 ：一种基于音频的旁路通信

CS339 Fall 2019, Group 7: Surdus Successore*
朱睿（休学），郑扬珞，马捷

摘　　要

智能手机、笔记本计算机等个人数字设备大都配备了音频输入输出设备，因而基于音频通信方式在即时的移动无线通信中具有广阔的应用场景。本文提出了一种基于音频信号的旁路通信方案，利用频率调制和相位调制相结合的方法，将信息编码至人耳不可闻的音频信号中，并探讨了不同的解码方案，最后给出了一种计算量小、延迟低、准确率高的解码算法，能够实现高达 1500 bps 的传输速率。

1　引言

旁路通信指一切利用非寻常介质进行的信息传递，是计算机网络的重要研究方向之一。而基于音频的旁路通信，就是将信息编码进音频信号中，利用扬声器、麦克风等音频设备进行信息的传递。随着智能手机等移动设备的普及，以及嵌入式处理器性能的提升，音频通信已经可以在移动终端上实现，我们相信这种通信方式有着广泛的应用场景。

二维码是一种日常生活中十分常见的信息载体，在身份识别、支付等场景下被人们广泛地使用。然而，二维码在传递信息过程中，可控制性并不强，从二维码显示在屏幕上开始，它的内容就对外界暴露了，其中的信息很容易被他人截获，从而造成隐私泄露、资金安全受损等后果。尽管许多二维码具有使用后立即作废的特性，但也难以防范二维码信息在使用前就已经泄露的情况。也有一些工作利用摩尔纹对二维码进行干扰 [1]，限制其有效使用范围，但正常扫描时，对距离和角度的要求较高，会给使用者带来不便。相比之下，利用音频作为信息的载体，具有更好的可控性，而且声波在空气中的衰减很迅速，因此安全性更强。我们认为，音频通信可以作为一种替代二维码的即时小规模通信方式，这也是本项目的主要动机和出发点。

本项目实际上是一个原创项目，我们在前期调研时检索的关键词不恰当，因而没有注意到很多有价值的工作。事实上，音频通信已有大量的探索，这些工作将在下一节进行介绍，其中有不少在安全性以及通信速率等方面都取得了优于本文的成果。当然，我们认为本项目中信息的编码方式以及解码环节的一些考虑都有一定的创新之处。

*项目源码：https://github.com/WunschUnreif/Surdus-Successore

2　相关工作

人们对于音频通信有深入的研究。事实上，曾经的拨号电话就是利用音频传递拨出的号码：电话用两个不同频率的正弦音频叠加表示一个可能出现在号码中的字符，然后将该音频通过电话线缆发送出去。当然，在电话拨号信息的传输过程中，音频信号首先被转换为干扰更小的电信号，而且这种方法随着数字信号通信的普及也逐渐消亡了。然而音频通信的研究并没有随之止步。

在 [2] 中，作者实现了多种音频通信方式。在不可闻频率区间内，作者利用 18.4kHz 的单一频率实现了 1470bps 的通信速率。而在利用整个 4kH 至 18kHz 频段的情况下，实现了 3.4kbps 的通信速率，这种通信方式使用 FFT 进行解码，一帧数据包含 80 位信息，长度为 1024 个音频采样点。

在 [3] 中，作者利用 OFDM 调制等方法，实现了一个可用于替代 NFC 的音频通信系统，通信速率可以达到 2.4kbps。此外，作者还使用在接收端产生干扰信号的方式，提高通信内容的安全性，但也限制了通信的距离，使得这项工作只适用于 20cm 以内的通信。

在 [4] 中，作者利用线性调频信号（chirp signal）进行音频调制，实现了在高达 25m 距离内的室内音频通信，而通信速率只有 16bps。

可以看到，对于音频通信，在通信的速率、安全性、传输距离等方面均有相应的探索。

3　音频通信的编码环节

3.1　工作频段的选择

音频通信工作频段的选择，主要考虑的因素是频段的抗干扰性以及对人的听觉感受。工作频段的噪声应该尽可能低，保证通信的可靠性，同时由于编码后的音频可能并不悦耳，因此需要选取人耳不敏感的频段。

图1是我们对日常生活中的环境噪音进行测量后绘制出的频谱图，测试环境为正在播放音乐的宿舍房间内。可以看到，环境噪音主要集中在 9kHz 以下的低频区间内，15kHz 以上的频率范围几乎不受噪音的影响，很适合进行通信。此外，大多数成年人对 17kHz 以上的音频极不敏感 [5]，因此使用 17kHz 以上的频段，在通信时大多数人是无法察觉的，也不会感到不适。又考虑到大多数手机扬声器的有效频率范围都在 20kHz 以内，我们最终将音频通信的工作频率范围取为 17kHz 至 20kHz。

