Compact Device Modeling Using Velilog-A and ADMS

Laurent Lemaitre, Colin Mc Andrew, Wladek Grabinski
Laurent.Lemaitre@motorola.com

Motorola, Geneva Modeling Center

207 route de Ferney

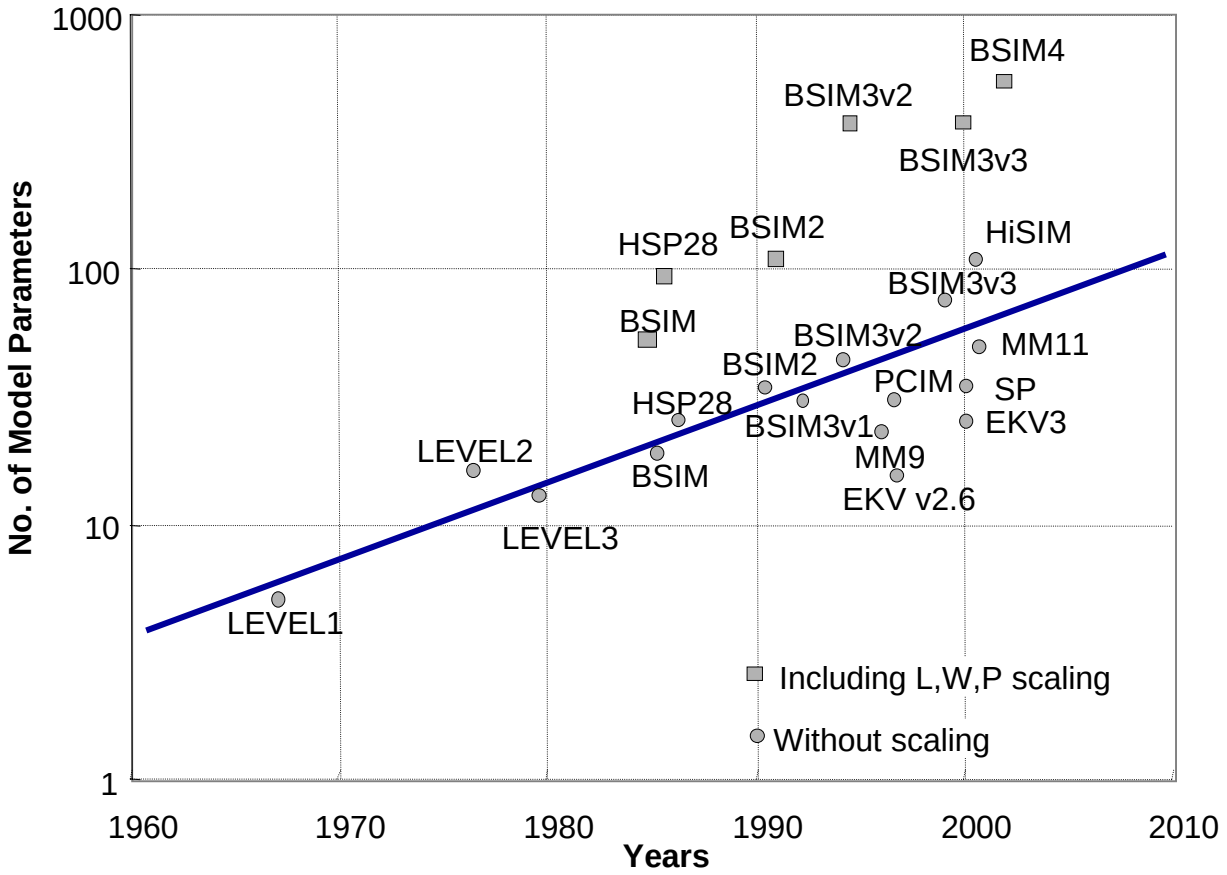
CH-1218 Le Grand Saconnex

Compact Device Modeling Using Velilog-A and ADMS

Outline:

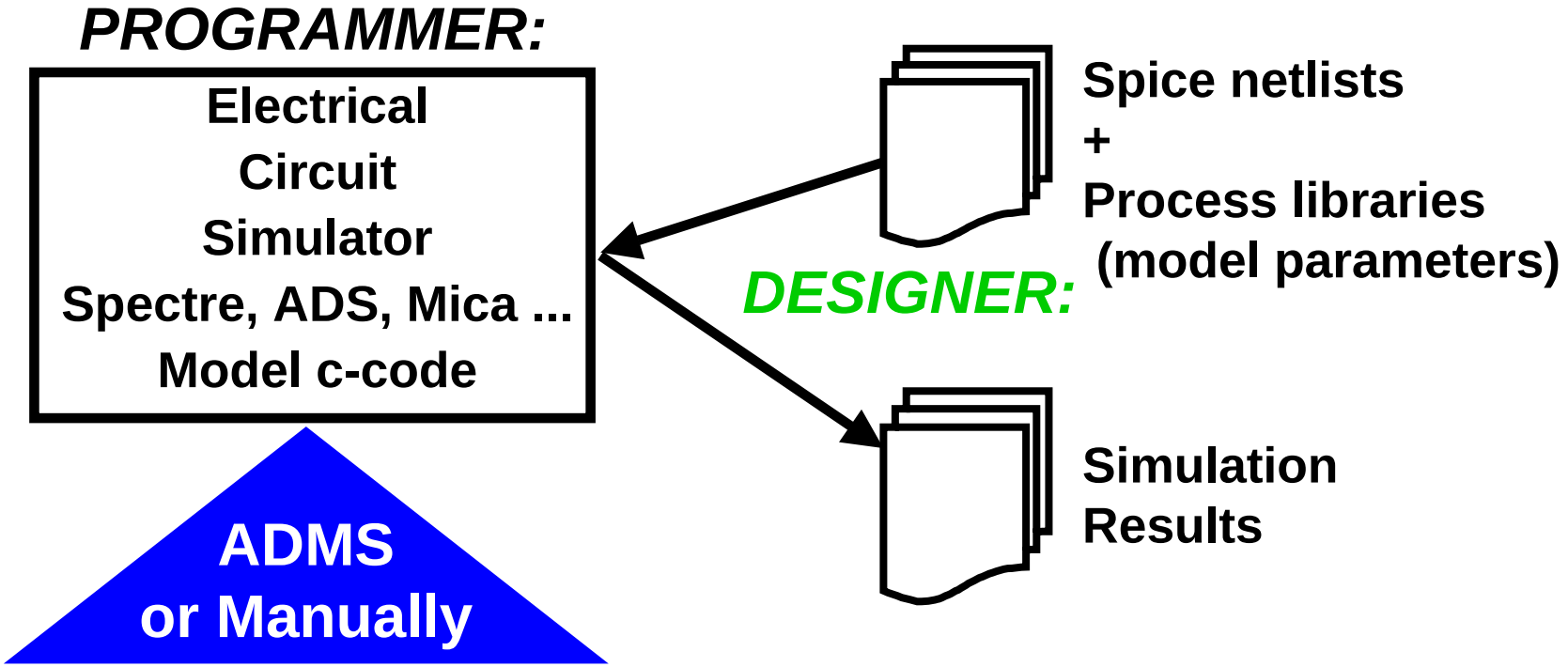
Developments of the Compact Models
Device Model Implementation Today
ADMS - Device Model Generator
Overview of the Device Model Generator
Example of Model Generation
Implementation of EKV 2.6 in ADS
Current Status - Availability
Conclusion

Developments of the Compact Models



- Number of DC model parameters vs. year of the model introduction
  - Significant growth of the parameter number that includes geometry (W/L) scaling
  - Most recent versions of the EKV, HiSIM, MM, PCIM and SP models are included

What is a Device Model?



DEVICE MODELING ENGINEER:

Built-in Device Model Equations
VBIC, EKV, SP, MOSCAP, R3, ..

