**Student Name: Omar Magdy Shaaban**

**<CESS>**

**<u><Project Phase 3></u>**

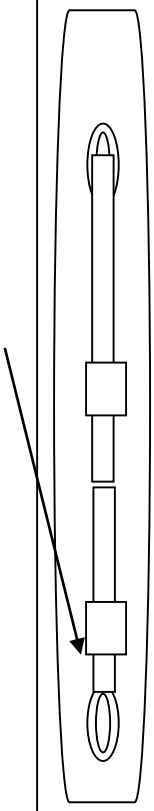
**UEL Module Title: Engineering Systems**

**Module Code: EG7423**

**ASU Course Title: Electronic Design Automation**

**Course Code: CSE215**

Using file fastener



**Date of Submission: 17/5/2019**

**Lecturer: Dr. Mohamed Dessouky**

**UELID: u1927296**

**ASUID: 16P6034**

**Email: 16P6034@eng.asu.edu.eg**

University of East London

# Abstract

This document illustrates the physical implementation of a chip. The sequence of the tools used is: ocp, nero, cougar, lvx, druc and s2r. Two additional tools can be used to view the layout between the steps which are graal (symbolic layout) and dreal (real layout). Screenshots and results are provided to view the output of each step.

# Contents

# Figures

# 1. Floorplan

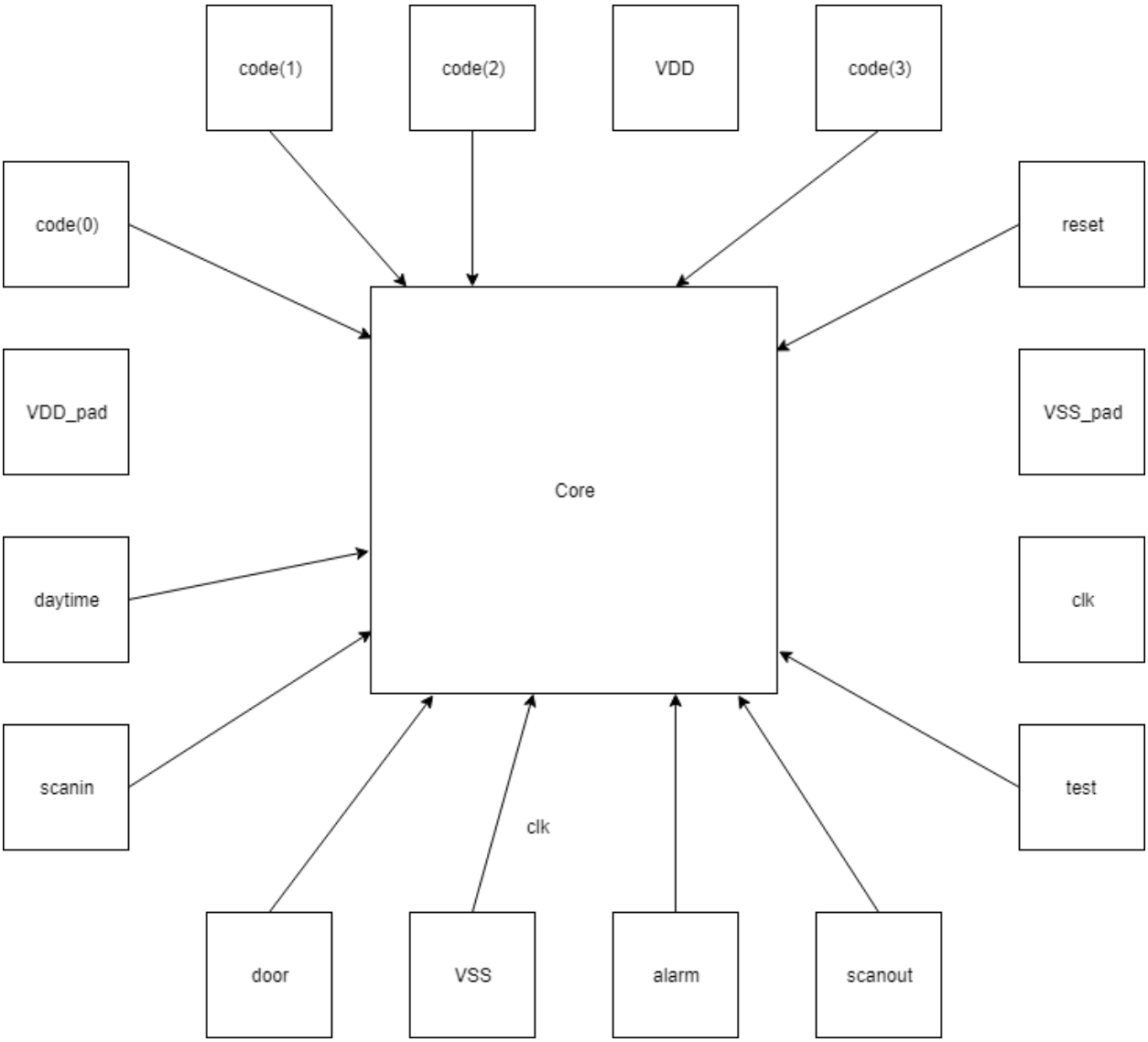



*Figure 1: Floorplanning*

## 2. Placement and Routing Steps

1) Placement:

a) Change the environment variables values (MBK_IN_LO and MBK_OUT_PH) in the Makefile in order to enter a .vst file and produce an output file of .ap file.

b) Execute the ocp command using the .ioc and .vst files.

c) View the .ap file using the graal tool.

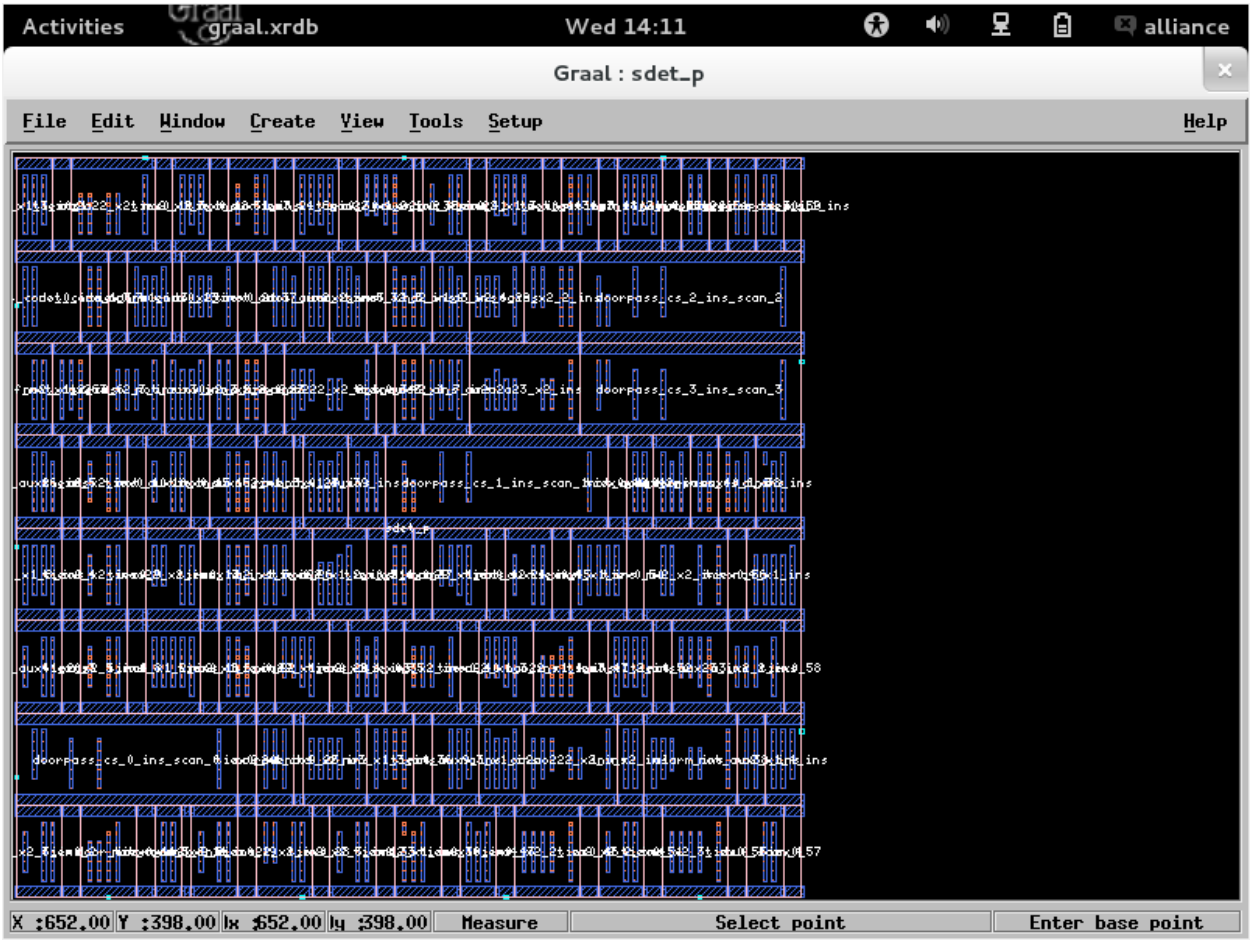



*Figure 2: Placement step output*

2) Routing:

a) Run the nero command using the .vst file and its placement output as inputs.

b) View the .ap file using the graal tool.

