**实验指导手册**



**自动化专业卓越工程师培养计划实训报告**

**Facein智能会议签到系统**

**Designed By：向明 李想**

1358218957@qq.com

478959566@qq.com

目录

[实验介绍 3](#_Toc393181022)

[实验目标 3](#_Toc393181023)

[设计描述 3](#_Toc393181024)

[流程概述](#_Toc393181025) 4

[结论](#_Toc393181031) 16

[参考文献](#_Toc393181032) 16



# 实验介绍

本实验将引导同学们在ZYBO板上搭建基于Linux系统运营的微信公众号平台，通过七牛云或者腾讯云实现云存储功能，微信调用面向对象平台开发的摄像头拍照，人脸识别实现智能签到的功能。另外，通过申请云主机来搭建一个永久服务器继续后续微信平台的开发工作。

# 实验目标

在完成本实验后，你将能够：

* 掌握Linux系统的一些简单操作以及命令
* 基本掌握Linux的脚本（shell）编程工作
* 掌握使用市面上大部分的云计算平台以及云存储平台
* 掌握微信收发信息，完成交互的基本流程
* 利用云主机继续后续开发工作

# 设计描述

本实验需完成搭建以下平台：ZYBO板Linaro最小系统平台、微信公众平台、照片拍摄云存储平台（将硬件平台上的USB摄像头所拍摄的图片上传到七牛云或者腾讯云）、人脸识别平台（借用Face++第三方技术平台），利用mutt邮件发送信息平台。

微信发送录入/签到指令

ZYBO

微信

ZYBO响应微信的拍照请求

返回结果

请求拍照

Face++人脸识别平台

摄像头

七牛云/腾讯云存储服务

调用Face++

照片上传至云端

# 流程概述

**Step1：ZYBO板Linaro系统配置**

----------------------------------------------------------------------

准备工作：

1. ZYBO板1个、TF卡1张（8GB）、VMware Workstation Pro
2. 通过前序实验使用Xilinx EDK创建Zynq最小硬件系统、在SDK下生成boot.bin文件
3. 了解如何编译Linux内核、在虚拟机环境下生成zImage文件（用于启动TF卡）
4. 文件见压缩包

----------------------------------------------------------------------------------

1. Linux内核编译常用命令及常见问题

ls /mnt/hgfs/：显示被共享的文件夹

pwd：查看当前目录

tar xzf：解压缩

make ARCH=arm digilent\_zybo\_defconfig：加载内核关于ZYBO板的默认设置

make ARCH=arm menuconfig：进行图形化内核设置

make ARCH=arm：编译内核

./scripts/dtc/dtc -I dts -O dtb -o devicetree.dtb ./digilent-zybo.dts：生成dtb文件

1. 准备Linaro系统配置常用命令

mkdir –p /tmp/linaro：创建临时文件夹

mkdir –p /tmp/ext\_4：建立TF卡临时目录

sudo mount /dev/sdb2 /tmp/sd\_ext4：加载TF卡

cd binary/boot/filesystem.dir：进入Linaro文件系统

sudo rsync –a ./ /tmp/sd\_ext4：同步文件系统到TF卡

sudo umount /tmp/sd\_ext4：安全移除TF卡

1. VNC服务器的安装与使用方法

sudo apt-get install tightvncserver

Linaro图形界面进程生成命令：vncserver -geometry 1366x768 -depth 24

Linaro图形界面进程杀死命令：vncserver -kill :1

打开VNCViewer,在地址栏填入：IP:1 (其中IP为ZYBO板的IP地址)

**Step2：微信公众平台搭建**

**----------------------------------------------------------------------**

准备工作：

1. 注册微信公众号（订阅号）1个，成为开发者，网址：<https://mp.weixin.qq.com/>
2. 注册Pagekite网站，网址：<http://pagekite.net/>
3. 运行正常Linaro系统的ZYBO开发板1个，连接串口终端
4. 文件见压缩包

----------------------------------------------------------------------------------

