

INSTITUTE FOR MICROELECTRONICS

SIMFAB - EXERCISE 2

Mario HITI, 01327428

Submission: May. 23, 2021

Contents

1	Requirements	1
2	Task 1 - Do you want to Build a Snowman?	1
2.1	Advercting a single sphere	1
2.2	Combining multiple spheres	2
3	Task 2 - Simulating the Bosch process	3
3.1	Initial setup	3
3.2	Simulation process	3
3.3	Results	5
3.4	Discussion	5
4	Creating transistor structures	7
4.1	Processing steps	7

1 Requirements

All vtk files and rendered images can be found in the ./out folder. In order to build and run the project the following software packages are required

- GCC compiler
- Python 3.8.5 or higher
- pyvista (for running display.py)
- pdflatex (to build the documentation)

While ParaView can be used to examine the vtk output files, pyvista is used to render images for the documentation by invoking a python script (display.py) PyVista still uses the vtk framework as a backend but offers a simple API for rendering files. This way the process of integrating simulation results into the documentation can be automated.

2 Task 1 - Do you want to Build a Snowman?

2.1 Advecting a single sphere

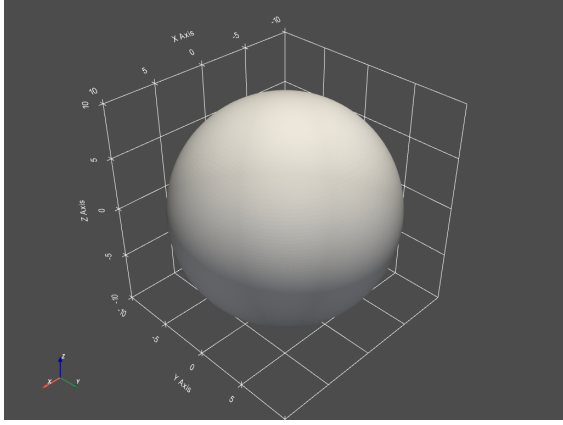
In this task the combination of multiple shapes and advections is explored. We start by creating a single sphere and use a constant velocity field to shrink it.

Algorithm 1: Sphere creation and advection

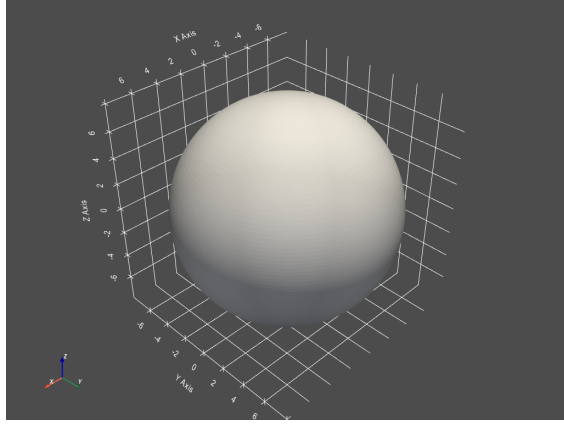
```
1 // create sphere
2 auto sphere1 = lsSmartPointer<lsDomain<NumericType, D>>::New(gridDelta);
3 {
4
5     NumericType origin[3] = {0., 0., 0.};
6     NumericType radius = 10.0;
7     lsMakeGeometry<NumericType, D>(
8         sphere1,
9         lsSmartPointer<lsSphere<NumericType, D>>::New(
10             origin,
11             radius)
12         ).apply();
13 }
14
15 lsAdvect<double, D> advectionKernel;
16 auto constant_vf = lsSmartPointer<ConstantVelocityField>::New(-1);
17
18 advectionKernel.insertNextLevelSet(sphere1);
19 advectionKernel.setVelocityField(constant_vf);
20 advectionKernel.apply();
21
22 double advectionSteps = advectionKernel.getNumberofTimeSteps();
23 std::cout << Number of Advection steps taken: << advectionSteps << endl;
24
25 >> Number of Advection steps taken: 28
```

Fig. 1 shows the sphere with an initial radius = 10 before and after advection. The difference in size can be read from the grid axes. A constant velocity field of -2 is used to shrink the object over the period of 1 unit of time. Therefore each point has to move 2 units of space towards the center, this resulting in a new radius = 6.

The results coincide with the expectations and the advection seems to be stable.



