**\*\*\*\*\*\*大学**

题 目：数字化语音存储与回放系统的设计

学生姓名： \*\*\*\*、\*\*\*\*、\*\*\*\*

指导教师： \*\*\*\*\*\*\*

完成时间： \*\*\*\*年\*\*月\*\*日

**目录**

1摘要·············································3

2系统方案论证·····································3

3系统硬件设计·····································4

3.1拾音器设计···································5

3.2前置放大器设计·······························5

3.3偏置电路设计·································5

3.4滤波器设计···································6

3.5功率放大器···································6

4系统软件设计·····································7

5参考文献·········································8

6 程序源码········································9

**1摘要**

本设计选用STM32单片机作为控制核心，加上其内置的A/D和D/A模块以及外部的滤波、放大等电路，制作了一款失真小容量大的数字化语音存储回放系统。其优点在于使用了外置512M的SD卡，使系统的存储量显著提高，此外其滤波电路部分采用两个均为10阶Sallen-Key的低通滤波器和高通滤波器级联构成，使系统的失真度得到有效控制。

经过实际测量，本设计达到了题目基本部分要求，并且实现了发挥部分录音时间超过20S的指标。

**2 系统方案论证**

语音编码方案：

人耳能听到的声音是一种频率范围为20 Hz～20000 Hz ,而一般语音频率最高为3400 Hz。语音的采集是指语音声波信号经麦克风和放大器转换成有一定幅度的模拟量电信号,然后再转换成数字量的全过程。题目要求300～3 400 Hz ,语音采集的采样频率为8 kHz，所以我们采用实时副值采样法。

实时副值采样法采样过程如图1.1所示：

抽样

量化

存储

**图1.1 采样过程**

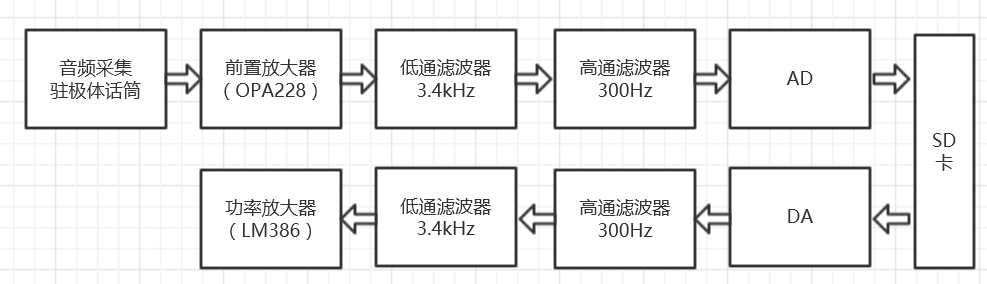
单片机语音生成过程,可以看成是语音采集过程的逆过程,但又不是原封不动地恢复原来的语音,而是对原来语音的可控制、可重组的实时恢复。

单片机选用STM32其自带A/D和D/A转换模块，只要在单片机输入位置加入偏置电路将电压调节至0-3.3V即可。

数据存储器的选择。当采样频率fs=8kHz，字长为8位时，一秒钟的语音需要8K字节的存储空间，则存储器至少需要有80k容量，由于单片机存储空间有限，且储存方式较为复杂，在这里我们选用单片机接外置512M大小的SD卡存储。

**3 系统硬件设计**

数字化语音存储与回放系统的基本思想是通过拾音器将声音信号转化成电信号，再经过放大器放大，然后通过滤波器滤波，模拟语音信号通过模数转换（A/D）转换成数字信号，再通过单片机控制将数据从SD卡中读出，然后通过数模转换（D/A）转换成模拟信号，经放大再扬声器或耳机上输出。整个系统框架图如图所示：