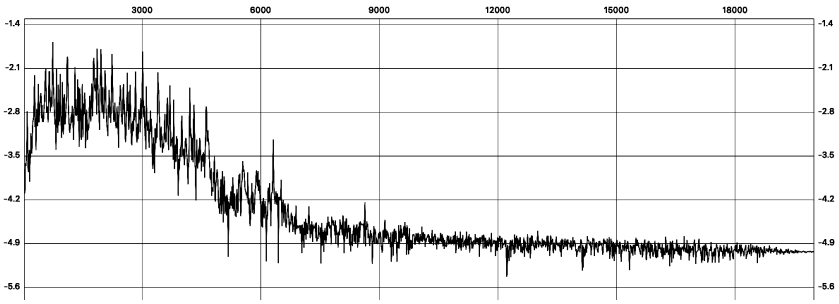


图 1: 环境噪音频谱

3.2 信息的编码方案

3.2.1 利用频率上信号的有无表示单个二进制位的方案

一个二进制位的取值 0 和 1 恰好可以对应某个频率上是否出现信号峰，因此很自然地可以想到将工作频率划分为几个离散的频率点，对应不同的二进制位，每个频率点上是否出现信号峰表示相应位的取值为 0 或 1。具体来说，我们从 18.0kHz 至 19.5kHz 每隔 100Hz 取一个频率点，每帧编码 16 位二进制信息。若待编码的二进制数可以表示为 $b = \overline{b_{15}b_{14}\cdots b_1b_0}$，则编码后一帧内的音频可用下式计算：

$$u(t) = \sum_{i=0}^{15} b_i \cdot \sin(2\pi f_i t) \tag{1}$$

其中，b_i 为待编码信息二进制表示的第 i 位数值，$f_i = 18000 + 100 \times i$ 为第 i 位对应的音频频率，t 为音频在当前帧内的时间。

这种编码方式的频谱利用率很高，因此理论上可以实现较高的通信速率，但实际上，其传递每一个二进制位的方法本质是幅度调制，在解码时必须确定划分信号有无的阈值。由于频谱泄漏的存在，即使某一个频率点上没有信号，其测量出的信号幅度也可能由于邻近频率点上信号的影响而超过阈值，造成误码。在实验中，我们尝试使用固定的阈值或利用 K-Means 算法动态计算阈值，通信中的误码率都非常高，因此最后我们放弃了这种编码方式。

3.2.2 利用频谱独热码表示多个二进制位的方案

在一个频率点上确定划分信号有无的阈值并不容易，但在一个频率区间内找到幅度最大的频率十分简单，并且频谱泄漏带来的干扰一般不会造成峰值位置的改变，因此，可以考虑在一个频率区间内使用独热码的编码方式表示多个二进制位。具体来说，我们仍选用 18.0kHz 至 19.5kHz 间隔 100Hz 的 16 个频率点来编码信息，并进一步将其按每 4 个相邻的频率点一组划分为 4 组频率区间，在一帧内，每个频率区间内的 4 个频率点中恰有 1 个可以出现信号，因此一组频率点可以编码 $\log_2 4 = 2$ 位二进制数，或一个四进制数，而 4 个频率区间一共可以表示 8 位二进制数。若待编码的 8 位二进制数在四进制下可以表示为 $q = \overline{q_3q_2q_1q_0}$，则编码后一帧内的音频可用下式计算：

$$u(t) = \sum_{i=0}^{3} \sin(2\pi \cdot (f_i + 100 \cdot q_i) \cdot t) \tag{2}$$

其中 $f_i = 18000 + 400 \times i$ 为第 i 个频率区间的起始频率，而 $f_i + 100 \cdot q_i$ 计算出了在第 i 个频率区间内表示四进制数 q_i 的频率点。

这种编码方式的频谱利用率并不高，16 个频率点只编码了 8 位二进制数，但采用了独热码之后，传递的每个四进制数都由所处频率区间内的 4 个频率点共同确定，因此解码时的误码率很低，具有较强的抗干扰性。这种编码的图示见图2。

18k ~ 18.3k				18.4k ~ 18.7k				18.8k ~ 19.1k				19.2k ~ 19.5k			
0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3
q_0				q_1				q_2				q_3			
b_0		b_1		b_2		b_3		b_4		b_5		b_6		b_7	

图 2: 频谱独热编码方式

3.2.3 引入相位调制的方案

一个正弦信号由幅度、频率和相位三部分组成，而之前的编码方案只用到了频率和幅度两项信息，如果再引入相位信息，可以提高频谱利用率，加快通信速率。我们在3.2.2中频谱独热码的基础上，对每一个频率点使用四分相位调制，即利用峰值频率点在频率区间内的位置以及该位置上正弦波的相位共同编码信息，从而一个频率区间可以具有 16 种不同的状态，能够表示 4 位二进制数，而全部 4 个频率区间共能表示 16 位二进制数。若待编码的 16 位二进制数在十六进制下可以表示为 $h = \overline{h_3h_2h_1h_0}$，则编码后一帧内的音频可用下式计算：

$$u(t) = \sum_{i=0}^{3} \sin(2\pi \cdot (f_i + 100 \cdot \lfloor h_i/4 \rfloor) \cdot t + \frac{\pi}{2} \cdot (h_i - 4 \cdot \lfloor h_i/4 \rfloor)) \tag{3}$$

可以看到，信号的频率由每一位十六进制数除以 4 的商决定，即 $f = f_i + 100 \cdot \lfloor h_i/4 \rfloor$，而相位由其余数决定，即 $\varphi = \frac{\pi}{2} \cdot (h_i - 4 \cdot \lfloor h_i/4 \rfloor)$。

这种编码方式利用了相位信息，从原来的频率和幅度信息中，又拓宽了一个维度，因此一帧能编码的信息量增加了一倍，通信速率也能获得提高。然而在移动通信的场景中，设备之间位置很小的改变都会引起相位的巨大变化，因此可能需要使用固定装置以保持相对位置不变。这种编码方式的图示见图3。

