# Simple Audio Module

## 1 Introduction

User can easily control the audio module by uart, which includes FAT filesystem, SD driver, audio decode and so on. It can play the music in the microSD card, receiving commands from users. Thus, it provides a true 'ALL-IN-ONE' module that is ideally suited for digital audio product.

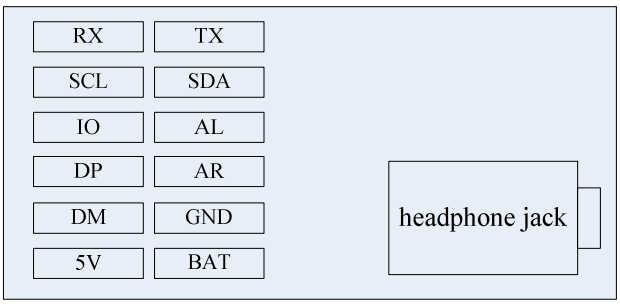
## 2 Features

**1)** Support MP3,WMA and WAV fortmat music.
**2)** Compatibility with FAT16 and FAT32 filesystem.
**3)** Support microSD card and USB disk.
**4)** 16 levels volume control**.**
**5)** 8 kinds of EQ.
**6)** Support selecting music by filename or file number.
**7)** Support expanding PA.
**8)** Support Li-ion battery Power supply.
**9)** Support calendar function.

## 3 Technical parameters

| Name | **Description** |
|---|---|
| Music format | MP3：MPEG1/2/2.5 Audio Layer 1,2,3 decoder，bit rate 8-448Kbps, sampling rate 8-48KHz, CBR/VBR<br>WMA： WMA Decoder, bit rate 32-384Kbps, sampling rate 8-48KHz<br>WAV：bit rate 32-384Kbps, sampling rate 8-48KHz |
| UART | 3.3V,TTL |
| Nominal voltage | 5V |
| Rated current | 40mA |
| Headphone driver output | 2x20mW @16ohm |
| Audio output | Frequency response range: 20Hz ~ 20KHz<br>Stereo 16-bit Sigma-Delta D/A<br>SNR：>85db |
| Size | 40.1mm*17.6mm |
| operating temperature | 0-70℃ |

# 4 Pin description




| Pin Number | Pin Name | Functions |
|---|---|---|
| 1 | RX | Uart receive IO |
| 2 | TX | Uart transmit IO |
| 3 | SCL | Status indication IO for playing. It will output 1Hz PWM when playing, and set to high when stopping, and set to low when being low power state. |
| 4 | SDA | Status indication IO of battery charging, high for charging and low for not. |
| 5 | IO | expanding PA chip select IO |
| 6 | AL | Audio left channel |
| 7 | DP | USB DP |
| 8 | AR | Audio right channel |
| 9 | DM | USB DM |
| 10 | GND | Supply voltage ground |
| 11 | 5V | Supply voltage |
| 12 | BAT | Li-ion battery supply voltage |

# 5 UART guide

## 5.1 UART Settings

Module receives commands from control system through UART. Parameters settings as below.
Communication standard: serial port, 3.3V, TTL
Baud rate: 9600bps
Data bits: 8
Stop bits: 1
Parity bits: none
Stream control: none

## 5.2 Power on

The storage medium, as microSD or USB disk, will be scanned as soon as power on. Only when module sends feedback string “24 01 E0 E1”, indicating the scaning success that users can send control command. If not success, please recheck the connection of hardware, the filesystem of storages, and the format of music.

