2;
VG = 0.2;
% compute
for k = 1:z,
p = pMat(T,N(1,k),tox(1,k));
ID(:,k) = IDsh(p,VS,VD,VG + VFB);
end
% plot --------------------------
M = mean(ID);
[n,x] = hist(ID(1,:),10);
bar(x/M,n)
h = findobj(gca,'Type','patch');
set(h,'FaceColor','w','EdgeColor','k')
axis([0 1.5 0 300]);
xlabel('I_D/mean(I_D)');
ylabel('histogram 1000 samples');
text(.3,200,'V_G = 0.2 V')
```

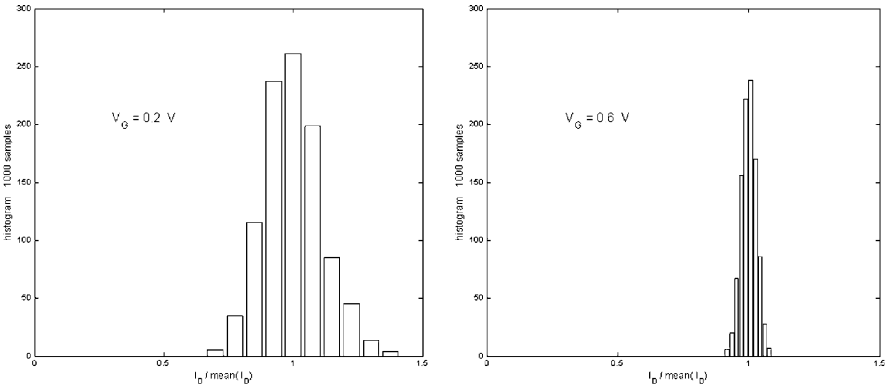

Fig. A3.4 Comparative histograms of relative drain currents spreads *left*, V_G is equal to 0.2 V (weak inversion), *right*, V_G is equal to 0.6 V (stong inversion)

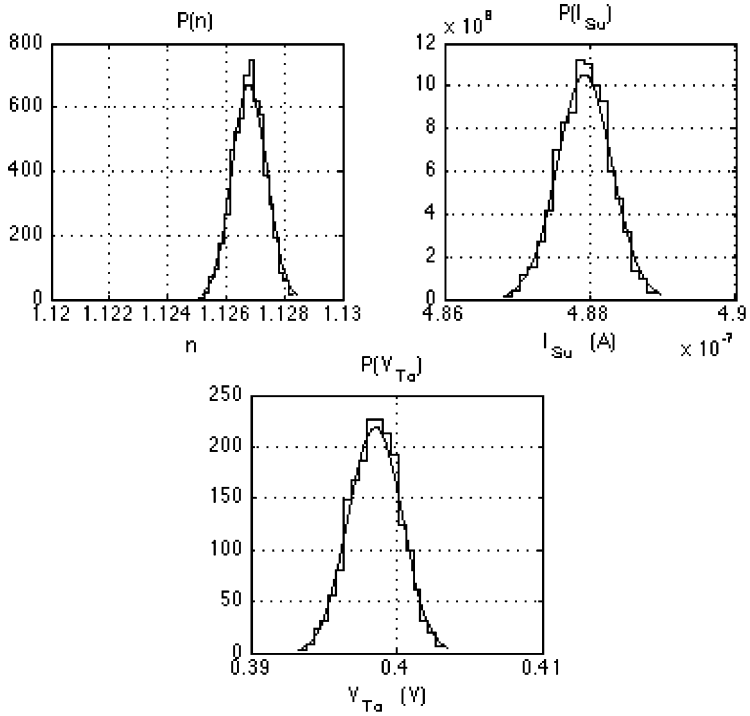

Fig. A3.5 Probability densities of E.K.V. model parameters

A3.5 Mismatch and E.K.V Parameters (MATLAB A35.m)

Since the C.S.M. offers the possibility to evaluate the impact of mismatches on drain currents, we can also evaluate their impact on the parameters of the equivalent E.K.V. model. We consider a Gaussian mismatch of the substrate impurity concentration centered around 10^{17} at.cm^{-3}. The sigma is equal to 1%. The oxide thickness and flat band voltage are constant and the same as in the previous example. The probability densities of n, V_{To} and I_{Suo} are displayed in Fig. A3.5.