1. apt-get更新Linaro系统软件库

sudo apt-get update

1. 安装ssh服务器

sudo apt-get install ssh

选择压缩包内的SSH Secure Shell或者putty实现远程登录到ZYBO板上，执行命令：ifconfig查看当前ZYBO板的内网IP地址（让ZYBO板共享计算机的网络），在地址栏上输入IP地址，设置root帐户密码命令：su passwd root，在终端输入密码登陆成功，通过这种方法可以通过网络的方式与ZYBO板进行文件的传输。

1. 修改ZYBO板网卡MAC地址

sudo /sbin/ifconfig eth0 down：关闭网卡设备

sudo /sbin/ifconfig eth0 hw ether 00:AA:BB:CC:DD:EE：修改MAC地址

sudo /sbin/ifconfig eht0 up：重启网卡设备

1. 安装apache2服务器

sudo apt-get install apache2

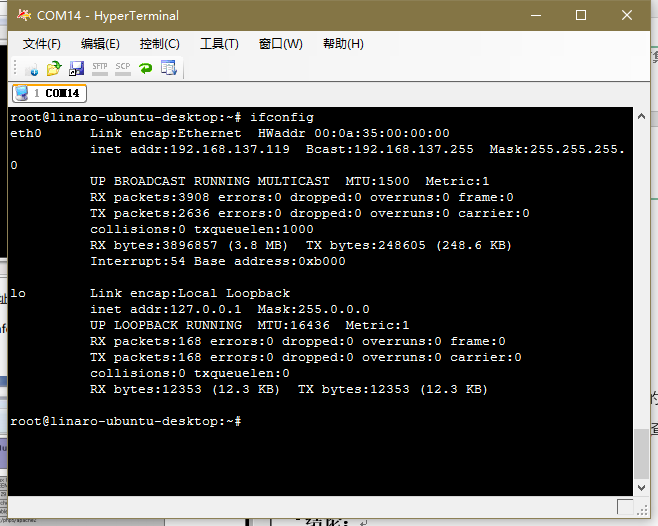
1. 安装PHP5

sudo apt-get install php5

1. 安装vim

sudo apt-get install vim

1. 测试php

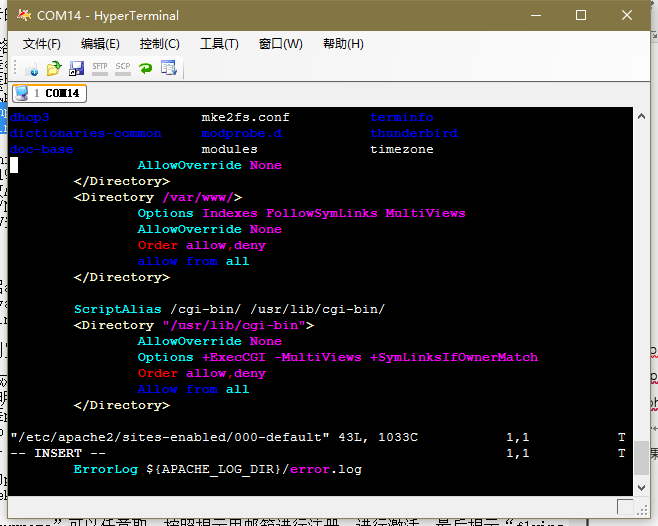
vim /var/www/info.php在Linaro网络服务器的目录/var/www下建立一个文件用于测试php是否安装成功，判断方法：运行命令ifconfig查看ZYBO当前的IP地址，在浏览器里输入<http://IP/info.php>（例如<http://192.168.137.119/info.php>），如果能够看到PHP信息，则表示php安装成功。

info.php的文件内容如下：

<?php

phpinfo();

?>

如果需要修改Linaro网络服务器的目录，运行命令：vim /etc/apache2/sites-enabled/000-default，修改当前文件中的/var/www/即可。

1. 修改apache2的配置文件

运行命令：vim /etc/apache2/sites-enabled/000-default.conf，在配置文件的最后(注意：在</VirtualHost>之前)添加如下字段，并保存退出。

