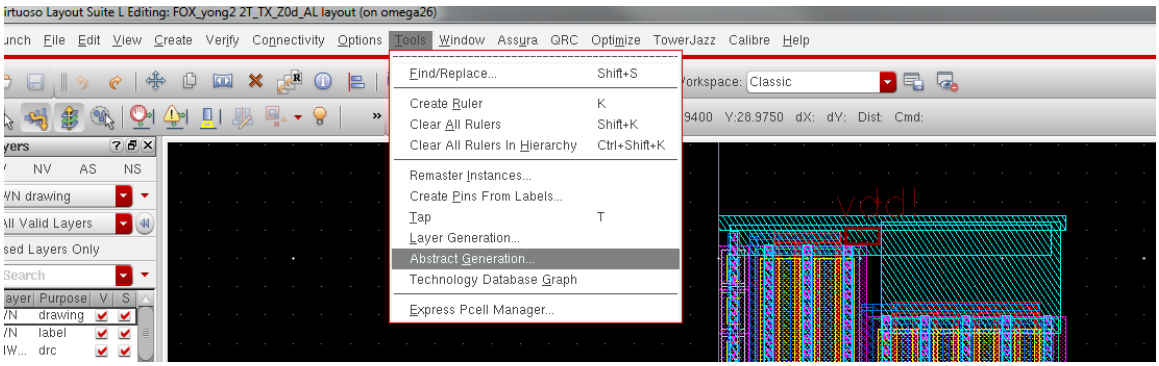
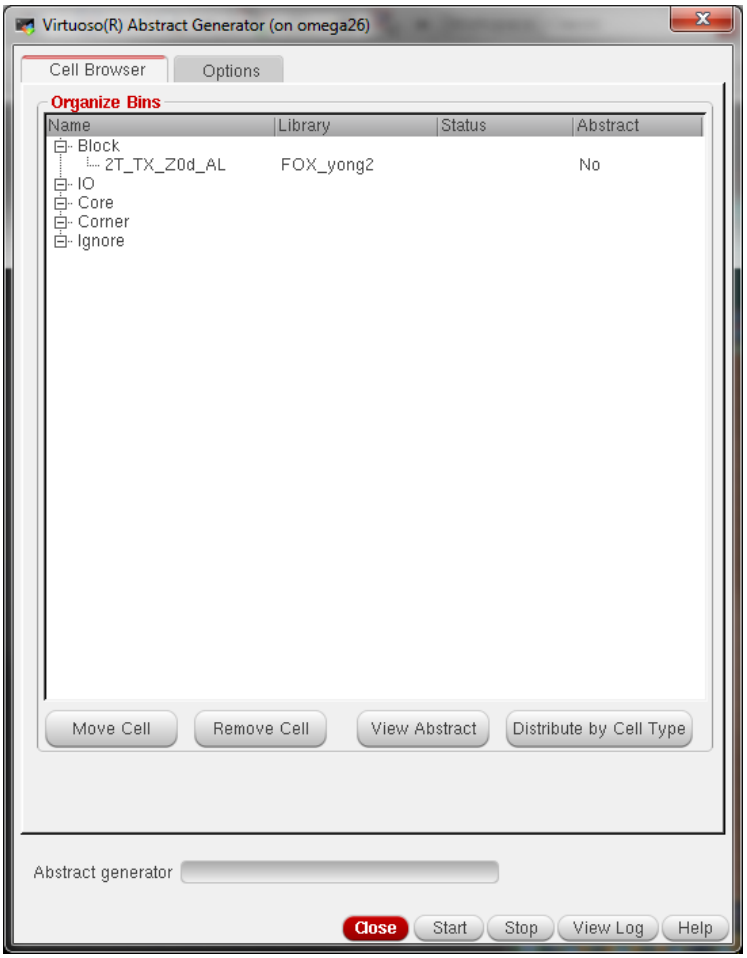
At this stage, we have had the layout of each block (we call it analog block) and we have the layout of the digital controller (from encounter)

