



广州大学华软软件学院
South China Institute of Software Engineering.GU

毕业论文(设计)

课题名称 基于Qt的语音聊天的系统
设计与实现

系 别 电子系

专业(方向) 电子信息工程(嵌入式系统软件开发)

班 级 14级统二班

学生姓名 叶宗文

指导教师 陈虹

完成日期 2018年4月10日

教务处 制

毕业论文（设计）真实性诚信保证书

本人所撰写的毕业论文（设计）是在老师指导下独立完成，没有弄虚作假，没有抄袭或拷贝他人研究成果。我郑重承诺：文责自负。

签 名：叶宗文

2018 年 4 月 10 日

广州大学华软软件学院

本科毕业设计任务书

设计题目 基于 Qt 的语音聊天系统的设计与实现

系 别 电子系

专 业 电子信息工程(嵌入式系统软件开发)

班 级 14 嵌统二班

学 号 1440913170

学生姓名 叶宗文

指导教师 陈 虹

下发时间： 2017 年 9 月 25 日

毕业设计须知

1、认真学习和执行广州大学华软软件学院学生毕业论文（设计）工作管理规程；

2、努力学习、勤于实践、勇于创新，保质保量地完成任务书规定的任务；

3、遵守纪律，保证出勤，因事、因病离岗，应事先向指导教师请假，否则作为缺席处理。凡随机抽查三次不到，总分降低 10 分。累计缺席时间达到全过程 1 / 4 者，取消答辩资格，成绩按不及格处理；

4、独立完成规定的工作任务，不弄虚作假，不抄袭和拷贝别人的工作内容。否则毕业设计成绩按不及格处理；

5、毕业设计必须符合《广州大学华软软件学院普通本科生毕业论文（设计）规范化要求》，否则不能取得参加答辩的资格；

6、实验时，爱护仪器设备，节约材料，严格遵守操作规程及实验室有关制度。

7、妥善保存《广州大学华软软件学院本科毕业设计任务书》。

8、定期打扫卫生，保持良好的学习和工作环境。

9、毕业设计成果、资料按规定要求装订好后交指导教师。凡涉及到国家机密、知识产权、技术专利、商业利益的成果，学生不得擅自带离学校。如需发表，必须在保守国家秘密的前提下，经指导教师推荐和院领导批准。

课题名称	基于 Qt 的语音聊天系统的设计与实现
完成日期:	2018 年 4 月 10 日
<p>一、题目来源及原始数据资料:</p> <p>本课题来源于设计一个语音聊天系统，能在局域网内，利用 FFMPGE 实现点对点的实时聊天。</p> <p>放眼看看身边，我们身边的人所接触的音频聊天的应用，大抵是 QQ，微信。但是它们都有一个共同点，也是它们的不足之处。就是它们只能在广域网内进行语音聊天，并不具备在局域网语音聊天的能力。音频的传输具备一定的传输容错率，可以忍受部分由于网络的状况导致的丢包问题，这比文字聊天好很多，文件聊天如果是丢包了，可能会导致信息让人无法去理解。传统的文字聊天同样一句话不同的语气能表达不相同的意思，所以采用文字聊天，是很难判断对方的语气。同时比较容易造成误解，还有文字聊天相对来说慢，不如语音聊天快。</p> <p>FFMPEG 是一套可以用来记录、转换数字音频、视频，并能将其转化为流的强大的音视频流媒体处理工具。它提供了录制、编码、解码、转化格式以及流化音视频的解决方案。</p> <p>基于 qt 的语音聊天系统的设计与实现正是由于这种背景之下而出现的以语音聊天为基础的一种人与人的沟通方式。方便了局域网的人们实时的交流，使人与人的亲切感增加，丰富了人们的日常生活方式。</p>	
<p>二、毕业设计要求:</p> <p>本设计通过 UDP 协议进行组播，每一个在线的用户接收到其数据后，分析其数据，从而更新用户在线列表。</p> <p>音频的采集使用了 DSHOW 来打开音频设备，然后开始采集音频，采集后的音频数据使用 FFMPEG 对其进行 AAC 编码的处理。最后会通过 UDP 协议来把音频发送到目的地。接受端收到音频，然后把音频数据放进一个缓冲区里面，经过解码之后，最终传送到音频设备播放出来。</p>	

三、进度安排、应完成的工作量:

- 1、第 1-4 周: 完成题目资料的搜集, 整理, 了解 FFPEG 的基础知识, 以及 SDL 的播放原理。
- 2、第 5-8 周: 系统汇总资料, 分析基于 Qt 的语音聊天系统功能需求, 系统整体组成框图和各系统模块, 准备中期检查。
- 3、第 9-12 周: 设计软件并且实现, 然后进行系统的测试, 修改规划方案有错误的地方, 并且撰写论文初稿。
- 4、第 13-16 周: 对语音聊天系统进行最后修正与分析及系统的功能完善, 完成毕业设计论文, 准备毕业设计答辩资料准备毕业答辩。

四、主要参考文献

- [1] 张珏. 一种基于 P2P 结构的即时通讯系统的设计与实现[D]. 西安: 西安电子科技大学软件工程 2010.25-26
- [2] 杨婷婷. 宽带中国开启美丽生活[J].现代工业经济和信息化, 2014(Z1).74-75
- [3] 江霏霏. 某型火炮营级指挥控制模拟系统的若干分系统设计[D]. 南京: 南京理工大学, 2012.61-62
- [4] 姚远. 视频监控图像后处理算法的研究与实现[D]. 北京: 北京邮电大学, 2010.48-49
- [5] 冀胜利. 基于嵌入式 Linux 的网络视频传输研究和实现[D]. 重庆: 重庆理工大学, 2009.33-34
- [6] 陈熙. 基于 Android 移动平台视频监控终端的设计与实现[D]. 成都: 电子科技大学, 2013.23-24
- [7] 余柳冰. 基于嵌入式 Linux 的电脑横机控制软件开发[D]. 杭州电子科技大学, 2011.40-41
- [8] 余东. 多描述视频编码研究[D]. 太原: 太原科技大学, 2013.50-51

指导教师 (签名):

谭石坚

系 (教研室) 主任 (签名):

夏国清

广州大学华软软件学院
本科毕业论文（设计）成绩单

姓名	叶崇文	学号	1440913170	专业（方向）	电子信息工程(嵌入式系统软件开发)
题目					
指导教师评语 (占论文总成绩的60%)	<p>该生在毕业设计期间,学习态度认真,毕业设计结构合理,思路清楚,论文符合学院要求,同意参加答辩。</p> <p>成绩: 52</p> <p style="text-align: right;">陈虹 签名: 谭石坚 2018年4月26日</p>				
评阅教师评语 (占论文总成绩的20%)	<p>该生实现语音聊天系统的基本功能,论文格式正确,条理清楚,同意参加答辩。</p> <p>成绩: 17.5</p> <p style="text-align: right;">签名: 徐礼国 2018年4月 日</p>				
答辩组评语 (占论文总成绩的20%)	<p>作品功能要求基本实现,实物制作质量尚可。论文内容较完整,结构较合理,条理较清晰。答辩时讲述的系统性、逻辑性一般,语言表达尚可,回答问题基本正确。</p> <p>成绩: 15</p> <p style="text-align: right;">答辩组成员: 董立国 何燕利 年 月 日</p>				
总成绩	<p>84.5 良</p> <p style="text-align: right;">系主任签名: 廖国洪 系(章) 18年5月4日</p>				