The impact of mismatches on the parameters is illustrated by the three-sigma deviations listed below:

$3\sigma(n) = 0.0018\%$
$3\sigma(V_{To}) = 5.6\,\text{mV}$
$3\sigma(I_{Su}) = 1.16\,\text{nA}$

Annex 4
E.K.V. Intrinsic Capacitance Model

The intrinsic gate-to-source and gate-to-drain capacitances of the E.K.V model are compared to their 'semi-empirical counterparts in this annex. We consider a grounded source N-channel transistor and sweep the gate and drain voltages from 0 to 1.2 V. The 'semi-empirical' capacitances are extracted from the global variables ***CGSn*** and ***CGDn*** (Courtesy of IMEC). We make use of the expressions below for the model, where C_{ox} stands for the oxide capacitance fixed by the width and the length of the transistor (Section 5.3.1 of Enz and Vittoz 2006):

$$C_{gsi} = C_{ox}\frac{q_F}{3}\cdot\frac{2q_F + 4q_R + 3}{(q_F + q_R + 1)^2} \tag{A4.1}$$

$$C_{gdi} = C_{ox}\frac{q_R}{3}\cdot\frac{2q_R + 4q_F + 3}{(q_F + q_R + 1)^2} \tag{A4.2}$$

To evaluate q_F and q_R versus the gate and drain voltages, the E.K.V. parameters are extracted first from 'semi-empirical' drain currents by means of the acquisition algorithm presented in Chapter 5.

The 'semi-empirical' capacitances include overlap capacitances that are ignored by the E.K.V model. To separate extrinsic from intrinsic 'semi-empirical' capacitances, we evaluate the 'semi-empirical' capacitances under bias conditions minimizing the contribution of the intrinsic capacitances. For instance, we get rid of the inversion layer by zeroing the gate-to-source voltage to evaluate the gate-to-source overlap capacitance C_{gsov}. The fact that the overlap capacitances per μm gate width listed in Table A4.1 are not affected by gate lengths changes while the gate capacitances per μm do, supports the idea.

Figures A4.1 and A4.2 compare 'semi-empirical' (left) to model intrinsic capacitances (right) considering two gate lengths: 500 and 100 nm. To make a fair comparison, we add the gate-to-source overlap capacitance derived from the 'experimental' data to the intrinsic capacitances of the model and adjust the vertical scale to get the same maximum capacitance. It is clear that the E.K.V. intrinsic gate-to-source capacitance is not a bad representation, except when the transistor is not saturated.

Caution is needed however as far as the overlap capacitances. These depend not only on extrinsic contributions but also on the underlying junction-to-substrate

Table A4.1 Extrinsic and intrinsic gate-to-source capacitances (exper. data)

L (μm)	C_{gsov} (fF/μm)	C_{gsi} (fF/μm)
0.100	0.363	0.413
0.110	0.413	0.469
0.120	0.413	0.574
0.130	0.413	0.683
0.140	0.412	0.792
0.160	0.412	1.010
0.500	0.408	4.806
1.00	0.408	10.258
4.00	0.419	42.189

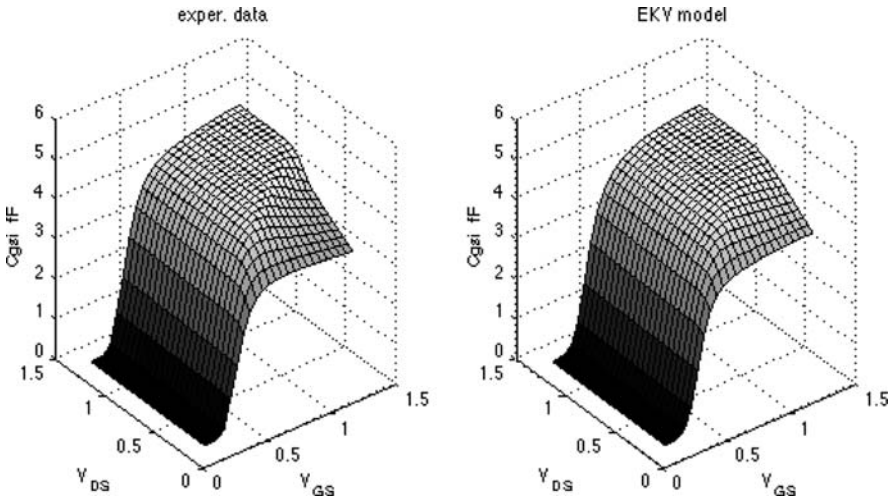

Fig. A4.1 The gate-to-source capacitance of the 500 nm gate length transistors

voltage (see Section 10.3 of Enz and Vittoz 2006). The phenomenon is clearly visible in Figs. A4.3 and A4.4, which displays gate-to-drain overlap capacitances C_{gdov} according to the method above.

When the transistor is saturated, the gate-to-drain capacitance is far from being constant, especially when the gate length effects are not visible on the gate-to-source. The gate-to-drain capacitances predicted by the model is a poor representations of C_{DS} when the transistor is saturated contrarily to C_{GS}.