## 5.3 Command format

General format：$ Len Instr Feedback Para CheckSum

| Format field | Description | Comment |
|---|---|---|
| $ | Start flag | Every CMD should begin with $, that is 0x24 |
| LEN | the byte number after LEN | Not including checksum |
| Instr | Instructions | Specific the operations,such as play and stop. |
| Feedback | Feedback flag | Indicate whether feedback or not, 1 for feedback , otherwise 0 for. |
| Para | Parameter | Instruction content(such as music number ) |
| CheckSum | Check sum | Chechsum = Len ^ CMD ^ Feedback ^ Para |

## 5.4 Command

| Command | Format | Indications | Example |
|---|---|---|---|
| Sigle playing in line with music number | 24 Len 01 00/01 NH NL XX | NH,NL meaning order high byte and order low byte. This’s in line with order of copy file | 24 04 01 00 00 01 04<br>Sigle playing the first music |
| Sequence playing in line with music number | 24 Len 02 00/01 NH NL XX | Total loop playing from music, which order is N | 24 04 02 00 00 02 04<br>Sequence playing from second music |
| Sigle loop playing in line with music number | 24 Len 03 00/01 NH NL XX | Sigle loop playing music, which order is N | 24 04 03 00 00 02 05<br>Loop second music |
| Sigle playing in line with music name | 24 Len 04 00/01 Name XX | The total number of filename can’t more than 12, include the suffix, as MP3 | 24 07 04 00 31 2E 4D 50 33 32<br>Sigle playing the music named “1.mp3” |
| Sequence playing in line with music name | 24 Len 05 00/01 Name XX | Total loop playing. The total number of filename can’t more than 12, include the suffix, as MP3 | 24 07 05 00 31 2E 4D 50 33 33<br>Sequence playing from “1.mp3” |
| Sigle loop playing in | 24 Len 06 00/01 Name | The total number of | 24 07 06 00 31 2E 4D |

| | | | |
|---|---|---|---|
| line with music name | XX | filename can't more than 12, include the suffix, as MP3 | 50 33 30<br>Loop the "1.mp3" |
| Stop | 24 02 07 00/01 XX | stop | 24 02 07 00 05 |
| Pause/Play | 24 02 08 00/01 XX | Pause in playing, and play in pausing | 24 02 08 00 0A |
| Next | 24 02 09 00/01 XX | Next music | 24 02 09 00 0B |
| Last | 24 02 0A 00/01 XX | Last music | 24 02 0A 00 08 |
| Volume add | 24 02 0B 00/01 XX | Volume add a level | 24 02 0B 00 09 |
| Volume subtract | 24 02 0C 00/01 XX | Volume subtract a level | 24 02 0C 00 0E |
| Set volume | 24 03 0D 00/01 Value XX | 16 level volumes in total, value range is 0-16 | 24 03 0D 00 05 0B<br>Set volume as 5th level |
| EQ change | 24 02 0E 00/01 XX | EQ is standard, jazz, popular, bass, rock, classical, soft and SRS in turn. | 24 02 0E 00 0C<br>Change next |
| Set EQ | 24 03 0F 00/01 Value XX | The EQ setting range is 0-7 | 24 03 0F 00 01 0D<br>Set EQ as jazz |
| Standby mode | 24 02 10 00/01 XX | Only battery power supply, this function can work. DC5V from "5V"pin lasting 0.5s can wake up module from standby. | Send command as<br>24 02 10 00 12 |
| Set time | 24 09 11 00/01 DATE TIME XX | DATE consisted of 2bytes of year, 1byte of month and 1byte of day.<br>TIME consisted of 1byte of hour, 1byte of minute and 1byte of seconds. | Send command as<br>24 09 11 00 07 DD 06 0D 0B 11 01 d2<br>Set time as 11h 17m 01s（0B 11 01） on June 13,2013（07 DD 06 0D）. |
| Get playing music name | 24 02 21 01 XX | Feedback flag is no effect in this command | Send command as<br>24 02 21 01 22<br>Get feedback as<br>24 06 00 31 2E 4D 50 33 37<br>Meaning music totality is "1.MP3" |
| Get music totality | 24 02 22 01 XX | Feedback flag is no effect in this command | Send command as<br>24 02 22 01 21<br>Get feedback as<br>24 03 00 00 08 0B<br>Meaning music totality is 8 |

