**题目1：开源HLS工具——Dynamatic**

1. **背景**

目前基于传统方法的HLS工具，学术界认可的开源工具有限。LegUp工具是十年前学术界认可的工具，但该工具在2014年最后一次开源后就进入了闭源状态。

传统的HLS工具，在调度过程中目前采用的较多的还是静态调度算法，即通过确定控制步，并分配operations到每个固定的控制步中，而后确定状态以及FSM。动态调度的概念是，数据准备好之后，对应的operation就可以执行，不需要等待或者限定在固定的控制步中。目前，欧洲的Lana团队在这方面有了前期的长足进步，而且开源了其基于动态调度的HLS工具。

1. **实验细节、交付内容**
2. 下载其开源代码（github），**在“自己的”Linux服务器（或虚拟机）上安装、执行、可以跑其自带的示例**。并学会查看其QoR结果。
   1. 该工具的网站：<https://dynamatic.epfl.ch/>
   2. 该工具源码下载地址：<https://github.com/EPFL-LAP/dynamatic>
   3. 与该工具有关的顶级会议期刊文章列表：<https://dynamo.ethz.ch/publications/>
3. 工具中的某一关键算法修改

关键算法例如，调度算法scheduling、Buffer优化算法等。

找到“目标算法”，可以根据上面链接提供的相关文章列表，阅读其中某一篇后，修订paper提到的算法。

例如，可考虑以下点算法：（也可以考虑其他）

发表在FPL’22的该团队的论文，主要研究了高性能dataflow电路的缓冲区放置和大小调整。dataflow电路具有一个非传统的特性：buffer可以放置在电路中的任何位置而不改变其语义，这与传统数据路径中发生的情况形成鲜明对比。然而，尽管在功能上无关紧要，但这种放置对电路的时序和吞吐量有重大影响。其IEEE论文下载地址是

<https://ieeexplore.ieee.org/abstract/document/10035122>

其代码的位置是：include/dynamatic/Transforms/BufferPlacement/FPL22Buffers.h

* **读懂代码中的算法策略，用到的数据结构、I/O输入输出格式含义；**
* **直接修改，或替换为自己的算法**
* 重新编译，并执行同一个测试例子，查看并对比其结果。

**最终交付内容：**

1. 实验报告（word）。中文，主体内容包括并不限于：工具的整体代码结构，关键算法点及其目录位置；实验的目标算法，主要策略、数据结构分析；自定义算法的策略，优势分析；实验对比数据。
2. PPT汇报（约15分钟）。

**题目2：开源EDA工具优化（点工具优化）**

1. **布局合法化算法改进**

具体请见鹏城实验室iEDA的主页：

<https://ieda.oscc.cc/train/practice/algorithms/a1.html>

另外，如果有兴趣修改其他的也可以，鹏城实验室有相关的一些EDA关键算法介绍的视频：

<https://ieda.oscc.cc/activities/communication/iEDA-Tutorial.html>

1. **实验细节、交付内容**
2. 下载其开源代码，**在Ubuntu上安装、能够执行**。
   1. 源码下载地址：
      1. Yosys: 实现Verilog代码到AIG的转换（AIG可以理解为门电路）

Yosys: https://github.com/YosysHQ/yosys

* + 1. iMap和ABC可以实现AIG到netlist的转换（这俩都可以选，目前应用最广泛的是Berkely大学开发的ABC工具）

iMap: <https://gitee.com/oscc-project/iMAP>

ABC：<https://people.eecs.berkeley.edu/~alanmi/abc/>

* + 1. 鹏城开发的后端工具，即netlist到版图GDS-II的转换

iEDA: <https://gitee.com/oscc-project/iEDA>

1. 找到目标算法的代码位置，熟悉其输入输出的数据格式，熟悉内部的数据结构。构思并实现自定义的算法，能够调试跑通，并得到对比结果。

**最终交付的“实验报告”：**

1. 目标算法的原理、输入输出格式含义、主要数据结构描述
2. 已有算法的执行策略，局限性，拟改进的方法
3. 自定义算法的策略。
4. 实验结果对比。
5. PPT汇报