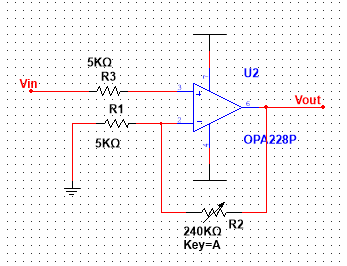


由输入通道、STM32单片机和输出通道等部分组成。输入通道部分由拾音器、前置放大电路和滤波器组成；输出通道由滤波器、后级功率放大电路组成[9]。拾音器输出的毫伏信号实测其范围约为10～30mV，此电信号太小不能够进行采样，后级A/D转换输入信号的动态范围为0～3.3V，此处将信号通过一增益为46dB的放大器，将其放大到伏特量级，考虑到语音信号的固有特点，将低于300Hz和高于3.4kHz的分量滤掉后语音质量仍然良好。将带通滤波器设计为典型的300Hz～3.4kHz,输出级带通滤波器也为300Hz～3.4kHz,这样既可滤掉低频分量又可滤掉D/A转换带来的高频分量，很好的滤除掉噪声。

**3.1拾音器设计**

驻极体话筒体积小、噪声小、频带宽、灵敏度高，是一种常用的话音传感器。本方案采用驻极体话筒收集音频信号。实际测得驻极体话筒正常工作时输出最大电压为30mV，不考虑信号衰减等因素，至少需放大110倍才可达到单片机最大输入电压值。

**3.2前置放大器设计**

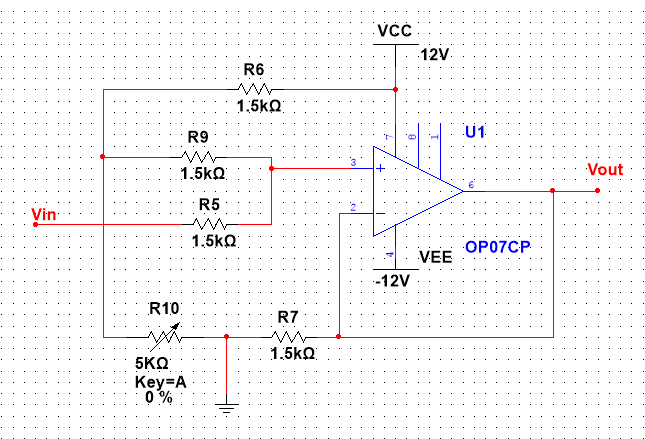
拾音器输出的毫伏信号实测其范围约为10～30mv，信号太小不能够进行采样，后级A/D转换输入信号的动态范围为0～3.3V，因为题目要求前置放大器增益应达到46dB，所以单级放大需要高性能运放。OPA228是一款高精度低噪声运算放大器，查阅其数据手册后发现采用OPA228单级放大即可达到题目要求的46dB增益，单级放大所需的器件数少，引入的噪声也更少，稳定性更高，因此选定该方案。

