**2020年新工科联盟-Xilinx暑期学校团队项目设计文档**

**设计文稿提交格式**

**(Project Paper Submission Template)**

|  |  |
| --- | --- |
| **作品名称** | 数字特征识别 |
| **板卡型号** | *xc7s15ftgb196-1* |
| **所在班级** | Xilinx暑期学校A班（西南交大通信2018-01班） |
| **成员姓名、学号、学校** | 金佳宁 2018112783 西南交通大学 信息科学技术学院  王文奕 2018112752 西南交通大学 信息科学技术学院 |
| **Github链接** | https://github.com/chatoyantking/project\_JW.git |

**第一部分**

设计概述 /Design Introduction

（1.请概括地描述一下你的设计，可包括本设计目的、学习到的知识点、应用方向或者设想的应用场景等；2. 经组内成员讨论后以表格的形式描述项目中各成员在项目中发挥的作用或者贡献百分比；3.作品的展示照片）

1.设计思想阐述

图像每时每刻都在我们的身边，是我们日常生活和工作领域不可缺少的一部分。不仅实在军事科技领域有图像识别的需求，就连我们生活中随处可见的人脸识别也是一项基于图像识别的技术。在普通的图像研究及应用过程中，一幅图像中的特定一部分包含了图像所要传达的大部分信息，这些信息包括数字或者特定的字符，所以字符及数字识别的重要性也变得越来越高。字符与数字识别的应用领域也十分广泛：可以用在车牌号识别、仪表数字以及点钞机钞票面额识别等诸多领域。相对应的，由于人们的需求越来越高，所以也有越来越多有关于数字字符识别的方法，逐渐提高识别的准确率和效率。可见数字识别的重要性。

FPGA可以加快处理速度，其中的内部静态存储SRAM可以储存一些图像信息，另外它还可以自动完成电路板的布局，有效节约手动布线时间，还可以连接外设，如：OV5647摄像头和显示器实现图像的获取与结果输出。这使得FPGA在图像处理方面有很大的优势，可对图像进行高效的灰度化、去噪处理、二值化等图像加工，明显减少图像数据、简化运算，并且更容易获取目标显示区域的几何特征，提高运算速率以及节省资源。所以，我们计划实现一个基于FPGA的数字识别系统。

近年来关于数字识别的算法有很多，有基于BP神经网络和粒子群等人工智能的数字识别算法，由于我们对于这些算法并不熟悉，且这些算法在硬件电路实现较为困难，所以我们采用数字特征识别算法。通过对图像分割提取特征点，实现不同数字的区分。

