Sentaurus Structure Editor
2. Generating 2D Boundaries

Objectives

- To generate 2D boundaries.

2.1 Overview

This section is intended to familiarize you with some fundamental functions that Sentaurus Structure Editor supports. For this purpose, a 2D SOI MOSFET device, similar to the one shown in Figure 1, will be created step by step, using basic 2D geometric operations.
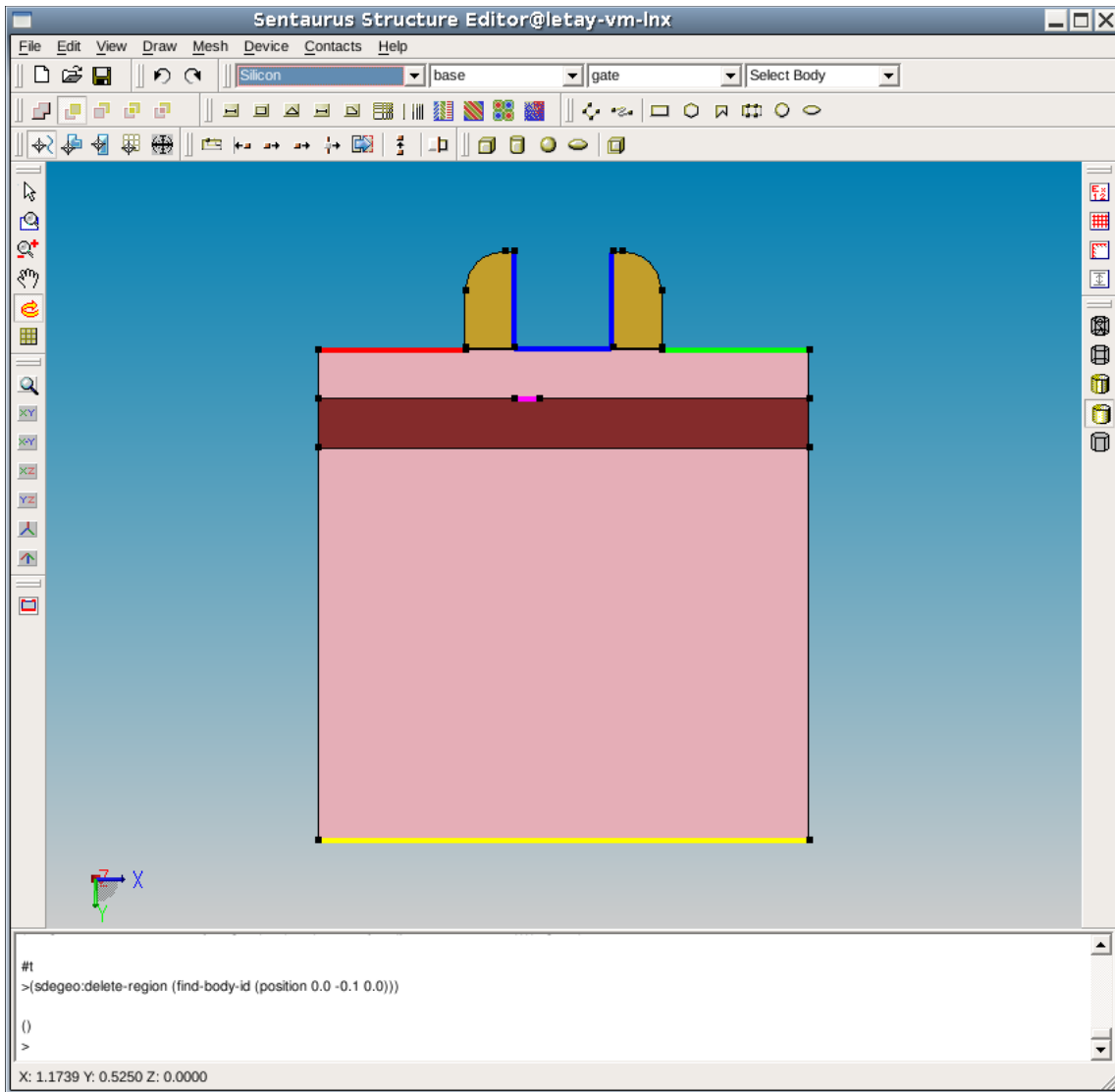
Figure 1. Two-dimensional SOI MOSFET device. (Click image for full-size view.)

2.2 Reinitializing Sentaurus Structure Editor

To start a new object and discard all objects that have been previously defined:

- **File** > **New**, or Ctrl+N, or click the corresponding toolbar button (⬜).