1. Make lef file for the analog blocks
   a. Open the analog block layout

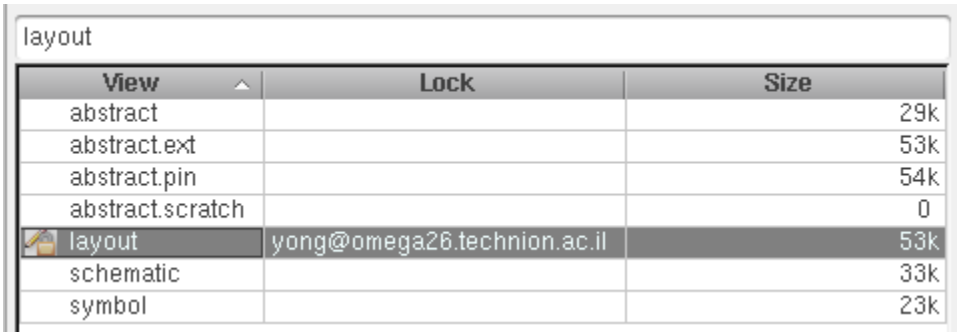

   b. Generate abstract file



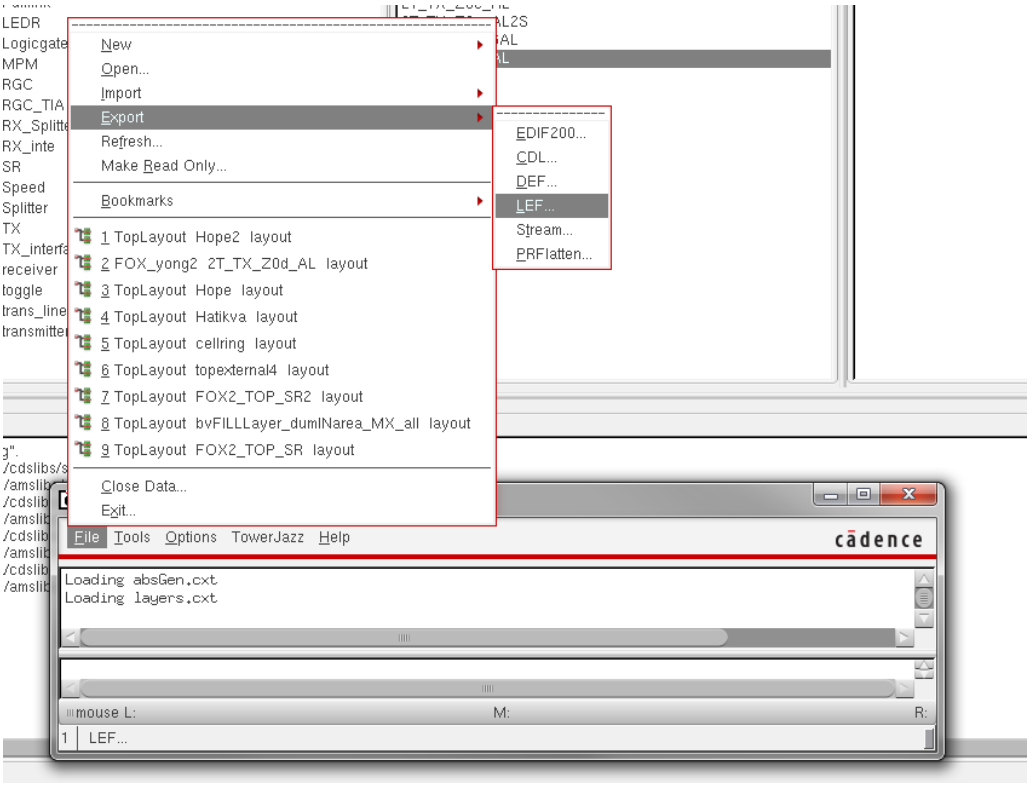



Then click → start

In this way, we can get the abstract file

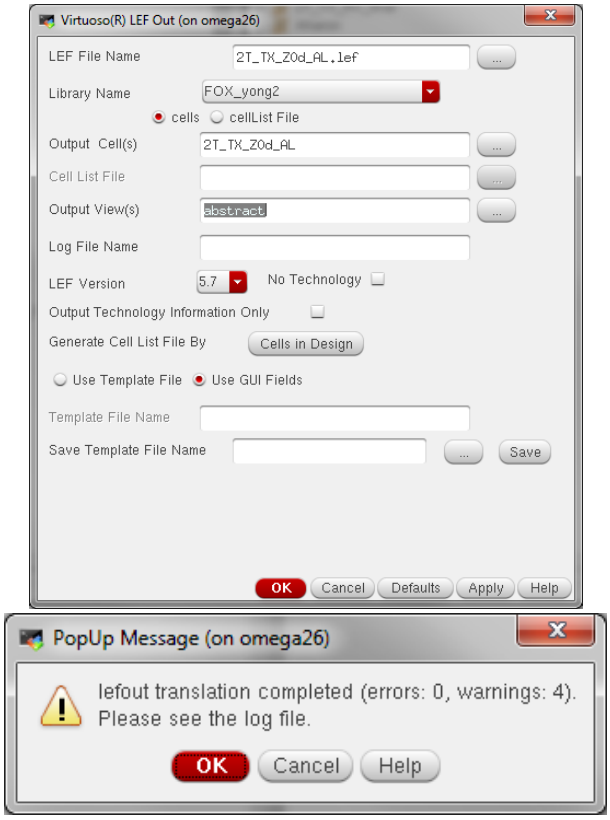



c. Export lef file




From the start window of virtuoso → export → lef

Manually set the library name, the output cell should be the cell you edited and the lef name should be the*******( cellname).lef and the output view should be abstract.



