



---

## 2019 年 SEU-Xilinx 国际暑期学校团队项目设计文档

(Project Paper Submission Template)

作品名称	卷积神经网络加速器
组员姓名	秦树洋 纪云鹏 薛臻
房间号及桌号	720 室 36 组



## 第一部分

### 小组成员分工

(各成员任务分配)

	姓名	任务分配
组长	秦树洋	Cnn ip
组员 1	纪云鹏	Block design
组员 2	薛臻	Hls ip 加速设计
组员 3		

## 第二部分

### 设计概述 /Design Introduction

(请简要描述一下你的设计：1. 功能说明；2. 所用到的设备清单)

1. 手写体识别分类加速：输入一张图片，输出分类的类型正确率
2. 设备：pynq-z2

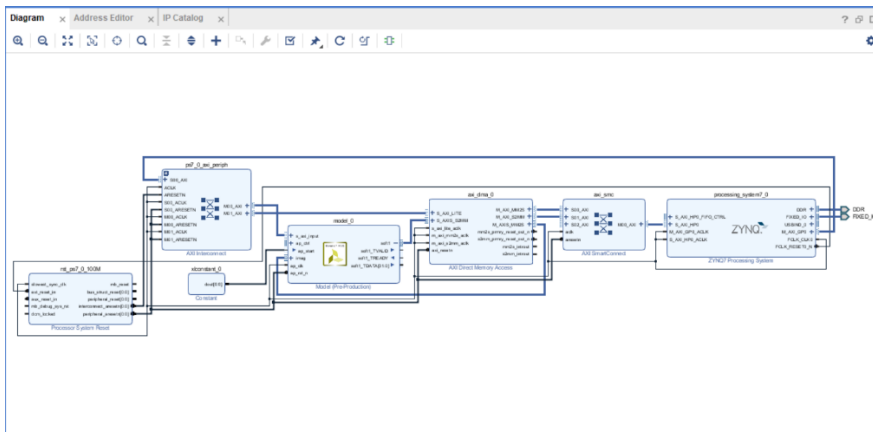
## 第三部分

### 详细设计 /Detailed Design

(请详细说明你作品要实现的所有功能以及如何组建系统以实现该功能，还包括为实现该功能需要用到的所有参数和所有操作的详细说明，必要的地方多用图表形式表述)

如因文档篇幅限制无法详述，可提供附件。

1. cnn ip: lenet 网络的 c++实现 (卷积层, 激活层, 全连接层, 池化层, soft 分类层)。  
详情: 原计划将 vgg16 网络使用 fft 算法实现成 ip, 但复现过 vgg16 网络后, 发现片上资源不够, 在 hls 编译器上也不能定义过大的数组。导致浪费了很多时间, 转而复现简单的 lenet 网络。
2. axi 总线设计: 输入, 输出采用 stream 接口, 权重的接口采用 lite 接口。
3. block design: 通过 zynq 控制 dma 对 hls 生成的 ip 进行数据流的读写。Dma 与 ddr 进行输入, 输出的数据交换。通过 axi\_lite 使 ip 与 ddr 进行权重的输入。输出正确率用串口打印。



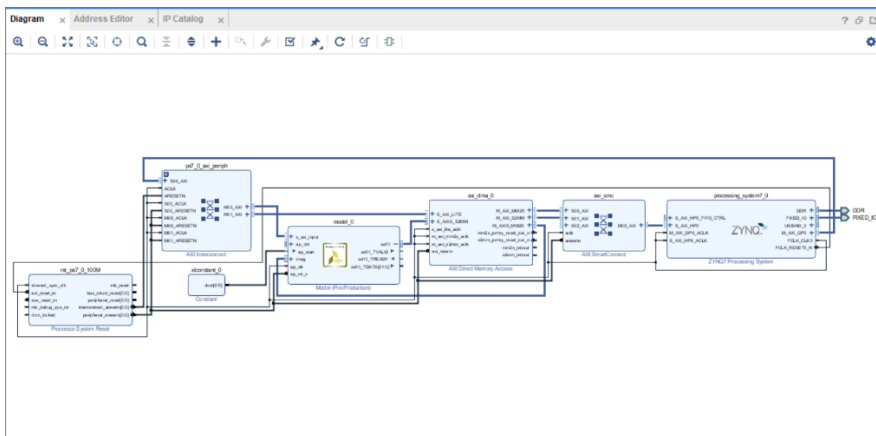


## 第四部分

### 完成情况 & 性能参数 /Final Design & Performance Parameters

(作品完成情况, 附照片/视频链接)

- 完成 ip 模块 c++实现
- hls 编译制导语句的添加引导编译器进行优化设计
- 通过 hls 生成所需 ip
- 将 ip 链接为系统
- 选择的神经网络模型过大, 实现一层的实现的 ip 并完成系统连接
- 学习 SDK 调用模块 (后续工作)



## 第五部分

### 项目总结 /Conclusions

(项目中所用到的知识点, 项目收获及心得)

- 在 hls 中不应定义过大的数组, 编译器会出现问题。
- 通过对源代码添加的合适编译制导语句可以指导编译器生成最优所需 ip。
- 生成 ip 核时的报告可以反映出很多问题, 通过分析该报告可以返回去改进算法的设计。
- AXI-Lite 总线和 AXI-Stream 总线的区别, AXI-Lite 总线可以多个输入合并为一个接口。
- 搭建系统时要充分了解各模块间的关系, 通过 zynq 的控制是整个系统可以运行起来, 特别注意每个 ip 的设置和需要的端口。
- SDK 中地址的调用在其输出的头函数中可以直接找到可调用的函数。

## 第六部分

### 源代码/Source code

(作品源码 Github 链接, github 操作步骤见 *Github 项目上传简单教程*)



---

## 第七部分（技术总结/心得/笔记等分享）

（可分享的技术文档附件或已发表的文章链接）

[https://blog.csdn.net/qq\\_41763108/article/details/96285226](https://blog.csdn.net/qq_41763108/article/details/96285226)

[https://blog.csdn.net/qq\\_41763108/article/details/96426670](https://blog.csdn.net/qq_41763108/article/details/96426670)