本课题在现有，拟用OV5647摄像头获取实际的图像信息，识别图像中的印刷体数字，在显示器上显示实时的图像信息，并在屏幕上显示对应的数字来展示识别结果。

本课题的内容主要包含以下方面：

1）完成印刷数字识别系统的设计方案，确定具体的数字识别算法以及硬件电路实现；

2）熟悉FPGA开发板，摄像头OV5647以及显示器的结构、各功能配置，以数字图像处理的方法及原理，图像显示原理；

3）数字识别的过程包括：图像获取，图像灰度化，图像二值化，图像边缘检测，字符分割、提取并统计特征值、模板匹配以及结果输出；

4）各模块功能集成，对系统进行综合、实现并下载到开发板验证数字识别的结果。

2.成员任务完成情况

|  |  |
| --- | --- |
| 小组成员 | 在项目中发挥的作用或者贡献 |
| 金佳宁 | 1）了解显示器图像输出的原理，结合项目指导书，编写彩条输出的代码，验证显示器能否使用。  2）了解图像处理的方法，结合实验指导书实现图像的灰度化，二值化以及图像的边缘检测。  3）调试摄像头成功驱动摄像头并实现实时图像输出。  4）编写数字特征识别的代码，虽然没有调试成功。在静态数字图像输出的基础上，尝试对静态数字进行特征识别，未成功。在期间编写了检测序列上升沿个数的代码（计划用于横轴和纵轴的交点个数提取），和检测序列对应数值输出的代码（计划用于提取横轴交点的坐标位置，与图像中轴的数值进行对比，区分交点的左右位置），之后询问老师后用异或门检测依旧无法准确识别出数字。 |
| 王文奕 | 1. 结合项目指导书，实现HDMI输出和MIPI 摄像头视频输入，使开发板和外设（摄像头、显示器）能够成功连接，为静态/动态图像输出做准备。 2. 查阅边缘检测原理及图像处理方法，结合项目指导书，实现动态图像预处理（灰度化、去噪处理、边缘检测、二值化）。 3. 编写字符分割代码，用水平垂直投影法，对图像进行X、Y方向投影，统计垂直于轴线的图像上的像素的黑点数量，由此判定图像的切割位置，用这个坐标来切割原图像，得到目标图像，并显示数字上下边界的分割边框，目的是将多位数字分为独立的单个字符图像，以实现多位数字的识别。（代码编写完成，但显示器上始终不能显示数字分割边框） 4. 编写单个数字特征动态识别代码、结果显示模块代码。（代码编写完成，但功能未实现） |
| 由于项目难度较大、两位成员此前均无FPGA基础以及疫情关系无法面对面讨论，在项目初期，团队成员同时进行相同任务，以保证两位成员熟悉FPGA、开发板外设、VIVADO平台的使用，并对项目有更深入的理解。在项目后期，成员分工完成任务，并及时交流进度、讨论设计难题，共同协作完成项目的设计。 | |

3.作品展示照片

**动态图像处理部分：**

|  |  |
| --- | --- |
| 功能 | 图片 |
| 摄像头视频输入 | 注：实时图像输出，摄像头实现输入到显示器输出调试正常 |
| 动态图像边缘检测结果 | 注：认识边缘检测的原理  C:\Users\mac\Documents\Tencent Files\864890070\FileRecv\MobileFile\IMG_7911.JPG |
| 动态图像灰度化且二值化处理结果 | 注：摄像头对准电脑屏幕上的数字，因此输出图形有波纹  C:\Users\mac\Desktop\拟提交文件\动态图像处理\动态边缘检测1.jpg |
| 由于总工程没有输出成功，所以此处没有总工程的效果图。 | |

**静态图像部分：**

|  |  |
| --- | --- |
| 功能 | 图片 |
| 彩条输出 | 注：认识图像输出原理  C:\Users\mac\Desktop\拟提交文件\静态图像处理\彩条显示\彩条显示.jpg |
| 静态图像灰度化 | C:\Users\mac\Documents\Tencent Files\864890070\FileRecv\MobileFile\IMG_7917.JPG |
| 静态图像边缘检测 | C:\Users\mac\Documents\Tencent Files\864890070\FileRecv\MobileFile\IMG_7914.JPG |
| 综合程序输出结果 | 注：绿色线条为输出结果，但结果不正确  C:\Users\mac\Documents\Tencent Files\864890070\FileRecv\MobileFile\IMG_7925.JPG |

**第二部分**

系统组成及功能说明 /System Construction & Function Description

（请对作品的1. 计划实现及已实现的功能；2. 项目系统框图；3. 使用的技术方向做说明）

总述：

作品预搭建的数字识别系统使用MIPI摄像头OV5647采集图像信息，通过FPGA实现图像预处理、数字信息识别和显示。该数字识别系统由四个基本模块组成:图像采集模块、预处理模块、数字特征提取及识别模块和HDMI显示模块。

作品计划实现：

1. 彩条显示等多种静态图案显示；
2. 摄像头视频输入及HDMI输出
3. 静态/动态图像的边缘检测；
4. 静态图像的数字识别；
5. 多位数字的动态识别功能，并使在显示器上显示结果数列。