d. Edit the lef file

```
VERSION 5.7 ;
BUSBITCHARS "[]" ;
DIVIDERCHAR "/" ;

PROPERTYDEFINITIONS
  LAYER LEF58_TYPE STRING ;
  LAYER LEF58_ENCLOSURE STRING ;
  LAYER LEF58_SPACING STRING ;
  LAYER LEF58_WIDTH STRING ;
END PROPERTYDEFINITIONS

UNITS
  TIME NANOSECONDS 1 ;
  CAPACITANCE PICOFARADS 1 ;
  RESISTANCE OHMS 1 ;
  DATABASE MICRONS 1000 ;
END UNITS
MANUFACTURINGGRID 0.005 ;
LAYER OverlapCheck
  TYPE OVERLAP ;
END OverlapCheck

LAYER WN
  TYPE MASTERSLICE ;
  PROPERTY LEF58_TYPE "TYPE NWELL ;" ;
  PROPERTY LEF58_SPACING "SPACING 2.8 ;" ;
  PROPERTY LEF58_WIDTH "WIDTH 0.86 ;" ;
END WN

LAYER ACTIVE
  TYPE MASTERSLICE ;
END ACTIVE

LAYER XP
  TYPE IMPLANT ;
  WIDTH 0.44 ;
  SPACING 0.44 ;
END XP

LAYER XN
  TYPE IMPLANT ;
  WIDTH 0.44 ;
  SPACING 0.44 ;
```

```
  SPACING 0 LAYER XP ;
END XN

LAYER GC
  TYPE MASTERSLICE ;
END GC

LAYER CS
  TYPE CUT ;
  SPACING 0.25 ;
  WIDTH 0.22 ;
  ENCLOSURE BELOW 0.1 0.1 ;
  ENCLOSURE ABOVE 0.06 0.005 ;
  ANTENNAMODEL OXIDE1 ;
END CS

LAYER M1
  TYPE ROUTING ;
  DIRECTION HORIZONTAL ;
  PITCH 0.56 0.56 ;
  WIDTH 0.23 ;
  AREA 0.202 ;
  SPACING 0.23 ;
  SPACING 0.6 RANGE 10 10000 ;
  SPACING 0.23 SAMENET ;
  RESISTANCE RPERSQ 0.08 ;
  CAPACITANCE CPERSQDIST 3.51e-05 ;
  HEIGHT 1.05 ;
  THICKNESS 0.54 ;
  ANTENNAMODEL OXIDE1 ;
    ANTENNASIDEAREARATIO 400 ;
    ANTENNADIFFSIDEAREARATIO PWL ( ( 0 400 ) ( 0.203 400 ) ( 0.203001
2281.2 ) ( 1.203 2681.2 ) ) ;
END M1

LAYER V2
  TYPE CUT ;
  SPACING 0.26 ;
  SPACING 0 LAYER CS ;
  WIDTH 0.26 ;
  ENCLOSURE BELOW 0.06 0.01 ;
  ENCLOSURE ABOVE 0.06 0.01 ;
  ANTENNAMODEL OXIDE1 ;
    ANTENNAAREARATIO 20 ;
```

```
    ANTENNADIFFAREARATIO PWL ( ( 0 20 ) ( 0.203 20 ) ( 0.203001 91.9161 )
( 1.203 175.246 ) ) ;
END V2

LAYER M2
  TYPE ROUTING ;
  DIRECTION VERTICAL ;
  PITCH 0.56 0.56 ;
  WIDTH 0.28 ;
  AREA 0.202 ;
  SPACING 0.28 ;
  SPACING 0.6 RANGE 10 10000 ;
  SPACING 0.28 SAMENET ;
  RESISTANCE RPERSQ 0.08 ;
  CAPACITANCE CPERSQDIST 1.46e-05 ;
  HEIGHT 2.41 ;
  THICKNESS 0.54 ;
  ANTENNAMODEL OXIDE1 ;
    ANTENNASIDEAREARATIO 400 ;
    ANTENNADIFFSIDEAREARATIO PWL ( ( 0 400 ) ( 0.203 400 ) ( 0.203001
2281.2 ) ( 1.203 2681.2 ) ) ;
END M2

LAYER V3
  TYPE CUT ;
  SPACING 0.26 ;
  SPACING 0 SAMENET LAYER V2 STACK ;
  WIDTH 0.26 ;
  ENCLOSURE BELOW 0.06 0.01 ;
  ENCLOSURE ABOVE 0.06 0.01 ;
  ANTENNAMODEL OXIDE1 ;
    ANTENNAAREARATIO 20 ;
    ANTENNADIFFAREARATIO PWL ( ( 0 20 ) ( 0.203 20 ) ( 0.203001 91.9161 )
( 1.203 175.246 ) ) ;
END V3

LAYER M3
  TYPE ROUTING ;
  DIRECTION HORIZONTAL ;
  PITCH 0.56 0.56 ;
  WIDTH 0.28 ;
  AREA 0.202 ;
  SPACING 0.28 ;
  SPACING 0.6 RANGE 10 10000 ;
```

```
  SPACING 0.28 SAMENET ;
  RESISTANCE RPERSQ 0.08 ;
  CAPACITANCE CPERSQDIST 9.24e-06 ;
  HEIGHT 3.77 ;
  THICKNESS 0.54 ;
  ANTENNAMODEL OXIDE1 ;
    ANTENNASIDEAREARATIO 400 ;
    ANTENNADIFFSIDEAREARATIO PWL ( ( 0 400 ) ( 0.203 400 ) ( 0.203001
2281.2 ) ( 1.203 2681.2 ) ) ;
END M3

LAYER V4
  TYPE CUT ;
  SPACING 0.26 ;
  SPACING 0 SAMENET LAYER V3 STACK ;
  WIDTH 0.26 ;
  ENCLOSURE BELOW 0.06 0.01 ;
  ENCLOSURE ABOVE 0.06 0.01 ;
  ANTENNAMODEL OXIDE1 ;
    ANTENNAAREARATIO 20 ;
    ANTENNADIFFAREARATIO PWL ( ( 0 20 ) ( 0.203 20 ) ( 0.203001 91.9161 )
( 1.203 175.246 ) ) ;
END V4

LAYER M4
  TYPE ROUTING ;
  DIRECTION VERTICAL ;
  PITCH 0.56 0.56 ;
  WIDTH 0.28 ;
  AREA 0.202 ;
  SPACING 0.28 ;
  SPACING 0.6 RANGE 10 10000 ;
  SPACING 0.28 SAMENET ;
  RESISTANCE RPERSQ 0.08 ;
  CAPACITANCE CPERSQDIST 6.75e-06 ;
  HEIGHT 5.13 ;
  THICKNESS 0.54 ;
  ANTENNAMODEL OXIDE1 ;
    ANTENNASIDEAREARATIO 400 ;
    ANTENNADIFFSIDEAREARATIO PWL ( ( 0 400 ) ( 0.203 400 ) ( 0.203001
2281.2 ) ( 1.203 2681.2 ) ) ;
END M4

LAYER V5
```

```
  TYPE CUT ;
  SPACING 0.26 ;
  SPACING 0 SAMENET LAYER V4 STACK ;
  WIDTH 0.26 ;
  ENCLOSURE BELOW 0.06 0.01 ;
  ENCLOSURE ABOVE 0.06 0.01 ;
  ANTENNAMODEL OXIDE1 ;
    ANTENNAAREARATIO 20 ;
    ANTENNADIFFAREARATIO PWL ( ( 0 20 ) ( 0.203 20 ) ( 0.203001 91.9161 )
( 1.203 175.246 ) ) ;
END V5

LAYER M5
  TYPE ROUTING ;
  DIRECTION HORIZONTAL ;
  PITCH 0.84 0.84 ;
  WIDTH 0.28 ;
  AREA 0.202 ;
  SPACING 0.28 ;
  SPACING 0.6 RANGE 10 10000 ;
  SPACING 0.28 SAMENET ;
  RESISTANCE RPERSQ 0.08 ;
  CAPACITANCE CPERSQDIST 5.32e-06 ;
  HEIGHT 6.49 ;
  THICKNESS 0.54 ;
  ANTENNAMODEL OXIDE1 ;
    ANTENNASIDEAREARATIO 400 ;
    ANTENNADIFFSIDEAREARATIO PWL ( ( 0 400 ) ( 0.203 400 ) ( 0.203001
2281.2 ) ( 1.203 2681.2 ) ) ;
END M5

LAYER TOP_V
  TYPE CUT ;
  SPACING 0.35 ;
  SPACING 0 SAMENET LAYER V5 STACK ;
  WIDTH 0.36 ;
  ENCLOSURE BELOW 0.06 0.01 ;
  ENCLOSURE ABOVE 0.09 0.09 ;
  ANTENNAMODEL OXIDE1 ;
    ANTENNAAREARATIO 20 ;
    ANTENNADIFFAREARATIO PWL ( ( 0 20 ) ( 0.203 20 ) ( 0.203001 91.9161 )
( 1.203 175.246 ) ) ;
END TOP_V
```

```
LAYER TOP_M
  TYPE ROUTING ;
  DIRECTION VERTICAL ;
  PITCH 1.12 1.12 ;
  WIDTH 0.44 ;
  AREA 0.562 ;
  SPACING 0.46 ;
  SPACING 0.6 RANGE 10 10000 ;
  SPACING 0.46 SAMENET ;
  RESISTANCE RPERSQ 0.04 ;
  CAPACITANCE CPERSQDIST 4.389e-06 ;
  HEIGHT 7.83 ;
  THICKNESS 0.84 ;
  ANTENNAMODEL OXIDE1 ;
    ANTENNASIDEAREARATIO 400 ;
    ANTENNADIFFSIDEAREARATIO PWL ( ( 0 400 ) ( 0.203 400 ) ( 0.203001
31624 ) ( 1.203 39624 ) ) ;
END TOP_M

VIARULE ML_M5_S GENERATE DEFAULT
  LAYER M5 ;
    ENCLOSURE 0.01 0.01 ;
  LAYER TOP_M ;
    ENCLOSURE 0.09 0.09 ;
  LAYER TOP_V ;
    RECT -0.18 -0.18 0.18 0.18 ;
    SPACING 0.71 BY 0.71 ;
END ML_M5_S

VIARULE M5_M4_S GENERATE DEFAULT
  LAYER M4 ;
    ENCLOSURE 0.01 0.01 ;
  LAYER M5 ;
    ENCLOSURE 0.01 0.01 ;
  LAYER V5 ;
    RECT -0.13 -0.13 0.13 0.13 ;
    SPACING 0.52 BY 0.52 ;
END M5_M4_S

VIARULE M4_M3_S GENERATE DEFAULT
  LAYER M3 ;
    ENCLOSURE 0.01 0.01 ;
  LAYER M4 ;
    ENCLOSURE 0.01 0.01 ;
```

```
  LAYER V4 ;
    RECT -0.13 -0.13 0.13 0.13 ;
    SPACING 0.52 BY 0.52 ;
END M4_M3_S

VIARULE M3_M2_S GENERATE DEFAULT
  LAYER M2 ;
    ENCLOSURE 0.01 0.01 ;
  LAYER M3 ;
    ENCLOSURE 0.01 0.01 ;
  LAYER V3 ;
    RECT -0.13 -0.13 0.13 0.13 ;
    SPACING 0.52 BY 0.52 ;
END M3_M2_S

VIARULE M2_M1_S GENERATE DEFAULT
  LAYER M1 ;
    ENCLOSURE 0.01 0.01 ;
  LAYER M2 ;
    ENCLOSURE 0.01 0.01 ;
  LAYER V2 ;