The corresponding Scheme command is:

```
(sde:clear)
```

> This command restarts the Sentaurus Structure Editor environment and resets most of the internal variables to their default values. However, some internal variables may remain defined at their previously set values.

2.3 Exact Coordinates Mode

In Sentaurus Structure Editor, geometric objects can be drawn manually. However, for most applications, it is convenient (and sometimes necessary) to specify explicitly the coordinates of the object, for example, to allow for precise lineups of different regions.

To do this, it is necessary to activate the Exact Coordinates mode. The mode is, by default, inactive when Sentaurus Structure Editor initially starts.

To activate the mode:

- **Draw** > **Exact Coordinates**.

A check mark now precedes the **Exact Coordinates** command indicating the mode is active. When the mode is active, all subsequent command operations that involve the placement of any object will display a dialog box in which exact coordinate values can be entered for the object being edited.

2.4 Selecting Materials

All materials used by Sentaurus Structure Editor are accessible from the Material list.

To set the material to be used for new objects:

- From the Material list, select the required material, for example, Silicon.

Figure 2. Material list showing available materials.

Until a new material type is selected, all newly added regions consist of the material type that is currently being selected.

2.5 Selecting the Default Boolean Expression

In some applications, a device can consist of only one region. In most other cases, a device consists of multiple regions. When the GUI is used to build a device with multiple regions, the later-added regions may intersect existing regions. If this occurs, a predefined scheme is required to resolve the overlapping region.

Sentaurus Structure Editor includes several predefined schemes. For example, to select the **New**

Replaces Old scheme in which newly added regions replace old regions where they overlap:

- **Draw** > **Overlap Behavior** > **New Replaces Old**, or click the toolbar button ( ).

The corresponding Scheme command is:

```
(sdegeo:set-default-boolean "ABA")
```

2.6 Creating Rectangular Regions

To create a rectangular, silicon substrate region:

1. **Draw** > **2D Create Tools** > **Rectangle Region**, or click the toolbar button ( ).
2. Drag the pointer to draw a rectangle in the view window.
3. In the Exact Coordinates dialog box, enter (-0.5 0.0), (0.5 1.0) in the corresponding fields and click **OK**.
4. If the rectangle does not fit the view window, **View** > **Zoom to Extents** to rescale the view.
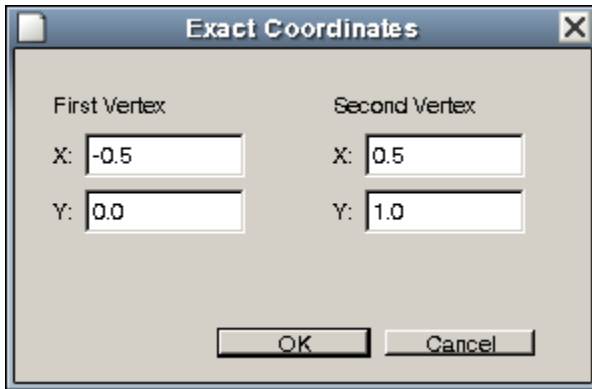
Figure 3. Exact Coordinates dialog box with values entered.

The corresponding Scheme command for the drawing operation is:

```
(sdegeo:create-rectangle
  (position -0.5  0.0   0.0) (position 0.5  1.0   0.0) "Silicon" "region_1")
```

Other device regions can be created in a similar way as the substrate region. However, different regions often consist of different material types. Therefore, before drawing a region, select the required material type from the Material list (see Section 2.4 Selecting Materials).

Follow these rules to create the gate oxide region, the nitride spacer region, the polysilicon gate region, and the buried oxide region. Use the coordinates as follows:

Region	Material	Coordinates
Gate oxide	SiO2	(-0.2 -40e-4), (0.2 0.0)
Nitride spacer	Si3N4	(-0.2 -0.2), (0.2 -40e-4)
Poly gate	PolySi	(-0.1 -0.2), (0.1 -40e-4)

Buried oxide	SiO2	(-0.5 0.1), (0.5 0.2)