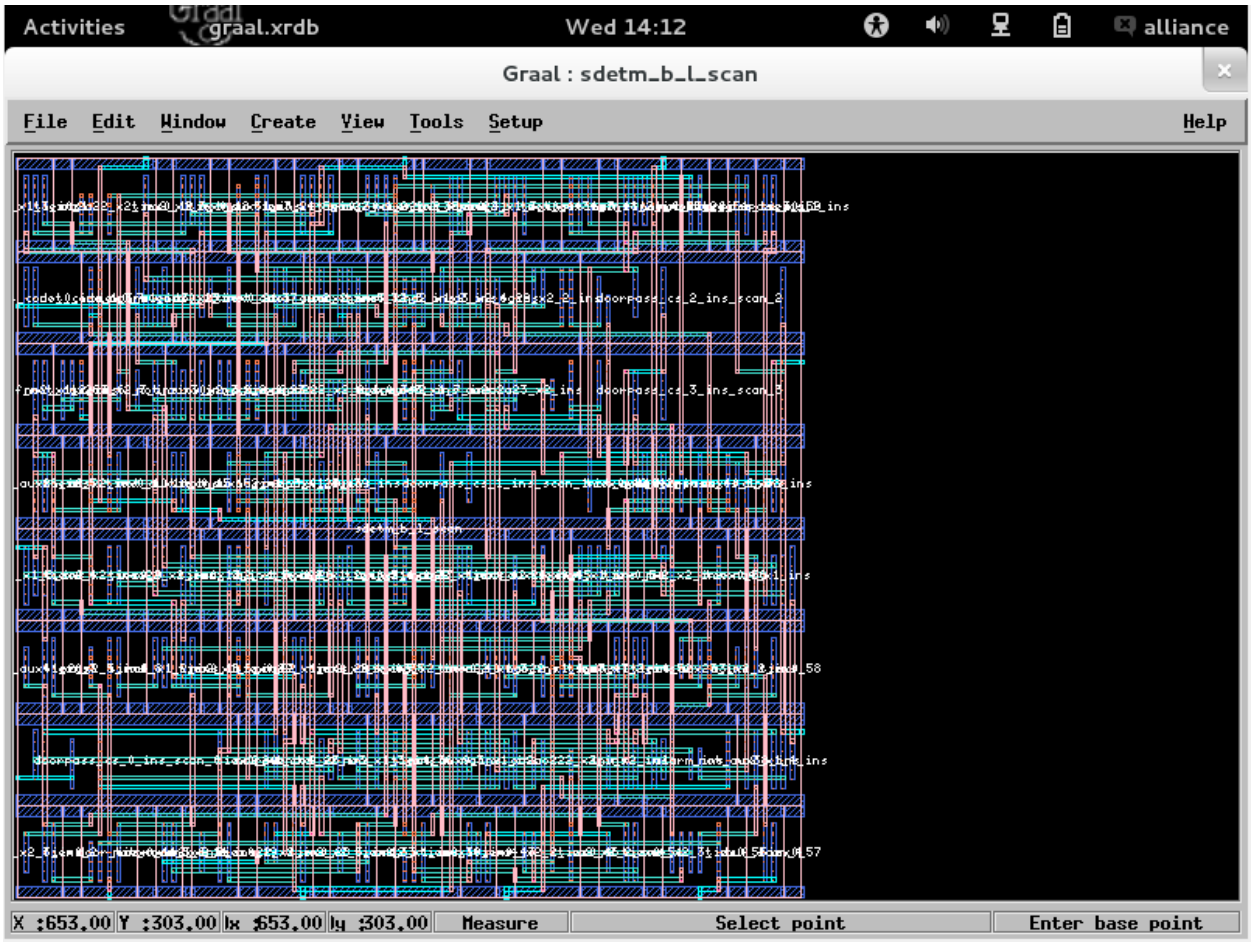



*Figure 3: Routing step output*

# 3. Final Layout

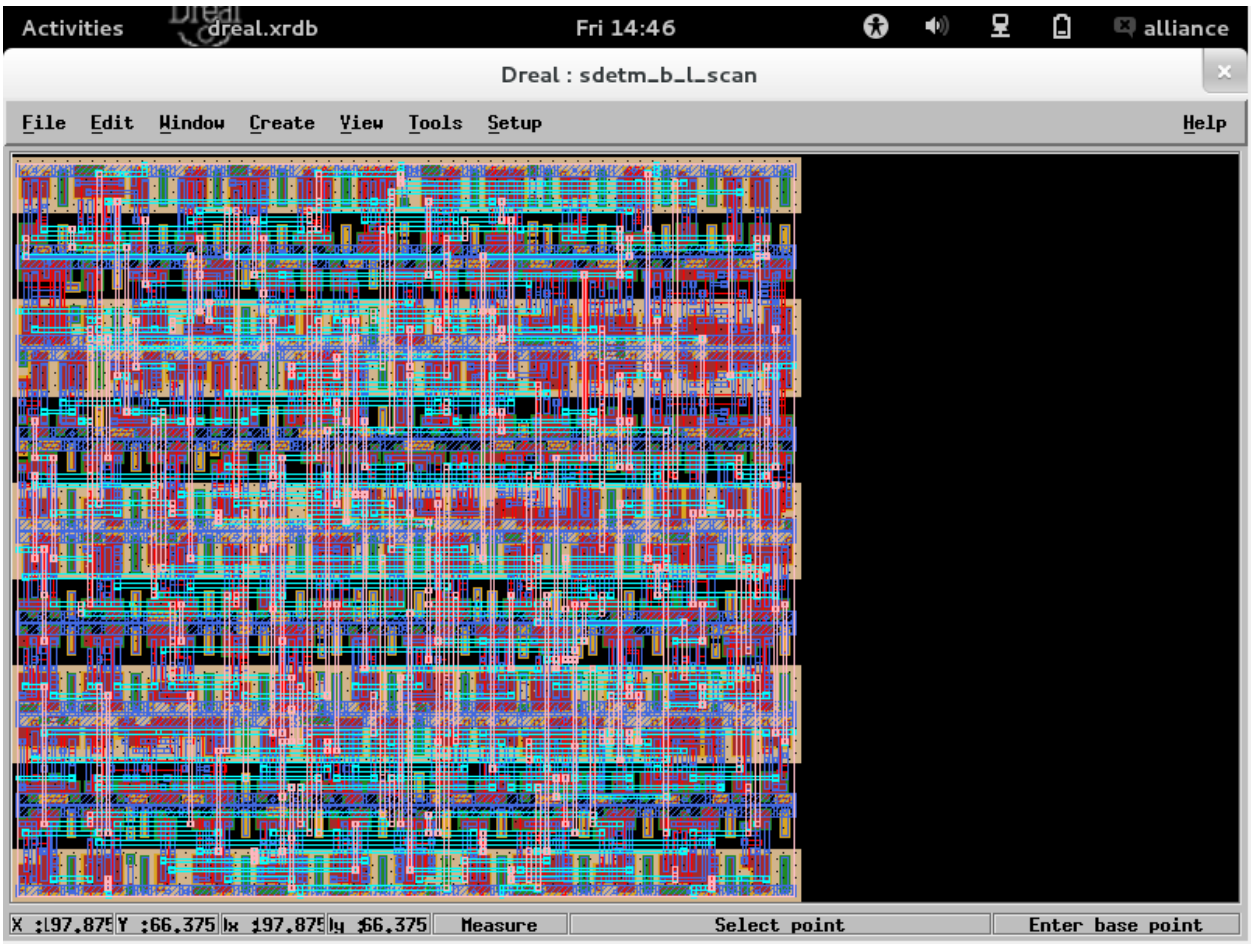



*Figure 4: Real flattened layout using dreal*

## 4. .Out Files Outputs

1) Ocp:

```
 o Final Optimization in process ...
 o Net Cost before Final Optimization... 4812
 o Final Optimization succeeded ...
 o Final Net Cost ..... 3704
 o Final Net Cost Optimization ..... 23.0258%
 o Total Net Optimization .... 48.4589%

Ocp : placement finished
NO PREPLACEMENT GIVEN
 o Destruction of DATABASE ....
```

*Figure 5: Ocp output*

2) Nero:

```
 o  Routing stats :
    - routing iterations    := 211475
    - re-routing iterations := 44102
    - ratio                 := 17.2559%.

 o  Dumping routing grid.
 o  Saving MBK figure "sdetm_b_l_scan".
 o  Saving layout as "sdetm_b_l_scan"...
```