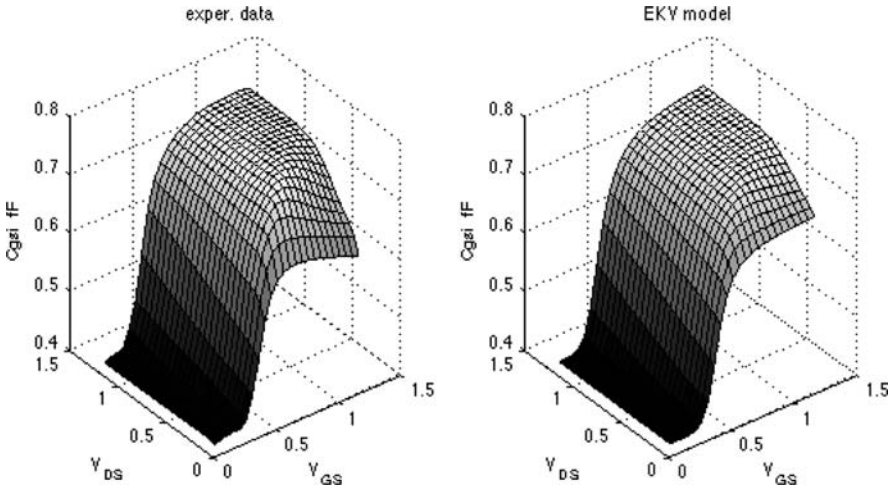

Fig. A4.2 The gate-to-source capacitance of the 100 nm gate length transistors

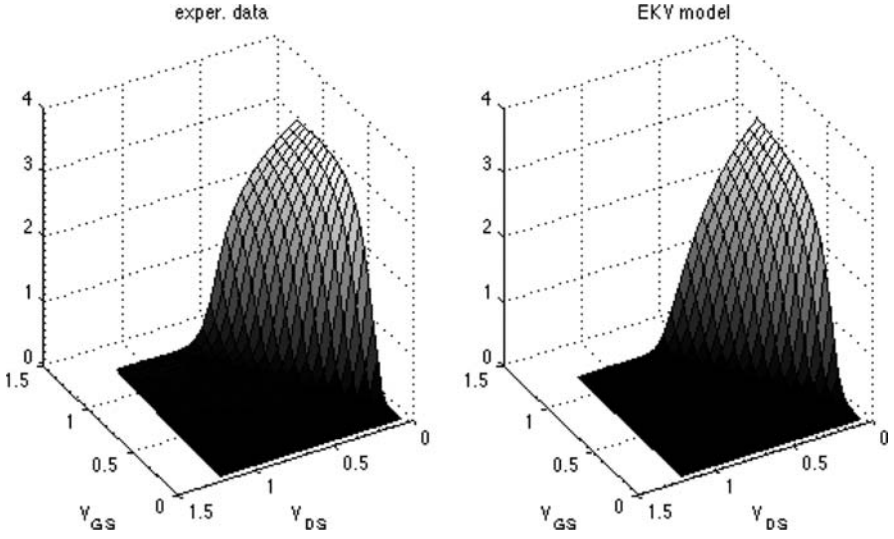

Fig. A4.3 The gate-to-drain capacitance of the 500 nm gate length transistor

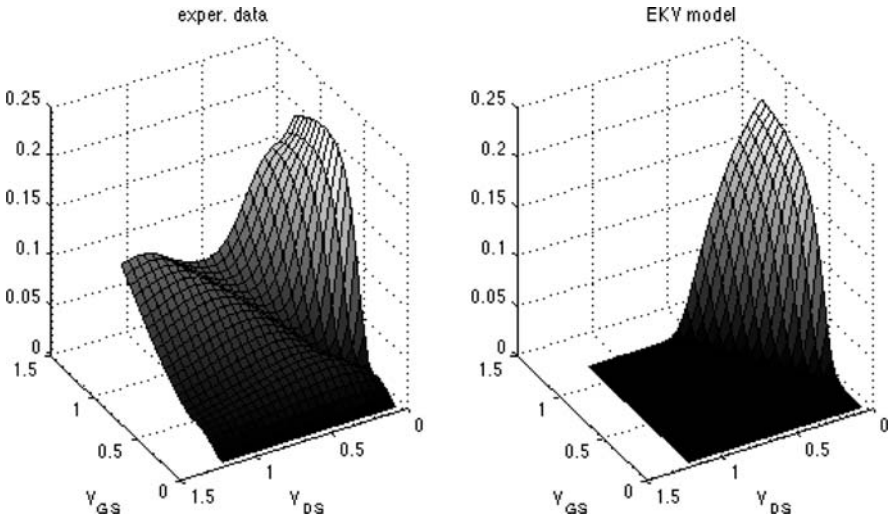

Fig. A4.4 The drain-to-source capacitance of the 100 nm, gate length transistor

Bibliography

Banu M, Tsividis Y (Oct 1984) Detailed analysis of nonidealities in MOS fully integrated active RC filters based on balanced networks. Electron Circuits Syst, IEE Proc G 131(5):190–196

Binkley DM (2007) Tradeoffs and optimization in analog CMOS design. Wiley, Chichester, England, ISBN 978–0–470–03136–0

Binkley DM et al (Feb 2003) A CAD methodology for optimizing transistor current and sizing in analog CMOS design. IEEE Trans Comput Aided Des Integr Circuits Syst 22(2):225–237

Brews JR (1978) A charge sheet model for the MOSFET. Solid State Electron 21:345–355

Brun Y, Lee K, Shur M (Jan 1990) Unified charge controlled model and subthreshold current in heterostructure field effect transistors. IEEE Electron Device Lett 11(1):50–53

Bucher M (1999) Analytical MOS transistor modelling for analog circuit simulation. Ph.D. thesis no. 2114, Swiss Federal Institute of Technology, Lausanne (EPFL)

Cand M, Lardy JL, Demoulin E, Senn P (1986) Conception des circuits integers. Annexe 1, Eyrolles, Paris, pp 163–169

Chatelain JD (1979) Traité d'Electricité, vol VII, Dispositifs à Semiconducteurs, EPFL

Coltinho RM, Spiller LH, Schneider MC, Montoro CG-M (2001) Metodologia simplificada de extraçao de Parâmetros para o modelo A.C.M. do transistor MOS. VII workshop de Iberchip – IWS'2001

Cunha AIA, Scheider MC, Galup-Montoro C (Oct 1998) An MOS transistor model for analog circuit design. IEEE JSCC 33(10):1510–1519

Enz CC (2008) A short story of the EKV MOS transistor model. IEEE Solid State Circuits News 13(3):24–30. www.ieee.org/SSCS-news

Enz CC, Vittoz EA (2006) Charge-based MOS Transistor Modeling. *The EKV model for low-power RF IC design*. Wiley, Chichester

Girardi A, Bampi S (2006) Power constrained design optimization of analog circuits based on physical gm/ID characteristics. 19th IEEE Symposium on Integrated Circuits and Systems, SBCCI 2006, pp 89–93

Girardi A, Cortes FP, Bampi S (2006) A tool for automatic design of analog circuits based on gm/ID methodology. IEEE ISCAS 2006

Grabinski W, Nauwelaers B, Schreurs D (2006) Transistor level modelling for analog/RF IC design. Springer, The Netherlands

Jespers PGA, Jusseret C, Leduc Y (June 1977) A fast sample and hold charge-sensing circuit for photodiode arrays. IEEE JSSC SC-12(3):232–237

