

目 录

|   |    |
|---|----|
| One, module hardware description                                | 2  |
| Second, module wiring diagram                                   | 3  |
| Third, testing methods  | 8  |
| Appendix I:SPI-FLASH Capacity and the length of the audio table | 9  |
| Third, the rate of conversion                                   | 9  |
| Four, module schematics   | 13 |
| Four examples, procedures                                       | 14 |

## 1, module hardware description

Module power supply modules the ideal working voltage as 4.2V。 So users are using 5V Power supply, it is recommended that thread a diode

## 2, Led module [Power state]

| 工作状态 | 下载模式 | 播放语音 | 暂停 | 睡眠 |
|------|------|------|----|----|
| 状态   | 快闪   | 慢闪   | 常亮 | 灭  |

备注：正常工作状态指示灯

## 3, Led module [Work State]

| 工作状态 | 一对一可打断 | 抬起停止          | 一对一不可打断        | 标准MP3功能       |
|------|--------|---------------|----------------|---------------|
| 状态   | 常亮2S   | 快闪2S[100ms取反] | 中慢闪2S[200ms取反] | 慢闪2S[500ms取反] |

备注：此时指示灯，只在上电初始化的时候，指示2秒

## 4, Audio transmission description

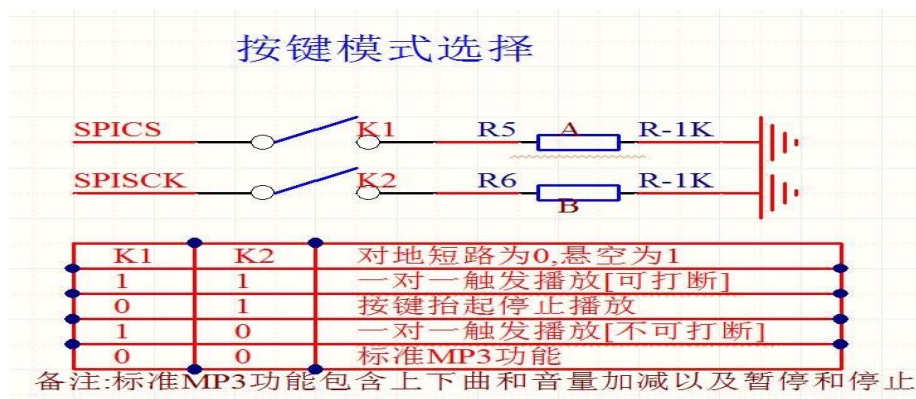
1. SPK1 和 SPK2 Received the trumpet two Swiss, regardless of polarity, note: only allow 2W Following Horn
2. Amplifier or headphones connected directly DAC—R 和 DAC—L Ends Note: common ground (GND)

## 5, Module debug instructions

(1), Our module is the default plug USB Line, enter download mode. When a user updates after your speech, triggered a IO , You can exit the download mode and return to normal working condition

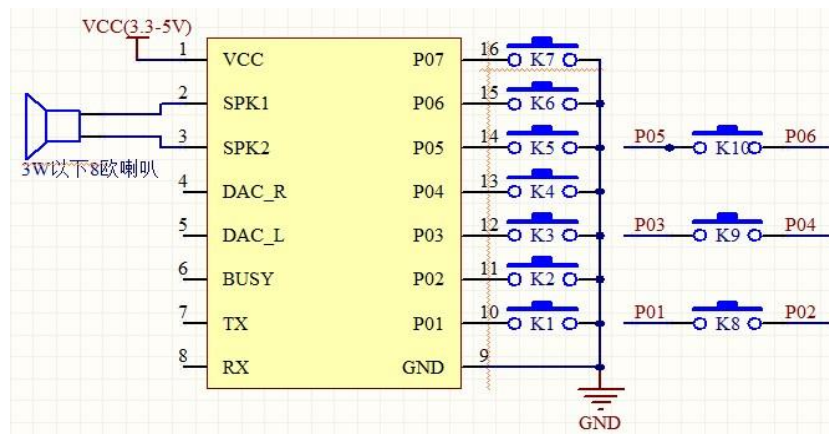
(2), Modules before shipping, have been download and test the sound, before the customers get the module, do not hurry to download sound, first by way of small speakers or amplifier or headphones, plug USB (When the power supply), short circuit to ground triggers trigger sound, then change the sound,

Note that update voice browsing open is completed, once again updated, because after every update module memory wiped clean burning

