Simulation Standard

TCAD Driven CAD

A Journal for CAD/CAE Engineers

Maverick – Hierarchcal Full-Chip Extractor

Recent Significant Advances

Introduction

Maverick is a sophisticated full chip hierarchical netlist extractor [1]. It augments the existing ***CELEBRITY*** framework which includes state-of-the-art software for VLSI layout editing, hierarchical design rule checking and layout versus schematic comparison. The latest release of ***Maverick*** is equipped with many new interface features, including the ability to search for a net by name, and new engine upgrades, such as extraction of parameters of active devices. A new numerical procedure has been developed for resistance extraction of complex geometrical shapes.

Parameter Extraction

An important feature of any netlist extractor is the ability to accurately report geometrical active device parameters to the netlist.

For MOS technology, ***Maverick*** provides channel length (*L*), channel width (*W*), drain diffusion area (*AD*), source diffusion area (*AS*), perimeter of the drain junction including the channel edge (*PD*), perimeter of the source junction including the channel edge (*PS*).

For diodes, extraction is performed for the area of the diode (*AREA*) and perimeter of junction (*PJ*).

For capacitors, ***Maverick*** reports the capacitance combining inputs from the technology file with extracted geometries. To obtain the capacitance the technology file must include two technology constants: area capacitance (*AreaAttr* in aF/μm^2) and perimeter capacitance per unit length (*PerimAttr* in aF/µm). These are supplied together with both layer names for the capacitor. A typical technology file input might be:

```
Capacitance
  {
  Layer1 = "Pin1"
  Layer2 = "Pin2"
  AreaAttr = 60
  PerimAttr = 90
  }
```

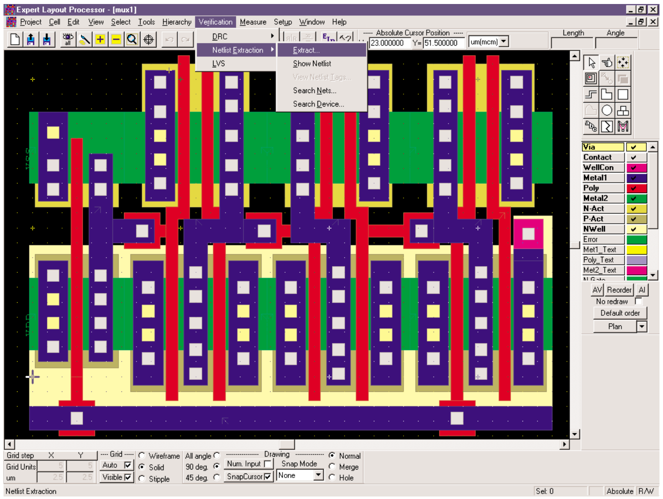


Figure 1. Calling Maverick from within the ***Expert*** layout editor.

Here *Pin1* and *Pin2* are layers of the capacitor.

Formula for calculating capacitance is standard

$$C = AreaAttr * A + PerimAttr * P,$$

where *A* and *P* are area and perimeter of capacitor.

Continued on page 2....

INSIDE

Calendar of Events 9

Hints, Tips, and Solutions 10

SILVACO INTERNATIONAL

Resistance Extraction

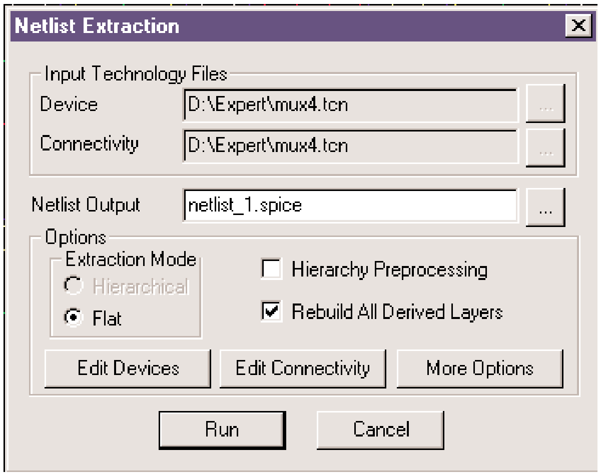
Resistance extraction for arbitrary shaped elements is a more sophisticated problem that the purely geometrical extraction active device geometries.

The resistance extraction routine always needs a sheet resistance value in Ohms/square to be specified for resistor *recognition* layer in technology file. The following fragment of an ***Expert*** technology file shows how this is done. Here the relevant row has been highlighted with bold type.

```
Layer
  {
  Name = "N-Act"
  Wire
    {
    MiterAngle = 20.00
    Width = 1.00
    Joint = LAYOUT
    End = EXTEND
    }
  Material
    {
    MaterialName = ""
    Resistivity = 555
    Permittivity = -1.00
    Thickness = 1.00
    }
  CIFName = "XXXX"
  Stipple = "0sparse11"
  Color = (200,190,0)
  ColorName = ""
  GDS2Num = 21
  GDS2DataType = 0
  Derive = ""
  Scope = CELLWISE
  Processing
    {
    DesignLayer = TRUE
    ProcessingStep = -1
    Operation = NONE
    StepCoverage = 0.00
    Undercut = 0.00
    Angle = 0.00
    }
  }
```

Here the layer N-Act is the resistor recognition layer with sheet resistance equal to 555 Ohm/square. The efficient numerical procedure implemented in ***Maverick*** provides high accuracy of extracted resistance value for a resistor of arbitrary configuration.

There are several geometrical shapes such rectangular resistors, bends, T-shaped and X-shaped resistive fragments of whose the resistance can be calculated using analytical formulas. ***Maverick*** uses these analytical expressions while dealing with resistors of these shapes. However all-angle geometry often results in even more complex shapes with multiple ports. ***Maverick*** is able to extract the resistance of these using more advanced numerical techniques.



Netlist Extraction ×

Input Technology Files

Device D:\Expert\mux4.tcn ...

Connectivity D:\Expert\mux4.tcn ...

Netlist Output netlist_1.spice ...