*Figure 6: Nero output*

3) Cougar:

```
          ---> Cut transistors
          <--- 0
          ---> Build equis
          <--- 114
          ---> Delete windows
          ---> Build signals
          <--- 114
          ---> Build instances
          <--- 155
          ---> Build transistors
          <--- 0
          ---> Save netlist

  <--- done !

  ---> Total extracted capacitance
  <--- 0.0pF
```

*Figure 7: Cougar output*

4) Lvx:

```
***** Loading and flattening sdetm_b_l_scan (vst)...

***** Loading and flattening sdetm_b_l_scan (al)...


***** Compare Terminals .............
***** O.K.      (0 sec)

***** Compare Instances ...........
***** O.K.      (0 sec)

***** Compare Connections ..................
***** O.K.      (0 sec)

===== Terminals .......... 14
===== Instances .......... 96
===== Connectors ......... 541


***** Netlists are Identical. *****     (0 sec)
```

*Figure 8: Lvx output*

5) Druc:

```
Create Ring : sdetm_b_l_scan_rng
Merge Errorfiles:

Merge Error Instances:
instructionCourante : 000   0   1   2   3
4   5   6   7   8   9  10  11
12  13  14  15  16  17  18  19
20  21  22  23  24  25  26  27
28  29  30  31  32  33  34  35
36  37  38  39  40  41  42  43
44  45  46  47  48  49  50  51
52  53  54  55  56
End DRC on: sdetm_b_l_scan
Saving the Error file figure
Done
15520

Some errors have been detected, see file: sdetm_b_l_scan.drc for
detailled
```

*Figure 9: Druc output*

6) S2r:

```
 --> post-treating model sdetm_b_l_scan
     ring flattenning  :
     . RDS_NWELL ..................................
     . RDS_NIMP ...................................
     . RDS_PIMP ...................................
     . RDS_ACTIV ..................................
     . RDS_POLY ...................................
     rectangle merging :
     . RDS_NWELL ..................................
     . RDS_NIMP ...................................
     . RDS_PIMP ...................................
     . RDS_ACTIV ..................................
     . RDS_POLY ...................................
     . RDS_ALU1 ...................................
     . RDS_ALU2 ...................................
     . RDS_ALU3 ...................................
 o saving sdetm_b_l_scan.cif
 o memory allocation informations
 --> required rectangles = 3367  really allocated = 7
 --> Number of allocated bytes: 564054
```

*Figure 10: S2r output*

# 5. Appendices

Makefile:

```
sdet_p.ap : sdet.ioc sdetm_b_l_scan.vst
    MBK_IN_LO=vst; export MBK_IN_LO; \
    MBK_OUT_PH=ap; export MBK_OUT_PH; \
    ocp -v -ring -ioc sdet sdetm_b_l_scan sdet_p > ocp.out

sdetm_b_l_scan.ap : sdet_p.ap sdetm_b_l_scan.vst
    nero -V -p sdet_p sdetm_b_l_scan sdetm_b_l_scan > nero.out

%.al : %.ap
    MBK_OUT_LO=al; export MBK_OUT_LO; \
    RDS_TECHNO_NAME=./techno/techno-035.rds; \
    export RDS_TECHNO_NAME; \
    cougar -v $* > cougar_$*.out
    lvx vst al $* $* -f > lvx_$*.out

druc_core : sdetm_b_l_scan.ap
    RDS_TECHNO_NAME=./techno/techno-symb.rds; \
    export RDS_TECHNO_NAME; \
    druc sdetm_b_l_scan > druc_core.out

sdet_chip.cif : sdetm_b_l_scan.ap
    RDS_TECHNO_NAME=./techno/techno-035.rds; \
    export RDS_TECHNO_NAME; \
    RDS_OUT=cif; export RDS_OUT; \
    s2r -v -r sdetm_b_l_scan > s2r.out
```

sdet.ioc:

LEFT ( # IOs from bottom to top

(IOPIN scanin.0 );

(IOPIN daytime.0 );

(IOPIN code(0).0 ); )

TOP ( # IOs from left to right

(IOPIN code(1).0 );

(IOPIN code(2).0 );

(IOPIN code(3).0 ); )

RIGHT( # IOs from bottom to top

(IOPIN test.0 );

(IOPIN reset.0 ); )

BOTTOM ( # IOs from left to right

(IOPIN door.0 );

(IOPIN clk.0 );

(IOPIN alarm.0 );

(IOPIN scanout.0 ); )