After these steps, the device structure is shown as in Figure 4.
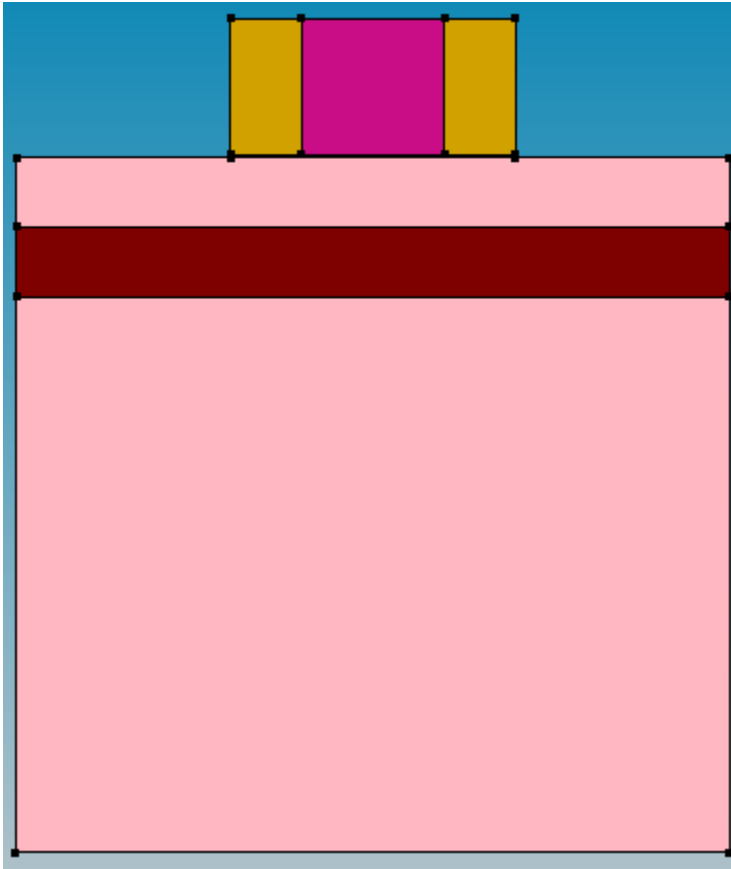

Figure 4. Device structure.

The corresponding Scheme commands are:

```
(sdegeo:create-rectangle
  (position -0.5  0.0   0.0) (position 0.5  1.0   0.0) "Silicon" "region_1")
(sdegeo:create-rectangle
  (position -0.2 -40e-4 0.0) (position 0.2  0.0   0.0) "SiO2" "region_2")
(sdegeo:create-rectangle
  (position -0.2 -0.2   0.0) (position 0.2 -40e-4 0.0) "Si3N4" "region_3")
(sdegeo:create-rectangle
  (position -0.1 -0.2   0.0) (position 0.1 -40e-4 0.0) "PolySilicon" "region_4")
```

> In this example, the later-defined polysilicon gate region overlaps with the previously defined spacer region. Since the **New Replaces Old** option has been set to be the default Boolean, the overlapping is resolved in such a way that the gate replaces the spacer region where the two regions overlap. Likewise, the later-defined buried oxide takes up the overlapped substrate region. Although this given structure can also be defined precisely without any overlap, the process would take more steps as more regions would have to be defined.

2.7 Creating Single-Lumped Regions

In this example, the silicon and nitrate regions are first created as single rectangles. Then, these regions are separated into two disconnected lumps by overlaying them with the buried oxide or gate polysilicon layer, respectively.

When saving the boundary file, each lump is redefined automatically as a separate region. In this case, however, doping and refinement definitions cannot refer to the region names because the final region names are not known before saving. This problem is avoided by explicitly calling the lump redefinition routine and manually renaming each lump.

To create single-lumped regions:

- **Edit** > **Separate Lumps**.

The corresponding Scheme command is:

```
(sde:assign-material-and-region-names "all")
```

The newly assigned names follows the sequence: `oldregionname_lump_1`, `oldregionname_lump_2`, and so on, with `oldregionname` being the name of the old region.

2.8 Rounding Edges

Sentaurus Structure Editor includes functions that can round the corners and edges of a structure.

To round the two outside corners of the nitride spacers:

1. **Edit** > **Parameters**.
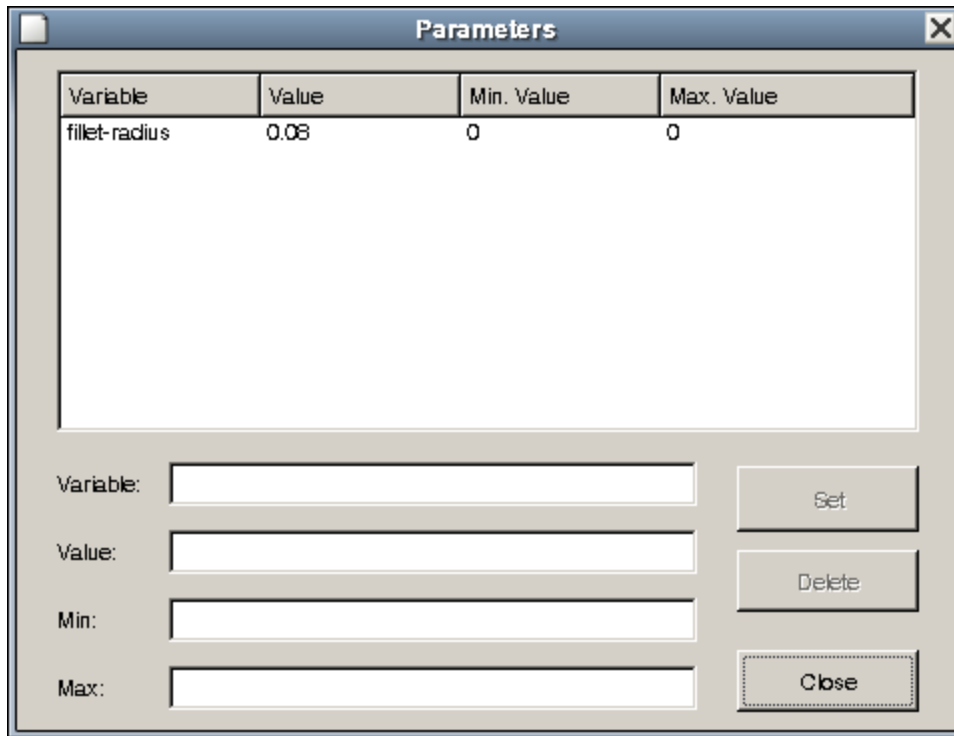   The Parameters dialog box is displayed.

Figure 5. Parameters dialog box.

2. Define `fillet-radius` in the **Variable** field and enter `0.08` for the **Value**.
3. Click **Set**, and then click **Close**.
4. From the Selection Level list, select **Select Vertex**.
5. Click the Select button in the toolbar (▷).
6. Click the upper-left corner of the spacer to highlight the vertex.
7. **Edit** > **2D Edit Tools** > **Fillet**.
   The selected corner is now rounded.
8. Repeat the last two steps with the upper-right corner of the spacer.

The result is shown in Figure 6.

Figure 6. Rounded corners of the spacer with the parameter "fillet-2d".

The corresponding Scheme commands for the rounding operations are:

```
(sde:define-parameter "fillet-radius" 0.08 0.0 0.0 )
(sdegeo:fillet-2d (find-vertex-id (position -0.2 -0.2 0.0)) fillet-radius)
(sdegeo:fillet-2d (find-vertex-id (position  0.2 -0.2 0.0)) fillet-radius)
```

> In the command-line window, the displayed rounding command is (sdegeo:fillet-2d
> (list(car(find-vertex-id (position -0.2 -0.2 0.0)))) 0.08). In the script
> given above, a simplified expression applicable for a single vertex is used. The vertex to be
> rounded is selected using the command (find-vertex-id (position x y z)) instead
> of using mouse operations.

2.9 Defining Contacts

Contacts can be defined to allow the constructed device to be connected to outside power sources.

To define a contact:

1. **Contacts** > **Contact Sets**.
   The Contact Sets dialog box is displayed.
2. Define the properties of the contact. Enter the name of the contact in the **Contact Name** field.
3. Assign RBG colors in the **Edge Color** fields, and modify the **Edge Thickness** value to mark the contact.
   The **Face Pattern** field is only effective for marking 3D contacts.
4. Click **Set** to add the contact to the Defined Contact Sets list. Multiple contacts can be defined in one session.
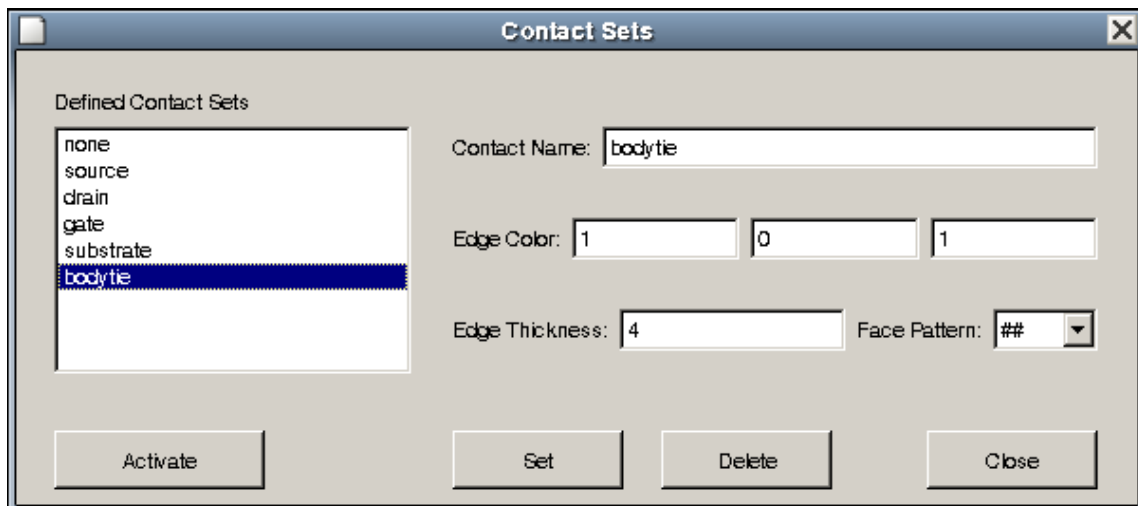5. Click **Close**.