Options

Extraction Mode

○ Hierarchical

● Flat

□ Hierarchy Preprocessing

☑ Rebuild All Derived Layers

Figure 2. Netlist Extractor dialog box

Maverick can extract a full equivalent resistance network from the shape of which the ports are connected to *N* different electrical nodes. *N* can be arbitrarily large however its typical value is two. This requires solving the following partial differential equation

$$\nabla \bullet \sigma \nabla\phi = 0 \tag{1}$$

N-1 times over the resistive region[2]. Here $\sigma=1/\rho$, ρ is a piecewise constant sheet resistance and ϕ is electric potential. The whole boundary Γ of the region can be subdivided into two major parts, namely Γ_ϕ and Γ_q, where Dirichlet and Neumann conditions respectively are imposed.

$$\phi = \phi_0 \qquad \text{on } \Gamma_\phi \quad \text{(contacts)} \tag{2}$$

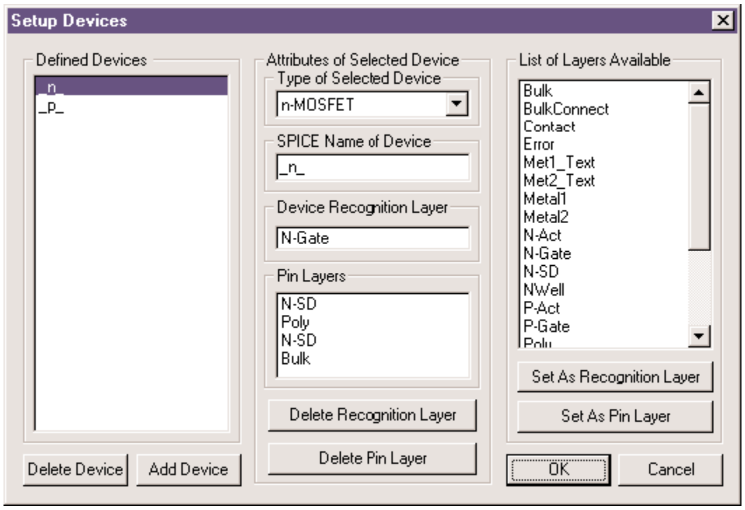
and

$$q = \frac{\partial\phi}{\partial n} = q_0 \qquad \text{on } \Gamma_q \quad \text{(other boundaries),} \tag{3}$$

where ϕ_0 and q_0 mean prescribed values of potential and its normal derivative respectively. In most cases sheet resistance is constant over the whole region under consideration. Then (1) reduces to Laplace's equation

$$\nabla^2 \phi = 0 \tag{4}$$

It has to be solved assuming one of the ports under unit potential and the rest of ports under zero potential. This should be done *N-1* times moving unit voltage to different port every time.



Setup Devices

Defined Devices

_n_
_p_

Attributes of Selected Device

Type of Selected Device: n-MOSFET

SPICE Name of Device: _n_

Device Recognition Layer: N-Gate

Pin Layers:
N-SD
Poly
N-SD
Bulk

List of Layers Available:
Bulk
BulkConnect
Contact
Error
Met1_Text
Met2_Text
Metal1
Metal2
N-Act
N-Gate
N-SD
NWell
P-Act
P-Gate
Poly

Figure 3. Graphical Interface for defining active devices.

Among the techniques for solving equations (2)-(4), two main categories of methods should be mentioned. For the first category of methods, one needs to discretize the whole region under consideration using spatial mesh and to solve resulting linear system of which the coefficient matrix is very large although sparse. Finite element method (FEM) [2], [3], [4] and finite difference method (FDM) [5] are well-known examples of such an approach. The large size of coefficient matrix is clearly a disadvantage of mesh-based methods. The second category of methods for extracting resistance implies reducing the governing partial differential equation to an equivalent integral equation formulation. In this case the system of linear equations to be solved is of significantly lower order although dense. Boundary element method (BEM) [6] is an integral equation approach, which does not need expensive computation of complicated Green's functions. Instead, it usually employs simple Green's functions, for instance, the free-plane one, as a fundamental solution.

Despite its simplicity, FDM faces difficulties in handling all-angle geometry. The former two approaches are more general. Advanced implementation of FEM needs sophisticated triangulation routine [7], which itself requires substantial computational resources. The geometrical engine in ***Maverick*** subdivides whole polygons into smaller pieces. This means that the problem of resistance extraction over an arbitrarily shaped polygon is usually stated for a relatively small region. Taking into account all the above mentioned considerations after intensive testing it has been found that BEM is more efficient for computing resistance of irregular regions.

There are two main formulations of BEM [8]: direct and indirect. Indirect or charge formulation is based on using a generalized charge as a variable. This can be found at the matching points by solving related discrete system of linear equations. Then boundary values of electric potential and its normal derivative can be evaluated. The conceptual disadvantage of indirect formulation is that the solution of the problem is obtained through computation of some intermediate variable namely the generalized charge. This sometimes bears no physical relation to the problem. In direct BEM potential and its normal derivative are used as variables. In addition direct BEM is less sensitive to smoothness of the boundary than indirect [8]. The direct formulation is implemented in ***Maverick***.

A new direct boundary element numerical procedure has been developed. This offers improved performance compared with [6] and can handle singularities in a quite natural manner avoiding generating excessive mesh points. An example of this approach is supplied below.

It can be shown that the error introduced by replacing ϕ and q by an approximate solution can be minimized by writing the following weighted residual expression

$$\int_{\Omega}(\nabla^2\phi)\phi^*\, d\Omega = \int_{\Gamma_q}(q-q_0)\phi^*\, d\Gamma - \int_{\Gamma_\phi}(\phi-\phi)q_0^*\, d\Gamma, \qquad (5)$$

where ϕ* is interpreted as a weighting function and

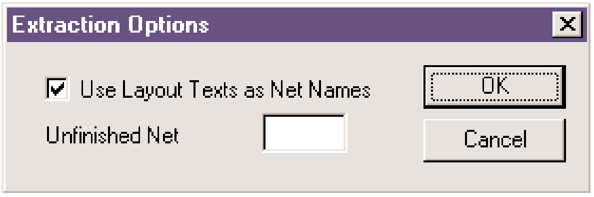
$$q^* = \frac{\partial\phi^*}{\partial n}.$$

Integrating (5) by parts twice and assuming ϕ* to be a fundamental solution to Laplace's equation we finally arrive at

$$c_i\,\phi_i + \int_{\Gamma}\phi q^*\, d\Gamma = \int_{\Gamma} q^*\phi\, d\Gamma. \qquad (6)$$

Here index *i* indicates the field point. The discussion of the coefficient c_i is left for the next section. Equation (6) expresses direct formulation of (4). For the points on Γ_ϕ potential is known and its normal derivative has to be found from (6). The opposite is true for the points on Γq.

