

# High Performance Integrated Audio System based on GOWIN FPGAs

Sun Zhenyu; Luo Xiangyu; Wang Yu

## Part I. Design Overview

### 1.1 design purpose

This design aims to provide a unified and highly integrated equipment platform for composite scenarios (especially for conferences, remote art creation, online multi-people chorus, etc., which require high audio quality). The device directly hit the three major pain points of the traditional separation of equipment, namely, complex setup, diverse equipment and expensive, can provide the most simple interactive interface and rich equipment interfaces for the user groups without professional equipment deployment and debugging experience, to meet the needs of most of the use of multi-input and output scenarios. The device adopts the standard interface, can be connected to all kinds of traditional audio equipment on the market, making the device more rich and diverse application scenarios.

### 1.2 Application Areas

The device can be fully applied to all kinds of online and offline teleconferencing, providing a complete, high-quality audio solution. In addition, the device supports up to 48kHz audio sampling rate and a variety of standard input interfaces.

(e.g. RCA and Canon interfaces), making it incredibly easy to set up a complete art creation and online broadcasting platform with this device as the centerpiece. The phantom microphone input makes it possible to use the device as a capture device for studio recording of professional singers.

### 1.3 Main technical features

The device covers the following main function points.

1. Active noise reduction. The unit offers three levels of noise reduction, which can be used in noisier environments and can significantly attenuate noise;
2. With automatic gain control function. The adaptive gain control provided by this device can compensate for microphone front-end attenuation and sound source volume changes, enabling the conference receiver to hear clear sound in most scenarios;
3. Echo suppression capability. The unit includes a variable step size NLMS filter capable of suppressing acoustic echoes up to 256mS;
4. Featuring a 10-band equalizer, the unit is preset with six sound effects, including rock, soft classic, and more;

5. With the matrix mixing function, the device supports up to 7 input channels and 3 output channels, the relationship between the input channels and output channels can be freely mapped and set the volume independently;

6. Features a Class D audio power amplifier that has a flexible topology setup (patent pending);

7. Driver-free across all platforms using standard UAC and CDC protocols.

#### **1.4 Key Performance Indicators**

##### Hardware specifications

1. DAC: Resolution 24Bit, THD+N @1kHz, -1dBFS < -83dB, DR > 95dB;
  2. ADC: Resolution 24Bit, SNR > 91dB, Input Impedance: 10kΩ;
  3. Class D amplifier: THD < 1.3% @ 1kHz, 60W mono, cement resistor load;
  4. Speaker overcurrent protection: response time <2uS.
  5. Phantom power: 24V/48V, 335uVpp ripple, 20MHz bandwidth; 6. Power consumption: average 8.5W (no audio input), PF ≈ 0.83.
- ##### Algorithm Specifications

1. Device delay: 64mS, the value is strictly fixed.
2. AEC: 11dB average suppression, using a linear filter, with a maximum of 256mS in the far end sequence;
3. Double talk test: p11 > 0.93;
4. AGC: Response time about 70mS, algorithm execution time <1mS;
5. ANS: 3-Level, each Level algorithm execution time is not greater than 5mS.

#### **1.5 Key innovations**

1. Take traditional echo suppressors, matrix mixers, power amplifiers, external sound cards, etc. and Highly integrated;
2. Provides a parameter-free interface, enabling inexperienced

users to quickly build a complete audio capture-processing-playback platform based on this device;

3. The use of pure FPGAs for signal processing, PWM generation and MOSFET driving of the dual-channel Class D amplifier, rather than using a finished IC, means that the design is extremely flexible. For example, this work, together with a specially designed Class D amplifier power stage, allows the device to flexibly change between bichannel, 2.1-channel (two channels with a subwoofer output), and trichannel (patent pending);

4. Integrated low-noise 24V, 48V phantom power supplies support most professional microphones;

5. Realization of a reverb effector for any room, which effector coefficients can be independently configured by the host computer. **Part II System Composition and Functional Description**

## 2.1 Hardware key part design and function description

### 2.1.1 Hardware structure of the whole machine

This design is based on the GW5AST-138 FPGA from Guangzhou Gowin Semiconductor, which has a LUT of up to 138K and a RISC-V core of 800MHz to meet the resource requirements of this design. The overall block

diagram of this design is shown in Figure 1.

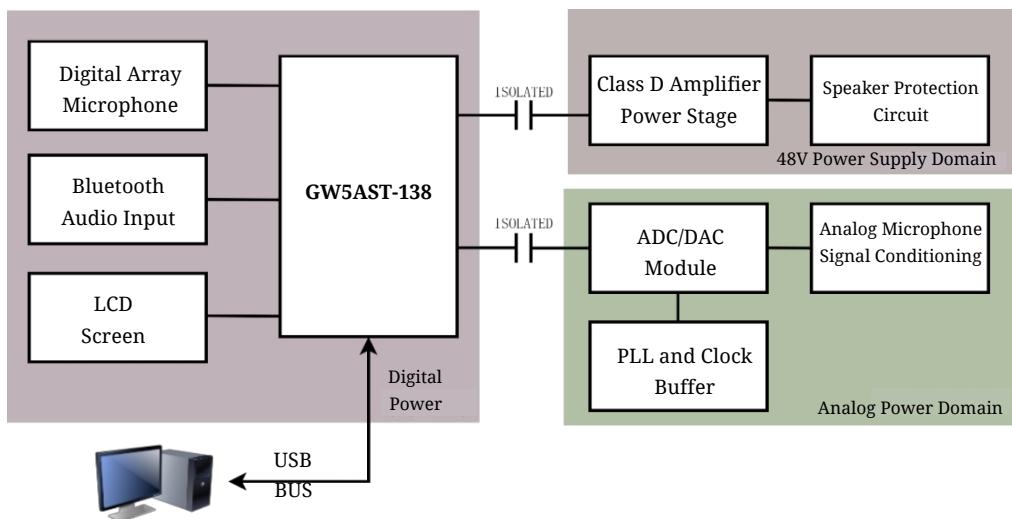


Fig. 1 Block diagram of overall design

The design is divided into three power domains because the design is for both power circuits and small-signal circuits. The digital power domain contains all the digital circuits that are not involved in audio acquisition and conditioning, including the FPGA, which is powered by a 48V power supply reduced to 5V by a 2-stage DC/DC converter. The power domain contains the Class D amplifier and speaker protection circuits, which are the most power-consuming part of the design, and is connected to the FPGA using capacitively coupled isolated drivers provided by NanoMicro. The analog power domain contains all the signal conditioning circuits, phantom power circuits and converter circuits, which are connected to the FPGA using a magnetically coupled isolator provided by ADI in order to minimize the noise conducted by the digital part. The internal schematic of the machine is shown in Figure 2.



Fig. 2 Internal diagram of the whole machine

### 2.1.2 Signal Acquisition Board

The PCB for the signal acquisition board is shown in the figure. The board adopts 4-layer PCB design and provides  $\pm 5V$  external input. The board adopts TI's PCM5102A for DAC and WOLFSON's WM8782S for ADC, and the difference between the signal ratio and the theoretical signal-to-noise ratio is less than 7% under 16-bit PCM input/output, which is considered to be able to realize the performance of the chip under the working condition.

