Intelligent Control Network Design Based on CAN-bus for Audio Equipment

XIULI ZHANG[1] , ZHONG WAN [2], DEYANG YANG [3]; WEIDONG CHEN[1]
[1] Electronic and Information Engineering Department
Zhejiang Wanli University
Ningbo, Zhejiang, 315100
[2] Computer Science and Information Technology Dpartment
Zhejiang Wanli University
Ningbo, Zhejiang, 315100
[3] Soundking Group Co. Ltd.
818# Chenxin Road Yinzhou Investment Industry Park,Ningbo, Zhejiang, 315104
P.R. CHINA
gaoyibo@gmail.com http://www.zwu.edu.cn

Abstract: - In order to solve the problem of inconvenience in network construction, short transmission distance of RS232 and non-intelligence audio device with different interface for network, this paper presented a design of the audio intelligent control network based on CAN(Controller Area Network)-Bus. The topology of audio control network, structured models of ICANNet and hardware are firstly discussed. Then the protocol of transport layer and application layer and software of intelligent control network is studied. Finally the test results were given to demonstrate the feasibility and reliability of the design.

Key-Words: - intelligent Control Network; CAN-bus; Audio Equipment;

1 Introduction

Presently, the background music systems are widely used in all kinds of large gymnasiums, residential leisure places, and most professional audio applying occasions, where a variety of audio equipment are used. Due to the location limitation，it is difficult to arrange these equipments in one concentrated location, so that they are generally distributed to different places. This will bring many problems. As in large gymnasiums, when the distances between audio equipment are more than 100 meters the sounds from different equipments can not reach audience ears simultaneously, so that sound synchronization is needed. Similarly, even though different equipments send out the same volume of sounds, the effect that the audience received is differed due to different distance and different attenuation, so that volume control is needed. How to combine the large number of audio equipment, which are usually non-intelligent and with incompatible interface, into one network so that to achieve acoustics control such as on-site audio volume control and sound synchronization adjustment, is one of the major problems that modern digital audio technology faces.

In order to achieve the control of non-intelligent audio equipment, many companies in the world, such as Hyfax, Accuphase companies use RS232 serial interface technology to achieve audio equipment field volume adjustment, sound synchronization and parameter balance. Because RS232 communication distance is so short (according to EAT/TAI-232 criteria, only 15 meters), and can only achieve the point to point communication, it has not been widely applied in control network of audio equipment. The CAN bus is a multi-master mode serial data communication protocol [1][2][3], it can achieve higher communication rates and longer transmission distances with high resistance to electromagnetic interference, and can detect any errors produced so that ensure the reliability of real-time communication. From the late-1980s until now, CAN bus has been widely used in process automation, manufacturing automation, building automation and other areas, to construct on-site intelligent equipments communication networks [4][5][6], and it became the first choice for audio control network. This article chose TMS320VC5402 DSP, AT82C250 as the core chip, designed a distributed audio equipment control network based on CAN-bus, which results well in the test.

2 System Structure and Hardware Design

The system structure of audio intelligent control network based on CAN-bus for distributed audio equipment is shown in Fig. 1. The server (Server in

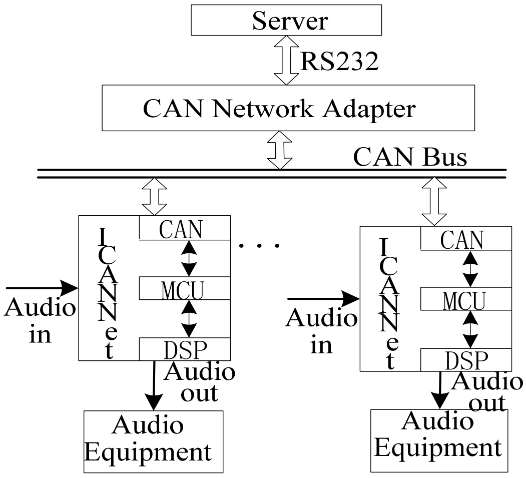

Fig.1 Structure of audio control network

Fig.1) is usually composed of PC machine, mainly realizes the audio equipment to monitor active status and to issue each kind of control command. The CAN-bus network adapter realizes the protocols conversion between the RS232 interface of Server and the CAN-bus interface of the intelligence CAN-bus network node (ICANNet in Fig.1). The distributed non-intelligence audio equipment connects to the CAN-bus audio control network through the CAN-bus interface of ICANNet. The ICANNet analysis and realizes each layer protocol of CAN-bus audio control network. The ICANNet node can divide into 3 parts according to its function: The CAN module, which composed by the CAN-bus controller and the CAN-bus driver, the MCU module and the DSP module. The entire CAN-bus audio control network may connect mostly 127 ICANNet nodes (or audio site). The ICANNet node hardware architecture is shown in Fig. 2.