If the problem is solved in a piecewise homogeneous region, the discrete analog of (6) contains two unknown values at the interface. In that case different regions are treated separately. This provides two different equations



Extraction Options

☑ Use Layout Texts as Net Names

Unfinished Net:

Figure 4. Netlist Extraction Options Dialog Box

for each pair of interface unknowns ϕ and q. These equations should be related to each other by imposing compatibility and equilibrium conditions as follows

$$\phi_1 = \phi_2 \quad \text{and} \quad \sigma_1 q_1 = -\sigma_2 q_2 ,$$

where subscripts appear to distinguish the values from two adjacent subregions.

It is clear that direct BEM is a diakoptic-based algorithm when applied to nonhomogeneous region. This intrinsic diakoptic property results in linear system of which the coefficient matrix is banded rather than fully dense which can be employed in developing efficient solver. On the other hand, this matrix has block structure, which can be utilized for multiprocessor handling.

To solve (6), the boundary is discretized into a series of boundary elements. n_k matching points (nodes) are assigned to kth boundary element. (For instance, $n_k = 1$ for zero-order constant element, $n_k=2$ for first-order linear element, $n_k = 3$ for second-order quadratic element and so on.) Particular placement of nodes within each element depends on type of that element. Then the fundamental solution has to be chosen. The free-plane Green's function

$$\phi^* = \frac{1}{2\pi} \mathrm{Ln}\left(\frac{1}{r}\right) \tag{7}$$

will be used throughout this paper, where r is the position vector of the source. Equation (6) is discretized as follows

$$c_i\,\phi_i + \sum_{j=1} \int_{\Gamma_j} \phi q^{*} d\Gamma = \sum_{j=1} \int_{\Gamma_j} q^{*} \phi d\Gamma\ . \tag{8}$$

Here N is the total number of boundary elements. For each kth boundary element one introduces local interpolation functions φ_m, $m=1, 2,..., n_k$, which have the unity value at mth node of the current element and zero values at all the other nodes of that element. The integrals along an element j on the left-hand side and the right-hand side of (8) can be written as

$$\int_{\Gamma_j} \phi q^* d\Gamma = \sum_{m=1}^{n_j} \left[\int_{\Gamma_j} \phi_m q^* d\Gamma \right] \phi_m\ , \tag{9a}$$

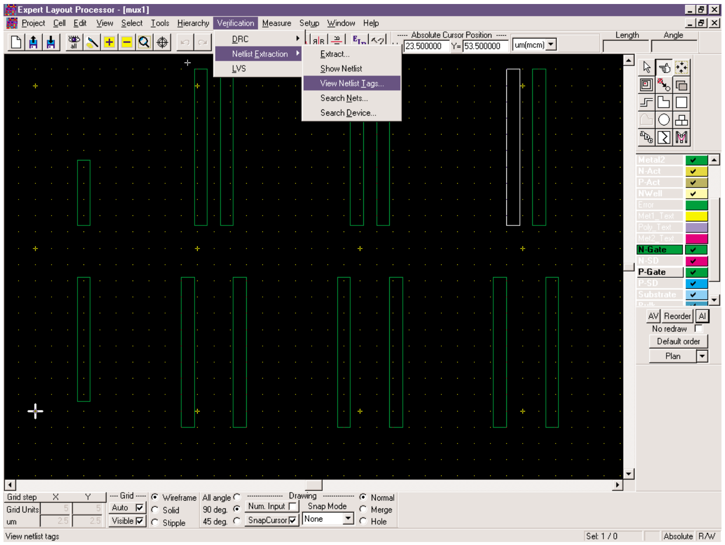
$$\int_{\Gamma_j} q \phi^* d\Gamma = \sum_{m=1}^{n_j} \left[\int_{\Gamma_j} \phi_m \phi^* d\Gamma \right] q_m \tag{9b}$$

respectively. Integrals on the right-hand sides of (9) can be evaluated numerically using Gaussian quadrature formulas. Hence, (8) represents the assembled equation for node i and can be written as

$$\sum_{j=1}^{N} H_{ij}\ \phi_j = \sum_{j=1}^{N} G_{ij}\, q_j\ , \tag{10}$$

where H_{ii} includes the term $c_i\phi_i$. The complete set of equations (10) may be rearranged to take the form of standard system of linear equations $Ax=b$, where the vector x contains all of the unknown values ϕ and q. The resulting linear system can be solved by LU-factorization.

The accuracy of boundary element solution depends on the set of interpolation functions assigned to each element. Using higher order elements it is possible to obtain more accurate resistance values. However higher order elements require more nodes and the order of linear system matrix is increased. The latter is rather disappointing because assembling and solving system of linear equations takes the major part of overall computational time. By defining computational efficiency as the primary objective, zero-order constant elements are employed in resistance extraction routine as default. This yields minimum possible number of nodes. Linear continuous element does not work here without special adjustments because of multiply defined normal derivative at the corner nodes. Linear and higher order elements can be used to enhance the accuracy of numerical solution if necessary. In contrast to [6], a concept of partially discontinuous element has been developed which decreases the total number of nodes by a half for linear element, by one third for quadratic element, etc. In the most cases constant element provides quick solutions which are accurate enough compared to third order solutions. In the worst case has been faced during testing, the maximum difference of zero-order solution from first-order one is less than 15%, while typical difference is below 5%.


Expert Layout Processor - [mux1]

Absolute Cursor Position: 23.500000 Y= 53.500000 um(mcm)

Grid step X Y
Grid Units 5 5
um 2.5 2.5

Grid: Auto ☑ Visible ☑

● Wireframe ○ Solid ○ Stipple

All angle ○ 90 deg. ● 45 deg. ○

Drawing: Num. Input □ SnapCursor ☑ Snap Mode: None

● Normal ○ Merge ○ Hole

View netlist tags

Sel: 1 / 0 Absolute R/W

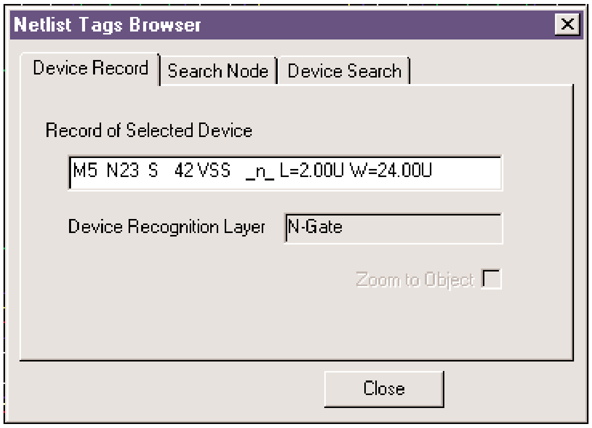
Figure 5. Calling the Netlist Tag Browser.