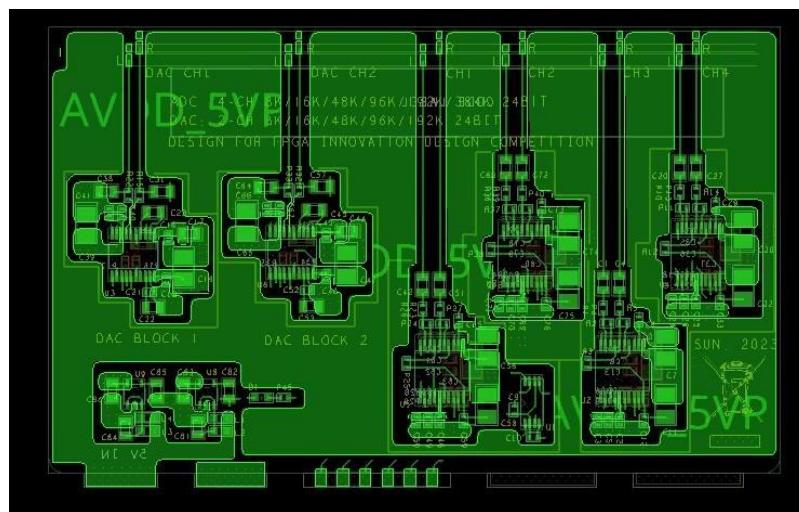


Figure 3 Signal Acquisition Board PCB Design Schematic

This design uses the SR-M500 as the reference standard for conference microphones to design the microphone front-end circuit.

The sensitivity of this microphone is  $-40\text{dB-V/Pa}$ . If we assume that the maximum ambient sound pressure level of this device is  $120\text{dB SPL}$ , then it can be

The maximum RMS value of the microphone is found to be

$$V_{RMS} = 10^{-\frac{40}{20}(V - Pa)^{-1}} \cdot 10^{\frac{(120-94)}{20}} (Pa) \approx 199.52mV \quad (1)$$

It is always desirable to have a microphone that produces a maximum excitation no greater than 85% of the ADC's effective range, so 4.2 times the gain of the preamp is introduced. To ensure the highest CMRR, the differential-single-ended converter in this design is accomplished using the INA1650. The chip has a set of laser-tuned resistors, which are matched to each other with an error of no more than 0.25%, giving the device a common mode rejection ratio of up to 90dB. Also during the PCB design we used active shielded drivers to drive audio cable shields that can be tens of meters long, where the distribution parameters between the shield and the core should not be neglected.

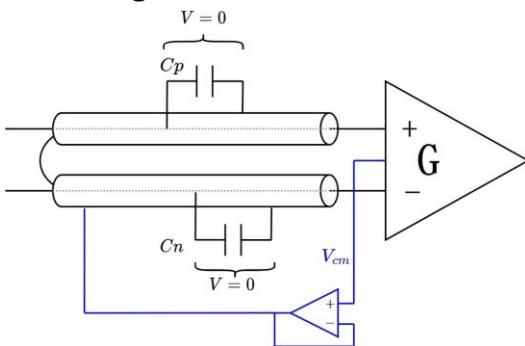


Fig. 4 Schematic diagram of shield drive

### 2.1.3 Class D Amplifier Power Stage

The Class D amplifier power stage board PCB is shown in the figure. The board is a 4-layer PCB design using Infineon's IRF6643 as the main switching device.

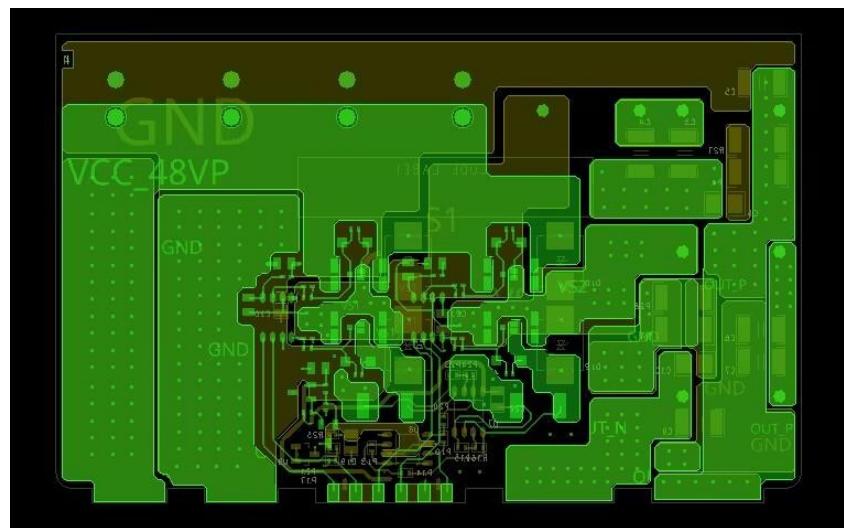


Figure 5 Class-D Amplifier Power Stage Board PCB Design Diagram

In order to ensure sufficient overcurrent capability, the PCB is machined with 2oz copper thickness on both the outer and inner layers, and the vias are spaced at critical locations. The board has complete over-current and over-temperature protection circuitry.

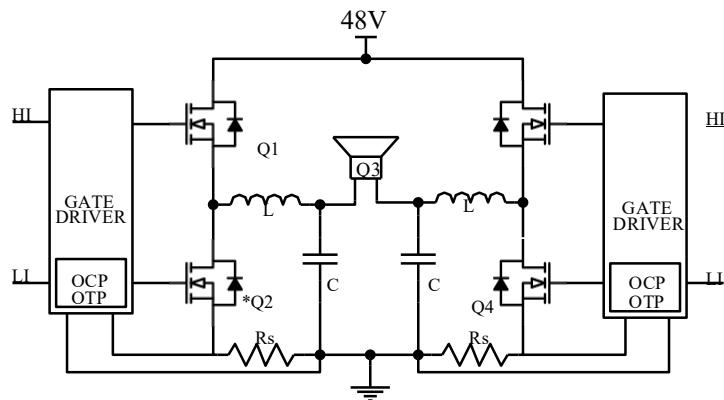


Figure 6 Class-D amplifier power stage block diagram

As shown in Fig. 7, the loudspeaker exhibits different impedance characteristics under different frequency conditions, so it has different impedance characteristics on the output LC filter network is a varying load.

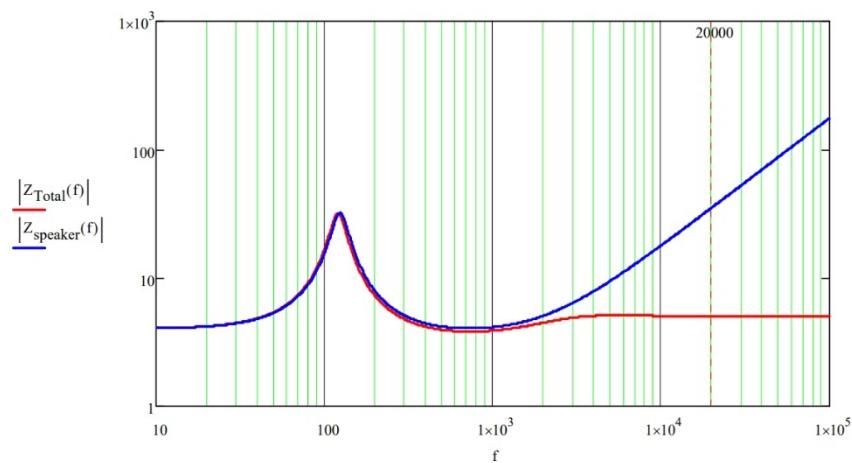


Fig. 7 Impedance characteristics of loudspeaker and loudspeaker with ZOBEL compensation network

When the characteristics of the loudspeaker at high frequencies are close to those of an inductor and the inductive impedance increases dramatically compared to the DC resistance, underdamping may occur at

the cutoff frequency of the LC filter network, especially when the Q value of the filter network is high, it is very easy to trigger the current limiting protection or burn the switching device when playing audio with a sufficient amount of high harmonics, so a ZOBEL network is introduced to compensate for the AC characteristics of the loudspeaker.

