McCAE'S SIMULATORS

J.W. Bandler, Q.J. Zhang, S. Ye,
R.M. Biernacki and S.H. Chen

SOS-90-4-R

February 1990

McCAE'S SIMULATORS

J.W. Bandler, Q.J. Zhang, S. Ye, R.M. Biernacki and S.H. Chen

Abstract The circuit simulation procedures in the CAD program McCAE are described step by step in this report. The procedures cover DC, small-signal, and large-signal simulations. The derivation of the gradient calculation of DC and small-signal responses is also provided.

This work was supported in part by the Natural Sciences and Engineering Research Council of Canada under Grants STR0040923, OGP0007239 and OGP0042444 and in part by Optimization Systems Associates Inc.

The authors are with the Simulation Optimization Systems Research Laboratory and the Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada L8S 4L7. J.W. Bandler, Q.J. Zhang, R.M. Biernacki and S.H. Chen are also with Optimization Systems Associates Inc., P.O. Box 8083, Dundas, Ontario, Canada L9H 5E7.

I. INTRODUCTION

Circuit simulation is fundamental to any circuit CAD program. Examples are linear circuit simulation in Touchstone [1], and nonlinear circuit simulation in Microwave Harmonica [2]. In this report we describe the circuit simulation part of the microwave CAD program McCAE [3], which has both linear and nonlinear circuit simulation capabilities.

The basic logical connections among circuit devices, simulators, and circuit responses in McCAE are illustrated in Fig. 1, where the nonlinear DC and linear small-signal analyses are performed sequentially on the one hand, and the harmonic balance (HB) technique is employed to perform nonlinear large-signal analysis on the other. The same device models are used by McCAE to achieve consistent results [4] for both linear and nonlinear simulations.

In the following sections, we consider the DC/small-signal and HB (large-signal) simulations of a single FET circuit, as shown in Fig. 2, respectively, where the intrinsic part of the FET device is nonlinear and bias-dependent. In Sections II and III, the DC/small-signal simulation is discussed. The harmonic balance simulation is presented in Sections IV and V.

II. DC/SMALL-SIGNAL SIMULATION

We use the conventional DC/small-signal simulation to analyze the circuit which operates at a given bias point under small-signal conditions. When given linear circuit element parameters, nonlinear FET parameters, and bias voltages, the simulation procedure first determines the DC operating point of the circuit, and then calculates DC and/or small-signal AC responses. The simulation procedure is described as follows, where we use italic capital letters to denote related subroutine names in McCAE.

DC/Small-Signal Simulation Procedure

Step 1 Determine the DC operating point of the circuit for the given bias excitation. This is accomplished by two substeps.

Step 1.1 Call *R_GETY* to construct the Y matrix of the DC equivalent circuit, where Y contains only the entries from the linear elements of the DC equivalent circuit.

Step 1.2 Call *DCSIM* to simulate the nonlinear DC circuit by solving a set of nonlinear nodal equations.

(1). *R_DC_S* computes the equivalent current excitation of the DC circuit.

(2). *VMODLS* sets the initial values for the nodal voltages, where the values are chosen such that they are close to the corresponding bias voltages.

(3). *NEWTON* is called to solve the nonlinear nodal equations, where an efficient nonlinear equation solver *HYBRID* is utilized.

Comment When the solution can not be obtained due to nonconvergence, a loop inside *DCSIM* is executed to try to resolve the nonlinear equations by gradually increasing the drain bias voltage. Also in *DCSIM* the gradient of the nodal voltages w.r.t. the linear and nonlinear parameters are computed via adjoint analysis if the simulation is called for optimization purposes. See the Appendix for details of the gradient computation.

Step 2 Calculate the DC responses by *R_DC_RESP*, which is called inside *DCSIM*. The DC responses are the currents of the gate and drain bias voltage sources.

Step 3 Compute the small-signal parameters of the nonlinear device at the operating point. After the DC solution, the operating point of the nonlinear device is obtained. *SML_PA* is then called to compute the small-signal parameters of the nonlinear device, where a perturbation method is employed for the computation.

Step 4 Simulate the small-signal equivalent circuit by calling SIMFET at each frequency point specified at the bias point.

(1). *R_GETY* is called to get the Y matrix at the requested frequency. Y contains only entries from the linear elements in the circuit.

