基于 Simulink 的调幅广播通信系统设计

黄泽熙

(电子科技大学 英才实验学院 611731)

摘 要:调幅广播通信系统是模拟通信系统的一个最经典的实例,对其原理和特性的考察是理解现代通信原理的一个有效途径.本文利用 Simulink 工具箱搭建了一个基于成都调幅广播系统的多信道模拟通信系统模型,讨论了信道噪声强度与衰减情况对系统性能的影响,可用于信号与系统、通信原理等课程的教学案例.

关键词:调幅广播 模拟通信系统 教学案例

Amplitude-Modulation Broadcast System Design Based On Simulink Toolbox

Huang Zexi

(Yingcai Experimental School University of Electronic Science and Technology of China 611731)

Abstract: Amplitude-Modulation (AM) broadcast system is one of the most classic examples of analog communication system, the discussion of the theory and characteristics of which is an effective approach to obtain insight of modern communication principles. This article constructs a multi-channel analog communication system based on AM broadcast system in Chengdu with Simulink toolbox, and discusses the influence of noise intensity and damping rate on the system performance, which may serve as an educational example in courses such as Signal and System and Communication Principles.

Key words: AM broadcast system Analog communication system Educational example

1. 引言

模拟调幅广播系统作为经典的现代模拟通信系统案例(图1),在信号与系统、通信原理等课程中均属于重要的教学内容.大部分教材仅对原理进行讨论[1,2],对具体的仿真仅限于部分模块[3,4]或者整个系统[5].

本文利用 MATLAB 的 Simulink 工具箱,对模拟调幅广播系统的完整实现和信号在各个模块间的流动情况进行讨论,以单输入单输出为基础,结合多输入单输出,多输入多输出的真实情况进行原理阐释.同时,为了使设计结果与现实情况接轨,设计中的频段与成都人民广播电台所采用的频段保持一致,而样本则从真实的广播信号 BBC World Service 里提取,通过对输入与输出音频文件的播放,让学生直观的感受真实信号在调幅广播系统下的传输效果,体会噪声与衰减对信号的影响,并据此引出除杂、同步解调与自动增益控制的必要性.

将原理与仿真实现结合,便于加深学生对调幅广播系统的理解.

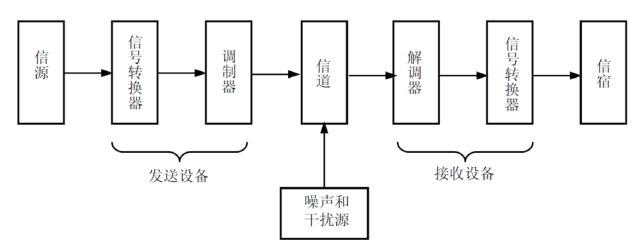


图 1 模拟通信系统方框图

2. 方案与原理

2.1. 仿真环境及输入输出设置

2.1.1. 仿真环境

本设计的整体仿真环境为 MATLAB 8.4.0.150421 (R2014b),输入数据由工作空间读入, 之后输出至工作空间,不限制输出数据的大小,仿真步长设定为固定步长,仿真时间为信号 时域长度,利用 ode3 算法进行仿真.

我国调幅广播波段的频带范围为 $550KH_Z \sim 1605KH_Z$,根据 Nyquist 采样定理,系统仿真时的采样率 f_c 应满足

$$f_{s} > 2f_{\text{max}} = 3210KHz \tag{1.1}$$

为避免交叠,取

$$f_s = 5000KHz \tag{1.2}$$

作为采样频率,那么对应的仿真步长

$$t_s = \frac{1}{f_s} = 2 \times 10^{-7} s \tag{1.3}$$

2.1.2. 音频信号输入和输出设置

利用 MATLAB 的音频接口函数 audioread 和 audiowrite 进行输入与输出,采样频率均设置为默认采样频率,实现的方法见附录 1.

2.1.3. 图形输出设置

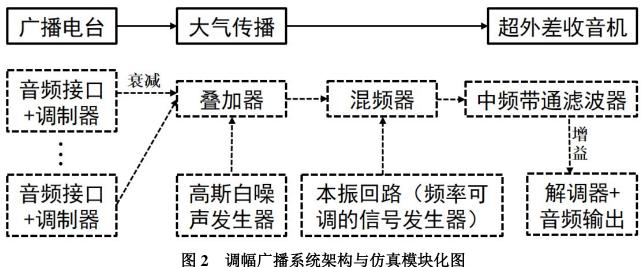
为了使学生能够清楚地认识信号经过各个模块的时域和频域特性的变化,利用 plot 函数和 fft 函数实现信号的时域和频域输出,方法见附录 2.

2.1.4. 单信道多信道转换与输出输出转换设置

对于多信道的情况,需要将各个信道的输入信号合成成一个输入信号矩阵输入系统,将输出矩阵还原回各个信道,方法见附录 3.在各个模块间将本模块的输出信号转换为下一模块的输入信号的方法见附录 4.

2.2. 调幅广播系统整体架构

如图 2 所示,调幅广播系统由广播电台(信源)、大气介质(信道)和超外差收音机(信宿)三个部分组成,对应的模块依次为音频输入接口与调制器、噪声源与叠加器、混频解调器 和 音 频 输 出 接 口 .Simulink 搭 建 的 系 统 模 型 如 图 3 , 对 应 的 模 型 文 件 AM_Multiple_Channels.slx.



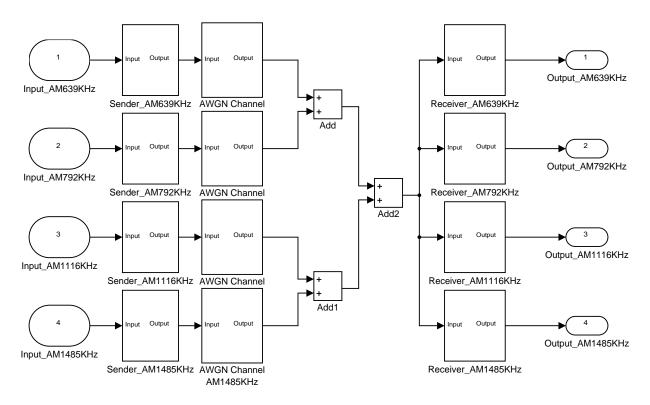


图 3 调幅广播系统整体模型

2.3. 发射端: 调制器

在调幅广播系统中,发射端为广播台,通常情况下,为了降低接收端硬件成本,一般让接受设备采取非同步解调的方式还原信号,因此发射端也采用满足非同步解调的调制方式,模型图如图 4,对应的模型文件为 Sender.slx.