    RECT -0.13 -0.13 0.13 0.13 ;
    SPACING 0.52 BY 0.52 ;
END M2_M1_S

VIARULE ML_M5 GENERATE
  LAYER M5 ;
    ENCLOSURE 0.06 0.06 ;
  LAYER TOP_M ;
    ENCLOSURE 0.09 0.09 ;
  LAYER TOP_V ;
    RECT -0.18 -0.18 0.18 0.18 ;
    SPACING 0.71 BY 0.71 ;
END ML_M5

VIARULE M5_M4 GENERATE
  LAYER M4 ;
    ENCLOSURE 0.06 0.06 ;
  LAYER M5 ;
    ENCLOSURE 0.06 0.06 ;
  LAYER V5 ;
    RECT -0.13 -0.13 0.13 0.13 ;
    SPACING 0.52 BY 0.52 ;
END M5_M4
```

```
VIARULE M4_M3 GENERATE
  LAYER M3 ;
    ENCLOSURE 0.06 0.06 ;
  LAYER M4 ;
    ENCLOSURE 0.06 0.06 ;
  LAYER V4 ;
    RECT -0.13 -0.13 0.13 0.13 ;
    SPACING 0.52 BY 0.52 ;
END M4_M3

VIARULE M3_M2 GENERATE
  LAYER M2 ;
    ENCLOSURE 0.06 0.06 ;
  LAYER M3 ;
    ENCLOSURE 0.06 0.06 ;
  LAYER V3 ;
    RECT -0.13 -0.13 0.13 0.13 ;
    SPACING 0.52 BY 0.52 ;
END M3_M2

VIARULE M2_M1 GENERATE
  LAYER M1 ;
    ENCLOSURE 0.06 0.06 ;
  LAYER M2 ;
    ENCLOSURE 0.06 0.06 ;
  LAYER V2 ;
    RECT -0.13 -0.13 0.13 0.13 ;
    SPACING 0.52 BY 0.52 ;
END M2_M1

VIARULE M1_PO_4X GENERATE
  LAYER GC ;
    ENCLOSURE 0.1 0.1 ;
  LAYER M1 ;
    ENCLOSURE 0.06 0.06 ;
  LAYER CS ;
    RECT -0.11 -0.11 0.11 0.11 ;
    SPACING 0.5 BY 0.5 ;
END M1_PO_4X

VIARULE M1_PO GENERATE
  LAYER GC ;
    ENCLOSURE 0.1 0.1 ;
```

```
  LAYER M1 ;
    ENCLOSURE 0.06 0.06 ;
  LAYER CS ;
    RECT -0.11 -0.11 0.11 0.11 ;
    SPACING 0.47 BY 0.47 ;
END M1_PO

VIARULE PTAP_4X_LG GENERATE
  LAYER ACTIVE ;
    ENCLOSURE 0.1 0.1 ;
  LAYER M1 ;
    ENCLOSURE 0.06 0.06 ;
  LAYER CS ;
    RECT -0.11 -0.11 0.11 0.11 ;
    SPACING 0.5 BY 0.5 ;
END PTAP_4X_LG

VIARULE PTAP_4X GENERATE
  LAYER ACTIVE ;
    ENCLOSURE 0.1 0.1 ;
  LAYER M1 ;
    ENCLOSURE 0.06 0.06 ;
  LAYER CS ;
    RECT -0.11 -0.11 0.11 0.11 ;
    SPACING 0.5 BY 0.5 ;
END PTAP_4X

VIARULE PTAP_LG GENERATE
  LAYER ACTIVE ;
    ENCLOSURE 0.1 0.1 ;
  LAYER M1 ;
    ENCLOSURE 0.06 0.06 ;
  LAYER CS ;
    RECT -0.11 -0.11 0.11 0.11 ;
    SPACING 0.47 BY 0.47 ;
END PTAP_LG

VIARULE PTAP GENERATE
  LAYER ACTIVE ;
    ENCLOSURE 0.1 0.1 ;
  LAYER M1 ;
    ENCLOSURE 0.06 0.06 ;
  LAYER CS ;
    RECT -0.11 -0.11 0.11 0.11 ;
```

```
    SPACING 0.47 BY 0.47 ;
END PTAP

VIARULE NTAP_4X_LG GENERATE
  LAYER ACTIVE ;
    ENCLOSURE 0.1 0.1 ;
  LAYER M1 ;
    ENCLOSURE 0.06 0.06 ;
  LAYER CS ;
    RECT -0.11 -0.11 0.11 0.11 ;
    SPACING 0.5 BY 0.5 ;
END NTAP_4X_LG

VIARULE NTAP_4X GENERATE
  LAYER ACTIVE ;
    ENCLOSURE 0.1 0.1 ;
  LAYER M1 ;
    ENCLOSURE 0.06 0.06 ;
  LAYER CS ;
    RECT -0.11 -0.11 0.11 0.11 ;
    SPACING 0.5 BY 0.5 ;
END NTAP_4X

VIARULE NTAP_LG GENERATE
  LAYER ACTIVE ;
    ENCLOSURE 0.1 0.1 ;
  LAYER M1 ;
    ENCLOSURE 0.06 0.06 ;
  LAYER CS ;
    RECT -0.11 -0.11 0.11 0.11 ;
    SPACING 0.47 BY 0.47 ;
END NTAP_LG

VIARULE NTAP GENERATE
  LAYER ACTIVE ;
    ENCLOSURE 0.1 0.1 ;
  LAYER M1 ;
    ENCLOSURE 0.06 0.06 ;
  LAYER CS ;
    RECT -0.11 -0.11 0.11 0.11 ;
    SPACING 0.47 BY 0.47 ;
END NTAP

VIARULE M1_WN GENERATE
```

```
  LAYER ACTIVE ;
    ENCLOSURE 0.1 0.1 ;
  LAYER M1 ;
    ENCLOSURE 0.06 0.06 ;
  LAYER CS ;
    RECT -0.11 -0.11 0.11 0.11 ;
    SPACING 0.47 BY 0.47 ;
END M1_WN

SITE CoreSite_hv
  CLASS CORE ;
  SYMMETRY Y ;
  SIZE 0.56 BY 6.16 ;
END CoreSite_hv

MACRO 2T_TX_Z0d_AL
  CLASS BLOCK ;
  ORIGIN 0 0 ;
  FOREIGN 2T_TX_Z0d_AL 0 0 ;
  SIZE 12.25 BY 26.5 ;
  SYMMETRY X Y R90 ;
  PIN In
    DIRECTION INOUT ;
    USE SIGNAL ;
    PORT
      LAYER M1 ;
        RECT 1.115 6.85 1.345 7.305 ;
    END
  END In
  PIN gnd!
    DIRECTION INOUT ;
    USE GROUND ;
    PORT
      LAYER M2 ;
        RECT 5.13 0 5.365 1.105 ;
        RECT 4.735 0 6.22 0.715 ;
    END
  END gnd!
  PIN Out
    DIRECTION INOUT ;
    USE SIGNAL ;
    PORT
      LAYER M1 ;
        RECT 10.275 6.69 10.405 7.31 ;
```

```
      RECT 10.275 6.865 11.005 7.15 ;
    END
  END Out
  PIN vdd!
    DIRECTION INOUT ;
    USE POWER ;
    PORT
      LAYER M2 ;
        RECT 5.13 25.105 5.31 26.065 ;
        RECT 4.49 25.51 5.67 26.065 ;
    END
  END vdd!
  OBS
    LAYER M1 ;
      RECT 0.335 8.975 0.715 25.895 ;
      RECT 0.4 0.43 0.78 6.43 ;
      RECT 1.115 7.905 1.345 25.775 ;
      RECT 1.115 0.77 1.345 6.25 ;
      RECT 1.575 1.54 1.805 6.31 ;
      RECT 1.575 7.54 1.805 25.42 ;
      RECT 1.575 7.98 1.955 25.42 ;
    LAYER M1 SPACING 0.23 ;
      RECT 0 0 12.25 6.32 ;
      RECT 0 0 10.285 6.46 ;
      RECT 0 0 9.905 6.48 ;
      RECT 10.775 0 12.25 6.495 ;
      RECT 1.715 0 9.905 26.5 ;
      RECT 1.575 7.54 10.285 26.5 ;
      RECT 0 7.675 10.285 26.5 ;
      RECT 10.775 7.52 12.25 26.5 ;
      RECT 11.375 0 12.25 26.5 ;
      RECT 0 7.68 12.25 26.5 ;
    LAYER M2 ;
      RECT 1.625 7.93 1.905 26.5 ;
      RECT 0.385 25.105 3.89 26.5 ;
      RECT 0 26 3.89 26.5 ;
      RECT 0 0 4.135 0.5 ;
      RECT 0.45 0 4.135 1.105 ;
      RECT 0.45 0 0.73 6.48 ;
      RECT 4.43 7.93 4.71 24.505 ;
      RECT 4.43 1.705 4.71 6.09 ;
      RECT 5.785 1.705 6.065 6.09 ;
      RECT 10.02 8.03 10.3 26.5 ;
      RECT 11.465 8.03 11.745 26.5 ;
```

```
      RECT 5.73 22.35 12.045 24.505 ;
      RECT 5.91 22.35 12.045 24.91 ;
      RECT 6.27 22.35 12.045 26.5 ;
      RECT 6.27 26 12.25 26.5 ;
      RECT 6.82 0 12.25 0.5 ;
      RECT 6.82 0 11.75 1.105 ;
      RECT 10.03 0 10.31 6.09 ;
      RECT 11.47 0 11.75 6.09 ;
    LAYER M2 SPACING 0.28 ;
      RECT 6.64 0 12.25 6.27 ;
      RECT 0 0 4.315 6.43 ;
      RECT 0.45 0 0.73 6.48 ;
      RECT 1.765 1.525 9.855 24.685 ;
      RECT 1.765 1.135 4.71 25.09 ;
      RECT 5.785 1.135 9.855 25.09 ;
      RECT 5.73 7.73 12.25 25.09 ;
      RECT 0 7.725 4.07 26.5 ;
      RECT 6.09 7.73 12.25 26.5 ;
    LAYER M3 SPACING 0.28 ;
      RECT 0 0 4.315 25.09 ;
      RECT 0 1.525 12.25 24.685 ;
      RECT 0 1.135 4.71 25.09 ;
      RECT 5.785 1.135 12.25 25.09 ;
      RECT 5.73 1.525 12.25 25.09 ;
      RECT 0 0 4.07 26.5 ;
      RECT 6.64 0 12.25 26.5 ;
      RECT 6.09 1.135 12.25 26.5 ;
    LAYER M4 SPACING 0.28 ;
      RECT 0 0 12.25 26.5 ;
    LAYER M5 SPACING 0.28 ;
      RECT 0 0 12.25 26.5 ;
    LAYER TOP_M SPACING 0.46 ;
      RECT 0 0 12.25 26.5 ;
  END
END 2T_TX_Z0d_AL

END LIBRARY
```