Figure 7. Contact Sets dialog box. (Click image for full-size view.)

The corresponding Scheme commands for adding the five contacts are:

```
(sdegeo:define-contact-set "source"    4.0 (color:rgb 1.0 0.0 0.0 ) "##" )
(sdegeo:define-contact-set "drain"     4.0 (color:rgb 0.0 1.0 0.0 ) "##" )
(sdegeo:define-contact-set "gate"      4.0 (color:rgb 0.0 0.0 1.0 ) "##" )
(sdegeo:define-contact-set "substrate" 4.0 (color:rgb 1.0 1.0 0.0 ) "##" )
(sdegeo:define-contact-set "bodytie"   4.0 (color:rgb 1.0 0.0 1.0 ) "##" )
```

2.10 Setting Contacts at Existing Edges

Contacts become effective only after they are associated with part of the device structure. To associate a contact with a particular edge of a device:

1. **Contacts** > **Contact Sets**.
   The Contact Sets dialog box is displayed.
2. Select the required contact from the Defined Contact Sets list, for example, the source contact.
3. Click **Activate** to activate the selected contact.
4. Alternatively, a contact can be activated from the Contact list.
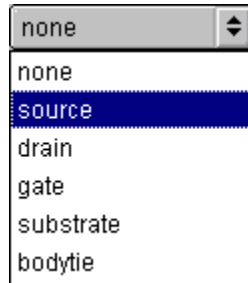
5. In the Selection Level list, select **Edge** to ensure that in the following steps only edges of the device can be selected.
6. Click the Select button (⟡).
7. Click the edge of the structure where the contact, for example, source, will be defined. The selected edge is now highlighted.
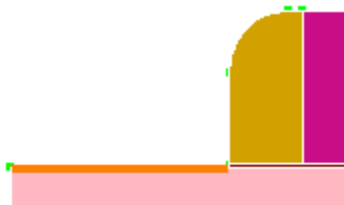

Figure 8. Selecting edges of the source.

8. **Contacts** > **Set Edges**.
   This defines the selected contact, for example, source, at the highlighted edge. The edge is now characterized by the color and line styles previously set for the contact.
9. Repeat the above operations to associate the drain contact with the silicon epilayer edge to the right of the gate stack and the substrate contact with the bottom substrate edge.

The corresponding Scheme commands are:

```
(sdegeo:define-2d-contact (find-edge-id (position -0.4  0.0 0.0)) "source")
(sdegeo:define-2d-contact (find-edge-id (position  0.4  0.0 0.0)) "drain")
(sdegeo:define-2d-contact (find-edge-id (position  0.0  1.0 0.0)) "substrate")
```

The script version allows edges to be selected using explicit coordinates with the (find-edge-id

(position x y z)) command instead of using mouse operations.

2.11 Adding Vertices

For the simulation of 2D SOI MOSFETs, artificial body-tied contacts are sometimes useful. These contacts are placed typically at the interface between the silicon epilayer and the buried oxide near the source side of the gate corner. Before such a contact can be defined, two additional vertices must be introduced.

To add vertices:

1. **Draw** > **Add Vertex**, or click the toolbar button (⬚).
2. Click a location in the view window where the vertex is to be inserted.
   The Exact Coordinates dialog box is displayed.
3. Enter the coordinates (-0.1 0.1) to define the location of the vertex.
4. Repeat the steps to add another vertex at (-0.05 0.1).
5. In the 2D default rendering mode (Gouraud shaded), the newly added vertices are difficult to see.
   To change the rendering mode, **View** > **Render** > **Wireframe**, or click the ▦ toolbar button.
6. Associate the bodytie contact with the edge defined by the two newly added vertices. Follow the procedure described in Section 2.10 Setting Contacts at Existing Edges.