摘要 针对在广域网内实时性大量传输音频，有可能会出现延时、丢包的这一问题，提出了在局域网内基于 Qt 的实时语音聊天系统。在局域网内通过 UDP 传输协议实现了音频的实时传输，使用了 FFmpeg 对音频进行处理。系统里面开启了两个线程，一个用 DSHOW 来打开音频设备，接着进行音频数据的收集，紧接着把得到的 PCM 数据,用 AAC 压缩算法来编码，最后把音频数据传输到目的地。另外一个线程会接收到对端发过来的音频数据，然后解码成 PCM 数据，最后通过 SDL 播放出来。实验结果最终实现了语音的实时聊天。

关键字 FFmpeg；SDL；多线程

Abstract In view of the large real-time transmission of audio in wan, the problem of delay and packet loss is possible, and a real-time voice chat system based on Qt is proposed in the LAN. The real-time transmission of audio is realized through UDP transfer protocol in LAN, and FFMPEG is used for processing audio. System opens the two threads, a use DSHOW to open the audio equipment, then carries on the audio data collection, and then put the data, with AAC compression algorithm to encode, finally, the audio data transmission to the destination. Another thread will receive the audio data from the terminal, then decode it into PCM data, and finally play it through SDL. The final result is a real-time voice chat.

Keywords FFMPEG; SDL; Multithreading

目 录

- 1. 前 言..... 1
 - 1.1 研究背景与现状..... 1
 - 1.2 选题的目的与意义..... 1
 - 1.3 选题采用的研究方法..... 2
- 2. 系统分析与设计..... 3
 - 2.1 市场分析..... 3
 - 2.2 可行性分析..... 3
 - 2.3 系统的总体设计与实现..... 4
- 3. 系统的详细设计与实现..... 6
 - 3.1 登录模块..... 6
 - 3.2 用户列表显示模块..... 7
 - 3.3 语音聊天模块..... 9
 - 3.4 音频的采集模块..... 10
 - 3.5 音频的播放模块..... 11
- 4. 系统测试..... 15
 - 4.1 用户列表显示模块测试..... 15
 - 4.2 网络流量测试..... 15
 - 4.3 音频采集模块测试..... 17
 - 4.4 音频播放模块测试..... 17
 - 4.5 语音聊天测试..... 18
 - 4.6 测试结果与分析..... 18
- 5. 总 结..... 20
 - 5.1 总结..... 20
 - 5.2 期望..... 20
- 参考文献..... 21
- 致 谢..... 22

1.前 言

1.1 研究背景与现状

宽带网络是未来整个经济社会信息传播的主要载体，它改变的不仅是信息传播的方式，对人的生活方式和行为习惯也会带来翻天覆地的变化^[2]。高速传输接口 FDDI 的成熟与网络的速度高速发展,使得各种各样的数据在网络上快速传输不再是梦想了。在这个网络时代中，网络上兴起了很多基于多媒体技术的服务,比如远程会议，视频监控，在线教育，直播等等，语音聊天也是其中之一。局域网内的实时语音聊天有很多特点，相对于电话来说，它不需要主人支付费用。网络产品的高速发展和相关技术的不断成熟，以 tcp/ip 协议为核心的网络设备不断涌现和发展导致通信带宽不断增加，人们上网可以享受飞一般的网速。在线看高清电视不在话下。同时有些应用为了更好地用户体验，往往会把语音聊天做为他们所作的产品的一个功能。

1.2 选题的目的与意义

放眼看看身边，我们身边的人所接触的音频聊天的应用，大抵是 QQ，微信。但是它们都有一个共同点，也是它们的不足之处。就是它们只能在广域网内进行语音聊天，并不具备在局域网语音聊天的能力。音频的传输具备一定的传输容错率，可以忍受部分由于网络的状况导致的丢包问题，这比文字聊天好很多，文件聊天如果是丢包了，可能会导致信息让人无法去理解。传统的文字聊天同样一句话不同的语气能表达不相同的意思，所以采用文字聊天，是很难判断对方的语气。同时比较容易造成误解，还有文字聊天相对来说慢，不如语音聊天快。基于 qt 的语音聊天系统的设计与实现正是由于这种背景之下而出现的以语音聊天为基础的一种人与人的沟通方式。方便了局域网的人们实时的交流，使人与人的亲切感增加，丰富了人们的日常生活方式。

1.3 选题采用的研究方法

本设计使用了 UDP 组播实现了在线用户列表的实时更新。每一位用户上线都会组播自己的登陆信息。其他用户收到登陆信息后，然后更新在线登陆用户列表。音频的采集是采用 DSHOW 来采集的，然后经过编码之后再通过 UDP 发送出去。本设计用 FFMPEG 这个开源库对音频进行格式的转化、编码、解码。最后把解码后的 PCM 数据使用了 SDL 进行播放。

2.系统分析与设计

2.1 市场分析

在如今这个信息化时代，信息技术已经渗透到了各行各业，信息化技术也越来越成熟。随着网络的日渐发展，网络即时通讯工具已经成为人们不可缺少的一部分。在中国，QQ 和微信占据了即时通讯这个市场的半壁江山。但在这个市场上，文字聊天还是占着重要的地位。当前的语音通讯主要还是以电话的形式为主，成本比较高。当然了，也有相当多的人是使用 QQ 和微信里面的语音聊天功能的。但是它们有一个局限性，就是当网络质量不太理想的情况下，可能会导致语音延时，从而导致用户体验差，通讯质量很难满足人们的需求。

经过具体的市场调查，基于局域网的语音聊天，效果逼真，成本低廉是未来的主流。潜在着巨大的市场，有必要在音频这个领域深入研究。在企业内部，为了语音聊天信息的安全性，对局域网的音频聊天是有一定的需求的。再者，在中国，虽然光纤都已经慢慢普及到农村了，手机的网络从 3G 变为 4G，但是并不能排除网络环境不好的偏僻地带。这个时候像 QQ、微信这些依赖于良好网络环境的应用就没有多大的意义了。这个时候，局域网的语音聊天的意义将彰显出来了。

2.2 可行性分析

（1）经济可行性分析

本项目是由个人设计，作为本人的毕业设计，投入费用可忽略不计，主要开支是个人的日常生活费用，因此项目开发计划可行。

（2）技术可行性分析

本项目软件是在 Qt 框架下进行开发的，Qt 拥有多种窗口部件和丰富的 API 接口，为本设计提供了很多的便利，同时 Qt 可移植性强，所以本设计也具备一定的可移植性。本设计需要熟练掌握 C 和 C++ 语言，主要运用到的 SOCKET 网络编程，Qt 的信号和槽，AAC 的编码，AAC 的解码，SDL 播放音频等技术。