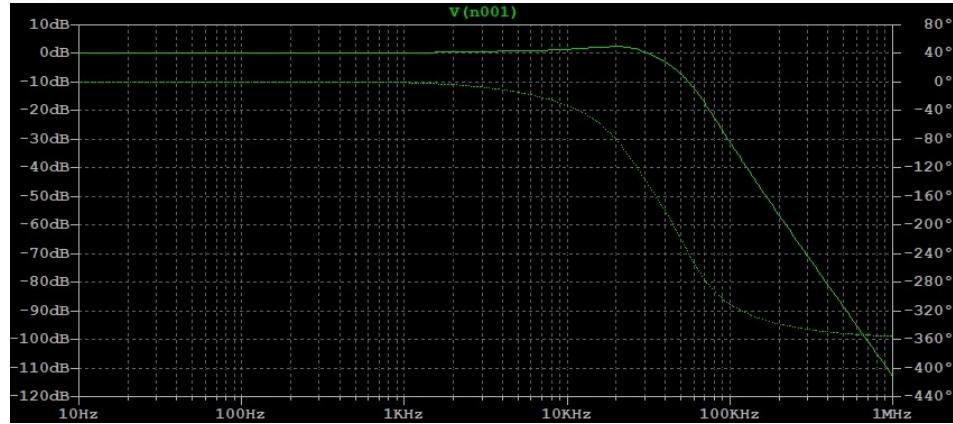


Fig. 8 SPICE simulation results of output filter (with speaker) after ZOBEL network compensation

The SPICE simulation results of the compensated loudspeaker are shown in Fig. 8, although there is still some overshoot at the filter resonance point, it is within acceptable limits.

## 2.2 FPGA Part Key Design and Function Description

### 2.2.1 Variable coefficient FIR filters

The variable coefficient FIR filter is the most used module in this design, which constitutes the circuit design of AEC, multirate signal processing, and reverb simulation in this design. As shown in the figure, the variable coefficient FIR filter consists of four main parts: control state machine, coefficient RAM, shift register and multiply-accumulator, and the clock and reset signals are omitted.

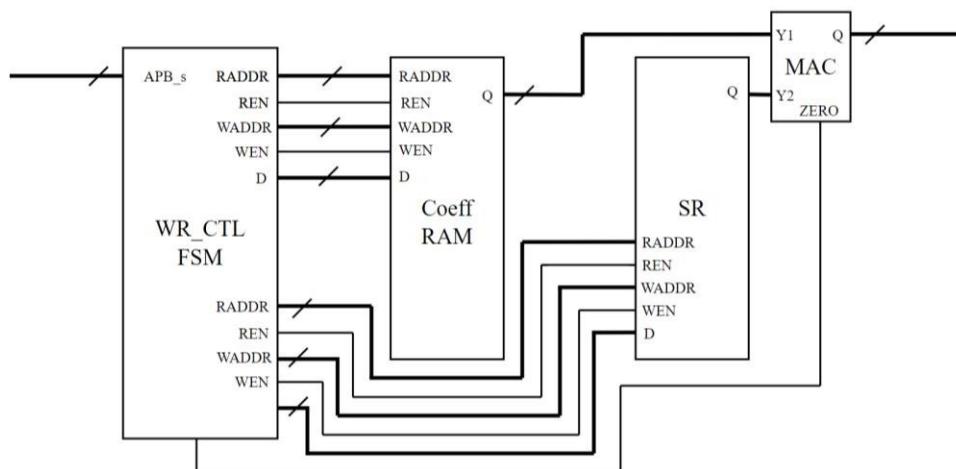


Figure 9 Variable Coefficient FIR Filter Block Diagram

The LFSR is also based on the BSRAM implementation, where the data is shifted by controlling the read and write addresses. It should be noted that this FIR is implemented serially, so only one FIR needs to be used.

1 DSP SLICE. when the number of FIR taps is large it can be done in serial-parallel combination. This technique is used for the core filter of the NLMS filter in this design.

### 2.2.2 AEC Preprocessing Module

The NLMS filter is the core of the AEC preprocessing module, which is composed of adjustable parameter FIR filter, error calculation circuit, normalization circuit and other parts. Among them, the idea and principle of adjustable parameter filter have been described in section, and only the coefficient write-back logic is described here.

The update law for the NLMS filter is determined by the following equation<sup>[1]</sup>.

$$(2) \quad h(n + 1) = h(n) + \eta \frac{x(n)}{\epsilon + ||x(n)||^2 e(n)}$$

where  $\eta$  is a hyperparameter and  $\epsilon$  is a regularization parameter set to prevent the divisor from being zero. This equation involves the operation of fixed-point division, which is accomplished using the trial quotient method and consumes more than 50 clocks Cycles. Indeed, for each deterministic input,  $x^{(n)}$  This item can always be

$$\epsilon + ||x(n)||^2$$

calculated in advance.

Therefore, the delay of the output of this module mainly depends on the delay of the serial adjustable parameter FIR. The critical path of this design on the FIR coefficients written back to the logic. In addition, the AEC preprocessing module also needs to include a double-talk detection circuit, when double-talk occurs, the NLMS filter error increases steeply, and at this time, if the filter coefficients are updated according to the original strategy, it is easy to cause the system to diverge. There are two common processing ideas, one is to use a variable step-size strategy and set the hyperparameter  $\eta$  to a smaller

value when double-talk detection occurs, and the other is to use a strategy that pauses the FIR update. The second simpler implementation is used in this design.

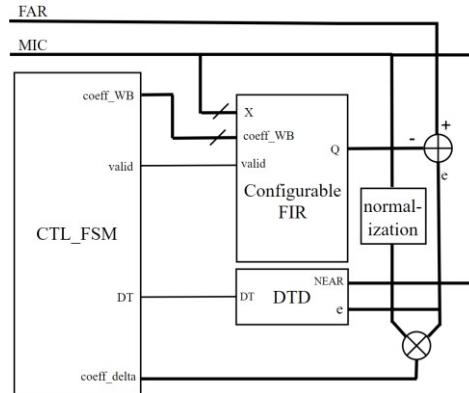


Figure 10 Block diagram of the AEC preprocessing module

The DTD module is used for practical double-talk detection, in which we utilize the microphone signal with the error

The interrelationship  $r_{re,mic}$  of the difference signals is used as a decision variable and this interrelationship is estimated using the EWMA method, i.e.

