Unit-AudioPlayer

N9301 Control Protocol

Table of Contents

1、Communication Protocol Structure

1.1 Communication Protocol Parameters

1.1.1 Communication Interface Parameters

The N930X series is equipped with a standard UART asynchronous serial interface. It is a TTL level interface and can be converted to RS232 level through the MAX232 chip. The module communication adopts full-duplex serial port communication, with a baud rate of 9600, 8 data bits, 1 stop bit, and no parity bit. It is recommended to wait for 5 seconds after powering on before sending commands, with each 2 commands spaced more than 500ms apart.

1.2 Data Packet Format

1.2.1 Data Packet Format

Format: [CMD] [CheckCMD] [Len] [Data1] [Data2] …… [Datan] [Checksum]		
CMD	Command	Command, type of received data
CheckCMD	Check Command	Check Command, the inverted code of the command code, is used to verify the command code
Len	Data Length	Total length of all data in the entire segment, that is, total number of bytes of the data
Data1	Data1	The first byte of data contained in this command
Data2	Data2	The second byte of data contained in this command
……	……	……
Datan	Datan	The nrd byte of data contained in this command
Checksum	Checksum	Checksum of the data, taking the lower 8 bits of the 16-bit checksum, is referred to as SM below

Note: One piece of data can only receive a maximum of 32 bytes (including checksum and other items).

For example, to play the 8th song on the current disk, the message to send would be:

Command	Command Complement	Len	Data1-3	Checksum
04	FB	3	06 00 08	10

The data can be one piece or multiple pieces, mainly depending on the rules specified in the CMD command, corresponding to what kind of data.

1.2.2 Fault tolerance requirements

The recipient must verify the instruction code and its complement code in real time, as well as the correctness of the verification. The number of data bytes for each instruction may vary, so after receiving the number of bytes, the subsequent number of data bytes to be received must be determined. If there is an error, the received instruction should be discarded. **If the receiving time of an instruction exceeds 500ms, the received instruction bytes that are too many should be forcibly discarded, and the instruction receiving synchronization should be restarted.**

1.2.3 Protocol

- When a new track is played, its track number will be sent. The track number here is a unified system number for the entire system. If you need to distinguish between parent directories and child directories, you can query information such as path and path depth.
- When the disk drive is unplugged, a disk drive removal command will be sent and disk drive replacement processing will be carried out, while waiting for the operation command.
- All the data in the agreement are expressed in HEX format.
- The SM checksum represents the sum of all bytes (excluding the checksum itself), and is the low 8 bits of the 16-bit checksum calculated.
- The files referred to in the protocol are those that can be played. They do not include non-playable files.
- After switching the mode, all devices are in the stopped playback state. A playback command must be sent for them to play. When powered on, they default to a single song being stopped.
- The volume is set to the maximum by default. You can query related command or set the volume level.

2、Configuration Command Set (HEX)

2.1 Play Control

- Command: 04 ("SM" represents the checksum below)
- Meaning of Return code：

  Playing State：00--Stopped　01--Playing　02--Paused

  Device (drive letter) : 00--USB FLASH DISK　01--SD　02--SPI FLASH

Note: The command will only be responded to after the SD card is inserted.

- Query play status (00) :

Command	Command Complement	Len	Data1	Checksum
04	FB	01	00	00

Return: **04 FB 02 00 [Play State Code] SM**

- Playing (01) :

Command	Command Complement	Len	Data1	Checksum
04	FB	01	01	01

Return: **04 FB 02 00 [Play State Code] SM**

- Paused (02) :

Command	Command Complement	Len	Data1	Checksum
04	FB	01	02	02

Return: **04 FB 02 00 [Play State Code] SM**

- Stopped (03) :

Command	Command Complement	Len	Data1	Checksum
04	FB	01	03	03

Return: **04 FB 02 00 [Play State Code] SM**

- Previous track (04) :

Command	Command Complement	Len	Data1	Checksum
04	FB	01	04	04

Return: **(Current track) 04 FB 03 0E [Play State Code] SM**

- Next track (05) :

Command	Command Complement	Len	Data1	Checksum
04	FB	01	05	05

Return: **(Current track) 04 FB 03 0E [Track H] [Track L] SM**

- Specified track (06) :

Command	Command Complement	Len	Data1	Data2	Data3	Checksum
04	FB	03	06	Track H	Track L	SM

Return: **(Current track) 04 FB 03 0E [Track H] [Track L] SM**

Note: The track number is represented by two bytes, high and low. If it's the 6th track, it would be 0006.

00 06=0*16^3+0*16^2+0*16^1+6*16^0=6

For example:

04 FB 03 06 00 08 10 specifies to play the 8th track on the current disk. The number of tracks tracks from 1 to 65535 (the track number cannot be 0), and is represented in hexadecimal as 0000H to FFFFH.

