SIMULATION OF SEMICONDUCTOR DEVICES

G. SRINIKETH BALACHANDAR Dr. RITURAJ
Department of Electrical Engineering

Abstract and Introduction

- The primary objective of this project is to enhance the **optical fill factor** of image sensors used in digital cameras. Improving the fill factor increases the fraction of incident light that effectively reaches the photodiodes, thereby enhancing image quality by minimizing optical loss due to reflection and obstruction from metallic interconnects.
- In conventional MOS image sensor technology, the active imaging area consists of a **two-dimensional pixel array**, with each pixel containing a **photodiode** and associated circuitry. However, a significant portion of the pixel area is occupied by **metal interconnects** used for readout and biasing, which reflect or scatter incoming photons. This **metal-induced optical shadowing** reduces the effective light collection, thereby degrading sensor efficiency and image clarity.
- This project aims to address this limitation by:
  - Designing and simulating **larger-area diodes** that can extend across multiple pixels, exploring optimized **device configurations** that maximize light absorption and spatially resolve illumination using **fewer electrical contacts** and developing techniques to infer both the **position** and **intensity** of incident illumination by measuring the resulting photocurrents across these sparse contacts.

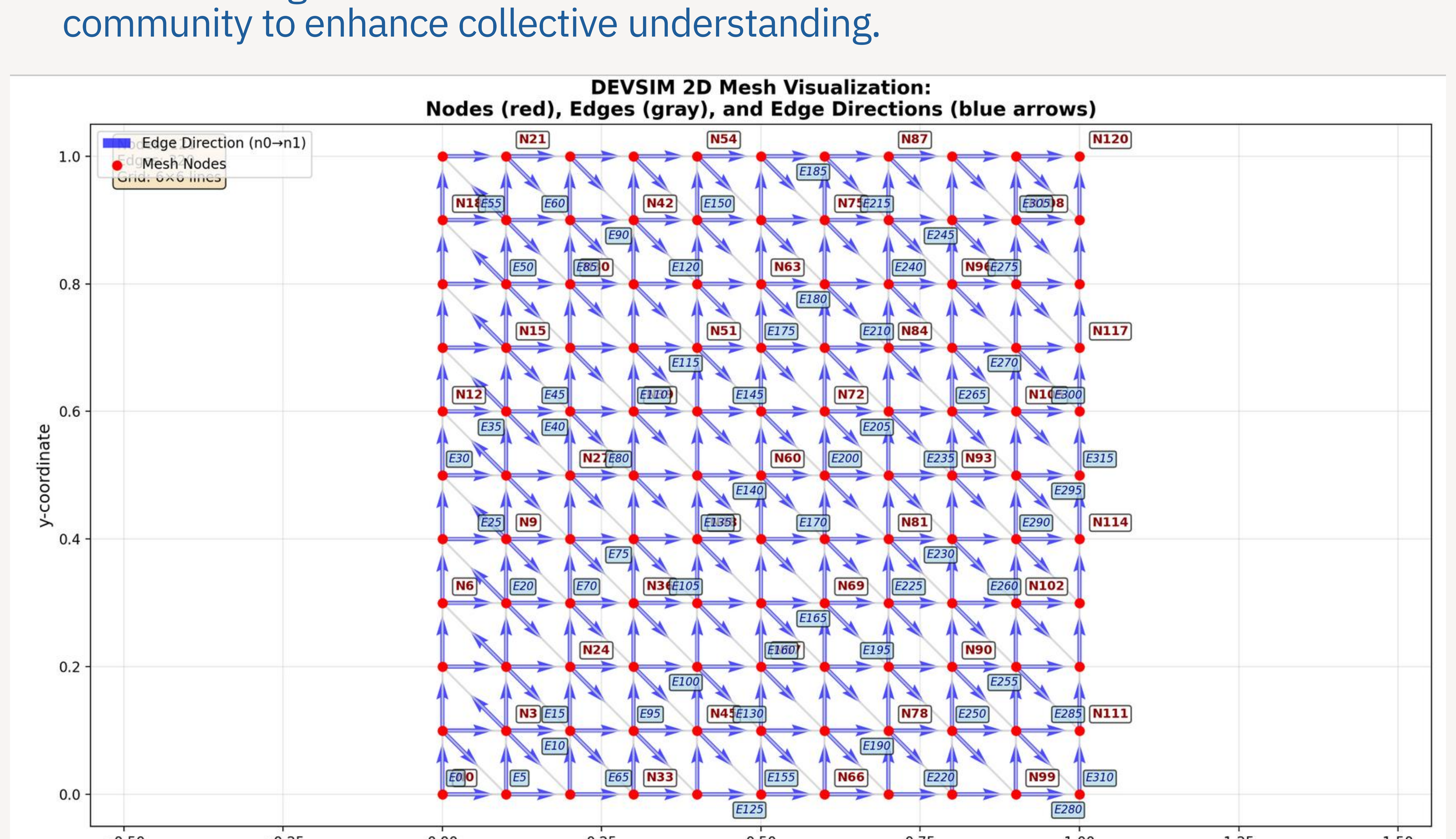
Approach to problem solving

The solutions in general, are obtained using the five basic equations that govern the physics behind the behavior of semiconductor devices, subjected to external bias, illumination, etc..

