# 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:

```
fayed@fayed: ~/Downloads/Verilog Netlist to DEF Format

File Edit View Search Terminal Help

fayed@fayed:~/Downloads/Verilog Netlist to DEF Format$ ./v2DEF.py

Your LEF file path is, osu035_stdcells: osu035_stdcells.lef

Your .v file to be converted's path: mux4x1.rtlnopwr.v

Your .txt Pins file: mux4x1.txt

aspect ratio is: 1

core utilization is: 0.7

fayed@fayed:~/Downloads/Verilog Netlist to DEF Format$
```

#### Test 1, Decoder 2 to 4

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

```
📄 decoder_2to4.v
                        ×
     module decoder_2to4(Y3, Y2, Y1, Y0, A, B, en);
2
     output Y3, Y2, Y1, Y0;
3
     input A, B;
4
     input en;
5
            Y3, Y2, Y1, Y0;
     reg
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;
              \{Y3,Y2,Y1,Y0\} = 4'b1011;
14
     2'b10:
              \{Y3,Y2,Y1,Y0\} = 4'b0111;
15
     2'b11:
     default: {Y3,Y2,Y1,Y0} = 4'bxxxx;
16
17
     endcase
18
19
    if (en == 0)
20
    \{Y3,Y2,Y1,Y0\} = 4'b1111;
21
22
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:

```
decoder_2to4.def
 Open ▼ 🖭
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 ;
```

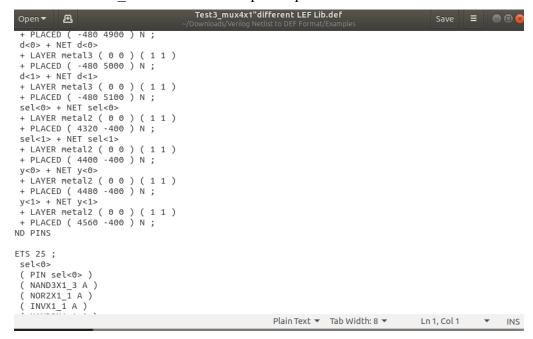
### • Test2, MUX4x1:

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:

```
Test2 mux4x1.def
          Æ
                                                                                             - BUFX2_1 BUFX2 + PLACED ( 0 0 );
- BUFX2_2 BUFX2 + PLACED ( 0 0 );
END COMPONENTS
PINS 12 ;
· a<0> + NET a<0>
 + LAYER metal3 ( 0 0 ) ( 1 1 )
 + PLACED ( -480 4400 ) N ;
· a<1> + NET a<1>
 + LAYER metal3 ( 0 0 ) ( 1 1 )
 + PLACED ( -480 4600 ) N ;
· b<0> + NET b<0>
 + LAYER metal3 ( 0 0 ) ( 1 1 )
 + PLACED ( 13760 4800 ) N ;
· b<1> + NET b<1>
 + LAYER metal3 ( 0 0 ) ( 1 1 )
 + PLACED ( 13760 5000 ) N ;
· c<0> + NET c<0>
 + LAYER metal3 ( 0 0 ) ( 1 1 )
 + PLACED ( -480 5200 ) N;
· c<1> + NET c<1>
 + LAYER metal3 ( 0 0 ) ( 1 1 )
 + PLACED ( -480 5400 ) N;
· d<0> + NET d<0>
 + LAYER metal3 ( 0 0 ) ( 1 1 )
  + PLACED ( -480 5600 ) N ;
· d<1> + NET d<1>
 1 I V V FD matal 2 ( A A ) ( 1 1 )
                                                     Plain Text ▼ Tab Width: 8 ▼ Ln 6, Col 29 ▼ INS
```

#### 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:

```
Test4_rca4.def
 Open ▼
                                                                                           \equiv
          Æ
  + PLACED (
- b<2> + NET b<2>
  + LAYER metal3 ( 0 0 ) ( 1 1 )
  + PLACED ( -480 6400 ) N ;
- b<3> + NET b<3>
  + LAYER metal3 ( 0 0 ) ( 1 1 )
  + PLACED ( -480 6600 ) N ;
- s<0> + NET s<0>
  + LAYER metal1 ( 0 0 ) ( 1 1 )
  + PLACED ( 18080 5200 ) N ;
- s<1> + NET s<1>
  + LAYER metal1 ( 0 0 ) ( 1 1 )
  + PLACED ( 18080 5400 ) N ;
- s<2> + NET s<2>
  + LAYER metal1 ( 0 0 ) ( 1 1 )
  + PLACED ( 18080 5600 ) N;
- s<3> + NET s<3>
  + LAYER metal1 ( 0 0 ) ( 1 1 )
  + PLACED ( 18080 5800 ) N ;
END PINS
NETS 50;
- fa0.s
  ( NAND2X1_2 Y )
  ( BUFX2_1 A );
- s<0>
  ( PIN s<0> )
  / DIIEV2 1 V \
                                                    Plain Text ▼ Tab Width: 8 ▼
                                                                                 Ln 1, Col 1 ▼ INS
```

## • 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
```

```
Test7_Simple_AOI.def
                                                                                  Save ≡ ••
 Open ▼ 🖭
COMPONENTS 1;
 OAI21X1_8 OAI21X1 + PLACED ( 0 0 ) ;
END COMPONENTS
PINS 5;
- A + NET A
  + LAYER metal3 ( 0 0 ) ( 1 1 )
  + PLACED ( -480 1600 ) N ;
- B + NET B
  + LAYER metal3 ( 0 0 ) ( 1 1 )
  + PLACED ( -480 1800 ) N ;
- C + NET C
+ LAYER metal3 ( 0 0 ) ( 1 1 )
  + PLACED ( -480 2000 ) N;
  + LAYER metal3 ( 0 0 ) ( 1 1 )
  + PLACED ( -480 2200 ) N ;
- F + NET F
 + LAYER metal1 ( 0 0 ) ( 1 1 )
 + PLACED ( 4800 1600 ) N;
- + NET
  + LAYER ( 0 0 ) ( 1 1 )
END PINS
NETS 5;
- D
  ( OAI21X1_8 A ) ;
                                                   Plain Text ▼ Tab Width: 8 ▼ Ln 1, Col 1 ▼ INS
```