(2). *R_MODY* modifies the Y matrix by adding small-signal parameters from the nonlinear device into the existing Y matrix.

(3). The linear nodal equation in matrix form is solved by *CSOLU*, a linear equation solver using LU factorization.

(4). Construct the two-port Z matrix, and then convert the Z matrix to S-parameter [5].

Comment In *R_SIMFET*, the gradient of the S-parameters w.r.t. all the circuit parameters is also computed by the adjoint analysis if the simulation is called for optimization purpose. See Appendix for the combined DC/small-signal gradient calculation.

Step 5 If DC/small-signal simulation is required at another bias point, go to Step 1, otherwise simulation is completed.

III. NONLINEAR DC NODAL EQUATION AND ITS JACOBIAN

In Step 1 of the DC/small-signal simulation procedure, subroutine *HYBRID* is utilized to solve the nonlinear nodal equation, where the nonlinear nodal equation and its corresponding Jacobian matrix are provided by subroutines *FCN_R* and *JAC_R*, respectively.

The nonlinear nodal equation is in the form of

$$\mathbf{F(V)} = \mathbf{I(V)} + \mathbf{YV} + \mathbf{I_s} \tag{1}$$

where **V** contains nodal voltages, **I(V)** contains current from the nonlinear components, **Y** is the admittance matrix from the linear elements in the DC equivalent circuit, and $\mathbf{I_s}$ is the equivalent bias source vector. The subroutine *FCN_R* calculates **I(V)** by calling *GETNL_R*, and forms **F(V)** by adding to **I(V)** the linear term $\mathbf{YV} + \mathbf{I_s}$.

The Jacobian matrix of **F(V)** can be expressed as

$$[\partial \mathbf{F}^T(\mathbf{V})/\partial \mathbf{V}]^T = [\partial \mathbf{I}^T(\mathbf{V})/\partial \mathbf{V}]^T + \mathbf{Y} \tag{2}$$

where superscript T stands for transposition. Perturbation method is used in *JAC_R* to compute $[\partial \mathbf{I}^T(\mathbf{V})/\partial \mathbf{V}]^T$. $[\partial \mathbf{F}^T(\mathbf{V})/\partial \mathbf{V}]^T$ is computed by combining $[\partial \mathbf{I}^T(\mathbf{V})/\partial \mathbf{V}]^T$ with the bias-independent

Y matrix.

IV. HARMONIC BALANCE SIMULATION

The harmonic balance (HB) technique [6] is highly efficient in the simulation of steady-state nonlinear circuit. The basic idea of HB simulation is that we separate the nonlinear circuit into linear and nonlinear subcircuits, assume finite harmonics in the system, simulate linear subcircuit in the frequency domain and nonlinear subcircuit in the time domain, and the solution is reached when the currents from the linear and nonlinear subcircuits become consistent at the linear-nonlinear subcircuit interface.

To facilitate the presentation, we use the term state variable to denote the variables to be solved by HB analysis from which any type of circuit responses can be easily constructed. In McCAE, the state variables are port voltages between the linear and nonlinear parts of the circuit. We consider again that the circuit to be solved is a single FET circuit. The term bias-input-frequency combination (BIF) is used to denote that the circuit is simulated in one fundamental frequency, with one power input, and at one bias point. The HB procedure implemented in McCAE is described below which should be used for every BIF specified by the user.

HB Simulation Procedure

Step 1 Initialization.

Step 1.1 Set up the Y matrix of the linear subcircuit by *FORMY* which generates the port admittance matrix for HB analysis.

Step 1.2 Set up the constant discrete Fourier transformation (DFT) matrix by *SETDFT*.

Comment Since the matrix only depends on the number of harmonics included in the HB analysis, it does not need to be updated if the HB simulator is repeatedly called during optimization.

Step 2 Solving the HB equation. The HB equation is solved in *SIMUFET*.

Step 2.1 *SETDFT1* is called to set up a supplementary DFT matrix to include the time delay factor τ.

Comment This routine could have been called in Step 1. However, since τ may be an optimization variable, the supplementary DFT matrix corresponding to τ may not be constant.