- The first approach is, discretizing these continuous differential equations and trying to model the current flow similar to the **Transmission Line Model(TLM)**. This approach is expected to result in **2D layout of resistors and nodal analysis using Conductance matrices** would be useful in determining the illumination intensity (which are modelled as current sources into nodes)
- The second approach is to **train a Machine Learning model**, which takes the inputs as current values out of those contacts as input and give us the desired output. This method requires us to simulate various cases, with multiple contacts and different illumination settings and use them to train the ML model.
- For either of the approaches, we have to run simulations of various semiconductor structures, on DEVSIM and choose the one, which is able bring out the effects of illumination quite well.

Software used and issues faced

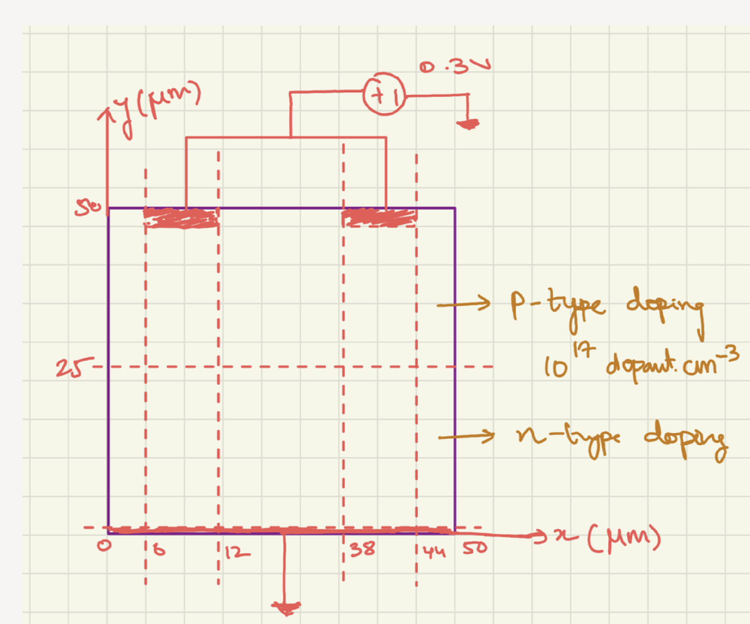
- DEVSIM was used for simulations of various cases. NumPy arrays were used for handling the data from these simulations and numerical calculations. Matplotlib was used to visualize various plots (i.e. Potential, Current (magnitude and direction), Free carriers, concentration as a function of x \&y. etc...).
- DEVSIM lacks clarity on solver techniques and the implementation of node- and edge-based models. To address this, I developed a Python script using Matplotlib to visualize edge directions—from start to end nodes—greatly aiding in understanding current flow in 2D devices and shared this tool with the DEVSIM community to enhance collective understanding.