<IfModule dir\_module>

DirectoryIndex index.php index.html

</IfModule>

1. 重启apache2服务

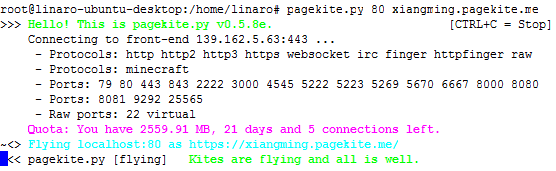
sudo service apache2 reload

1. 安装内网穿透工具curl

sudo apt-get install curl

curl -s https://pagekite.net/pk/ |sudo bash

1. 启动pagekite服务：

pagekite.py 80 yourname.pagekite.me（yourname可以自己自由取，按照提示用邮箱进行注册，进行激活。最后提示“flying well”表示pagekite服务启动成功。）

1. 登陆微信公众平台

打开微信公众平台，网址：<https://mp.weixin.qq.com/>，点击左侧的开发基本配置修改配置填写以下信息启用

URL(服务器地址)：http://yourname.pagekite.me

Token(令牌) ：XXXXXX(用户随意填写)

EncodingAESKey：点击右边的按钮自动生成

消息加解密方式：明文模式

1. 上传php文件

先将压缩包中的Wechat官方的php放到/var/www文件夹下，修改Token值，点击提交。

解压压缩包，将解压之后的index.php文件上传到ZYBO的网站根目录下(默认为/var/www,注意操作权限)。修改index.php的Token，要和微信公众平台开发者中心设置里TOKEN相一致。

一定要注意这里的php一定要是先用官方的php，不然会出现请求URL失败或者验证Token失败的情况。

1. 测试

之后用微信客户端添加微信公众订阅号，然后发送“?”可以获取ZYBO发出的回复响应。

**Step3：云计算/存储平台的搭建和使用**

----------------------------------------------------------------------------------

准备工作：

1. 七牛云/腾讯云其中选一的云平台帐户1个
2. USB接口的摄像头1个
3. 文件见压缩包

----------------------------------------------------------------------------------

1. 配置七牛云

网址：<http://www.qiniu.com/>

配置：邮箱注册，登陆之后点击左侧添加资源对象存储立即添加填写bucket识别符确定创建找到测试域名保存点击右上角按钮密钥管理保存AccessKey/SecretKey

1. 查看七牛云官方开发者文档

网址：<http://developer.qiniu.com/code/v6/sdk/legacy-php.html>

SDK源码GitHub网址：<https://github.com/qiniu/php-sdk/tags>

由于本实验采用的SDK版本为6.1.11（提供SDK 适用于 PHP 5.1.0 及其以上版本），所以本实验的所有PHP（服务端）开发都是在6.1.11版本上进行开发的，同学们也可以尝试更高的版本SDK，这里不做赘述。

将压缩包内文件夹/qiniuyun通过上文提到的SSH方法解压到/home/linaro文件夹下，将/home/linaro/qiniuyun /php-sdk-6.1.11/docs/gist/目录下的php文件中的bucket/AccessKey/SecretKey/domain（测试域名）/key（上传到云端的文件重命名）/file（本地要上传的文件路径）值按照自己的七牛云设置。

1. 安装php5-curl插件

sudo apt-get install curl libcurl3 libcurl3-dev php5-curl

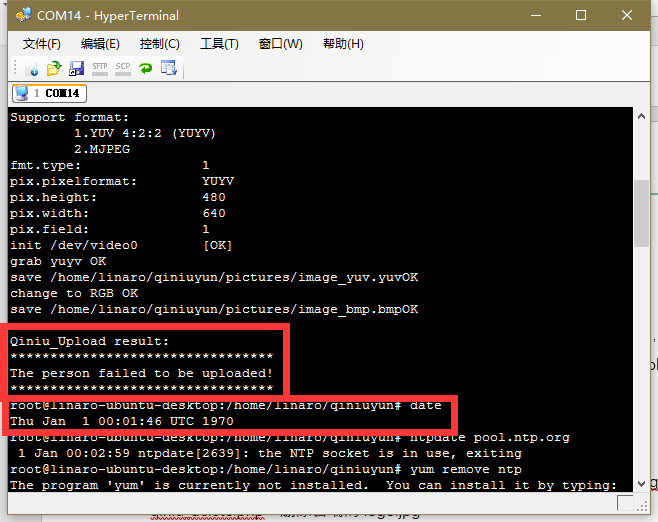
1. 重启apache2服务

sudo service apache2 reload

1. 安装ntpdate同步网络时间

sudo apt-get install ntpdate

ntpdate pool.ntp.org

\*注：同步网络时间这一步很重要，笔者运行的ZYBO板上没有硬件时钟，一旦断电重启之后都会恢复Unix系统元年时间戳，即1970.01.01，在这样的情况下，微信的Token验证就不会得以通过，七牛云的接口请求也不会执行，谨记！