Step 2.2 If necessary, the state variables are initialized by calling *INITSTV*. The initialization criterion for the state variables is that the DC components are initialized to their corresponding bias voltages, fundamental frequency components to approximately the same order of magnitude of the input voltage, and higher harmonic components decrease proportionally w.r.t. their corresponding fundamental components.

Comment With the assumption that the solution of a slightly perturbed HB equation is close to that of the unperturbed, different initialization schemes are applied to different situations. For example, (i) in power sweep simulation, i.e., from low power input to high power input, the initial values of the state variables are taken from the solution of the previous simulation if the current one is not the one with the lowest power input; (ii) if the simulation is performed within optimization, the initial values of the state variables at current iteration are taken from the corresponding solution of the previous iteration. Therefore *INITSTV* is called only if there is no suitable reference for the state variable or if the corresponding simulation in the previous optimization iteration failed to converge.

Step 2.3 The equivalent source vector is computed by calling *INITEXCI*.

Step 2.4 *NEWTON* is called to solve the nonlinear HB equation.

Comment In *NEWTON* the nonlinear equation solver *HYBRID* is utilized to solve the nonlinear equations created from the HB equations.

Comment When *SIMUFET* is first called by power sweep, frequency sweep, or optimization

routine, the nonlinear circuit is solved by gradually increasing the input power level in order to achieve better convergence of the solution of the HB equation.

Step 3 Circuit response calculation in *SIMUFET* by calling *RESP*.

Comment The gradient of the circuit response w.r.t. circuit parameters is also computed in *SIMUFET*, if it is called from the optimization routines.

V. THE HARMONIC BALANCE EQUATION AND ITS JACOBIAN

As presented in Section IV, the HB equation is solved by calling subroutine *NEWTON* which further calls *HYBRID*, the same routine as we used in the DC simulation. The HB equation and its corresponding Jacobian are provided to *HYBRID* by subroutines *FCN_HB* and *JAC_HB*, respectively.

In *FCN_HB*, the state variables which are in the frequency domain are first transformed to their time domain form by calling *DFTFT*. Then *MODEL* is called by *FCN_HB* which takes the time domain form of the state variable as input, computes the time domain responses of the nonlinear components, and transforms the responses back to the frequency domain as output. On the other hand, the frequency domain state variable is used to obtain the response of the linear subcircuit in frequency domain. The HB equation is formed by combining linear and nonlinear frequency domain responses together according to [6]

$$\mathbf{F(V)} = \mathbf{I(V)} + j\mathbf{\Omega Q(V)} + \mathbf{YV} + \mathbf{I}_s \tag{3}$$

where **V** contains the state variables in frequency domain, **I(V)** is the current contribution due to nonlinear current sources from the nonlinear subcircuit, $j\mathbf{\Omega Q(V)}$ the current contribution due to nonlinear capacitors also from the nonlinear subcircuit, and $\mathbf{YV} + \mathbf{I}_s$ is the current from linear subcircuit and the circuit excitations.

The subroutine *JAC_HB* formulates the Jacobian matrix of the HB equation, i.e, $[\partial \mathbf{F}^T(\mathbf{V})/\partial \mathbf{V}]^T$. Similar to *FCN_HB*, *JAC_HB* first transforms the frequency domain state variables into their corresponding time domain form. Then *JACOBO* is called to compute the

time domain derivative of each nonlinear component w.r.t. to the time domain state variables by perturbation, and transform the time domain derivative back to the frequency domain. Then, $\partial \mathbf{I}^T(\mathbf{V})/\partial \mathbf{V}$ is assembled by *FITID*, $\partial \mathbf{Q}^T(\mathbf{V})/\partial \mathbf{V}$ by *JACQ*, and $\partial(\mathbf{YV})^T/\partial \mathbf{V} = \mathbf{Y}$ by *JACLIN*. The whole Jacobian is assembled by combining those three parts,

$$[\partial \mathbf{F}^T(\mathbf{V})/\partial \mathbf{V}]^T = [\partial \mathbf{I}^T(\mathbf{V})/\partial \mathbf{V}]^T + j\Omega[\partial \mathbf{Q}^T(\mathbf{V})/\partial \mathbf{V}]^T + \mathbf{Y} \ . \tag{4}$$

VI. CONCLUSIONS

In this report we have presented the circuit simulation parts of the CAD program McCAE. The DC, small-signal, and large-signal HB simulation procedures together with the corresponding subroutines in McCAE are described step-by-step. Even though the presentation is focused on a single FET circuit, the principles are general and can be applied to more general circuits.