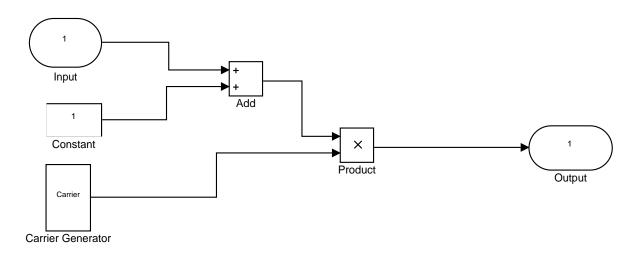


图 4 调幅广播系统的发射端模型

设基带信号为 $x_0(t)$,为一带限有限功率信号,即

$$X(j\omega) = 0, |\omega| > \omega_{M} \tag{3.1}$$

$$\left| x_0(t) \right| \le K \tag{3.2}$$

其中 $X(j\omega)$ 为x(t)的傅里叶变换.在调幅广播系统中,信号总能满足或近似满足这两个条件. 为保证非同步解调时被调信号始终为非负值的条件,利用加法器,将

$$x_1(t) = A + x_0(t) (3.3)$$

作为被调信号, 其中 A 满足

$$A \ge K \tag{3.4}$$

并把

$$m = \frac{K}{A} \le 1 \tag{3.5}$$

称为调制率.

将产生的 $x_1(t)$ 与由载波发生器(由压控振荡器和增益器组成)生成的三角载波 $\cos \omega_c t$ 利用乘法器相乘即调制得到

$$x_2(t) = x_1(t)\cos\omega_c t \tag{3.6}$$

 $x_2(t)$ 就是发射设备实际发射出的信号,在频域上,这相当于把被调信号中心频率移动至 $\pm \omega_c$ 处.为了保证非同步解调的可行性,要求载波频率满足

$$\omega_c >> \omega_M$$
 (3.7)

在本设计中,音频信号的幅度最大值 K=1,信号的截止频率 $\omega_M=6000H_Z$,这一频率可以有效的覆盖人声段和各种乐音的频段,调制率 m=1.各个信道的载波频率与成都人民广播电台的调幅广播波段的频道保持一致,如表 1 所示.

44 IN 18 October 2000 C Maria					
信道编号	载波频率/KHz	对应频道			
1	639	中央人民广播电台中国之声			
2	792	成都人民广播电台新闻广播			
3	1116	四川人民广播电台新闻			
4	1485	成都人民广播电台文化休闲频道(飞扬之声)			

表 1 各信道的载波频率及对应频道

2.4. 信道: 高斯白噪声源

由发射设备调制后发射的信号 $x_2(t)$ 进入空气中,在到达接受设备的过程中会受到诸多干扰.加性高斯白噪声信道是一种最基本的噪声和干扰模型,该信道的模型如图 5,模型文件为 Channel.slx.引入的高斯白噪声 n(t) 满足

$$e(t) \sim N(0, \sigma^2) \tag{4.1}$$

$$\left| E(j\omega) \right|^2 = C \tag{4.2}$$

式(4.1)表明高斯白噪声在任意时刻的值满足高斯分布,式(4.2)表明高斯白噪声的功率谱密度为常数,即不同频率分量的功率相同,C为功率谱密度.

叠加噪声后的信号还需经历一个衰减,设衰减因子为k,则该信道的输入输出关系为

$$x_3(t) = k[x_2(t) + e(t)]$$
 (4.3)

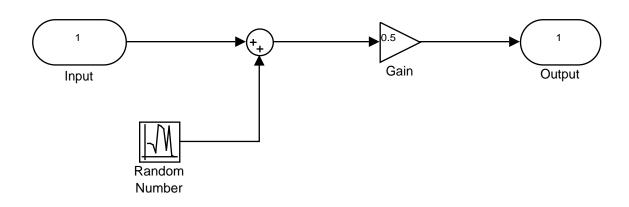


图 5 调幅广播系统的信道模型

在本设计中,噪声的方差分别取 $\sigma_1^2 = 0$, $\sigma_2^2 = 0.01$, $\sigma_3^2 = 0.1$,各个信道的衰减因子 k 如表 2 所示.

信道编号	载波频率/KHz	衰减因子	
1	639	0.5	
2	792	1	
3	1116	0.3	
4	1485	0.9	

表 2 各信道衰减因子

2.5. 接收端: 混频、滤波、解调

在实际情况下,多个广播台利用频分互用的机理同时进行信号广播,因此,接收设备需要能够将各个频段的信号分别提取出来,这就需要频率可调的滤波器.但实际上,频率可调的滤波器实现困难,因此在常用的调幅广播信号接收设备——超外差收音机中,利用混频和固定中心频率的滤波器进行替代,并将得到的信号放大,整个模型如图 6,模型文件为Receiver.slx.

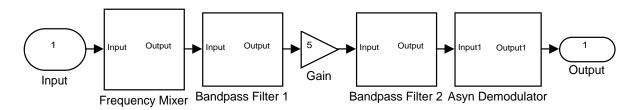


图 6 调幅广播系统的接收端模型

2.5.1. 混频器

超外差收音机的混频部分模型如图 7,模型文件为 Frequency_Mixer.slx.该部分与调制过程类似,由本振信号发生器产生一个频率可调的本振信号

$$l(t) = \cos \omega_t t \tag{5.1}$$

将其输入信号x3(t)作乘积之后即得

$$x_4(t) = x_3(t)\cos\omega_l t \tag{5.2}$$

对于 $x_3(t)$ 中的有效信号部分 $\widetilde{x_3}(t) = x_0(t)\cos\omega_c t$, 乘积后为

$$\widetilde{x_4}(t) = x_0(t)\cos\omega_c t\cos\omega_l t$$

$$= \frac{1}{2}x_0(t)[\cos(\omega_c + \omega_l)t + \cos(\omega_l - \omega_c)t]$$
(5.3)

这就相当于把基带信号频带又移动至 $\omega_l - \omega_c$ 处.

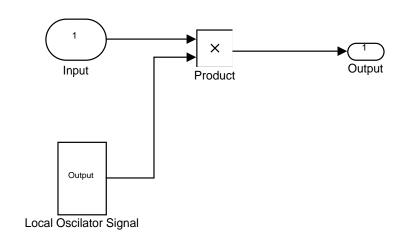


图 7 超外差收音机的混频器模型

2.5.2. 中频滤波部分

超外差收音机的中频滤波部分模型如图 8,模型文件为 Bandpass_Filter.slx.模型中零阶保持的作用是使得之前的模拟信号能够通过后面的数字滤波器,重塑模块的作用是使得输出信号仍然具有数组形式.对于输入的有效信号 $\widetilde{x_4}(t)$,只需令中频带通滤波器的中心频率 ω_0 满足

$$\omega_0 = \omega_l - \omega_c \tag{5.4}$$

就可以仅使载波频率为 ω_c 的信号通过滤波器.这样,只需要调节本振信号频率 ω_l 的取值,就可以让提取不同频率的信号,实现选频的目的.