How does the simulation work

1. Define device geometry
   - Define 2D rectangular mesh over a 5µm × 5µm silicon slab.
   - Place denser vertical (y-axis) and horizontal (x-axis) mesh lines.
   - Add sub-regions for metal contacts (top and bottom).
2. Define doping profile
   - NDn (donor) on the lower half and NAp (acceptor) on the upper half
3. Initialize Poisson's equation
   - Set up only the electrostatic potential solution.
   - Solve Poisson's equation under equilibrium (no carriers yet).
   - Initialize potential with zero bias at both contacts.
4. Define Drift-Diffusion and SRH Recombination Models
   - Set up drift-diffusion model equations for electrons and holes.
   - Define SRH recombination model using carrier lifetimes and conc.
   - Introduce a placeholder node model "generation rates" (initially zero).
   - Solve the potential and the drift-diffusion equations self-consistently.
5. To include the illumination effects, apply gaussian generation profile
   - Use three Gaussian light sources to simulate spot illumination.
   - Normalize the spatial generation function to match total desired generation rate.
   - Update `generation rates` node model and solve the equations.

Simulation results

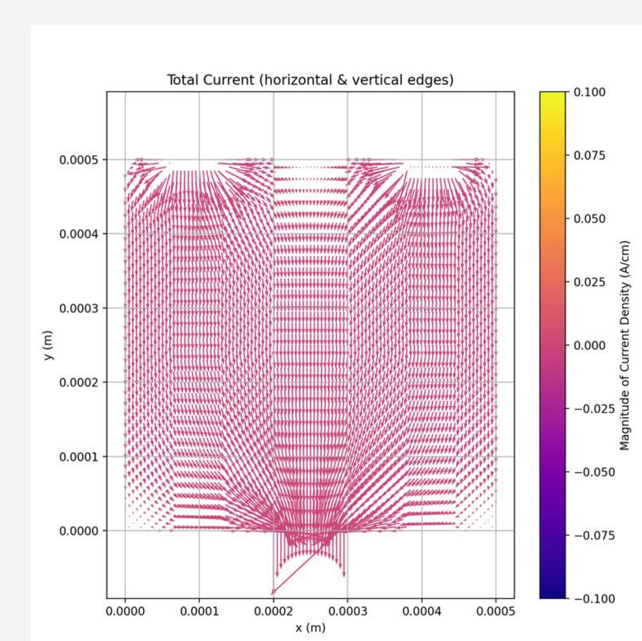


Device Setup

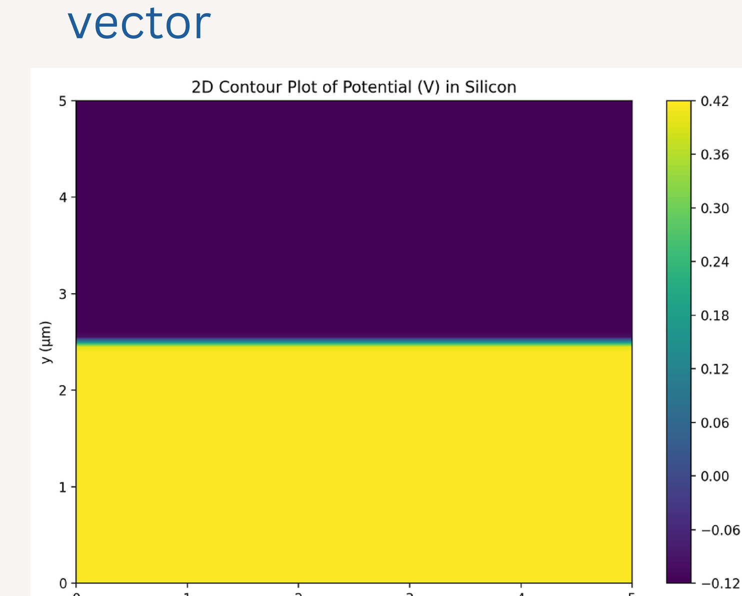
Two contacts on the top, one from x = 0.6 to 1.2 µm and the other from x = 3.8 to 4.4 µm, both at y = 5 µm, with an applied bias of 0.3 V. One contact on the bottom from x = 2 to 3 µm, y = 0 µm, grounded.
Doping: 10^{17} dopants.cm^{-3}

How I managed to get the current density directions ?