Model Implementation Today

Device Modeling Engineer:

provides equations of new model to programmer. **No standard.**

Programmer:

hand-codes the model in source code of the electrical circuit simulator (most of the time the language is C). **No standard.**

- C code must comply with the Simulator Programming Interface. Much coding needs to be done again for each simulator.
- C code involves the manual computation and coding of partial derivatives. This process is tedious and error-prone.
- Feedback to the Device Modeling Engineer is made difficult. C-code is hard to read.
- The process is a barrier to model maintenance and enhancement.

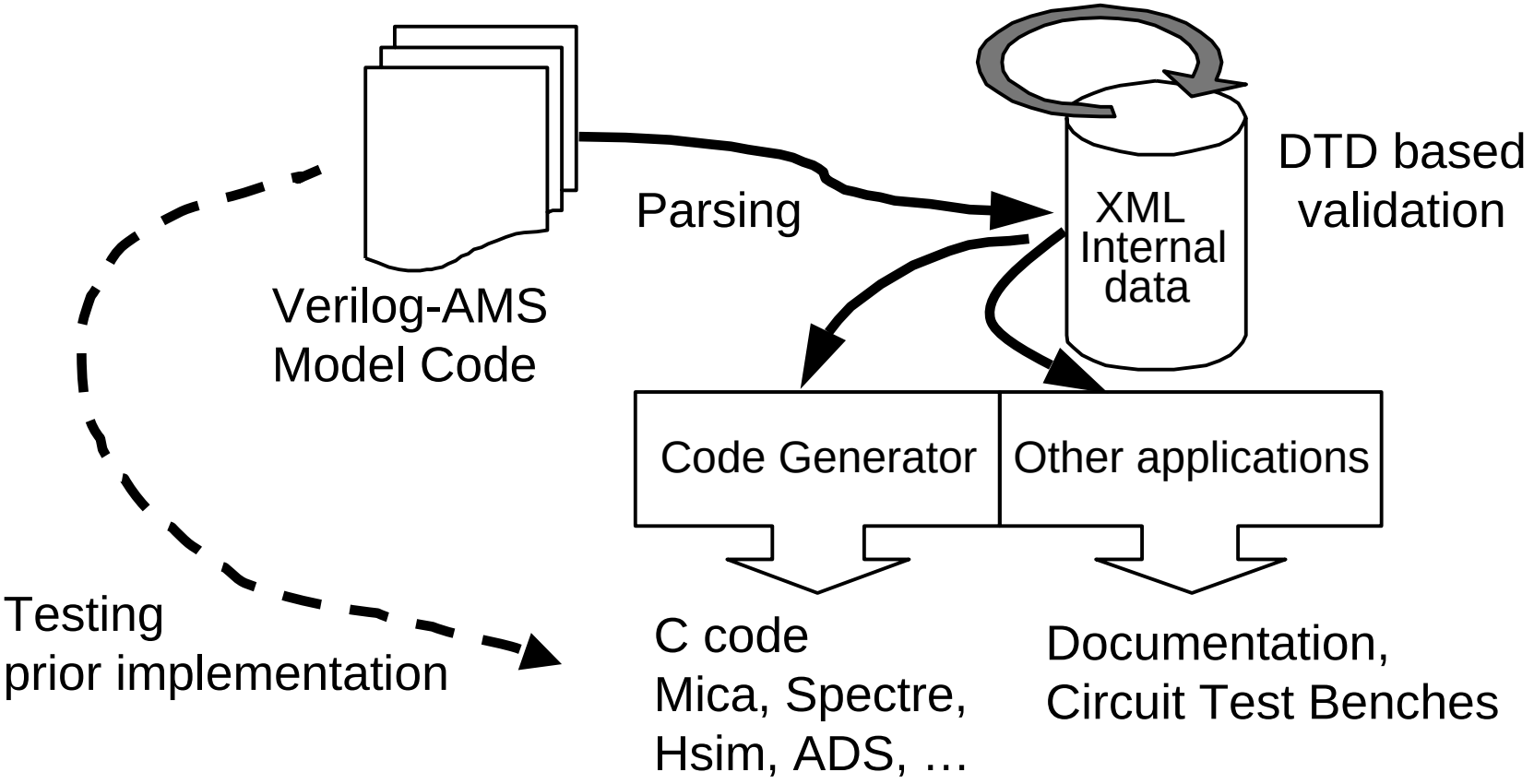
ADMS Approach

- ADMS is a Code Generator from a high-level language to ready-to-compile C code for simulators
- ADMS uses Verilog-AMS as a description language for device models. Verilog-AMS is a behavioral description language for mixed-signal electrical circuits.
  - Verilog-AMS code easy to read - no extra code specific to simulators.
  - Model can be easily and completely tested prior implementation!
- ADMS uses a subset of XML as an intermediate language between Verilog-AMS and Simulator Programming Interfaces.
  - Simplifies development of new features of ADMS and support of multiple simulators