$r_{re,mic}(n + 1) = \lambda r_{re,mic}(n) + (1 - \lambda)r_{re,mic}(n)e(n)$  (3) where  $\lambda$  is a hyperparameter that determines the time horizon of smoothing. If  $1 - \lambda \ll 1$ , then it is recognized that

$$r_{rmic,mic}$$

For double talk to occur, here  $r_{mic,mic}$  is also estimated using the EWMA method.

$$r_{mic,mic}(n + 1) = \lambda r_{mic,mic}(n) + (1 - \lambda)r_{mic,mic}(n)e(n) + (1 - \lambda)r_{rmic,mic}(n)e(n) \quad (4)$$

In practice, this module can have a large impact on voice quality.

### 2.2.3 Class D Amplifier Modules

As shown in Figure 11, the Class D amplifier module consists of four main parts: the upsampling module, the sigma-delta module, PWM modulator module and MOSFET driver and protection module.

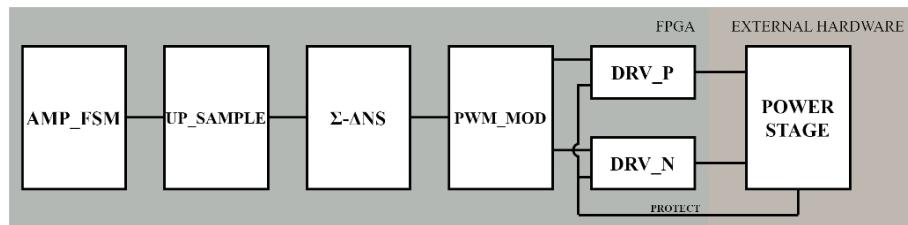


Figure 11 Class D Amplifier Module Block Diagram

The sampling rate of the audio stream output from the USB of this system is 48kHz, if it is modulated directly, it is an extremely great challenge for both the PWM modulator design and the filter design<sup>[2]</sup>. Therefore, this design uses the upper sampling module will be 768kHz, and then the fifth-order noise shaping to get 8bit oversampling signal, the signal through the unipolar PWM modulator to get the modulation signal,

the modulation signal through the MOSFET driver and protection circuit to increase the dead zone, the dead zone can protect the MOSFET upper and lower bridge arms in the alternating moment will not appear at the same time on, otherwise it will damage the MOSFET.

In this design, a  $\Sigma\Delta$  noise shaper with a fifth-order CIFB structure is used, the block diagram of which is shown in Fig. 12, and the coefficients are computed using SDtoolbox and the root trajectory is plotted to verify its stability. The simulation results show that all the poles of the system are located in the unit circle, so it can be considered as closed-loop stable.

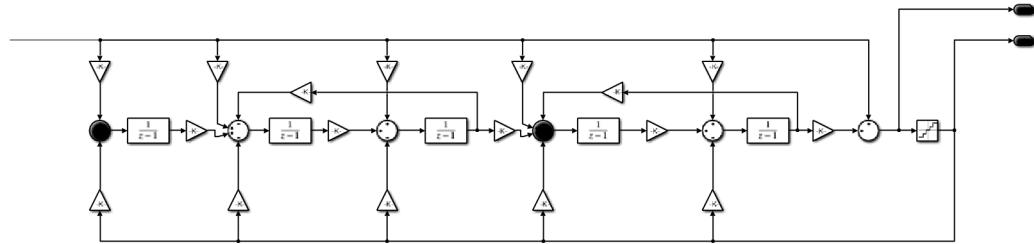


Fig. 12 Simulink block diagram of sigma-delta noise shaper

The upsampling design of this module is multiplexed with the upsampling filter of the Multi-Rate Signal Processing Module, so it is not discussed in detail in this section.

#### 2.2.4 3A and multirate signal processing modules

In this design, 3A (ANS, AGC, AEC) is a core function which ensures the audio quality of the teleconference. The algorithm runs on the RISC-V core built into the GW5AST-138 and uses the APB bus to interact with the external module. In fact, due to the high sampling rate of the external audio and the limited performance of the RISC-V core, the algorithm is downsampled to 16kHz and then upsampled and interpolated to 48kHz.

The block diagram of the module is shown in Figure 12.

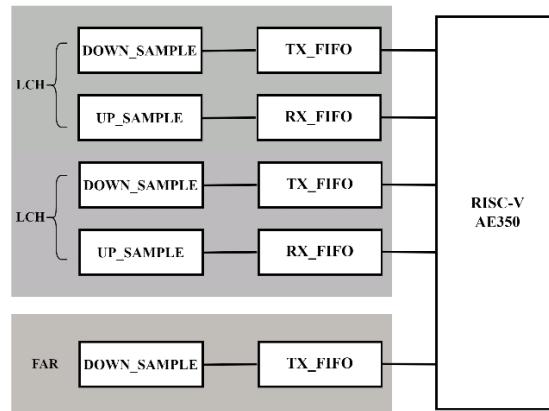


Figure 13 Block diagram of 3A and multirate signal processing

The remote input comes from an external USB data stream, but this data stream is only used as a reference signal and does not need to be

output, so no upsampling is done in this branch. This design upsampling and down-sampling multiplier are 3 times, so the use of a level of extraction or interpolation can be completed, in this parameter, the downsampling filter passband is  $8k$ , the transition bandwidth is set to 200Hz, the use of the first type of linear phase filter design; the up-sampling filter passband is  $8k$ , the transition bandwidth is appropriately relaxed, the use of the first type of linear phase filter, all the filter coefficients are used in a fixed-point Q15 quantization format. All filter coefficients are in Q15 fixed-point quantization format. In fact, for the upsampling process, the data contains a large number of 0 can be optimized, but the

This design uses a serial FIR filter, leaving aside the data delay time, the optimization is not very meaningful, and this module needs to be reused many times in the project, so the circuit is still designed according to the direct-type filter framework.

The 3A algorithm, which is the most complex part of the design, is based on the WebRTC open-source framework, and all data is fixedpoint processed to allow the algorithm to run in real-time on an AE350 RISC-V processor.

The AEC algorithm is accomplished by the FPGA in collaboration with the AE350, in which the AE350 processes the 早 phase echo, which is accomplished in the frequency domain<sup>[3]</sup>, as shown in the following block diagram.

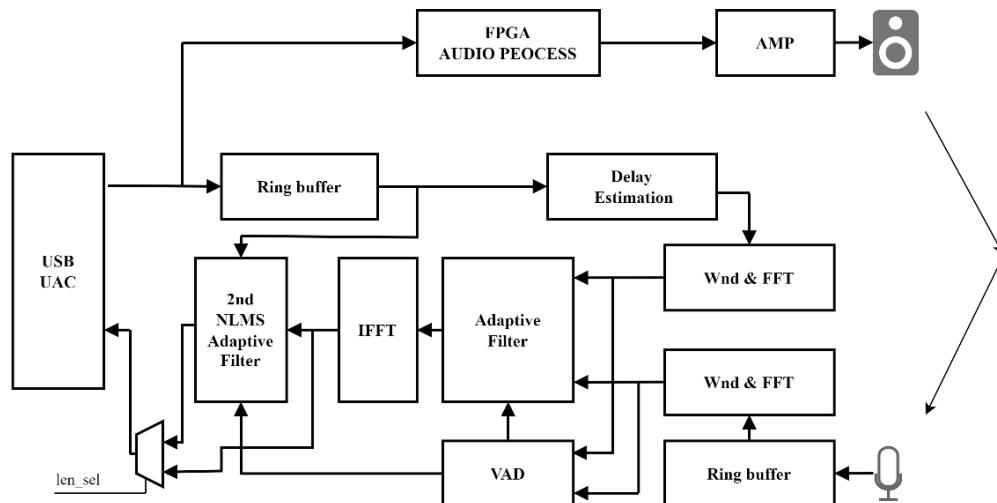


Figure 14 AEC Processing Module Block Diagram

Since the second stage of the NLMS filter has a very large impact on the quality of the speech signal, a multiplexer is introduced here that allows this filter to be bypassed in the program.