(a) The initial sphere

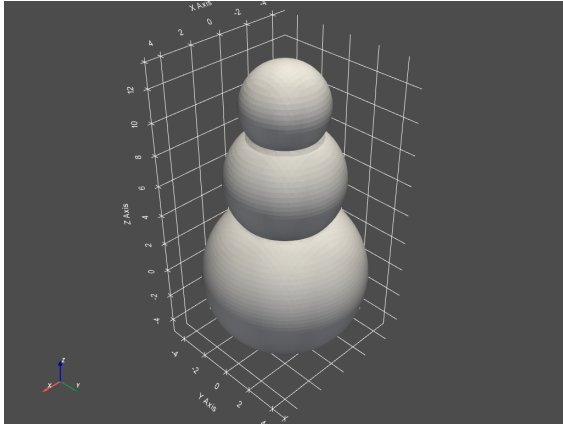


(b) The sphere after advection

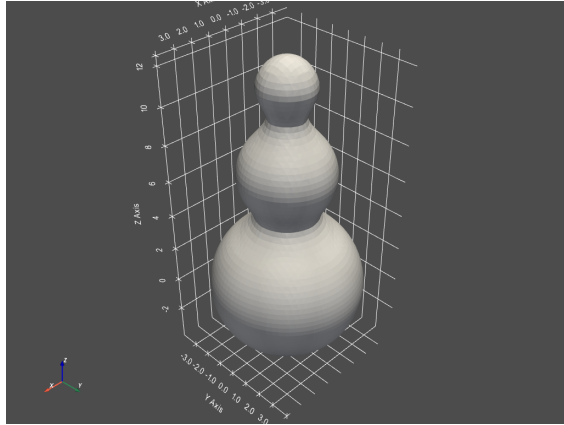
Fig. 1: Shrinking a sphere by constant advection. Note that both images use different grid spacings. A positive velocity would result in an expanding sphere (not depicted).

2.2 Combining multiple spheres

Next multiple spheres are combined to construct a snowman. Then a constant velocity field is applied to simulate uniform melting/etching.



(a) The initial snowman



(b) The snowman after melting

Fig. 2: Melting a snowman consisting of multiple spheres. Sharp edges where the initial spheres touch become rounded by the advection.

3 Task 2 - Simulating the Bosch process

The Bosch- or DRIE-etch process is a method to create deep trenches or shappes with high aspect ratios. Such structures can be used for integrated capacitors or memory devices.

For this task the "drilling" of a circular hole into a substrate is examined.

3.1 Initial setup

The Bosch process requires an initial hole in the mask which has to be created by some process as well. However for this task we assume a perfect cylindrical hole already exists.

We can create the initial layout by stacking a substrate box and a mask box while taking the relative complement with the latter

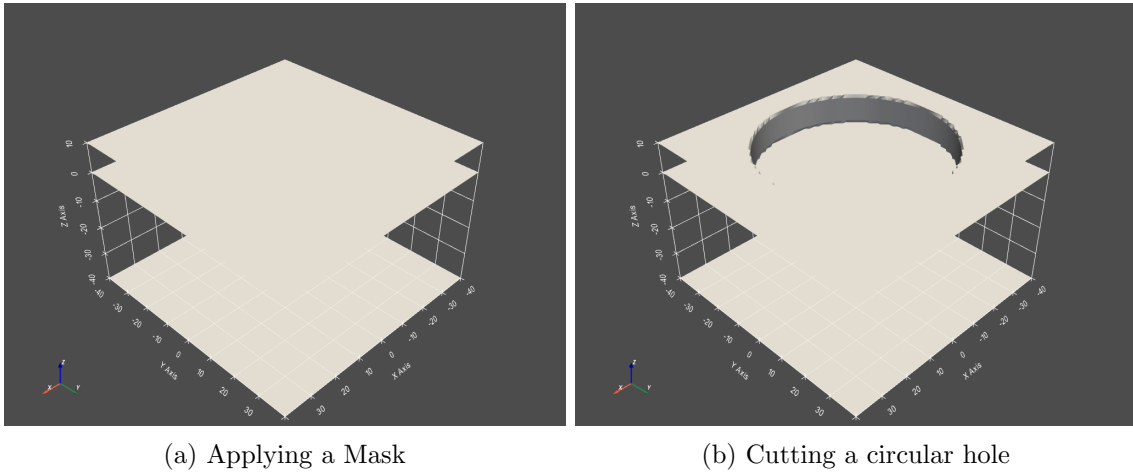


Fig. 3: Creating the initial setup. Mask and substrate appear as planes due to the usage of reflective and infinite boundary conditions

3.2 Simulation process

The process is simulated by performing an initial etch into the substrate followed by 10 passivation-etch cycles. Substrate etching and passivation is performed uniformly. The passivation layer is etched directionally (rate is multiplied by the z-component of the surface normal)

Additionally a randomness factor is introduced to simulate imperfect etching of substrate and and passivation layer. This is performed by adding a random number from a normal distribution with mean = 0 to the etch rate of all materials as shown in Alg. 2