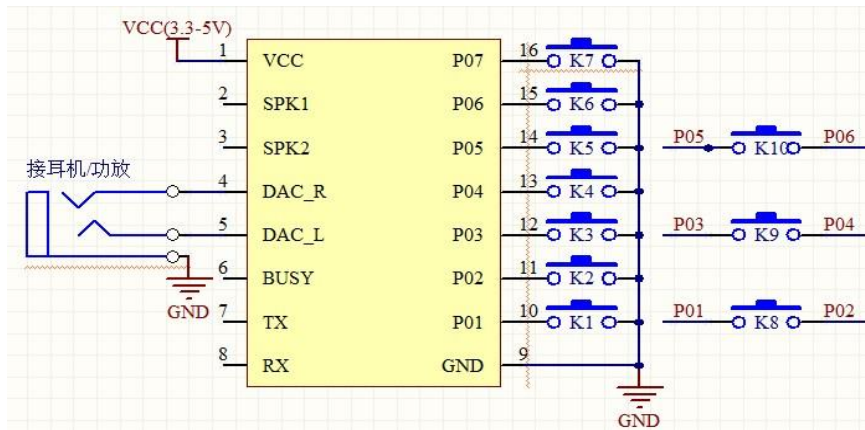


### Second, module wiring diagram

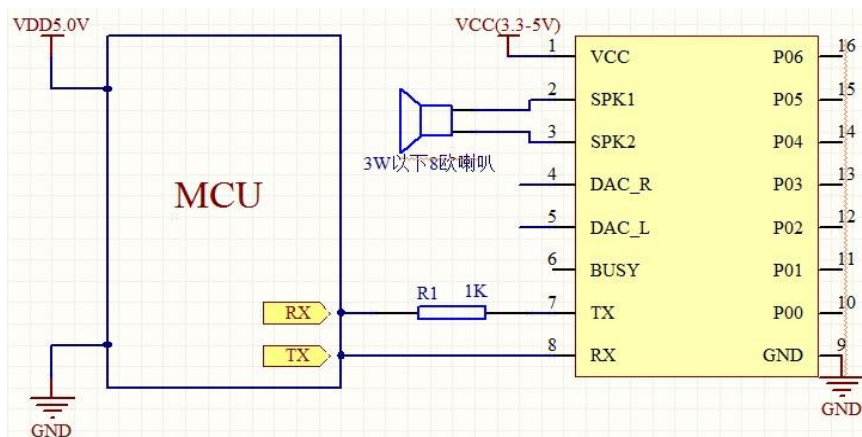
Press the key mode-trumpet



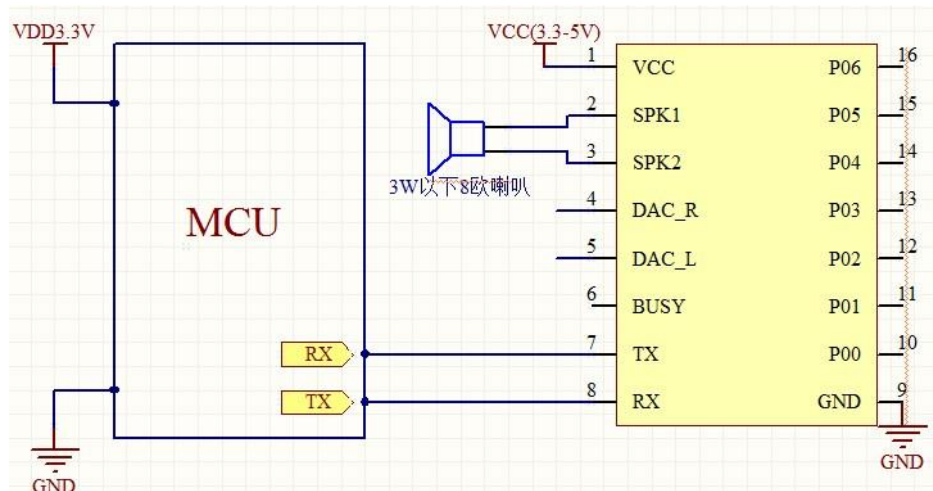
Press the key mode-amplifier or headphones