The ANS is implemented using the NS module in WebRTC, which calculates the a priori SNR and classification features for Wiener filter denoising by means of spectral estimation, so the whole noise reduction

algorithm is implemented in the frequency domain. In fact, in the test, we found that the noise suppression performance of WebRTC is weak for some single frequencies, so we added a filter bank containing 32 independent second-order IIR filters and a gain-tuned gain coefficient, and the center frequencies of the 32 filters are mapped by the Meier scale.

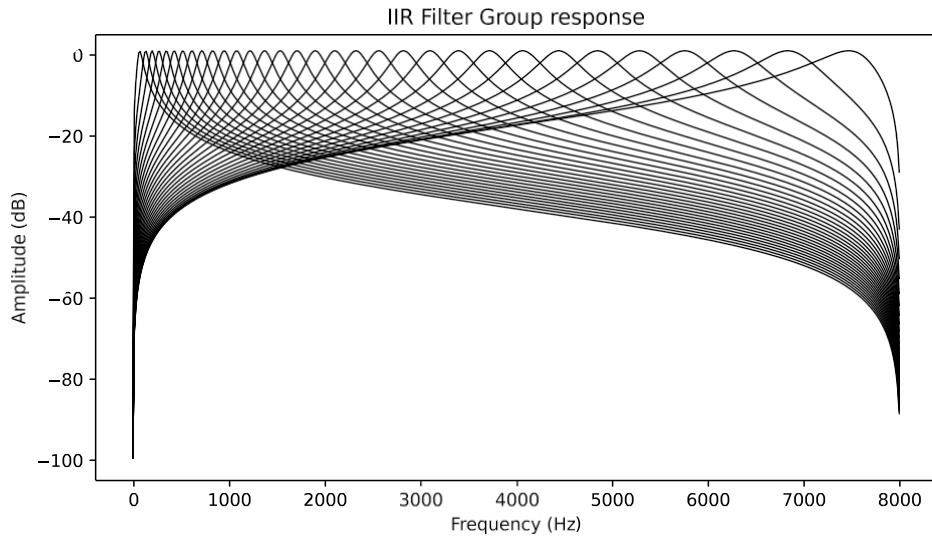


Fig. 15 Frequency Response Characteristics of Filter Bank

The transfer function of a single IIR filter is given by the following equation.

$$H(s) = G_0 \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{a_0 + a_1 z^{-1} + a_2 z^{-2}} \quad (5)$$

included among these

$$\begin{cases} b_0 = \sin \omega^0 \\ b_1 = 0 \\ b_2 = -\sin \omega^0 \\ a_0 = 1 \\ a_1 = -2 \cos \omega_0 \\ a_2 = 1 \end{cases} \quad (6)$$

When the microphone receives a certain number of strong signals with extremely concentrated frequencies, it will change the filter gain coefficient

$\circ G$ , i.e., suppressing signals in this frequency band. It should be noted that although this filter bank is able to achieve a trap-like effect, it is still unable to suppress microphone whistling in the presence of acoustic feedback, and the whistling suppression trap for microphones is

discussed separately below. The overall block diagram of the ANS is shown in the following figure.

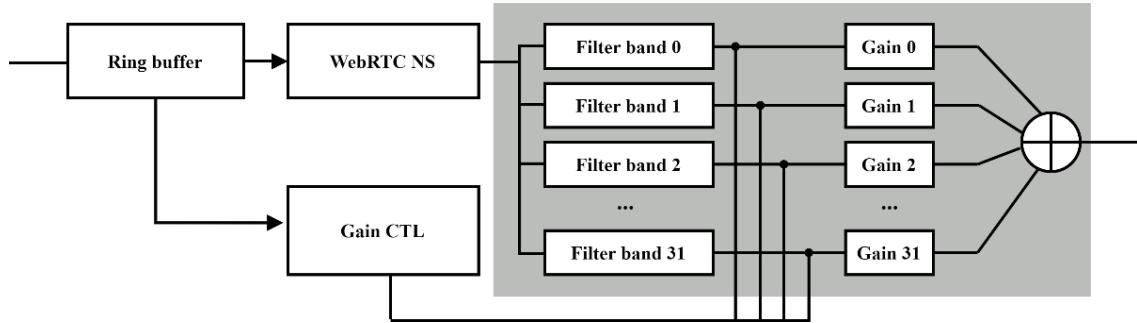


Figure 16 ANS Design Block Diagram

The AGC is a very critical module in the link, which compensates for the envelope fluctuations caused by the ANS, and provides an additional amplitude boost when the input amplitude is low. In fact, after the introduction of AGC, for some scenarios where the amplitude of the audio signal changes drastically, such as an orchestra performance with some strong drum beats, the output audio amplitude may fluctuate, so the AGC module can also be bypassed in this design.

### 2.2.5 Whistling Suppression Module

Whistling suppression is one of the more commonly used modules in audio systems. There are three main ways to realize whistling suppression: frequency shift method, trap method and adaptive filtering method<sup>[4]</sup>. In this design, we first try the frequency shift method, the method through the time domain resampling, the speaker signal all frequency band to the lowfrequency shift of about 150Hz, but after trying the method will seriously affect the sense of hearing, especially for the human voice in the midfrequency band of the sense of hearing change is very strong. In fact, the commonly used frequency shifted whistling suppression equipment will not be used to shift the full spectrum, but rather to shift a specific frequency band, which will involve a series of operations such as subbanding. The whistling suppression based on adaptive filter, despite the better simulation effect, is still a linear estimator in nature, and its suppression effect is poor in the actual phantom test, and finally we adopt the trap method. The key to the trap method is the design of a fixed frequency trap, which has the following requirements:

1. computationally convenient, it is desirable to give closed formulas for the construction of the traps;
2. It is a linear phase system to avoid frequency dispersion affecting the listening experience;
3. Deep enough trap volume to adequately suppress whistling.