Delete the blue lines of the lef file (note that if there is some lines with property before th end library, you should also delete it).

2. Make lef file for the digital controller
   This is an easy process, first, open the layout of your digital controller in encounter and then use the command

lefOut ***.lef

That's it

3. Make the top.io file based on your chip

```
#########################################################
#                                       #
#  Silicon Perspective, A Cadence Company          #
#  FirstEncounter IO Assignment                  #
#                                       #
#########################################################

Version: 2of

Pad: PAD_G00   E
Pad: PAD_I00   E
Pad: PAD_G01   E
Pad: PAD_I01   E
Pad: PAD_G02   E
Pad: I10              E
Pad: I7               E
Pad: I12              E
Pad: PAD_I02   E
Pad: I6               E
Pad: PAD_I03   E
Pad: I9               E
Orient: R90
Pad: Pcornerlr NE

Pad: PAD_I07   N
Pad: PAD_G06   N
Pad: PAD_I06   N
Pad: PAD_G05   N
Pad: PAD_I05   N
Pad: PAD_I04   N
Pad: I8               N
Pad: I16              N
Pad: PAD_G04   N
Pad: I15              N
Pad: PAD_G03   N
Pad: I17              N
Orient: R180
Pad: Pcornerur NW
```

Pad: PAD_G10 W
Pad: PAD_I11 W
Pad: PAD_G09 W
Pad: PAD_I10 W
Pad: PAD_G08 W
Pad: PAD_I09 W
Pad: I5 W
Pad: I18 W
Pad: PAD_I08 W
Pad: I14 W
Pad: I13 W
Pad: PAD_G07 W
Orient: R270
Pad: Pcornerul SW

Pad: PAD_I12 S
Pad: PAD_G11 S
Pad: PAD_I13 S
Pad: PAD_G12 S
Pad: PAD_I14 S
Pad: PAD_I15 S
Pad: PAD_I16 S
Pad: PAD_G13 S
Pad: PAD_I17 S
Pad: I11 S
Pad: PAD_I18 S
Pad: PAD_G14 S
Orient: R0
Pad: Pcornerll SE

The PAD_I or PAD_G stands for the power pads (VDD and VSS)

4. Make a final top level schematic of your top circuit

   This is an easy step, but make sure you make the lef file for all the first level blocks you used in the top level schematic.