APPENDIX

GRADIENT CALCULATION OF DC AND SMALL-SIGNAL RESPONSES

For integrated DC/small-signal optimization of circuits [7], the gradient of the DC and/or small-signal S parameter responses with respect to circuit parameters are required. The DC response f_{DC} is constructed from DC nodal voltages $\mathbf{V}_{DC}$. We can write $f_{DC} = f(\phi_{DC}, \mathbf{V}_{DC})$, where ϕ_{DC} represents either parameters ϕ_{DCL} from the linear circuit components and ϕ_{DCN} from the nonlinear circuit components in the DC equivalent circuit. The gradient calculation of f_{DC} with respect to any parameter in ϕ_{DC} in the DC equivalent circuit can be expressed symbolically as

$$\partial f_{DC}/\partial \phi_{DC} = \partial f/\partial \phi_{DC} + (\partial f/\partial \mathbf{V}_{DC})^T(\partial \mathbf{V}_{DC}/\partial \phi_{DC}). \tag{A1}$$

In McCAE, $\partial f_{DC}/\partial \phi_{DC}$ is assembled in *DCSIM*, where $\partial \mathbf{V}_{DC}/\partial \phi_{DC}$ is obtained by *R_DC_ADJ* and *R_DFDX* in the way of nonlinear adjoint analysis. $\partial f/\partial \phi_{DC}$ and $\partial f/\partial \mathbf{V}_{DC}$ can be determined according to the way that f_{DC} is constructed.

The small-signal S parameter response f_{AC} is a direct function of the parameters in the small-signal equivalent circuit: ϕ_{DCL}, ϕ_{ACL}, $\phi_{DCS}(\phi_{DCN},V_{DC})$, and $\phi_{ACS}(\phi_{ACN},V_{DC})$. ϕ_{ACL} is the parameter from linear elements which only affect the small-signal equivalent circuit, ϕ_{DCS} is the small-signal parameter derived from the nonlinear components existing in the DC equivalent circuit, and ϕ_{ACS} is the small-signal parameter derived from nonlinear components (with parameter ϕ_{ACN}) which do not affect the DC equivalent circuit. Therefore, f_{AC} can be expressed as

$$f_{AC} = f(\phi_{DCL}, \phi_{ACL}, \phi_{DCS}(\phi_{DCN},V_{DC}), \phi_{ACS}(\phi_{ACN},V_{DC})). \tag{A2}$$

In actual implementation, f_{AC}, i.e., the two-port S parameter of the small-signal equivalent circuit, is computed in *R_SIMFET* by first calculating the Z matrix of the two-port circuit and then converting the Z matrix to S parameters. See [5] for detailed derivations.

The calculation of the derivative of f_{AC} with respect to a circuit parameter is divided into four parts corresponding to different categories of parameters,

$$\partial f_{AC}/\partial\phi_{DCL} = \partial f/\partial\phi_{DCL} + (\partial f/\partial\phi_{DCS})(\partial\phi_{DCS}/\partial V_{DC})^T(\partial V_{DC}/\partial\phi_{DCL}) + (\partial f/\partial\phi_{ACS})(\partial\phi_{ACS}/\partial V_{DC})^T(\partial V_{DC}/\partial\phi_{DCL}) \tag{A3}$$

$$\partial f_{AC}/\partial\phi_{ACL} = \partial f/\partial\phi_{ACL} \tag{A4}$$

$$\partial f_{AC}/\partial\phi_{DCN} = (\partial f/\partial\phi_{DCS})[\partial\phi_{DCS}/\partial\phi_{DCN} + (\partial\phi_{DCS}/\partial V_{DC})^T(\partial V_{DC}/\partial\phi_{DCN})] + (\partial f/\partial\phi_{ACS})(\partial\phi_{ACS}/\partial V_{DC})^T(\partial V_{DC}/\partial\phi_{DCN}) \tag{A5}$$

$$\partial f_{AC}/\partial\phi_{ACN} = (\partial f/\partial\phi_{ACS})(\partial\phi_{ACS}/\partial\phi_{ACN}) \tag{A6}$$