The DSP processor in ICANNet is used to realize the audio effect processing such as parameter equalization, sound synchronization, pressure limitation and so on. Because the general audio equipment like power amplifier, sound console, and speaker, is non-intelligent equipment, it cannot receive directly the control command transmitted from Server. For this reason we designed an intelligence audio effect processing module using audio CODEC and the DSP processor. The DSP audio effect processing module receives the control command from the server through the CAN-bus network, changes acoustics of the audio equipment. TMS320VC5402 DSP processor is chosen for it's low cost and high performance, which has also big

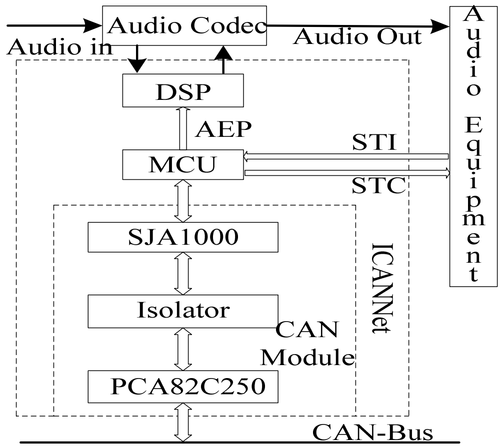

Fig.2 Hardware architecture of ICANNet node

internal storage capacity, quick operating speed. TMS320VC5402 chip has two high-speed, full-duplex multi-channel buffer serial ports [7][8][9][10]. These ports allow direct interface to the serial input output audio CODEC in a system, also support master slave working mode. These ports have double-buffered data registers which allow continuous data stream transmission and full-duplex communication. Audio CODEC uses the competitive 24-bit encoder-decoder TLC320AD77. It is a full-duplex stereo (A/D) and (D/A) converter using delta-sigma technique, with 100dB dynamic range, 100dB harmonic distortion and the noise, and 16 kHz to 96 kHz adjustable sampling rates. The chip uses the differential input and output. On the chip an anti-aliasing filter, an output smoothing filter and a digit De-emphasis filter are also offered. It supports slave working mode.

Because the operating voltage and temperature of power amplifier and other audio equipment affect the acoustics, the working conditions monitoring and control of these equipment is needed, which is done by MCU module. TMS320VC5402DSP mainly realizes the audio effect processing function and there are no extra resources remained for it to realize the data receive and the transmission on the CAN bus. Thus the MCU needs to realize the data receive and the transmission on the CAN bus, and transforms the data packet which needs by the DSP audio control. The MCU should have big memory and high operating frequency. For this reason STC12C5416AD is chosen, which has the operating frequency upper to 33 MHz and works with single instruction cycle.

Both DSP chip that the system used and MCU STC12C5416AD do not contain the CAN controller. So that a CAN module is required which is composed of CAN drive and CAN controller. In order to save the cost and reduces the size, SMD Package independent CAN controller SJA1000 and CAN driver PCA82C250 are selected. To strengthen the node anti-interference ability, the pins TX0 and RX0 of SJA1000 do not directly connect to the pins TXD and the RXD of PCA82C250, but connect through the optical coupler ADUM1201 to PCA82C250 [11][12]. This also realized the bus electrical isolation between each node.

3 Communication Protocol

Audio control network communication protocol includes three layers, which are physical layer, transmission layer, and application layer. Fig. 3 shows its hierarchical structure. The CAN controller

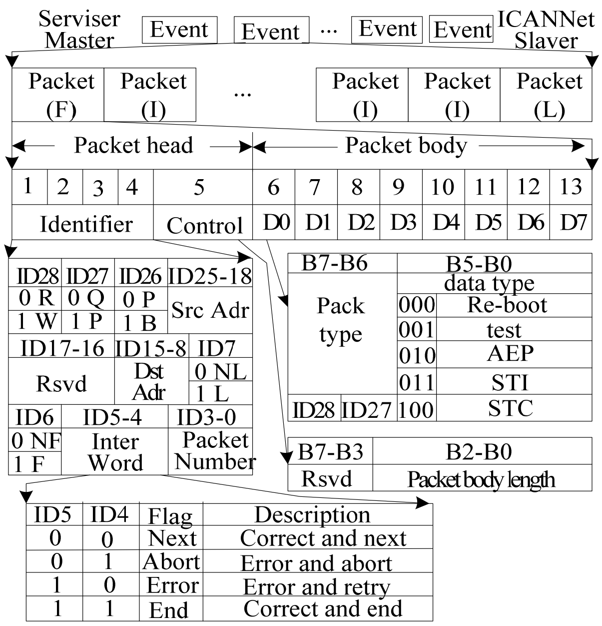

Fig.3 Layers of CAN audio control network communication protocol

SJA1000 realized the physical layer protocol. CAN-bus network is a multi-host network. Communication events can be initiated by different nodes that connected to the network. But in the audio control network design, it is master/slave mode. The server is the master node and each ICANNet is the slave node. The node address is the identification code to decide whether to receive the data packet that other node transmit. The message packet CAN-bus network communication used is 101bit, as is the information carrier in the audio control network. A 101bit message packet is composed of one 37bit packet head and 64bit (8byte) application data (Packet body). Packet head contains two parts, one is expands CAN ID, which length is 29bit, the other is control word, which length is 8bit. Expands CAN ID is detail introduced in section3.1. Application data in packet body is detail introduced in section3.2. Because of the packet body only 8 byte, so we need to three bits in control word to represent its length. We use Low three bits in control word; upper five bits are reserved as shown in Fig.3. A message packet includes the transmission layer and application layer protocol.