一般来说。即时通讯的系统主要有两种设计模式：一种是传统的 C/S 模式，另外一种 P2P 模式。C/S 模式的基本思想是把集中在一起的应用划分成为不同的两部分，采用 C/S 模式必须在因特网上设置拥有强大处理能力和大宽带的高性能计算机，再将大量的数据存放到上面^[1]。随着用户量的增加，服务器的负担会越来越让人担心。一旦服务器奔溃了，那么与服务器有着关系的客户端就会收到很大的影响。而 P2P 架构就不一样了。它不依赖与服务器，具有高容错，耐攻击的特性。而且 P2P 网络中，由于信息的传输是不需要经过中间节点的，导致用户的信息不容易被泄露出去。P2P 传输数据不需要经过服务器，直接传输，因此它的传输速度相对较快。

现在的局域网, 可达 2M 左右的带宽。当音频数据使用 16bit 存储，双通道，采样率为 44100 的话，一秒中收集的音频也就大约 172Kb，如果再经过 AAC 压缩的话那么它的体积大约 10kb。所以局域网为音频的传输创造了非常良好的环境。所以本设计的软件设计从理论上是可行的。

2.3 系统的总体设计与实现

本设计对音频的采集、编码、解码、播放这些技术进行了一定的研究, 实现了在局域网内语音的实时聊天的功能。它拥有着登陆模块、用户列表显示模块、音频采集模块、音频播放模块。

本设计需要在登陆之后方可进行语音聊天。由于每一个用户的信息是保存在 MySQL 数据库里面的，所以在登陆界面，需要对数据库进行操作，从数据库获取要登陆的用户的密码，然后与用户输入的密码做比较，倘若相同，则允许用户进行登陆操作。

在显示用户在线模块，会把用户在登陆界面输入登录的信息，然后通过 UDP 协议进行组播，每一个在线的用户接收到其数据后，分析其数据，从而更新用户在线列表。

在音频传输协议的选择上，因为 TCP 协议在传输的过程中要使用三次握手、重传确认等手段来保证数据传输的可靠性。使用 TCP 协议会有很大的时延，因此不适合对实时性要求较高的应用^[5]。所以为了音频更快地传输，所以选择了 UDP 协议进行音频的传输。

音频的采集使用了 DSHOW 来打开音频设备，然后开始采集音频，采集后的音频数据会采用 FFMPEG 对其进行 AAC 编码的处理。最后会通过 UDP 协议来把音频发送到目的地。接受端收到音频，然后把音频数据放进一个缓冲区里面，经过解码之后，最终传送到音频设备播放出来。

如图 2-1 所示，当语音连接建立之后，客户端 A 会向客户端 B 发送本机采集的经过 AAC 编码的音频数据，与此同时客户端 A 也会播放从客户端 B 发送过来的音频数据。客户端 B 也会向客户端 A 发送本机采集的经过 AAC 编码的音频数据，与此同时客户端 B 也会播放从客户端 A 发送过来的音频数据。从图 3-2 也可以清晰地看到。出于客户端之间音频的传输不需要服务器的中转，因此在成本方面，还是传输的速度方面，都可以充分体现了 P2P 架构的优势。

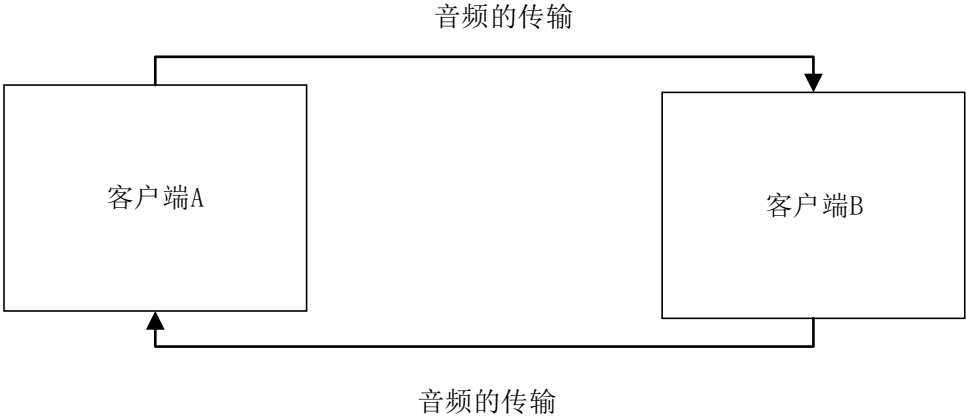


图 2-1 语音聊天系统的系统架构

综上所述，本设计主要利用了 UDP 广播实现了在线用户列表更新，在 windows 下利用 DSHOW 进行音频的采集，接着使用 UDP 协议进行音频的传输，然后接收端通过 SDL 进行音频的播放，最终实现了音频的实时播放。

3.系统的详细设计与实现

3.1 登录模块

窗口 loginWindow 的两个 QLineEdit 为用户提供了输入用户名跟密码的能力。它的界面如图 3-1 所示。

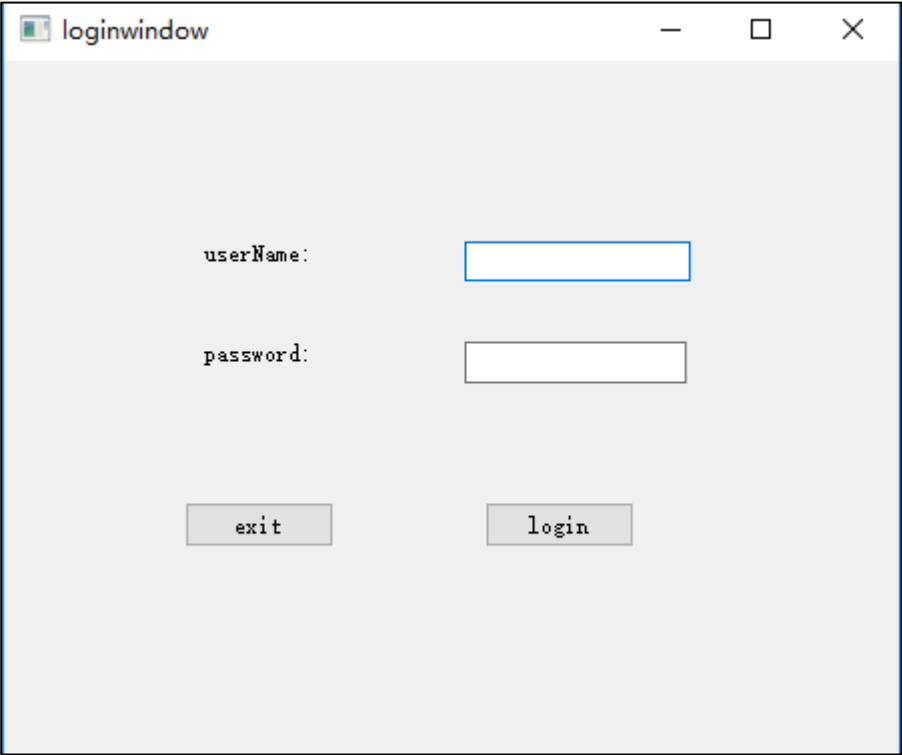


图 3-1 longinWindow 界面图

在 longinWindow 新建之际，会对 Msql 进行连接的操作，主要是新建一个 QSqlDatabase 类型的对象，然后对其设置用户连接的 Mysql 的 ip 地址，要连接的用户名和密码。当 Mysql 连接失败的时候会出现一个弹窗，用来告诉用户连接数据库时出现的问题。用户就可以对弹窗的信息进行分析，进而定位问题所在。上述的弹窗是使用 QMessageBox 来实现的。用它可以很好地把程序运行的内部信息反馈给每一个用户。