00 08=0*16^3+0*16^2+0*16^1+8*16^0=8

- Play from the specified path on the current drive letter (07) :

Command	Command Complement	Len	Data	path	Checksum
04	FB	**17**	07	The data for modulus operation (less than 27 bytes)	SM

Return: **(Current track) 04 FB 03 0E [Track H] [Track L] SM**

For example: /DIR1/track3.mp3

04 FB 17 07 2F 44 49 52 31 2F 74 72 61 63 6B 33 2E 6D 70 33 0D

Note: There are one small space between each two lines of code.

How did **17** come about?

17 = path + 1-- **2F/44/49/52/31/2F/74/72/61/63/6B/33/2E/6D/70/33**

Note: The +1 in the equation represents 07, which means one data length.The path in the equation is **2F 44 49 52 31 2F 74 72 61 63 6B 33 2E 6D 70 33** (these are the paths) **(For details on the path format, please refer to the path format description)**

- Query the current number of online devices (08) :

Command	Command Complement	Len	Data1	Checksum
04	FB	01	08	08

Return: **04 FB 02 08 [Number of Online Devices] SM**

Note: USB FLASH DISK--01 SD--02 FLASH--04 FLASH/USB FLASH DISK--05

FLASH/SD--06 USB FLASH DISK/SD--03

- Query the current playing device (09) :

Command	Command Complement	Len	Data1	Checksum
04	FB	01	09	09

Return: **04 FB 02 09 [Drive Letter] SM**

- Query total number of tracks (0D) :

Command	Command Complement	Len	Data1	Checksum
04	FB	01	0D	0D

Return: **04 FB 02 0D [Total tracks H] [Total tracks L] SM**

- Query current track (0E) :

Command	Command Complement	Len	Data1	Checksum
04	FB	01	0E	0E

Return: **04 FB 03 0E [Track H] [Track L] SM**

- Current driver letter, current track, specific playing time (0F) :

Command	Command Complement	Len	Data1	Data2	Data3	Checksum
04	FB	03	0F	min	sec	SM

Return: null

- Current driver letter, specific track, specific playing time (10) :

Command	Command Complement	Len	Data1	Data2	Data3	Data4	Data5	Checksum
04	FB	05	10	Track H	Track L	min	sec	SM

Return: null

- Previous directory (12) :

Command	Command Complement	Len	Data1	Checksum
04	FB	01	12	12

Return: **Return the track number when playing.**

- Next directory (13) :

Command	Command Complement	Len	Data1	Checksum
04	FB	01	13	13

Return: **Return the track number when playing.**

- Stop playing (14) :

Command	Command Complement	Len	Data1	Checksum
04	FB	01	14	14

Return: null

- Obtain short file names (15) :

Command	Command Complement	Len	Data1	Checksum
04	FB	01	15	15

Return: **04 FB 0C 15 [Short File Name(11 Bytes)] SM**

- Select the track but do not play it (16) :

Command	Command Complement	Len	Data1	Data2	Data3	Checksum
04	FB	03	16	Track H	Track L	SM

Return: **(Current track) 04 FB 03 0E [Track H] [Track L] SM**

- Query the total number of files in the current directory (18) :

Command	Command Complement	Len	Data1	Checksum
04	FB	01	18	18

Return: **04 FB 03 18 [Track H] [Track L] SM**

2.2 Playing Time Control

- Command: 05

- Total play time query (00) :

Command	Command Complement	Len	Data1	Checksum
05	FA	01	00	00

Return: **0A F5 04 00 [Hour] [Min] [Sec] SM**

- Send current playback time (01) :

Command	Command Complement	Len	Data1	Data2	Data3	Data4	Checksum
05	FA	04	01	hour	min	sec	SM

Return: null

- Set playback time to on (02) :

Command	Command Complement	Len	Data1	Checksum
05	FA	01	02	02

Return: **0A F5 04 01 [Hour] [Min] [Sec] SM**

- Set playback time to off (03) :

Command	Command Complement	Len	Data1	Checksum
05	FA	01	03	03

Return: null

2.3 Volume Control

- Command: 06

- Volume range is adjustable from 0 to 30 levels. The maximum volume is 30. The default maximum volume is set when the device is turned on.

- Query volume (00) :

Command	Command Complement	Len	Data1	Checksum
06	F9	01	00	00

Return: **06 F9 02 00 VOL SM**

- Volume setting (01) :

Command	Command Complement	Len	Data1	Data2	Checksum
06	F9	02	01	VOL	SM

Return: null

For example:

06 F9 02 01 14 16 set the volume to level 20, 14 is in hexadecimal and represents level 20.

1*16^1+4*16^0=20

- Volume plus (02) :

Command	Command Complement	Len	Data1	Checksum
06	F9	01	02	02

Return: null

- Volume down (03) :

Command	Command Complement	Len	Data1	Checksum
06	F9	01	03	03

Return: null

2.4 Repetition Control

- Command: 08

- Repeat in a specific time (00) :

Command	Command Complement	Len	Data1	Data2	Data3	Data4	Data5	Checksum
08	F7	05	00	Start min	Start sec	End min	End sec	SM

Return: null

For example:

08 F7 05 00 02 06 02 20 2E represents that it will repeat from 2 minutes 06 seconds to 2 minutes 32 seconds.

- End repeating (01) :

Command	Command Complement	Len	Data1	Checksum
08	F7	01	01	01

Return: null

Note : After executing the termination command, this repetition is invalid, or other operations can also terminate the repetition.