Due to the conversion of Z parameters to S parameter, we have

$$\partial S/\partial\phi = (1 - S)(\partial Z/\partial\phi)(1 - S)/2Z_0. \tag{A7}$$

On the other hand, from [5] we know by adjoint analysis that

$$\partial Z/\partial\phi = -[\hat{\mathbf{p}} \quad \hat{\mathbf{q}}]^T (\partial Y/\partial\phi) [\mathbf{p} \quad \mathbf{q}], \tag{A8}$$

where Y is the admittance matrix of the small-signal equivalent circuit, and the definitions of $\hat{\mathbf{p}}$, $\hat{\mathbf{q}}$, $\mathbf{p}$ and $\mathbf{q}$ can be found in [5]. Therefore, in *R_SIMFET*, we first calculate $\partial Z/\partial\phi_{DCL}$, $\partial Z/\partial\phi_{ACL}$, $\partial Z/\phi_{DCN}$, $\partial Z/\partial\phi_{ACN}$, and $\partial Z/\partial V_{DC}$ by employing (A8), and then from (A7) we get

$\partial f/\partial\phi_{DCL}$, $\partial f/\partial\phi_{ACL}$, $\partial f/\partial\phi_{DCN}$, $\partial f/\partial\phi_{ACN}$, and $\partial f/\partial V_{DC}$, where

$$\partial f/\partial\phi_{DCN} = (\partial f/\partial\phi_{DCS})(\partial\phi_{DCS}/\partial\phi_{DCN})$$

$$\partial f/\partial\phi_{ACN} = (\partial f/\partial\phi_{ACS})(\partial\phi_{ACS}/\partial\phi_{ACN})$$

$$\partial f/\partial V_{DC} = (\partial f/\partial\phi_{DCS})(\partial\phi_{DCS}/\partial V_{DC}) + (\partial f/\partial\phi_{ACS})(\partial\phi_{ACS}/\partial V_{DC}) \ .$$

The final gradient of f_{AC}, i.e., (A3)-(A6), is assembled in the subroutine *FDF_R* by utilizing $\partial f/\partial\phi_{DCL}$, $\partial f/\partial\phi_{ACL}$, $\partial f/\partial\phi_{DCN}$, $\partial f/\partial\phi_{ACN}$ and $\partial f/\partial V_{DC}$ from *R_SIMFET*, and $\partial V_{DC}/\partial\phi_{DCL}$ and $\partial V_{DC}/\partial\phi_{DCN}$ available from *DCSIM*.

REFERENCES

[1] *Touchstone*, EEsof Inc., Westlake Village, CA 91362.

[2] *Microwave Harmonica User's Manual*, Compact Software Inc., Paterson, NJ, 07504, 1987.

[3] *McCAE*, Simulation Optimization Systems Research Laboratory and Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada, L8S 4L7, 1989.

[4] J.W. Bandler, R.M. Biernacki, S.H. Chen, J. Song, S. Ye and Q.J. Zhang, "Analytically unified DC/small-signal/large-signal circuit design", Simulation Optimization Systems Research Laboratory and Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada, Report SOS-89-5-R, 1989.

[5] S. Daijavad, "Design and modelling of microwave circuits using optimization methods", Ph.D thesis, McMaster University, Hamilton, Canada, 1986.

[6] K.S. Kundert and A. Sangiovanni-Vincentelli, "Simulation of nonlinear circuits in the frequency domain", *IEEE Trans. Computer-Aided Design.*, vol. CAD-5, 1986, pp. 521-535.

[7] J.W. Bandler, S.H. Chen, S. Ye and Q.J. Zhang, "Integrated model parameter extraction using large-scale optimization concepts", *IEEE Trans. Microwave Theory Tech.*, vol. 36, 1988, pp. 1629-1638.