连接 Mysql 成功后，倘若用户有登录的意愿，输入的用户名和密码都是正确的话，那么，在点击名为 login 的按钮之后，按钮会发送一个 clicked 的信号，然后相对应的槽函数会执行，在这个槽函数里面会有个根据用户输入的用

户名去查询密码的 Mysql 操作。查询道的密码不能为空，然后跟之前用户输入的密码做比较，假如是无差异的，在跳转下一个窗口 userListDialog。

json 是一种轻便量级的数据交换格式，具有非常良好的可读性和良好的便于快速编写的特性。它对象表示为键值对，花括号保存对象，数据由分号分隔。现实设计中，大部分的服务器跟客户端都是通过 JSON 进行数据传输的。在 Qt 中，已经为开发者提供了对 JSON 进行操作的 API。

本设计设计发送或者接收的数据都是 json 数据。第一个 key (status)，代表用户的状态。上线其 value 为 login，下线其 value 为 exit。第二个 key (userName)，代表着用户名，第三个 key (ipAddress) 代表着用户的 IP 地址。

当用户发出一个登陆的行为，那么就会广播一条其第一个 key 为 login 的数据出去，收到此数据的其他用户会解析 json 数据，然后再更新用户在线状态。

在跳转之前，需要为输入的用户名、ip 地址和用户的状态 status 创建 Json 数据，然后传给 userListDialog 对象。在这里获取 ip 地址的操作，主要是 QNetworkInterface 来获取。QNetworkInterface 表示了当前程序正在运行时与主机绑定的一个网络接口。每个网络接口可能包含 0 个或多个 IP 地址，每个 IP 地址都可选择性地与一个子网掩码和/或一个广播地址相关联。这样的列表可以通过 addressEntries() 方法获得。当子网掩码或者广播地址不必要时，可以使用 allAddresses() 便捷函数来仅仅获得 IP 地址。

3.2 用户列表显示模块

此模块主要实现了用户在线状态的实时更新功能，窗口是 userListDialog。其界面如图 4-2 所示。

userListDialog 窗口在新建对象之初，父窗口——loginWindow 为它传入了一个包括用户名、ip 地址和用户状态 status 的 json 数据。然后再把这个 json 数据广播出去。

UDP 是无连接的不可靠的传输协议，可以用在可靠传输不是很重要的情况下使用。它和 QTcpSocket 最大的区别也就是，发送数据之前不需要建立连接。这里需要实现 udp 的广播功能，所以 loginWindow 的 myUdpSocket 通过 bind() 需要绑定一个空闲的端口。然后在发送数据的时候在 myUdpSocket 的

writeDatagram 方法内传入 QHostAddress::Broadcast,它是一个广播地址来的,其值为 255.255.255.255。Qt 提供 QListWidget 类列表框控件用来加载并显示多个列表项。QListWidgetItem 类就是列表项类。这里使用了 QListWidget 来显示用户在线列表。

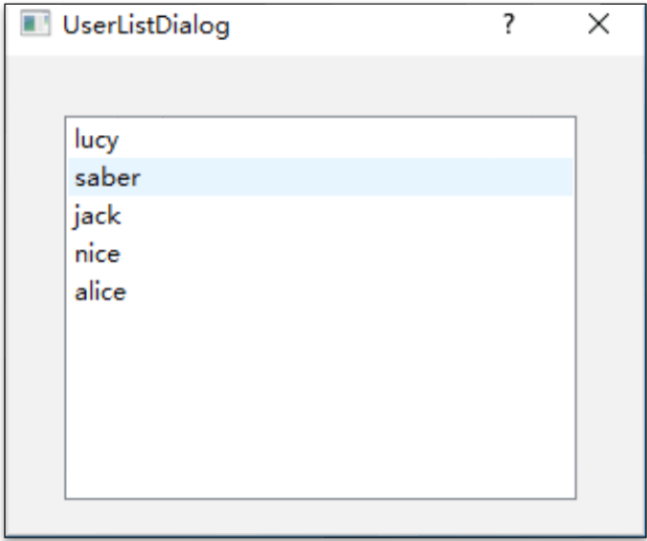


图 3-2 userListDialog 界面图

这里定义了两个类型为 QMap 的成员变量,分别是 onlineMap 和 chatMap。onlineMap 的作用是记录在线用户的信息。chatMap 的作用是记录有意向进行语音聊天的用户信息。在线好友的存储图如图 3-3 所示。

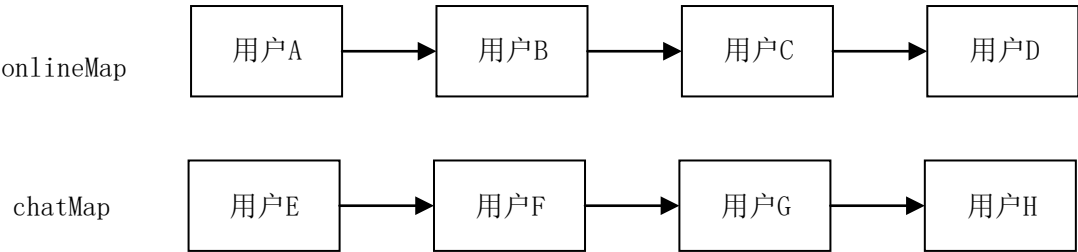


图 3-3 用户信息存储图

在同一个局域网内,倘若用户 A 上线了,A 系统的会发送一个广播,其数据类似{"status":"login","userName":"lucy","ipAddress":"172.16.120.62"}。然后用户 A 和其他在线用户的 myUdpSocket 会发出一个 redyRead 的信号,然后执行相应的槽函数,在这个槽函数里面会对接受的数据进行 json 的解析,倘若 key 是 status,其 value 为 login 的时候,就表明是一个登录的操作。然后把 key 为 userName,ipAddress 的数据放进类型为 QMap 的 onlineMap 里面。

接着新建一个 QListWidgetItem 的对象, 设置其显示的文本为登录用户的用户名, 最后把它添加到 listWidget 里面。仅接着每个收到数据的用户把自己的包括 status、userName、ipAddress 的 json 数据广播出去, 这样就可以实现局域网所有的用户列表同步了。