| | | | |
|---|---|---|---|
| Get playing music order | 24 02 23 01 XX | Feedback flag is no effect in this command | Send command as<br>24 02 23 01 20<br>Get feedback as<br>24 03 00 00 01 02<br>Meaning first music is playing |
| Get time | 24 02 25 01 XX | Get date and time, and format as same as command of set time. | Send command as<br>24 02 25 01 26<br>Get feedback as<br>24 08 00 07 DD 06 0D 0B 11 07 C4<br>Meaning 11h 17m 7s on June 13,2013. |
| Set feedback at music end | 24 03 A0 01 **00/01** XX | 5th byte being 1, control system should get feedback as "24 01 F0 F1" when music over. Otherwise, 5th byte being 0, feedback been not exist. Default not feedback. | Set music-end feedback as<br>24 03 A0 01 01 A3<br>Unset music-end feedback as<br>24 03 A0 01 00 A2 |
| Get software version | 24 02 80 01 XX | Feedback flag is no effect in this command | Send command as<br>24 02 80 01 83<br>Get feedback as<br>56 31 2E 30 00<br>Meaning version is "V1.0" |

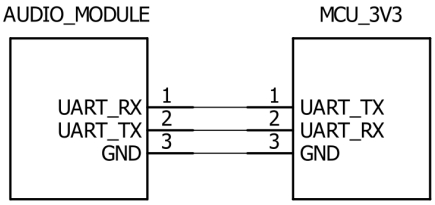
## 5.5 Feedback format

$ Len Result Para CheckSum

| Format field | Description | Comment |
|---|---|---|
| $ | Start flag | Everyone CMD should begin with $, that is 0x24 |
| Len | the byte number after *Len* | Not include checksum |
| Result | Indicate the result of command execution | 0x00：execution success<br>0x10：error start flag<br>0x11：error checksum<br>0x12：parameter out of range<br>0x13：error Instruction<br>0x14：can't find file<br>0xE0：scan storage successfully |
| Para | Parameter | The value of parameter |
| CheckSum | Check sum | Chechsum = Len ^ Result ^ Para |

# 6 User guide

1) Insert microSD into slot, which has music file.
2) Connect module uart TX to control system uart RX, and module uart RX to control system uart TX, module GND to control system GND. If control system output 5TTL, please refer to attentions.



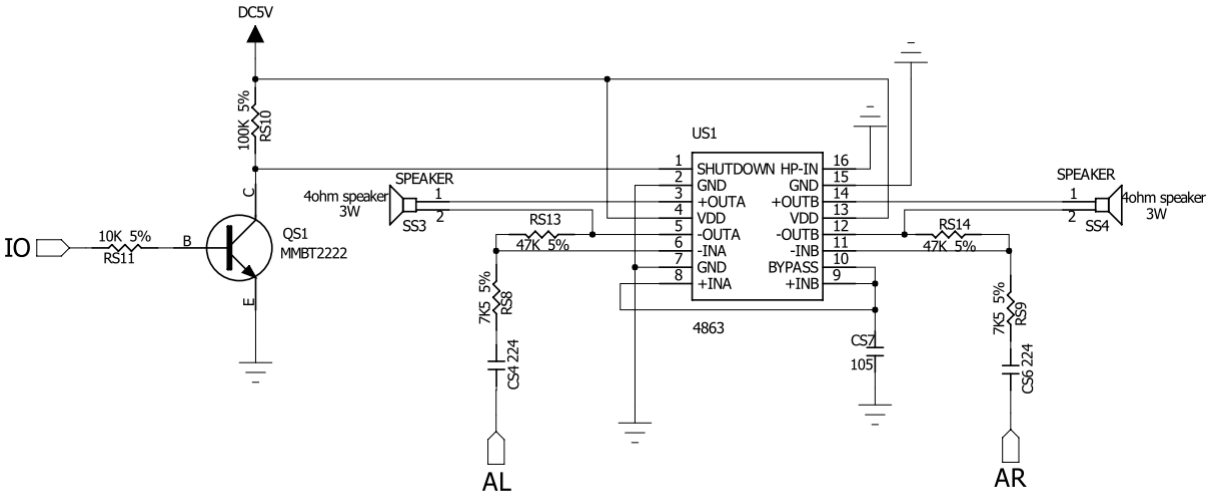

3) Connect DC5V power supply to “5V” pin.
4) Control system send command to module