5. Make an equivalent verilog file for the top level schematic (for example, call it core.v)

   For this verilog log, we just need to illustrate the connection of the top level circuit, so it is not hard. But since we make it manually, take care of it, it's easy to make mistakes

```
module core (CLK, DIN, RDEN, RESET, START, WREN, TX_IN, CGC_TEST, DOUT, ENDF, CLKOS, RGC_OUT, A2T_OUT, CGC_OUT);
input   CLK, DIN, RDEN, RESET, START, WREN, TX_IN, CGC_TEST;

output  DOUT, ENDF, CLKOS, RGC_OUT, A2T_OUT, CGC_OUT;

wire [29:0]     latch_en;
wire [15:0]     RX_data00, RX_data01, RX_data02, RX_data03, RX_data04, RX_data05, RX_data06, RX_data07, RX_data08, RX_data09, RX_data10, RX_data11, RX_data12, RX_data13, RX_data14, RX_da
wire [3:0]      cgc_fine;
wire [6:0]      cgc_slow;
wire [29:0]     load_en;

wire RING_OS_EN, LINK_RESET;

wire [29:0] send_en;
wire [15:0] TX_data;

wire d00, dn00, s00, sn00, d01, dn01, s01, sn01, d02, dn02, s02, sn02, d03, dn03, s03, sn03, d04, dn04, s04, sn04, d05, dn05, s05, sn05, d06, dn06, s06, sn06, d07, dn07, s07, sn07, d08,

wire b00, bn00, r00, rn00, b01, bn01, r01, rn01, b02, bn02, r02, rn02, b03, bn03, r03, rn03, b04, bn04, r04, rn04, b05, bn05, r05, rn05, b06, bn06, r06, rn06, b07, bn07, r07, rn07, b08,

wire s2d1, s2d2, tx101, tx1o2, rxo1, txo2;

wire high, low, cgct0, cgct90, cgctn0, cgctn90, mpwt, mpwtn;

fox2_test_ctrl u0 (.CLK(CLK), .DIN(DIN),
.LATCH_EN_29(latch_en[29]), .LATCH_EN_28(latch_en[28]), .LATCH_EN_27(latch_en[27]), .LATCH_EN_26(latch_en[26]), .LATCH_EN_25(latch_en[25]), .LATCH_EN_24(latch_en[24]), .LATCH_EN_23(latch
.RDEN(RDEN), .RESET(RESET),

.RX_DATA_00_15(RX_data00[15]), .RX_DATA_00_14(RX_data00[14]), .RX_DATA_00_13(RX_data00[13]), .RX_DATA_00_12(RX_data00[12]), .RX_DATA_00_11(RX_data00[11]), .RX_DATA_00_10(RX_data00[10]),
.RX_DATA_01_15(RX_data01[15]), .RX_DATA_01_14(RX_data01[14]), .RX_DATA_01_13(RX_data01[13]), .RX_DATA_01_12(RX_data01[12]), .RX_DATA_01_11(RX_data01[11]), .RX_DATA_01_10(RX_data01[10]),
.RX_DATA_02_15(RX_data02[15]), .RX_DATA_02_14(RX_data02[14]), .RX_DATA_02_13(RX_data02[13]), .RX_DATA_02_12(RX_data02[12]), .RX_DATA_02_11(RX_data02[11]), .RX_DATA_02_10(RX_data02[10]),
.RX_DATA_03_15(RX_data03[15]), .RX_DATA_03_14(RX_data03[14]), .RX_DATA_03_13(RX_data03[13]), .RX_DATA_03_12(RX_data03[12]), .RX_DATA_03_11(RX_data03[11]), .RX_DATA_03_10(RX_data03[10]),
.RX_DATA_04_15(RX_data04[15]), .RX_DATA_04_14(RX_data04[14]), .RX_DATA_04_13(RX_data04[13]), .RX_DATA_04_12(RX_data04[12]), .RX_DATA_04_11(RX_data04[11]), .RX_DATA_04_10(RX_data04[10]),
.RX_DATA_05_15(RX_data05[15]), .RX_DATA_05_14(RX_data05[14]), .RX_DATA_05_13(RX_data05[13]), .RX_DATA_05_12(RX_data05[12]), .RX_DATA_05_11(RX_data05[11]), .RX_DATA_05_10(RX_data05[10]),
.RX_DATA_06_15(RX_data06[15]), .RX_DATA_06_14(RX_data06[14]), .RX_DATA_06_13(RX_data06[13]), .RX_DATA_06_12(RX_data06[12]), .RX_DATA_06_11(RX_data06[11]), .RX_DATA_06_10(RX_data06[10]),
.RX_DATA_07_15(RX_data07[15]), .RX_DATA_07_14(RX_data07[14]), .RX_DATA_07_13(RX_data07[13]), .RX_DATA_07_12(RX_data07[12]), .RX_DATA_07_11(RX_data07[11]), .RX_DATA_07_10(RX_data07[10]),
.RX_DATA_08_15(RX_data08[15]), .RX_DATA_08_14(RX_data08[14]), .RX_DATA_08_13(RX_data08[13]), .RX_DATA_08_12(RX_data08[12]), .RX_DATA_08_11(RX_data08[11]), .RX_DATA_08_10(RX_data08[10]),
.RX_DATA_09_15(RX_data09[15]), .RX_DATA_09_14(RX_data09[14]), .RX_DATA_09_13(RX_data09[13]), .RX_DATA_09_12(RX_data09[12]), .RX_DATA_09_11(RX_data09[11]), .RX_DATA_09_10(RX_data09[10]),
.RX_DATA_10_15(RX_data10[15]), .RX_DATA_10_14(RX_data10[14]), .RX_DATA_10_13(RX_data10[13]), .RX_DATA_10_12(RX_data10[12]), .RX_DATA_10_11(RX_data10[11]), .RX_DATA_10_10(RX_data10[10]),
.RX_DATA_11_15(RX_data11[15]), .RX_DATA_11_14(RX_data11[14]), .RX_DATA_11_13(RX_data11[13]), .RX_DATA_11_12(RX_data11[12]), .RX_DATA_11_11(RX_data11[11]), .RX_DATA_11_10(RX_data11[10]),
.RX_DATA_12_15(RX_data12[15]), .RX_DATA_12_14(RX_data12[14]), .RX_DATA_12_13(RX_data12[13]), .RX_DATA_12_12(RX_data12[12]), .RX_DATA_12_11(RX_data12[11]), .RX_DATA_12_10(RX_data12[10]),
.RX_DATA_13_15(RX_data13[15]), .RX_DATA_13_14(RX_data13[14]), .RX_DATA_13_13(RX_data13[13]), .RX_DATA_13_12(RX_data13[12]), .RX_DATA_13_11(RX_data13[11]), .RX_DATA_13_10(RX_data13[10]),
.RX_DATA_14_15(RX_data14[15]), .RX_DATA_14_14(RX_data14[14]), .RX_DATA_14_13(RX_data14[13]), .RX_DATA_14_12(RX_data14[12]), .RX_DATA_14_11(RX_data14[11]), .RX_DATA_14_10(RX_data14[10]),
.RX_DATA_15_15(RX_data15[15]), .RX_DATA_15_14(RX_data15[14]), .RX_DATA_15_13(RX_data15[13]), .RX_DATA_15_12(RX_data15[12]), .RX_DATA_15_11(RX_data15[11]), .RX_DATA_15_10(RX_data15[10]),
.RX_DATA_16_15(RX_data16[15]), .RX_DATA_16_14(RX_data16[14]), .RX_DATA_16_13(RX_data16[13]), .RX_DATA_16_12(RX_data16[12]), .RX_DATA_16_11(RX_data16[11]), .RX_DATA_16_10(RX_data16[10]),
.RX_DATA_17_15(RX_data17[15]), .RX_DATA_17_14(RX_data17[14]), .RX_DATA_17_13(RX_data17[13]), .RX_DATA_17_12(RX_data17[12]), .RX_DATA_17_11(RX_data17[11]), .RX_DATA_17_10(RX_data17[10]),
.RX_DATA_18_15(RX_data18[15]), .RX_DATA_18_14(RX_data18[14]), .RX_DATA_18_13(RX_data18[13]), .RX_DATA_18_12(RX_data18[12]), .RX_DATA_18_11(RX_data18[11]), .RX_DATA_18_10(RX_data18[10]),
.RX_DATA_19_15(RX_data19[15]), .RX_DATA_19_14(RX_data19[14]), .RX_DATA_19_13(RX_data19[13]), .RX_DATA_19_12(RX_data19[12]), .RX_DATA_19_11(RX_data19[11]), .RX_DATA_19_10(RX_data19[10]),
.RX_DATA_20_15(RX_data20[15]), .RX_DATA_20_14(RX_data20[14]), .RX_DATA_20_13(RX_data20[13]), .RX_DATA_20_12(RX_data20[12]), .RX_DATA_20_11(RX_data20[11]), .RX_DATA_20_10(RX_data20[10]),
.RX_DATA_21_15(RX_data21[15]), .RX_DATA_21_14(RX_data21[14]), .RX_DATA_21_13(RX_data21[13]), .RX_DATA_21_12(RX_data21[12]), .RX_DATA_21_11(RX_data21[11]), .RX_DATA_21_10(RX_data21[10]),
.RX_DATA_22_15(RX_data22[15]), .RX_DATA_22_14(RX_data22[14]), .RX_DATA_22_13(RX_data22[13]), .RX_DATA_22_12(RX_data22[12]), .RX_DATA_22_11(RX_data22[11]), .RX_DATA_22_10(RX_data22[10]),
.RX_DATA_23_15(RX_data23[15]), .RX_DATA_23_14(RX_data23[14]), .RX_DATA_23_13(RX_data23[13]), .RX_DATA_23_12(RX_data23[12]), .RX_DATA_23_11(RX_data23[11]), .RX_DATA_23_10(RX_data23[10]),
.RX_DATA_24_15(RX_data24[15]), .RX_DATA_24_14(RX_data24[14]), .RX_DATA_24_13(RX_data24[13]), .RX_DATA_24_12(RX_data24[12]), .RX_DATA_24_11(RX_data24[11]), .RX_DATA_24_10(RX_data24[10]),
.RX_DATA_25_15(RX_data25[15]), .RX_DATA_25_14(RX_data25[14]), .RX_DATA_25_13(RX_data25[13]), .RX_DATA_25_12(RX_data25[12]), .RX_DATA_25_11(RX_data25[11]), .RX_DATA_25_10(RX_data25[10]),
.RX_DATA_26_15(RX_data26[15]), .RX_DATA_26_14(RX_data26[14]), .RX_DATA_26_13(RX_data26[13]), .RX_DATA_26_12(RX_data26[12]), .RX_DATA_26_11(RX_data26[11]), .RX_DATA_26_10(RX_data26[10]),
.RX_DATA_27_15(RX_data27[15]), .RX_DATA_27_14(RX_data27[14]), .RX_DATA_27_13(RX_data27[13]), .RX_DATA_27_12(RX_data27[12]), .RX_DATA_27_11(RX_data27[11]), .RX_DATA_27_10(RX_data27[10]),
.RX_DATA_28_15(RX_data28[15]), .RX_DATA_28_14(RX_data28[14]), .RX_DATA_28_13(RX_data28[13]), .RX_DATA_28_12(RX_data28[12]), .RX_DATA_28_11(RX_data28[11]), .RX_DATA_28_10(RX_data28[10]),
.RX_DATA_29_15(RX_data29[15]), .RX_DATA_29_14(RX_data29[14]), .RX_DATA_29_13(RX_data29[13]), .RX_DATA_29_12(RX_data29[12]), .RX_DATA_29_11(RX_data29[11]), .RX_DATA_29_10(RX_data29[10]),

.START(START), .WREN(WREN),
.CGC_FINE_3(cgc_fine[3]), .CGC_FINE_2(cgc_fine[2]), .CGC_FINE_1(cgc_fine[1]), .CGC_FINE_0(cgc_fine[0]),
.CGC_SLOW_6(cgc_slow[6]), .CGC_SLOW_5(cgc_slow[5]), .CGC_SLOW_4(cgc_slow[4]), .CGC_SLOW_3(cgc_slow[3]), .CGC_SLOW_2(cgc_slow[2]), .CGC_SLOW_1(cgc_slow[1]), .CGC_SLOW_0(cgc_slow[0]),
.DOUT(DOUT), .ENDF(ENDF),

.LINK_LOAD_EN_29(load_en[29]), .LINK_LOAD_EN_28(load_en[28]), .LINK_LOAD_EN_27(load_en[27]), .LINK_LOAD_EN_26(load_en[26]), .LINK_LOAD_EN_25(load_en[25]), .LINK_LOAD_EN_24(load_en[24]),

.LINK_RESET(LINK_RESET), .RING_OS_EN(RING_OS_EN),

.SEND_WORD_EN_29(send_en[29]), .SEND_WORD_EN_28(send_en[28]), .SEND_WORD_EN_27(send_en[27]), .SEND_WORD_EN_26(send_en[26]), .SEND_WORD_EN_25(send_en[25]), .SEND_WORD_EN_24(send_en[24]),
```