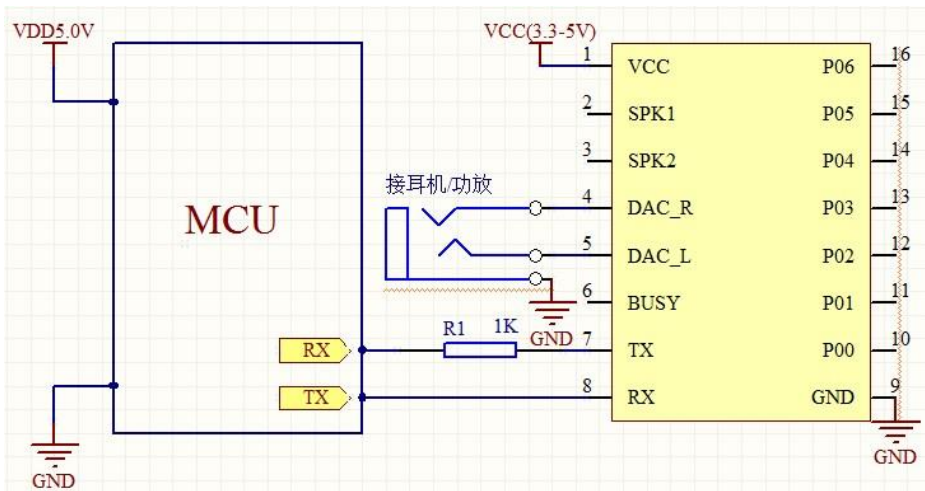
Single-chip microcomputer 5V-Trumpet



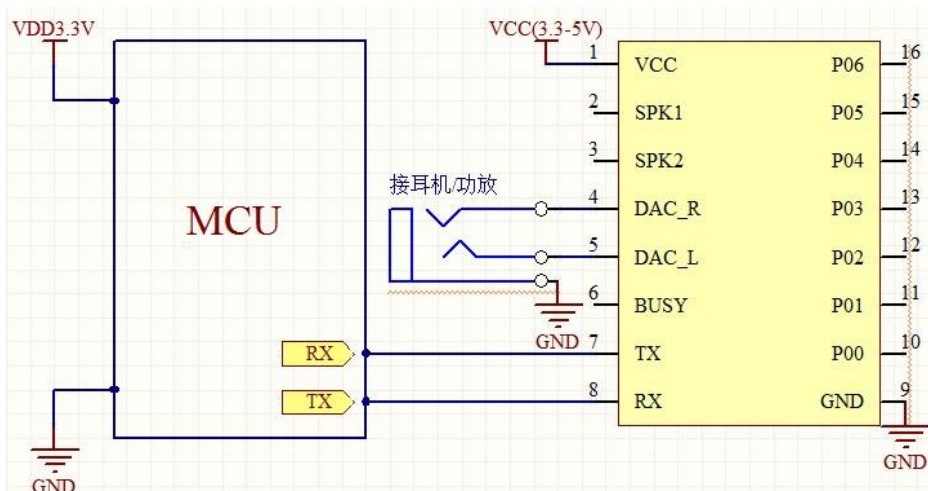
Single-chip microcomputer 3.3V-Trumpet



Single-chip microcomputer 5V-Headset or amplifier



Single-chip microcomputer 5V-Headset or amplifier



| Testing serial debugging assistant | Send command[With checksum]   | Send command[Without calibration] | Notes                           |
|------------------------------------|-------------------------------|-----------------------------------|---------------------------------|
| [The next song]                    | 7E FF 06 01 00 00 00 FE FA EF | 7E FF 06 01 00 00 00 EF           |                                 |
| [Previous track]                   | 7E FF 06 02 00 00 00 FE F9 EF | 7E FF 06 02 00 00 00 EF           |                                 |
| [The specified track]              | 7E FF 06 03 00 00 01 FE F7 EF | 7E FF 06 03 00 00 01 EF           | Specifies the first play        |
|                                    | 7E FF 06 03 00 00 02 FE F6 EF | 7E FF 06 03 00 00 02 EF           | Specifies the second song       |
|                                    | 7E FF 06 03 00 00 0A FE EE EF | 7E FF 06 03 00 00 0A EF           | XAF10 首                         |
| [Specifies the volume]             | 7E FF 06 06 00 00 1E FE D7 EF | 7E FF 06 06 00 00 1E EF           | Specifies that the volume is30级 |
| [The specifiedEQ]                  | 7E FF 06 07 00 00 01 FE F3 EF | 7E FF 06 07 00 00 01 EF           | Reservation                     |
| [Loop playlist]                    | 7E FF 06 08 00 00 01 FE F2 EF | 7E FF 06 08 00 00 01 EF           | Loop the first                  |
|                                    | 7E FF 06 08 00 00 02 FE F1 EF | 7E FF 06 08 00 00 02 EF           | Loops II                        |
|                                    | 7E FF 06 08 00 00 0A FE E9 EF | 7E FF 06 08 00 00 0A EF           | Looping the tenth               |
| [Specifies the playback device]    | 7E FF 06 09 00 00 01 FE F1 EF | 7E FF 06 09 00 00 01 EF           | Specifies the playbackFLASH     |
| [Enter sleep mode]                 | 7E FF 06 0A 00 00 00 FE F1 EF | 7E FF 06 0A 00 00 00 EF           |                                 |
| [Wake up sleeping]                 | 7E FF 06 0B 00 00 00 FE F0 EF | 7E FF 06 0B 00 00 00 EF           |                                 |
| [Module reset]                     | 7E FF 06 0C 00 00 00 FE EF EF | 7E FF 06 0C 00 00 00 EF           |                                 |
| [Play]                             | 7E FF 06 0D 00 00 00 FE EE EF | 7E FF 06 0D 00 00 00 EF           |                                 |
| [Pause]                            | 7E FF 06 0E 00 00 00 FE ED EF | 7E FF 06 0E 00 00 00 EF           |                                 |
| To stop playback                   | 7E FF 06 16 00 00 00 FE E5 EF | 7E FF 06 16 00 00 00 EF           | Stop software decoding          |
|                                    |                               |                                   |                                 |
|                                    |                               |                                   |                                 |
|                                    |                               |                                   |                                 |

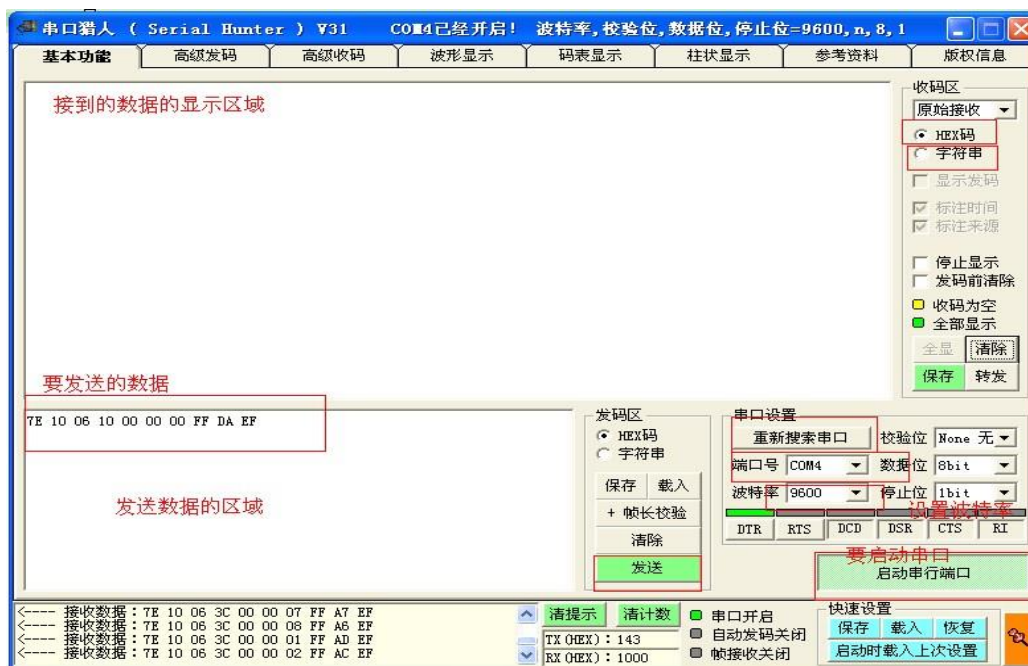
|   |  |  |   |
|---|--|--|---|
| Combination play                            | 7E FF 09 21 00 05 01 02 03 04<br>FE C8 EF          | 7E FF 09 21 00 05 01 02 03<br>04 EF          | Combination play5、1、2、3<br>、4               |
| Combination play                            | 7E FF 0C 21 00 05 01 02 03 04<br>06 07 08 FE B0 EF | 7E FF 0C 21 00 05 01 02 03<br>04 06 07 08 EF | Combination play5、1、2、3<br>、4、<br>6、7、8     |
|   |  |  |   |
|   |  |  |   |
| Play with volume                            | 7E FF 06 22 00 1E 01 FE BA EF                      | 7E FF 06 22 00 1E 01 EF                      | 30Level volume play1 曲                      |
|   | 7E FF 06 22 00 0F 01 FE C9 EF                      | 7E FF 06 22 00 0F 01 EF                      | 15Level volume play1 曲                      |
|   | 7E FF 06 22 00 0F 02 FE C8 EF                      | 7E FF 06 22 00 0F 02 EF                      | 15Level volume play2 曲                      |
|   |  |  |   |
| Play it a specified folder<br>tracks        | 7E FF 06 30 00 01 01 FE C9 EF                      | 7E FF 06 30 00 01 01 EF                      | FOLDER1Subsection1 曲                        |
|   | 7E FF 06 30 00 02 01 FE C8 EF                      | 7E FF 06 30 00 02 01 EF                      | FOLDER2Subsection1 曲                        |
|   | 7E FF 06 30 00 03 01 FE C7 EF                      | 7E FF 06 30 00 03 01 EF                      | FOLDER3Subsection1 曲                        |
|   | 7E FF 06 30 00 03 03 FE C5 EF                      | 7E FF 06 30 00 03 03 EF                      | FOLDER3Subsection3 曲                        |
|   |  |  |   |
| Specifies the folder tracks<br>cycle sowing | 7E FF 06 31 00 03 03 FE C4 EF                      | 7E FF 06 31 00 03 03 EF                      | FOLDER3 第 3Music looping                    |
|   | 7E FF 06 31 00 01 03 FE C6 EF                      | 7E FF 06 31 00 01 03 EF                      | FOLDER1Subsection3Song<br>cycle sowing<br>放 |
|   |  |  |   |
| Specify a folder loop                       | 7E FF 06 32 00 03 01 FE C5 EF                      | 7E FF 06 32 00 03 01 EF                      | FOLDER3Subsection1Song<br>start looping     |
|   |  |  |   |