3.1 Transmission Layer Protocol

The CAN audio control network transmission layer follows the master/slave model. This transport protocols mainly contains 2 kinds of the transmission mode: write mode and read mode. The transmission mode definite Sever (Master) as the starting point. When Sever (Master) needs to transmit the sound effect control parameters to each ICANNet (audio node), Server transmits the data packet to each ICANNet in write mode. This write type application data packet is 102 bytes in length as shown in Table 1. When Server needs to learn the working condition of any audio equipment that connected to a CAN node, it sends out an inquiry packet in read mode. The ICANNet that Server inquired of sends back the 7 bytes read type application data packet (equipment working condition).

Each communication process between Master node and Slave node includes one or more events. Each event includes a pair of information, one is from Master to Slave, and the other is from Slave to Master. Each communications process begins with a write/read request packet sent out by the Master. After the reply packet that sent from slave packet is received, follow-up information exchange will be done. The mutual information mainly are write packet and read packet, and also includes: Next packet expresses it may normal transmit next packet in the multi-packet exchange process; Error packet indicates that has the mistake to occur in network, and the wrong can be corrected through the reproduction; Abort packet indicates that has had the mistake which cannot be corrected; End packet expressed the process of communications conclusion. Fig. 4 shows several kinds of typical transmission layer packet flows in the CAN-bus audio control network. In the case of the single Packet exchange, the 1st data packet also means the last data Packet. "F&L" expresses "first and last"; "F", "I" and "L" means respectively "first", "intermediate" and "last". These are used in multi-packet exchanges.

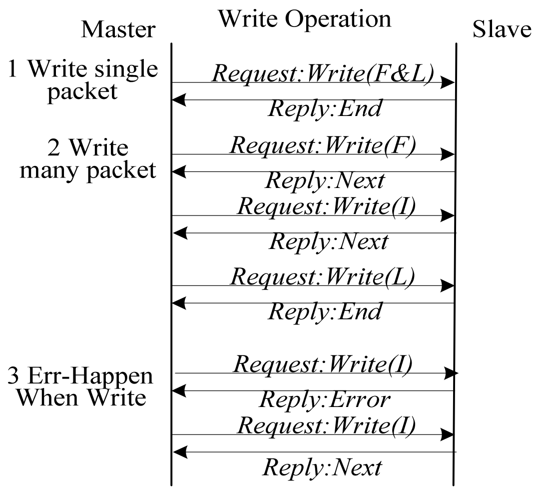

Fig.4 Packet flows in transmission layer

As shown in Fig.3, the following bits in 29bit expands CAN ID are used to support the transmission level protocol for CAN audio control network: ID28 indicates whether current communication event that Master initiated is read (R) Packet or write (W) Packet; ID27 indicates the current data Packet is a Master Request (Q) Packet or a Slave (P); ID26 indicates whether communication mode is broadcast (B) or point-to-point (P); ID25-ID18 expresses the packet source address; ID15-ID8 expresses the packet destination address; The ID7-ID0 support one or more message packets reading/writing.

3.2 Application Layer Protocol

The application layer protocol for CAN-bus audio control network defines by the data type, data type locates at the first byte of the first message packet application data (Packet body) of a communication event. Fig.3 shows the D0 byte format of the CAN-bus audio control network application protocol. The bit7 and the bit6 of D0 byte copy from ID28 and ID27, and bit5 to bit0 define the data type. The data type definition command include: test (test), ICANNet reboot (Re-boot), the equipment active status information collection, the equipment active status control command and the acoustics control parameters and so on. The application protocol packet is composed by one or more message packet. The length of each packet body depends on the specific data type. It is indicated in Control byte B2-0. The number of packets for one communication event is indicated in ID7, ID6 and ID3-0. Because the biggest length of a CAN-bus packet body is 8byte, therefore one application protocol may consist several CAN message packets.

Application Layer defines five types of data: Re-boot (remote reboot), test (Test), AEP (audio effect control parameters), STI (working state information) and STC (working state control). Different types of data are also different in length. The length of Re-boot type is one byte. Test type is one byte too. STI type is 3 bytes, which are used to test the voltages,

Table 1 Definition of AEP application data

Fun	Parameter					
Frame Flag	01111110					
A/B	000000（channel A）					
Flag	111111（channel B）					
delay	delay				pass	
	H/L				H/L	
Level	level				pass	
	H/L				H/L	
Phase	inverse				pass	
	H/L				H/L	
EQ	1 Band PEQ*4					Pass
	b0h	b1h	b2h	-a1h	-a2h	H/L
	H/L	H/L	H/L	H/L	H/L	
	b0l	b1l	b2l	-a1l	-a2l	
	H/L	H/L	H/L	H/L	H/L	
Limiter	threshold		Ratio		pass	
	H/L		H/L		H/L	

currents and work status information of devices that connected to the network; STC type is 7 bytes, which are used to control the working status of equipments in the network. AEP type is 102 bytes, which are used to control audio devices to produce different audio effects.

AEP application data is defined in Table 1, They are audio effect control parameters which contains five kinds of audio processing effects. These audio effect control parameters including: delay, digital volume control, phase control, 4 bands parametric equalizer with centre frequency, gain and quality factor adjustable, and digital audio limiter with threshold and proportion adjustable. The most complicated one is the effect of parametric equalizer. We have done a lot of research work about parametric equalizer. Specific design methods are published in reference [13].