电路图如图所示：

前置放大器其输出为Vout = （1+R2/R1）\*Vin。

R2为可调，故放大电压最大值可以满足要求。

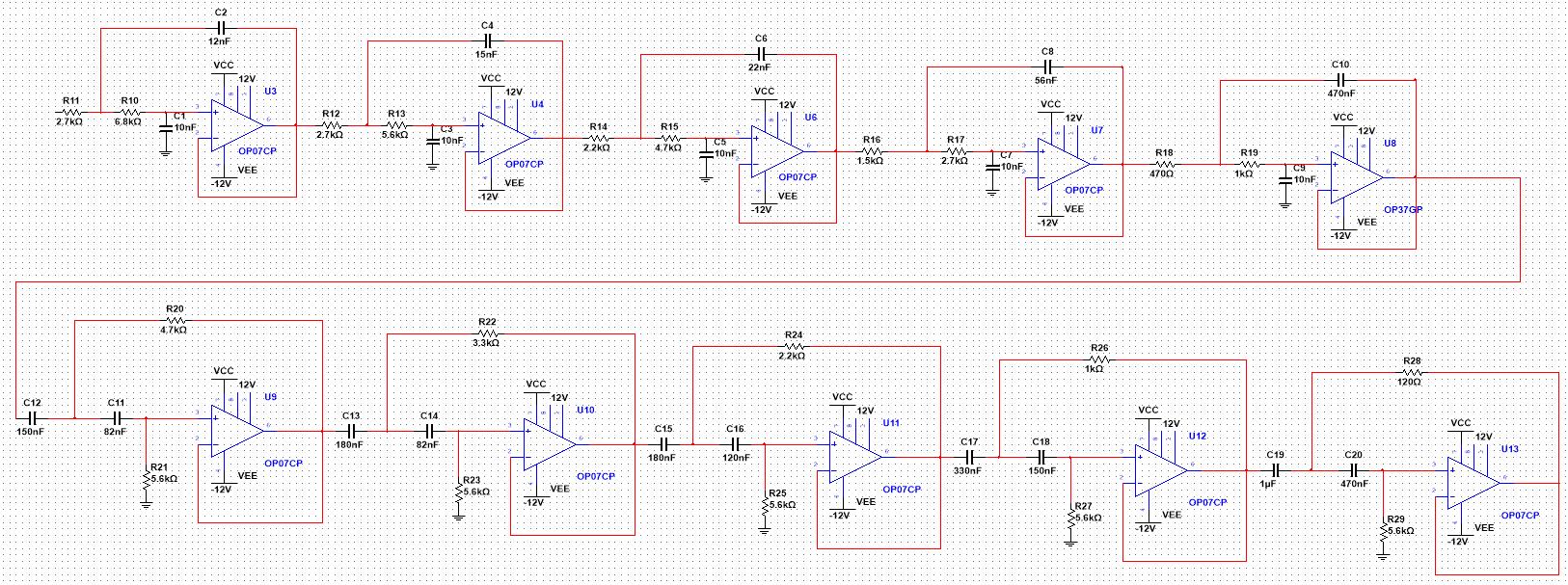
**3.3偏置电路设计：**

因为A/D转换能够接受的电压在0～3.3V之间，所以在带通滤波之后还设计了偏置模块，电路图：

**3.4滤波器设计：**

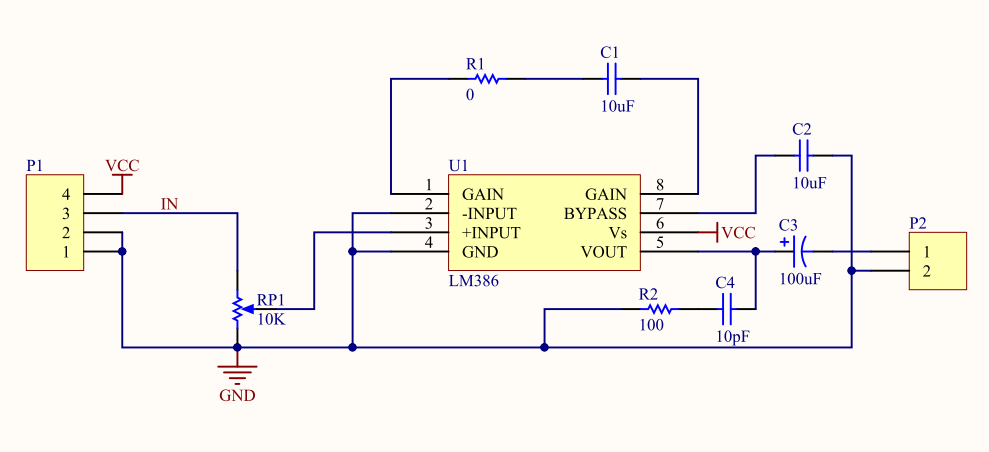
根据题目要求应设计通带频率为300Hz～3.4kHz的带通滤波器，其上下截止频率之比3400/300=11.3>>2，因此是属于宽带滤波器，无法采用一般带通滤波器的设计方法设计。我们采用低通滤波器与高通滤波器级联的方式构成带通滤波器，为实现更好的滤波效果，适当提高了滤波器的阶数。如下图所示：

宽带滤波器由两个均为10阶Sallen-Key的低通滤波器和高通滤波器级联构成



**3.5功率放大器**

从单片机输出的电压为0～3.3V，但是输出的电流较小，故需要接功率放大器才可以驱动扬声器，末端采用LM386功率放大器在1脚和8脚之间接5k的滑动变阻器和10uF的电容使电路增益在20～200倍可调，最大放大倍数大于题目所给的40dB，符合要求。电路图如下：