Algorithm 2: etch velocity field

```
1 double getScalarVelocity(const std::array<double, 3> & /*coordinate*/,
2     int material,
3     const std::array<double, 3> &normalVector,
4     unsigned long /*pointId*/) {
5     // if the surface of material 1 is facing upwards, etch it anisotropically
6     double vel;
7     switch(material)
8     {
9         case 0:
10            return 0;    // Mask
11
12        case 1:    // substrate
13            vel = (-1.)*m_velocity_substrate;
14            vel += vel*m_randomNess*distribution(generator);
15            return vel;
16
17        default:    // passivation layer
18            if(normalVector[2] > 0 )
19            {
20                vel = (-1)*m_velocity_passivationLayer*std::abs(normalVector[2]);
21                vel += vel*m_randomNess*distribution(generator);
22                return vel;
23            }
24            else
25            {
26                return 0;
27            }
28        }
29    }
30 }
```

Since the mean value of the normal distribution is placed at $x=0$ both a local increase and decrease in etch rate is possible. This can be used to model effects such as:

- locally increased etch rate caused by particles "breaking off" due mechanical stress such as vibrations
- locally decreased etch rate caused by residue being deposited on surface or gas bubbles
- crystal granularity if the material is no monocrystal
- crystal defects

For the purpose of this simulation the deviation from the mean is modelled using a symmetric gauss distribution.

The simulation is set up such that the following parameters are available:

- etch time: time in seconds the wafer is submerged in acid
- etch velocity passivation layer: rate at which the passivation material is removed.
- etch velocity substrate: rate at which substrate is removed
- deposition time: thickness of the passivation layer
- steps: number of etch-passivation cycles after the initial etch
- randomness: standard deviation of the gauss distribution described above. Random etch rates are calculated relative to the base etch rate. I.e. If the substrate is etched twice as fast the standard deviation is twice as big.

3.3 Results

The following section shows a selection of possible simulation results with uniform and randomized etch rates after 10 process steps. The caption below each image shows the simulation parameters which correspond to the listing above. A quarter of the simulation domain is cut away in order to make the inside of the hole better visible. Axis units are in nano meters.

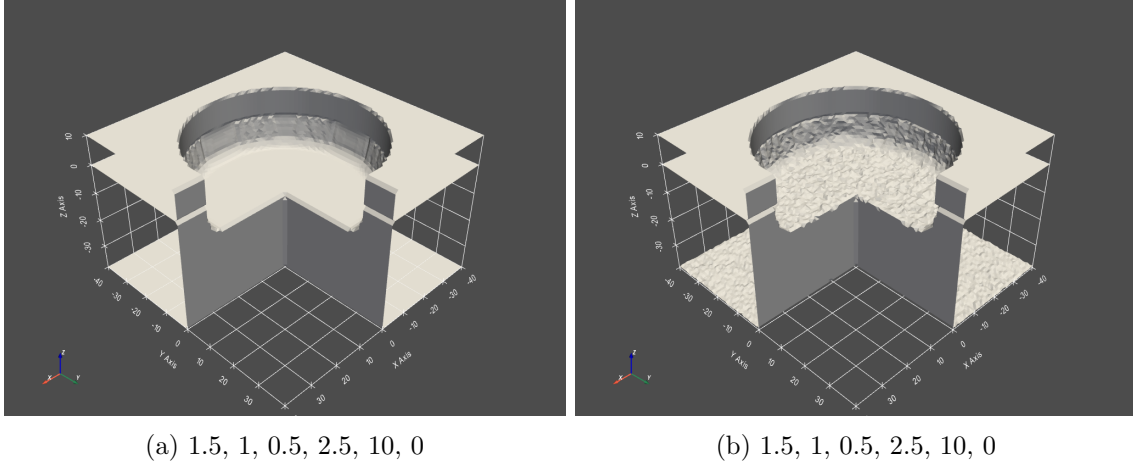


Fig. 4: Short etchrate. Walls are smoother and single scallops are barely visible

