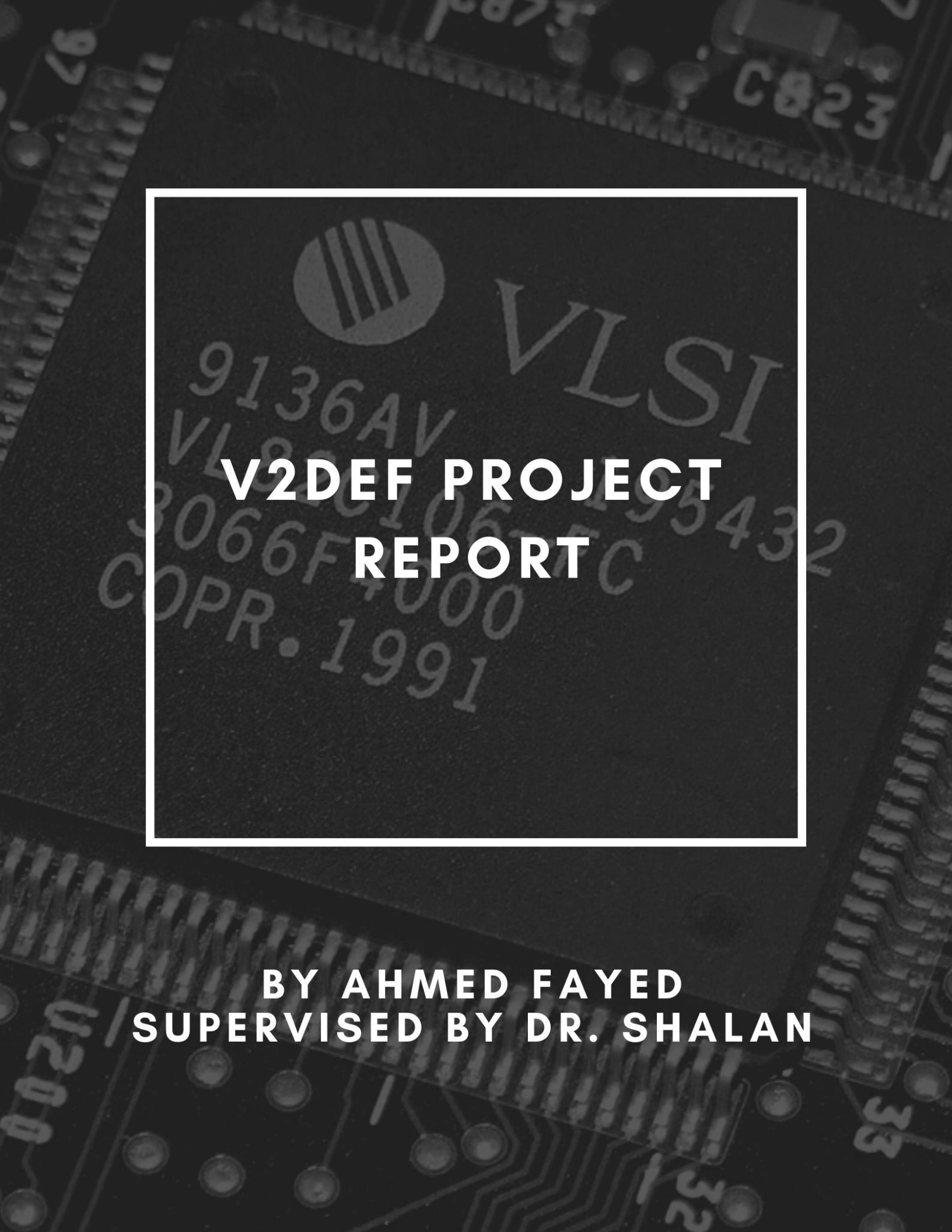


V2DEF PROJECT REPORT

BY AHMED FAYED
SUPERVISED BY DR. SHALAN

Introduction:

In electronic design, a netlist is a description of the connectivity of an electronic circuit. In its simplest form, a netlist consists of a list of the electronic components in a circuit and a list of the nodes they are connected to. A network (net) is a collection of two or more interconnected components.

