



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	基于 PYNQ 的调酒机器人
组员姓名	李明莉、葛倍倍、张俊杰、钱佳
房间号及桌号	715 教室 第 2 桌



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	李明莉	编写识别算法、机械臂控制程序
组员 1	葛倍倍	制定项目整体方案、编写 SDK 裸机程序
组员 2	张俊杰	编写 HLS 图像预处理、跟踪算法程序
组员 3	钱佳	编写 Vivado 工程程序、调试 HLS 算法仿真



第二部分 设计概述

(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)

2.1 功能说明

本作品通过 OV5640 CMOS 单目摄像头获取 1280*720 像素的图像，在 PYNQ 板卡 1 上用 HLS 高层次综合工具实现高斯滤波、腐蚀、膨胀等图像预处理算法和颜色分类识别算法，通过 HDMI 将视频流传输到 PYNQ 板卡 2 上，在 Jupyter 开发环境中实现目标跟踪算法，当目标进入摄像头视野中心位置时机械臂开始完成抓取酒瓶等动作，最后完成调酒功能。

2.2 设备清单

名称	数量
PYNQ 板卡	2 块
OV5640 模组	1 个
机械臂	1 个
RIO 酒瓶	3 个
液晶显示器	1 台
笔记本电脑	1 台
12V 电源适配器	1 个

第三部分 详细设计

（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。

3.1 系统详述

本作品由两块 PYNQ 板卡组成，板卡 1 主要完成 OV5640 驱动、图像预处理、颜色识别算法等功能，板卡 2 主要完成跟踪算法、机械臂控制、视频显示等功能。

