

数码相框设计文档

名词介绍:

tslib:是一个开源的程序;能够为触摸屏驱动获得的采样提供诸如滤波、去抖、校准等功能,通常作为触摸屏驱动的适配层,为上层的应用提供了一个统一的接口。

libjpeg:jpeg 图片处理相关开源库。

dma:不依赖 **cpu** 的高性能总线传输方式

tips:

1. 如果有些板子没有触摸屏,则在硬件那块使用按键,我们重新定义封装那边的接口,这样如果后面更换硬件,则只需要修改封装那一块的内容即可。否则,限制太大,不可扩展。
2. 共分为两个进程,输入进程和显示进程,且分别创建如下线程:
 - a) 输入进程可以分为如下线程:
 - 如果是触摸屏,则创建 **ts** 线程
 - 如果是按键,则创建按键线程
 - 主控线程收到数据(上面 **ts** 或者按键上报过来的数据)之后,发出 **socket**
 - b) 显示进程包括如下线程:
 - 接收 **socket** 消息线程.
 - 分别创建当前和上下左右,放大,缩小等线程(用来准备好相关图片,分别对应放大,缩小,左移,右移,上一幅图片,下一幅图片,且都是准备好的图片数据).
 - 把上面的图片通过 **libjpeg** 库解压出来存入 **framebuffer** 里面.
 - 上面的五块存储都需要开辟一块内存存储,此处使用 **mmap** 内存映射的方式进行.
 - 还有一个主控线程,根据 **socket** 线程收到的事件,决定显示上下左右哪一幅,然后我们从相应的内存中,通过 **dma** 把图片数据拷贝到显存中.
3. 为什么要使用进程和线程,因为需要保持各个功能的独立,且独立的功能进程中又有一些共享的资源.
4. 我们还需要书写一个驱动,包括提供五块内存和 **dma** 的操作,使用 **mmap** 分配这五块内存.