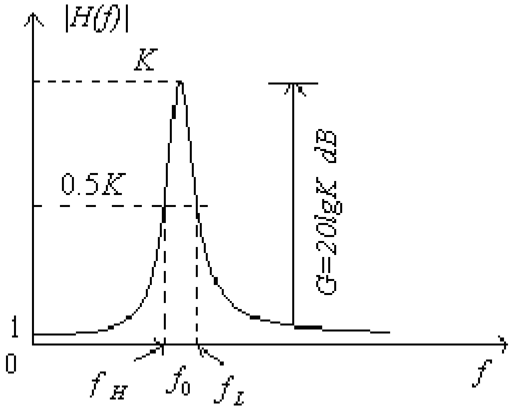

Fig.5-a Amplitude-frequency characteristic curve of the peaking filter

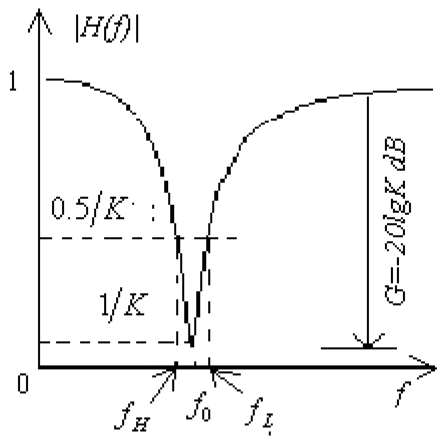

Fig.5-b Amplitude-frequency characteristic curve of the cutting filter

Fig.5 Amplitude-frequency characteristic curve of the peaking and cutting analog filter

The ideal parametric equalizer is only to peaking or cutting the desired frequency components, made no effect to the other audio components. So the parametric equalizer is composed of a group of peaking and cutting filters as shown in Fig.5. The performance of the peaking and cutting filter is mainly determined by three parameters: the centre frequency f_0, gain G and quality factor Q. As long as you change one, two or three parameter, performance of the peaking and cutting will change as your need .Filter quality factor Q value is defined as follows:

$$Q = \frac{f_0}{f_H - f_L} \qquad (1)$$

$$0 < f_H < f_0 < f_L$$

Where f_H is the upper cut off frequency of filter when the transmission level is 3dB lower than centre frequency and f_L is the lower cut off frequency of filter when the transmission level is 3dB lower than centre frequency.

So each band of parametric equalizer has four technical specifications. Three of them are the parameters of peaking and cutting filter：centre frequency f_0, gain G, quality factor Q .The another parameter is the filter type parameter T (T = 1 means peaking filter, T = 0 means cutting filter). In DSP module a second-order digital IIR filter is used to achieve one band parametric equalization effect, the transfer function of the second-order IIR filter is

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} \qquad (2)$$

Suppose the sample period of the input audio signal is T_s , we get the relation of filter coefficients and equalizer parameters as

$$b_0 = \frac{1 + G^T \frac{\pi f_0 T_s}{Q} + (\pi f_0 T_s)^{-2}}{1 + \frac{\pi f_0 T}{G^{T-1} Q} + (\pi f_0 T_s)^{-2}} \qquad (3)$$

$$a_1 = b_1 = \frac{2(\pi f_0 T_s)^{-2} - 2}{1 + \frac{\pi f_0 T}{G^{T-1} Q} + (\pi f_0 T_s)^{-2}} \qquad (4)$$

$$b_2 = \frac{1 - G^T \frac{\pi f_0 T_s}{Q} + (\pi f_0 T_s)^{-2}}{1 + \frac{\pi f_0 T}{G^{T-1} Q} + (\pi f_0 T_s)^{-2}} \qquad (5)$$

$$a_2 = \frac{1 - \frac{\pi f_0 T_s}{G^{T-1} Q} + (\pi f_0 T_s)^{-2}}{1 + \frac{\pi f_0 T}{G^{T-1} Q} + (\pi f_0 T_s)^{-2}} \qquad (6)$$

As TMS320VC5402DSP is a 16-bit fixed-point DSP, in order to prevent data overflow during audio effects processing, double-precision data format Q24 is used. Each audio data is 32-bit consisting of two 16-bit bytes, where the highest bit is a sign bit; 7bits for integer, low 24 bits for decimal. In the table1, channel A is the left channel of stereo signal, while channel B is the right channel of stereo signals. After the server receives the equalizer parameter information, it calculates those 5 filter coefficients according to equation (3) ~ (6), convert them into double-precision data format Q24 which can be identified by TMS320VC5402DSP, and package into one application data packet, transmit it to ICANNet, and then generate parametric equalization and other audio effects.

4 Software Design

In this article software design mainly includes three parts: Server software design, CAN network adapter software design and software design for intelligent CAN-bus network interface (ICANNet).