Laker KR, Sansen WMC (1994) Design of analog integrated circuits and systems, McGraw-Hill series in Electrical and Computer Engineering

Miller JM (1920) Dependence of the input impedance of a three-electrode vacuum tube upon the load in the plate circuit. Scientific Papers of the Bureau of Standards, vol 15, no. 351, pp 367–385

Muller RS, Kamins ThI (1977) Device electronics for integrated circuits, 2nd edn. Wiley, New York, p 36

Oguey H, Cserveny S (1982) Sonderdruck aus dem Bulletin des SEV/VSE, Bd. 73, 1982, pp 113–119

PSP (2006) http://pspmodel.asu.edu http://www.nxp.com/Philips_Models/mos_models/psp/

Silveira F, Flandre D, Jespers P (Sept 1996) A g_m/I_D based methodology for the design of CMOS analog circuits and its application to the synthesis of a silcon-on-isulator micropower OTA. IEEE J Solid State Circuits 31(9):1314–1319

Tsividis Y (1999) Operation and modelling of the MOS transistor, EE series. Mc-Graw Hill, New York

Van de Wiele (Dec 1979) A long channel MODFET model. Solid State Electron 22(12):991–997

Vittoz E, Fellrath J (June 1977) Small signal model of MOS transistors in Weak Inversion Operation, JSSC IEEE 12(3):232–237

Wallinga H, Bult K (June 1989) Design and analysis of CMOS analog signal processing circuits by means of a graphical MOST model. JSSC IEEE J 24(3):672–680

Index

A
A.C.M. model, 41

B
body effect, 79
Boltzmann statistics, 13, 25, 42

C
cascoded Intrinsic Gain Stage
  gain evaluation, 117
cascoded Intrinsic Gain Stage
  frequency response, 118
  poles and zeros, 118
  sizing, 115
Channel length modulation (C.L.M.), 78, 80
Charge Sheet Model (C.S.M.), 11
  common-gate configuration, 23
  drain current equation, 13
  drain current versus drain voltage, 15
  drain current versus gate voltage, 17
  g_m/I_D, 20
  weak inversion approximation, 18
common-gate configuration
  compact model drain current and g_{ms}/I_D, 113
compact model for real transistors, 68
  equations, 70
  g_d/I_D, 88
  g_m/I_D, 85
  $I_D(V_{DS})$, 82
  mobility degradation polynomial, 73
  parameter acquisition, 70
  parameter dependence on bias conditions, 78
  parameter dependence on the gate length, 76
cut-off angular frequency, *see* cut-off frequency
cut-off frequency, 3

D
diffusion current, 12, 18
drain induced barrier lowering (D.I.B.L.), 68, 79, 89
  impact on the pinch-off voltage, 83
drift current, 12, 18

E
E.K.V. model, 41
  drain current, 45, 48
  equations, 46
  g_m/I_D, 54
  g_{ms}/I_D, 57
  mobility degradation, 59
  parameter acquisition, 50
  weak and strong inversion approximations, 50
Early voltage, 2, 89
extrinsic capacitances, 99
  transistor partitioning, 101

G
gain-bandwidth product, 3
gate voltage overdrive (G.V.O.), 6
global variables, 143
  compact model parameters, 144
