**A 软件基本编程技能**

本部分意在考查应聘者基本的软件编写技能，请详细阅读题目要求，根据你的背景情况，选择2题或以上完成。

要求：

1) 编程语言优先用C++，其他语言也可

2) 优先在Linux平台实现，开发工具不限

3) 提供源代码、编译方法和必要的说明文档

4) 提供应用程序运行时的截图

5) 是Console 还是 GUI App请自定

6) 如果你愿意，可以自行扩充程序软件呈现的功能

7) 可以参考网上资源例子，但杜绝原本照抄，需要有自己的理解和实现

1.1 SDL, an excellent open source project, is successfully ported to many OS platform. <http://www.libsdl.org/>

Using SDL Thread API (http://wiki.libsdl.org/APIByCategory#Threads), implement two threads accessing a shared queue. One WRITE, the other READ, to demonstrate producer/consumer model.

SDL是一个开源项目， 利用SDL thread API, 实现生产者消费者模型。

1.2 利用开源或第三方库（比如mongoose, libghttp, libcurl, qt等），实现一个简单的http server 和http client交互演示：client通过POST向server提交数据，比如一张图片，然后client通过GET向server获取之前的提交数据。server端能够兼容curl命令或postman。具体交互内容的细节自行定义。

1.3 实现将一幅4K分辨率（3840\*2160）的YUV420SP(NV21)格式的图像转换为YUV422(YUYV)格式。如果有余力，可以使用CUDA编程实现。

(1) YUV格式参考：<https://www.jianshu.com/p/eb72a55b98aa>

(2) 4K YUV文件请自行准备，可通过jpg文件转换：ffmpeg -i sample.jpg -pix\_fmt nv21 sample\_nv21.yuv

(3) 查看YUV图像：ffplay -f rawvideo -video\_size 3840x2160 -pix\_fmt nv21 sample\_nv21.yuv

ffplay -f rawvideo -video\_size 3840x2160 -pix\_fmt yuyv422 sample\_yuyv422.yuv

其中ffmpeg/ffplay命令都在ffmpeg中，ubuntu下安装：sudo apt install ffmpeg

1.4 开发一个应用程序，实现对摄像头(笔记本自带摄像头或usb摄像头)的访问，并进行实时人脸检测和呈现，可以使用开源或免费的人脸SDK。可选功能：进一步添加人脸识别比对功能。

1.5 咖啡机程序。（题目描述见附1）

**B 综合测试**

请根据你的背景和项目经验, 可以概要地探讨一些解决的思路; 可能的话,也可以具体地给出软硬件实现的细节。 请根据你的情况选择两题或者全部都做。

2.1 智能嵌入式Linux产品因为其软件的迭代更新，升级需求也会相对频繁。请分别从系统层升级与应用层升级、本地升级与在线升级两个角度，调研并简要叙述相关的技术、方案与流程。

2.2 现在有些公司专业搞“抄板”, 把产品拆解,拿出 PCB,分析用到了哪些主芯片,从 Flash存储器读出你的 image, 然后完整复制出一个同样的产品。 你辛辛苦苦设计了一个电子产品,不想这么轻易就被盗版,请谈谈你有什么(软硬件结合的)方法来应对“硬件盗版”。

2.3 IPCamera 网络相机通常可以在局域网内直接访问视频流（常见的是RTSP）。如果IPC在局域网内，外网没有办法直接访问到IP地址，也就没法看到视频。如果想在外网访问到IPC设备查看视频，有哪些办法？

**附1：**

**咖啡机程序题**

X型咖啡机一次可以制作一壶咖啡。用户把过滤器放置在支架上，在其中装入研磨好的咖啡,然后把支架推入。接着，用户向水箱倒入足够的水并按下冲煮(Brew)按钮。加热器加热水一直加热到沸腾。不断产生的水蒸气压使水洒在咖啡粉末上,凝结的水滴通过过滤器流入到咖啡壶中。咖啡壶由一个保温盘进行长期保温，仅当壶中有咖啡时，保温盘才进行工作。如果在水还在向咖啡粉喷洒时从保温盘上拿走咖啡壶，水流就会停止，这样煮好的咖啡就不会溅在保温盘上。以下是需要监控的硬件设备。

加热器 (Boiler Heater)：加热可以开启和关闭。

保温器 (Warmer)：加热可以开启和关闭。

保温盘 (Warmer Plate)传感器：它有3个状态——warmerEmpty（咖啡壶不在保温盘上）、potEmpty（咖啡壶在保温盘上但没有咖啡）和potNotEmpty（咖啡壶在保温盘上且有没有咖啡）。

加热器 (Boiler)传感器：用来判断水箱是否有水。它有两个状态:　boilerEmpty、boilerNotEmpty。煮咖啡过程由boilerEmpty作为结束的节点。

冲煮按钮 (Brew Button)：这个瞬时按钮启动冲煮流程。用户开始煮咖啡时如果水箱无水，或者上一次煮咖啡过程没有结束，则不会重新开始煮咖啡。

指示灯 (Indicator)：冲煮开始后熄灭，直到当冲煮流程结束时再次亮起，表示咖啡已经煮好。

减压阀门 (Relief Valve)：在开启时可以降低加热器中的压力。压力降低会阻止水流向过滤器。该阀门可以开启和关闭。

X型专用咖啡机的硬件已经设计完成，硬件工程师为我们提供了低层的API如下。现在请使用底层API用C/C++编写一个咖啡机程序，并提供必要的流程图和文档说明。

namespace CoffeeMaker {

enum WarmerPlateStatus {

WARMER\_EMPTY, POT\_EMPTY, POT\_NOT\_EMPTY

};

enum BoilerStatus {

EMPTY, NOT\_EMPTY

};

enum BrewButtonStatus {

NOT\_PUSHED, PUSHED

};

enum BoilerHeaterState {

OFF, ON

};

enum WarmerState {

OFF, ON

};

enum IndicatorState {

OFF, ON

};

enum ReliefValveState {

CLOSED, OPEN

};

/\*

\* This function returns the status of the warmer-plate

\* sensor. This sensor detects the presence of the pot

\* and whether it has coffee in it.

\*/

WarmerPlateStatus GetWarmerPlateStatus();

/\*

\* This function returns the status of the boiler switch.

\* The boiler switch is a float switch that detects if

\* there is more than 1/2 cup of water in the boiler.

\*/

BoilerStatus GetBoilerStatus();

/\*

\* This function returns the status of the brew button.

\* The brew button is a momentary switch that remembers

\* its state. Each call to this function returns the

\* remembered state and then resets that state to

\* NOT\_PUSHED.

\*

\* Thus, even if this function is polled at a very slow

\* rate, it will still detect when the brew button is

\* pushed.

\*/

BrewButtonStatus GetBrewButtonStatus();

/\*

\* This function turns the heating element in the boiler

\* on or off.

\*/

void SetBoilerHeaterState(BoilerHeaterState s);

/\*

\* This function turns the heating element in the warmer

\* plate on or off.

\*/

void SetWarmerState(WarmerState s);

/\*

\* This function turns the indicator light on or off.

\* The indicator light should be turned on at the end

\* of the brewing cycle. It should be turned off when

\* the user presses the brew button.

\*/

void SetIndicatorState(IndicatorState s);

/\*

\* This function opens and closes the pressure-relief

\* valve. When this valve is closed, steam pressure in

\* the boiler will force hot water to spray out over

\* the coffee filter. When the valve is open, the steam

\* in the boiler escapes into the environment, and the

\* water in the boiler will not spray out over the filter.

\*/

void SetReliefValveState(ReliefValveState s);

}