4.1 Server Software Design

Server is composed of PC. It is the control center of the distributed audio equipment control network based on the CAN bus. Server software achieves these functions: user login (UL), audio equipment address management (AM), network testing (NT), on-line equipment display, complete machine information (CMI) and audio effect information (AEI). Server software flow shows in Fig.6. Before accessing to network, any equipment should be allocated it's own address by AM module. The address can be updated when needed by AM module. AM is an independent module, other modules can be manually switched with each other. When the server program begins to run, UL interface is firstly entered. After finishing user login in UL interface, enter NT interface, here you can select testing, equipment arrangement, background decorate, and so on functions. If select testing function, the sever send an inquiry packet to every network node to confirm a network node is on power or not. If a node returns a reply packet to Server, there is an active icon in the on-line equipment interface standing for the equipment connected the node. If Server receives no reply in stipulated time, there is an inactive icon in the on-line equipment interface. An inactive icon shows that the equipment connected the node is power off and you can't control or monitor its working state. If you select background decorate function, you can use location map of audio equipment placed as background, then drag all icons into the corresponding location to easy to operate. You also can select one of a background from drop-down menu. Double click any active icon, select CMI or AEI in the pop-up dialog box. Then enter CMI interface or AEI interface. Operating of CMI and AEI is detail introduced in section5. Fig.9 shows the operation interface of CMI.

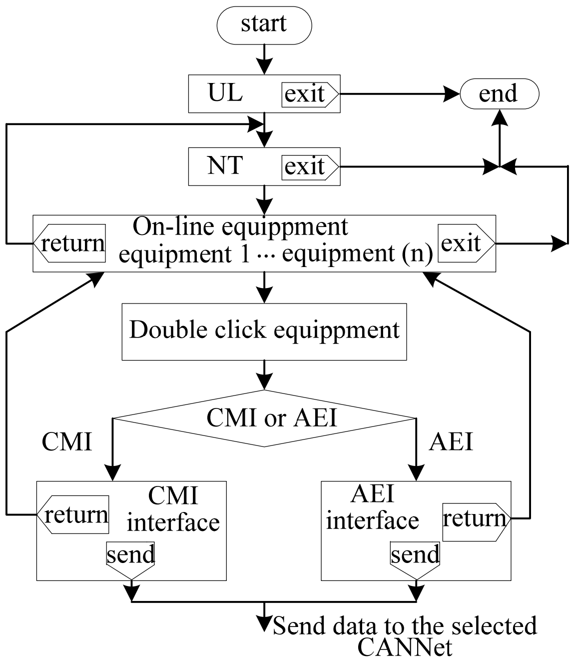

Fig.6 Server software flow

4.2 CAN Network Adapter Software Design

The network adapter is bridge between Server and ICANNet node. The main functions of Serve are send Write packets from Server to each ICANNet nodes and forwarded Read packets from ICANNet nodes to Server. Its major operating objects are CAN controller, FIFO and RS232 controller. Fig. 7 has given the network adapter simplified state transition diagram. In the diagram there are 3 kinds of lines, solid lines indicate state transition, hollow double thread arrows indicate memory read-write, and dashed lines means that the signal is interactive. The main transition states of CAN network adapter are the initial state of CAN controller and RS232 controller(C-Idle, R-Idle), reading and writing FIFO(RR-men, WR-men, WC-men, RC-men), data receiving and sending in CAN bus(R-ph, R-pb, W-ph, W-pb), data receiving and sending in RS232 bus(RRS232, WRS232), transferring protocol from CAN to RS232 (C-R) and from RS232 to CAN (R-C).

To Read transmission mode, after ICANNet node receive a request from Serve, the node send an inquiry packet with response information to Server through CAN bus. When the inquiry packet arrives at network adapter, the interrupt signal CR-init of CAN controller in network adapter will trigger its C-Idle module, R-ph module then read the data from packet head and R-pb module read the data from packet body. C-R module will be analyzed inquiry packet in CAN protocol , re-packed according to the RS232 protocol and WC-men module preserved its in the FIFO. Finally, FR-init signal will trigger its R-Idle module, RR-men module then read the data from FIFO, invoke WRS232 modules to forward inquiry packet in RS232 protocol to Server through RS232 bus.

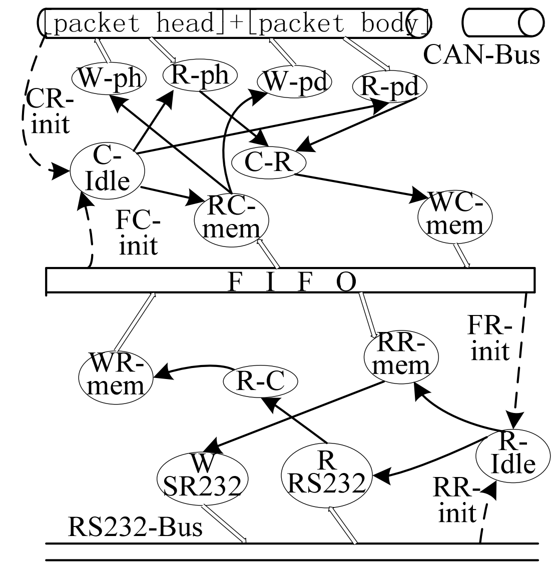

Fig.7 Diagram of network adapter state transition

To Write transmission mode, after Serve issue a request, the interrupt signal RR-init of network adapter RS232 will trigger its R-Idle module, RRS232 module then read the data on the RS232 bus, R-C module will be analyzed data packet in RS232 protocol , re-packed according to CAN protocol and WR-men module preserved its in the FIFO. Finally, FC-init signal will trigger its C-Idle module, RC-men module then read the data from FIFO, invoke W-ph and W-pd modules to forward request packet in CAN protocol to each ICANNe nodes through CAN bus.