2.5 Loop Mode Control

- Command: 0B

- Query loop mode (00) :

Command	Command Complement	Len	Data1	Checksum
0B	F4	01	00	00

Return: **0B F4 02 00 [Loop Mode] SM**

- Set loop mode (01) :

Command	Command Complement	Len	Data1	Data2	Checksum
0B	F4	02	01	Loop Mode	SM

Return: null

Set single-Track Loop mode :

Command	Command Complement	Len	Data1	Data2	Checksum
0B	F4	02	01	01	03

Return: null

The function corresponding to the loop playback mode code :

- All loop : 00

  Play all the tracks in sequence. After the playback is complete, it will loop.

- Single loop : 01

  Play the current track in a continuous loop.

- Loop within the folder : 02

  Play in sequence loop + tracks in the current folder.

- Shuffle play : 03

  Randomly play the tracks within the disc tray.

- Single song stopped : 04

  Stop playing the current track once it finishes.

- Sequential play : 05

  Play all the tracks in sequence and stop when finished.

2.6 Composite Playback Command

- Command: 0C

- Composite playback (0C) :

Command	Command Complement	Len	Mode	Data1	Data2	Datan	Checksum
0C	F3	LEN	Mode	Track1	Track2	Trackn (Max 20)	SM

Return: null

Mode: 01 -- Document Number , 02 -- Document Name

For example (**Document Number**) :

0C F3 05 01 04 06 07 0A 20 indicates that the playback sequence is: the 4th, 6th, 7th and 10th tracks are played together.

For example (**Document Name**) :

0C F3 0A 02 2E 77 61 76 33 23 31 30 23 61 represents the playback sequence: the audio combination named 3 and 10 in the root directory (not under the file folder) is played (2E is " . " 77 is w, 61 is a, 76 is v, so the format of the combined audio file is WAV format. 33 is character 3 in ASCII standard, 23 is character "#" in ASCII standard. In this program, the semicolon is used to distinguish the combined path, so the above code is "character: 3#10#;").

End playlist playback :

Command	Command Complement	Len	Data1	Checksum
0C	F3	01	02	02

Return: **04 FB 02 00 00 01**

Note: The combination playback can be terminated in advance during the process.

2.7 Sleep Command

- Command: 0D

Command	Command Complement	Len	Data1	Checksum
0D	F3	01	01	02

Return: **0D F3 01 01 02**

Note : Executing this command will cause the chip to enter a low-power mode; after entering sleep mode, the first transmitted code is invalid and serves as a wake-up signal. The state after waking up is equivalent to the reset state.

2.8 Return the Error Message

- Command: AA
  - AA 55 02 FF 01 01 -- Serial port received data error
  - AA 55 02 FF 02 02 -- Specified drive letter could not be found

- AA 55 02 FF 03 03 -- No available drive letter for playback
- AA 55 02 FF 04 04 -- File playback error, such as the file cannot be found, etc

2.9 Return Device Insertion and Removal Status Information

- Command: BB
- BB 44 01 02 02 SD -- Insert SD card
- BB 44 01 02 02 SD -- Remove SD card

3、File Path Format Description and Verification Code Algorithm

3.1 Folder Path Requirements

- The name of a folder can be no longer than 8 bytes. 8 bytes equal 4 Chinese characters or 8 letters.
- The file name can be no more than 8 bytes. You can write the first few bytes of the file name and then use an asterisk (*) to indicate the remaining part of the name (for example, "/Zhuang Xinyan - Goodbye, Just a Stranger.mp3", it can be represented as "Zhuang Xinyan*.mp3")
- One Chinese character occupies two bytes, while a space or a character occupies one byte.

  For example: Audio in the specified folder

  /DIR1/track3.mp3

  2F 44 49 52 31 2F 74 72 61 63 6B 33 2E 6D 70 33

3.2 Checksum Calculation Method

In this UART data protocol, the parity bit samples the checksum, which is the 16-bit checksum obtained by adding up the data bytes excluding the checksum itself. Then only the lower 8 bits are taken. For example, if the data sent is 04 FB 03 06 00 08 10, which specifies to play the 8th song on the current drive, the first 6 bytes of this data will be used to calculate the checksum, and then the lower 8 bits are taken, resulting in a checksum of 10.