  example calculate $I_D(V_{GS})$ from compact model, 147
  example extract g_m/I_D from semi-empirical data, 144
  semi-empirical, 143
g_m/I_D sizing methodology, 7
gradual channel approximation, 11, 41, 67

graphical construction, 27
CMOS transmission gates, 35
compact model, 47
implementation of linear resistors, 36
small signal transconductances, 34
source bootstrapping, 37
the CMOS inverter, 33
the MOS diode, 32
the MOS source follower, 32

I
intrinsic capacitances (E.K.V. model), 163
Intrinsic Gain Stage (I.G.S.), 1
equivalent circuit, 1
frequency response, 1
gain evaluation with var. param. compact model, 106, 107
simplified sizing procedure making use of the var. param. compact model, 110
sizing in moderate inversion, 5
sizing in strong inversion, 4
sizing in weak inversion, 4
sizing the cascoded I.G.S., 115
sizing with E.K.V model, 55
sizing with E.K.V. model and mobility degradation, 65
sizing with semi-empirical data (constant output capacitance), 95
sizing with semi-emprical data (with output junction capacitance), 103
sizing with variable param. compact model, 104
transfer function, 108

J
junction capacitances
vertical and side-wall capacitances, 101

M
MATLAB
IDsh function, 15
Indentif3.m function, 50
pMat function, 15
surfpot function, 15
MATLAB toolbox, 149
Miller Op. Amp., 121
analysis, 122
current mirror, 126
frequency response, 124
phase margin, 129
pole splitting, 123
poles and zeros, 127
sizing a high-frequency low-power Miller Op. Amp., 140
sizing a low-voltage Miller Op. Amp., 130
sizing methodology, 129
transfer function, 127
mismatch, 155
mobility coefficient, 12
mobility degradation, 80, 83
critical field, 60
first order approximation, 59
impact of mobility degradation on the drain current, 60
impact of mobility degradation on g_m/I_D, 64
impact on the specific current, 80
longitudinal and vertical electrical fields, 80
MOS
quadratic model, 4
weak inversion model, 4

N
normalized drain current, 45
forward, 46
reverse, 46
normalized mobile charge density, 42

P
pinch-off voltage, 27, 38, 43, 44, 62, 83

Q
quasi-stationarity, 98

R
reverse short channel effect, 67, 78, 79
roll-off, 67, 78, 79

S
semi-empirical g_m/I_D, g_d/I_D and gain dependence on bias conditions, 93
short channel effects, 67
sizing-space dimensions, 121
slew-rate, 7, 111, 142
slope factor, 41
specific current, 45
unary specific current, 50
specifications and attributes, 121
subthreshold slope, 18, 22
surface potential, 12

T
temperature, 155
threshold voltage, 24
  of E.K.V. model, 45
  with respect to the source, 26
  with respect to the substrate, 26, 28, 30, 31
transistor partitioning, 101
transition angular frequency, 3
transition frequency, 3