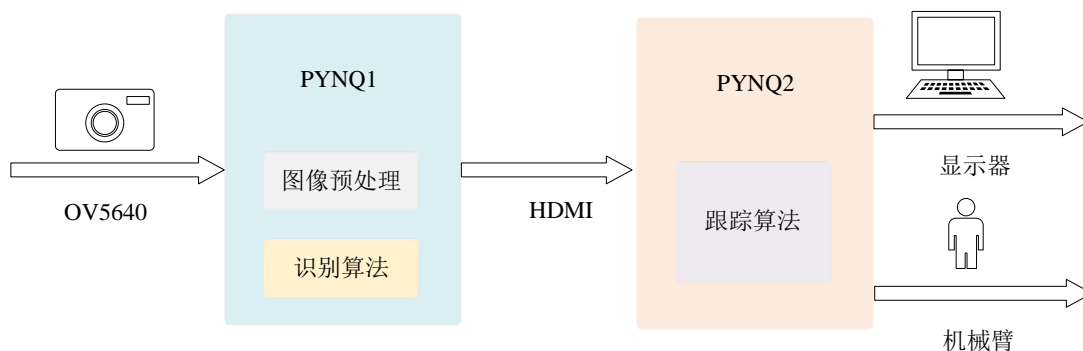


图 3-1 整体系统框图

3.2 板卡 1 功能详述

1、系统框架

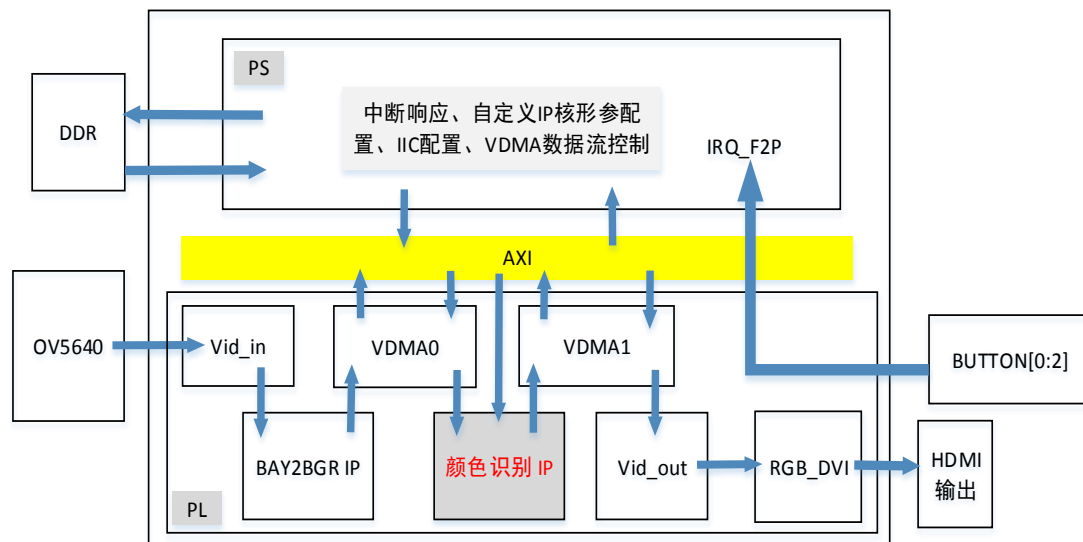


图 3-2 系统框图

该视频颜色识别系统主要是基于 PYNQ 板卡实现，该系统的硬件组成部分主要有 OV5640、PYNQ 板卡、HDMI 输出线。此系统设计思想主要按照异构架构框架进行设计，主要分为 PL 部分和 PS 部分，PL 部分主要完成按键中断响应、视频采集、颜色转换、颜色识别、HDMI 输出等功能；PS 部分主要完成该系统的初始化、中断服务、对 IP 核进行配置、数据流交互等功能。该系统的工作流程为：首先通过 IIC 总线对 OV5640 进行配置，其次



通过按键选择颜色识别的具体模式（B、G、R），并且完成整个系统的初始化。OV5640 摄像头采集的视频数据通过 Vid_in IP 核传输给 BAY2BGR IP 核完成视频的 RGB 转换，通过 VDMA 将转化后的视频数据通过 AXI 总线传输给 DDR 并将该数据保存。然后 VDMA 通过 AXI 总线将上部处理后的视频数据传输给颜色识别 IP 进行预先设定好的颜色识别模式进行颜色视频并将识别到的颜色数据通过 VDMA 传输给 DDR 并保存。最后 VDMA 将上部识别到的颜色数据通过 AXI 总线传输给 Vid_out IP 核，并传输给 RGB_DVI 转化成 HDMI 数据格式，通过 HDMI 接口传输给 PYNQ 板卡 2。

2、IP 核设计

该部分主要完成图像的颜色识别，主要包括视频流转化成 Mat 格式、图像滤波、颜色识别、膨胀、腐蚀、灰度图像转 RGB 格式、Mat 格式转化为视频流等。

首先调用 HLS 中 hls_video.h 的 hls::AXIvideo2Mat 函数将视频流转化为 Mat 数据格式，然后使用高斯滤波对视频数据进行降噪处理，

然后使用自定义的颜色检测算法进行颜色识别，该算法的原理主要是针对图像每个像素点对应的 RGB 三通道的值根据之前选择模式的 RGB 对应的阈值进行比较，满足要求的设置其灰度值为 255，否则为 0 并输出二值图像。

膨胀、腐蚀调用 HLS 中 hls_video.h 的 hls::Erode、hls::Dilate 函数，将上述得到的二值图像进行降噪处理。

然后将上述处理后的二值图像经过 hls::CvtColor 将二值图像转化为 RBG 图像。

最后调用用 HLS 中 Mat2AXIvideo 函数将 Mat 数据格式转化为视频流格式。

该 IP 核设计的主要接口分为两类，主要为 AXI HP 和 AXI GP 接口。AXI HP 主要负责视频数据的输入和处理后视频的输出、AXI GP 接口主要用于对该 IP 核的参数配置、以及返回值的接收。该 IP 核的参数配置主要有三种颜色识别模式的选择、R、G、B 分别设定的阈值。

颜色识别算法：首先读取图片每个像素点对应的 RGB 三通道的值。然后根据预先设定好的颜色识别模式进行颜色识别方法的选择，最后根据选择的方法对每个像素点的 RGB 三通道的值进行遍历，满足条件的设定该点的像素值为 255，负责为 0。

