

Ahmed Sharafeldin 900162792

Ahmed Marzouk 900161641

Digital Design II Fall 2019

Option 1

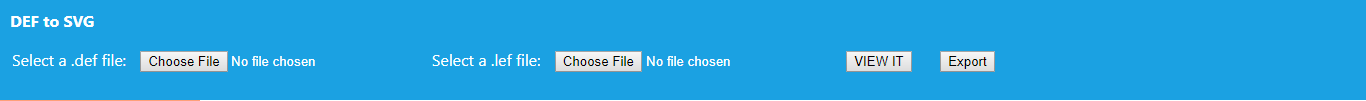
We built this project usingg Javascript. The project is mainly to parts; The parser and the SVG viewer.

The Parser that gets the def file and returns dictionary that includes the pins, nets, and die, and the cells. Inside every cell is information about its name and type and the position relative to the die. The nets include information about the net name and the connection and routes in every layer. The pins include information name, layer, position, dimension and orientation. And also dictionary that contains that contains information about the dimension of every cell type. The Svg viewer is a library that we created and contains four main functions:

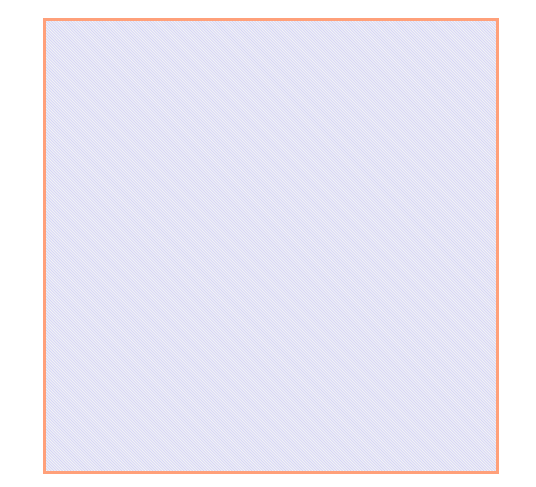
1. CreateLayer() : That creates a new layer and add a checkbox to hide/show that layer and every nets inside
2. CreateNet(Layer, x1, y1, x2, y2): That creates a net (Line) in a given layer(Layer) with first point (x1 and y1) to the second point (x2 and y2)
3. CreateClkNet(Layer, x1, y1, x2, y2) : Does the exact same thing except that it produces a highlighted nets
4. CreateCell ( x, y, height, width, name) : Creates a cell with a given name with an offset (x and y) and dimensions (height and width)
5. CreateFlipFlop CreateCell ( x, y, height, width) : Acts like the CreateCell function except it produces a special type of cell which is a flipflop and also highlights it.
6. CreatePin(x,y, height, width, name) : Creates a pin with a given name and offset and dimensions

**How it works:**

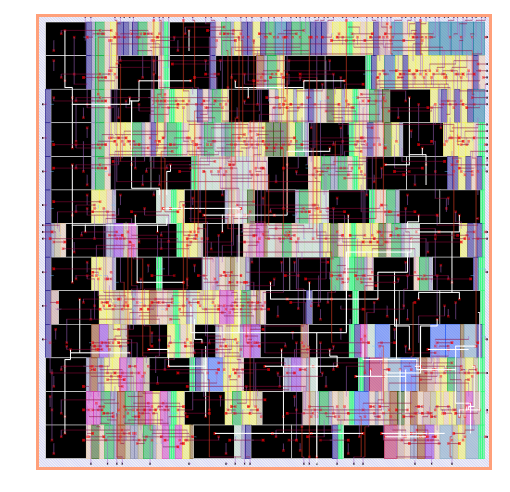
We give the web app a Def and lef file then the parser takes those files and parse them to generate dictionaries that will be given to the svg viewer to represent.



The SVG viewer takes those dictionaries and represent them in a canvas like that:

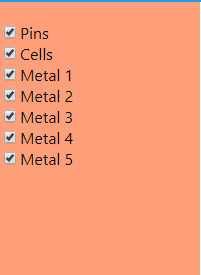


The final product after representation will be something like that:



Where the black rectangles are the flipflops and the white wires are the clock nets. And the other things are represented with some random colors.

The user can toggle any layer in the canvas through a control panel in a side nav looks like this:



The user can also export the SVG file and download it.