Device Setup in Maverick

The user is given two options to define devices to extract from a layout. First one is to use special statements and keywords in the ***Expert*** technology file. The example of such description would read like this:

```
Device
  {
  Type = "n-MOSFET"
  SpName = "_n_"
  DevLay
    {
    NameL = "N-Gate"
    TypeL = REC
    }
  DevLay
    {
    NameL = "N-SD"
    TypeL = PIN
    }
  DevLay
    {
    NameL = "Poly"
    TypeL = PIN
    }
  DevLay
    {
    NameL = "N-SD"
    TypeL = PIN
    }
  DevLay
    {
    NameL = "Bulk"
    TypeL = PIN
    }
  }
```

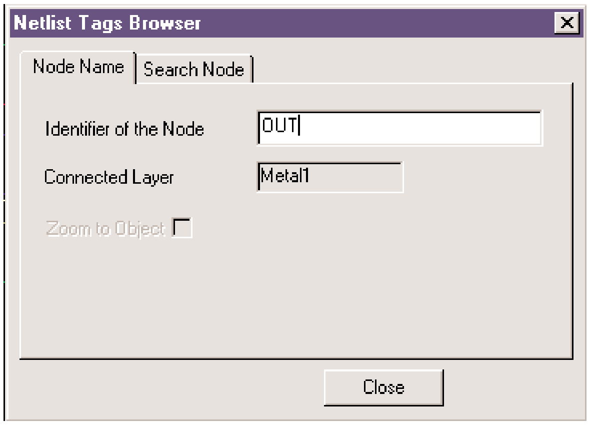
Here *Device* is a keyword followed by a number of attributes in parenthesis. The first of these attributes has *Type* as a keyword. Its value is a string. Allowed values of that string include "n-MOSFET", "p-MOSFET", "npn-BJT", "pnp-BJT", "diode", "resistor", "capacitor", and some others. They specify the type of a particular active or passive SPICE netlist circuit element. The other attributes define the SPICE model name of the corresponding model in string format. Its keyword is *SpName*. The next attribute occurs several times in a typical *Device* statement. Its keyword is *DevLay*. This construction implies two inner attributesof whose keywords are *NameL* and *TypeL*. *NameL* must be a string containing previously defined possibly derived layer. TypeL can take just one of two allowed values REC or PIN, specifying device recognition layer and pin layer respectively. It is recommended that the user list pin layers in the same order as they appear in SPICE statement. For instance, in the above example pin layers of n-channel MOSFET are listed in the sequence drain, gate, source, substrate exactly as SPICE netlist format requires. The presence of device recognition layer is obligatory. The number of pin layers depends on the type of device.



Record of Selected Device: M5 N23 S 42 VSS _n_ L=2.00U W=24.00U

Device Recognition Layer: N-Gate

Figure 6. Device Recognition Shape Dialog Box



Identifier of the Node: OUT

Connected Layer: Metal1

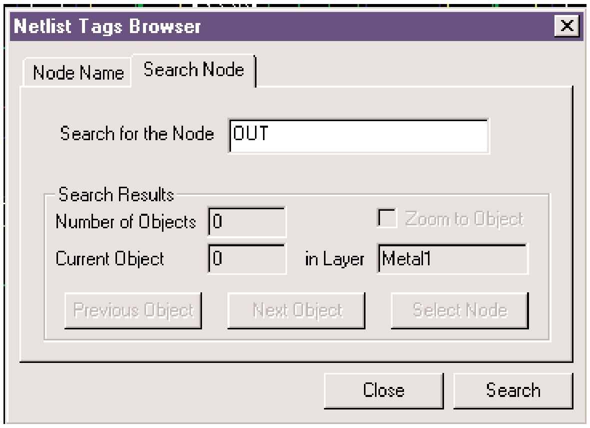
Figure 7. Node Name Dialog Box

The other way to set up devices is to use an intuitive GUI in ***Maverick***. To call the extractor, the user has to choose *Verification>>Netlist Extraction>>Extract* menu command, as it is shown in Figure 1. This action opens the dialog window presented in Figure 2.

Now to enter device settings users should click *Edit Devices* button. Then the user arrives in dialog window shown in Figure 3.

This window can be conditionally displayed as three columns. The left column contains the list of all devices defined for the project. The user is able to add and delete devices by clicking on corresponding button at the very bottom of this column. All the devices listed in this box have distinctive SPICE model names. This means that one can define several kinds of the same device type by referring it to different SPICE models.

Once a particular device has been selected in the left column, its attributes appear in four windows in the middle column of the dialog box. The top window in the middle column is dropdown list box. The user can



Search for the Node: OUT
Number of Objects: 0
□ Zoom to Object
Current Object: 0
in Layer: Metal1

Figure 8. Search Node Property Page.

select device type from the set of predefined ones. The second window from the top is an edit box. The user can type directly in this window to specify SPICE name of selected device. The third window from the top contains recognition layers for selected device. The user can delete this by clicking on the corresponding button. The bottom window in the middle column is ordered list of pin layers of the device selected in the left column. Users can delete either of them by selecting particular layer and then clicking on the button at the very bottom of this column.

The right column includes the list of all the layers defined in the project. The user can select any of them and then set it as device recognition layer or as pin layer for the device selected in the list in the left column of the dialog box by clicking corresponding button under layers list box.

An example of the operations required to add a device to the extraction follows. After clicking the Add Device button, a new device is added to the list box and becomes selected. Its type is initially set up as p-MOSFET and SPICE name appears as noname_1, noname_2, etc. The windows for device recognition layer and for pin layers are empty. The user is responsible for filling-in all the four windows in the middle column with the necessary information.

Working with dialog shown in Figure3 the user creates the same information pattern as the technology file specifies. It is recommended to use dialog capabilities rather than manual updating the technology file because several consistency checks are performed when the OK button is clicked. For instance, ***Maverick*** verifies the presence of device recognition layer and the correctness of the number of pin layers specified.