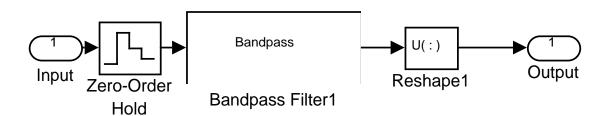


图 8 超外差收音机的中频滤波部分模型

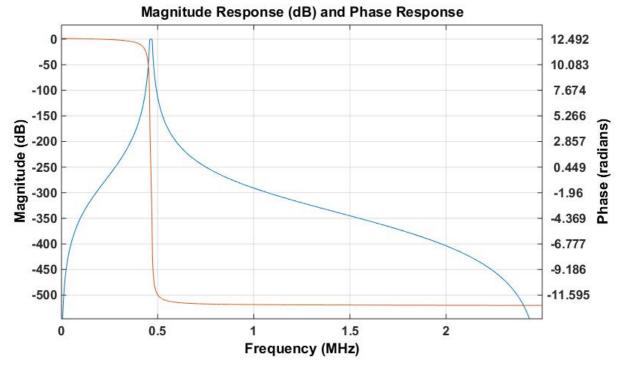


图 9 中频带通滤波器的幅频特性(蓝色)和相频特性(红色)

在本设计中,滤波器的幅频特性与相频特性如图 9 所示,类型为 Butterworth 型,阶数由系统给出(Order mode: Minimum),中心频率取为国家标准的 $465KH_Z$,通带范围为 $459KH_Z \sim 471KH_Z$,截止频率为 $449KH_Z$ 和 $481KH_Z$.由式(5.4)可得各信道对应的混频器的本振频率,如表 3 所示.滤波后的中频增益因子 $\lambda = 5$.

(V3 (V4) / / / / / / / / / / / / / / / / / / /							
	信道编号	载波频率/KHz	本振频率/KHz				
	1	639	1104				
	2	792	1257				
	3	1116	1581				
	4	1485	1950				

表 3 各信道混频器的本振频率

2.5.3. 解调器

选频过后的信号再经过解调即可得到基带信号.超外差收音机一般通过包络检波手段进行非同步解调,其模型如图 10 所示,模型文件为 Asyn_Demodulator.slx.在实际工程中,可以通过简单的 RC 电路就能实现检波的作用.仿真模型采用饱和器和低通滤波器共同实现. 将输入信号记为 $x_5(t)$.饱和器用于将输入信号限制在一定范围内,模拟检波时将输出限制在

(0,+∞)范围内,即

$$x_6(t) = \begin{cases} x_5(t), x_5(t) > 0\\ 0, x_5(t) \le 0 \end{cases}$$
 (5.5)

之后,在通过带宽为基带信号频带宽度的低通滤波器即得最终的输出信号 $x_7(t)$.

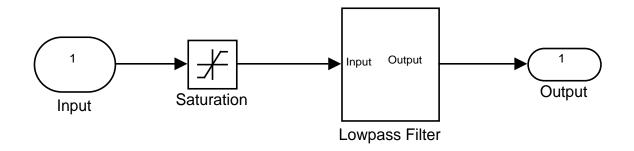


图 10 超外差收音机的解调器模型

在本设计中,低通滤波器的幅频特性与相频特性如图 11 所示,类型为 Butterworth 型,阶数由系统给出(Order mode: Minimum),通带频率取音频信号的上限频率 $6KH_Z$,截止频率取 $7KH_Z$.

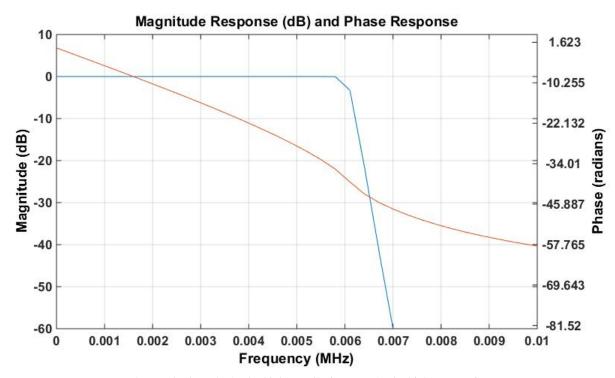


图 11 低通滤波器的幅频特性(蓝色)和相频特性(红色)

3. 结果与讨论

3.1. 单输入单输出

先讨论只有一个信道工作时系统的响应结果.输入音频信号长度为1s,其时域和频域图如图 12,文件为 sample0.wav.该信号采用的信道为信道 2,载波频率为 $792KH_Z$.

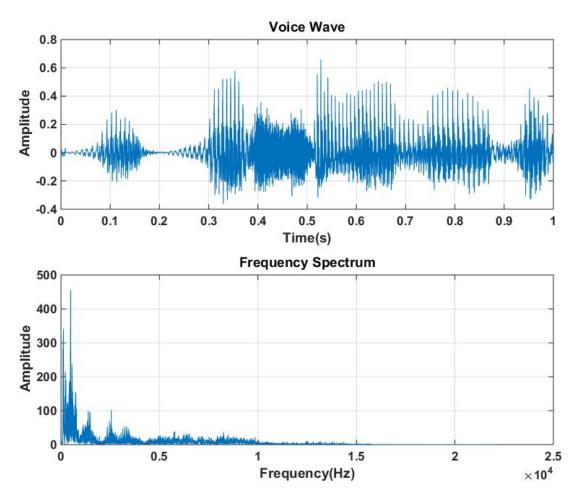


图 12 单输入单输出中输入信号的时域和频域图

3.1.1. 调制器

输入信号经过调制器后的时域频域对比如图 13.从时域图中可以注意到调制后信号的包络线一定程度上反映了输入信号的时域特点,从频域图中可以看出,输入信号的频段整体被搬移至载波频段792*KHz*.