	18k ~ 18.3k				18.4k ~ 18.7k				18.8k ~ 19.1k				19.2k ~ 19.5k			
0°	0	4	8	C	0	4	8	C	0	4	8	C	0	4	8	C
90°	1	5	9	D	1	5	9	D	1	5	9	D	1	5	9	D
180°	2	6	A	E	2	6	A	E	2	6	A	E	2	6	A	E
270°	3	7	B	F	3	7	B	F	3	7	B	F	3	7	B	F
	h_0				h_1				h_2				h_3			
	byte 0								byte 1							

图 3: 引入相位的编码方式

3.3 同步信号的编码

3.3.1 时钟信号

在音频通信的过程中，我们希望音频自身包含时钟信号，即采取同步通信的方式。利用同步通信，一方面接收方可以自动适应通信时采用的传输速率，不需要额外调整，另一方面不会造成时钟不同步而引起的重复接收、遗漏以及误码。此外，我们还希望时钟信号中带有信息编码方案的标志，即可以从时钟信号中了解到当前音频中的信息是采用3.2.2中的方式还是3.2.3中的方式进行编码的，从而接收方可以自动选取相应的解码方式。

基于这些考虑，我们依然采用频谱独热码的方式对时钟信号进行编码，选取 17.4kHz, 17.5kHz, 17.6kHz, 17.7kHz 四个频率点作为四种不同含义的时钟信号，它们的具体含义见表1。

对于编码方式的区分，只是简单地为每种方式分配不同的时钟信号频率点，而时钟信号实际上是一个模 2 的帧序号，在信息传输过程中用于区分相邻的两帧数据，防止两帧内容相同时接收方无法区分。

表 1: 时钟信号含义

频率	含义
17400	帧序号为 1，编码方式为3.2.2中的方式
17500	帧序号为 0，编码方式为3.2.2中的方式
17600	帧序号为 1，编码方式为3.2.3中的方式
17700	帧序号为 0，编码方式为3.2.3中的方式

3.3.2 相位同步信号

当采用3.2.3中的相位调制方式进行编码时，在实际的通信过程中，发送端的扬声器与接收端的麦克风之间存在一定的距离，因而实际接收到的相位会发生偏移。设对于频率点 f，扬声器发出的音频为：

$$u(t) = \sin(2\pi f t + \varphi) \tag{4}$$

我们认为，麦克风接收到的音频应该为：

$$y(t) = \sin(2\pi f t + \varphi - d \cdot \frac{2\pi}{\lambda} + 2\pi f \Delta t + \Delta\varphi) \tag{5}$$

其中引起相位偏移的有 3 项：由扬声器和麦克风之间距离引起的偏移，即 $-d \cdot \frac{2\pi}{\lambda}$，其中 d 为距离，λ 为频率 f 对应的波长；由收发双方采样时间起点不一致引起的偏移，即 $2\pi f \Delta t$；以及其他各种环境因素引起的偏移 $\Delta\varphi$。可以看到，至少有两项因素对接收到的相位产生的偏移是与频率有关的，这就导致在接收端检测出的不同频率的相位偏移是不同的，并且难以根据采样值计算出实际的相位偏移。但是，当环境因素不变时，上述引起相位偏移的因素都是不随时间变化的，因此只要接收方在通信开始时对各频率上的相位进行采样，就能够在整个通信过程中还原实际的相位。这就是引入相位同步信号的作用。

在实现上，只需要依次发送 16 个信息编码频率点上相位为 0 的正弦波音频，让接收方进行采样即可，但每次只同步一个频率点过于耗时，因此我们选取每帧同步 8 个频率的相位，并在相位同步时，利用 17.8kHz 和 17.9kHz 两个频率点来标记当前同步的是哪些频率点。两帧相位同步帧的音频可由下式计算：

$$u_1(t) = \sin(2\pi \cdot 17800 \cdot t) + \sum_{i=0}^{7} \sin(2\pi \cdot (18000 + 2i \times 100) \cdot t + 0) \tag{6}$$

$$u_2(t) = \sin(2\pi \cdot 17900 \cdot t) + \sum_{i=0}^{7} \sin(2\pi \cdot (18000 + (2i+1) \times 100) \cdot t + 0) \tag{7}$$

在第一帧中，同步偶数位对应的编码频率，并用 17.8kHz 上的信号作为标志；在第二帧中，同步奇数位对应的编码频率，用 17.9kHz 上的信号作为标志。

通信过程中，相位同步信号既可以只在开始时出现一次，也可以在传输过程中反复出现，以消除可能会随时间累积的相位偏移。

3.4 信息编码的其他细节

3.4.1 音频采样率与位深度

音频采样率与音频的位深度决定了音频播放时对原始信号的还原精度。由于在编码中，最高的频率可以达到 19.5kHz，因此音频的采样率至少需要达到 39kHz。此外，当使用相位调制的编码方式时，一个振动周期内需要更多的采样点数，因此需要更高的采样率。最终我们在编码时选取的音频采样率为 96kHz。对于音频的位深度，为了程序实现的便利，我们直接采用了 32 位的浮点格式，但由于编码中信号的幅度并不需要十分精确，因此 8 位的音频位深度应该就足够了，选用较小的位深度可以缩小编码后音频文件的大小。

3.4.2 通信速率与帧长度

由于编码中包含了时钟信号，因此通信所采用的速率可以在合理范围内任意选择，而一帧数据的持续长度也有通信速率决定。设期望的通信速率位 v bps，当采用3.2.2中仅用频谱的编码方式时，一帧传输的数据为 8 位，因此相应的帧长度为 $T_{frame} = 8/v$ (s)；当采用3.2.2中引入相位的编码方式时，一帧传输的数据为 16 位，相应的帧长度为 $T_{frame} = 16/v$ (s)。