4.3 ICANNet Software Design

ICANNet has two main functions: Firstly, it receives data packets from CAN bus, resolves protocol, produces specific sound effects through the DSP, achieves audio effect control or change device working conditions through microcontroller, realizes remote device control; secondly, it collects device work status through MCU and packages the data into packets according to the protocol in Fig. 2, sends to the CAN bus, forward to the server, realizes monitoring of the equipment work state. Therefore, ICANNet software includes MCU software and DSP software.

MCU is the control core of ICANNet module. It realizes protocol resolution, device working status information collecting and control. DSP produces specific sound effects according to the audio parameters that microcontroller receives. The communication between MCU and DSP uses SPI protocol. Therefore the ICANNet software mainly includes the following parts: System initialization module, receive data module, transmit data module and protocol analysis module. ICANNet software flow is shown in Fig. 8.

ICANNet software implementation process includes three steps: 1) After node power on, ICANNet automatically calls the initialization module to realize the system initialization. 2) After completed the initialization, ICANNet enters the pre-operating status. When there isn't a data receive

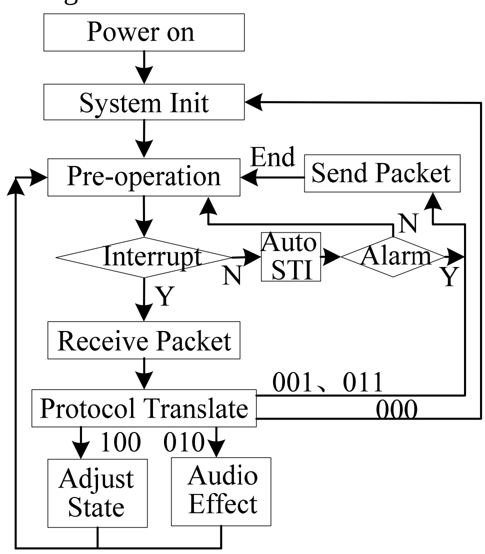

Fig.8 ICANNet software flow

interrupt, the ICANNet call gathering equipment active status information module program to get the equipments active status, packages the data into packets according to the protocol shown in Fig. 3, inspects whether the warning mark which defines in table 2 is fined. If no warning mark occurs, the packet will not be sent. Otherwise ICANNet calls the data transmission module to send the equipment active status information and then gets back to pre-operating status. 3) In pre-operating status, when there is a data receive interrupt, ICANNet receives packets from CAN bus, resolves protocol, and make corresponding movements with respect to those 5 kinds of application data defined in Fig. 3.

5 Experiment and Test

According to the above protocol, we have independently designed the intelligence network interface card ICANNet and the network adapter that the server access to network. We have completed the software design for Server operation interface, software for network adapter, the DSP module software and the MCU module software compilation in ICANNet. After the system software and hardware design completed, we have made the multiple joint network test to the system.

Testing environment design is given according to the CAN audio network topology shown in Fig. 1, including a Server (PC machine with Windows XP operating system) and 6 ICANNet nodes. The server connects to the audio control network through the serial port and the network adapter. Communication tests are done to those 5 kinds of application data in Fig.3. Addresses are assigned to different ICANNet through address burning program，that is AM module of Server software. The test command in the NT interface clicked, you can know all equipments connected the network are power on or power off. Double click the icon standing for power on equipment, select CMI in the pop-up dialog box, then enter CMI interface, the ICANNet which you have selected will return its work status information by STI data, as shown in fig.9. Work status information of selected equipment is updated every two seconds. In CMI interface, you can set the equipment active status control functions such as “Low-cut, mute”. Then click the “send” button to realize STC. In corresponds address on the ICANNet module indicator “Low-cut, mute” light on, showing the equipment active status. Double click the icon standing for power on equipment, select AEI in the pop-up dialog box, then enter AEI interface, Click the Re-boot command on the interface in AEI, Re-boot indicator lights on the working state indicating module in the corresponding ICANNet devices turn on, indicating a successful ICANNnet reboot.

Fig.9 The interface of CMI

Set audio effect control parameters in the AEI interface to realize AEP. The input signal is a sine wave from DF1641D signal generator. TDS5054 oscilloscope is used to measure the input signal and output signal of DSP audio effect module in ICANNet. Firstly, we select the node which address is 0x4, and select bypass function for all five audio effects. Adjust the signal generator output signal’s frequency to 1250Hz, and it's amplitude to 0.7v. When added the same signal to the left (A) channel and the right (B) channel of the stereo input of the DSP audio effect module, the output signal of A and B channel is shown in Fig.10-a. They are almost the same except a little delay due to analog signal processing circuit can't be completely identical.

Select delay function for channel B (others are bypass), the delay time is set to 4ms. The input signal of channel A and B is still same. The output signal of channel A and B is shown in Fig.10-b. There are nearly 4ms delay between channel A and channel B. This is because the channel A signal is set to bypass and the channel B signal is set to 4 ms delay.

Select inverse function for channel B (others are bypass). The input signal of channel A and B is still same. The output signal of channel A and B is shown in Fig.10-c. Channel A signal phase is almost opposite channel B signal phase.