下图为封装好的 IP 核。

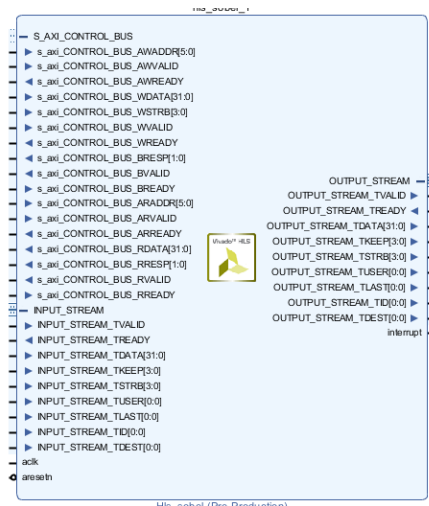


图 3-3 IP 核

3、Vivado 硬件

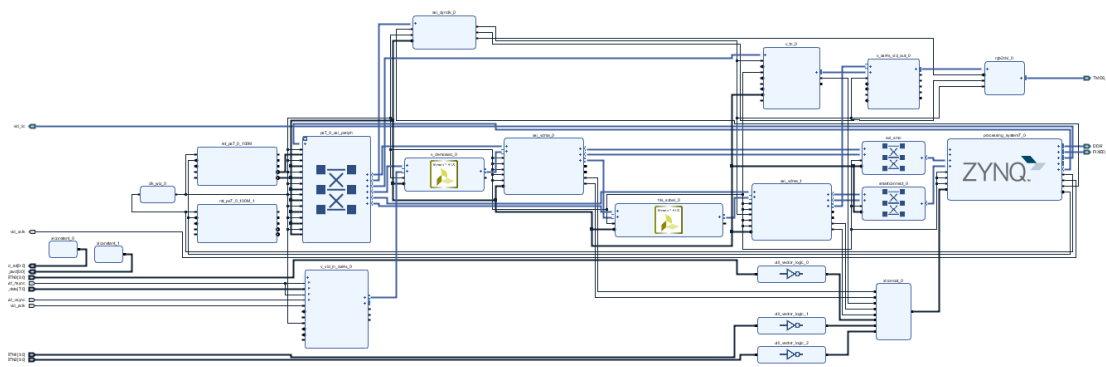


图 3-4 硬件工程

首先三个按键连接到 PS 端的中断上产生中断响应。

OV5640 摄像头采集的视频数据通过 Vid_in IP 核传输给 BAY2BGR IP 核完成视频的 RGB 转换，通过 VDMA0 将转化后的视频数据通过 AXI GP 总线传输给 DDR3 并将该数据保存在 DDR3。然后 VDMA0 通过 AXI GP 总线将上部处理后的视频数据传输给颜色识别 IP 进行预先设定好的颜色识别模式进行颜色视频并将识别到的颜色数据通过 VDMA1 传输给 DDR 并保存。最后 VDMA1 将上部识别到的颜色数据通过 AXI 总线传输给 Vid_out IP 核，并传输给 RGB_DVI 转化成 HDMI 数据格式，通过 HDMI 接口传输给 PYNQ 板卡 2。

4、SDK 应用程序

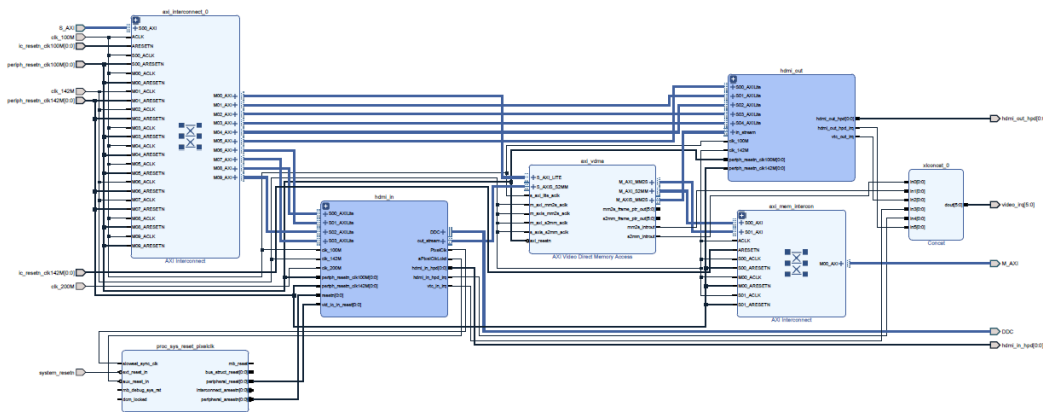
PS 部分主要完成该系统的初始化、中断服务、对 IP 核进行配置、数据流交互等功能。

首先通过 IIC 函数完成对 OV5640 的配置，XV_demosaic IP 核的配置以及初始化，整个系统的初始化，根据按键产生的中断请求，对颜色识别 IP 核进行对应的配置，配置的参数包括颜色识别的模式选择以及对应每种模式的 RGB 的阈值。完成上述配置后，分别对 VAMA0 IP 核、VDMA1 IP 核的读写进行配置，完成该系统的数据传输控制。

3.3 板卡 2 功能详述

1、硬件部分

整个系统在 Vivado 开发套件中，以模块化的形式搭建，下图是 PYNQ2 板卡的 Block design 设计图：



2、软件部分

本作品整个系统运行在 Ubuntu 嵌入式 linux 操作系统上，采用 python 语言设计目标检测和跟踪算法，并驱动相应的 FPGA 模块，软件流程图如下图 3-7 所示：