时间未同步前发生错误

1. php文件功能说明

qiniu\_upload.php：上传对象：logo,jpg，单次执行上传一次本地logo,jpg

qiniu\_delete.php：删除云端的logo.jpg

qiniu\_status.php：查询云端的logo.jpg是否存在

qiniu\_view.php：预览云端的logo.jpg，可以在php文件里进行修改预览参数（长，宽）,返回预览的URL，复制到浏览器打开即可

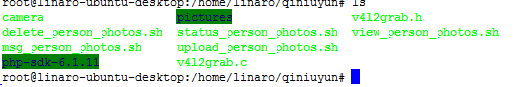
qiniu\_msg.php：查询云端的logo.jpg的信息，返回信息的URL，包含长，宽，色彩信息

1. 其他说明

压缩包中的qiniuyun/文件夹解压到/home/linaro下后，将摄像头连接到ZYBO板上，请同学们运行命令如下：

cd /home/linaro

chmod –R 777 qiniuyun/

执行后看到qiniuyun/变绿了就能执行里面存放的脚本文件，同学们也可以自行执行脚本文件测试效果。

如果执行成功的话，请自行进入七牛网空间，查看内容管理，查看是否有logo.jpg文件存在。

1. 配置腾讯云

网址：<https://www.qcloud.com/>

配置：QQ帐号登陆管理中心对象存储服务左侧点击Bucket列表创建Bucket填写Bucket名称勾选CDN加速创建密钥管理保存AccessKey/SecretKey

1. 查看腾讯云官方开发者文档

网址：<https://www.qcloud.com/doc/product/227/3384>

SDK源码GitHub网址： <https://github.com/tencentyun/cos-php-sdk>

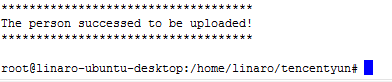
将压缩包内文件夹/tencentyun通过上文提到的SSH方法解压到/home/linaro文件夹下，将/home/linaro/tencentyun /cos-php-sdk-master目录下的php文件中的bucketName/ dstFolder（在云端创建的文件夹名）/ domain（测试域名）/person\_name（上传到云端的文件重命名）/srcPath（本地要上传的文件路径）值按照自己的腾讯云设置，将/home/linaro/tencentyun /cos-php-sdk-master/Qcloud\_cos目录下的Conf.php文件中的APPID/ SECRET\_ID/ SECRET\_KEY值按照自己的腾讯云设置。

腾讯云的SDK使用方法类似于七牛云，请同学们自行测试，这里不再赘述。

1. 七牛云/腾讯云优缺点分析

七牛云：开放的功能接口比较单一，偏重于上传文件和图片预览，代码量偏大，同样的功能所要编写的参数和代码量多于腾讯云。

腾讯云：开放的功能接口更加丰富，结合腾讯云CDN服务器加速上传图片和浏览图片，和腾讯云主机的结合更加紧密，可以在Bucket内进行文件夹的创建工作，注意删除文件夹命令需要保证文件夹唯恐，类似于Linux系统，其次，腾讯云代码调用方便。

请同学们自行选择要使用的云存储服务器。

腾讯云上传成功

1. 申请云主机

点击云产品云服务器选择配置（1核/1G/20GB Ubuntu）系统分配云主机

同学们可以将搭建微信服务器的工作在云端复现一次，这里不再赘述。

**Step4：项目搭建和php代码分析**

----------------------------------------------------------------------

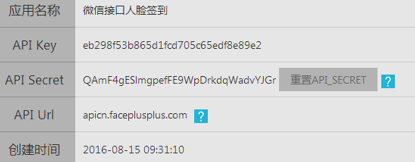
准备工作：

1. 运行微信服务器/连接摄像头的ZYBO板1个
2. 文件见映像文件

----------------------------------------------------------------------

1. php代码总览：

验证签名及响应消息；接收消息；回复消息。（具体代码见映像文件中的index.php)