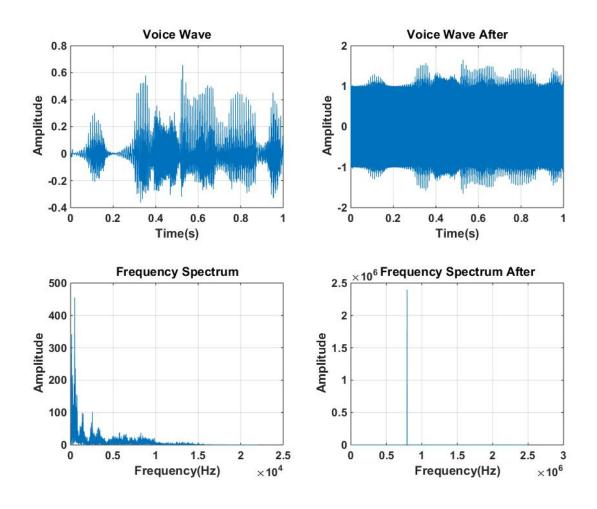


图 13 单输入单输出中信号经过调制器的时域和频域对比图

3.1.2. 信道

调制后的信号经过噪声叠加后得到的时域图如图 14.图中三幅时域图依次对应的噪声的 方差为 $\sigma_1^2 = 0$, $\sigma_2^2 = 0.01$, $\sigma_3^2 = 0.1$.从图中可以看见,加中等程度噪声($\sigma_2^2 = 0.01$)对信号的影响适中,既使原始信号存在一定程度的失真又不至于让信号失真至无法还原,对于音频信号来说是一个可以接受的结果,而加大噪声($\sigma_3^2 = 0.1$)则直接使信号失真严重,无法再被恢复.故在之后的仿真过程中,都取中等程度噪声的输出为输入量.

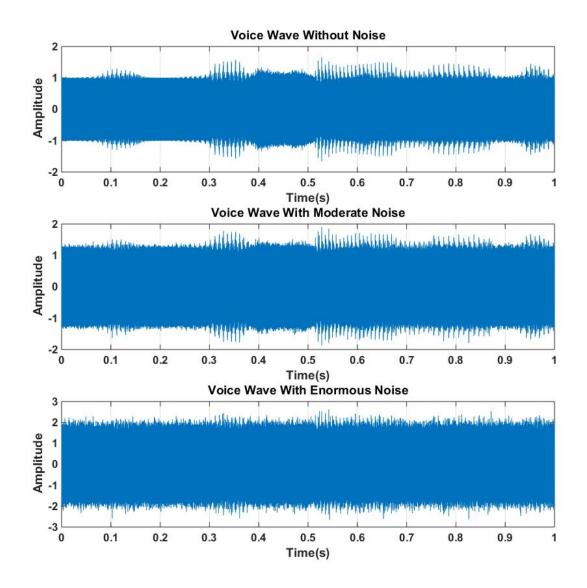


图 14 单输入单输出中输入信号与噪声叠加后的时域图,从上至下噪声方差分别为 $\sigma_1^2 = 0, \sigma_2^2 = 0.01, \sigma_3^2 = 0.1.$

3.1.3. 混频器

通过信道的信号进入接收端超外差收音机,经过混频后的时域频域对比图如图 15.两者时域上的差别并不明显,但从频域图中可以注意到原来处于 $792KH_Z$ 的已调信号被移动至 $2049KH_Z$ 和固定中频段 $465KH_Z$.

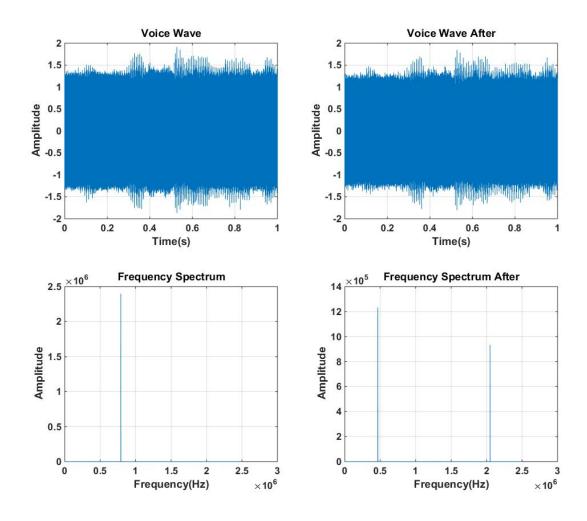


图 15 单输入单输出中信号经过混频器的时域和频域对比图

3.1.4. 中频滤波器

混频后的信号经过中频带通滤波器的时域和频域对比图如图 16 所示.从频域图中可以看出,仅有频率在 465 KHz 附近的信号得到了保留,其他频段的信号都被滤去了.

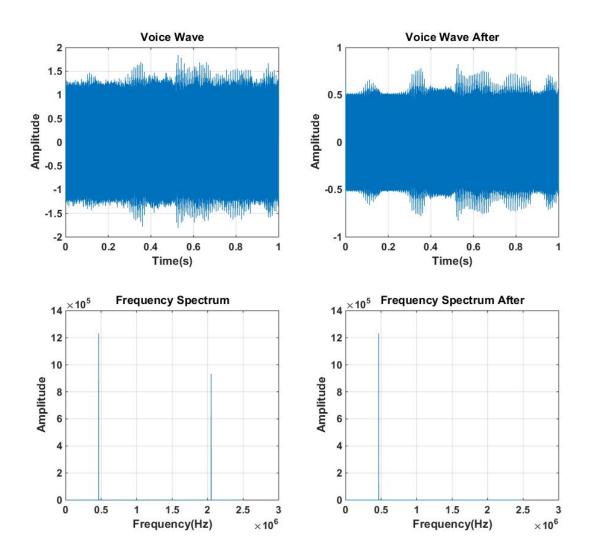


图 16 单输入单输出中信号经过中频滤波器的时域和频域对比图

3.1.5. 解调器

经过中频滤波后的信号经过非同步解调器的时域和频域对比图如图 17 所示.从时域图中可以看出,解调后的信号又基本具备了原始信号的时域波形,已经没有高频分量,而频域图中也可以看见基带信号又从中频段 465 KHz, 移回了音频段 0~6000 Hz.