Using Layout Texts as Net Names

It is common practice to use texts such as VDD and VSS for naming electrical nodes. ***Maverick*** provides the flexibility to add user defined names. First of all one should place the text to be accepted as net name into the connected layer from that net. It is recommended to click the *More Options* button in extractor dialog box (Figure 2) which results in the dialog box presented in Figure 4, and make sure the control option *Use Layout Texts as Net Names* is checked. While accepting layout texts during extraction, ***Maverick*** can encounter several problems. A typical problem is when two or more different texts belong to the same net. The current version of ***Maverick*** assigns new name to such a node. That new name would appear as,

```
MULTILABEL-35,
```

where MULTILABEL- is a key word and 35 is an example of internally generated node identifier. In addition a warning is generated and added to *message file.*

The other typical problem is when the *same text is used for different nets.* ***Maverick*** generates different name for one of two equally labeled nets and puts an appropriate warning into the message file. However the user often needs to do so while designing subcells where power supply or ground may be represented by multiple shapes that are intended to be connected somewhere else in parent cell. In this case the user can specify a string of special symbols in *Unfinished Net Symbols* edit control of *Extraction Options* dialog box. Any of these symbols can then be used to indicate intentionally unfinished nets by simply adding them to the text from

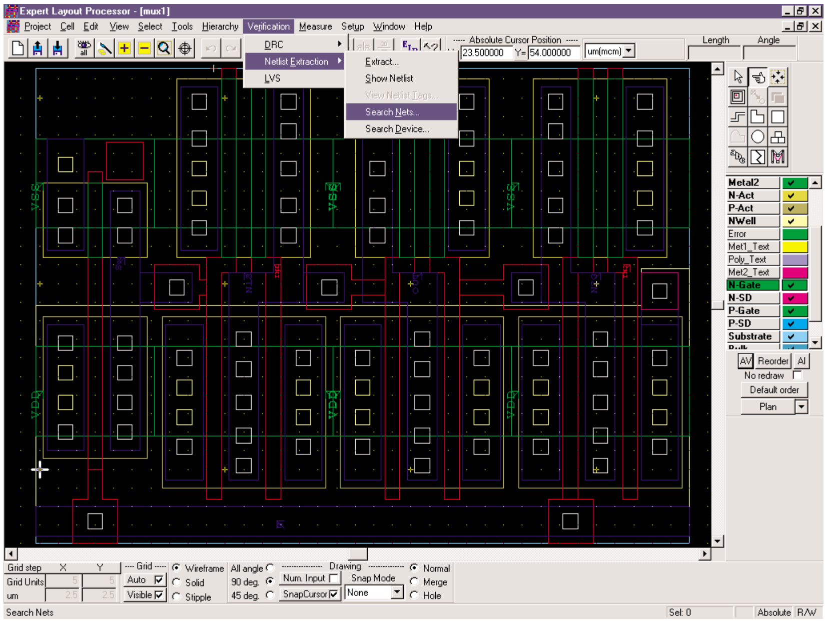
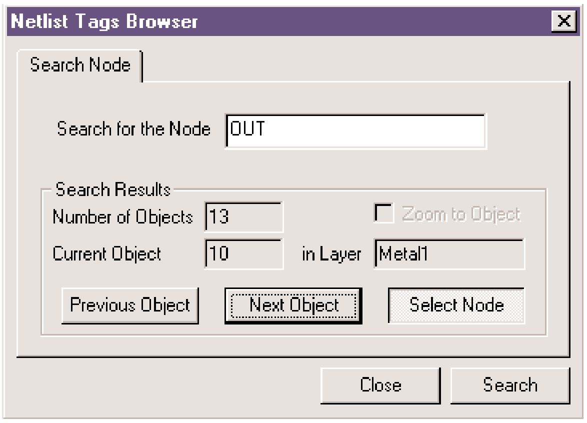


Figure 9. Calling the Search Node Feature from Maverick



Netlist Tags Browser

Search Node

Search for the Node: OUT

Search Results

Number of Objects: 13

□ Zoom to Object

Current Object: 10

in Layer: Metal1

Figure 10. Calling Search Node Feature from Maverick

either side. For example, if user specified the string "!:%", then the following texts mean different shapes of the same net *VDD*

```
VDD
VDD:
:VDD
!VDD%
%VDD:
 VDD!
```

The name *VDD* without one of the above specified symbols may only be used once. Those special symbols will never appear in output netlist so that in the previous example the only output node is *VDD*.

Connectivity records

Once the extraction is completed, connectivity information is also been added to the property. The user can select a single shape of any connected layers. After this the Menu **Verification >> View Netlist Tags** calls the dialog box presented in Figure 7.

This box includes the name of the node and the name of the layer from which the shape has been selected.

Search for Node by Name

After extraction has been completed, the user can obtain all the shapes connected to a in specific net using *Search Node* property page of *Netlist Tag Browser* presented in Figure 8. This page may be easily activated when one is viewing netlist tags as has been shown in Figures 6 and 7.

This page can also be activated directly from the ***Maverick*** main menu asseen in Figure 9.

To perform a search the user has to specify the name of the node to be found within corresponding edit control. Then it is necessary simply to click the *Search* button. After that the number of objects representing the specified node and zero-based index of the current object appear in corresponding gray windows. This is shown in Figure 10.