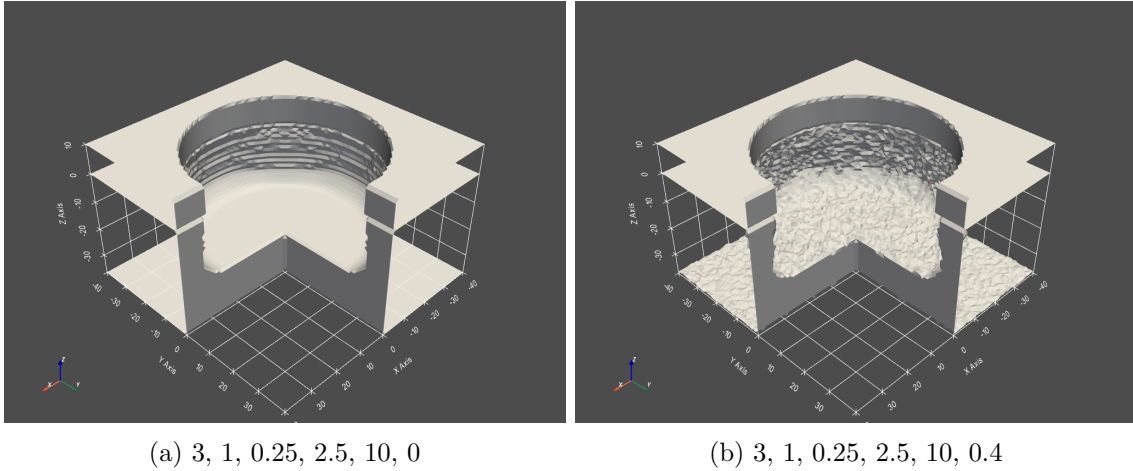
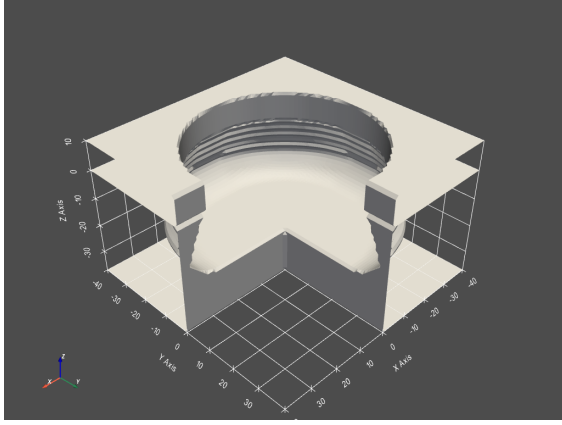


Fig. 5: Longer etchrate. The hole is significantly deeper but grooves are clearly visible. The randomness blurs "smears" out the grooves but can also create random cavities on the wall

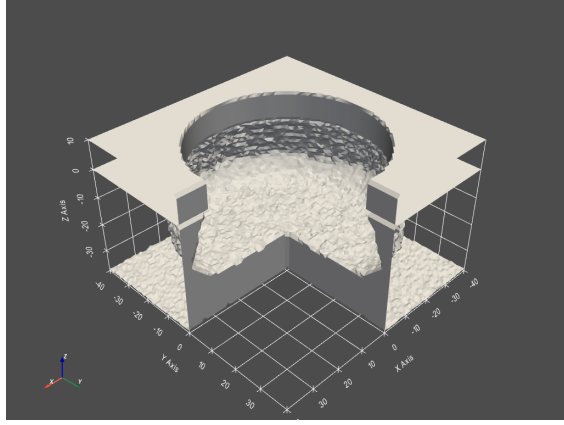
3.4 Discussion

In general it is rather difficult to balance etch- and deposition-rates such that the resulting wall structure is vertical. This becomes even harder with real materials where rates cannot be arbitrarily set but are defined by the properties of the chosen material.

Depth and wall roughness can best be controlled by adjusting the etch time. While longer etch times can remove more material in each process step the texture of the wall becomes increasingly rough. Adding randomness to the simulation may smear the edges out by a bit but can also result in bigger random cavities which may have a negative impact on the structural integrity if the material is exposed to vibrations or surface tensions during a wet process step. A rough surface may also be useful for certain applications where a big surface area is needed.



(a) 2.5, 1, 0.5, 2.5, 10, 0



(b) 2.5, 1, 0.5, 2.5, 10, 0.4

Fig. 6: Deep holes but with unbalanced etch and deposition rates. Walls are clearly skewed