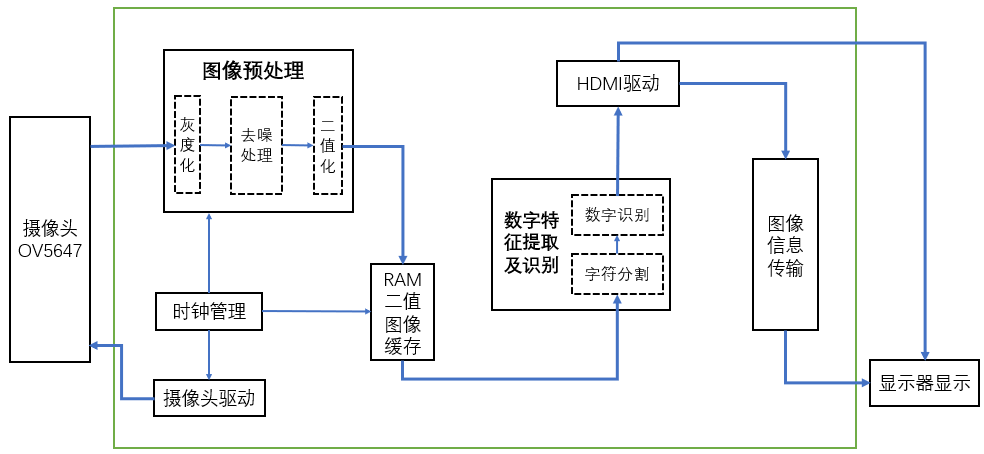


图1.项目系统框图

1. 图像采集模块

本模块利用已有IP核以及Verilog编程语言，通过例化IIC模块与摄像头进行初始化，实现摄像头图像读取与输出。模块设计时，通过查阅OV5647数据手册，设置摄像头输出图像的配置及定时参数。根据摄像头内部时钟的控制，通过改变寄存器的值，来设置摄像头内部PLL时钟分频，进而调节输出图像的频率。

1. 图像预处理模块

图像预处理模块的整体设计框图如图所示，分别为灰度化模块、去噪处理模块、二值化模块、数据存储模块。

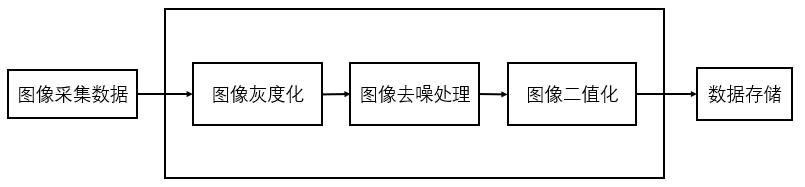


图2.图像预处理模块设计框图

2.1图像灰度化

图像灰度化模块是先将摄像头采集到的彩色图像数据转化为灰度图像数据。彩色图像是指每个像素由R、G、B分量构成的图像，其中R、G、B是由不同的灰度级来描述的。在处理这种图像格式时，因其每个像素都有三个分量且代表意义不同，故不能统一处理，而要分开来处理，但这会浪费大量的时间以及资源。因而，可以采取降维的方法来对其灰度化处理，这样可以提高系统运行效率以及节省硬件资源。当彩色图像三种分量的值相等时，图像将变成灰色。

彩色图像中R、G、B三分量分别代表红色、绿色、蓝色的亮度值，而人眼对这三种颜色的敏感度是不同的，其中绿色的敏感度最高，红色次之，蓝色的敏感度最低。因此，对这三个分量要以不同的权值进行加权平均得到较合理的灰度图像，公式如下：

，

但对于FPGA来说，这样的浮点运算耗费资源过多，需要在此公式基础上进行算法优化,即去浮点运算。

2.2图像去噪处理

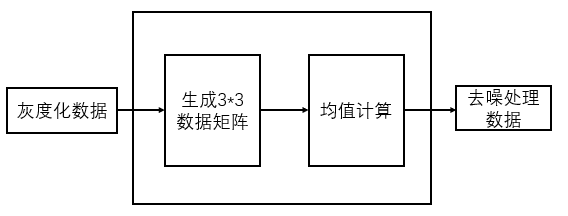


图3.图像去噪模块设计框图

图像去噪模块是对图像数据进行去噪处理，减少图像噪声对后期识别结果的影响。因为图像采集与处理过程中会产生噪声，降低图像的质量，对后期的处理结果会产生影响，所以要进行图像去噪处理。