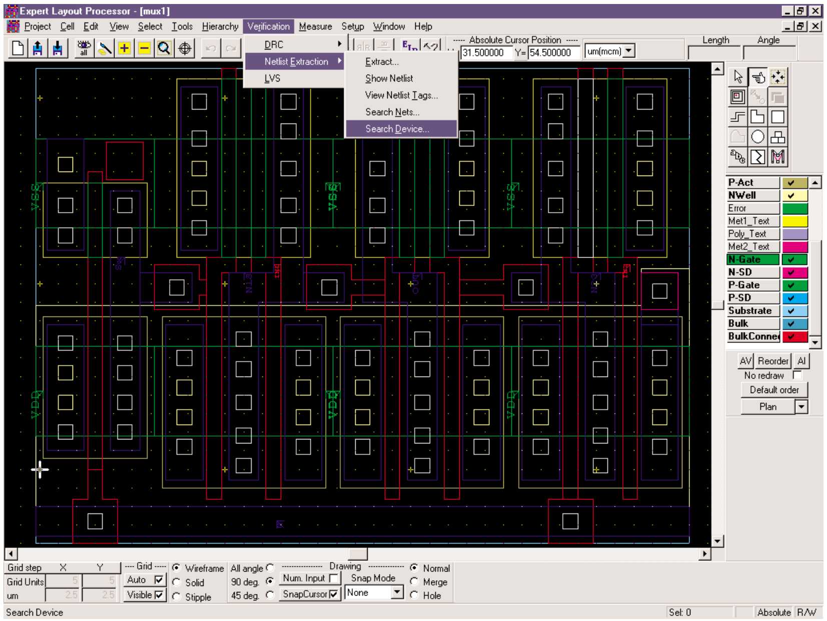
Furthermore, the current selection is highlighted in the layout as a bright white hatched object. Using *Previous Object* and *Next Object* buttons, the user can browse through all the objects and view them in the layout.

Viewing Netlist Tags

During execution ***Maverick*** adds certain records to the property lists attached to several geometrical primitives apart from creating netlist file. Once the extraction has been completed, this information becomes available to the user. The user can view these records by selecting a single shape of a noninstance layer, and further choosing special Menu item **Verification >> View Netlist Tags**, as shown in Figure 5.

Device Tags

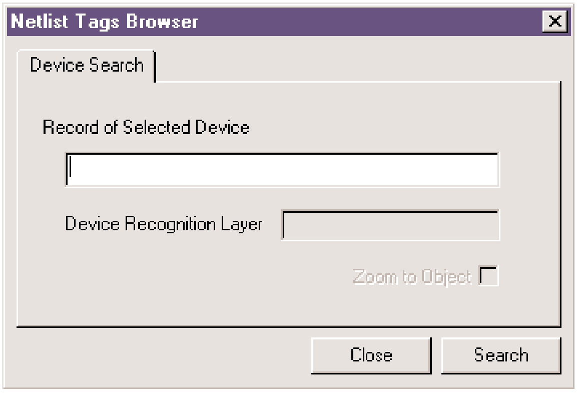
If the shape from the device recognition layer has been selected as in Figure 5, the user is presented with the dialog box in Figure 6. This box includes netlist record of the selected device. This dialog box provides the name of related device recognition layer.



Expert Layout Processor - [mux1]

Absolute Cursor Position X= 31.500000 Y= 54.500000 um(mcm)

Figure 11. Calling Search Device Feature Directly from Expert



Netlist Tags Browser

Device Search

Record of Selected Device

Device Recognition Layer

Zoom to Object ☐

Close | Search

Figure 12. Search Device Property Page

If the user has selected the shape in a connected layer, the name of the node given to that shape by ***Maverick*** is automatically substituted into *Search for the Node* field of *Netlist Tag Browser.*

The *Select Node* button allows selecting / deselecting of all the shapes of the specified node at once. After the *Netlist Tag Browser* is closed, the status of selection of the node is preserved.

Search for Device

The ***Expert*** Menu option shown in Figure 11 opens the dialog window presented in Figure 12. The user can also bring up this window by selecting the *Device Search* property page while viewing device tag as shown in Figure 6. The user can now type the name of any device present in the netlist within edit control and then click on the *Search* button. For example if the string "M10" has been typed for the multiplexer layout shown in Figure 11, then the result of search might be as shown in Figure 13. After the search is done, the edit control now contains the full SPICE record of the device, the shape of device recognition layer associated with that device is highlighted in the layout.

Acknowledgement

We recognize the contribution of S. Bielawski, Belarusian State University, to the content of this article.

References

[1] A.Azarenok and V.Feinberg, "Maverick: hierarchical netlist extractor for PC platforms", *Simulation Standard,* vol. 9, pp. 1-6, Jun 1998.

[2] C.M.Sakkas, "Potential distribution and multi-terminal DC resistance computations for LSI technology", IBM J. Res. Develop, vol. 23, pp. 640-651, Nov 1979.

[3] E.Barke, "Resistance calculation from mask artwork data by finite element method", in Proc. 22nd DAC, Jun 1985, pp. 305-311.

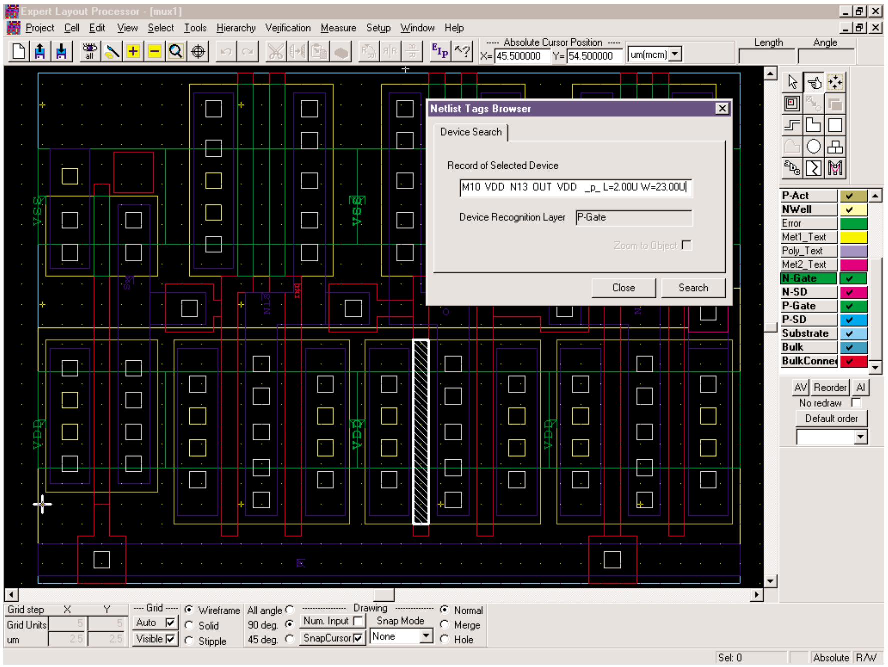
[4] T.Mitsuhashi and K.Yoshida, "A resistance calculation algorithm and its application to circuit extraction", IEEE Trans. Comput.-Aid.Design, vol. CAD-6, pp. 337-345, May 1987.