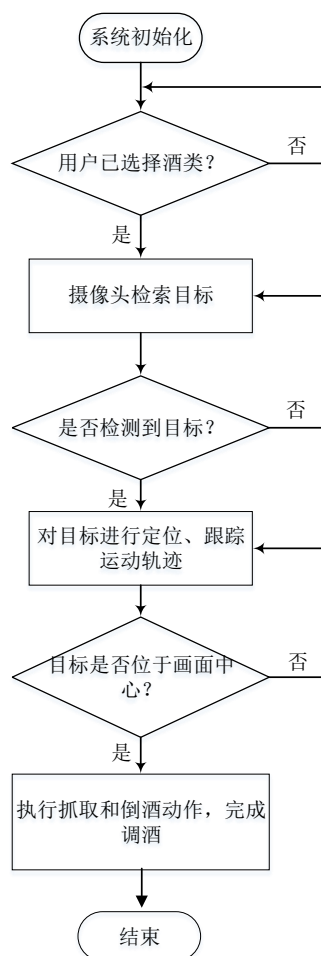


图 3-7 软件流程图

板卡上电后，可通过网络访问板卡，在 Jupyter 内运行软件应用程序，通过显示屏，观察目标识别和跟踪效果，观察机械臂的调酒过程，且从 Jupyter 中一边运行，可以一边观察进程的中间结果。

第四部分 完成情况及性能参数

（作品完成情况，附照片/视频链接）

4.1 作品完成情况

基本功能已经实现，包括 OV55640 获取图像，在 PYNQ 板卡 1 上实现图像预处理和对 RGB 三种颜色的识别算法，在 PYNQ 板卡 2 上实现跟踪贴有颜色标签的酒瓶算法和控制机械臂完成抓取酒瓶等动作。

