2022/2/23

使用以下命令

ffmpeg -i stereo.wav -filter\_complex "[0:a]channelsplit=channel\_layout=stereo[left][right]" -map "[left]" left.wav -map "[right]" right.wav

将双声道的立体声文件分割为左右两声道后分割后的文件与源文件没有太大区别，不存在清晰的只有左耳有声音或者右耳有声音的声道区分表现，不清楚是源文件的问题还是分割的问题还是收音设备的问题。

尝试用单例模式把cmd控制逻辑跟widget解耦合，把他单独分出来，但是在这种情况下信号与槽函数的连接我有点没底，因为对底层机制了解不深入，不知道在此模式下能否触发槽函数。

2022/2/24

继续重构项目，完成ProcessControl模块的编写。理了下内部类（嵌套类）与外部类的关系，以及如何用内部类访问外部类的静态成员。

信号与槽函数的连接应放在ProcessControl类的实例化之后，才能连接到具体地址, 我这边把连接的代码放在了getInstance（）静态函数里，就避免必须把饿汉式写成懒汉式（在最开始初始化的时候实例化并连接信号与槽。

为了降低UI与processControl模块的耦合度，自定义信号接力传递来自process的输出与错误信息再在ui界面中显示。

2022/3/9

加入tcpserver模块，用于建立主机（服务器）与移动设备（客户端）的可靠连接用以传输音频数据。遇到些困难，比如如果在一个界面中打开另一个界面，开始包含进另一个界面的头文件后在代码中定义界面实体总是报错，后面搞清楚Ui::tcpserver 与 tcpserver后成功通过编译，但是界面闪过一瞬间后就没了。在意识到界面可能（待后面深入学习查证）相当于另外一个新的在跑循环的线程，而原来界面的线程不会被阻塞在那而是继续执行，故当原来线程执行完函数之后新的界面由于是局部变量于是同样被销毁了，销毁的同时调用析构函数将new出来的ui界面实体给delete掉了，于是我改用私有成员变量指向堆区动态new出来的新ui界面的方式，在需要的时候show（）显示界面，不需要（或是执行特定动作后）hide（）隐藏或者退出界面（这里经我自己验证，我猜测点击右上角的关闭按钮后应该只是关闭了画面渲染，实体依然存在于内存中没有被delete掉），在主界面析构的同时一块删除这个二级界面。

建立移动端工程与类图，利用qt跨平台特性尝试先直接将已经完成的tcp连接的客户端编译到Android平台。

2022/3/11

成功将client部署到手机，发现几个bug

[TODO]:

1. server的本机ip自动获取错误，获取的是VMware的虚拟ip地址，想办法获取到真实ip
2. Client与server通信的时候中文无法解码，应该是因为我用的是原始的tolocal8bit做转换，明天换成QTextstream。

2022/3/19

不应该，或者没必要用QTextStream，现在需要的仅仅是传输音频文件上去，以及传输一些譬如播放暂停，调低调高音量之类的简单指令。

不对，还是得用QTextStream，然后把QDataStream结合起来用，可以考虑采用tlv的传参格式，需要设计一个小的指令系统。先验证音频传输能否成功把，然后把手机的音频播放调用起来。