Select volume function for channel A (others are bypass). The volume function makes the input signal amplitude multiply the volume parameter. In other words, the output signal amplitude is equal to the input signal amplitude multiplied by the volume parameter. The volume parameter is set to 05. That is, the attitude of channel A output signal is only half of channel B output signal attitude when they are the same input. The output signal of channel A and B is shown in Fig.10-d. The channel A output signal amplitude is indeed half of the channel B output signal amplitude.

Select parametric equalize function for signal in channel A (others are bypass) through the server interface. The parameters are as below: Centre frequency 1383Hz, Gain 3dB, quality factor 50. Input signal of channel A / B is the same, and the frequency is 1383Hz. ICANNet resolved the protocol and then forward the parameters to DSP module to produce corresponding sound effects; Measured with an oscilloscope the waveforms of channel A / B output signal of the ICANNet DSP module with the address 0x4 is shown in Fig. 10-e. The signal frequency of channel A is 1383Hz, peak-peak value VA is 358mv, the signal frequency of channel B also remains unchanged, peak-peak value VB is 253mv, the ratio of them (G) dB = 20log10 (VA / VB) = 3.02. All these results shows the DSP module correctly received control commands sent from the server and produced a corresponding parametric equalizing movement; Increase the gain parameter to 16dB for parametric equalizer function, the output signal of channel A and B is shown in Fig.10-f. The channel A output signal is distortion because of DSP data overflow. Change channel A parameters in centralized control software interface, select pressure limit function, choose pressure parameter threshold limit equal to 0.5 volts, compression ratio equal to 1:8 (choose the other parameters as bypass), re-send the sound control parameters. The measured channel A / B signal waveform is shown in Fig. 10 -g. As can be seen from the figure when the B-channel signal surpassed the threshold, it was compressed, and so that the signal distortion was eliminated. Multi-node network experimental results showed that all five types of data can be normally transmission, which demonstrates the design of CAN-based Distributed audio Control Network Protocol is credible and each module operates reliably.

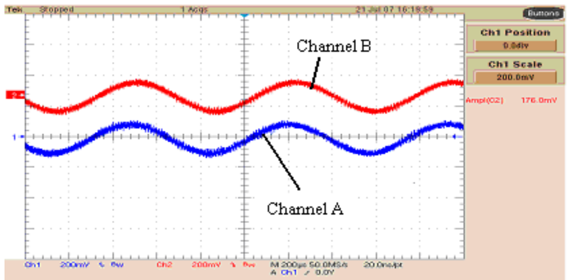

Fig.10-a Output signal of channel A and B when select bypass function for all five function

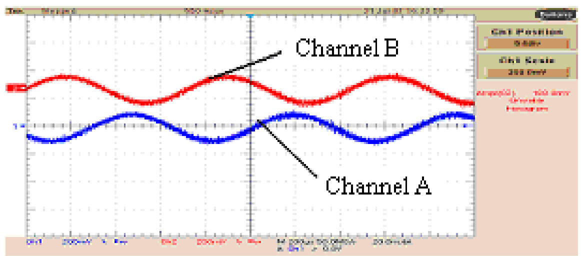

Fig.10-b Output signal of channel A and B when select delay 0.4ms for channel B (others all bypass)

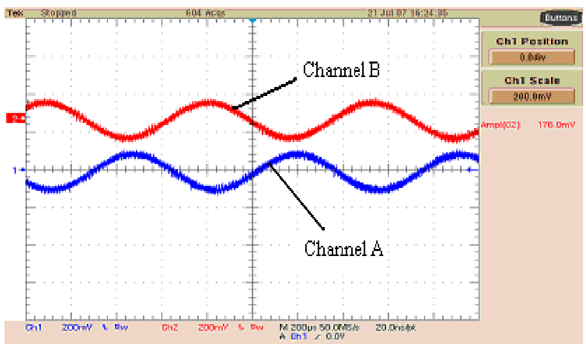

Fig.10-c Output signal of channel A and B when select inverse function for channel B (others all bypass)

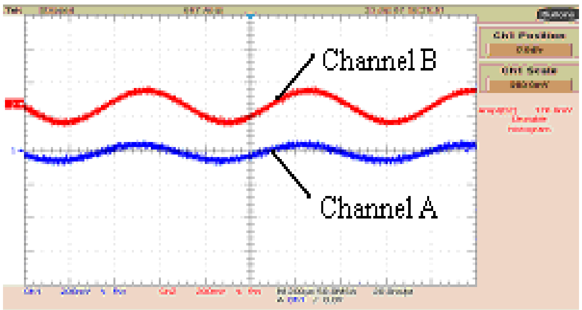

Fig.10-d Output signal of channel A and B when select volume function for channel A (others all bypass)

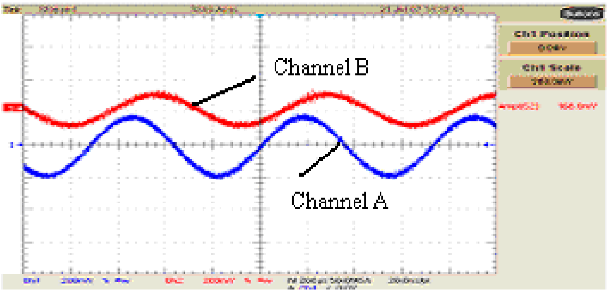

Fig.10-e Output signal of channel A and B when select parameter equalize and 3dB gain for channel A (others all bypass)