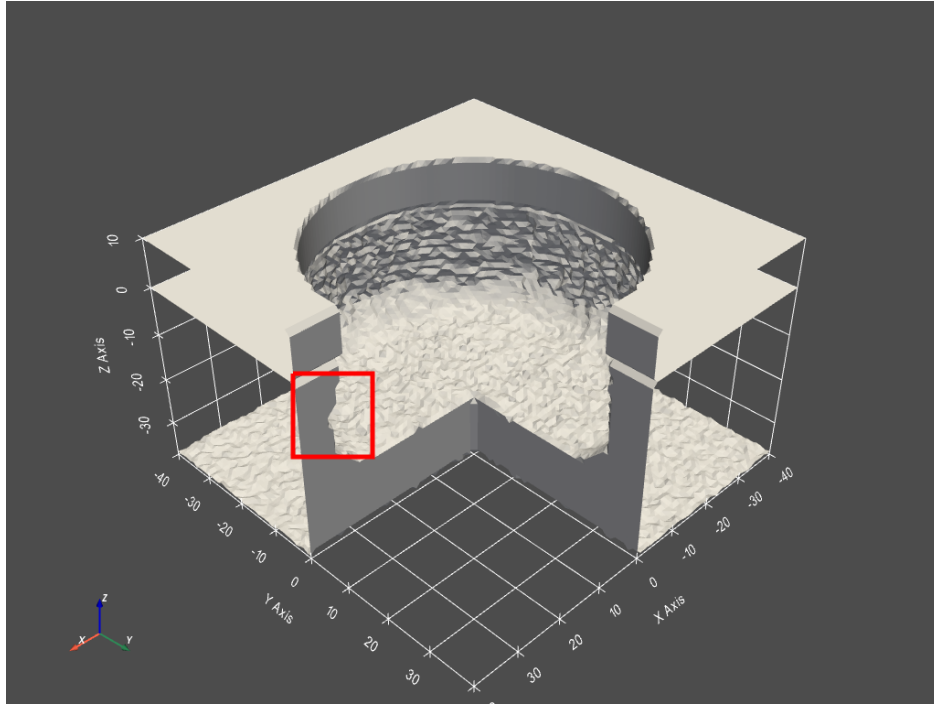


Fig. 7: An example of a bigger random cavity

While this process may allow for the creation of structures with high aspect ratios it is also very expensive in terms of processing steps. In addition to etching and deposition further intermediary steps such as cleaning/flushin and quality control are needed (although maybe not between every step) to achieve good yields. All of these steps are usually performed on different machines.

4 Creating transistor structures

Unfortunately I did not manage to complete this task since it took me much longer than expected to understand the lsVienna framework. Therefore I will present only the steps that I did manage to complete and discuss possible further outcomes.