[5] M.Glez Harbour and J.M.Drake, "Calculation of multiterminal resistances in integrated circuits", IEEE Trans. Circ. Syst., vol. CAS-33, pp. 462-465, Apr 1986.

[6] Z.Wang and Q.Wu, "A two-dimensional resistance simulator using the boundary element method", IEEE Trans. Comput.-Aid.Design, vol. CAD-11, pp. 497-504, Apr 1992.

[7] A.J.Kemp, J.A.Pretorius, and W.Smit, "The generation of a mesh for resistance calculation in integrated circuits", IEEE Trans.Comput.-Aid.Design, vol. CAD-7, pp. 1029-1037, Oct 1988.

[8] C.A.Brebbia, J.C.F.Telles, and L.C.Wrobel, Boundary Element Techniques. Berlin: Springer-Verlag, 1984.



Absolute Cursor Position X= 45.500000 Y= 54.500000 um(mcm)

Netlist Tags Browser

Device Search

Record of Selected Device: M10 VDD N13 OUT VDD _p_ L=2.00U W=23.00U

Device Recognition Layer: P-Gate

Zoom to Object ☐

Close | Search

Figure 13. Results of the Device Search highlighted in the layout.

Calendar of Events

June

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21 DAC ‘99 - New Orleans

22 DAC ‘99 - New Orleans

23 DAC ‘99 - New Orleans

24

25

26

27

28 Adv. Microelectronics, France

29 Adv. Microelectronics, France

30 Adv. Microelectronics, France

July

1 Adv. Microelectronics, France

2 Adv. Microelectronics, France

3 Adv. Microelectronics, France

4 Adv. Microelectronics, France

5

6

7

8

9

10

11

12 NSREC - Norfolk, VA
INSREC, Oxford

13 NSREC - Norfolk, VA
INSREC, Oxford

14 NSREC - Norfolk, VA
INSREC, Oxford

15 NSREC - Norfolk, VA

16 NSREC - Norfolk, VA

17

18

19 MSE - Arlington, VA

20 MSE - Arlington, VA

21 MSE - Arlington, VA

22

23

24

25

26

27

28

29

30

31

Bulletin Board

See Silvaco at DAC ‘99 Booth #514!

A large number of Silvaco new line engineers will be on hand at our booth for DAC ‘99 - New Orleans, LA from June 21-23. Stop by and get a demonstration in the latest TCAD Driven CAD. Highlights include:

- ***Expert*** and ***Savage*** - NT design system.
- ***Scholar*** - NT schematic editor
- ***Guardian*** - hierachical LVS
- ***Maverick*** - hierachical post-layout extractor
- parallel ***SmartSpice*** for UNIX and NT
- ***EXACT*** - technology driven interconnect parasitic extraction

NSREC ‘99

Silvaco enters the final frontier of success at the National Space and Radiation Effects Conference. Take your designs where no engineer has gone before at this July 12-16 conference in Norfolk, VA.

Spotlight on Systems Education

MSE ‘99 and Silvaco International join forces for the premier conference for educators in Microelectronic Systems. Our East Coast team will be present to continue our commitment to education.

For more information on any of our workshops, please check our web site at ***http://www.silvaco.com***

The Simulation Standard, circulation 17,000 Vol. 10, No. 4, April 1999 is copyrighted by Silvaco International. If you, or someone you know wants a subscription to this free publication, please call (408) 567-1000 (USA), (44) (1483) 401-800 (UK), (81)(45) 341-7220 (Japan), or your nearest Silvaco distributor.

Hints, Tips and Solutions

Mikalai Karneyenka, Applications and Support Engineer

Q: I want to draw several long wires, with precise positioning of both ends. I am used to doing this as follows:

- **zoom to the start position and start drawing;**
- **zoom to the whole view;**
- **zoom to the end position and finish drawing.**

Can *Expert* do something similar: opening two windows, starting drawing in one window and finishing in another one?

A: Any object creation must be performed within one window. However the Multiview feature in ***Expert*** may help you with this operation. You can do as follows:

- Run View>>Multiview command
- zoom to the start position with sufficient resolution and start drawing a wire. You will see something similar to Figure 1.
- In the Map View window, grab the frame corresponding to the current window... (Figure 2.)
- ...and move it to the end position.
- You can finish drawing the wire with good resolution for precise positioning (Figure 3).

In the same way you may use the built-in ruler to measure long distances on the chip.

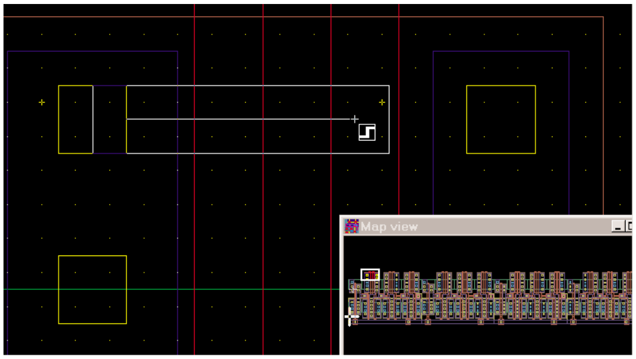


Figure 1. Drawing the start of a long line in ***Expert.***

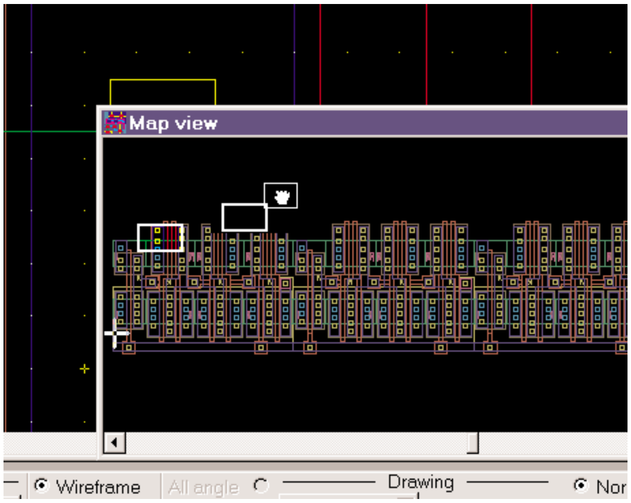


Figure 2. Moving the window frame in the middle of drawing a long line.