本设计拟利用均值滤波法对图像数据展开图像去噪，该方式为线性滤波算法。先对图像中的像素构建模板，而后对模板中的像素展开处理，即求平均值，最后将得到的平均值赋给原来的像素值。即对表3.2.1中的P11、P12、P13、P21、P23、P31、P32、P33像素求平均值，用求得的平均值替代P22的像素值。

|  |  |  |
| --- | --- | --- |
| P11 | P12 | P13 |
| P21 | **P22** | P23 |
| P31 | P32 | P33 |

表2.2图像去噪模块设计框图

除此之外，本项目还尝试采用Sobel算法，实现像素图像边缘检测。在技术上，sobel算子是一个离散的一阶差分算子，用来计算图像亮度函数的一阶梯度之近似值。在图像的任何一点使用此算子，将会产生该点对应的梯度矢量或是其法矢量。Soble边缘检测通常带有方向性，可以只检测竖直边缘或垂直边缘，或者都检测。算法如下。

X方向Sobel算子为，；Y方向Sobel算子为，。

原始图像P为，。

公式为，,(符号‘\*’为卷积计算符号)。

2.3图像二值化

对摄像头所采集到的图像信息去噪声后，还要经过图像二值化处理，将滤波后的数据变成二值化数据。经过这样的处理，图像显示结果将会出现泾渭分明的黑白色彩，常常用于图像的分割、图像的识别等领域。此模块在设定一个阈值T后，将所有采集到的像素点的灰度值与T相比较，便可以将所有的像素点划分为两部分，这样便能成功的将有效信息从无效信息中提取出来。经处理后的图像数据明显得到减少、处理起来更为简单，可以更方便的进行运算，并且更容易获取目标显示区域的几何特征，可以提高运算速率以及节省资源，并且最大限度的将有效图像信息保存起来，便于后期进行处理。

以上过程2.1-2.3不是实时的图像处理，会相对于原始数据延迟几个时钟，因此需要另外设计一个延时器来优化图像预处理的功能。

2.4数据存储

对图像的灰度化、去噪处理、二值化实现后，需要对数据进行存储，以降低对存储资源的需求。

为提高存储数据的效率，使用简化双端口RAM完成存储功能。存储原理是，将一块大小为16 bit×2 048大小的Block RAM分成上下两部分存储空间，每一部分空间恰好能够存储一帧完整图像数据。通过控制写RAM的时序使相邻两帧图像存储于相邻的存储空间，读取RAM时根据控制信号来读取对应区域的图像信息。在本设计中，将使用IP核Block Memory Generator进行数据存储。

1. 数字特征提取及识别模块

本模块采用了基于投影特征的字符分割和基于统计特征的字符识别的方法。

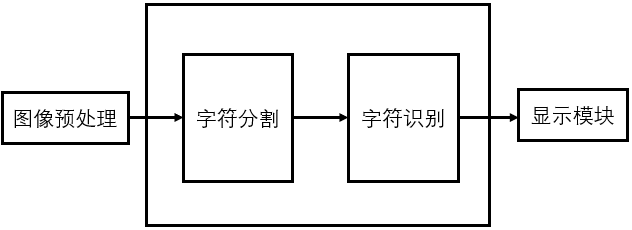


图4. 数字特征提取及识别模块结构框图

3.1字符分割

当采集到的图像中同时存在多位数字时，使用字符分割算法处理，可识别数字的上下边界，将其分为独立的单个字符图像。

本系统选定用水平垂直投影法实现字符分割处理，并将分割后的单个数字字符的边界信息输送给字符识别模块。其原理是，对二值化图像进行投影，在选定方向取一条直线，统计垂直于该直线的图像上的像素的黑点数量，累加求和作为该轴该位置的值；基于图像投影的切割就是将图像映射成这种特征后，基于这种特征判定图像的切割坐标，用这个坐标来切割原图像，得到目标图像。