4.1 Processing steps

All axis units are in nanometers

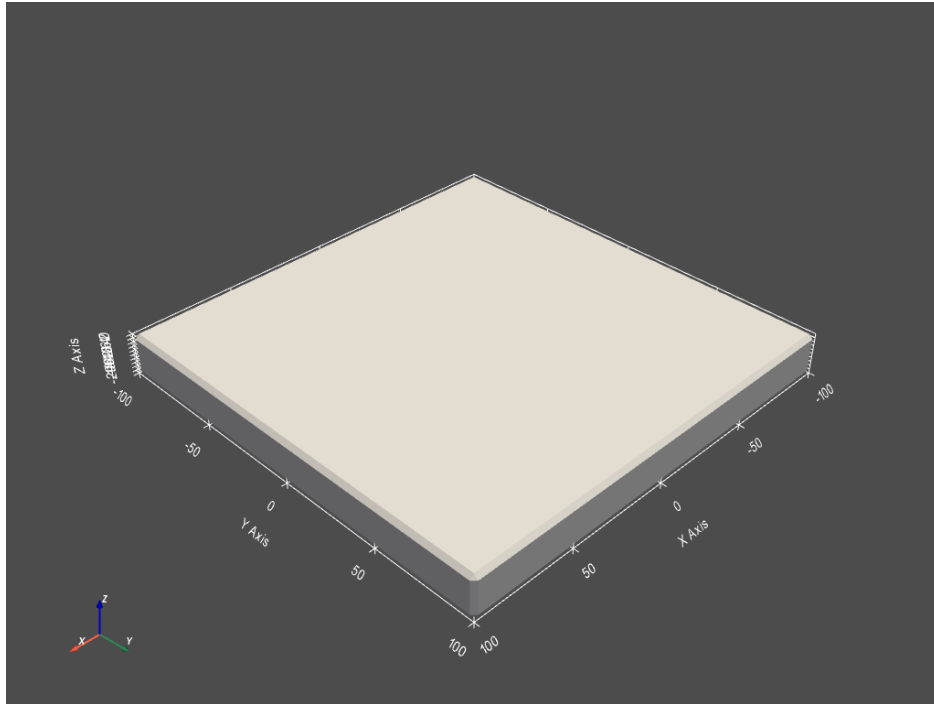


Fig. 8: Step 1: Starting with a single 200x200x20nm plate of silicon

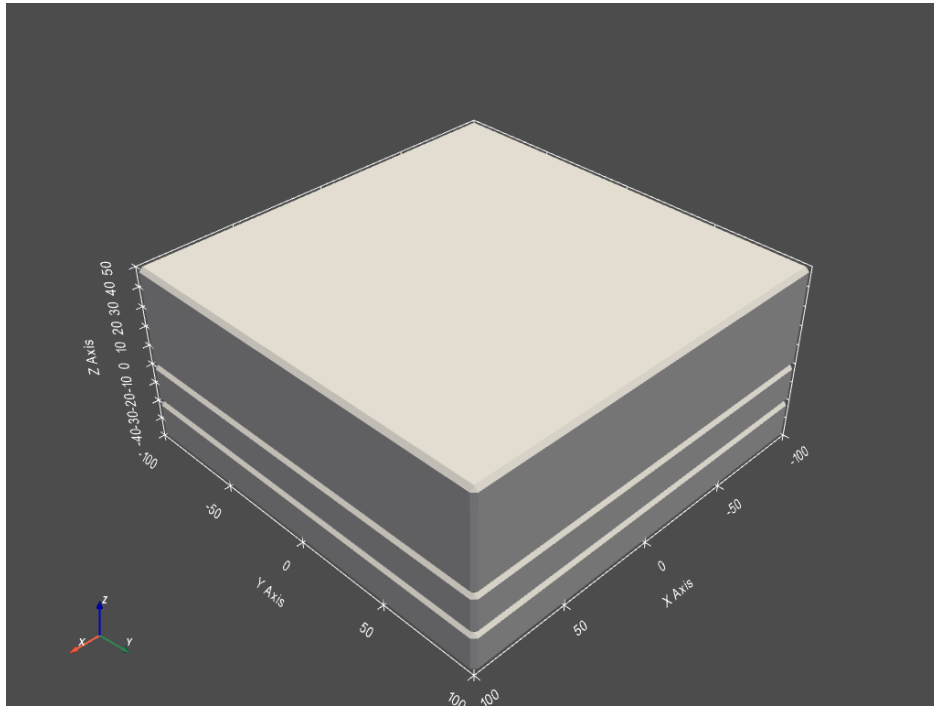


Fig. 9: Step 2: Depositing Oxide (middle) and mask (top) on the silicon substrate