在同一个局域网内当用户 A 发出了退出的行为。那么其他在线用户会收到类似于这样的数据 {"status": "exit", "userName": "lucy", "ipAddress": "172.16.120.62"}。然后会在 listWidget 和 onlineMap 删除对应的用户信息。在同一个局域网内, 当用户 A 想跟用户 B 进行语音聊天, 那么会发送类似这样的数据 {"status": "chat", "userName": "lucy", "ipAddress": "172.16.120.62"}。然后就会跳转到 chatDialog 的界面等待对方连接。用户 B 收到数据, 然后解析, 倘若 key 为 status 的 value 是 chat 的话, 用户 A 的信息就会保存在 chatMap 里面, 然后在用户 B 里面的用户 A 列表, 其颜色设置为青色, 预示着是一个企图跟用户 B 进行语音聊天的用户。用户 B 倘若要对此行为做出回应的话, 可以点击一个用户 A 所在的列表, 就可以跳转在 chatDialog, 进行语音聊天了。在跳转之后需要对 chatMap 进行清空的操作。

3.3 语音聊天模块



图 3-4 chatDialog 界面图

在 chatDialog 窗口里, 在这个对象的构造方法会传入将要通讯的 ip 地址。这个 chatDialog 界面相对比较简陋, 它的界面如图 3-4 所示。当用户 A 和用户

B 都点击其名为 chatPushButton 的按钮，会发送一个 clicked 的信号，运行相应的槽函数，槽函数会创建两个线程，一个是 recordThread, 专门用于音频的采集。另外一个为 playThread，专门用于对端传过来的音频播放。这样子，就可以实现语音的聊天了。这个时候 chatPushButton 的 text 会变为 cancel。用户如果再按一次的 chatPushButton，会退出语音聊天，返回到用户列表的界面。

3.4 音频的采集模块

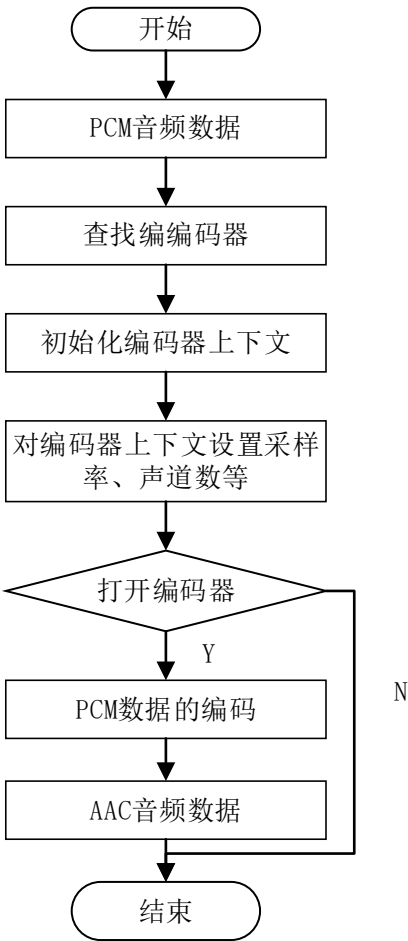


图 3-5 AAC 编码流程图

FFMPEG 是一套可以用来记录、转换数字音频、视频，并能将其转化为流的强大的音视频流媒体处理工具^[3]。采用 LGPL 或 GPL 许可证。它提供了录制、编码、解码、转化格式以及流化音视频的解决方案。它包含了非常先进的音频/视频编解码库 libavcodec^[4]。同时 FFMPEG 也具有跨平台的特点。

在 windows 可以通过 dshow 来打开音频输入设备，然后可以捕获麦克风的音

频流。得到的音频流不能立刻压缩成 aac 格式的音频。因为 aac 编码的话，前提是把获得的音频流，解码成音频帧，再通过一个过滤器把音频帧的大小改为 1024。原因是 MPEG2 AAC 规定 1024 个 sample 数据为一个音频帧。这里面过滤的操作可以理解为一个缓冲区，它存储着大量的音频采样数据。而我们需要的是每一次去那个缓冲区取数据的时候，取 1024 个音频采样数据。

AAC 是 Advanced Audio Coding 的缩写，跟 mp3 一样，是一种专门为音频数据压缩的一种文件格式。它的编码采用了新的算法，更加高效，具有更高的“性价比”。使用 AAC 进行编码，体积比较小，同时音质质量相对比较好^[6]。相对于 mp3，AAC 格式的音质更佳，文件更小。AAC 属于有损压缩的格式，所以在音质方面，无法跟 APE、FLAC 这些无损格式相比。经过处理的 PCM 数据，然后进行 AAC 的编码，其流程图如图 3-5 所示。最终编码后的数据通过 UDP 协议发送出去。

3.5 音频的播放模块

在音频的播放模块，使用了 SDL 对 udp 音频流进行播放。

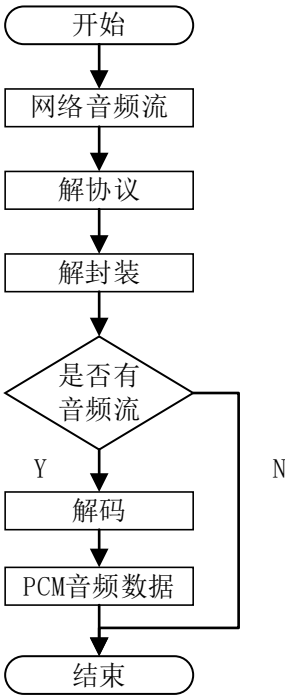


图 3-6 网络音频流转 PCM 流程图

SDL (Simple DirectMedia Layer) 是一套开放源代码的跨平台多媒体开发

库，使用 C 语言写成。SDL 提供了数种控制图像、声音、输出入的函数，让开发者只要用相同或是相似的代码就可以开发出跨多个平台（Linux、Windows、Mac OS X 等）的应用软件。目前 SDL 多用于开发游戏、模拟器、媒体播放器等多媒体应用领域。实际上 SDL 本身并不提供视音频播放的功能，它只是封装了视音频播放的底层 API。在 Windows 平台下，SDL 封装了 Direct3D 这类的 API 用于播放视频；封装了 DirectSound 这类的 API 用于播放音频。

要播放的是本地的地址，端口是 8888。这个端口是为人设定的，并非唯一。利用 UDP 协议传输过来的音频，不能直接用于播放。需要经过解协议、音频解码这些步骤。其流程图如 3-6 所示。解协议的作用是，音频在网络上传输才用了 UDP 协议，它在传输音频的同时也会传送一些信令数据。解协议的过程就是去除信令数据，只保留音频的数据。音频解码的作用是把音频的压缩编码数据，解码成为非压缩的音频原始数据。解码后的 PCM 数据可用于 SDL 进行播放。播放音频的时候，使用了 FFMPEG 对传输过来的音频解码成 PCM 音频数据。

