CP16发送软件部分

文本-波形转换库

使用C++编写

接收文本数据，转换为波形数组

使用C++/CLI封装成.NET类库

CP16 GUI

使用C#编写

与用户交互，根据用户控制规划重复发送等功能，调用文本-波形转换库并将波形送给操作系统

操作系统

接收波形送往电台设备播放

虽然我们已经有了根据待发送字符串生成波形数值的C++代码，为了使用户能够方便的利用CP16编码进行通讯，还必须设计一个简洁易懂的UI界面，方便用户与软硬件进行交互。

1. 语言和框架选取

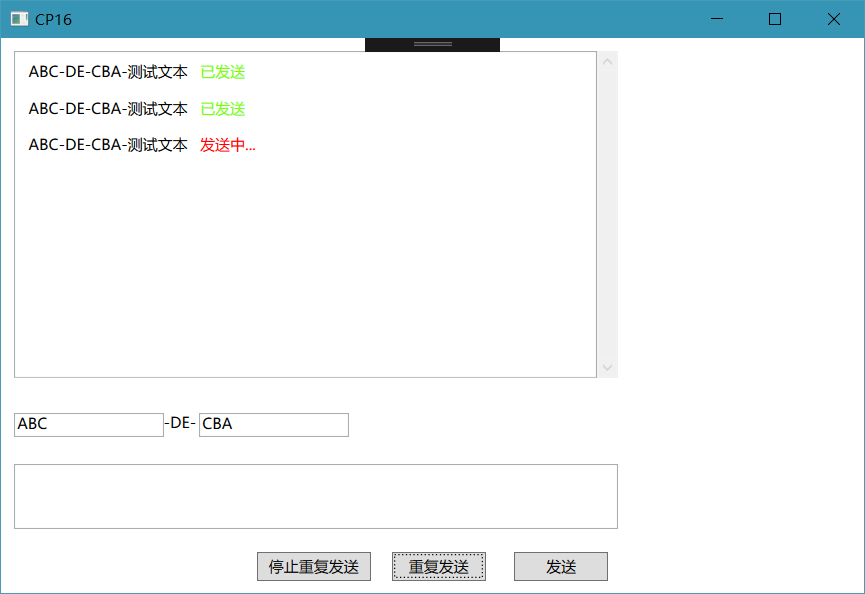
在GUI编程语言和框架方面，我们选择了使用C#语言和WPF用户界面框架。选择C#的原因是作为一种拥有自动垃圾回收机制和大量内置库函数（例如线程操作相关函数）的语言，C#使用起来非常方便，同时作为.NET语言家族的一员，C#可以很简单的调用其他.NET语言生成的类库；我们只需要将使用C++编写的文本-波形转换库包装为同为.NET语言的C++/CLI类，即可生成便于调用的dll库。选择WPF框架则是因为使用该框架制作GUI程序非常方便，同时生成的程序不会像传统的MFC或WinForm架构下生成的程序那样，在高分辨率屏幕下显示模糊。

1. 程序结构设计

程序主要分为两个线程：一个是程序主线程，用于控制UI元素的显示；一个是后台工作线程，用于从UI获取用户输入的文本，使用文本-波形转换库转换为波形数据，然后使用系统音频播放API进行播放，同时通知主线程更新UI的相关元素。

在进行音频播放相关代码编写时我们遇到了一个问题，C#内置库函数中，从内存数据流播放音频的函数只能播放wav文件格式的数据，而不能播放无格式的PCM数据。因此我们必须现在内存中手动构建wav文件头，并将浮点PCM数据转换为可用的16位整型数据。

1. UI界面设计和效果



如图所示，最终实现的UI界面类似于QQ等即时聊天工具，允许用户输入任意待发送文本并选择单次发送或重复发送功能。另外由于CP16文本发送较慢，还添加了使用不同颜色的提示文字来提示用户文本是否已经发送完毕。同时在老师的建议下，还添加了无线电呼号的设置，能够自动在每条发送的文本前添加相应的呼号文本。

界面右侧的空白部分本来是预留给UI界面功能扩展（例如增加发送内容瀑布图显示等），不过由于时间原因最终没有用到。