Based on the above requirements, we propose a method based on frequency domain design which utilizes an all-pass filter decimated with a point-pass filter to obtain a trap, i.e.

$$H(e^{j\omega}) = H_0 [1 - G(e^{j\omega})] \quad (7)$$

where  $G(e_{jw})$  is a linear phase function, i.e., the group delay is a constant. For hardware implementation, we use the first type of linear phase system design where the tap coefficients are of even symmetric form, which in practice means that we perform a time-domain truncation of this filter transfer function. From the signal correlation theory, it is known that the type of the time domain window function determines the trap depth and Q value of the traps, which are checks and balances. We choose the Kaiser window as the time domain window function with the following expression:

$$Kaiser(n) = \frac{I_0[\beta \sqrt{1 - (1 - \frac{2n}{N-1})^2}]}{I_0(\beta)} \quad (8)$$

Although the window function contains a zero-order Bessel function, this does not violate Requirement 1 because the window function is given as a table of coefficients in the engineering implementation, and the advantage of using the Kaiser window is that the trap Q value can be adjusted by choosing different  $\beta$

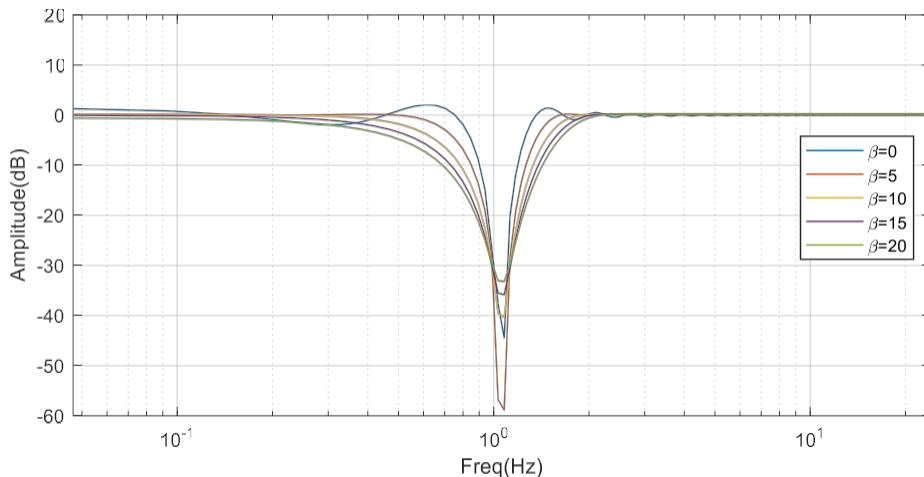


Figure 17. Relationship between the trap frequency response and  $\beta$

The closed-form expression of the trap based on this theory is given by Eq. (9).

$$\left( \frac{2\pi N f_s}{f_c} \right) = 0 \quad , \quad n = \frac{h(n, N-2)}{h(N, N-2)} \quad (9)$$

$$\{-2 \cdot \text{Kaiser}(n, \beta) \cdot \cos f_c(2N - 1), \quad \text{otherwise}$$

From (9) it is easy to construct a linear phase trap of the first class of length  $2N-1$  with trap center frequency  $f_c$ , in order to obtain  $f_c$  it is necessary to use some estimation means. The whole idea of the algorithm is as follows.

1. The 64ms signal was windowed using a Hamming window and fast Fourier transformed;

2. Calculate the spectral peak point amplitude  $A$  and frequency  $f_{peak}$ , which is considered to be the suspected whistling point if  $A > A_0$ , where  $A_0$  is a threshold;
3. If  $\Delta f_{peak} < f_0$  in 3 consecutive frames, where  $f_0$  is a threshold, it is considered to be in the  $f_{peak}$ . A whistling occurs at the place;
4. Iteratively solving  $f_{peak}$ , which is obtained based on an for  $f$  using spectral interpolation algorithm we propose; interpolation
5. Bring in (9), solve for the filter coefficients, and update  $w_i$  to the FPGA via the APB bus.

It should be noted that the Q value of the filter should not be too large, otherwise it is not easy to align the trap point to the whistling frequency.

#### 2.2.6 Array Signal Processing Module

This module mainly realizes the delayed summation beam assignment of the circular array, due to time constraints we did not complete the adaptive beam assignment before the time limit, due to the fact that this part involves the theory of array signal processing, the realization is more complex, and here in the simplest and most clear way to illustrate our design ideas. Consider the response of a circular array to a narrowband far-field plane wave.

$f(t, p)$ , it is important to note that even though the speech signal does not conform to this assumption, it is still engineered to be treated as if it satisfies this premise, then the response can be expressed as<sup>[5]</sup>

$$f(t, p) = f_{src}(t, p) \circledast v(t, v) p \quad (10)$$

where  $\circledast$  denotes element-by-element convolution and  $v(t, p)$  is defined as  $v(t, p) = [\delta(t - \mathbf{a}^T \mathbf{p}_0), \delta(t - \mathbf{a}^T \mathbf{p}) \dots, \delta(t - \mathbf{a}^T \mathbf{p}_1), \dots, \delta(t - \mathbf{a}^T \mathbf{p}_n)]^T$

Where  $\mathbf{a}$  is the wave vector, pointing in the direction of the incoming wave towards the array, and  $p$  are the coordinates of the array elements under the Cartesian system. This signal is now weighted by the filter to obtain the array output signal

$$y(t) = \sum_{n=0}^{N-1} \int_{-\infty}^{+\infty} f_n(t, p_n) h_n(t - \mathbf{a}^T \mathbf{p}_n) d\tau \quad (12)$$

Swap the order of integration and summation and do the Fourier transform to get

$$Y(w, k) = H(w)^T v(k)_{src}(w) \quad (13)$$

and let  $\gamma(w, k) = H(w)_T v(k)$  where  $k$  is the number of waves defined as  $k = 2 \pi / \lambda a$ . delayed summation beam assignment

The main thing about the shape is to choose the appropriate  $H$  to change the frequency-beam domain response  $\gamma(w, k)$  of the array so that the direction of maximum gain points to the direction of the sound source. In the engineering realization, we measure the array element parameters and compute the generation of  $H$ , which is written into the variable coefficient FIR filter described in the previous section, so that the whole block diagram is as follows.

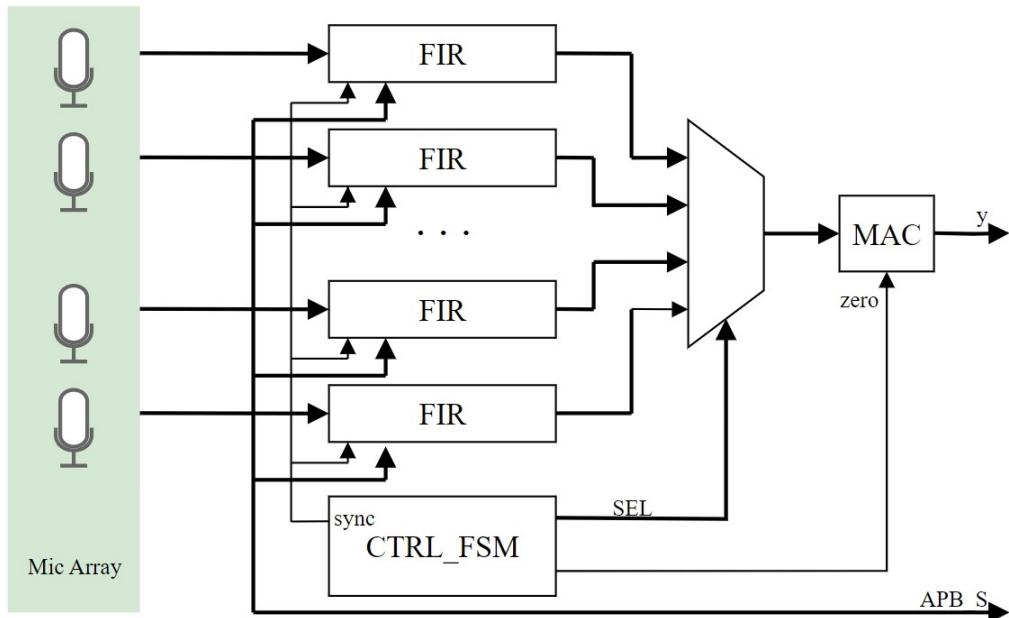


Fig. 18 Block diagram of delayed summing beamformer

### 2.2.7 USB Controller Module

The USB controller in this work is done with FPGA, no external controller or PHY chip is used. In fact, the function of USB in this work is relatively simple, there are two main parts: using UAC protocol to transmit audio streams, and using CDC protocol to transmit data streams.