4.2 照片

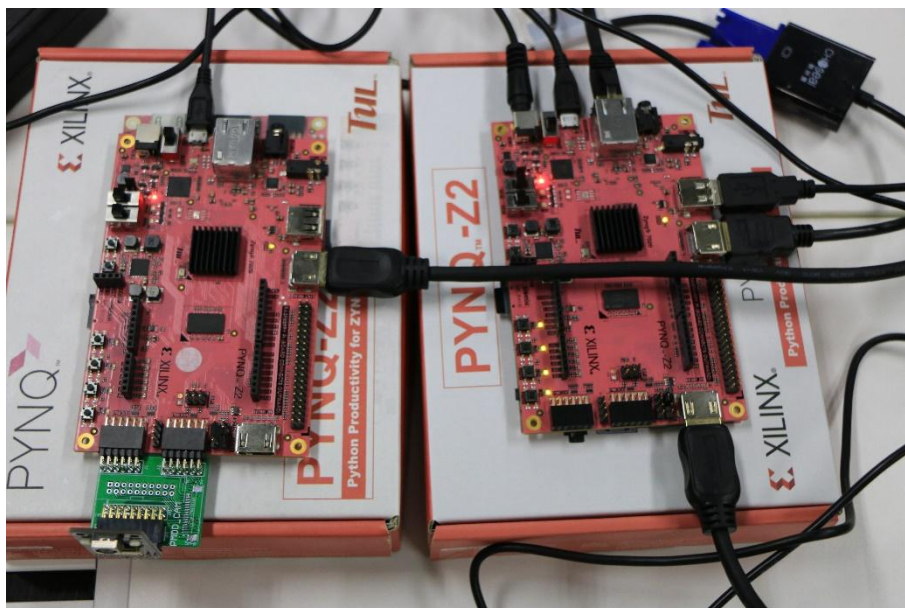


图 4-1 PYNQ 板卡

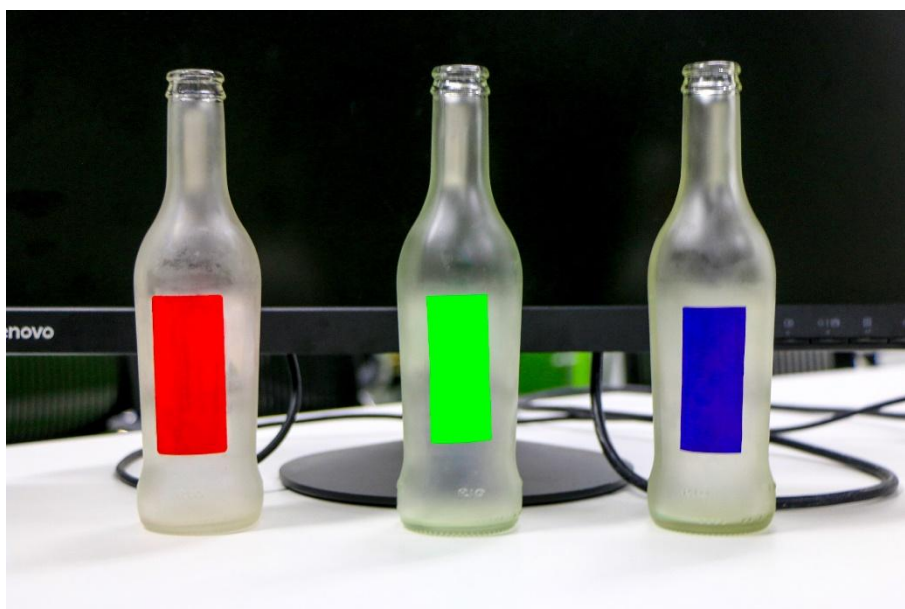


图 4-2 RGB 酒瓶

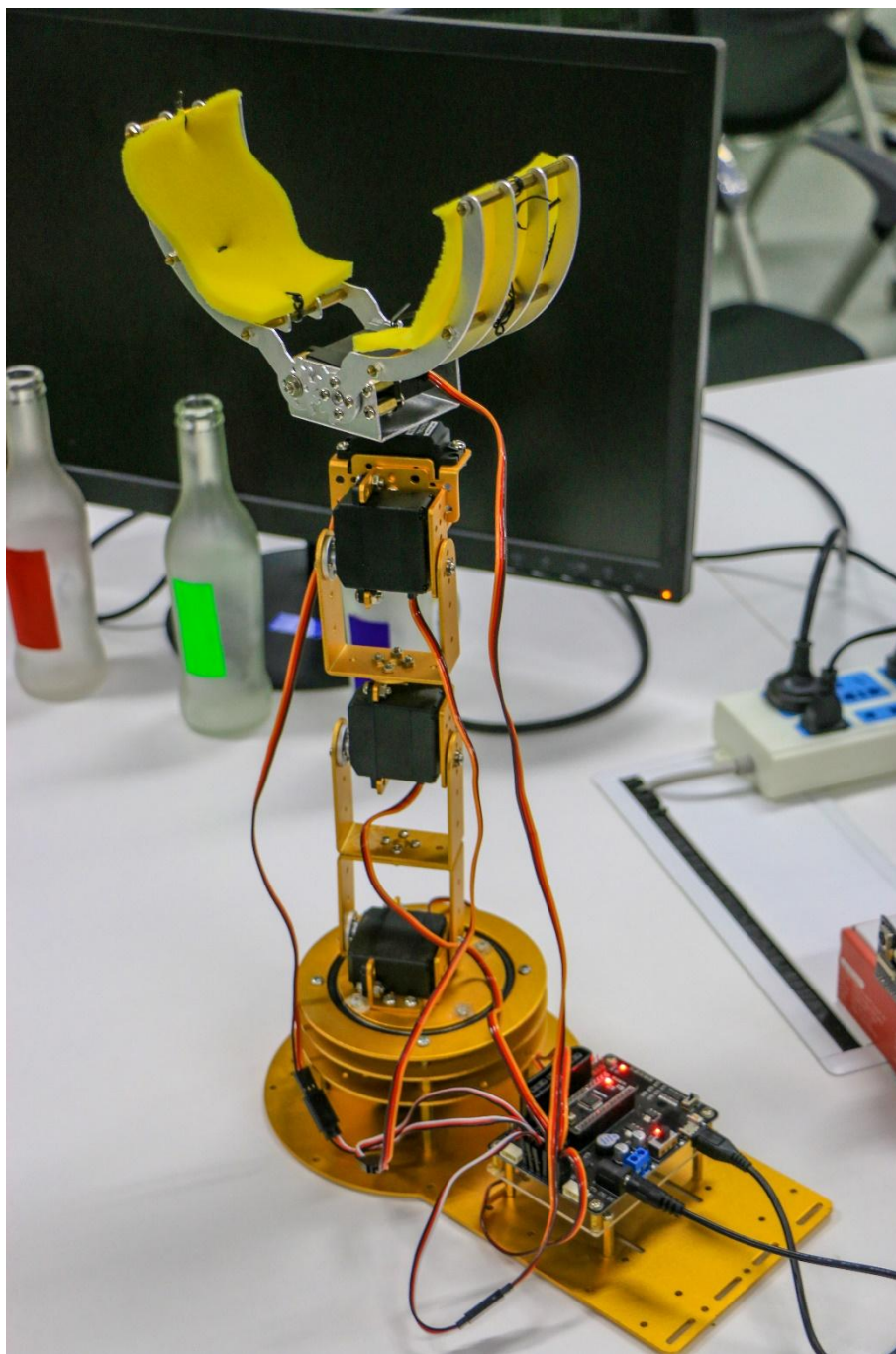


图 4-3 机械臂

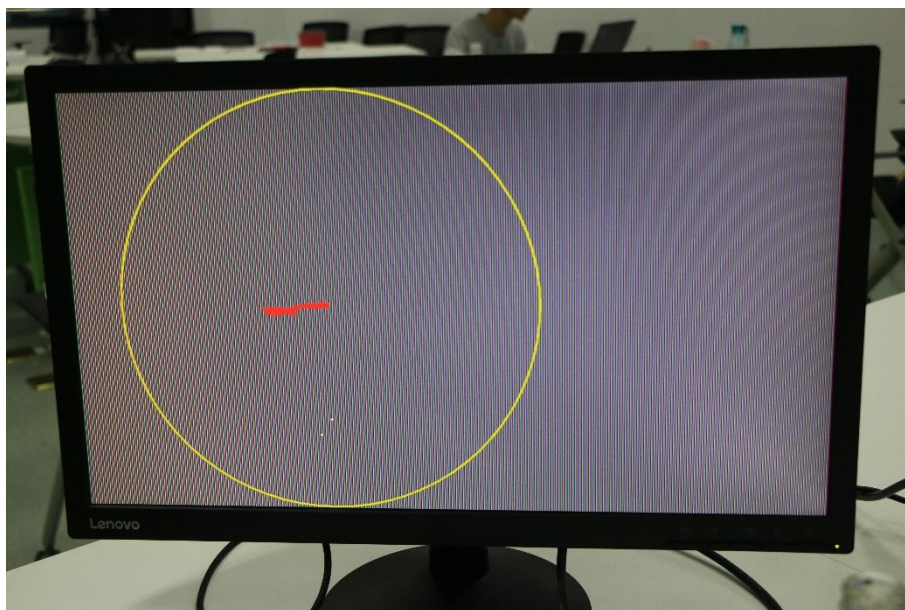


图 4-4 跟踪结果



图 4-5 整体系统

第五部分 项目总结

（项目中所用到的知识点，项目收获及心得）

5.1 知识点

1、PYNQ 框架

PYNQ-Z2 开发板是一款支持 PYNQ 开源框架的开发平台。PYNQ 开源框架可以使嵌入式编程用户在无需设计可编程逻辑电路的情况下充分发挥 Xilinx Zynq All Programmable SoC (APSoC) 的功能。PYNQ-Z2 除支持传统 ZYNQ 开发方式外，还可支持 Python 进行 APSoC 编程，并且代码可直接在 PYNQ-Z2 上进行开发和调试。可编程逻辑电路以硬件库的形式导入并且可以通过 API 编程，这种方式基本上与软件库的导入和编程方式相同。