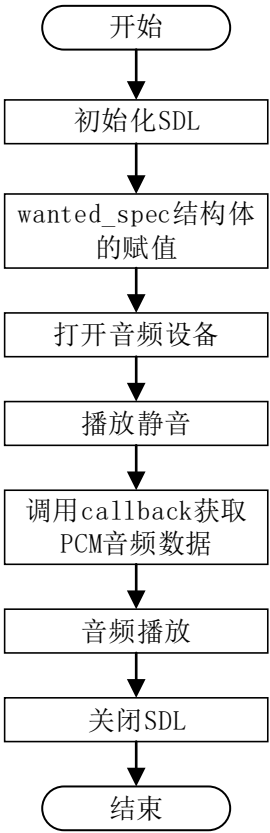


图 3-7 SDL 播放流程图

SDL 播放音频的流程如图 3-7 所示。使用 SDL 播放声音前，首先要对名为

SDL_AudioSpec 类型的 wanted_spec 结构体设置一些关于音频的选项：采样率，通道数，采样精度，然后还要指定一个回调函数 callback 以及用户数据（在播放时需要用到的数据指针）。然后 SDL 需要通过 SDL_OpenAudio(&wanted_spec, &spec) 打开设备，成功返回零。通过调用 SDL_PauseAudio(0) 来让音频设备刚开始播放静音，当 callback 变得可用时，音频设备的播放状态才会发生改变。

这里有两个线程，一个是接收音频数据的线程，另外一个 SDL 获取播放数据的线程。接收音频的数据缓存到一个数据队列里面—audioPacketQueue。先初始化 audioPacketQueue，然后由 FFMEG 会从相应的 8888 端口中取出 Packet 并填充到 audioPacketQueue。

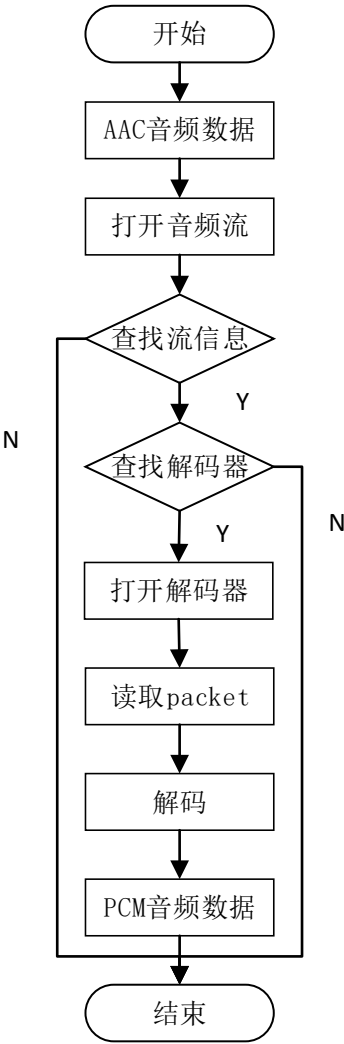


图 3-8 AAC 解码成 PCM 流程图

SDL 播放需要音频数据的时候，会启动获取播放数据的线程，然后获取播放数据的线程会运行 callback 函数中会对 Packet 进行解码，并将解码后的数据

传送到音频设备播放。解码的流程如图 3-8 所示。

在 callback 函数中，设置了一个静态的数据缓冲区，用于存放解码后的音频数据。callback 函数每次向音频设备传送数据时，首先检测静态缓存区中是否有数据，有则直接复制到 stream 中，否则调用 `audio_decode_frame()`，从 Packet 队列中取出 Packet，将解码后的数据填充到静态缓冲区，然后再传送到音频设备。

得到解码后的数据后，还有一个问题就是，输入的音频格式可能和 SDL 打开的音频设备所支持的格式不一致，所以在将数据放入到 callback 的静态缓冲区前，需要对音频进行重采样。

首先创建一个类型为 `SwrContext` 的指针 `swr_ctx`，并设定好转换参数，并初始化该 `SwrContext`。设定的 `swr_ctx` 转换前后的数据布局 `channel_layout`，`format` 和每秒的 `sample` 个数 `sample_rate`。然后调用 `swr_convert()` 就完成了输入的音频到输出音频格式的转化了。音频重采样的流程如图 3-9 所示。

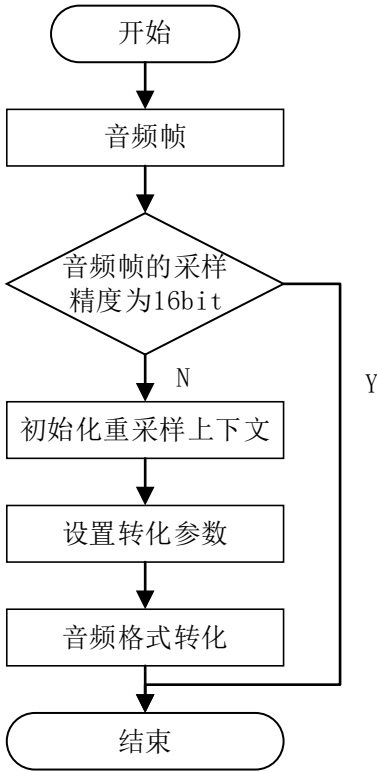


图 3-9 音频重采样流程图

4.系统测试

4.1 用户列表显示模块测试

在这里，使用了远程桌面的方式，实时查看测试的两台 PC 的用户登陆状态。其结果如图 4-1 所示。用户 **lucy** 先登陆，然后再到用户 **saber** 登陆。测试表明用户列表显示模块实现了用户列表的实时更新的功能。

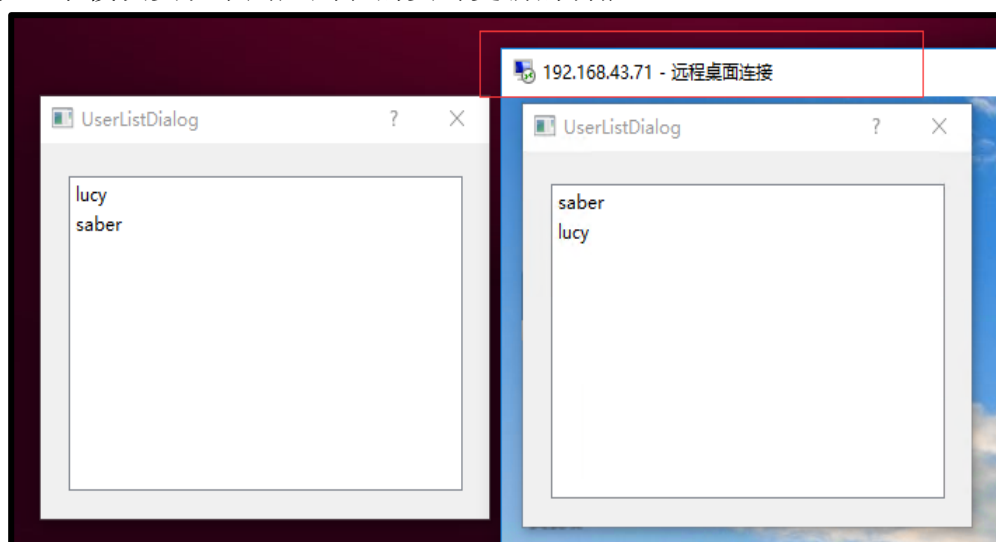


图 4-1 用户登陆图

4.2 网络流量测试

网络流量测试主要是测试音频在传输的时候，所占用的带宽。这里使用了 wireshark 对其进行网络流量分析，wireshark 是一个强大的网络封包分析软件。能指定要捕获的网络接口，同时也拥有着强大的过滤器引擎。使用 wireshark 的时候可以使用过滤器筛选指定目的地址的 ip、通讯的协议和端口的数据包。部分捕获的数据如图 4-2 所示。测试的本机地址 ip 是 192.168.43.16, 测试的目的地址是 192.168.43.71。所以这里需要用一个过滤器，筛选出目的主机传输过来的数据。可用 `ip.src == 192.168.43.71` 进行数据的筛选。在图 4-2 中，可以看到每一个包接收的时间，通讯的本机地址和目的地址，还有通讯的端口号 8888。