# 7 Power supply mode

Module power supply can choose DC5V input or Li-ion battery. If choose DC5V, power positive pole link to “5V”pin while negative pole link to “GND”pin. If choose Li_ion battery, power positive pole link to “BAT”pin while negative pole link to “GND”pin, and “5V”pin to be a pin for battery charging.

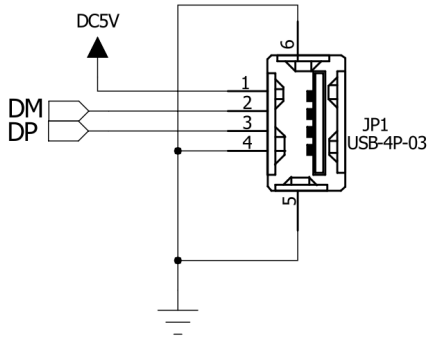
# 8 External hardware

## 8.1 External power amplifier



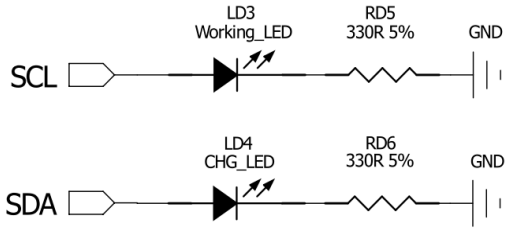

“IO”pin, “AL”pin and “AR”pin will be used in this case. “IO” pin is chip select for power amplifier IC.

## 8.2 External USB



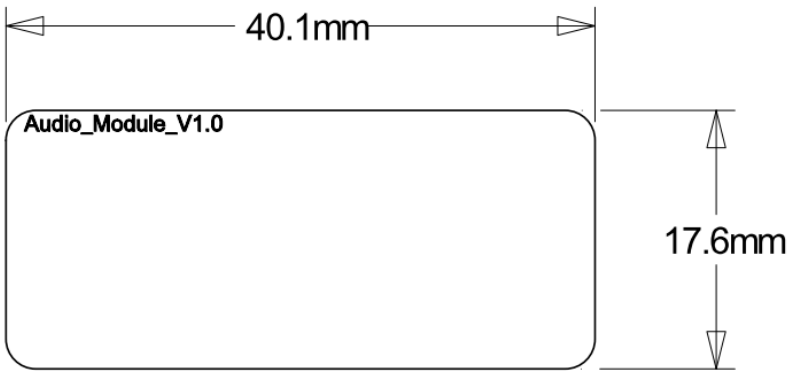

"DM"pin and "DP"pin will be used in this case.Only this module be the host, not device.

## 8.3 External LED light



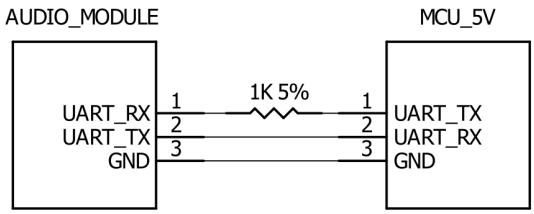

"SCL"pin and "SDA"pin will be used in this case. SCL indicate the state of play while SDA indicate the state of charging. They can output max-current 12mA. Resistance changes follow LED light.

# 9 Module size




# 10 Attentions

1) Module uart used 3.3V TTL. It's necessary to series connect 1k resistor when control system is 5V TTL.




2) It may take a long time for scanning if too many file in microSD. Please delete the Redundant file.
3) Input current will be more than 400mA from “5V”pin, when Li-ion charging.