下图是利用Matlab内置函数获得图像的水平垂直投影及其还原图像。当然，在本项目中，将利用Verilog语言进行算法编写。

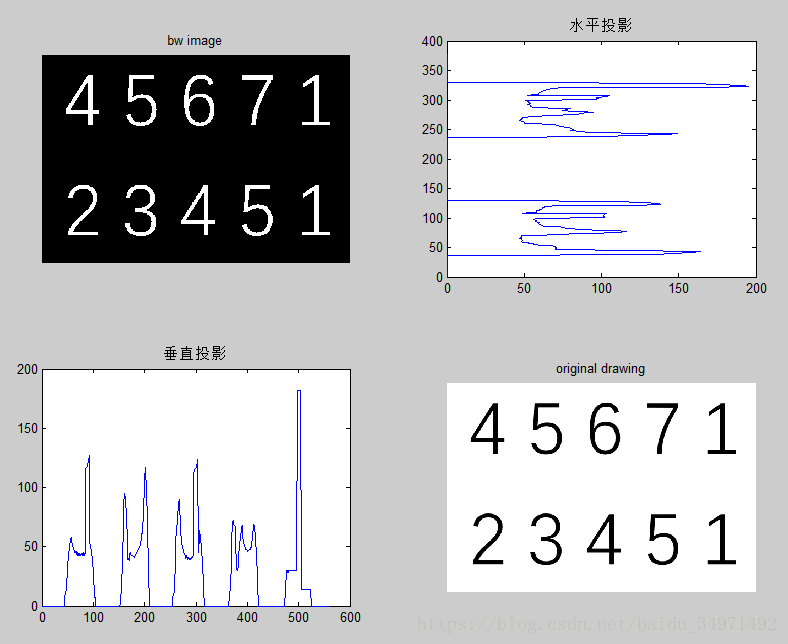


图5.Matlab实现图像的水平垂直投影（图源网络）

3.2字符识别

本项目拟利用基于统计特征的字符识别的方法进行数字识别。假设每个字符存在于一个边框里，边框可以随着字符大小自动调节设置出3 条特征线（X1,X2,Y），然后统计通过线上像素点的变化次数来提取每个字符的特征值。在分割后的二值化图像中，分别从字符横向2/5 和2/3 处以及字符纵向 1/2 处做扫描线，统计扫描线上字符变化的次数，得到3个穿越次数特征，对其进行分类，在此基础上对字符进行识别。以数字7的识别为例，统计特征示意图如图6所示，将统计图中横线 X1、横线 X2、竖线 Y 中像素值的变化次数作为每个数字的特征，再将每个数字的特征进行分类即可对数字0-9进行识别。

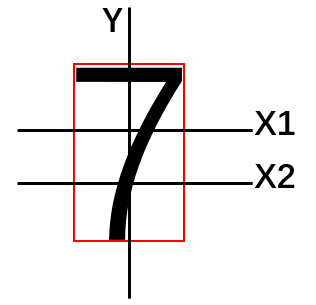


图6. 单个字符的特征参数统计

通过统计数据得到每个数字对应不同扫描线上像素变化的分类情况，如表3.2.1所示。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数字 | X1像素变化次数 | X2像素变化次数 | Y像素变化次数 | 分类 |
| 0 | 4 | 4 | 4 | 1类 |
| 1 | 2 | 2 | 2 | 2类 |
| 2 | 2 | 2 | 6 | 3类 |
| 3 | 2 | 2 | 6 | 3类 |
| 4 | 4 | 2 | 4 | 4类 |
| 5 | 2 | 2 | 6 | 3类 |
| 6 | 2 | 4 | 6 | 5类 |
| 7 | 2 | 2 | 4 | 6类 |
| 8 | 4 | 4 | 6 | 7类 |
| 9 | 4 | 2 | 6 | 8类 |

表3.2.1数字的扫描线特征分类

从表1 可以看出，根据X1，X2，Y的像素变化次数，可以将数字分为8 类，只有第3类的字符不能根据此方法直接识别出来。对于此类字符，可以再根据第1个发生变化的像素所在列与Y的位置关系来判断。位置关系如表3.2.2所示，可以根据位置关系的不同，将2、3、5也分别识别出来。

|  |  |  |  |
| --- | --- | --- | --- |
| 数字 | X1变化与Y的位置变化关系 | X2变化与Y的位置变化关系 | 分类 |
| 2 | 右边 | 左边 | A类 |
| 3 | 右边 | 右边 | B类 |
| 5 | 左边 | 右边 | C类 |

表3.2.2数字2， 3， 5 的扫描线特征分类

1. HDMI显示模块

本模块主要利用IP 核 rgb2dvi\_0实现HDMI驱动，通过mini HDMI转VGA连接线传输信号至显示器，显示器上将显示预处理后的动态图像、字符分割框、特征提取线以及所识别的数列结果。