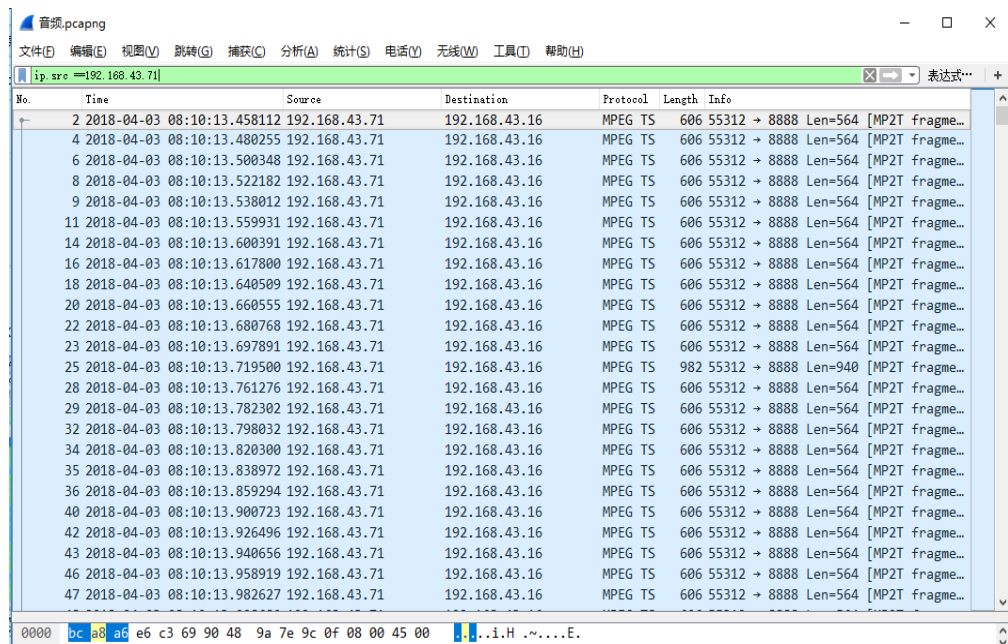


图 4-2 网络流量捕获图

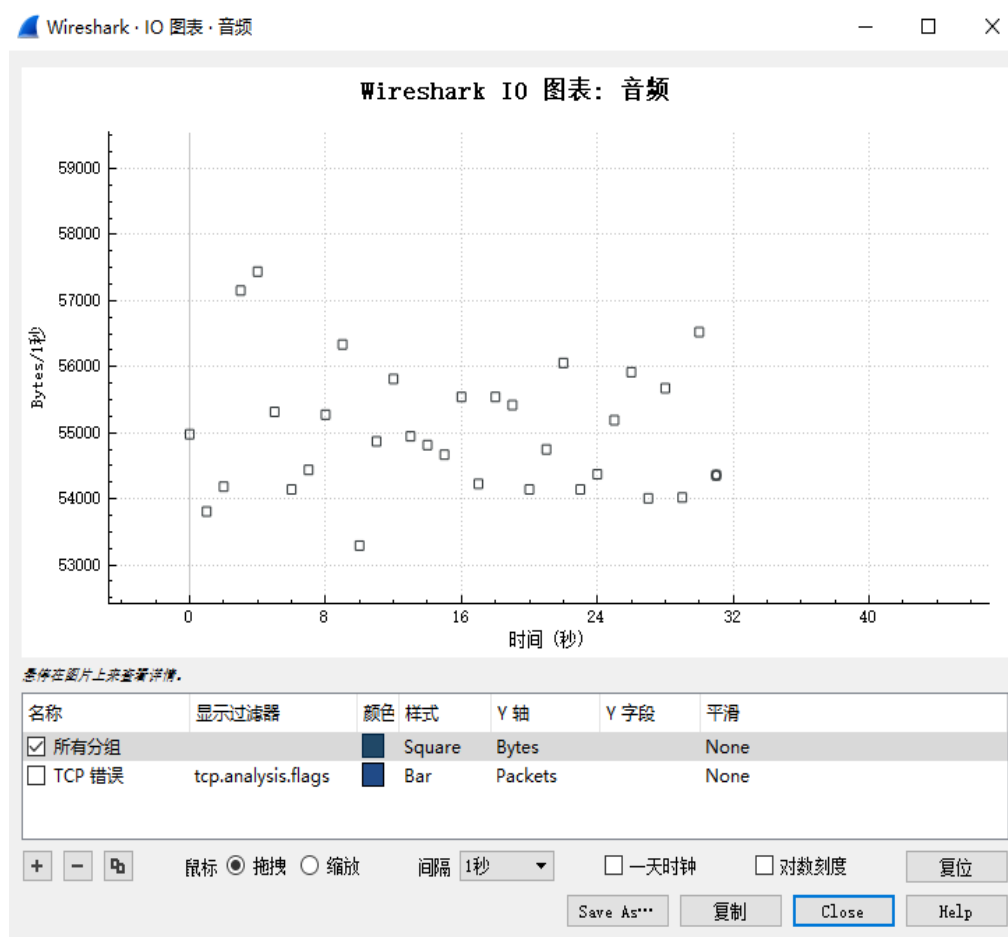


图 4-3 网络流量分析图

然后对捕获后的数据进行分析。wireshark 已经集成这一功能了。分析所占

带宽情况如图 4-3 所示。从图 4-3 可以看出，本软件所接收的音频数据所占用的带宽在 53000byte 到 58000byte 之间，转化为 kb 的话，大约是在 51kb 到 57kb 之间。

4.3 音频采集模块测试

音频采集模块测试是为了测试采集后的音频经过 UDP 传输后其数据是否能正常播放。这里使用了 ffplay 播放器对其进行测试，ffplay 是一个使用了 FFMPEG 和 SDL 库的媒体播放器。由于通讯对端传输过来的音频是传到本机的 8888 端口，所以 ffplay 的播放命令是 `ffplay -i udp://127.0.0.1:8888`。当 8888 端口有音频数据的时候，ffplay 就会对其进行播放。从图 4-4 可以看到音频的采样率等信息。如果端口没有音频数据，那么就不能看到这些信息。

```
Input #0, mpegts, from 'udp://127.0.0.1:8888':0KB sq= 0B f=0
/0
Duration: N/A, start: 0.000000, bitrate: 158 kb/s
Program 1
Metadata:
  service_name      : Service01
  service_provider : FFmpeg
Stream #0:0[0x100]: Audio: aac (LC) ([15][0][0][0] / 0x000F
), 44100 Hz, stereo, fltp, 158 kb/s
1.48 M-A: 0.000 fd= 0 aq= 101KB vq= 0KB sq= 0B f=0
```