对于相位同步信号的帧长度，为了能够让接收端采集尽量多的相位样本，应该比通信时的帧长度更长。在我们的实现中，通信开始时的相位同步信号的帧长度为数据帧长度的 5 倍。

3.4.3 时间的连续性

利用相位调制方式进行编码时，公式 (3) 中每帧音频的时间是独立计算的，即编码好一帧后，下一帧的波形是从时间 $t = 0$ 开始计算的，但由于帧长度不能保证一帧之内每个频率上的波形都出现了整数个周期，因此如果在实现时采取这样的计时方式，每经过一帧，音频的相位都会发生变化。为了使相位调制编码能够正常工作，必须保证整个音频计算中时间的连续性，即从相位同步信号开始，到所有信息编码结束，都采用一个全局的时间，一帧编码完成后，这个时间不清零。而对于不采用相位调制的编码，相位信息并不重要，因此不需要考虑时间的连续性。

3.4.4 相位同步插入

在实验中，我们发现不同设备播放音频的速度略有差异，由此会造成一定的频率偏移，但不会累积，而速度差异还会造成相位偏移，并会随时间累积，如果只在通信开始时进行一次相位同步，则能够正确传输数据的时间很有限。为了解决这个问题，我们在通信的过程中按照一定间隔加入相位同步帧，保证在相位偏差累积到不可忽视的程度之前会重新同步相位，从而延长了有效传输时间。在我们的实现中，每传输 32 字节的数据就加入一次相位同步帧，需要注意的是，为了提高通信效率，这里插入的相位同步帧长度与数据帧长度一致，而不需要使用通信开始时 5 倍于数据帧长度的相位同步帧。对于相位同步的插入间隔，需要根据设备的具体情况来进行调试，如果数据长度较短或收发双方的音频播放与采样速度差异不严重，也可以不插入相位同步帧。

4　音频通信的解码环节

4.1　解码的一般步骤

在解码中，一般的过程是首先对音频进行采样，从得到的数据中提取出关键频率点（即编码中会用到的频率）的功率和相位信息，以此对音频表示的数据进行估计，估计结果送入一个滑动窗口中，寻找该窗口中出现最频繁的元素就可以得到当前的数据帧。解码环节的流程图见图4。下面将首先对音频采样和数据估计两部分进行简要的介绍，然后对解码环节中的核心步骤——相幅分析进行详细的讨论。

4.2　音频采样

对于音频采样环节，唯一需要考虑到就是采样率的设置。为了准确获得发送端发送的音频波形，解码端音频的采样率应当不低于编码时的采样率。在编码中，采样率为 96kHz，因此这里我们同样选择 96kHz 作为解码时的音频采样率。

4.3　数据估计

在数据估计中，需要根据关键频率点的功率和相位信息给出数据内容的估计或进行相位同步。具体的过程如下：

1. *判断是否有数据正在传输*：当发送端正在发送数据时，时钟信号对应的 4 个频率点以及用于相位同步标记的 2 个频率点中一定有一个频率上有信号，即功率远大于背景噪声。因此，求出 17.4kHz 至 17.9kHz 的 6 个频率点中的最大功率，并与 16kHz 的背景噪音功率进行比较，若两者之差超过一个阈值（在我们的实现中，为 40dB），就认为正在传输数据。
2. *判断数据帧的类型*：求出 17.4kHz 至 17.9kHz 的 6 个频率点中最大功率对应的频率，就可以根据编码时的约定得到当前是数据帧还是相位同步帧，如果是数据帧，还可以进一步得到采用的编码方式，以及作为时钟的帧序号。
3. *对于数据帧的处理*：求出每一个频率区间的峰值频率以及对应的相位，并按照3.2.2和3.2.3中的方法得到 4 个频率区间中编码的信息，然后组合成一个二进制数，就得到了对数据帧内容的估计。
4. *对相位同步帧的处理*：对于相位同步帧，需要将各数据频率上采集到的相位放进一个缓冲区中，待相位同步帧结束之后，对缓冲区中的相位角求均值（需要分别对角度的正余弦值平均，然后得到平均角度），作为各频率上相位的修正值。

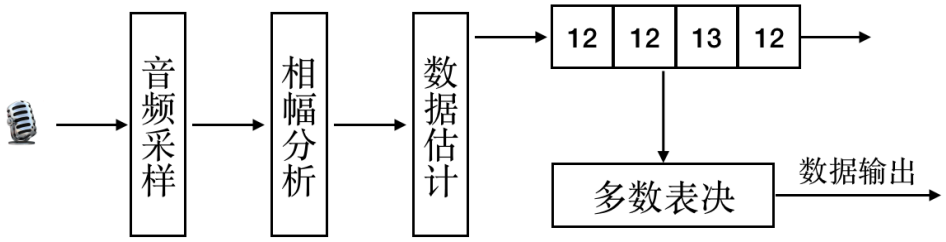


图 4: 解码的一般步骤

4.4 功率谱与相位谱的计算

4.4.1 利用快速傅立叶变换

设由麦克风采样得到一个窗口内的音频信号为 $y(k\Delta T),\ k = 0, 1, \ldots, N-1$，其中 N 为窗口大小，ΔT 为采样间隔。则该窗口上的离散傅立叶变换为：

$$Y(k) = \sum_{n=0}^{N-1} y(n\Delta T) \cdot \mathrm{e}^{-i \cdot \frac{2\pi}{N} kn},\ k = 0, 1, \ldots, N-1 \tag{8}$$