此外，在静态图像显示时，还需编写HDMI驱动代码以及图像产生模块，使HDMI输出目标图像。考虑到显示屏清晰度为720p（1280\*720@60Hz），需要的分辨率参数化信息如表2.1所示，本项目利用了编写彩条显示代码及添加COE图片数据文件的方式，进行了HDMI显示测试。

|  |  |  |
| --- | --- | --- |
| Name | Digital | Note |
| H\_ACTIVE | 1280 | Line effective length (number of pixel clock cycles) |
| H\_FP | 110 | Line sync front shoulder length |
| H\_SYNC | 40 | Line sync length |
| H\_BP | 220 | Line sync back shoulder length |
| V\_ACTIVE | 720 | Field effective length (number of rows) |
| V\_FP | 5 | Field sync front shoulder length |
| V\_SYNC | 5 | Field sync length |
| V\_BP | 20 | Field sync back shoulder length |
| H\_TOTAL | 1650 | Total length of line |
| V\_TOTAL | 750 | Total length of field |

表4.1 720P参数化信息

**第三部分**

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

（作品已实现的功能及性能指标）

关于本课题，我们完成了摄像头输入和HDMI显示器图像输出的调试，可以将摄像头采集到的图像实时输出在显示器上。关于图像处理的部分，我们实现了图像的灰度化和二值化处理，另外，我们还尝试了边缘检测。关于数字特征提取的部分，我们首先尝试了对二值化后的图像数据进行上升沿检测，但一直不成功，请教老师之后我们尝试用异或门检测前后是否发生变化。除此之外，我们还编写了信号变换时的位置输出的代码，但各部分代码综合后依旧不能成功识别。现在我们依旧在研究数字特征识别的代码，尝试先实现静态数字图像数字的正确识别。

表5 完成情况及性能参数表

|  |  |
| --- | --- |
| 作品已实现功能 | 性能指标 |
| 彩条显示等多种静态图案显示 | 输出画面分辨率：720P  图像刷新频率：74.25MHz  传输端口：HDMI连接 |
| 摄像头视频输入及HDMI输出 |
| 静态/动态图像的灰度化、二值化处理 |
| 静态/动态图像的边缘检测 |
| 图像的数字识别及结果显示（尚未完全实现） |

**第四部分**

总结 /Conclusions

（谈一谈完成暑期学校课程后的收获与感想。请每位组员分开写。）

**金佳宁：**

以前从来都没想过暑期的短学期课程会以这样的方式进行，虽然今年的方式比较特殊，但是在这一次的暑期学校中，我还是收获颇丰。在这次课程之前，我对FPGA的认识比较浅，并不能很好的认识到其功能。在暑期当中，我们首先先学习了一些有关于数字电路与逻辑设计的知识，从基础的逻辑设计入手，学习相关知识，为后续的课程设计打好基础。在本次课程中学习了一个以前从没有接触过的软件VIVADO，学习了其使用方法。在本课程中较之前的数电课程最大的区别是，它讲了很多关于逻辑设计模块的知识，包括：IP核的使用核封装，RAM接口控制器等。

在暑期过程中最大的收获就是在我们的课程设计之中。本次的课程设计对于我来说很有难度，原因有：1）以前从来没有使用过外设，。比如摄像头以及显示器，都是第一次接触到，所以在调试的过程中很有难度；2）关于数字图像处理也是一个很大的挑战，由于对图像的存储方式，以及显示器图像显示的原理并不熟悉，在后续的编程过程中有很大的难度；3）本课题涉及有关时序的问题也比较多，对于我来说也有困难。虽然有以上种种困难，我在解决问题的过程中也学到了很多，首先是对图像显示有了一定的认识，图像的显示是通过行扫描和列扫描实现的，逐行扫描从屏幕的左上角开始，从上到下，从左到右进行扫描，当扫描完所有的行，会形成一帧，然后扫描回到屏幕左上角，通过行同步信号和场同步信号，确保图像能正确显示。除此之外对图像的处理也有了更深刻的认识，学习了如何将RGB图像转换成灰度图，将图像进行二值化处理以及边缘检测。在项目程序编写的过程中要注意时序的问题，只有时序正确，行列扫描与图像值匹配，才能正确输出图像。在本次实习中除了对VIVADO的使用有了一定的了解，我还对matlab的使用有了更多的了解，在处理图像的过程中，我曾尝试在matlab中实现图像的灰度化和二值化，也是通过matlab对图像的存储方式有了更加深刻的理解。