图 4-4 ffplay 播放传输的音频图

4.4 音频播放模块测试

音频播放模块测试是为了检测其模块的音频播放功能是否正常。这里使用了 FFMPEG 进行麦克风推流，推送到本机的 8888 端口。在 cmd 窗口输入推流命令如图 4-5 所示。

```
C:\Windows\system32\cmd.exe

C:\Users\jiegeng>ffmpeg -f dshow -i audio="麦克风 (Real
tek High Definition Audio)" -f flv udp://127.0.0.1:8888
```

图 4-5 FFMPEG 推流命令图

然后在同一个 cmd 窗口出现推流成功的信息，如图 4-6 所示。在图中可以看到音频的采样率等信息。然后把本设计关于音频播放的代码分离出来，组成一个专门用于音频播放小项目。然后用那个音频播放项目测试端口为 8888 的音频流是否能正常播放。

```
Input #0, dshow, from 'audio=榼~厠棕?(Realtek High Definition Audio)': Duration: N/A, start: 58820.992000, bitrate: 1411 kb/s
  Stream #0:0: Audio: pcm_s16le, 44100 Hz, stereo, s16, 1411 kb/s
Stream mapping:
  Stream #0:0 -> #0:0 (pcm_s16le (native) -> mp3 (libmp3lame))
Press [q] to stop, [?] for help
Output #0, flv, to 'udp://127.0.0.1:8888':
  Metadata:
    encoder      : Lavf58.9.100
  Stream #0:0: Audio: mp3 (libmp3lame) ([2][0][0][0] / 0x0002), 44100 Hz, stereo, s16p
  Metadata:
    encoder      : Lavc58.11.101 libmp3lame
size=      15kB time=00:00:00.90 bitrate= 136.8kbits/s s
```

图 4-6 ffmpeg 推流成功显示图

4.5 语音聊天测试

语音聊天测试，是为了测试两位用户使用本系统的时候能否正常交流，声音传输有没有延时，声音的音质是否差。两位用户登陆本系统，然后就可以测试了。

4.6 测试结果与分析

经过测试，得出以下结果：

- (1) 用户在线列表能够实时进行更新。
- (2) 本系统通讯的时候，网络带宽占用大约是 55kb/s。
- (3) 音频采集模块的功能正常。

(4) 音频播放模块的功能正常。

(5) 能够正常语音聊天，音频聊天有不到 1s 延时，同时还有回声存在。

测试结果分析，在局域网内，两三兆每秒的速度是可以达到的。根据本系统的网络带宽的数据来看，本系统采集的音频完全可以在局域网顺利传输。由于声音是在网络传输的，而且音频解码也是需要时间的，所以出现延时是不可避免的。回声的出现是因为本系统会同时采集用户说话的声音和对端传过来的声音，本系统并没有对回声进行消除。回声消除可以使用 webrtc 的 ace 模块对回声进行消除，后续继续完善。

5. 总 结

5.1 总结

科技的发展改变了我们的生活方式，人们从传统的书信通讯变为了与网络有关的通讯，比如语音聊天。现在的视音频压缩的技术日趋完善，再加上网络传输速度不断提供，使得音频的通讯成为了 internet 上的重要的业务。

本系统以为 windows10 为开发平台，在 qt creator 上进行开发，研究了在局域网内实时语音聊天，主要取得了这几个成果。

(1) 主要阐述了通过 DSHOW 的音频的录制，音频的编码，音频的解码，通过 FFMPEG 的音频格式的转化。

(2) 学习了 Qt 重要的机制，信号与插槽。信号和槽是 Qt 自行定义的一种通信机制，同时也可以跨越不同的组件。

(3) 使用 UDP 进行广播，局域网内的其他的 UDP 用户全部可以收到广播的消息。本设计通过这个特点，完成了实时用户列表更新的功能。

(4) 知道了 Qt 连接 Mysql 数据库的时候要加载驱动，同时也要设置一些连接的属性。同时也知道了默认情况下，Mysql Server 是不允许远程操作的。

5.2 期望

本系统对音频的录制，编码，播放进行了研究，实现了 Qt 的语音聊天。但是只是实现了部分功能，实现的功能也不是很完善。在本系统的基础之上，可以从这几点去继续完善。

(1) 只能局限于局域网的通讯。

(2) 当两个人利用本系统进行语音聊天时，系统并没有对回声进行消除。

(3) 功能尚且不够丰富，缺少文件传输，文字聊天等功能。

(4) 与用户进行交互的 ui 界面不够吸引人的眼球，后续可以进行美化。

参考文献

- [1] 张珏. 一种基于 P2P 结构的即时通讯系统的设计与实现[D]. 西安: 西安电子科技大学软件工程 2010. 25-26
- [2] 杨婷婷. 宽带中国开启美丽生活[J]. 现代工业经济和信息化, 2014(Z1). 74-75
- [3] 江霏霏. 某型火炮营级指挥控制模拟系统的若干分系统设计[D]. 南京: 南京理工大学, 2012. 61-62
- [4] 姚远. 视频监控图像后处理算法的研究与实现[D]. 北京: 北京邮电大学, 2010. 48-49
- [5] 冀胜利. 基于嵌入式 Linux 的网络视频传输研究和实现[D]. 重庆: 重庆理工大学, 2009. 33-34
- [6] 陈熙. 基于 Android 移动平台视频监控终端的设计与实现[D]. 成都: 电子科技大学, 2013. 23-24
- [7] 余柳冰. 基于嵌入式 Linux 的电脑横机控制软件开发[D]. 杭州电子科技大学, 2011. 40-41
- [8] 余东. 多描述视频编码研究[D]. 太原: 太原科技大学, 2013. 50-51

致 谢

在这论文结束之际，在这里首先感谢我的指导老师谭石坚老师，他耐心地引导我进行学习有关音频开发方面的知识，每当我在毕业设计遇到困境的时候，总是开导我，要我保持身心的愉悦，不要轻易言放弃，并引导我解决在做毕业设计遇到的问题。当然了，前提是要我经过自己的努力，实在难以自己解决的时候。再次感谢指导老师，是指导老师的鼓励让我有信心去奋斗，去攻克种种在做毕业设计遇到的问题，让我的毕业设计顺利地在规定时间内完成。

指导老师还对我的论文提出了许多宝贵的建议，让我的论文不再像是流水账般的无趣，增强了论文的可读性。在此谨向谭老师致以衷心的感谢和深深的敬意！还要对所有授课老师表示浓浓的谢意！正是他们的教导，才能使我的知识得以丰富，使我的视野得以拓宽，使我的自身修养得以提高！

感谢四年养育我的华软，感谢四年陪伴我的同学，感谢父母的养育之恩，你们的帮助和鼓励给予了我坚定的信念和坚强的决心，让我勇敢的向前迈进。

最后，对阅读本论文的老师表示衷心的感谢。

