第7次课

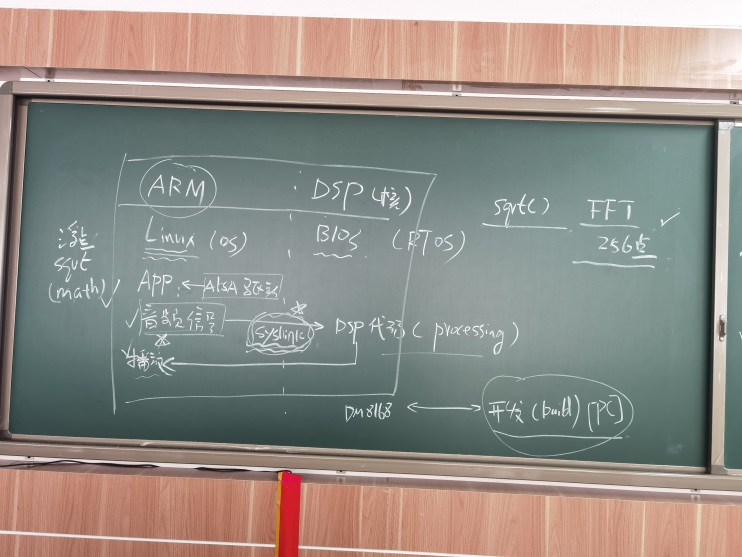
第2阶段：syslink组件实现DSP（异构）加速：

一．整体理解

1-20230905-代码结构框图.jpg ：目标板有ARM(主核)、DSP（从核）。音频处理主-从模式

ARM（Linux应用）完成音频捕获、处理调度、播放

DSP核完成具体的处理算法，DSP核RTOS



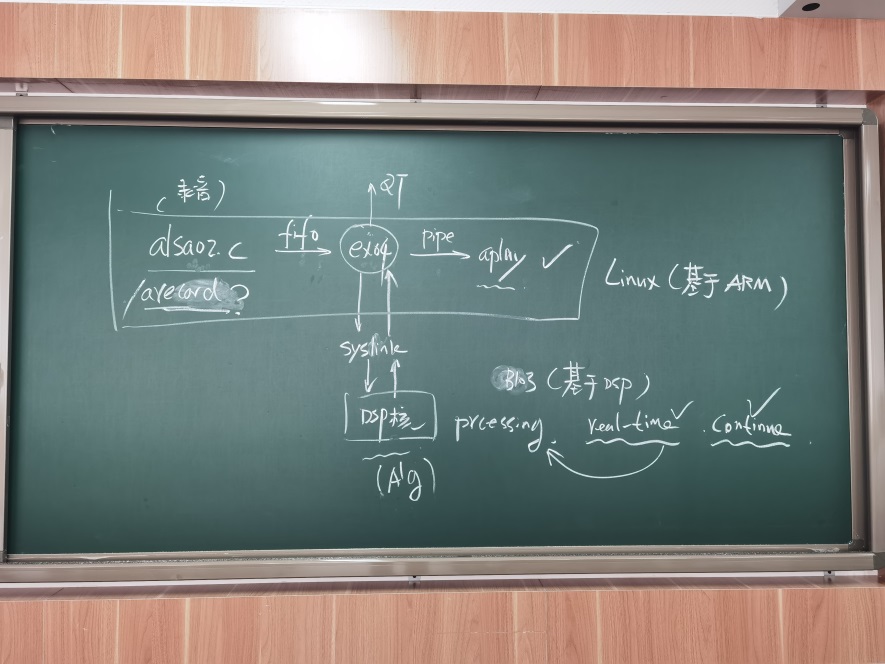
2-20230905-综合设计II框架思路框图.jpg

从完成目标任务的角度，对任务进行“拆分”！下下面这个框架图：

横向：Linux应用：录音、播放、处理

纵项：DSP核和ARM核直接的数据传递、消息同步! SysLink加速框架！

整合：将横向的处理，替换成纵项的syslink加速框架，实现DSP核的音频处理。



二、syslink框架学习

1. 如何运行syslink框架【用来实现DSP核加速处理的应用】例程 [在目板上运行]
2. bootargs参数 MEM=169M
3. 关闭启动时的3个进程
4. 启动例程./run.sh

./slaveloader startup DSP server\_dsp.xe674 🡪 启动DSP核，加载DSP固件

（固件server\_dsp.xe674，就是DSP核上的应用代码！）

./app\_host DSP -🡪 启动syslink的例程

./slaveloader shutdown DSP --》 关闭DSP核

1. 分析例程，找到一个最贴近我们工程的一个例程
2. 修改这个贴近的例程，实现我们的目标。

开发2个：DSP核固件server\_dsp.xe674，ARM核的Linux应用app\_host ！

参考文档：DM816x\_EZ\_Software\_Developers\_Guide.pdf 第5页syslink运行

第12页 介绍syslink

DM8168实验140718.pdf 第53页 syslink的运行，编译（参考！）

1. Syslink的例程的编译

关于新安装ubuntu的环境.txt 第69行- 文档结束 关于编译例程！