PYNQ-Z2 开发板以 ZYNQ XC7Z020 FPGA 为核心，配备有以太网，HDMI 输入/输出，MIC 输入，音频输出，Arduino 接口，树莓派接口，2 个 Pmod，用户 LED，按钮和开关。兼容树莓派连接器、Arduino 屏蔽连接器以及 Pmod 连接器可以支持多种配件拓展，同时这些接口也可以用作 GPIO。

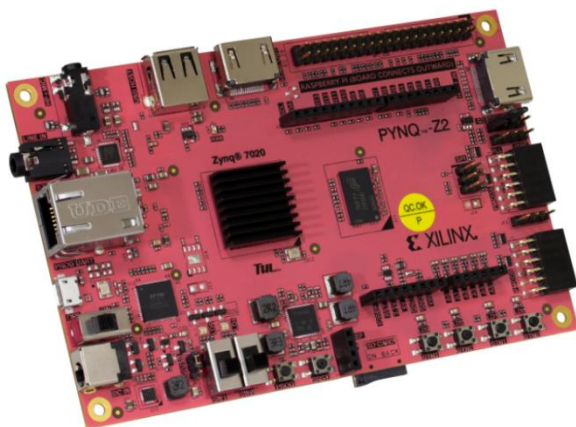


图 5-1 PYNQ-Z2 开发板

2、机械臂

本作品采用的是如图 5-2 所示的 6 路舵机控制器，其中集成了稳压模块，主控制器，低压报警模块，过载保护模块，无线通信模块和串口通信模块。其中舵机供电需求在 5 到 8V 左右，主控的供电在 3.3 到 5V 左右，所以需要稳压模块调节电压。同时使用串口通信支持图形化编程，可以快速调出各种动作，支持在线调试和下载动作组和脱机运行。此外，可以通过 wifi 或者蓝牙来实现无线通信，可以远程调节机械臂和监控。