|   |                               |                         |                                     |
|---|-------------------------------|-------------------------|-------------------------------------|
|   |                               |                         |                                     |
|   |                               |                         |                                     |
| Query the current state                           | 7E FF 06 42 00 00 00 FE B9 EF | 7E FF 06 42 00 00 00 EF |                                     |
| [Query volume]                                    | 7E FF 06 43 00 00 00 FE B8 EF | 7E FF 06 43 00 00 00 EF |                                     |
| [Query the currentEQ]                             | 7E FF 06 44 00 00 00 FE B7 EF | 7E FF 06 44 00 00 00 EF |                                     |
| The current play mode                             | 7E FF 06 47 00 00 00 FE B4 EF | 7E FF 06 47 00 00 00 EF |                                     |
| FLASHTotal number of files                        | 7E FF 06 49 00 00 00 FE B2 EF | 7E FF 06 49 00 00 00 EF |                                     |
| FLASHCurrent folder track pointers                | 7E FF 06 4D 00 00 00 FE AE EF | 7E FF 06 4D 00 00 00 EF |                                     |
| Currently online folder                           | 7E FF 06 60 00 00 00 FE 9B EF | 7E FF 06 60 00 00 00 EF | Query online folder                 |
| Pointer to current folder                         | 7E FF 06 61 00 00 00 FE 9A EF | 7E FF 06 61 00 00 00 EF | Queries the current playback folder |
| Check the specified total number of folder tracks | 7E FF 06 62 00 00 01 FE 98 EF | 7E FF 06 62 00 00 01 EF | QueryFOLDER1Total files             |
|   |                               |                         |                                     |
|   |                               |                         |                                     |



### Third, testing methods

#### 1、串口软件的操作



(1)、首先 Installation information"串口猎人"Software, 打开软件, 首先要搜索串口, 找 To refer to 定的端口之后, 指定“波特率”, 我们的模块默认的波特率为 9600, The most 后就是“启动串行端口”, 这样软件就配置好了。这里有两个 Reading required 要明确一下第一是“HEX 码”, 我们默认是这个, 这个是用来显 Readings 据的。所以必须设置这里第二是“字符串”, 这个是用来显示打 Printing 符的, 我们这里用不到。

(3), Soft 件配置 OK 之后 Will 需要的指令复制到发送区域, That is, 可。With 体的 Instructions 请参照模块的数据手册

(4), Such as 果模块的数据手册没有的测试指令的话, Please 自行计算, Especially 其需 To note 意的是校验和这两个字节如何计算不对的话, 模块是不接受指令