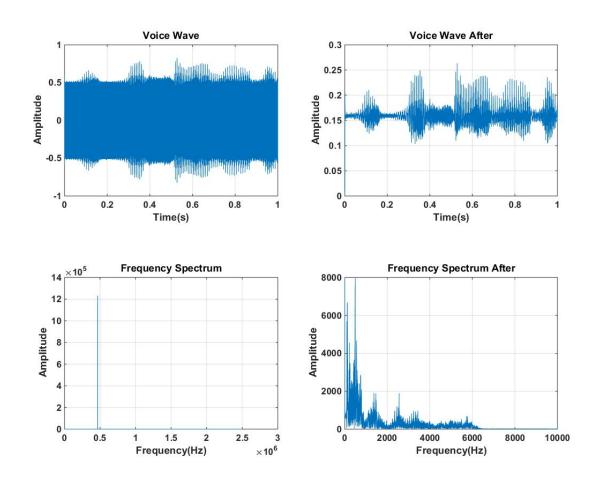


图 17 单输入单输出中信号经过中解调器的时域和频域对比图

3.1.6. 全系统输入输出

基带信号经过整个广播系统的输入与输出的时域和频域对比图如图 18 所示.在时域图中,除了在幅度轴上信号有一定的向上平移外和加入了一些小幅高频噪声外,信号的波形与原始波形基本保持一致.而在时域图中,输出信号的频谱除了在频率 f=0 和 $f>6000H_Z$ 与输入信号有差别和一些高频噪声分量外,音频部分的频谱基本无变化,.这说明调幅广播系统能够将低频信号经过调制进行高频传输后再还原回近似的原始信号.音频的输出文件为 output0.wav. 该输出音频的内容可以如输入音频一样被轻松辨认.

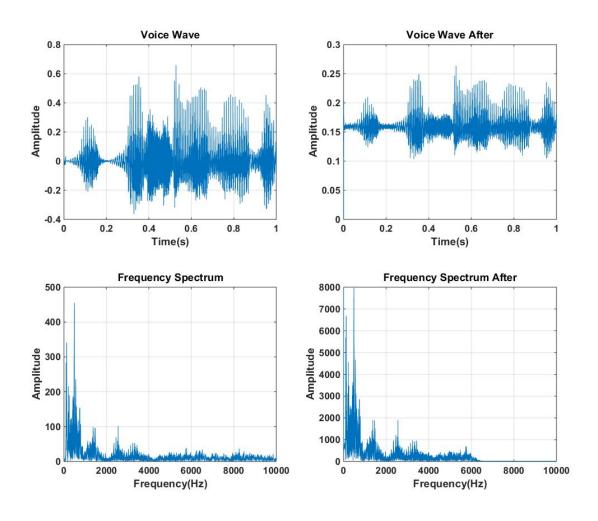


图 18 单输入单输出中信号调幅广播系统的时域和频域对比图

3.2. 多输入单输出

接下来讨论有多个发射端同时工作(对应于多个广播电台同时广播),接收端只输出一个信道的信号的情况.各个信道输入的音频文件均为6s,时域和频域图如图 19.文件分别为 sample1.wav、sample2.wav、sample3.wav、sample4.wav.调节接收端本振频率为 $1257KH_Z$,对应的载波频段为 $792KH_Z$,信道标号为 2.

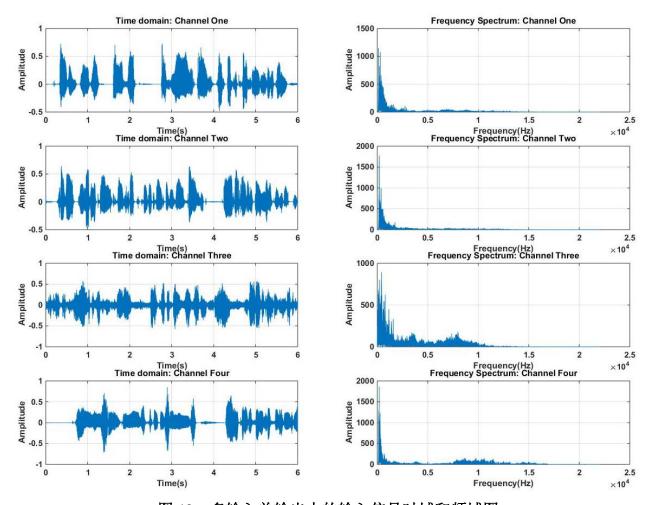


图 19 多输入单输出中的输入信号时域和频域图

3.2.1. 输入端与信道

输入的四个音频信号经过输入端和信道叠加成一个信号的的时域和频域对比图如图 20 所示.将时域图与频域图对比可以发现,虽然时域上四个信号已经完全交叠在一起,无法区分,但是由于采用的不同频率的载波进行调制,频域图上四个调制后的四个信号仍然可以清楚区分.并且,信号的衰减程度也能够很容易地从频域图中得到(载波频率从低到高的四个信号的衰减因子分别为 $k_1=0.5,k_2=1,k_3=0.3,k_4=0.9$).

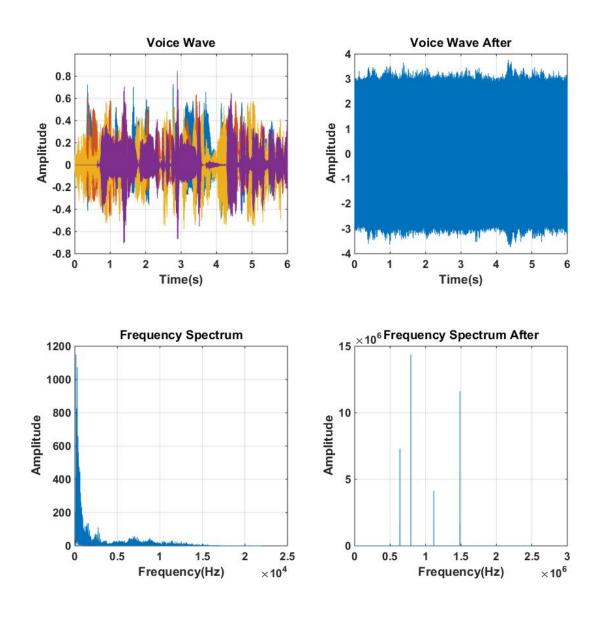


图 20 多输入单输出中的信号经过发射端和信道的时域和频域对比图

3.2.2. 混频器与中频滤波器

各信道叠加后的信号经过混频和中频滤波后的时域和频域对比图为图 21.从图中可以看出,经过混频和滤波后,仅有一个频段的信号得到了保留,并且被移动至了中频段 465*KH*_Z.