在本次的课程中除了理论知识的学习外，还学习了资料的搜集，第一次使用了Github，从上面找自己想要的资料以及创建自己的Github并实现文件按的上传。在完成课程设计的过程中，虽然遇到了很多问题，但也要继续学下去。另外小组成员之间的合作也十分重要。虽然，我们没有完成课程设计，但是，还是学习了很多新的知识，锻炼了自己的能力。对于FPGA的学习也不能止步于此，在课程中我认识到了FPGA的强大功能，在之后还要继续学习，尝试用其做一些有意思的设计。

**王文奕：**

这次暑期学校课程利用假期的空闲时间，采用不同于平时学校课程的上课模式，给我带来了前所未有的暑期新体验。在这两周里，网课和实验的结合让我对FPGA有了初步认识，项目设计的过程加深了我对基于FPGA系统设计的理解，学习到了很多新算法。

起初，我难以理解FPGA到底是什么，甚至总是把它的概念和单片机混淆。通过几次课程的学习以及课后查阅资料，我对FPGA有了新的认识。

FPGA采用了逻辑单元阵列LCA这样一个概念，内部包括可配置逻辑模块、输入输出模块IOB和内部连线三个部分。FPGA是在PAL、GAL等可编程器件的基础上进一步发展的产物。它是作为专用集成电路（ASIC）领域中的一种半定制电路而出现的，既解决了定制电路的不足，又克服了原有可编程器件门电路数有限的缺点。FPGA设计不是简单的芯片研究，主要是利用 FPGA 的模式进行其他行业产品的设计，在通信行业等各个领域有着广泛的应用。

光看FPGA的概念可能还是不够形象，因此我通过网络查看了许多网友对FPGA的讨论，了解到，可以将FPGA类比为一个空白的大脑。在这个空白的大脑中已经有了完善的神经元细胞，但这些神经元之间没有相互连接关系，所以这个大脑无法思考，无法对外界信息作出反应。但是，当我们用硬件描述性语言编写代码，对FPGA进行布局布线后，就让这个空白大脑中的一些神经元进行了相互连接，使大脑具备功能，比如控制双手（FPGA的外设）做鼓掌动作等。而单片机更接近与人类大脑，已经具备完成各种任务的能力。因此，FPGA有很强的可塑性，通过改变FPGA的bitstream，可以实现各种不同功能，而单片机功能很难根据自己的需求进行更改。

通过课程的学习以及实验与习题练习，我学会了在FPGA上实现数电课程中的各种逻辑电路、复杂IP的调用和一些系统的设计。除了开发板上自带的器件，FPGA上还有很多接口，通过这些接口连接不同的设备，就可以达到扩展功能的目的。利用这些，我们进行了项目的设计与开发。

我们组的选题是基于FPGA的数字识别。通过MIPI摄像头采集图像信息，并进行图像处理，结合数字特征统计的方法对图像中的数字进行识别。项目的结构比较复杂，算法偏难，因此我们没有成功实现数字识别的最终功能，仅仅完成了对图像的处理，这是我们在暑期学校的一大遗憾。在暑期学校结束之后，我们会继续研究系统的代码编写和FPGA的使用，努力完成数字识别系统。

虽然项目的最终目标还没有完成，但我在开发过程中学到了很多知识，特别是图像处理方面的知识。在这之前，我曾在Matlab上编写基于DCT变换的图像压缩处理方法，打开了我对图像处理认识的大门。这次，在VIVADO开发平台上，利用FPGA等硬件设备，尝试了灰度化、去噪处理、边缘检测、二值化等各种图像处理方法，了解了sobel算子、水平垂直投影的字符分割算法、基于数字特征的识别算法等各种有很高实用性的编程知识。这让我对通信工程这个专业的学习有了一定的知识储备和十分的信心。

最后，感谢这次Xilinx暑期学校的学习机会，让我学习了FPGA的有关知识，提高了动手实践能力、自学能力。我会好好归纳、总结、吸收这次暑期学校的收获，将其运用在往后的学习和工作中。