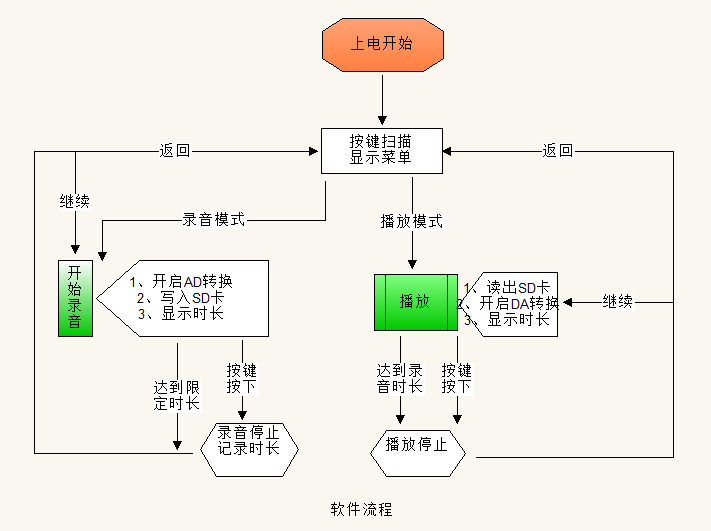


**4系统软件设计**

本系统使用S T M 3 2单片机作控制器，S T M 3 2内部集成了A D / D A模块功能，最高采样率为1 M H z，所以不需外加转换芯片，完全可以满足题目要求，因此使用内置A D / D A转换器。存储器使用容量为512M的S D卡，容量满足发挥部分录音2 0 s以上的需求，因此选用该方案。原理如下：

进入显示控制子程序，如果是功能选择按键就按照键值切换当前指示位置，如果是确认键就执行相应的子程序，系统包括三种模式：录音、播放和直通，并且支持暂停功能。最大录音时间可以达到2.5小时,可以显示当前录音时间和播放时间。三种模式功能不可同时开启。

存储过程中SD卡按页写入，一页512个字节，当写满一页后，置位一个标志位，在主程序里进行写入操作，当暂停或结束按下时，就将当前的写入页数写到SD卡的第一页里，播放时先从第一页读出写入的页数，然后进行播放操作，从而实现自动记录录音时长的功能。



**5参考文献**

1.《模拟电子技术基础》童诗白 华成英

2.《电路设计》（美）赛尓吉欧·佛朗哥

3.《新型功率放大锦集》人民出版社

4.《电子电路百科全书》科学出版社

5.《运算放大器权威指南》孙宗晓 译

6.《OP放大电路设计》科学出版社

7.《基于STM32ARM处理器的编程技术》西安电子科技大学出版社

**6 部分程序源码**

int main(void)

{

/\*------------------------------------\*/

u8 KEY\_Scan\_sult,LCD\_Show;

/\*------------------------------------\*/

delay\_init();

uart\_init(115200);

KEY\_Init();

LCD\_Init();

GPIOS\_Init();

LCD\_Show\_Inhand();

SD\_Init();

ADC\_init();

Dac1\_Init();

TIM2\_Int\_Init(1999,35999);

TIM3\_Int\_Init(9,799);

TIM4\_Int\_Init(9,799);

TIM5\_Int\_Init(9,799);

while(1)

{

/\*------------------------------------\*/

KEY\_Scan\_sult=KEY\_Scan(); LCD\_Show=LCD\_Show\_Munes(KEY\_Scan\_sult);

DEAL\_Keys\_number(LCD\_Show);

/\*------------------------------------\*/

if(recording\_s)

{

recording\_s=0;

SD\_WriteDisk(ADC\_Date\_Buff,page\_count,1);

page\_count++;

if(page\_count>=86\*SEC)

{

DEAL\_Keys\_number(LCD\_Show\_Munes(6));

}

}

/\*------------------------------------\*/

if(play\_s)

{

play\_s = 0 ;

while(SD\_ReadDisk(DAC\_Date\_Buff,page\_count,1)!=0);

page\_count++;

if(page\_count>=last\_page)

{

DEAL\_Keys\_number(LCD\_Show\_Munes(6));

page\_count = 1 ;

sec\_time = 0;

}

}

}

}