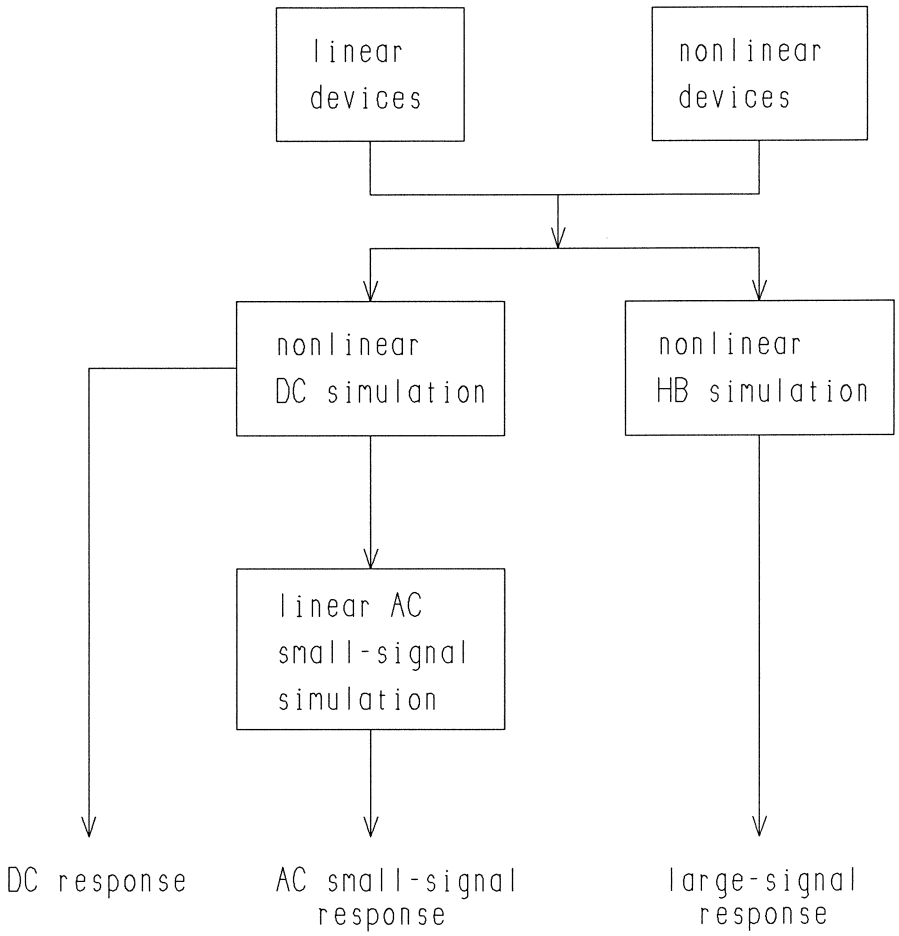

Fig. 1 Block diagram of logical connections among devices, simulators, and responses.

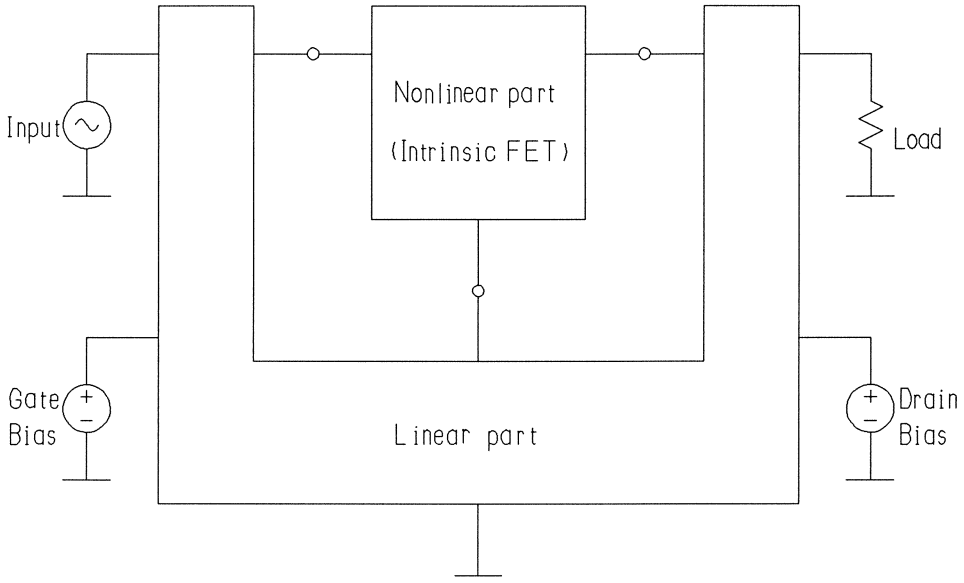

Fig. 2 A generic representation of a FET circuit for McCAE simulation.