其中 $Y(k)$ 代表频率为 $f(k) = \frac{k}{N\Delta T} = \frac{k}{N} f_s$ 的变换结果，f_s 为采样率。设 $Y(k) = \sigma + i\omega$，则可以得到频率 $f(k)$ 处对应的信号功率为 $P_{f(k)} = \sigma^2 + \omega^2$，相位为 $\varphi_{f(k)} = \arctan \frac{\omega}{\sigma}$。因此利用傅立叶变换就可以获得关键频率点附近的功率和相位信息。

当窗口大小为 2 的幂次时，离散傅立叶变换有时间复杂度为 $O(n \log n)$ 的快速算法，即快速傅立叶变换，并且这一算法在许多计算机上都有硬件加速或高度优化的实现（例如 [6]），因此可以实现实时计算。在实现时，我们将窗口大小定为 4096，从而在采样率为 96kHz 时，能够达到的频谱分辨率为 $\Delta f = f_s/N = 96000/4096 = 23.4\mathrm{Hz}$，对于关键频率点的间隔 100Hz 来说，已经足够了。而一个大小为 4096 的采样窗口对应的采样时间约为 42.6ms，因此在通信时，音频信号变化的波特率最高只能到 23.4Hz。实际上，为了降低误码率，可用的波特率上限更低，通信速率受到很大的限制。此外，在实验中，我们发现利用快速傅立叶变换得到的相位信息很不稳定，完全无法解析利用相位调制的音频。

经过分析，我们认为基于傅立叶变换的解码具有下面几个弊端，从而不适合作为解码端获得功率和相位的工具。

- 产生的无用信息过多：傅立叶变换可以得到小于采样率一半的所有频率上的正弦和余弦分量，但解码时需要关心的只是一些频率已知且固定的频率点。事实上，傅立叶变换一般只适用于对未知频谱的估计。
- 需要的窗口长度过大：为了提高频谱分辨率，必须增加采样窗口的长度，从而得到一次功率谱和相位谱的时间也更长，限制了音频变化的波特率，也就限制了通信速率。
- 分解的基函数是固定的：对于每一个窗口，傅立叶变换都是用相位为 0 的三角函数进行分解，但窗口长度并不能保证每个频率都经过整数个周期，因此不能获得稳定的相位谱，无法解析利用相位调制的音频信息。

4.4.2 利用时域滤波器

受到 [7] 的启发，我们发现可以直接对音频信号进行时域操作，得到稳定、准确、低延时的功率和相位信息。

时域滤波器基本框架 对于扬声器接收到的频率 f 的音频 $y_f(t) = A\cos(2\pi f t + \varphi)$，在时域上分别乘以函数 $\cos(2\pi f t)$ 和 $-\sin(2\pi f t)$，可以得到

$$C_f(t) = 1/2(A\cos\varphi + A\cos(2\pi(2f)t + \varphi)) \tag{9}$$

$$S_f(t) = 1/2(A\sin\varphi - A\sin(2\pi(2f)t + \varphi)) \tag{10}$$

所得函数一项是只与幅度和相位有关的常数，另一项是一个频率为 $2f$ 的高频分量。通过一个低通滤波器后，高频分量可以被滤除，从而得到幅度和相位项 $c_f = 1/2A\cos\varphi$, $s_f = 1/2A\sin\varphi$，进而计算出频率 f 上的功率为 $P_f = c_f^2 + s_f^2$，相位为 $\varphi_f = \arctan\frac{s_f}{c_f}$。

对于其中涉及到的低通滤波器，[7] 中使用的是只涉及到加减法操作的 CIC 滤波器，而我们为了实现的简便，采用了效果等价的滑动平均滤波器，但每次计算额外需要一次浮点乘法操作（即乘以窗口大小的倒数）。这种额外的开销并不会对解码端引入过多的计算负担。由于滤波后得到的是基带信号，频率不会很高，为了降低后级运算量，我们加入了下采样环节，降低信号的采样率。滤波器总体的方框图见图5。

对通信时所有的关键频率点都应用这样的滤波器，就能得到解码所需的功率与相位信息。容易看出，这种滤波器中，每个采样信号最多经过 2 次运算（一次进入滑动窗口，一次离开滑动窗口），因此对于长度为 n 的音频采样序列，算法的复杂度为 $O(n)$。又由于其不产生任何无用信息，在实验中，没有经过特殊优化的算法实现也比快速傅立叶变换更快。

滑动平均滤波器设计 在滤波器设计中，滑动平均环节的设计需要特殊的考虑，这一环节作为一个低通滤波器，一方面要滤除频率为 $2f$ 的高频分量，另一方面也要滤除不同频率之间的干扰。考虑到滑动平均滤波器的频率响应为 |sinc| 形的函数，即其零点分布具有周期性，另一方面，编码中选取的频率点之间的间隔为 100Hz，即编码后的音频中，不同频率会产生 $n\times100$Hz 的拍频，如果将滑动平均滤波器的零点放置在 $n\times100$Hz 的频率上，就能完全滤除不同频率之间的干扰。这种做法实际上与 OFDM 调制的思想很相似。当采样率为 f_s，滑动窗口长度为 N 时，滑动平均滤波器的第一个零点位于 $f_{z1} = f_s/N$ 处。由于我们的实现中，音频采样率为 96kHz，并且希望第一个零点位于 100Hz 处，因此需要设定滑动窗口的长度为 960 点。此时，滑动平均滤波器的截止频率为 $f_T = 0.443 f_{z1} = 44.3$Hz，显然也可以有效滤除 $2f$ 频率的高频分量。由于截止频率只是频率响应为 -3dB 处的频率，因此实际上基带频率可以适当高于 44.3Hz，但需要低于零点频率，即 100Hz，这一点限制了通信速率。