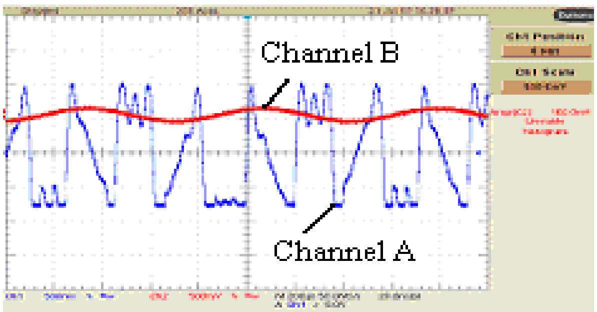

Fig.10-f Output signal of channel A and B when select parameter equalize and 16dB gain for channel A (others all bypass)

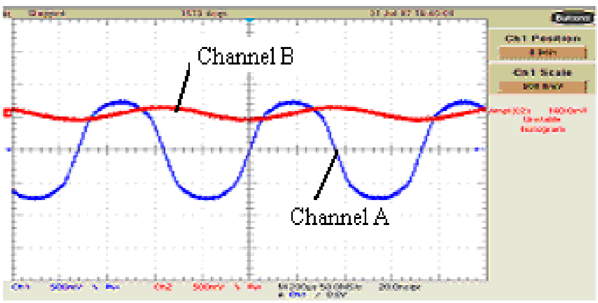

Fig.10-g Output signal of channel A and B when select limiter function for channel A(Channel A parameter equalize16dB, others all bypass)

Fig.10 Test waves of AEP

6. Conclusion

The CAN bus has the features that the transmission distance is far; the anti-interference ability is strong; the network building is easy. This article proposes one kind of distributed audio equipment control network based on the CAN bus. It introduces the network topology and the hardware design of ICANNet. It defines the physical layer, the transmission layer, the network layer protocol and the application data structure in detail. The software is compiled to realize the network control of distributed audio equipment base on the CAN bus. The experiment and the actual test result show that the proposed design of control network of distributed audio equipment base on CAN bus achieved the audio effect control and work state monitoring and control of audio equipment. Its performance is reliable. The design has a certain application value for domestic companies to develop network-controlled audio products. It provides useful references.

References:

[1] Zhang Qi-zhi, He Yu-yao, Li Lin, Shen Xiao-hui, The design of PROFIBUS-DP and CAN bus communications unit in drilling rig control system, *Proceeding of 2010 Computer Engineering and Technology*, Vol.3, 2010, pp:492-494.

[2] Feng Guo-sheng, Zhang Wei, Jia Su-me , Wu Han-sheng, CAN Bus Application in Automotive Network Control, *Proceeding of 2010 Measuring Technology and Mechatronics Automation*,Vol.1, 2010, pp: 582-585.

[3] Farsi M, Ratcliff K, Barbosa M, An overview of controller area network, *Computer & Control Engineering Journal*, vol. 3, No. 10, 1999, pp : 113-120.

[4] Marino P., Hernandez H., Dominguez M. A., Poza F., Machado F., Vazquez F., Applied fieldbus knowledge for ICT engineers , *Proceeding of 2004 International Conference on Coding and Computing*, Volume2, 2004, pp:178-182.

[5] Venzke M, Weyer C, Turau V, Application specific vs. standard Web service interfaces for the vertical integration of fieldbus systems, *Proceeding of 2005 International Workshop on Intelligent Sulutions in Embedded Systems*, 2005, pp:153-162.

[6] Cui Peng-peng, Liu Yang-ping, Research of X-ray Electric Cable Eccentricity On-line

Measurement System Based on DSP, *Instrument Technique and Sensor*, Vol.3, pp: 83-86.

[7] Liu Hai-ying, Xiong jun-qiao, Dai Lu-ping, Zheng Kuan-lei, Implementation of Adaptive Filtering on TMS320VC5402DSK, *Industry Control Computer*, Vol.23, No.5, 2010, pp: 100-101.

[8] Zhu Chao-wen, Ceng Shui-ping,Hardware Design and Realization of High-Speed Audio Acquisition System Based on DSP, *Communication and Information Processing*, Vol.29, No.4, 2010, pp:30-33.

[9] Cavendish D. Operation, administration and maintenance of Ethernet services in wide area networks, *IEEE Communications Magazine*, Vol.42 ,No.3,2004, pp : 72-79.

[10]Wang Jian-hua, Lu Xin, Number recognition terminal of lottery ticket based on DSP, *Journal of Shenzhen Institute of Information Tcchnology*, Vol.7, No.2, 2009, pp:19-23.

[11] Song Qing-kun; JiangJi-cheng, The design of CAN-BUS nodes based on the AT89C52 SCM, *Techniques of Automation and Applications*, Vol.28, No.5, 2009, pp:55-59.

[12] Dong Jing, Liu Shu-dong, Dai Xue-feng,etal. Intelligent ultrasonic distance measurement system of CAN bus based on P87C591, *Computer Engineering and Design*, Vol.30, No.12, 2009, pp: 2852-2855.

[13] Zhang Xiu-li, Zhang Rui-hua, Chen Wei-dong, Design of Digital Parametric Equalizer Based on Second-order Function, Proceeding of 2010 International Conference on Image Analysis and Signal Proceeding, 2010, pp: 182-185.