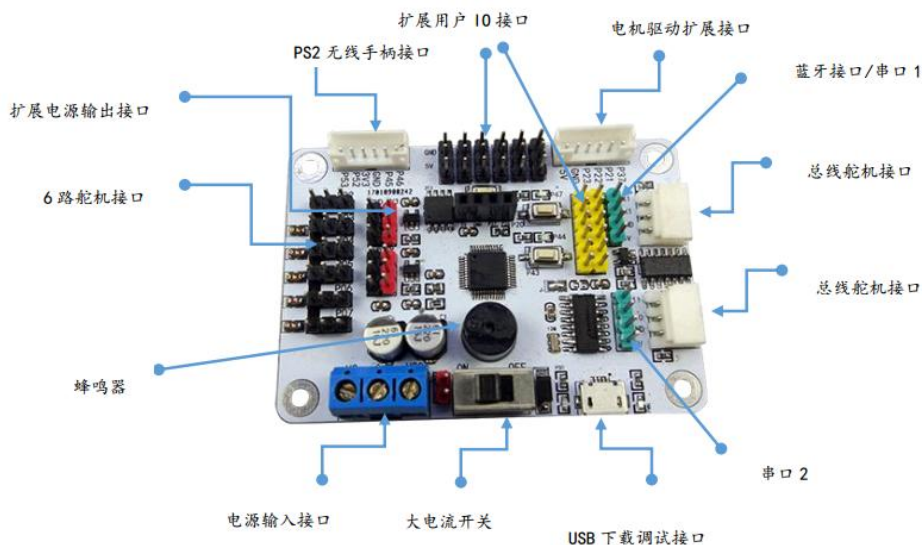


图 5-2 舵机控制器

舵机采用的是 TBSN-K15 和 TBSN-K20 数字大扭矩舵机，分别可以支持 15kg 和 20kg 的抓力。可控角度范围可以达到 180 度，线性度好，适用于精确控制，断电情况下可以支持 360 度旋转。同时具有防堵转保护，以防止角度超出阈值或者在长时间卡在某一角度值而烧毁电机。舵机的伺服系统通过 PWM 波来调节角度，周期 20ms，占空比为 0.5ms-2.5ms，采用 500-2500 的数值对应，最小控制精度达到 0.24 度。



图 5-3 机械臂舵机及控制器实物图

3、HLS 高层次综合

当今无线、医疗、军用产品和消费类产品应用中使用的高级算法比以往更加复杂。Vivado 高层次综合 (HLS) 在所有 Vivado HLx 版本中以免费升级形式提供，可以实现直接使用 C, C++以及 System C 语言规范对赛灵思可编程器件进行编程，无需手动创建 RTL，

从而可加速 IP 创建。Vivado 高层次综合使系统和设计架构师支持 ISE 和 Vivado 设计环境，能够以更快速的方式创建 IP。

4、图像处理算法

bayer 格式图片是伊士曼·柯达公司科学家 Bryce Bayer 发明的，Bryce Bayer 所发明的拜耳阵列被广泛运用数字图像。

对于彩色图像，需要采集多种最基本的颜色，如 RGB 三种颜色，最简单的方法就是用滤镜的方法，红色的滤镜透过红色的波长，绿色的滤镜透过绿色的波长，蓝色的滤镜透过蓝色的波长。如果要采集 RGB 三个基本色，则需要三块滤镜，这样价格昂贵，且不好制造，因为三块滤镜都必须保证每一个像素点都对齐。当用 bayer 格式的时候，很好的解决了这个问题。bayer 格式图片在一块滤镜上设置的不同的颜色，通过分析人眼对颜色的感知发现，人眼对绿色比较敏感，所以一般 bayer 格式的图片绿色格式的像素是 r 和 g 像素的和。

另外，Bayer 格式是相机内部的原始图片，一般后缀名为.raw。很多软件都可以查看，比如 PS。我们相机拍照下来存储在存储卡上的.jpeg 或其它格式的图片，都是从.raw 格式转化过来的。如图 5-4 所示为 bayer 色彩滤波阵列，由一半的 G，1/4 的 R，1/4 的 B 组成。