1. 由于项目的主体是人脸识别，这里我们使用face++一个免费提供人脸识别API接口的网址：<http://www.faceplusplus.com.cn/>
2. 首先，在该网站首页注册账户，然后在<我的应用>下创建应用。你会得到API\_KEY和API\_SECTER。这两串字符串在以后使用网站资源的接口。其他的选项可以随便选，不过服务器建议选国内的。
3. 接下来如果想引用例程的，可以进入<开发者工具与SDK>下选择相应语言的Demo。但是这里我们没有用到开发者的例程。因为例程会很有束缚感，自己想要的数据例程中不一定会提供。而且传参的时候会有很大问题。
4. 这里我们用到的是API文档，也就是编写例程的人使用的网站资源，由于使用API文档可以获取到全部的FACE++提供的参数，所以我们在做数据分析时会比较轻松。
5. 使用API 文档说明：API文档只是提供了一个又一个网址，并没有提供其他代码，我们每次可以调用这些网址及这些网址返回的信息。并将这些信息进行提取和拼装组合来为下一次的调用进行准备。下一次的调用又为下下一次的调用做准备。道理很简单，但实践起来你会发现有很多的问题。
6. 上面讲的是该系统最终实现功能的设计思路，下面讲讲具体流程及对一些函数功能说明。首先API文档是从上往下看的，前面的网址调用结果一般会在下面的网址调用中使用。如果你在调用网址中发现先没见过的参数请到前面的API文档中的结果中寻找。
7. 下面说说一些网址调用的事情，接口页面前面会给出一些参数和返回值的说明，之后会给一个网址，我们就是要修改这个网址的参数，使它成为我们需要的网址。然后访问这个网址，会得到一个json。[JSON](http://baike.baidu.com/view/136475.htm" \t "http://baike.baidu.com/_blank)([JavaScript](http://baike.baidu.com/view/16168.htm" \t "http://baike.baidu.com/_blank) Object Notation) 是一种轻量级的数据交换格式。它基于[ECMAScript](http://baike.baidu.com/view/810176.htm" \t "http://baike.baidu.com/_blank)的一个子集。 JSON采用完全独立于语言的文本格式，但是也使用了类似于C语言家族的习惯（包括[C](http://baike.baidu.com/subview/10075/6770152.htm" \t "http://baike.baidu.com/_blank)、C++、[C#](http://baike.baidu.com/view/6590.htm" \t "http://baike.baidu.com/_blank)、Java、JavaScript、[Perl](http://baike.baidu.com/view/46614.htm" \t "http://baike.baidu.com/_blank)、[Python](http://baike.baidu.com/view/21087.htm" \t "http://baike.baidu.com/_blank)等）。这些特性使JSON成为理想的数据交换语言。 易于人阅读和编写，同时也易于机器解析和生成(一般用于提升网络传输速率)。
8. 我把json当成一个文本文件来处理，可以通过cat，fopen，fwrite，fread等命令来进行修改和拼装，在本文档的最后我会说明一下该人脸识别模块所用的命令及其功能。
9. 下面讲一下怎么返回json的，每次可以在shell脚本文件中使用curl（前面安装过的）提取到文本数据（可以用txt文件接收）。然后用grep命令再次提取。json就到你指定的文本文件中去了。
10. 接下来是移植须知：API\_KEY和API\_SECTER以及组的更改数据全都放在了index.php中的如下位置，但是组的创建并没有写到用微信来实现，因为该系统中的人都在一个分组中，并不需要多个组，至于为什么写一个创建组的shell，是之后的网址调用必须用到组名所以写了一个脚本文件。用户可通过运行该脚本自行创建组。哦还有微信的token也改一下！
11. 最后列一下人脸识别模块用到的shell脚本文件及其功能：
    * + 1. Group.sh:创建一个组
        2. Person.sh:创建一个人
        3. Face.sh:录入一张脸并把它添加给一个人
        4. Recognition.sh：识别一张脸，判断它与组中人脸匹配的相似度
        5. Delete\_all.sh:删除组内的所有人（删除的人脸还存在你的应用中）
        6. Traingroup.sh:对组内的人脸进行训练（相当于保存每次添加或删除人脸时调用）
12. 关于邮件部分的代码应用
13. 安装mutt