的。  
CheckCode is calculated by:

```
004501: /***** 功能描述: 求和和校验 *****/
004502: - 功能描述: 求和和校验
004503: - 参数说明:
004504: - 返回说明:
004505: - 注: 和校验的思路如下
004506: 发送的指令, 去掉起始和结束。将中间的6个字节进行累加, 最后取反码
004507: 接收端就将接收到的一帧数据, 去掉起始和结束。将中间的数据累加, 再加上接收到的校验
004508: 字节。刚好为0, 这样就代表接收到的数据完全正确。
004509: *****/
004510: void DoSum( INT8U *Str, INT8U len)
004511: {
004512:     INT8U xorsum = 0;
004513:     INT8U i;
004514:     for(i=0; i<len; i++)
004515:     {
004516:         xorsum = xorsum + Str[i];
004517:     }
004518:     xorsum = 0 - xorsum;
004519:     * (Str+i) = (INT8U) (xorsum >> 8);
004520:     * (Str+i+1) = (INT8U) (xorsum & 0x00FF);
004521: }
004522:
004523:
004524:
004525:
004526:
004527:
004528:
004529:
004530:
004531:
004532:
004533:
004534:
004535:
004536:
004537:
004538:
004539:
004540:
004541:
004542:
004543:
004544:
004545:
004546:
004547:
004548:
004549:
004550:
004551:
004552:
004553:
004554:
004555:
004556:
004557:
004558:
004559:
004560:
004561:
004562:
004563:
004564:
004565:
004566:
004567:
004568:
004569:
004570:
004571:
004572:
004573:
004574:
004575:
004576:
004577:
004578:
004579:
004580:
004581:
004582:
004583:
004584:
004585:
004586:
004587:
004588:
004589:
004590:
004591:
004592:
004593:
004594:
004595:
004596:
004597:
004598:
004599:
004600:
004601:
004602:
004603:
004604:
004605:
004606:
004607:
004608:
004609:
004610:
004611:
004612:
004613:
004614:
004615:
004616:
004617:
004618:
004619:
004620:
004621:
004622:
004623:
004624:
004625:
004626:
004627:
004628:
004629:
004630:
004631:
004632:
004633:
004634:
004635:
004636:
004637:
004638:
004639:
004640:
004641:
004642:
004643:
004644:
004645:
004646:
004647:
004648:
004649:
004650:
004651:
004652:
004653:
004654:
004655:
004656:
004657:
004658:
004659:
004660:
004661:
004662:
004663:
004664:
004665:
004666:
004667:
004668:
004669:
004670:
004671:
004672:
004673:
004674:
004675:
004676:
004677:
004678:
004679:
004680:
004681:
004682:
004683:
004684:
004685:
004686:
004687:
004688:
004689:
004690:
004691:
004692:
004693:
004694:
004695:
004696:
004697:
004698:
004699:
004700:
004701:
004702:
004703:
004704:
004705:
004706:
004707:
004708:
004709:
004710:
004711:
004712:
004713:
004714:
004715:
004716:
004717:
004718:
004719:
004720:
004721:
004722:
004723:
004724:
004725:
004726:
004727:
004728:
004729:
004730:
004731:
004732:
004733:
004734:
004735:
004736:
004737:
004738:
004739:
004740:
004741:
004742:
004743:
004744:
004745:
004746:
004747:
004748:
004749:
004750:
004751:
004752:
004753:
004754:
004755:
004756:
004757:
004758:
004759:
004760:
004761:
004762:
004763:
004764:
004765:
004766:
004767:
004768:
004769:
004770:
004771:
004772:
004773:
004774:
004775:
004776:
004777:
004778:
004779:
004780:
004781:
004782:
004783:
004784:
004785:
004786:
004787:
004788:
004789:
004790:
004791:
004792:
004793:
004794:
004795:
004796:
004797:
004798:
004799:
004800:
004801:
004802:
004803:
004804:
004805:
004806:
004807:
004808:
004809:
004810:
004811:
004812:
004813:
004814:
004815:
004816:
004817:
004818:
004819:
004820:
004821:
004822:
004823:
004824:
004825:
004826:
004827:
004828:
004829:
004830:
004831:
004832:
004833:
004834:
004835:
004836:
004837:
004838:
004839:
004840:
004841:
004842:
004843:
004844:
004845:
004846:
004847:
004848:
004849:
004850:
004851:
004852:
004853:
004854:
004855:
004856:
004857:
004858:
004859:
004860:
004861:
004862:
004863:
004864:
004865:
004866:
004867:
004868:
004869:
004870:
004871:
004872:
004873:
004874:
004875:
004876:
004877:
004878:
004879:
004880:
004881:
004882:
004883:
004884:
004885:
004886:
004887:
004888:
004889:
004890:
004891:
004892:
004893:
004894:
004895:
004896:
004897:
004898:
004899:
004900:
004901:
004902:
004903:
004904:
004905:
004906:
004907:
004908:
004909:
004910:
004911:
004912:
004913:
004914:
004915:
004916:
004917:
004918:
004919:
004920:
004921:
004922:
004923:
004924:
004925:
004926:
004927:
004928:
004929:
004930:
004931:
004932:
004933:
004934:
004935:
004936:
004937:
004938:
004939:
004940:
004941:
004942:
004943:
004944:
004945:
004946:
004947:
004948:
004949:
004950:
004951:
004952:
004953:
004954:
004955:
004956:
004957:
004958:
004959:
004960:
004961:
004962:
004963:
004964:
004965:
004966:
004967:
004968:
004969:
004970:
004971:
004972:
004973:
004974:
004975:
004976:
004977:
004978:
004979:
004980:
004981:
004982:
004983:
004984:
004985:
004986:
004987:
004988:
004989:
004990:
004991:
004992:
004993:
004994:
004995:
004996:
004997:
004998:
004999:
005000:
005001:
005002:
005003:
005004:
005005:
005006:
005007:
005008:
005009:
005010:
005011:
005012:
005013:
005014:
005015:
005016:
005017:
005018:
005019:
005020:
005021:
005022:
005023:
005024:
005025:
005026:
005027:
005028:
005029:
005030:
005031:
005032:
005033:
005034:
005035:
005036:
005037:
005038:
005039:
005040:
005041:
005042:
005043:
005044:
005045:
005046:
005047:
005048:
005049:
005050:
005051:
005052:
005053:
005054:
005055:
005056:
005057:
005058:
005059:
005060:
005061:
005062:
005063:
005064:
005065:
005066:
005067:
005068:
005069:
005070:
005071:
005072:
005073:
005074:
005075:
005076:
005077:
005078:
005079:
005080:
005081:
005082:
005083:
005084:
005085:
005086:
005087:
005088:
005089:
005090:
005091:
005092:
005093:
005094:
005095:
005096:
005097:
005098:
005099:
005100:
005101:
005102:
005103:
005104:
005105:
005106:
005107:
005108:
005109:
005110:
005111:
005112:
005113:
005114:
005115:
005116:
005117:
005118:
005119:
005120:
005121:
005122:
005123:
005124:
005125:
005126:
005127:
005128:
005129:
005130:
005131:
005132:
005133:
005134:
005135:
005136:
005137:
005138:
005139:
005140:
005141:
005142:
005143:
005144:
005145:
005146:
005147:
005148:
005149:
005150:
005151:
005152:
005153:
005154:
005155:
005156:
005157:
005158:
005159:
005160:
005161:
005162:
005163:
005164:
005165:
005166:
005167:
005168:
005169:
005170:
005171:
005172:
005173:
005174:
005175:
005176:
005177:
005178:
005179:
005180:
005181:
005182:
005183:
005184:
005185:
005186:
005187:
005188:
005189:
005190:
005191:
005192:
005193:
005194:
005195:
005196:
005197:
005198:
005199:
005200:
005201:
005202:
005203:
005204:
005205:
005206:
005207:
005208:
005209:
005210:
005211:
005212:
005213:
005214:
005215:
005216:
005217:
005218:
005219:
005220:
005221:
005222:
005223:
005224:
005225:
005226:
005227:
005228:
005229:
005230:
005231:
005232:
005233:
005234:
005235:
005236:
005237:
005238:
005239:
005240:
005241:
005242:
005243:
005244:
005245:
005246:
005247:
005248:
005249:
005250:
005251:
005252:
005253:
005254:
005255:
005256:
005257:
005258:
005259:
005260:
005261:
005262:
005263:
005264:
005265:
005266:
005267:
005268:
005269:
005270:
005271:
005272:
005273:
005274:
005275:
005276:
005277:
005278:
005279:
005280:
005281:
005282:
005283:
005284:
005285:
005286:
005287:
005288:
005289:
005290:
005291:
005292:
005293:
005294:
005295:
005296:
005297:
005298:
005299:
005300:
005301:
005302:
005303:
005304:
005305:
005306:
005307:
005308:
005309:
005310:
005311:
005312:
005313:
005314:
005315:
005316:
005317:
005318:
005319:
005320:
005321:
005322:
005323:
005324:
005325:
005326:
005327:
005328:
005329:
005330:
005331:
005332:
005333:
005334:
005335:
005336:
005337:
005338:
005339:
005340:
005341:
005342:
005343:
005344:
005345:
005346:
005347:
005348:
005349:
005350:
005351:
005352:
005353:
005354:
005355:
005356:
005357:
005358:
005359:
005360:
005361:
005362:
005363:
005364:
005365:
005366:
005367:
005368:
005369:
005370:
005371:
005372:
005373:
005374:
005375:
005376:
005377:
005378:
005379:
005380:
005381:
005382:
005383:
005384:
005385:
005386:
005387:
005388:
005389:
005390:
005391:
005392:
005393:
005394:
005395:
005396:
005397:
005398:
005399:
005400:
005401:
005402:
005403:
005404:
005405:
005406:
005407:
005408:
005409:
005410:
005411:
005412:
005413:
005414:
005415:
005416:
005417:
005418:
005419:
005420:
005421:
005422:
005423:
005424:
005425:
005426:
005427:
005428:
005429:
005430:
005431:
005432:
005433:
005434:
005435:
005436:
005437:
005438:
005439:
005440:
005441:
005442:
005443:
005444:
005445:
005446:
005447:
005448:
005449:
005450:
005451:
005452:
005453:
005454:
005455:
005456:
005457:
005458:
005459:
005460:
005461:
005462:
005463:
005464:
005465:
005466:
005467:
005468:
005469:
005470:
005471:
005472:
005473:
005474:
005475:
005476:
005477:
005478:
005479:
005480:
005481:
005482:
005483:
005484:
005485:
005486:
005487:
005488:
005489:
005490:
005491:
005492:
005493:
005494:
005495:
005496:
005497:
005498:
005499:
005500:
005501:
005502:
005503:
005504:
005505:
005506:
005507:
005508:
005509:
005510:
005511:
005512:
005513:
005514:
005515:
005516:
005517:
005518:
005519:
005520:
005521:
005522:
005523:
005524:
005525:
005526:
005527:
005528:
005529:
005530:
005531:
005532:
005533:
005534:
005535:
005536:
005537:
005538:
005539:
005540:
005541:
005542:
005543:
005544:
005545:
005546:
005547:
005548:
005549:
005550:
005551:
005552:
005553:
005554:
005555:
005556:
005557:
005558:
005559:
005560:
005561:
005562:
005563:
005564:
005565:
005566:
005567:
005568:
005569:
005570:
005571:
005572:
005573:
005574:
005575:
005576:
005577:
005578:
005579:
005580:
005581:
005582:
005583:
005584:
005585:
005586:
005587:
005588:
005589:
005590:
005591:
005592:
005593:
005594:
005595:
005596:
005597:
005598:
005599:
005600:
005601:
005602:
005603:
005604:
005605:
005606:
005607:
005608:
005609:
005610:
005611:
005612:
005613:
005614:
005615:
005616:
005617:
005618:
005619:
005620:
005621:
005622:
005623:
005624:
005625:
005626:
005627:
005628:
005629:
005630:
005631:
005632:
005633:
005634:
005635:
005636:
005637:
005638:
005639:
005640:
005641:
005642:
005643:
005644:
005645:
005646:
005647:
005648:
005649:
005650:
005651:
005652:
005653:
005654:
005655:
005656:
005657:
005658:
005659:
005660:
005661:
005662:
005663:
005664:
005665:
005666:
005667:
005668:
005669:
005670:
005671:
005672:
005673:
005674:
005675:
005676:
005677:
005678:
005679:
005680:
005681:
005682:
005683:
005684:
005685:
005686:
005687:
005688:
005689:
005690:
005691:
005692:
005693:
005694:
005695:
005696:
005697:
005698:
005699:
005700:
005701:
005702:
005703:
005704:
005705:
005706:
005707:
005708:
005709:
005710:
005711:
005712:
005713:
005714:
005715:
005716:
005717:
005718:
005719:
005720:
005721:
005722:
005723:
005724:
005725:
005726:
005727:
005728:
005729:
005730:
005731:
005732:
005733:
005734:
005735:
005736:
005737:
005738:
005739:
005740:
005741:
005742:
005743:
005744:
005745:
005746:
005747:
005748:
005749:
005750:
005751:
005752:
005753:
005754:
005755:
005756:
005757:
005758:
005759:
005760:
005761:
005762:
005763:
005764:
005765:
005766:
005767:
005768:
005769:
005770:
005771:
005772:
005773:
005774:
005775:
005776:
005777:
005778:
005779:
005780:
005781:
005782:
005783:
005784:
005785:
005786:
005787:
005788:
005789:
005790:
005791:
005792:
005793:
005794:
005795:
005796:
005797:
005798:
005799:
005800:
005801:
005802:
005803:
005804:
005805:
005806:
005807:
005808:
005809:
005810:
005811:
005812:
005813:
005814:
005815:
005816:
005817:
005818:
005819:
005820:
005821:
005822:
005823:
005824:
005825:
005826:
005827:
005828:
005829:
005830:
005831:
005832:
005833:
005834:
005835:
005836:
005837:
005838:
005839:
005840:
005841:
005842:
005843:
005844:
005845:
005846:
005847:
005848:
005849:
005850:
005851:
005852:
005853:
005854:
005855:
005856:
005857:
005858:
005859:
005860:
005861:
005862:
005863:
005864:
005865:
005866:
005867:
005868:
005869:
005870:
005871:
005872:
005873:
005874:
005875:
005876:
005877:
005878:
005879:
005880:
005881:
005882:
005883:
005884:
005885:
005886:
005887:
005888:
005889:
005890:
005891:
005892:
005893:
005894:
005895:
005896:
005897:
005898:
005899:
005900:
005901:
005902:
005903:
005904:
005905:
005906:
005907:
005908:
005909:
005910:
005911:
005912:
005913:
005914:
005915:
005916:
005917:
005918:
005919:
005920:
005921:
005922:
0059
```