Figure 9 shows the device viewed in the Wireframe mode with all the contacts that have been defined.
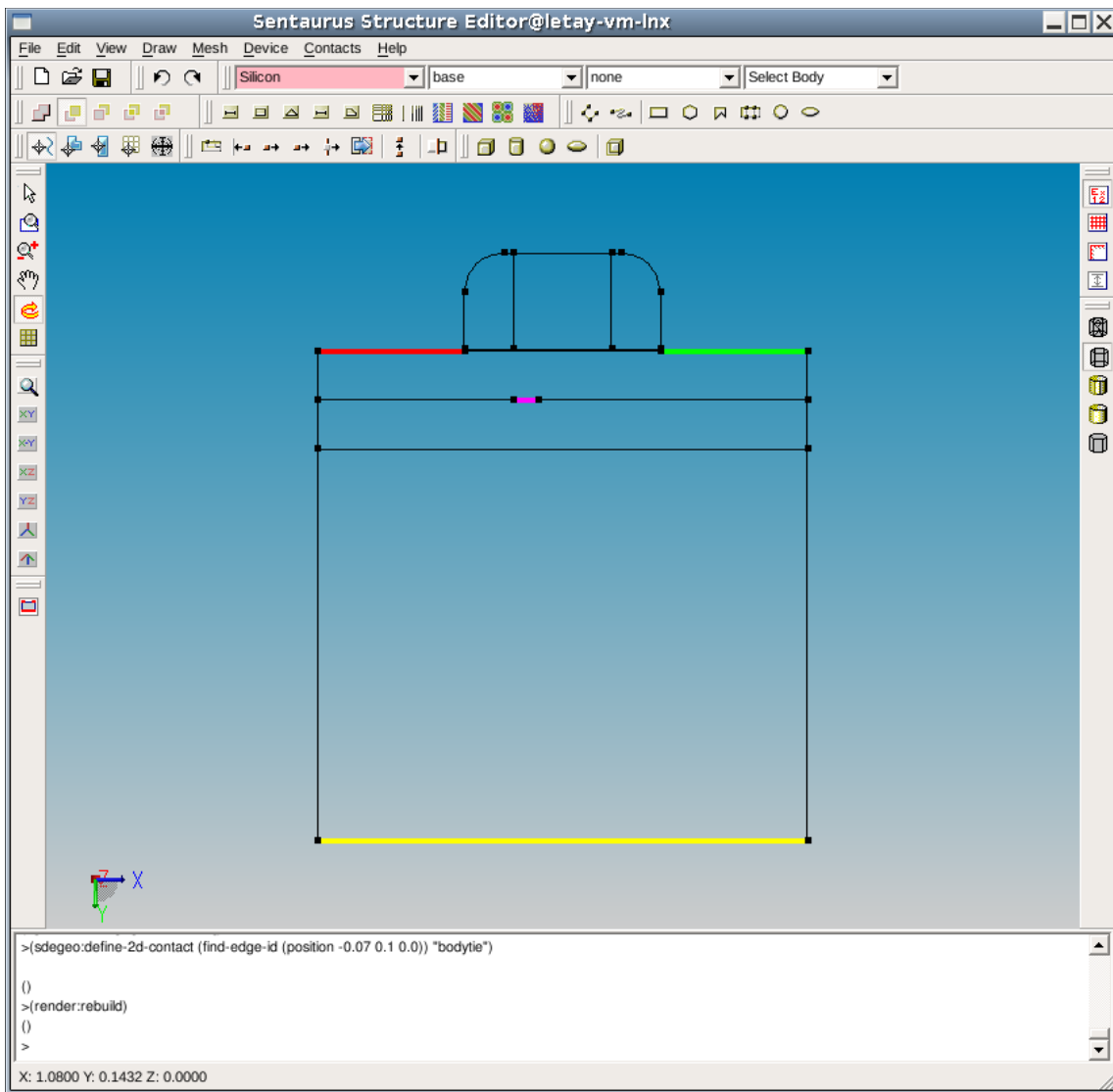
Figure 9. Two-dimensional SOI MOSFET viewed in Wireframe mode with several contacts defined. (Click image for full-size view.)

The corresponding Scheme commands are:

```
(sdegeo:insert-vertex (position -0.1  0.1 0.0))
(sdegeo:insert-vertex (position -0.05 0.1 0.0))
(sdegeo:define-2d-contact (find-edge-id (position -0.07 0.1 0.0)) "bodytie")
```

2.12 Defining a Region as a Contact

In some applications, a contact can cover the entire body of a region (2D) of the device. For example, the metal and polysilicon regions of a MOS device are often treated, as a whole, as ideal contacts.

To define such a contact:

1. Select the contact, for example, `gate`, from the Contact list.
2. From the Selection Level list, select **Select Body**.
3. Click the Select button on the toolbar.
4. Click the region where the contact is to be defined, for example, the poly gate region. The selected region is highlighted.
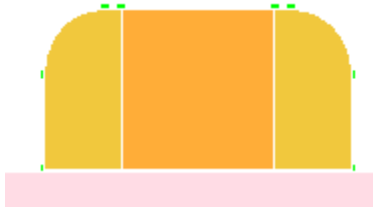

Figure 10. Selecting the polygate region.