ADMS Model Generator

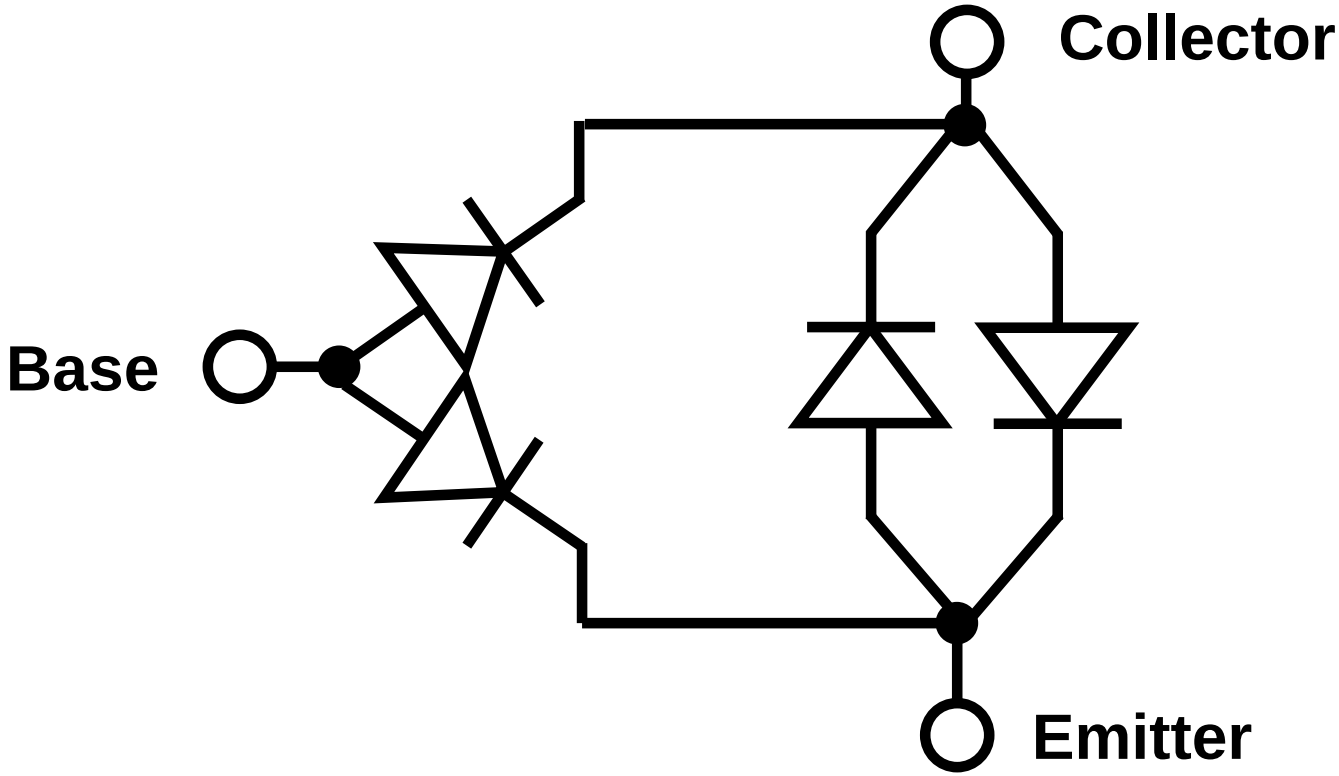
STANDARD 1:
MODEL LEVEL

STANDARD 2:
SIMULATOR LEVEL



Example – Symbolic View

Bipolar Device Model



- Branch assignment in the HBT model

Example – Verilog-AMS

```
‘define NPN +1
‘define PNP –1
module BIP (c,b,e);
// Nodes
   input       c,b; // input nodes
   output      e;// output nodes
   electrical  c,b,e; // all electrical
// Branches
   branch (b,c)        bc;
   branch (c,e)        ce;
   branch (e,c)        ec;
   branch (b,e)        be;
```

Example – Verilog-AMS (cont.)

```
// Parameters
   parameter   real      is      = 1.0e-16;
   parameter   real      bf      = 100;
   parameter   real      br      = 1;
   parameter   real      nf      = 1.0;
   parameter   real      nr      = 1.0;
   parameter   integer   type    = 'NPN;
// Variables
   real Tdev, Vtv;
   real Ifi, Ibf;
   real Iri, Ibr;
   real argf,expf;
```

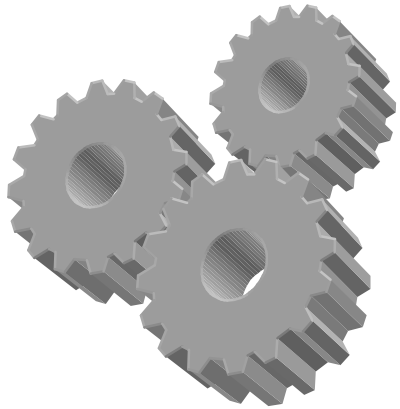
Example – Verilog-AMS (cont.)

```
analog begin // Analog section
    Tdev  =  $temperature;
    Vtv   =  1.380662e-23 * Tdev / 1.602189e-19;
    if ( type == ‘NPN ) begin
       argf = V(be) / ( nf * Vtv );
    end else if ( type == ‘PNP ) begin
...
    expf = exp(argf);
    Ifi  = is *(expf-1.0);
    Ibf  = Ifi/bf;
begin
       I(ce)  <+ Ifi;        // FORWARD Transport C-E
       I(be)  <+ Ibf;       // FORWARD Diode B-E
 end
```