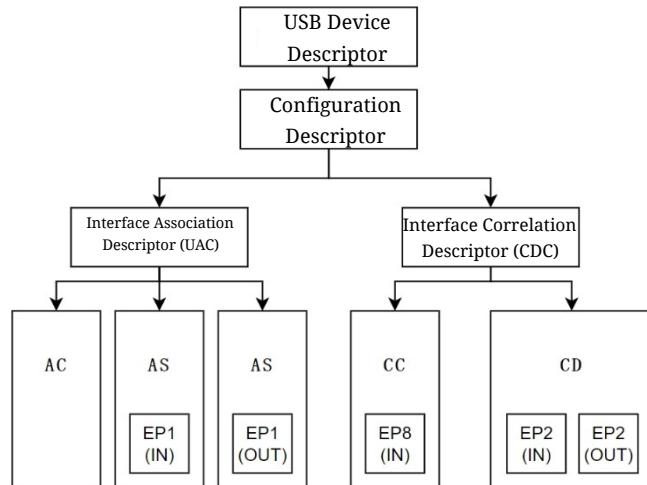


Figure 19 USB Composite Device Topology

The CDC is used to send configuration parameters, such as reverb configurations, filter parameters and EQ parameters, from the host computer to the AE350 side or FPGA side, using the xmodem protocol.

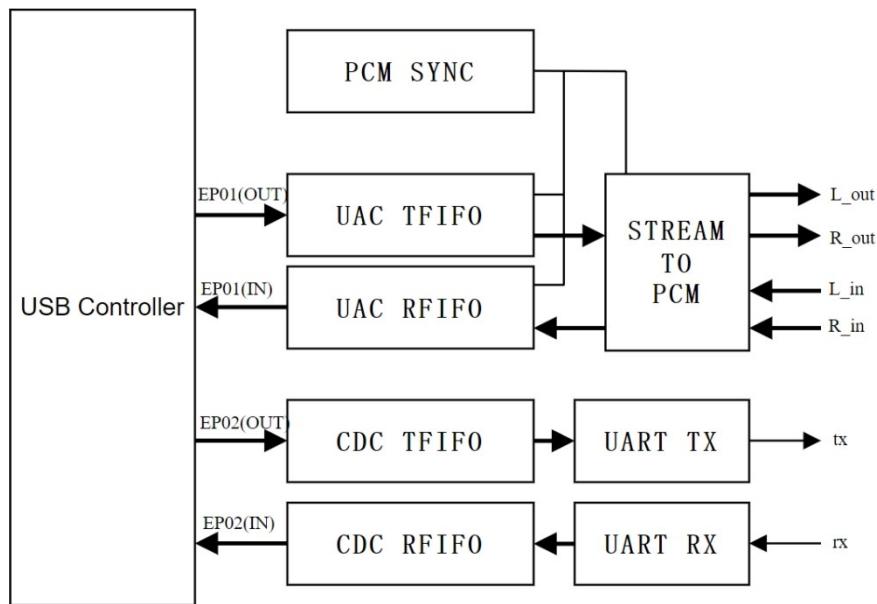


Figure 20 USB Composite Device External Interface Block Diagram

## Part III Completion and performance parameters

### 3.1 USB UAC Device Recognition Test

This work is developed based on the standard UAC protocol, so it has good platform compatibility. Take Ubuntu and Windows 10 platform as an example, when this device is turned on, connect this device to the PC via USB data cable, and the recognition result is shown in Fig.

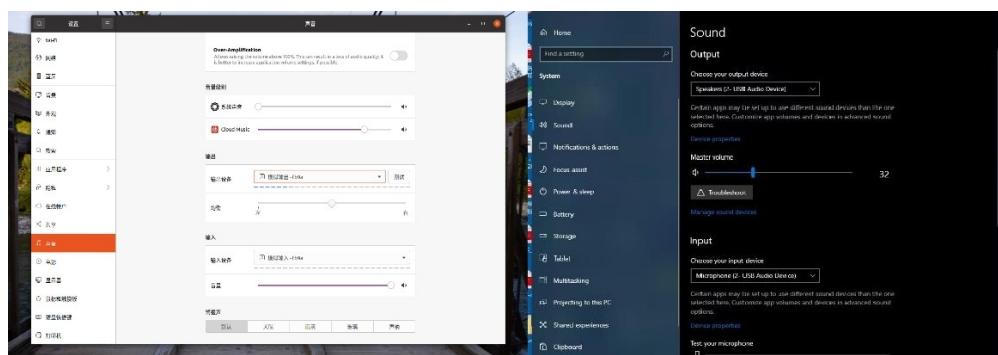


Figure 21 Device Recognition Results on Ubuntu and Windows Systems

Test results show that the software can correctly discover the device and automatically set the device as an input/output device.

### 3.2 Power Consumption Test

Connect the device to the power supply and in standby mode, measure the power consumption of the device and the results are shown below.



Figure 22 Standby Power Consumption Test

Tests show that when no audio signal is connected, the average power consumption of the device fluctuates around 8.5W. For this standby power consumption, we conducted a detailed module-by-module test, and the data are shown in Table 1.

Table 1 Standby power consumption component analysis and power efficiency analysis  
 power domain sports event wastage power Power efficiency

Switching Power Supply Domain	FPGA Boards and Peripheral Modules	~2.4W	/
LCD screen		~0.6W	/
Class D Amplifier No-load loss		~1.2W	/
Linear Power	Phantom Power Domain	~1.6W	72.4%
<b>3.3 Comprehensive signal processing circuit use scenario test</b>			
No-load and LDO losses		~0.2W	/
		~0.6W	/
		~1.6W	33.3%