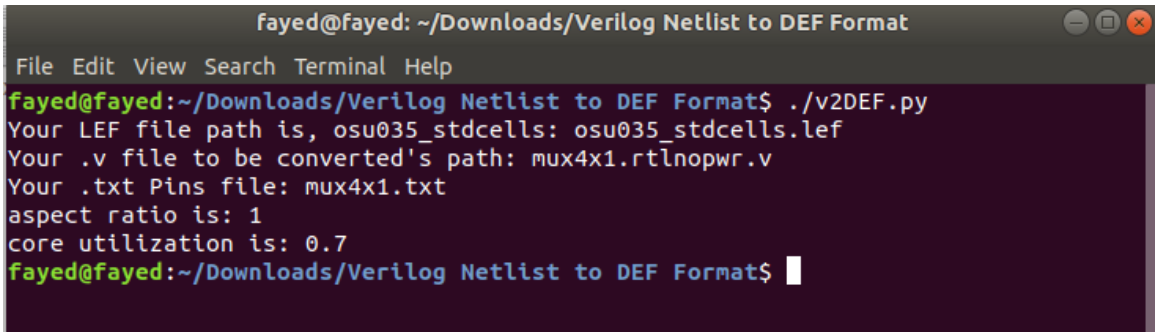
The structure, complexity and representation of netlists can vary considerably, but the fundamental purpose of every netlist is to convey connectivity information. Netlists usually provide nothing more than instances, nodes, and perhaps some attributes of the components involved. If they express much more than this, they are usually considered to be a hardware description language such as Verilog or VHDL, or one of several languages specifically designed for input to simulators.

The LEF file is the abstract view of cells. It only gives the idea about PR boundary, pin position and metal layer information of a cell. To get the complete information about the cell, DEF (Design Exchange Format) file is required. In the technology part layers, design rules, via definitions and metal capacitance are defined. In the site, site extension is defined and in the macros the information about cell description, dimension, layout of pins and blockages and capacitance are defined.

Test Cases:

- **Sample Compiling Test**

  The v2DEF utility can be compiled as shown in the snapshot below:



- **Test 1, Decoder 2 to 4**

  In this test, I used CloudV to run and synthesize the following Verilog code:

decoder_2to4.v

```
1  module decoder_2to4(Y3, Y2, Y1, Y0, A, B, en);
2  output Y3, Y2, Y1, Y0;
3  input  A, B;
4  input  en;
5  reg    Y3, Y2, Y1, Y0;
6
7  always @(A or B or en) begin
8
9  if (en == 1'b1)
10
11 case ( {A,B} )
12 2'b00:   {Y3,Y2,Y1,Y0} = 4'b1110;
13 2'b01:   {Y3,Y2,Y1,Y0} = 4'b1101;
14 2'b10:   {Y3,Y2,Y1,Y0} = 4'b1011;
15 2'b11:   {Y3,Y2,Y1,Y0} = 4'b0111;
16 default: {Y3,Y2,Y1,Y0} = 4'bxxxx;
17 endcase
18
19 if (en == 0)
20 {Y3,Y2,Y1,Y0} = 4'b1111;
21
22 end
23
24  endmodule
```