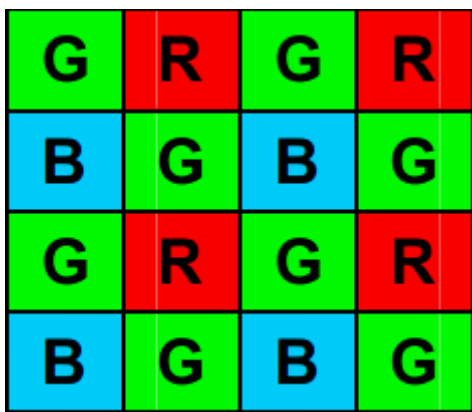


图 5-4 bayer 色彩滤波阵列

5.2 项目收获及心得

李明莉：

首先，非常感谢本次大赛组委会的大力支持，不仅为我们提供 FPGA 开发所需要的板卡，还安排了经验丰富的技术支持。两周的“1+1”学习和实战生活，让我收获颇丰。

第一周的课程，让我对 vivado、HLS 的开发套件的使用，有了由点到面的认识。特别是设计过程中，时序分析、优化方向等知识，细致全面，对工程设计意义非凡。另外，在学习过程中，更加深入地了解 ZYNQ 系列基于 ARM+FPGA 的异构模式和性能，也学习了 PYNQ 框架，使用 Python 生态，进行 FPGA SOC 系统开发的过程，并在项目实战中，将所学到的 HLS、PYNQ 框架融合起来，设计了一款基于 PYNQ 的机器人调酒师。其次，要感谢一直鼓励和关注着我们的指导老师，激励我们顺利完成作品。最后，要特别感谢，这段时间以来，感谢各位组员的肯定和信任，作为本次项目的队长，能与大家并肩作战、分工



协作、共同学习进步，让我们意识到团队的力量远远大于个人的力量，是大家的通力协作和努力耕耘，才取得如此成果。

葛倍倍：

对于第二次参加暑期学校的我而言，内心非常激动，一方面是可以和 Xilinx 原厂工程师近距离交流技术，另一方面可以和来自国内外各个学校的同学们交流学习生活等方面的问题。

时间紧、任务重，几乎每天都要在 ICisC 中心学习到晚上 10 点才回宾馆休息，累并快乐着。犹记得第一次分班时，刘老师和班委们忙到晚上 10 点把班级分好然后通过微信群通知大家；每天晚上提前把实验做完，第二天可以帮助同学们答疑解惑，更好地完成实验；项目分组之后，小伙伴们为了完成项目不断调整技术方案，不断尝试新的算法效果，在通路实现的瞬间，高兴地鼓掌，或许这就是团队的力量。

12 天，7 场专题讲座，3 个班委，3 个实习生，4 名组员，1 个项目，构成了这个夏天最美好的回忆，感谢东南大学，感谢 Xilinx，感谢 ICisC，期待下一次相遇。

张俊杰：

通过这一次培训学习，我更加深入地理解了 ZYNQ 的异构架构框架和优势，同时学到了 PYNQ 使用 Python 平台进行 FPGA SOC 系统的开发过程，对 PS 端的 ARM 处理器和 PL 端的可编程逻辑电路部分有了更深刻的理解和认识，其次学习到了这种异构架构做深度学习加速的思想，同时了解了该技术以后的发展趋势，此外，通过实际的项目开发，增加了我的专业知识和工作能力，并且提高了我的问题解决能力，提高了我的团队分工和协调合作等各方面的能力，我意识到团队的力量远远大于个人的力量，只有整个团队齐心协力，才能够攻坚克难，共同进步。

钱佳：

之于 FPGA 暑期学校，我算是一个“老学员”了。去年作为工作人员，我只是一个记录者，然而今年，我是参与者！

这次 FPGA 暑期学校对我来说，不仅仅是实习经历这么简单。习惯了学校授课的模式和节奏的我，在第一天看到我们的日程安排之后，就意识到这可能是我们普通的高校学生在学业过程中与企业离得最近的一次。因为这些课程都是我们在学校无法接触到的，甚至这种授课思路也见所未见。我们在老师的引导下得以跟上 FPGA 开发者的脚步，领略了 FPGA 得以从千军万马中脱颖而出的原因。老师为我们推开了 FPGA 的大门，让我们见识到了更为广阔的新世界！

后半段做项目的经历更为宝贵，短短一周的时间，我觉得自己有了很大的突破。跟着大神队友们，我学到了很多很宝贵的知识，获得了很多工程经验并且大大提升了自己的技术水平。更重要是我拓宽了视野，“原来 FPGA 还能做这些事”是我这几天脑子里出现最多的一句话！通过这次 FPGA 暑期学校，我学到的更多的是面对庞大知识量的学习方法，



这是一种软性的学习方式和思路，比所有固性的 FPGA 知识经验更重要。还有经过和老师同学的不断交流，我找到了自己的发展方向和前进道路。

这个夏天,FPGA 使大家相遇相识,一起领略了异构系统的精妙和 AI 加速的科技酷炫,了解到黑科技也能玩的花彩多样。人间戏梦,滚滚红尘, FPGA 盛宴,愿大家彼此交情不浅,江湖路远,有缘再见~



第六部分 源代码

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