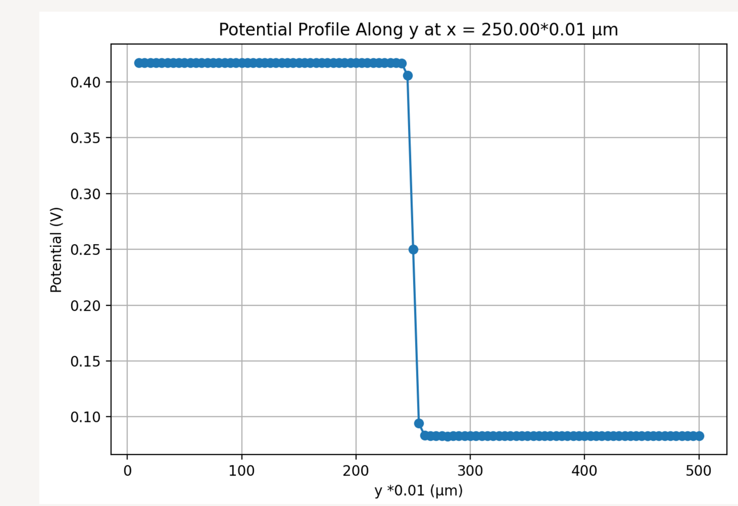
- Noticed that DEVSIM calculated values of magnitude of current density, along all the edges as shown in the figure.
- DEVSIM uses a finite volume method to calculate the current densities. Hence, figured out that the diagonal edges, if considered along with the x & y edges, would be a repetition.
- So, I only considered the horizontal and vertical edges, used them for vector addition and finally plotted the resultant in Matplotlib (Scaled version).