Example – Code for Spectre

BIPOLAR
TRANSISTOR
in
VERILOG-AMS

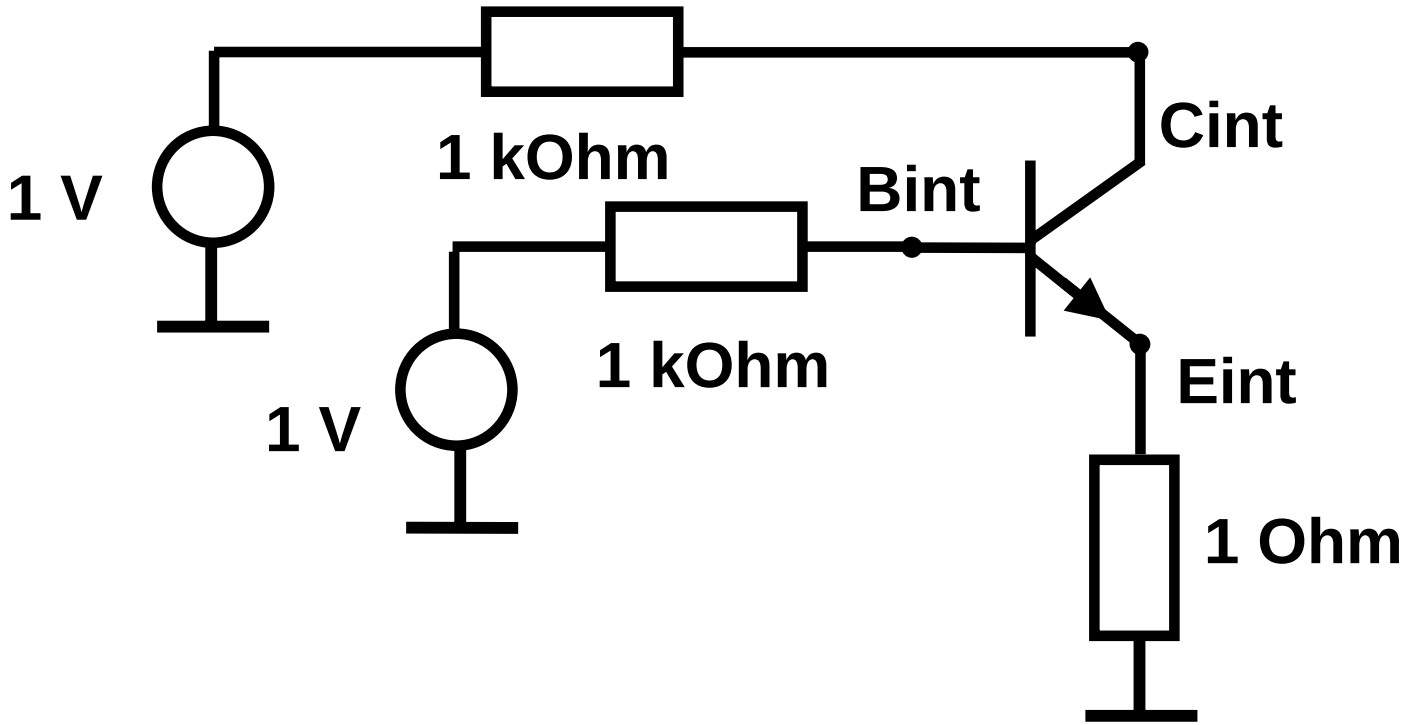
Run admsSpectre



SPECTREinterface.h
BIPdefs.h
BIPinitParameter.c
BIPloadJacobian.c
BIPevaluateStatic.c
BIPevaluateDynamic.c