#### • Test6, MUX2x1:

```
Test1_mux2x1.def
                                                                                           Save ≡ □ □
  + LAYER metal2 ( 0 0 ) ( 1 1 )
  + PLACED ( 2880 -400 ) N ;
- y<0> + NET y<0>
 + LAYER metal2 ( 0 0 ) ( 1 1 )
 + PLACED ( 3040 -400 ) N ;
- y<1> + NET y<1>
  + LAYER metal2 ( 0 0 ) ( 1 1 )
+ PLACED ( 3200 -400 ) N ;
END PINS
NETS 8 ;
- sel<0>
  ( PIN sel<0> )
  ( INVX1_1 A );
  _{1}^{1} ( INVX1_1 Y ) ;
  ( NAND2X1_1 A );
  _6_
( A0I22X1_1 Y )
  ( NAND2X1_1 B ) ;
  _11__1_
( NAND2X1_1 Y ) ;
- a<1>
  ( PIN a<1> )
  ( A0I22X1_1 A );
                                                        Plain Text ▼ Tab Width: 8 ▼ Ln 58, Col 20 ▼ INS
```

## Test7, random sample test:

```
Test5_sample.def
                                                                                          ≣
 Open ▼
          Æ
- d<1> + NET d<1>
 + LAYER metal3 ( 0 0 ) ( 1 1 )
 + PLACED ( -480 5800 ) N ;
- sel<0> + NET sel<0>
 + LAYER metal2 ( 0 0 ) ( 1 1 )
 + PLACED ( 4320 -400 ) N;
- sel<1> + NET sel<1>
 + LAYER metal2 ( 0 0 ) ( 1 1 )
 + PLACED ( 4480 -400 ) N ;
- y<0> + NET y<0>
 + LAYER metal2 ( 0 0 ) ( 1 1 )
 + PLACED ( 4640 -400 ) N;
- y<1> + NET y<1>
 + LAYER metal2 ( 0 0 ) ( 1 1 )
 + PLACED ( 4800 -400 ) N ;
END PINS
NETS 13;
- sel<0>
 ( PIN sel<0> )
  ( NOR2X1_1 A )
  ( INVX1 1 A )
  ( NAND3X1_1 A ) ;
- sel<1>
  ( PIN sel<1> )
  ( NOR2X1_2 A )
  ( NOR2X1_1 B )
                                                   Plain Text ▼ Tab Width: 8 ▼
                                                                               Ln 1, Col 1 ▼
```

# • Test8, random sample2:

```
Test6_sample2.def
 Open ▼ 🖭
                                 ~/Downloads/Verilog Netlist to DEF Format/Examples
 + PLACED ( -480 3600 ) N;
b<1> + NET b<1>
 + LAYER metal3 ( 0 0 ) ( 1 1 )
 + PLACED ( -480 3800 ) N ;
s<0> + NET s<0>
 + LAYER metal1 ( 0 0 ) ( 1 1 )
 + PLACED ( 12320 3200 ) N ;
s<1> + NET s<1>
 + LAYER metal1 ( 0 0 ) ( 1 1 )
 + PLACED ( 12320 3400 ) N ;
s<2> + NET s<2>
 + LAYER metal1 ( 0 0 ) ( 1 1 )
 + PLACED ( 12320 3600 ) N ;
END PINS
IETS 35 ;
· fa0.s
 ( NAND2X1_2 Y )
 ( BUFX2_1 A ) ;
· s<0>
 ( PIN s<0> )
 ( BUFX2_1 Y );
· fa1.s
 ( NAND2X1_4 Y )
 ( BUFX2_2 A );
· s<1>
( PTN c<15 )
                                                  Plain Text ▼ Tab Width: 8 ▼ Ln 1, Col 1 ▼ INS
```