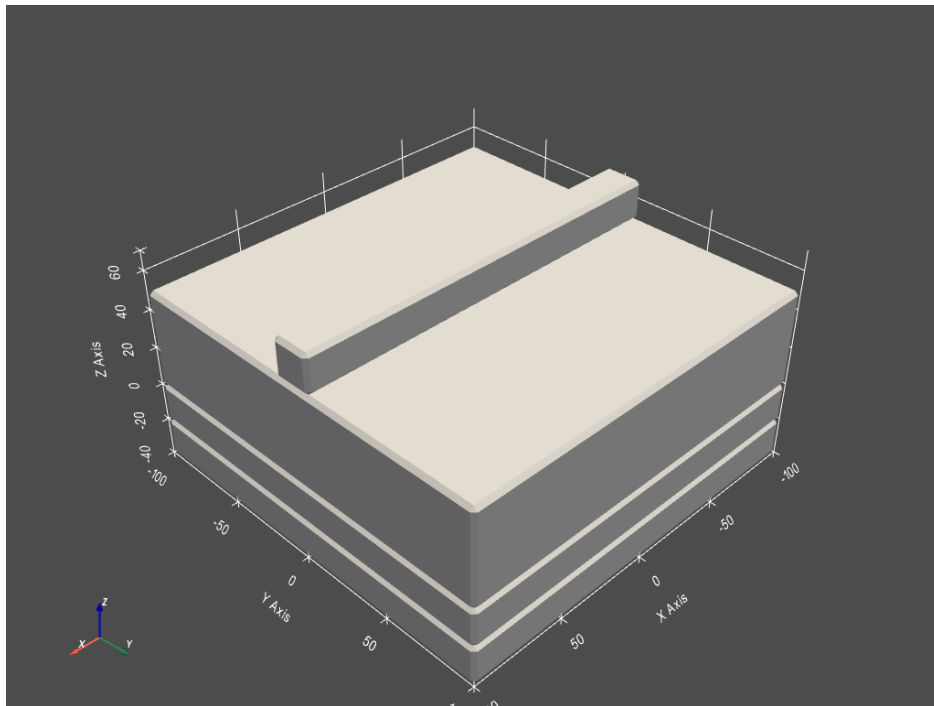


Fig. 10: Step 3: Creating a mask on top of the silicon

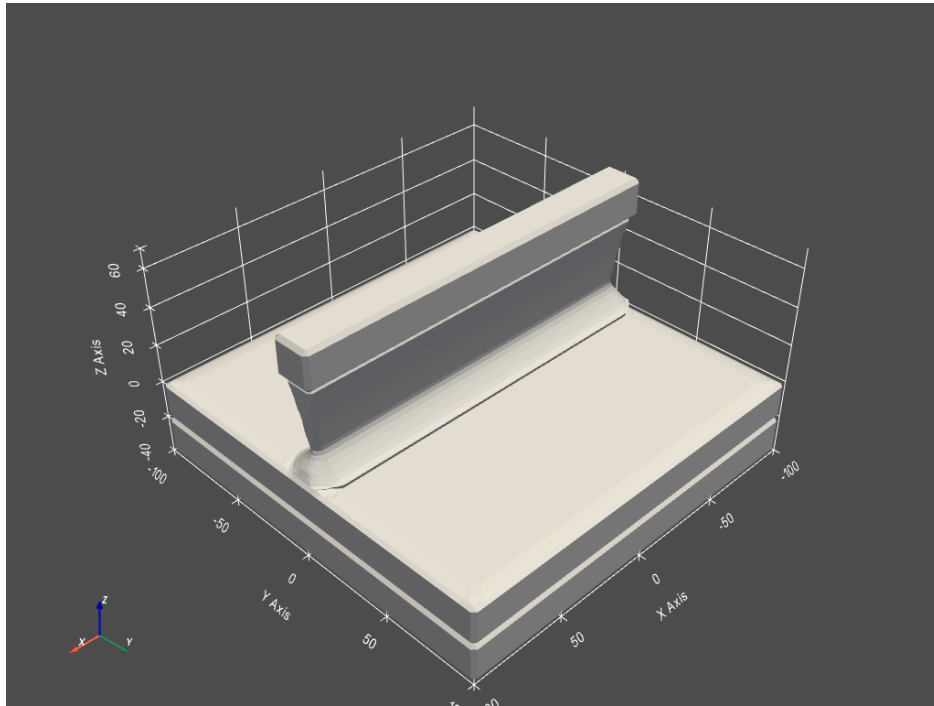


Fig. 11: Step 4: Directionally etching downwards. Etching is not perfectly vertical but is slightly "digging" below the mask.