Ready-to-compile C code

Example – Test-bench Circuit



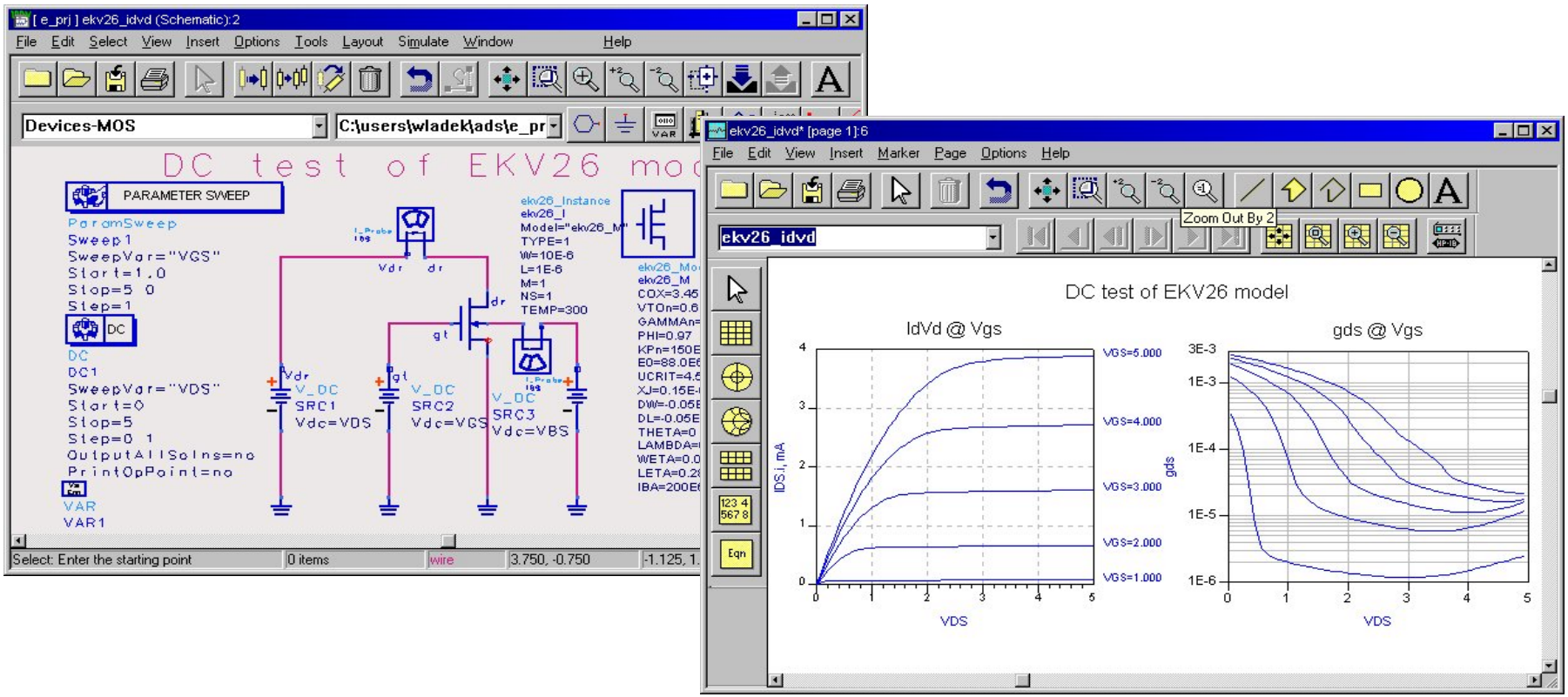
- Automatically Generated by ADMS

Example – Comparisons

Spectre	Mica	ADS
spectre (ver. 4.4.3). DC Analysis `opPoint' Operating at T = 27 C. V(Bint) = 650.428 mV V(Cint) = 921.346 mV V(Eint) = 79.0034 mV **I(vb:p) = -349.572 uA** **I(vc:p) = -78.6538 mA**	Using mica1.2.0 DC analysis Note: temp set to 27 v#Bint = 650.4275mV v#Cint = 921.3462 mV v#Eint = 79.00338 mV **vb#i = -349.572 uA** **vc#i = -78.6538 mA**	Ads(ver. "170") © Agilent Technologies DC Operating Point: V(Bint) = 650.428 mV V(Cint) = 921.346 mV V(Eint) = 79.0034 mV **vb.i = -349.572 uA** **vc.i = -78.6538 mA**

- ADMS generated model yields numerically identical results

Example - EKV ADS Implementation



The ADS EKV code was generated by ADMS tool using the EKV Verilog-AMS description: *legwww.epfl.ch/ekv/verilog-a*

Work in Progress

- Models available in Verilog-AMS:
  - R3 (Three Terminal Resistor)
  - MOSCAP (MOS varactor)
  - VBIC with self-heating (BJT)
  - SSIM (MOSFET, Motorola)
  - SP (MOSFET, PSU Prof. Gildenblatt)
  - EKV 2.6 (MOSFET, LEG-EPFL)
- All models are available in ADS (Agilent), Spectre (Cadence) and Mica (internal Motorola simulator)
- Some good results obtained with HSIM
- Looking into: Nanosim (Synopsys) and Eldo (Mentor)

Availability

- **ADMS** is written in the C language
- **ADMS** planned to be open-source
- Will come with public domain models (R3, MOSCAP, VBIC)
- On-going discussions with semiconductor and CAD vendors, and device modelers, on the best way to manage and control ADMS open-source

Conclusion

- **ADMS** = automatic implementation of compact models into circuit simulators
- **ADMS** automatically generates efficient, robust, correct-by-construction code
- **ADMS** has been successfully used for the integration of new device models into Mica, Spectre, HSIM and ADS
- Compact models are defined by Verilog-AMS, a standard high-level behavioral language