5. **Contacts** > **Set Region Boundary Edges**.
   This converts the selected region into a contact. The edges of the region have changed to the color and style of the selected contact.
6. To remove the gate region, select the region and **Edit** > **Remove** > **Region**.

Figure 11. Removing the polygate region.

The corresponding Scheme commands are:

```
(sdegeo:set-current-contact-set "gate")
(sdegeo:set-contact-boundary-edges (find-body-id (position 0.0 -0.1 0.0)))
(sdegeo:delete-region (find-body-id (position 0.0 -0.1 0.0)))
```

> In these Scheme commands, the command (`sdegeo:delete-region-GUI`), used for the mouse-selected regions, is replaced by (`sdegeo:delete-region ...`). This allows regions to be selected by giving explicit coordinates with (`find-region-id (position x y z)`) instead of using mouse operations.

2.13 Renaming Regions

In Sentaurus Structure Editor, if the default Auto Region Naming mode is active, region names such as `region_1`, `region_2 ...` or `region_1_lump_1`, `region_1_lump_2 ...` are assigned automatically to newly added regions or disjoint regions due to overlapping.

You are encouraged to reassign these regions with more descriptive names that are easier to remember and reference.

To rename a region:

1. From the Selection Level list, select **Select Body**.
2. Click the Select button on the toolbar.
3. Click and highlight the region.
4. **Edit** > **Change Region Name**.
   The Change Region Name dialog box is displayed in which the name of the selected region can be modified.
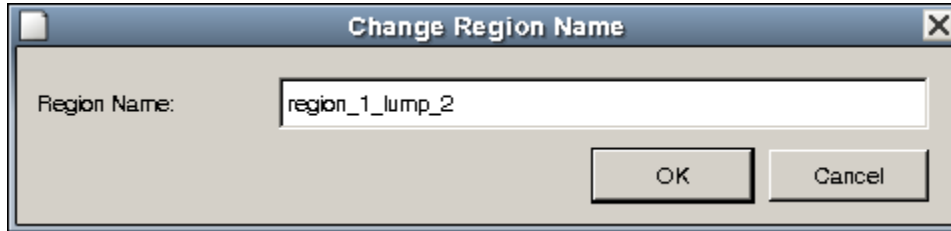

Figure 12. Change Region Name dialog box.

5. Use the following names to rename the corresponding regions that appear in the example device structure: R.Substrate, R.Box, R.Siliconepi, R.Gateox, R.Spacerleft, R.Spacerright.

The corresponding Scheme commands are:

```
(sde:add-material (find-body-id (position  0.0  0.8   0.0))
  "Silicon" "R.Substrate")
(sde:add-material (find-body-id (position  0.0  0.15  0.0))
  "SiO2" "R.Box")
(sde:add-material (find-body-id (position  0.0  0.05  0.0))
  "Silicon" "R.Siliconepi")
(sde:add-material (find-body-id (position  0.0 -20e-4 0.0))
  "SiO2" "R.Gateox")
(sde:add-material (find-body-id (position  0.0  -0.1  0.0))
  "PolySi" "R.Polygate")
(sde:add-material (find-body-id (position -0.15 -0.1  0.0))
  "Si3N4" "R.Spacerleft")
(sde:add-material (find-body-id (position  0.15 -0.1  0.0))
  "Si3N4" "R.Spacerright")
```

To view the results of these renaming operations, type the following command in the command-line window:

```
(sde:showattribs "all")
```

The output of the command is:

```
Entity Id;  Material name;  Region name;  Mask name:
#[entity 9 1]   Silicon R.Siliconepi
#[entity 11 1]  SiO2    R.Gateox
```

```
#[entity 12 1]  Si3N4    R.Spacerleft
#[entity 14 1]  SiO2     R.Box
#[entity 17 1]  Silicon R.Substrate
#[entity 21 1]  Si3N4    R.Spacerright
```

> The polygate has been removed.

2.14 Saving the Model

To save a model:

- **File** > **Save Model**, Ctrl+S, or click the corresponding toolbar button.

If the model has not been saved up to this point, the command displays a standard dialog box in which you can enter the file name for the output. Note that only the root name is required. All the extensions corresponding to different file formats are appended implicitly by Sentaurus Structure Editor.