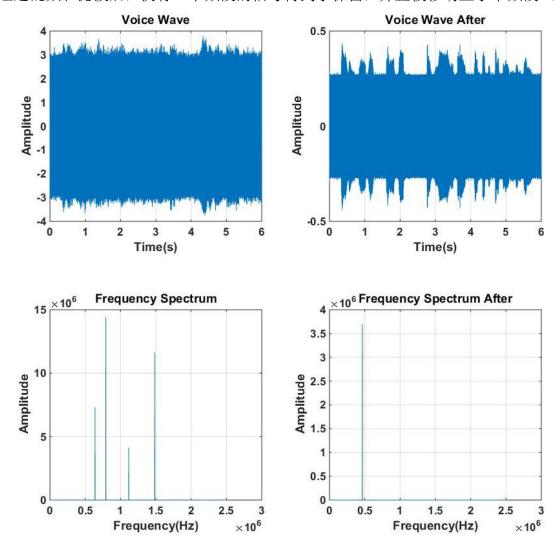


图 21 多输入单输出中的信号经过混频器和中频滤波器的时域和频域对比图

3.2.3. 解调器

选频后的信号经过非同步解调后的时域和频域对比图为图 22.与单输入单输出类似,多输入单输出的输出信号在时域上仅在幅度轴上有所平移,并且加入了小幅的高频噪声,频域上仅在 f=0 和 $f>6000H_Z$ 以及一些高频噪声分量上有差别.

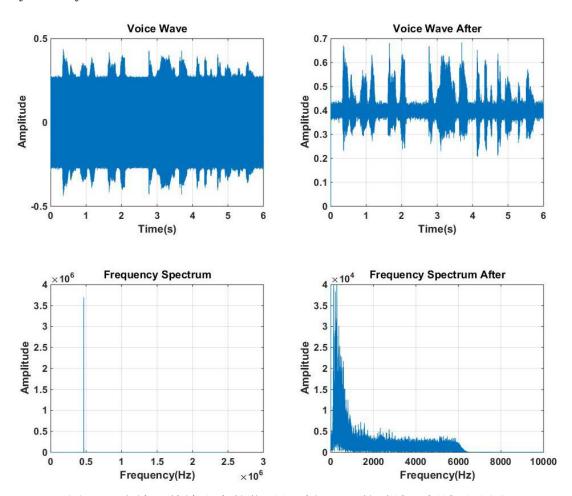
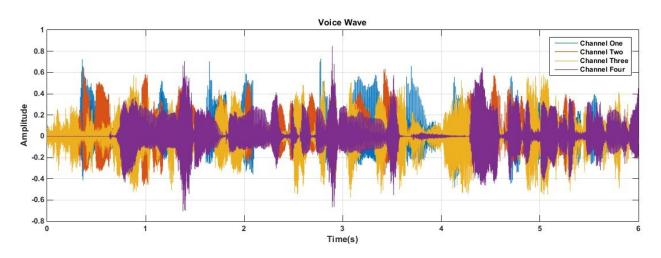


图 22 多输入单输出中的信号经过解调器的时域和频域对比图

3.3. 多输入多输出

多输入多输出的情况是为了检测不同的衰减程度对信号质量的影响.将多输入单输出对应的输入文件作为输入信号,四个本振频率对应四个信道的接收端的输出作为输出信号,得到的输入与输出的时域和频域对比图如图 23 所示,输出文件分别为 output1.wav、output2.wav、output3.wav、output4.wav.从图中可以看见,四个输出信号在时域上被有效的分离开来,但是不同输出信号在幅度轴上的平移量不同,且信噪比有明显差别.图中信道 2 的输出信号(橙色,衰减因子 k_2 =1)的输出信号幅度明显强于信道 3 的输出信号(橙色,衰减因子 k_3 =0.3)的信号幅度,信噪比也是如此.在播放输出音频时,可以很明显感觉到信道 2 的效果要远好于信道 3,其声音更加洪亮,杂音也更加不明显.因此,在实际情况下,常常引入一些除杂模块对输出信号进行滤波除杂,或者采用锁相环进行载波跟踪,使用同步解调方式进行解调,提高信噪比;引入自动增益控制模块来代替中频滤波时的恒定增益(如引入反馈模块),保证声音的响度.



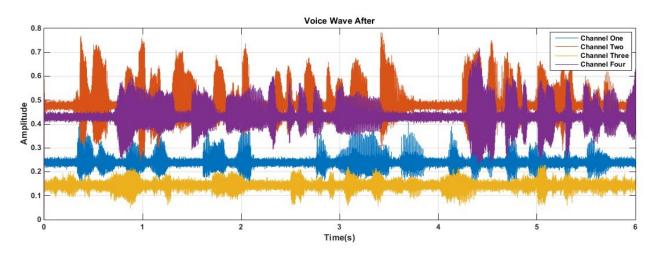


图 23 多输入多输出中的调幅广播系统的输入与输出信号时域对比图

4. 结论

本文基于对模拟调幅广播系统进行了系统仿真,讨论了信号在各个子系统间的变化情况,阐释了模拟调幅广播系统的工作机理,并对噪声与衰落强度对输出信号的影响进行了讨论,借此说明了同步解调、自动增益控制和除杂的必要性.本文提供的案例可以作为信号与系统和通信原理等通信专业基础课程的参考材料,帮助学生加强对现代通信系统的工作原理和思想的理解

5. 参考文献

- [1] A. V. Oppenheim, A. S. Willsky, and S. H. Nawab, *Signals and Systems*: Pearson Education, 1998.
- [2] 闵大镒 and 朱学勇, 信号与系统分析: 电子科技大学出版社, 2000.
- [3] J. R. Buck, M. M. Daniel, and A. C. Singer, *Computer explorations in signals and systems using MATLAB*: Prentice-Hall, Inc., 1997.
- [4] W. Y. Yang, Signals and Systems with MATLAB: Springer Science & Business Media, 2009.
- [5] 邵玉斌, Matlab/Simulink 通信系统建模与仿真实例分析: 清华大学出版社, 2009.

6. 附录

6.1. 输入及输出程序

6.1.1. 音频输入程序

```
%{
Read_Voice.m
读取Wav格式的声音信号
黄泽熙 2015年12月13日
%}
%读入n1秒到n2秒的声音信号
%function [SigTime,t] = Read_Voice(sample_location,n1,n2)
%或者完全读入声音信号
function [SigTime,t] = Read_Voice(sample_location)
%sample_location : 读入声音文件的地址
%SigTime: 信号的时域
%t: 信号的时间范围
%n1,n2:截取的声音信号的秒数
```