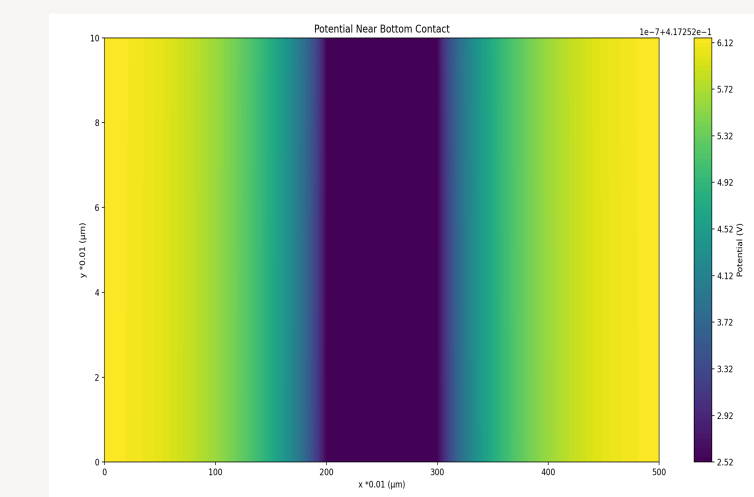
Current density direction vector



Potential as a function of position

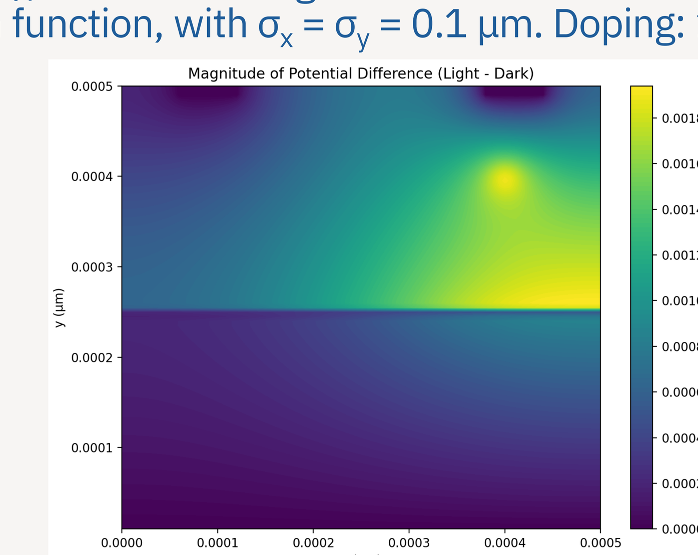


Potential along y-direction at x= 2.5 µm, as a 1D potential curve. In accordance with the theory

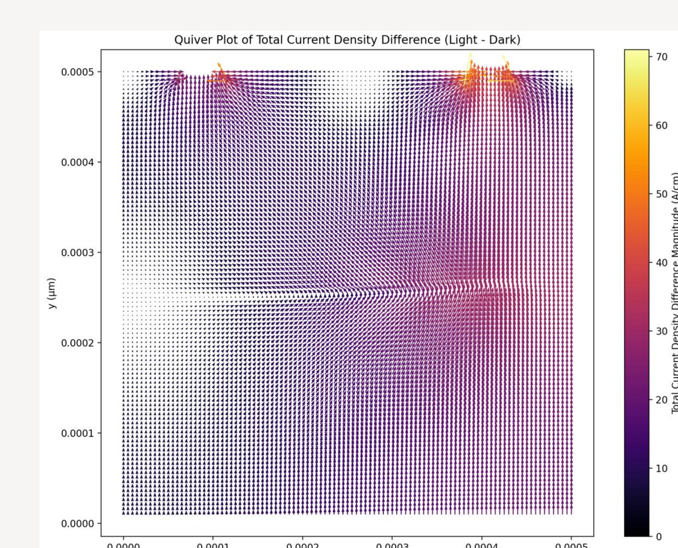


Potential variation near the bottom contact. The variations are very small because most of the drop occurs across the depletion region

Two contacts on the top, one from x = 0.6 to 1.2 µm and the other from x = 3.8 to 4.4 µm, both at y = 5 µm, with an applied bias of 0.3 V. One contact on the bottom from x = 0 to 5 µm, y = 0 µm, grounded. Light illuminations applied at one location, centered at (x0, y0) = (4 µm, 4 µm), modeled as a generation rate of 10^{17} e-h_pairs.cm^{-3} s^{-1} in the form of a 2D Gaussian function, with $\sigma_x = \sigma_y = 0.1$ µm. Doping: 10^{17} dopants.cm^{-3}



Difference in potential b/w the light and dark cases , to highlight the effects of illumination. **It can be noticed that the difference in potential is almost negligible.** Changes seen after three or four decimal places



This plot denotes the Difference in Current density, direction vector.

--- Illuminated Contact Currents ---
Contact bot: 7.126044e-03 A
Contact top1: -2.653920e-03 A
Contact top2: -4.472124e-03 A
Current bot = |Current top1| + |Current top2|

Ratio of current |I_top1| / |I_top2| = 0.5934
This ratio suggests that the illumination in closer to the contact top2 (i.e. right contact). Indeed it is, at (4 µm, 4 µm),

I finally verified that the **Total current flowing through the contacts** can be accurately estimated as the **product of the average y-component of the current density and the horizontal length of the contact**. This approximation is justified by the fact that the applied bias is along the y-direction, making the contribution from the x-component of the current density negligible. As a result, this simplification does not introduce significant error in the calculation.

Progress and References

- Studied the fundamentals of semiconductor physics from **Microelectronic Devices and Circuits by Clifton G. Fonstad**. to gain a clear understanding of the formulation of the five basic semiconductor equations and their application in analyzing PN-junction diodes under external bias, including the effects of illumination.
- DEVSIM user manual for documentation