下采样设计 经过滑动平均后信号的输出速率与音频采样率相同，为 96kHz，但滤波过后的基带信号有效频率只有不到 100Hz，因此为了降低后级运算负担，减少不必要的计算，可以加入下采样环节降低滤波过后基带信号的采样率。由于基带频率最高可以接近 100Hz，因此下采样的输出采样率应不低于 200Hz。为了获得更平滑的基带信号，达到插值的效果，我们实际采用的下采样输出频率为 1kHz。

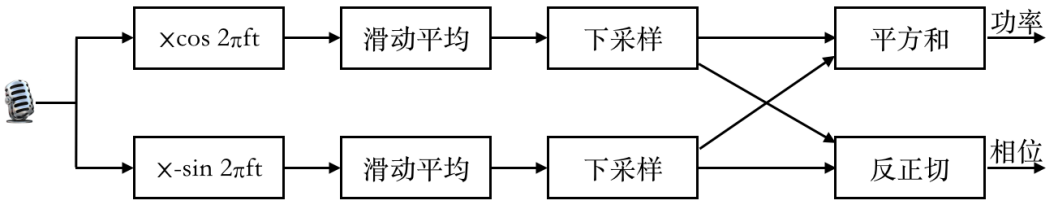
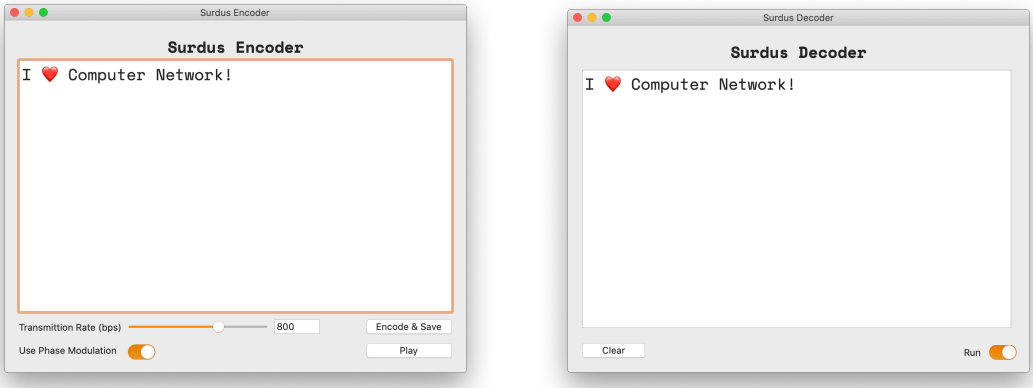


图 5: 时域滤波器框图

5　音频通信的实现与测试

5.1　实现

基于上文中讨论的编码与解码方法，我们在 macOS 10.15.1 平台上实现了一个完整的音频通信软件，主要部分使用 `Swift` 语言编写，而对于运算速度要求较高的解码中的时域滤波器，则使用 `C` 语言实现，并提供了 `Swift` 调用接口。最终实现的软件界面如图6所示。



(a) 编码器　　(b) 解码器

图 6: 音频通信软件界面

在编码器中，我们实现了3.2.2以及3.2.3中的两种编码方式，用户可以通过软件左下角的开关选择是否使用引入相位的编码方式。此外，编码时采用的通信速率也可以在一定范围内任意调节，当不使用相位调制时，通信速率的调节范围为 20bps 至 600bps，而使用相位调制的情况下通信速率的上限设为 1500bps。这样的速率限制根据实验中不出现显著误码情况下的最大速率而确定的。在编码器中，用户可以选择将编码后的音频保存下来，也可以直接播放。

在解码器中，我们实现了4.4.2中基于时域滤波器的解码算法，可以自动适应不同的编码方式以及通信速率，因此软件界面非常简单，只有一个运行开关和一个用于清空显示的按钮。

5.2　测试

5.2.1　测试内容与方法

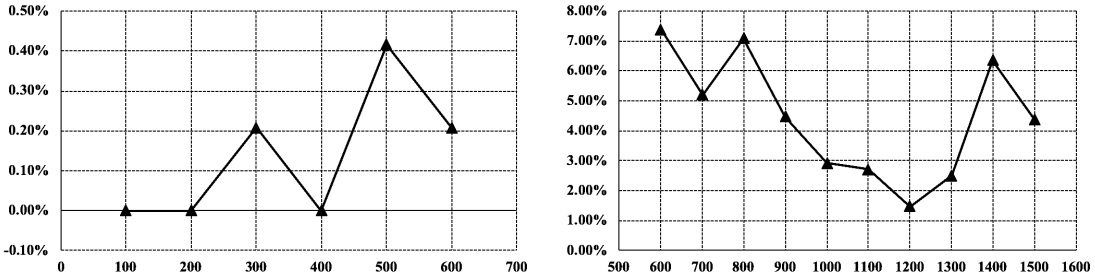
对于一种物理层通信方式的实现，最需要关心的是通信中的误码率，因此在测试中，我们对不同通信速率、不同音量大小以及收发双方不同距离情况下的误码率都进行了测试。测试方法是首先编码一定长度的信息，生成相应的音频文件，然后由发送设备播放，同时接收设备开始接收并解码，播放完毕后统计错误比特数，得到误码率。具体来说，我们将十六进制数字 `0x5a5a`，`0xa5a5`，`0x1234`，`0xcdef` 循环编码 15 次，得到一个总长为 960 比特的信息，而接收端只对收到的前 960 个比特进行检验，统计误码率。这样的测试数据可以涵盖不同的频率点以及相位，而测试数据的长度为 120 字节，对于日常的小规模通信，如身份验证等情况，也是足够的。