在ubuntu下安装mutt很方便，只需要sudo apt-get install mutt

1. 安装msmtp -> 一个发邮件的小工具

sudo apt-get insall msmtp

1. 配置文件

MUTT  
系统全局设置/etc/Muttrc,如果使用某个系统用户，可以在~/.muttrc中设置。  
vi .muttrc  
set sendmail="/usr/bin/msmtp"  
set use\_from=yes  
set realname="FengYuBin"  
set [from=fyb@163.com](mailto:from=fyb@163.com)  
set envelope\_from=yes

MSMTP  
创建~/.msmtprc和~/.msmtp.log,分别为配置和日志文件。  
vi .msmtprc  
account default  
host smtp.163.com

from [fyb@163.com](mailto:fyb@163.com)  
auth plain

user fyb

password 123456

logfile ~/.msmtp.log

由于password是明码，所以我们需要修改此文件的权限。  
chmod 600 .msmtprc

touch ~/.msmtp.log

查看SMTP服务器是否支持认证的TLS加密：

[oracle@oracle ~]$ msmtp --host=smtp.163.com --serverinfo  
SMTP server at smtp.163.com (m5-86.163.com [202.108.5.86]), port 25:  
    163.com Anti-spam GT for Coremail System (163com[20050206])  
Capabilities:  
    PIPELINING:  
        Support for command grouping for faster transmission  
    AUTH:  
        Supported authentication methods:  
        PLAIN LOGIN

163邮箱自身也要设置一下

首页<设置>--->POP3/SMTP/IMAP

按如下图将smtp服务启用



会得到一个第三方授权的密码，替换.msmtprc中的密码

到这里，你可以使用mutt来发送邮件了，我们测试一下。  
echo "test" |mutt -s "my\_first\_test" [fyb@gmail.com](mailto:fyb@gmail.com)

-s "subject"  
-c "carbon-copy"  
echo "test" |mutt -a dbms\_stats.txt -s my\_first\_test [fyb@gmail.com](mailto:fyb@gmail.com)

邮箱截图

1. 命令说明

Curl +”+网址+”+>\*\*.txt,将json返回到txt中

grep -Po '(?<=\"face\_id\"\:\ ").\*(?=\")' ./face.txt > ./face\_id.txt

上条命令含义是将face.txt中face\_id””和一个”之间的内容给face\_id.txt

face\_id=$(cat ./face\_id.txt) 将face\_id.txt中的所有内容给face\_id这个变量

Shell中基本上就含有这三个命令。关于php的相关命令可以参考最后所给的网址自行学习。

1. 一些写程序是遇到的问题及解决方案：
2. 执行一个脚本文件总报错还找不出问题：

可能是windows下编辑时换行符和linux下的换行符不一样，使linux编译时出错。

1. php向shell中传参数的问题：

这个问题可以定义超全局变量，不过并没有应用成功，最后是在执行shell函数内部定义变量，在执行shell语句后空格加上变量。不过在shell应用时需用$1,$2,$3表示传入的第1 2 3个变量。例：system ("sudo sh /var/www/ceshi/recognition.sh $YOUR\_API\_KEY $YOUR\_API\_SECRET $GROUP\_NAME")。

1. 有时候程序运行成功但找不到生成的文件的问题：

在php中执行shell文件时，当shell生成的文件的路径都是基于index.php所在路径。

**结论：**

**参考文献：**

**后记：关于face++删除人后添加同名人无法进入组的解决办法，**

**首先问题分析：我们向组里录入人时使用的是creat\_person，但是，当你删除这个人后再次录入，系统会认为人是不在组里的。**

**解决方法：解决这个问题要在creat\_person后，添加add\_person的语句，将人添加到组中。（问题完美解决）**

**之后完善了一下代码，添加了一个删除一个人的功能。其关键字的匹配是重点，首先把关键字择出来，放到txt中然后用grep匹配出全是字母的人名，之后FACE++提供的接口删除组内一个人。|（完活）**