Appendix I:SPI-FLASH Capacity and the length of the audio table

Schedule 1-1 MP3—FLASH-16p Module FLASH Capacity and length of audio swap tables:(Work unit:S)

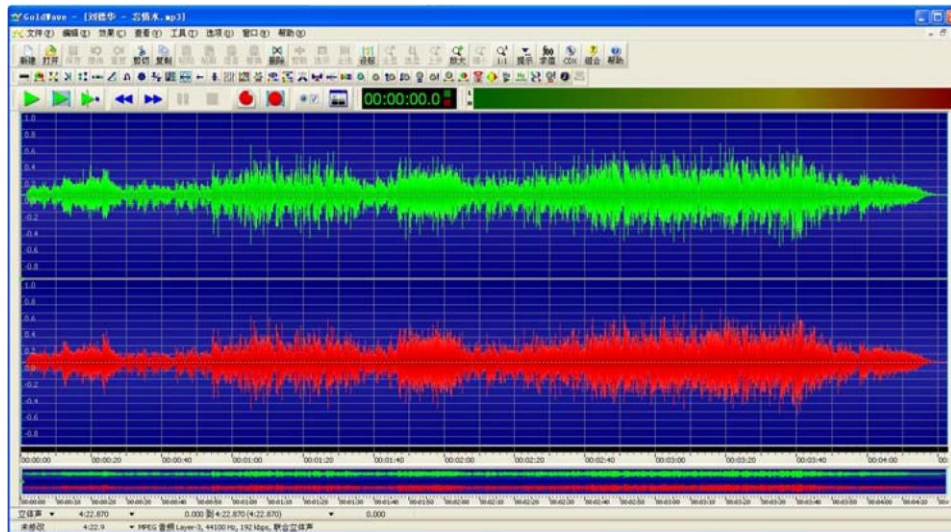
| 容量<br>码率 | 4Mbits | 8Mbits | 16MBit | 32MBit | 64MBit |  |
|----------|--------|--------|--------|--------|--------|--|
| 16Kbps   | 252    | 505    | 1011   | 2022   | 4045   |  |
| 24Kbps   | 163    | 327    | 654    | 1309   | 2618   |  |
| 32Kbps   | 113    | 226    | 453    | 906    | 1812   |  |
| 64Kbps   | 59     | 119    | 239    | 477    | 955    |  |
| 96Kbps   | 41     | 81     | 162    | 325    | 651    |  |
| 128Kbps  | 31     | 61     | 123    | 246    | 493    |  |
| 160Kbps  | 24     | 49     | 97     | 194    | 389    |  |
| 192Kbps  | 20     | 40     | 81     | 161    | 323    |  |
| 256Kbps  | 15     | 30     | 60     | 120    | 241    |  |
| 320Kbps  | 11     | 23     | 47     | 95     | 191    |  |

Note:MP3 File size depends on the bit rate, regardless of the sampling rate. Voice broadcast is recommended 16Kbps~64Kbps, Music recommended 32Kbps~96Kbps。

Third, the rate of conversion

To fight SPIFLASH The characteristics of small volume, stability, we have developed MP3—FLASH-16P Module. Directly through the phone's Microusb Line updates voice, but for the common MP3 Files, mostly 4M Bytes or so, use SPIFLASH,

Space is very difficult. But as voice broadcast and prompt, do not need high sampling rates



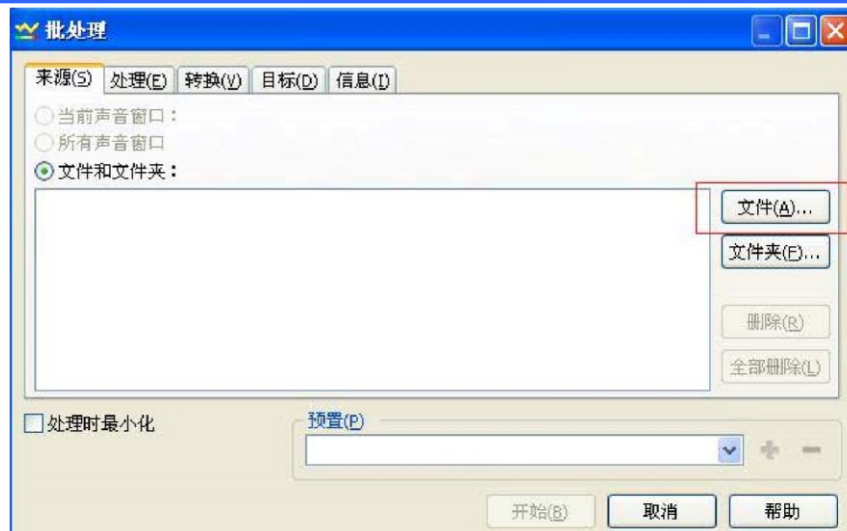
From the top left corner of the chart, we can see "The world's first. MP3" The sampling rate of up to 44100HZ. Bit rate 256KBS. This parameter describes the current song sound quality is quite good, so take a 4.5M Space.

But actually we do not need such a high sound quality, can be compressed. Are as follows:

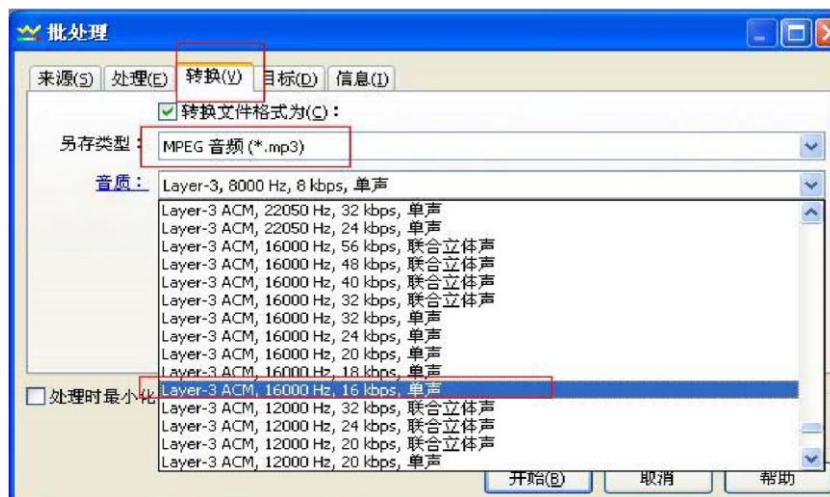
Use "GoldWave" This piece of software.



Click on batch processing,



Add a file



Choose "Conversion", Set the sampling rate for 16000KHZ and bit rate for 16KBS.