Scenario 1: Class D amplifier effect test



Figure 23 Class D Amplifier Effectiveness Test

Scenario 2: Noise reduction test under strong noise environment



Figure 24 Noise Reduction Tests in Noisy Environments

Scene 3: Pickup effect test at different distances



Fig. 25 Pickup effect test at different distances

Scenario 4: Conference effect test in an echo environment

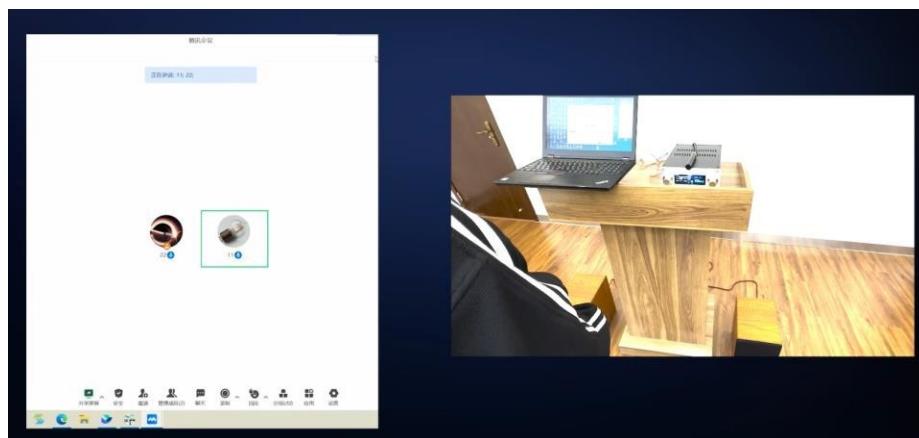


Fig. 26 Conference effect test in echo environment

Scenario 5: Array microphone pickup effect test



Figure 27 Microphone Array Conference Effectiveness Test

Additional Demo: Any Room Reverb Generation

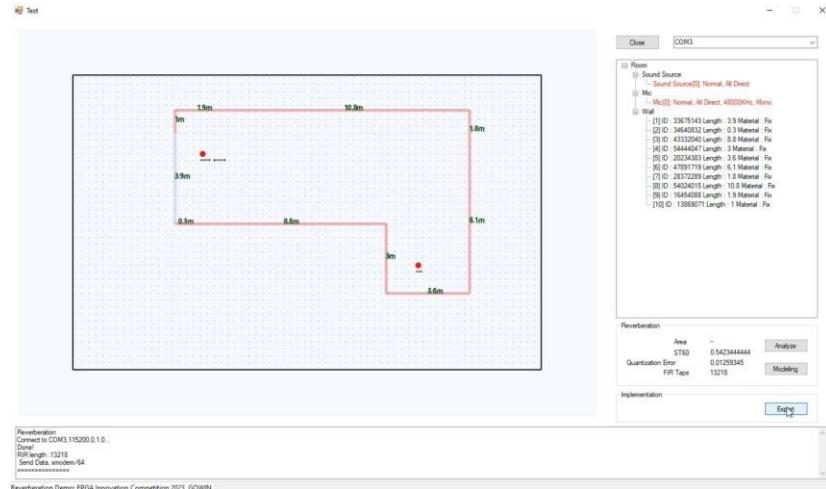


Figure 28 Upper computer configuration

The purpose of this demo is to demonstrate the concept of FPGA for real-time audio generation, where the upper computer is used for RIR Calculate (using the mirror source method), configure the variable coefficient FIR inside the FPGA via USB.



Fig. 29 Reverberation effect in any room in real time

## Summary of Part IV

This work realizes a high-performance integrated audio system based on GW5AST-138 FPGA, including Class D amplifier, 3A, matrix mixing and other functions commonly used in professional audio systems, and shows excellent performance in the actual scene test, basically completing the basic project included in the competition question. However, there are still places where this design can be expanded:

1. Add wireless microphone access;
2. Optimizing the signal-to-noise ratio of Class D amplifiers and introducing feedback inputs to make them commercially viable;
3. Add Network Push Streaming to enable users on the same network segment to access audio over Ethernet.

This work was completed independently by team members in a short project cycle, including basic hardware selection to RTL code writing, embedded development, so many features still have a certain distance compared with market products, but in the process, we practiced how to define the product designed from the perspective of product developers, how to set up the interaction of the product to meet the optimal user experience, as well as designing the corresponding test to test the product to see if it meets the expected targets.

During the development of this work, team members designed a series of instruments for testing the work, including some high-performance data acquisition boards, high-performance sinusoidal signal generators and other instruments, which are able to meet the parameter sweep measurement of conventional audio systems and acoustic devices.



Figure 30 High-performance data acquisition board designed by our team

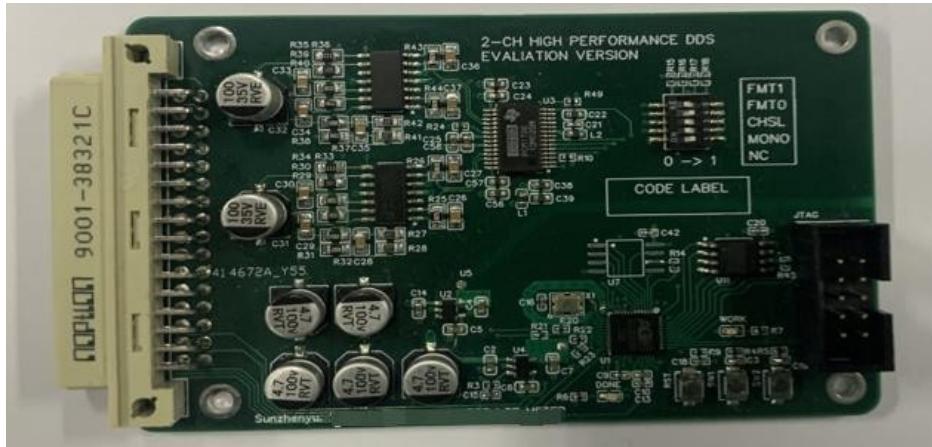


Figure 31 High-performance sinusoidal signal generator designed by our team

These instruments have been instrumental in developing our Class D amplifiers and building our entire audio test system.

In addition to this, most of the RTL pre-simulation code for the project was developed in an EDA suite by the team members.

This is done on a program (the software is based on iVerilog and GTKWave). The tool is completely open source.

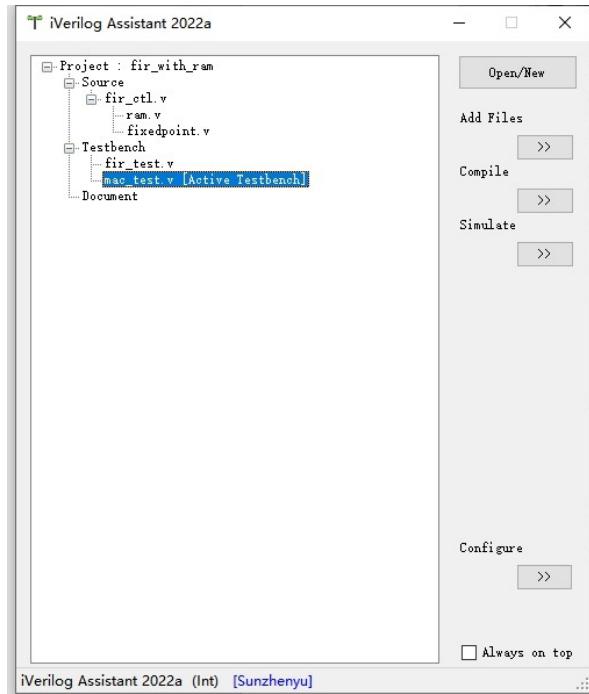


Figure 32 Tools designed by members of the team

## Part V      References

- [1] Dai Jing, Zhao Yanzhou, Zhang Hui, et al. FPGA realization of NLMS adaptive filter[J]. Journal of Shenyang Architecture University (Natural Science Edition),2011,27(1):190-195.
- [2] Yu Zeqi. Research on key technology of UPWM type digital class D audio amplifier[D]. Shaanxi:Northwestern Polytechnical University,2015. doi:10.7666/d.D689521.
- [3] Shen Lizhi. Research and implementation of audio system based on WebRTC echo cancellation[D]. Chongqing:Chongqing University of Posts and Telecommunications,2021.
- [4] Hao Guoli. Research on acoustic feedback whistling suppression algorithm[D]. Nanjing University,2012.DOI:CNKI:CDMD.2.1012.376060.
- [5] HarryL.Trees,DiS,TangJun. Optimal Array Processing Techniques[M]. Tsinghua University Press,2008.