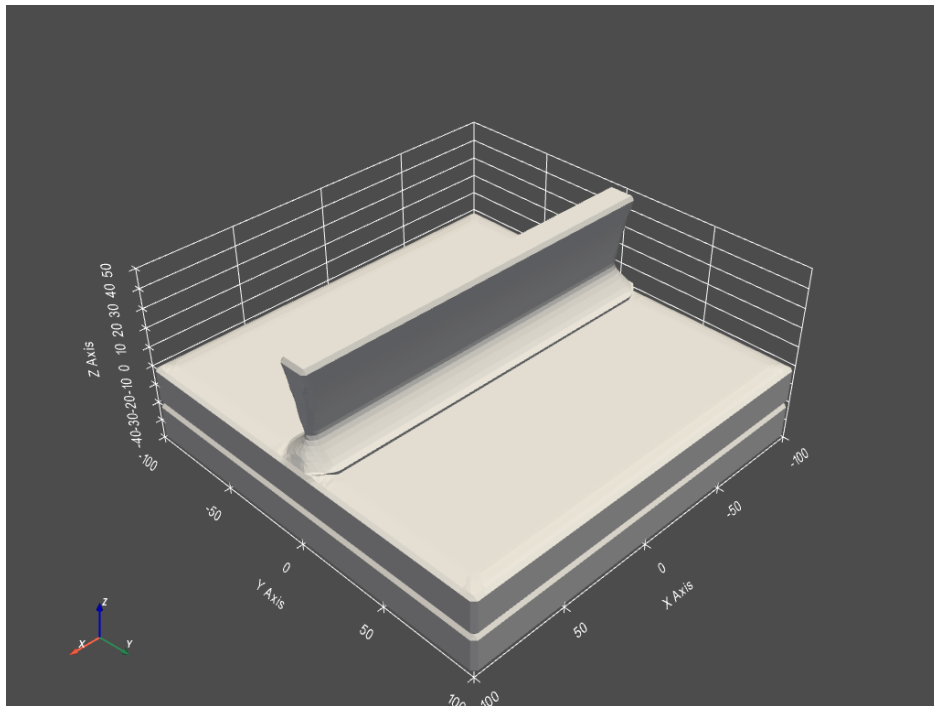


Fig. 12: Step 5: Mask removal

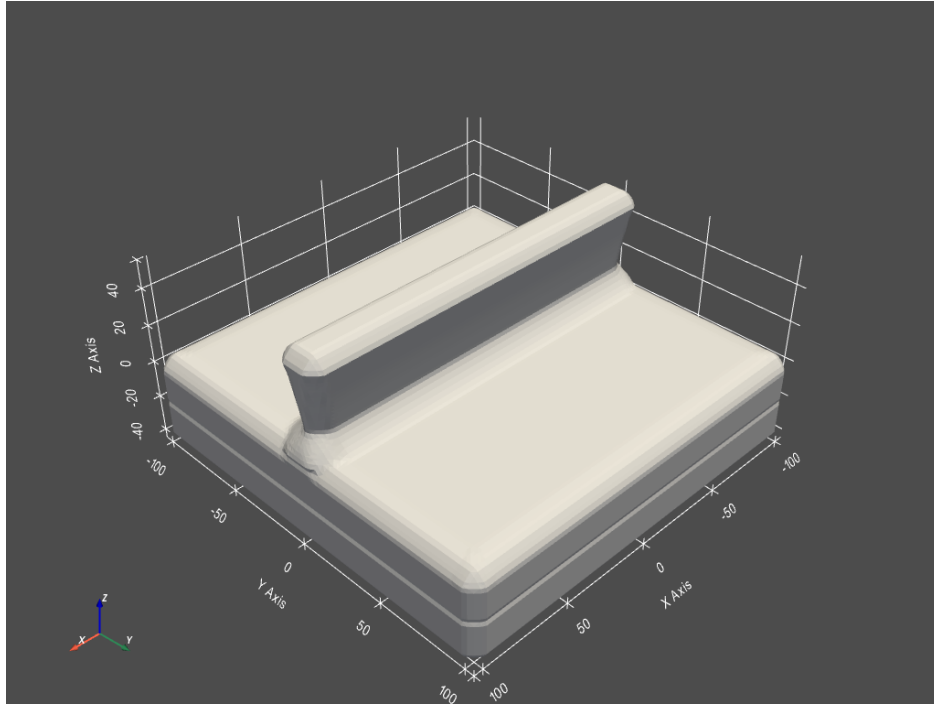


Fig. 13: Step 6: Uniform spacer deposition

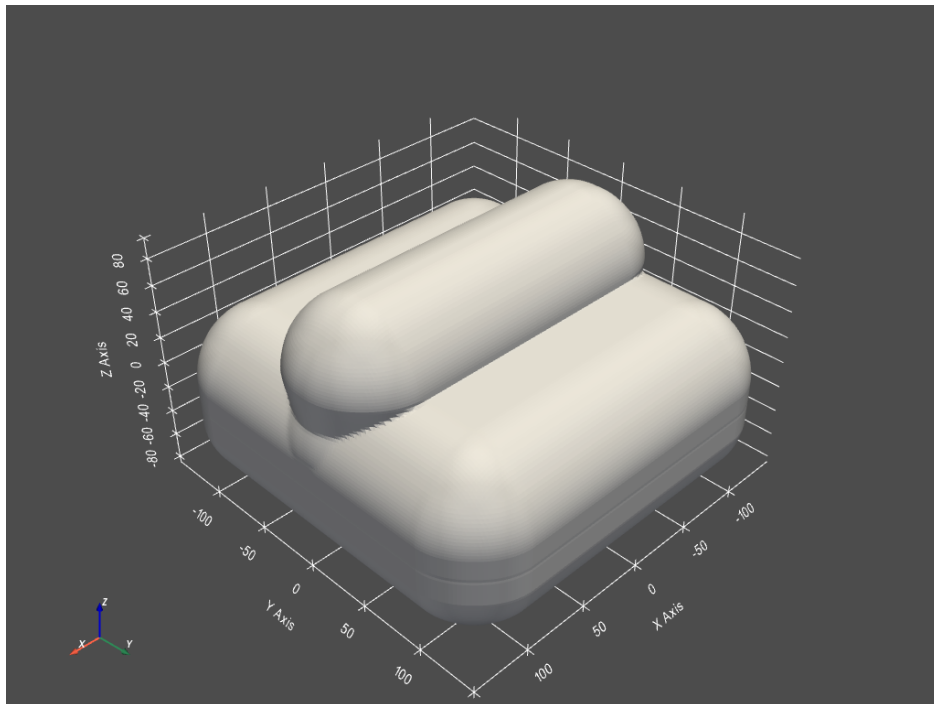


Fig. 14: Step 7: Gate deposition

One can see that something is not working in step 7 (Fig. 14). I suspect that the reason behind this is the absence of properly defined boundary conditions or I am not defining materials correctly. Unfortunately at this step I have run out of time to fix issues.

However if that problem would be fixed creating the gate structure should be analog to creating the fin structure.

As seen in 11 the etching process is not perfectly straight but a slight anisotropy is present. This results in a smaller cross section near the base of the fin. If we would continue to etch then the fin would just break off.

If a positive isotropic component were to be introduced then etched structures would have walls that are skewed outwards instead of inwards (as shown in Fig. 11).