An example for this top level verilog

6. Connect the pads with the inputs and outputs of your top level circuit

   Since the pads are blocks and we need to connect the pads with the inputs and outputs of the core, so we need to make connection of them

   module top();

   wire wire_CGC_TEST;
   wire wire_CLK;
   wire wire_DIN;
   wire wire_RDEN;
   wire wire_RESET;
   wire wire_START;
   wire wire_TX_IN;
   wire wire_WREN;
   wire wire_A2T_OUT;
   wire wire_CGC_OUT;
   wire wire_CLKOS;
   wire wire_DOUT;
   wire wire_ENDF;
   wire wire_RGC_OUT;

```
wire   net1000;
wire   net1001;

specify
   specparam CDS_LIBNAME  = "testLib";
   specparam CDS_CELLNAME = "top";
   specparam CDS_VIEWNAME = "schematic";
endspecify

core I0(.CGC_TEST(wire_CGC_TEST)
,.CLK(wire_CLK)
,.DIN(wire_DIN)
,.RDEN(wire_RDEN)
,.RESET(wire_RESET)
,.START(wire_START)
,.TX_IN(wire_TX_IN)
,.WREN(wire_WREN)
,.A2T_OUT(wire_A2T_OUT)
,.CGC_OUT(wire_CGC_OUT)
,.CLKOS(wire_CLKOS)
,.DOUT(wire_DOUT)
,.ENDF(wire_ENDF)
,.RGC_OUT(wire_RGC_OUT)
);

pv0i PAD_G00 (.VSS(VSS));
pv0a PAD_G01 (.VSS(VSS));
pv0i PAD_G02 (.VSS(VSS));
pv0a PAD_G03 (.VSS(VSS));
pv0i PAD_G04 (.VSS(VSS));
pv0a PAD_G05 (.VSS(VSS));
pv0i PAD_G06 (.VSS(VSS));
pv0a PAD_G07 (.VSS(VSS));
pv0i PAD_G08 (.VSS(VSS));
pv0a PAD_G09 (.VSS(VSS));
pv0i PAD_G10 (.VSS(VSS));
pv0a PAD_G11 (.VSS(VSS));
pv0i PAD_G12 (.VSS(VSS));
pv0a PAD_G13 (.VSS(VSS));
pv0i PAD_G14 (.VSS(VSS));

pvdi PAD_I00 (.VDD(VDD));
pvda PAD_I01 (.VDD(VDD));
pvdi PAD_I02 (.VDD(VDD));
```

```
pvda PAD_I03 (.VDD(VDD));
pvdi PAD_I04 (.VDD(VDD));
pvda PAD_I05 (.VDD(VDD));
pvdi PAD_I06 (.VDD(VDD));
pvda PAD_I07 (.VDD(VDD));
pvdi PAD_I08 (.VDD(VDD));
pvda PAD_I09 (.VDD(VDD));
pvdi PAD_I10 (.VDD(VDD));
pvda PAD_I11 (.VDD(VDD));
pvdi PAD_I12 (.VDD(VDD));
pvda PAD_I13 (.VDD(VDD));
pvdi PAD_I14 (.VDD(VDD));
pvda PAD_I15 (.VDD(VDD));
pvdi PAD_I16 (.VDD(VDD));
pvda PAD_I17 (.VDD(VDD));
pvdi PAD_I18 (.VDD(VDD));

pc3d01 I5 ( .CIN(wire_CGC_TEST), .PAD(net100));
pc3d01 I6 ( .CIN(wire_CLK), .PAD(net101));
pc3d01 I7 ( .CIN(wire_DIN), .PAD(net102));
pc3d01 I8 ( .CIN(wire_RDEN), .PAD(net103));
pc3d01 I9 ( .CIN(wire_RESET), .PAD(net104));
pc3d01 I10 ( .CIN(wire_START), .PAD(net105));
pc3d01 I11 ( .CIN(wire_TX_IN), .PAD(net106));
pc3d01 I12 ( .CIN(wire_WREN), .PAD(net107));

pt3o01 I13 ( .PAD(net108), .I(wire_A2T_OUT));
pt3o01 I14 ( .PAD(net109), .I(wire_CGC_OUT));
pt3o01 I15 ( .PAD(net110), .I(wire_CLKOS));
pt3o01 I16 ( .PAD(net111), .I(wire_DOUT));
pt3o01 I17 ( .PAD(net112), .I(wire_ENDF));
pt3o01 I18 ( .PAD(net113), .I(wire_RGC_OUT));

pfrelr Pcornerlr();
pfrelr Pcornerll();
pfrelr Pcornerur();
pfrelr Pcornerul();

endmodule
```

We give wire name for the pads and map the pads with the correct pins name and this is the top verilog file, we call it top.v

7. Encounter stage

a. Load file first

Almost the name as we used in the backend stage of the digital process, however, need to make some modifications.

```
set_global _enable_mmmc_by_default_flow      $CTE::mmmc_default
suppressMessage ENCEXT-2799
win
set defHierChar /
set fp_core_to_left 100
set fp_core_to_right 100
set init_oa_search_lib {}
set lsgOCPGainMult 1.000000
set conf_ioOri R0
set init_io_file ../design_in/top.io
set init_verilog { ../design_in/top.v}
#set init_verilog { ../design_in/toplayout.v}
#set init_verilog { ../design_in/top3.v}
set init_pwr_net {VDD VDDC VDDO}
set delaycal_input_transition_delay 120ps
set init_assign_buffer 1
set init_lef_file {/hm/iit/cadence/tsl018b/lef/tsl18_6lm_tech.lef /hm/iit/cadence/tsl018b/lef/tsl18cio150_6lm.lef /hm/iit/cadence/tsl018b/lef/tsl18fs120.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/fox2_test_ctrl.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/S2D_converter.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/transmitter_final.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/RSG_TIA_New_No_Inv_PMOS.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/TX_2T_TX_Z0b_6AL.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/RX_2T_RX_Z0b_6AL.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/ESD2.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/CGC2_16_FULL.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/mpm_21.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/CLKOS_FF.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/Ring2_OS_FF.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/TX_2T2X_Z0a_ALS_Beta.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/RX_2T2X_Z0a_HC_Beta.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/TX_2T2X_Z0a_6ALS_Beta.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/RX_2T2X_Z0a_6AL_Beta.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/TX_2T2X_Z0a_AL2SBeta.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/RX_2T2X_Z0a_AL2S_Beta.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/TX_2T2X_Z0b_ALS_Beta.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/RX_2T2X_Z0b_AL_Beta.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/TX_2T2X_Z0b_6ALS_Beta.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/RX_2T2X_Z0b_6AL_Beta.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/TX2_full_2T_2X_Noline_beta.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/RX_2T2X_Z0b_Noline_Beta.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/TX_RGCTIA_DFAM_AL_beta.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/RX_RGC_TIA1_Z0a_AL_Beta.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/TX_RGCTIA_DFAM_6AL_beta.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/RX_RGC_TIA1_Z0a_6AL_Beta.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/TX_RGCTIA_DFAM_ALb_beta.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/RX_RGC_TIA1_Z0b_AL_Beta.lef
/users/yong/towerOA/FOX2/work_libs/yong/cds/RX_RGC_TIA1_Z0a_Noline_Beta.lef
```