%[SigTimeD, Fs] = audioread('Sample.wav'); 读取一个WAVE文件,并返回采样数据到矩阵SigFreqD中,Fs表示采样频率

```
[SigTimeD, Fs] =audioread(sample_location);
%假设声音文件有两个声道,我们只分析第一个声道,如果要分析第二个声道可以改为:
SigTime=SigTimeD(:,2);gTime=SigTimeD(:,2);
SigTime = SigTimeD(:,1);
%取n1到n2秒的信号进行处理(MI中,分别处理四个样本各自6秒的时间)
%SigTime=SigTime((n1*Fs):(n2*Fs));
응或
%取第1秒的信号进行处理(SI中,处理样本的第一秒)
SigTime=SigTime(1:(1*Fs));
SigLength = length(SigTime); %获取声音长度
%可以使用sound函数来播放声音
%sound(SigTime, Fs);
%绘制波形图
t=(0:SigLength-1)/Fs;
figure;
subplot(2,1,1);
plot(t, SigTime), title('Voice Wave'),grid;
xlabel('Time(s)');
ylabel('Amplitude');
%Y = fft(X) 使用快速傅里叶变换算法返回向量X的离散型傅里叶变换
%Y = fft(X,n) 返回n点的离散傅里叶变换,如果向量X的长度小于n,函数要将向量X补零到长度n;如果
向量x的长度大于n,则函数阶段x使之长度为n.若x是矩阵,按相同方法对x进行处理。
응 {
%双边带
Y = fft(SigTime, SigLength);
Y=abs(Y); %计算Y的模
f=(0:SigLength)* Fs/SigLength;
subplot(2,1,2);
plot(f,Y), title('Frequency Spectrum'),grid;
xlabel('Frequency(Hz)');
ylabel('Amplitude');
%单边带
Y = fft(SigTime, SigLength);
halfLength = floor(SigLength/2);
Pyy =Y(1:halfLength + 1); % 只选取前半截部分
Pyy = abs(Pyy);%用于计算复向量的Y的振幅
f = ((0:halfLength)+1)* Fs/SigLength;
```

```
subplot(2,1,2);
plot(f,Pyy), title('Frequency spectrum'),grid;
xlabel('Frequency(Hz)');
ylabel('Amplitude');
응 }
Y = fft(SigTime, SigLength);
halfLength = floor(SigLength/2);
SigFreqAmp =Y(1:halfLength + 1); % 只选取前半截部分
SigFreqAmp = abs(SigFreqAmp);%用于计算复向量的Y的振幅
f = ((0:halfLength)+1)* Fs/SigLength;
subplot(2,1,2);
plot(f,SigFreqAmp), title('Frequency Spectrum'),grid;
xlabel('Frequency(Hz)');
ylabel('Amplitude');
t=t';
f=f';
```

6.1.2. 音频输出程序

```
%{
Write_Voice.m
输出Wav格式的声音信号
黄泽熙 2015年12月15日
%}
function Write_Voice(storage_location,SigTime0,t0)
% storage_location:储存声音信号的地址,例如
%'F:\matlab\Sig&Sys\output.wav'
%SigTime: 输入信号的时域
%t: 输入信号的时域范围
Step0=t(2)-t(1);%输入信号的采样步长
Fs0=1/Step0;%输入信号的采样频率
audiowrite(storage_location,SigTime,Fs0);%以原频率写入
```

6.2. 图形输出程序

6.2.1. 单信号的时域和频域图形输出

```
%{
FFT_With_Graphs.m
信号的快速傅里叶变换,只返回幅频特性,并作图
黄泽熙 2015年12月13日
%}
```

```
function [SigFreqAmp,f]=FFT_With_Graphs(SigTime,t)
%SigFreq: 信号的频域
%f: 信号的频域范围
%SigTime: 信号的时域
%t: 信号的时域范围
왕 {
Example
t=-pi:0.1:pi;
SigTime=cos(t);
왕 }
figure;
%计算采样率
Step=t(2)-t(1);
Fs=1/Step;
subplot(2,1,1);
plot(t,SigTime),title('Voice Wave'),grid;
xlabel('Time(s)');
ylabel('Amplitude');
SigLength=length(SigTime);
SigFreq=fft(SigTime, SigLength);
subplot(2,1,2);
halfLength = floor(SigLength/2);
SigFreqAmp =SigFreq(1:halfLength + 1); %只选取前半截部分
SigFreqAmp = abs(SigFreqAmp);% 用于计算复向量的Y的振幅
f = ((0:halfLength)+1)* Fs/SigLength;
plot(f,SigFreqAmp),title('Frequency Spectrum'),grid;
xlabel('Frequency(Hz)');
ylabel('Amplitude');
```

6.2.2. 输入输出信号的时域图形比较输出

```
%{
Plot_Time_Domain_Only.m
仅比较两个信号的时域图
黄泽熙 2015年12月19日
%}
```

```
function Plot_Time_Domain_Only(SigTime,t,SigTimeOut,tOut)
figure;
subplot(2,1,1);
plot(t,SigTime), title('Voice Wave'),grid;
xlabel('Time(s)');
ylabel('Amplitude');
subplot(2,1,2);
plot(tOut, SigTimeOut), title('Voice Wave After'),grid;
xlabel('Time(s)');
ylabel('Amplitude');
```

