CS 516000 FPGA Architecture & CAD

Final Project (Due: Jan 11, 2022)

This design project is completely open in terms of how you solve the problem. (You may propose your own approach or follow previously proposed approaches in the literature.)

You may work individually or in a team of two persons, you will receive an additional bonus if you finish this project on your own.

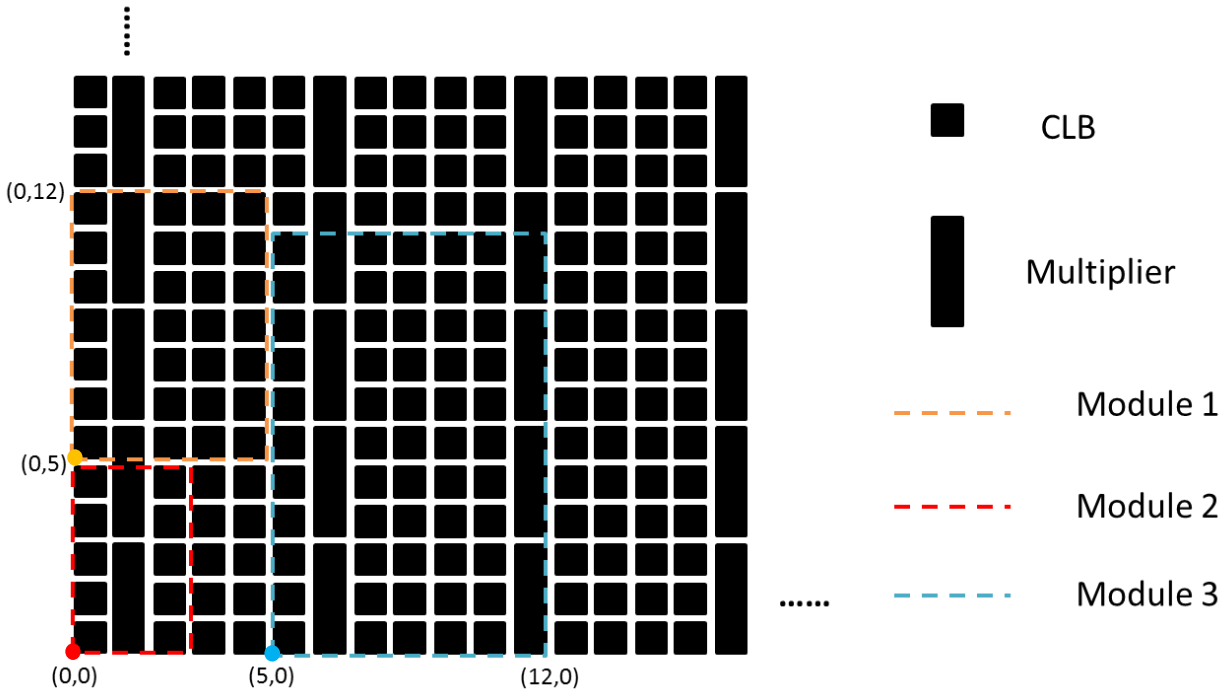
Floorplanning for FPGAs with Heterogeneous Resources

Implement a fixed-outline floorplanner for FPGAs with heterogeneous resources.

For this project, we assume that a FPGA contains two types of logic resources, CLBs and multipliers, and each module is to be placed in a *rectangular region* satisfying its resource requirement within the FPGA. We want to find a feasible floorplan (i.e. all modules must be placed within the given chip area with no overlap).

The objective is to minimize the *total wirelength* of the floorplan, where the wirelength for each net is determined by the half-perimeter wirelength (HPWL) of the minimum bounding box of pins of the net. For simplicity, we assume each pin of Module m_i is located at the center of m_i.

The floorplanner should read three files, an FPGA architecture description file, a Module description file and a Net description file, and generate an output file describing a feasible floorplan in the formats specified below.



Input file:

(i) FPGA architecture description file (.arch file)

The 1st line contains 4 integers: R C S D.

(a) "R", the number of rows in the FPGA (We assume each multiplier is three-row tall, so R will be a multiple of 3.).

(b) "C", the number of columns in the FPGA.

(c) "S", the index of the first multiplier column.

(d) "D", the multiplier columns are columns S, S+D, S+2D, S+3D, … .

(e) row and column index starts from 0.

A sample architecture description file looks like:

48 50 1 5

(ii) Module description file (.module file)

This input file specifies a list of modules. Each line contains 3 integers describing a module.

(a) The 1st integer is a unique module ID.

(b) The 2nd integer is the number of CLBs required by the module.

(c) The 3rd integer is the number of multipliers required by the module.

A sample module description file looks like:

1 23 2

2 9 0

3 55 4

(iii) Net description file (.net file)

This input file specifies a list of nets. Each net statement starts with the ID of the net, the modules that are connected by the net are listed between a pair of braces following the net name.

A sample module description file looks like:

1 { 1 2 }

2 { 2 3 }

3 { 1 3 }

Output file: (.floorplan file)

The output file contains one line for each module. Each line contains five integers id, x, y, w, h. id is the module ID, (x, y) is the coordinate of the module's lower left corner, w is the width and h is the height of the module. The last line of the file is the total wirelength of the floorplan.

A sample output file looks like:

1 0 5 5 7

2 0 0 3 5

3 5 0 7 11

26

Project Submission

Source codes should be uploaded to eLearn. Please include a Makefile for compiling your codes and a Readme file to illustrate how to compile and execute your program.
In addition, upload a report describing the details of your approach. If it is a team work, each member has to explain what he has done and the percentage of his contribution.

Environment and Execution

(1) Language: C/C++
(2) Platform: Linux
(3) The executable file must be named as **project**, please use the following command format to run your program.
    $ project <.arch file> <.module file> <.net file> <.floorplan file>
    E.g.: $ project case1.arch case1.module case1.net case1.floorplan

Required Items & Submission Format

Please compress Project/ (using tar) into one with the name Project_{StudentID}.tar.gz before uploading to eLearn.

➔ E.g.: Project_110062xxx.tar.gz

(1) Project/src/ contains all your source codes, Makefile and README
(2) Project/outputs/ contains all your output files with ".floorplan" extension of all benchmarks
(3) Project/benchmarks/ contains all benchmarks
(4) Project/bin/ contains your compiled executable file
(5) Project/Project_{StudentID}_report.pdf which is your report

Evaluation

- For each benchmark, if your floorplan result violates the aforementioned constraint, the quality score on that benchmark will be **0**.
- If your program takes more than **10** minutes to generate a floorplan result, it fails on that benchmark.
- Any plagiarism will result in a 0 grade for the project.

Grading

- 20%: The completeness of your program and report
- 80%: The solution quality (hidden testcases included)
  - The quality score is based on the *total wirelength* of your floorplan result compared to other students when your solution is valid. Here is the equation for score calculation:

$$100 - 30 \times \min\{\ 1, \left(\frac{\text{Your HPWL}}{\text{The smallest HPWL among all teams}} - 1\right)\ \}$$