5.2.2 测试环境与设备

在测试中，我们使用一部 vivo X9 手机作为编码后音频的播放设备，并使用一台 MacBook Pro 作为接收端，在其上进行音频的采样与解码。测试在一间安静的宿舍房间内进行。

5.2.3 测试结果

通信速率及编码方式对误码率的影响 我们固定发送端与接收端的距离为 10cm，并使用手机音量的 50% 播放编码后的音频，使用3.2.2和3.2.3中的两种编码方式以及不同的通信速率，测得两组组误码率，得到的结果如图7所示。图中横轴为通信速率（以 bps 计），纵轴为误码率。



(a) 使用频谱独热编码 (b) 引入相位调制编码

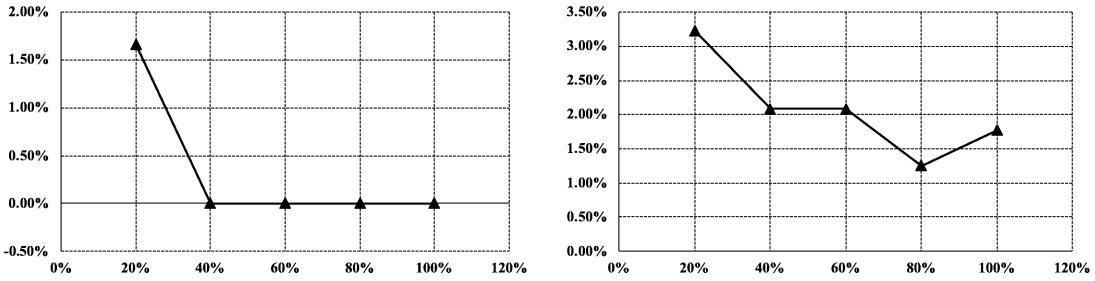
图 7: 通信速率及编码方式对误码率的影响

对于使用频谱独热编码的通信方式，通信速率加快时，误码率的总体趋势是增加的，但可以看出，最高的误码率不超过 0.5%，图中曲线的波动可能是偶然误差。在测试中，当通信速率达到 700bps 时，接收端只能识别出 584 比特的数据，并且大多数是错误的，因此可以认为这种通信方式最高能达到的通信速率在 600 至 700bps 之间。

对于引入相位调制的编码方式，测试的通信速率为 600 至 1500bps，可以看到通信速率为 1200bps 时，误码率是最低的，但也高达 1.5%。通信速率远离 1200bps 时，误码率呈上升趋势。我们认为，在通信速率小于 1200bps 时，相位同步帧插入的间隔仍为 32 字节，但时间上间隔增大了，因此可能无法完全弥补收发双方音频播放与采样速率上的差异，造成误码率升高；而通信速率大于 1200bps 时，基带信号的频率已经超过了 75Hz，因此解码端滑动平均滤波器对其的衰减很明显，导致误码率上升。在测试中，当通信速率达到 1600bps 时，接收端只能识别出 864 比特的数据，并且大多数是错误的，因此可以认为这种通信方式最高能达到的通信速率在 1500 至 1600bps 之间，但在 1200bps 的通信速率下表现最佳。

音量大小对误码率的影响 我们固定发送端与接收端的距离为 10cm，分别使用3.2.2中的编码方式以 400bps 的通信速率进行编码，以及3.2.3中的编码方式以 1200bps 的通信速率进行编码，使用不同的音量大小播放，测得两组组误码率，得到的结果如图8所示。图中横轴为音量百分比，纵轴为误码率。

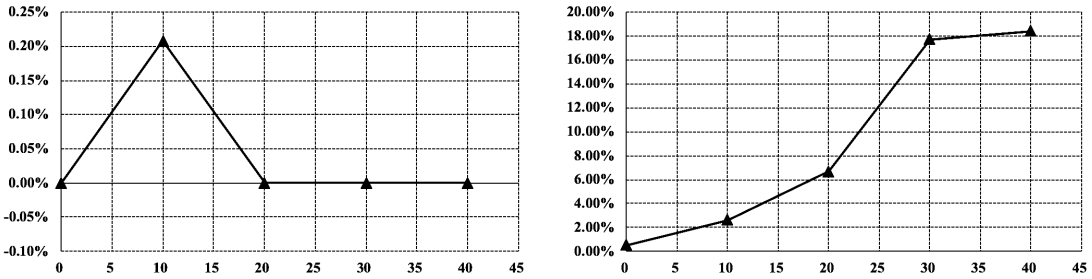
可以看到，对于两种编码方式，音量增大时误码率都呈现降低的趋势，这样的结果很符合直观的预期。并且当音量超过 40% 时，在 10cm 的通信距离上就可以得到能够接受的误码率。