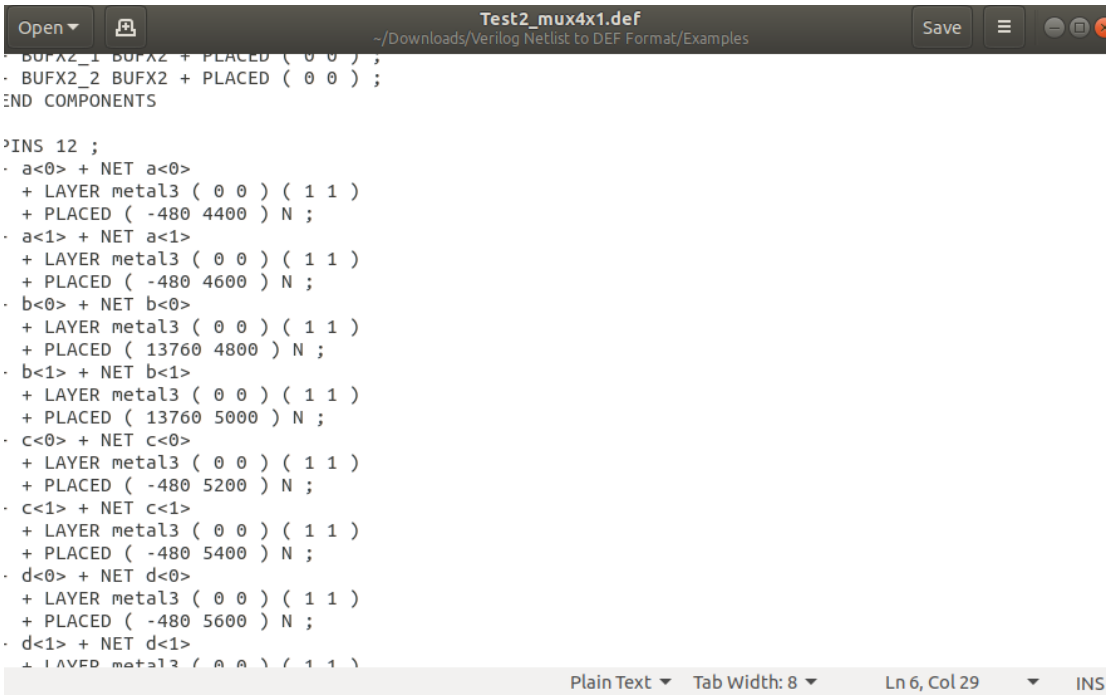
The generated netlist was used in a .v file to be converted into the DEF format, a sample snapshot of the output is shown below:

Open | decoder_2to4.def ~/Downloads/Verilog Netlist to DEF Format | Save

```
VERSION 5.6 ;
NAMESCASESENSITIVE ON ;
DIVIDERCHAR "/" ;
BUSBITCHARS "<>" ;
DESIGN decoder_2to4 ;
UNITS DISTANCE MICRONS 100 ;

DIEAREA ( -480 -400 ) ( 8160 8200 ) ;

TRACKS Y -400 DO 44 STEP 200 LAYER metal1 ;
TRACKS X -480 DO 55 STEP 160 LAYER metal2 ;
TRACKS Y -400 DO 44 STEP 200 LAYER metal3 ;
TRACKS X -480 DO 28 STEP 320 LAYER metal4 ;

COMPONENTS 6 ;
- INVX8_2 INVX8 + PLACED ( 0 0 ) ;
- NAND3X1_3 NAND3X1 + PLACED ( 0 0 ) ;
- INVX4_4 INVX4 + PLACED ( 0 0 ) ;
- NAND3X1_5 NAND3X1 + PLACED ( 0 0 ) ;
- NAND3X1_6 NAND3X1 + PLACED ( 0 0 ) ;
- NAND3X1_7 NAND3X1 + PLACED ( 0 0 ) ;
END COMPONENTS

PINS 7 ;
- A + NET A
  + LAYER metal3 ( 0 0 ) ( 1 1 )
  + PLACED ( -480 2400 ) N ;
- B + NET B
  + LAYER metal3 ( 0 0 ) ( 1 1 )
  + PLACED ( 8160 2600 ) N ;
```