6.2.3. 输入输出信号的时域和频域图形比较输出

```
응 {
Plot_two_figures.m
同时画出输入输出信号的波形进行比较
黄泽熙 2015年12月13日
용 }
function Plot_two_figures(SigTime,t,SigTimeOut,tOut)
%SigTime: 输入信号的时域
%t: 输入信号的时域范围
%SigTimeOut: 输出信号的时域
%tOut:输出信号的时域范围
figure;
subplot(2,2,1);
plot(t,SigTime), title('Voice Wave'),grid;
xlabel('Time(s)');
ylabel('Amplitude');
subplot(2,2,2);
plot(tOut, SigTimeOut), title('Voice Wave After'),grid;
xlabel('Time(s)');
ylabel('Amplitude');
%对输入信号
SigLength=length(SigTime);
Y = fft(SigTime, SigLength);
halfLength = floor(SigLength/2);
SigFreqAmp =Y(1:halfLength + 1); % 只选取前半截部分
SigFreqAmp = abs(SigFreqAmp);%用于计算复向量的Y的振幅
Step=t(2)-t(1);
```

```
Fs=1/Step;
f = ((0:halfLength)+1)* Fs/SigLength;
subplot(2,2,3);
plot(f,SigFreqAmp),title('Frequency Spectrum'),grid;
xlabel('Frequency(Hz)');
ylabel('Amplitude');
%对输出信号
SigLengthOut=length(tOut);
YOut = fft(SigTimeOut,SigLengthOut);
halfLengthOut = floor(SigLengthOut/2);
SigFreqAmpOut =YOut(1:halfLengthOut + 1); % 只选取前半截部分
SigFreqAmpOut = abs(SigFreqAmpOut);%用于计算复向量的Y的振幅
StepOut=tOut(2)-tOut(1);
FsOut=1/StepOut;
fOut = ((0:halfLengthOut)+1)* FsOut/SigLengthOut;
subplot(2,2,4);
plot(fOut,SigFreqAmpOut),title('Frequency Spectrum After'),grid;
xlabel('Frequency(Hz)');
ylabel('Amplitude');
  6.2.4. 多信道信号时域和频域图形比较输出
```

```
응 {
Draw_Four_Figures.m
将四个信道信号同时展示出来
黄泽熙 2015年12月15日
응 }
function Draw_Four_Figures(SigTime1,SigTime2,SigTime3,SigTime4,t)
figure;
Step=t(2)-t(1);
Fs=1/Step;
SigLength=length(SigTime1);
halfLength = floor(SigLength/2);
f = ((0:halfLength)+1)* Fs/SigLength;
subplot(4,2,1);
plot(t,SigTime1),title('Time domain: Channel One'),grid;
xlabel('Time(s)');
ylabel('Amplitude');
```

```
subplot(4,2,2);
SigFreq1=fft(SigTime1,SigLength);
SigFreqAmp1 =abs(SigFreq1(1:halfLength + 1));
plot(f,SigFreqAmp1),title('Frequency Spectrum: Channel One'),grid;
xlabel('Frequency(Hz)');
ylabel('Amplitude');
subplot(4,2,3);
plot(t,SigTime2),title('Time domain: Channel Two'),grid;
xlabel('Time(s)');
ylabel('Amplitude');
subplot(4,2,4);
SigFreq2=fft(SigTime2,SigLength);
SigFreqAmp2 =abs(SigFreq2(1:halfLength + 1));
plot(f,SigFreqAmp2),title('Frequency Spectrum: Channel Two'),grid;
xlabel('Frequency(Hz)');
ylabel('Amplitude');
subplot(4,2,5);
plot(t,SigTime3),title('Time domain: Channel Three'),grid;
xlabel('Time(s)');
ylabel('Amplitude');
subplot(4,2,6);
SigFreq3=fft(SigTime3,SigLength);
SigFreqAmp3 =abs(SigFreq3(1:halfLength + 1));
plot(f,SigFreqAmp3),title('Frequency Spectrum: Channel Three'),grid;
xlabel('Frequency(Hz)');
ylabel('Amplitude');
subplot(4,2,7);
plot(t,SigTime4),title('Time domain: Channel Four'),grid;
xlabel('Time(s)');
ylabel('Amplitude');
subplot(4,2,8);
SigFreq4=fft(SigTime4,SigLength);
SigFreqAmp4 =abs(SigFreq4(1:halfLength + 1));
plot(f,SigFreqAmp4),title('Frequency Spectrum: Channel Four'),grid;
xlabel('Frequency(Hz)');
ylabel('Amplitude');
```

6.2.5. 加入不同大小噪声的信号时域图形比较输出

```
응 {
Noise_Comparsison.m
画三个图比较噪声大小对时域的影响
黄泽熙 2015年12月15日
응 }
function Noise_Comparison(SigTime, SigTimeOut1, SigTimeOut2, t)
figure;
Step=t(2)-t(1);
Fs=1/Step;
subplot(3,1,1);
plot(t,SigTime),title('Voice Wave Without Noise'),grid;
xlabel('Time(s)');
ylabel('Amplitude');
subplot(3,1,2);
plot(t,SigTimeOut1),title('Voice Wave With Moderate Noise'),grid;
xlabel('Time(s)');
ylabel('Amplitude');
subplot(3,1,3);
plot(t,SigTimeOut2),title('Voice Wave With Enormous Noise'),grid;
xlabel('Time(s)');
ylabel('Amplitude');
```

6.3. 单信号多信道转换与输入输出转换

6.3.1. 多个信号合成成单一信号矩阵

```
%{
Combine_Voice.m
将多个声音信号合成至一个输入信号中
黄泽熙 2015年12月15日
%}
function SigTime=Combine_Voice(SigTime1,SigTime2,SigTime3,SigTime4)
SigTime=[SigTime1,SigTime2,SigTime3,SigTime4];
```

6.3.2. 单一信号矩阵分解为多个信号

```
%{
Multiple_Output.m
```

```
将最终信号化为多个输出
黄泽熙 2015年12月15日
%}
SigTimeOut1=SigTimeOut(:,1);
SigTimeOut2=SigTimeOut(:,2);
SigTimeOut3=SigTimeOut(:,3);
SigTimeOut4=SigTimeOut(:,4);
```

6.4. 模块间输入输出转换

```
%{
Cleanings.m
将输出信号作为新的输入信号,并清空空间
黄泽熙 2015年12月13日
%}
SigTime=SigTimeOut;
t=tOut;
clear SigTimeOut tOut f fout SigFreq SigFreqOut tout yout;
clc;
```

6.5. 用户说明

Amplitude-Modulation Broadcast System Design Based On Simulink Toolbox 基于 Simulink 的调幅广播通信系统设计 Copyright <c> 2015 Eitima (Huang_Zexi). All rights reserved. 2015 黄泽熙. 保留所有权利.

一、设计综述

本设计总共包括以下六个部分:

- 0.本用户说明.
- 1.设计报告: .pdf 文档.
- *2.答辩演示文稿: .pptx 格式文稿
- *3.MATLAB 程序源代码: .m 文件
- *4.Simulink 模型: .slx 格式模型
- *5.输入样本和输出文件:.wav 格式单声道音频
- *如有需要,请联系作者获得相应文件.
- 二、仿真说明

本设计的仿真环境为 MATLAB 8.4.0.150421 (R2014b).在进行仿真前,请确保你已经熟悉了本设计的基本原理和设计报告中仿真环境的设置.

每个 Simulink 模块均从工作空间中读入[t,SigTime]作为输入, t 为输入信号的时域范围, SigTime 为输入信号的时域,并向工作空间输出[tOut,SigTimeOut], t 为输出信号的时域范围, SigTime 为输出信号的时域.

在仿真开始前,可利用用于输入的的 Read_Voice.m 文件从所给音频样本.wav 文件中读入声音信号,之后可以直接运行整个系统模块(AM.slx/Am_Multiple_Channels.slx)得到最终的输出结果.对于系统的各个参数,可以在.slx 文件修改.之后利用用于输出的 Write_Voice.m 文件将得出的结果输出成音频文件播放,检查系统性能.若需逐个检测各个模块的功能,可将输入信号逐个通过每个模块,通过后利用 Cleanings.m 文件完成输入输出的转换再通过下一个模块.

如果需要观察系统/各模块的信号的时域与频域特性,可以运行模块间的时域频域作图函数(如 Plot_Two_Figures.m)和特殊模块的作图函数(如 Noise_Comparison.m)进行图形生成.

三、联系作者

如果在阅读或者仿真过程中遇到问题及对本设计有任何建议,欢迎联系作者.

邮箱: Eitima@163.com