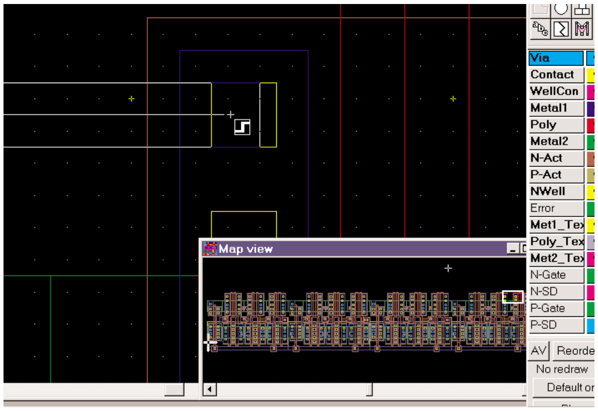
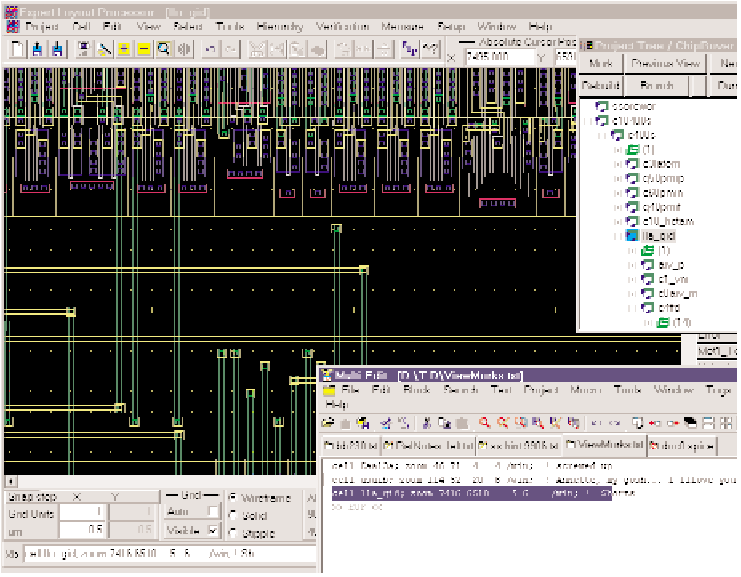


Figure 3. Completing the edit of a long line in ***Expert***.

Q: I would like to save a definable window while editing a cell. Once you have clicked the command button, you can then define a window by using the mouse and cursor. After the window is defined it asks for a name for the new window. After you have assigned a name the function is completed and the window is saved for recall. To recall use a new command such as "Window Restore", and that saved window would immediately be viewed.

A: ***Expert*** does not need any new buttons to do this. In fact, ***Expert*** allows you to do much more using xi-scripts and command-line input, as shown below. First, the sequence to save and restore "windows" is defined as follows:



```
cell Cell3a; zoom 46 71   4   4 /win;   ! screwed up
cell [illegible] zoom 114 52   20   6 /win;   ! [illegible]
cell lla_gid; zoom 7416 6510    5 6    /win; !  Shorts
>> EOF <<
```

```
cell lla_gid; zoom 7416 6510   5  6   /win; ! Sh
```

Figure 4. Using a script to define a particluar window view.

Step 1.

- Turn on the option "Activity log" in Setup>>General. Now all your actions will be stored in the log file.
- Open the required cell and zoom to a window using the command View>>Zoom to Window.
- Zoom to several other windows;
- Open another cell and zoom to still other windows, and so on.

Now your "bookmarks" are stored in the activity log This name is also specified in the Setup>>General.

Step 2.

- Open the activity log in your favorite text editor, cut and paste, add comments, so that your file can look like the one visible in Figure 4.

Step 3

- Select the required line in the text editor and press Ctrl-C to copy it into WinNT clipboard.
- Go to the Xi-command line of Expert and press Ctrl-V to paste the command into the command line.
- Look at Figure 4 once more. If you hit "Enter" key, Expert will execute two commands shown in the command line:
- The cell "lla_gid" will be opened
- its layout will zoom to box with position (7416 6510) and sizes (5 6).

You may even go further. You may save the following xi-script as an activity log:

```
proj "bigchip.eld";
cell lla_gid;
zoom /win 7416 6510 5 6;
layer "comments";
box 7418.5 6511.8 0.2 0.2;
```

The activity log can be emailed to your colleagues with appropriate comments. He can start Expert, run your script, and zoom into the place in question marked by a box in the layer "comments". This can replace the manual process of plotting pieces of the layout and mark it with red marker to pass your corrections or questions.

The range of viewing tasks that Certainly, no number of new menu buttons will give you such flexibility.

Q: When you are in Single Mode of Selection if you use "select Instances" twice, the first instances selected stay selected.

A: The Single mode acts only for selection of objects one by one (menu command Select>>Object). It is not applicable to other types of selection, like Select by box, Select Layer, or Select instances.

Q: What does the syntax

```
    s= &L&"->"&a;
```

from one of the *Expert* script examples mean?

A: This command computes a string to be displayed or output into a file as follows:

&L, &a mean conversion of the values of variables 'L', 'a' into strings.

```
        &L & "->"
```

means concatenation of string values '&L' and '->' into one string.

So, if L = 28.3, a= 47.1, you will obtain the string "28.3->47.1".

Join the Winning Team!

Silvaco Wants You!

l Process and Device Application Engineers

l SPICE Application Engineers

l CAD Applications Engineers

l Software Developers

fax your resume to:
408-496-6080, or
e-mail to:
jobs@*silvaco.com*

Opportunities worldwide for apps engineers: Santa Clara, Phoenix, Austin, Boston, Tokyo, Guildford, Munich, Grenoble, Seoul, Hsinchu. Opportunities for developers at our California headquarters.

SILVACO
INTERNATIONAL

USA Headquarters

Silvaco International
4701 Patrick Henry Drive
Building 2
Santa Clara, CA 95054
USA

Phone: 408-567-1000
Fax: 408-496-6080

sales@silvaco.com
www.silvaco.com

Contacts:

Silvaco Japan
jpsales@silvaco.com

Silvaco Korea
krsales@silvaco.com

Silvaco Taiwan
twsales@silvaco.com

Silvaco Singapore
sgsales@silvaco.com

Silvaco UK
uksales@silvaco.com

Silvaco France
frsales@silvaco.com

Silvaco Germany
desales@silvaco.com

Products Licensed through Silvaco or e*ECAD

e*ECAD
Vendor Partner