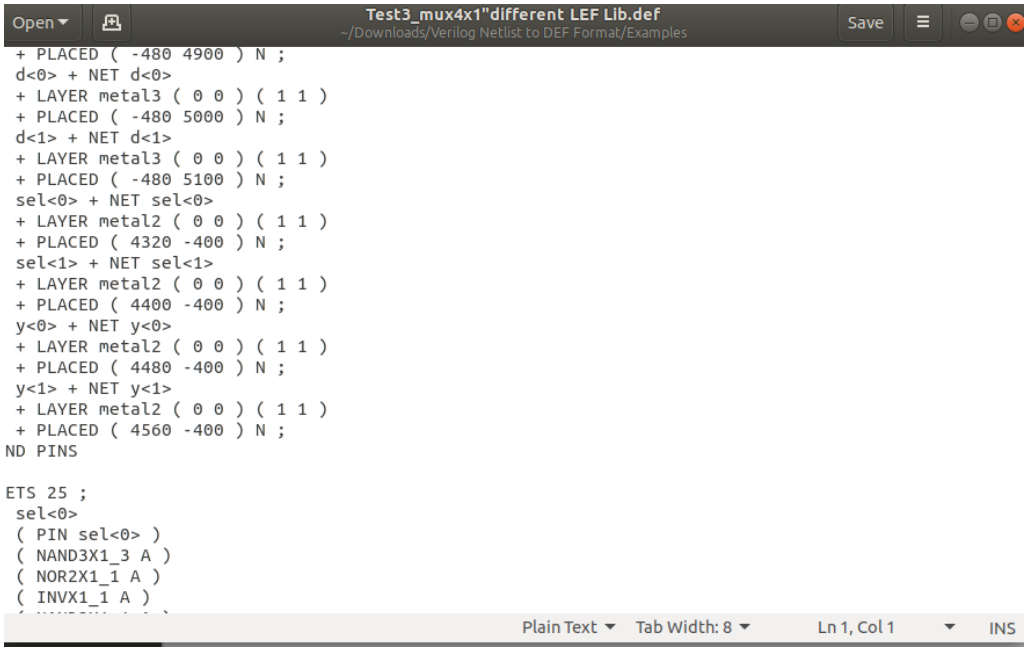
- **Test2, MUX4x1:**

  In this test, the .v netlist file was given in the DDIIF19 Slack channel. It was used to generate the .DEF file and a sample snapshot is below:



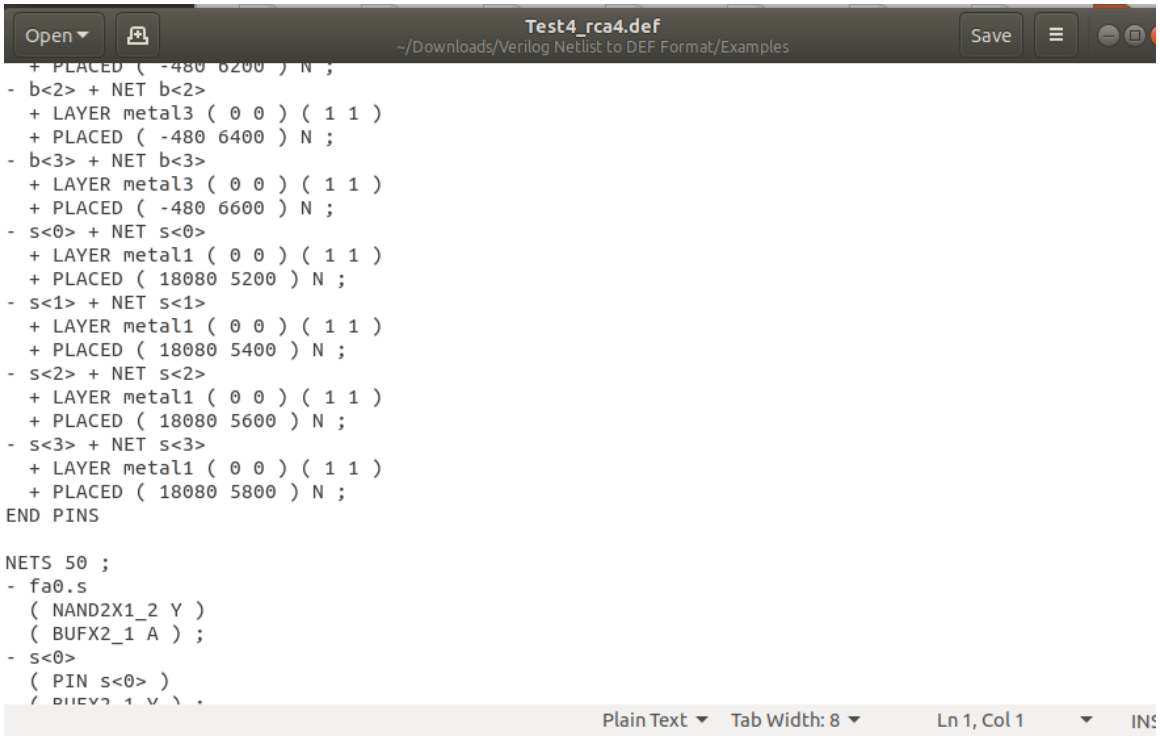
- **Test3, MUX4x1, different LEF:**

  This test also takes the MUX4x1 netlist file given but compiles it with a different .lef file “osu018_stdcells.lef”. A sample output is as below:



- **Test4, rca4:**

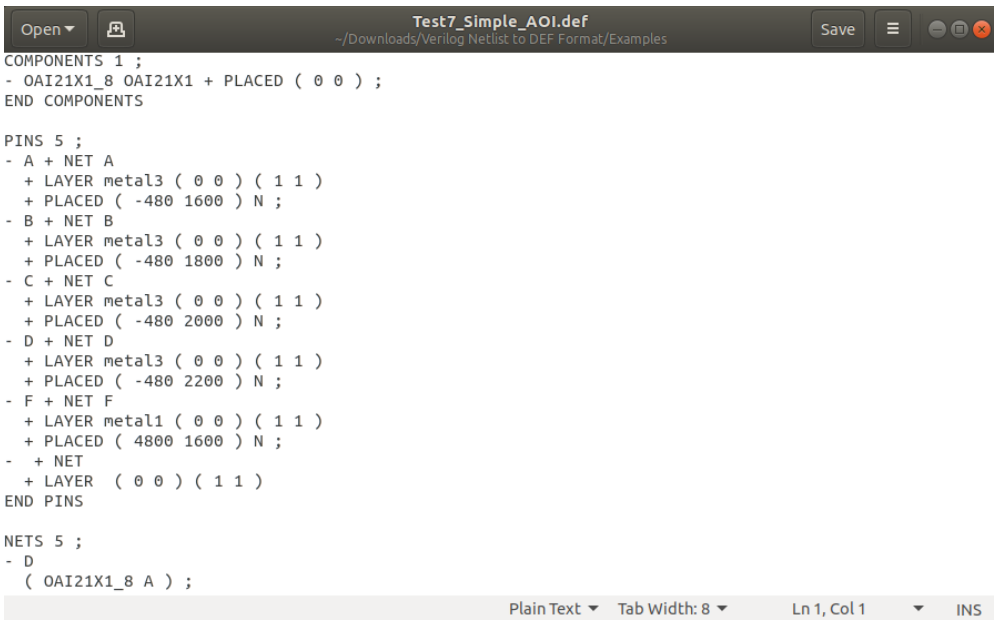
  In this test, the .v netlist file was given in the DDIIF19 Slack channel. It was used to generate the .DEF file and a sample snapshot is below:



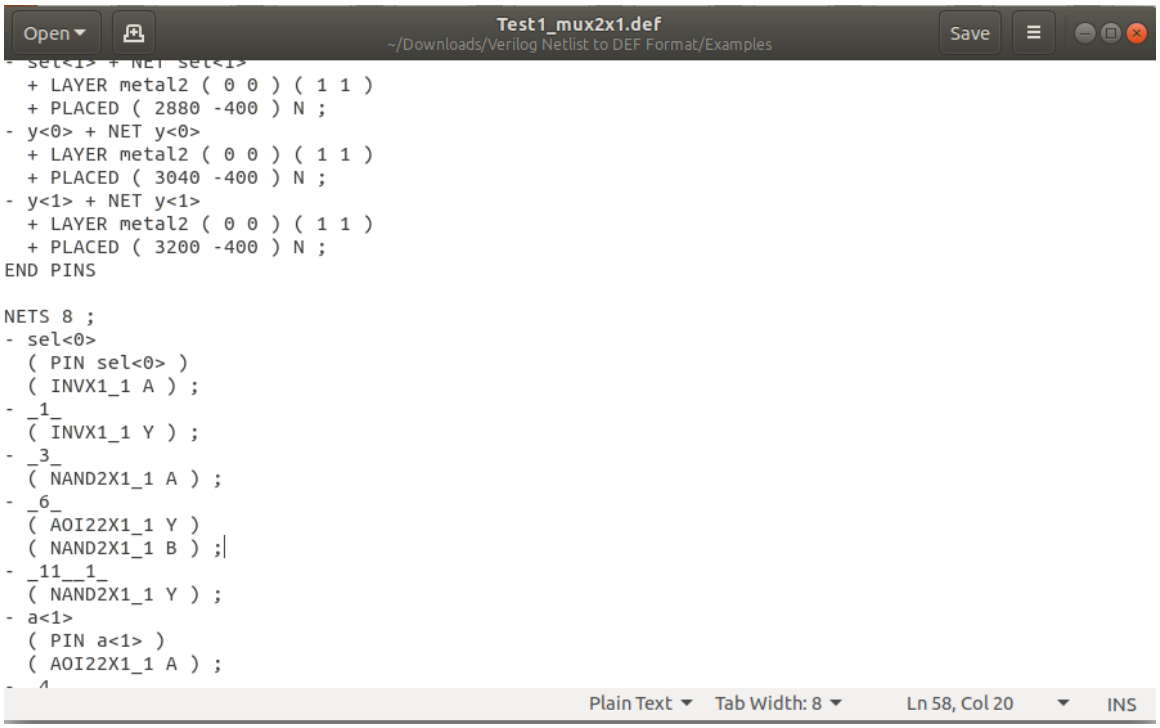
- **Test5, simple AOI:**

  In this test, a simple AndOrInverter code was used to generate a netlist through CloudV:

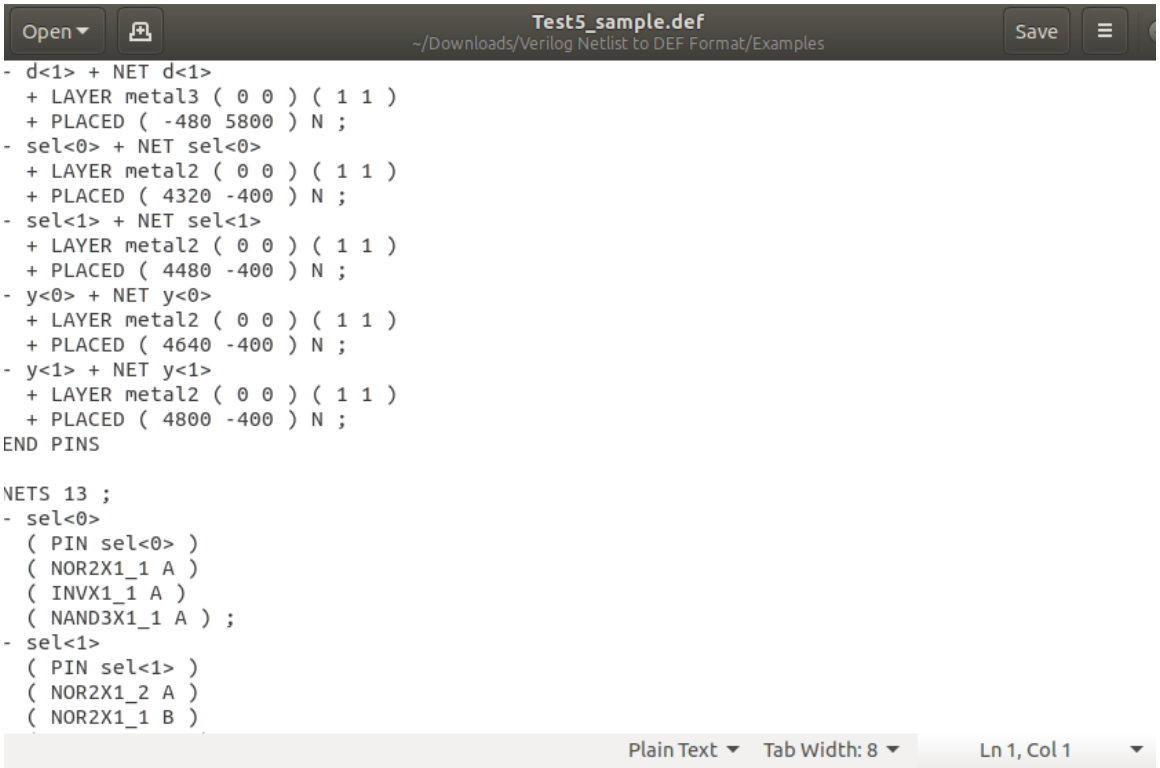
```
module AOI (input A, B, C, D, output F);

  assign F = ~((A & B) | (C & D));

endmodule
```



- **Test6, MUX2x1:**



- **Test7, random sample test:**



- **Test8, random sample2:**