The corresponding Scheme command is:

```
(sde:save-model "n@node@_geo")
```

This command saves the model geometry in the native ACIS format file n@node@_geo.sat, according to the supplied file name n@node@_geo). Note, that @node@ is replaced by the corresponding node number of Sentaurus Workbench. The Ref/Eval windows and parameters are saved in the Scheme script file n@node@_geo.scm, the mesh-related refinement and doping-related information in the file n@node@_geo_msh.cmd, and the TDR boundary representation in the file n@node@_geo_bnd.tdr.

> In this example, no Ref/Eval windows have been created so far, so only the setting of the variable fillet-radius appears in the file n@node@_geo.scm.

Click to view all the commands discussed in this section in the command file geometry_dvs.cmd.

The complete project can be investigated from within Sentaurus Workbench in the directory Applications_Library/GettingStarted/sde/soifet.

2.15 Assignment

In this assignment, you are encouraged to build a bipolar transistor structure as shown in Figure 13.
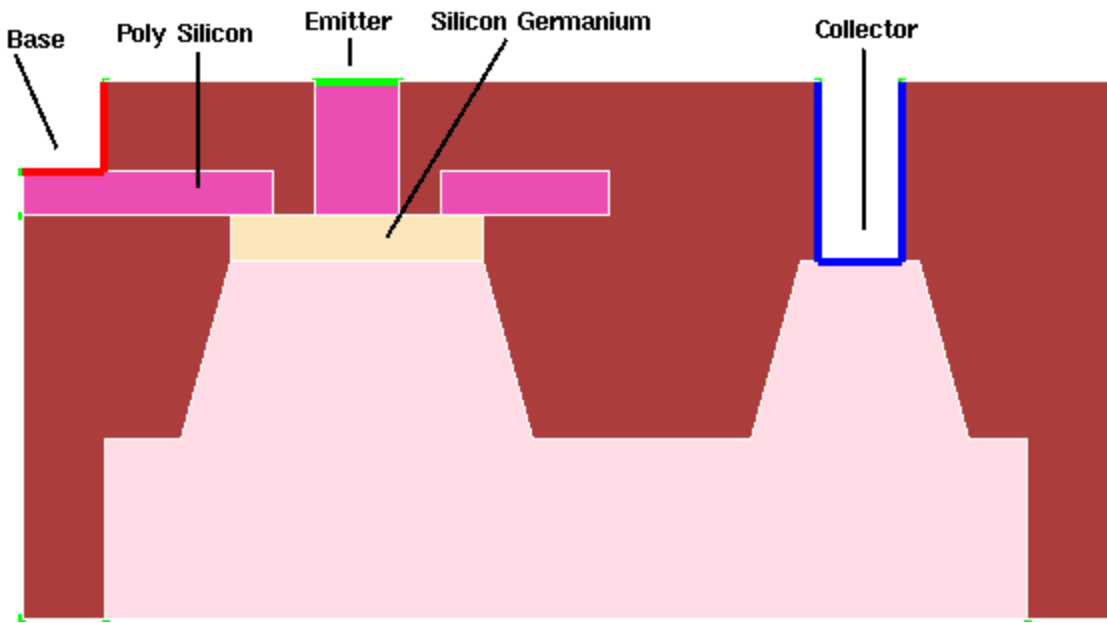
Figure 13. Two-dimensional bipolar transistor structure: SiGe HBT.

The tools discussed so far should allow you to work out most of the structure. However, to complete the assignment, you may need to be familiar with some additional tools that are explained here.

2.15.1 Moving Vertices

To move a vertex:

1. **Draw** > **Move Vertex** or click the toolbar button (➡).
2. In the view window, click the vertex to be moved and drag the vertex to its new location.
   If the Exact Coordinates mode is active, a dialog box is displayed in which you can enter precisely the final location of the vertex.

The corresponding Scheme command is:

```
(sdegeo:move-vertex (car (find-vertex-id (position Xold Yold Zold)))
  (position Xnew Ynew Znew))
```

2.15.2 Creating Polygonal Regions

To create a polygonal region:

1. **Draw** > **Create 2D Regions** > **Polygon**, or click the toolbar button (◪).
2. Click a location in the view window where the first vertex of the polygon is to be placed.
   If the Exact Coordinates mode is active, a dialog box is displayed in which the exact coordinates for the vertex can be entered.
3. Repeat the previous steps for all the other vertices except the last vertex of the polygon.
4. For the final vertex, click the middle mouse button to place it.

The corresponding Scheme commands are:

```
(sdegeo:create-polygon
 (list  (position -0.45 0.6 0.0)
        (position  1.2 0.6 0.0)
        ...)

  "Silicon" "Substate")
```

Click to view the command file sigehbt_dvs.cmd.

The complete project can be investigated from within Sentaurus Workbench in the directory
`Applications_Library/GettingStarted/sde/sigehbt`.

---

main menu | module menu | << previous section | next section >>