Then the path Specifies the converted files can be

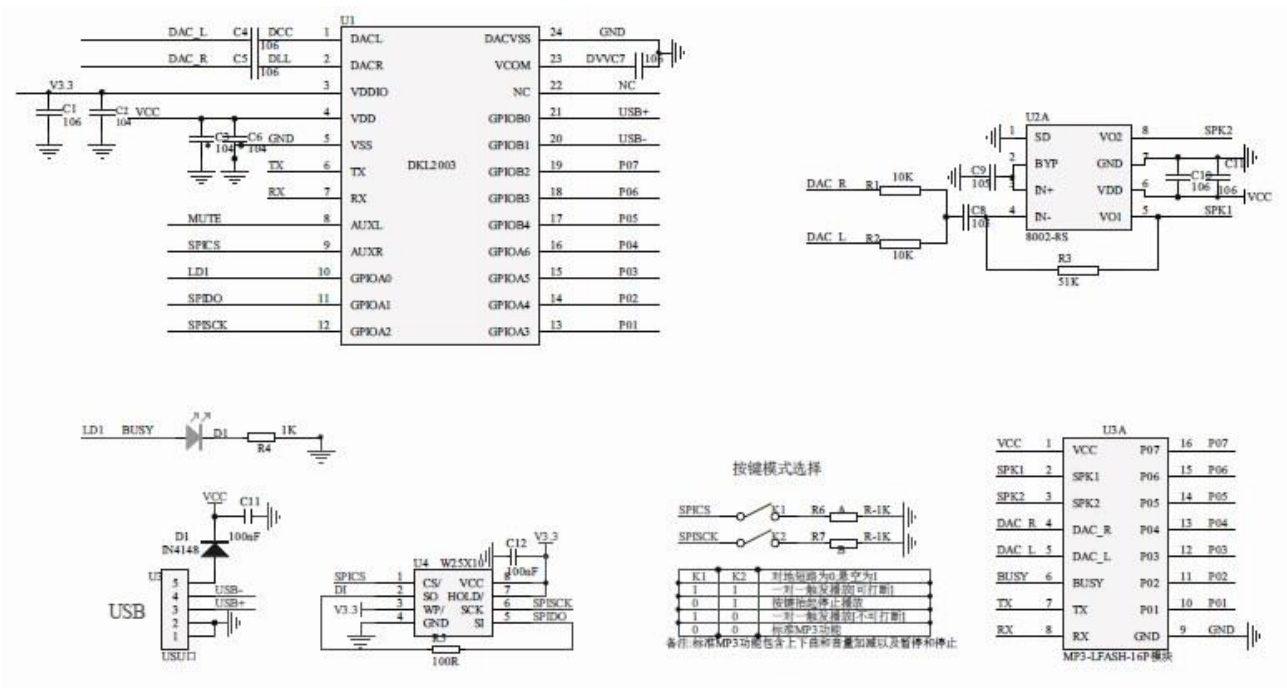


After compression, 4.5M Files into 507K . Step is

Notes:

1. If it is wav File, you can also use this software convert MP3
2. After the transition effects, the user can directly on the computer listen for a minute effect, play computer effects, and We play is consistent with the results of the module
3. If sound quality is bad, you can increase the sample rate and bit rate of these two parameters. You can try it for yourself
4. If you need to modify the volume of the audio source, and trim an audio source, you can use this software

Four, module schematics



**FourProgramming examples****Program example: serial port named**

/\*\*\*\*\*\*

Function: towards a chip named the first song and the second song, the basic procedures for user testing

Date : The year 2013-05-06

Operating environment:STC Crystal:11.0592M Baud rate: 9600

Notes : In General, science and technology 51 Development Board debug OK --- STC89C516RD+

1, The testing program must be a module or chip equipment online in the programme, such as U Plates,TF Card,FLASH

\*\*\*\*\*/ #include "REG52.h"

#define COMM\_BAUD\_RATE 9600 //Serial baud rate

#define OSC\_FREQ 11059200 //Run Crystal:11.05926MHZ static INT8U Send\_buf[10] = {0} ;

void Delay\_Ms(INT32U z)

```
{  
    INT32U x=0 , y=0; for(x=110 ; x>0 ;x--)  
        for(y=z; y>0;y-- );  
}
```

/\*\*\*\*\*\*

Function description: Serial port 1 Initialize

Note: Set to 9600 Baud rate

\*\*\*\*\*/ void Serial\_init(void)

```
{  
    TMOD = 0x20; //Set T1 Baud rate generator  
    SCON = 0x50; 0101,0000 8Data bits, No parity  
    PCON = 0x00; PCON=0;  
    TH1=256-(OSC_FREQ/COMM_BAUD_RATE/32/12);//Set to 9600 Baud rate  
    TL1=256-(OSC_FREQ/COMM_BAUD_RATE/32/12);  
    TR1 = 1; //Timer 1 Open it  
    REN = 1; //Serial port 1 Receive enabled  
    ES = 1; //Serial port 1 Interrupt enable  
}  
void Uart_PutByte(INT8U ch)  
{  
    SBUF = ch; while(! TI){;}  
    TI = 0;  
}
```

/\*\*\*\*\*

Function description: Serial port sends out commands[Including control and query]

Parameter description: CMD:Control instructions, consult the instruction list also includes queries related instruction  
you need an answer[0:Without answering,1:Need an answer]

feedback:Do

data:Transmission parameters

\*\*\*\*\*/ void SendCmd(INT8U len)

```
{
    INT8U i = 0 ;
    Uart_PutByte(0x7E); //Start   for(i=0; i<len; i++)//Data
    {
        Uart_PutByte(Send_buf[i]) ;
    }
    Uart_PutByte(0xEF) ;//End
}
```

/\*\*\*\*\*

Function: checksum

And calibration of thought are as follows:

Send instructions, remove the start and end. The intermediate 6 Bytes added to finally take back yards. Receiver will receive the frame, remove the start and end. Data accumulated in the Middle, plus receive parity bytes. Just as 0.On behalf of received data is completely correct.

/ void DoSum( INT8U \*Str, INT8U len)

```
{
    INT16U xorsum = 0; INT8U i;
    for(i=0; i<len; i++)
    {
        xorsum = xorsum + Str[i];
    }
    xorsum = 0 -xorsum;          *(Str+i) = (INT8U)(xorsum >>8);
    *(Str+i+1) = (INT8U)(xorsum & 0x00ff);
}
```

void Uart\_SendCMD(INT8U CMD ,INT8U feedback , INT16U dat)

```
{
    Send_buf[0] = 0xff; //Reserved bytes
    Send_buf[1] = 0x06; //Length
    Send_buf[2] = CMD; //Control commands   Send_buf[3] = feedback; //Whether you need feedback   Send_buf[4] = (INT8U)(dat >> 8); //datah
    Send_buf[5] = (INT8U)(dat); dataal
    DoSum(&Send_buf[0],6); //Check
```



```
SendCmd(8); //Send this frame data
}

void main()
{
    Serial_init() ;//Initialization of the serial register setting
    Uart_SendCMD(0x03 , 0 , 0x01) ;//Play the first song
    Delay_Ms(1000) ;//Delay probably 6S
    Uart_SendCMD(0x03 , 0 , 0x02) ;//Play next song
    Delay_Ms(1000) ;//Delay probably 6S
    Uart_SendCMD(0x03 , 0 , 0x04) ;//Play the fourth song    while(1) ;
}
```