First, add the top.io in the script

Second, add the lef file in the set init_lef_file, which includes all the lef files we had

b. Add the iofill in the manual/ run the iofill script
addIoFiller -cell pfeed30000 -prefix pfeed
addIoFiller -cell pfeed10000 -prefix pfeed
addIoFiller -cell pfeed02000 -prefix pfeed
addIoFiller -cell pfeed01000 -prefix pfeed
addIoFiller -cell pfeed00540 -prefix pfeed
addIoFiller -cell pfeed00120 -prefix pfeed
addIoFiller -cell pfeed00040 -prefix pfeed
addIoFiller -cell pfeed00010 -prefix pfeed

Make sure there is no empty space between the pads after the iofill script, it is very hard to fix it in virtuoso

c. Then do the power grid generation, and route stage as usual and finally stramOut the gds file as we did before for the digital controller

8. DRC

   The same process we did to the digital controller, but if the power grid and the connection of the power grid to the pads are not connected. You must fix it manually in encounter

   a. Dummy fill if needed

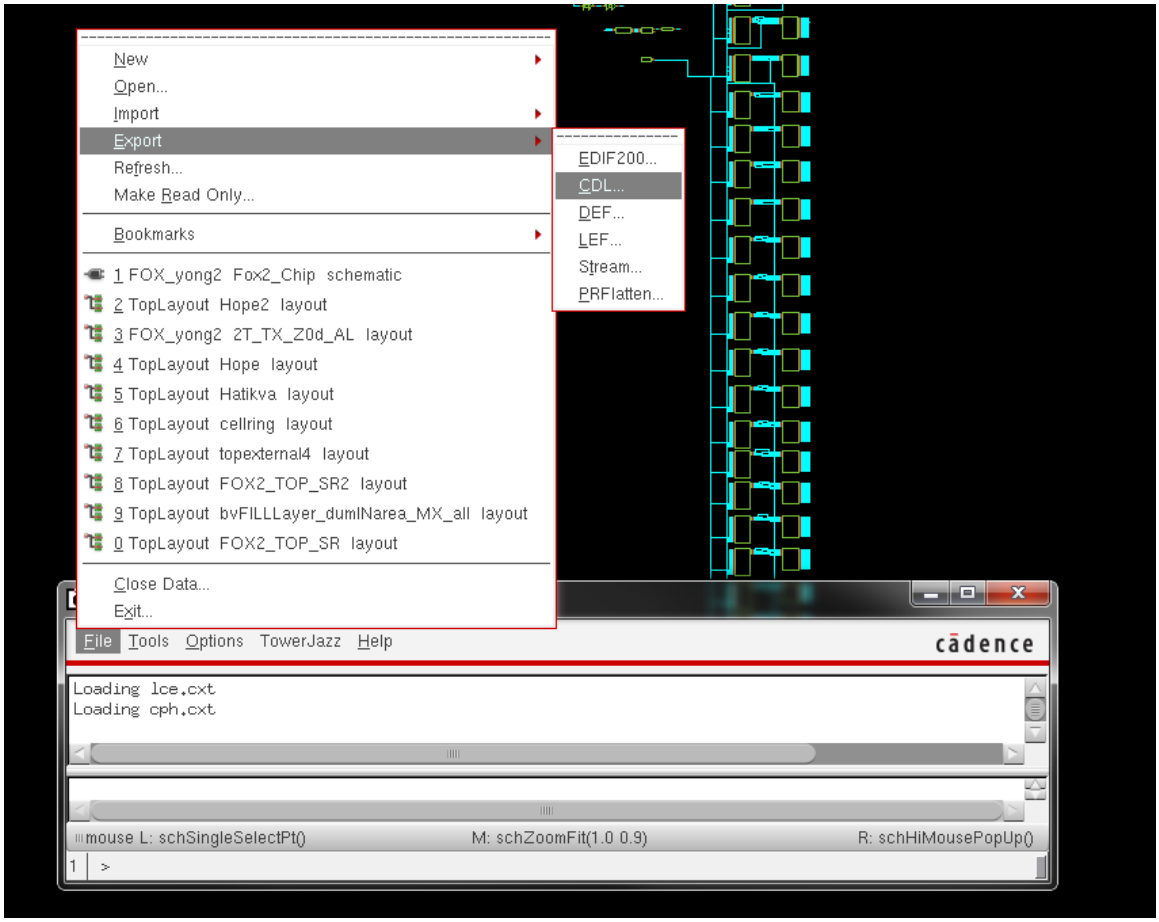
      Since this is the final chip layout, sometimes it is needed to add the dummy layers for the fabrication consideration. For details, consult Goel

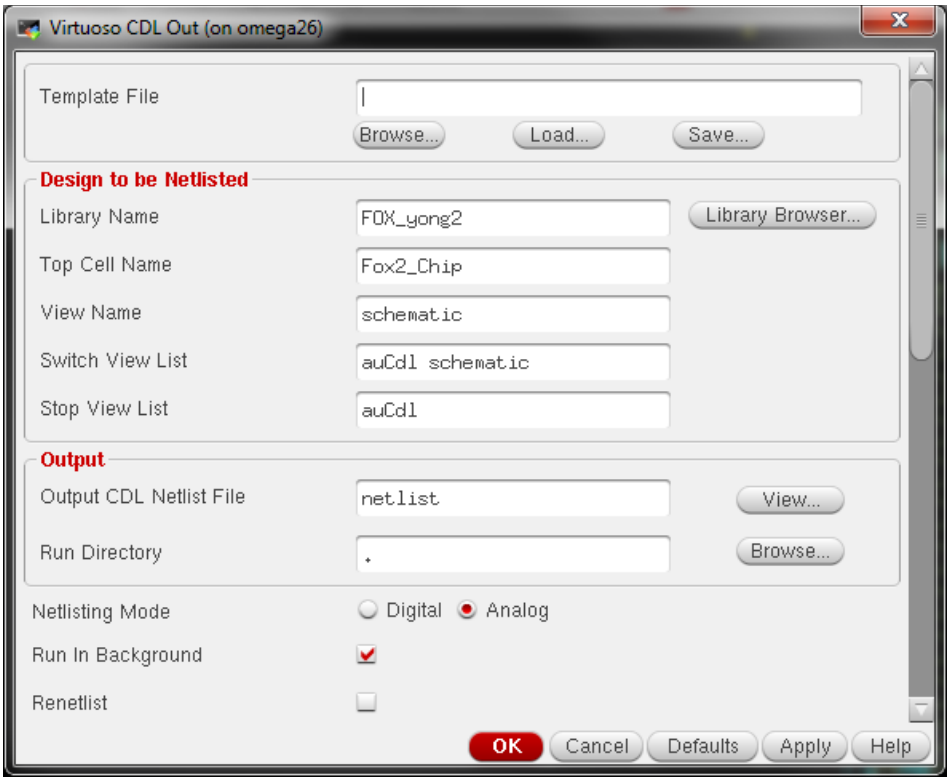
   b. Run the DRC check after dummy, the same method as we did before

9. LVS

   After get the final layout, we need to get the netlist for the LVS check. The method we used here is export the cdl file and run the LVS between the layout and the cdl file

   a. Get the cdl file






Then click → OK/Apply

b. Make necessary changes to the cdl file

Since the digital controller here is the system level verilog file, we cannot use this cdl for the LVS with the layout. in the digital controller process, we had the cdl file for the digital controller only, thus, we can manually edit the top cdl file and replace the definition digital controller in this file with the digital controller from the digital part (v2cdl command from the gate level verilog)

While for the instance of the digital controller, make sure the pin name order is the same as the cdl definition part

c. Run the final LVS, the same technique as we use in the digital controller part