(a) 使用频谱独热编码 (b) 引入相位调制编码

图 8: 音量大小对误码率的影响

通信距离对误码率的影响 我们固定发送端音量大小为 40%，分别使用3.2.2中的编码方式以 400bps 的通信速率进行编码，以及3.2.3中的编码方式以 1200bps 的通信速率进行编码，在不同距离处播放，测得两组组误码率，得到的结果如图9所示。图中横轴为收发端之间的距离（以 cm 计），纵轴为误码率。



(a) 使用频谱独热编码 (b) 引入相位调制编码

图 9: 通信距离对误码率的影响

对于使用频谱独热编码的方式，可以看到在 40cm 以内其误码率都很低，有较好的鲁棒性。而对于引入相位的编码方式，随着通信距离的增加，其误码率迅速上升，并在 20cm 到 30cm 的过程中上升剧烈。我们认为，随着距离的增加，可以影响声波传输的环境因素越来越复杂，多路径问题对相位信息的干扰很大，导致误码率迅速上升。而不使用相位调制时，多路径问题对振幅的影响不大，因此误码率并不会显著增加。另一方面，也可以认为使用相位调制的音频中的信息不容易被远距离的接收端截获，更加安全。

5.2.4 测试中的其他问题

在测试中，我们还注意到一些可能影响使用体验的问题：

- 在两帧的交界处，音频的频率突变会导致音频能量泄漏到整个频谱中，表现为可以听到的爆音。这个问题可以通过对每帧的波形加上梯形窗函数限制来解决，但会使有效帧时间减小，影响通信速率。
- 在通信结束时，扬声器一般不能立即停止振动 [3]，导致接收端会接收到一些额外的数据。这个问题可以通过对数据加上限界符来解决，但并不是物理层需要着重考虑的问题，因此本文对此并不详细讨论。

6 总结与展望

6.1 总结

在本项目中，我们实现了一个完整的基于音频的旁路通信系统，探讨了 3 种编码方式以及 2 种解码方式的原理及优劣，并在最终的实现中采用了其中的 2 种编码方式以及 1 种解码方式。随后，我们测试了通信时使用的通信速率、发送端播放音频的音量大小以及收发双方之间距离 3 种因素对通信误码率的影响，并对得到的变化趋势及其可能的原因进行了简要的探讨。

从结果上看，本项目利用3.2.3中的相位与频率共同调制的方式，结合5中基于时域滤波器的解码算法，最终实现了高达 1500bps 的通信速率，并且通常情况下误码率不超过 2%。对于音频这种高信噪比的传输介质而言，这样的误码率实际上是可接受的。当使用3.2.2中的频谱独热编码方式时，误码率在较大的距离范围内甚至可以保持在 1% 以下，尽管其通信速率只能达到 600bps。我们认为，本项目实现的成果足以替代一部分二维码，在日常生活中的身份核验等场景作为信息传递的途径，例如，可以将通讯软件的 ID 或网页链接编码进音频中进行传递。

6.2 局限性

我们认为，本项目仍存在许多不足之处，包括：

- 编码后音频在两帧交界处会产生爆音，影响用户的体验。
- 引入相位调制后，高频声波的相位对传播距离的变化很敏感，通信时必须保证收发双方相对静止，为使用带来不便。
- 目前的工作仅测试了单工通信的情况，而对于需要双向通信的情况，包括可能的干扰与冲突等没有进行探索。

6.3 展望

本项目可能的进一步工作方向包括以下几点：

- 探索并实现基于音频的双向通信。例如用同样的频率点实现半双工的通信，或者另选一组频率点实现全双工通信。
- 探索频谱利用率与通信速率的平衡。对于本实验使用的解码方式，其低通滤波器的零点频率为通信频率点之间的间隔，因此增大频率点之间的距离，虽然会降低频谱利用率，但低通滤波器的通带也会扩大，支持更高的基带频率。因此可以寻找频谱利用率与通信速率之间的平衡。
- 探索利用一个给定音频本身的信息来编码信息，从而实现在不破坏音乐听感的前提下传递信息的效果，例如可以将信息编码在乐器声音的泛音当中。

7 参考文献

[1] PAN H, CHEN Y C, YANG L, et al. mQRCode: Secure QR Code Using Nonlinearity of Spatial Frequency in Light[C/OL]//2019: 1-18. DOI: 10.1145/3300061.3345428.

[2] GERASIMOV V, BENDER W. Things that talk: Using sound for device-to-device and device-to-human communication[J/OL]. IBM Systems Journal, 2000, 39:530 - 546. DOI: 10.1147/sj.393.0530.

[3] NANDAKUMAR R, CHINTALAPUDI K, PADMANABHAN V, et al. Dhwani: Secure peer-to-peer acoustic NFC[J/OL]. ACM SIGCOMM Computer Communication Review, 2013, 43. DOI: 10.1145/2534169.2486037.

[4] LEE H, KIM T, CHOI J, et al. Chirp signal-based aerial acoustic communication for smart devices [C/OL]//2015: 2407-2415. DOI: 10.1109/INFOCOM.2015.7218629.

[5] VALIENTE A, TRINIDAD A, GARCÍA-BERROCAL J, et al. Extended high-frequency (9-20 kHz) audiometry reference thresholds in 645 healthy subjects[J]. International journal of audiology, 2014, 53.

[6] Apple Inc. vDSP.FFT[EB/OL]. https://developer.apple.com/documentation/accelerate/vdsp/fft.

[7] WANG W, LIU A, SUN K. Device-free gesture tracking using acoustic signals[C/OL]//2016: 82-94. DOI: 10.1145/2973750.2973764.