Audio Engineering Society

Convention Paper

Presented at the 117th Convention
2004 October 28–31 San Francisco, CA, USA

Novel Subwoofer Signal Conditioner Design Using a Field Programmable Analog Array and Software Tools

Ian Macbeth[1], Tegid Roberts[2]

[1] Anadigm Ltd, Crewe, Cheshire, CW1 6ZQ, UK
ian.macbeth@anadigm.com

[2] Cadarn Consulting Ltd, Eynsham, Oxfordshire OX29 4HH, UK
jtr@cadarn.net

ABSTRACT

The control and real-time software programmability of low frequency audio signals in the analog domain is inaccurate, cumbersome and expensive. In the digital domain there are issues relating to low frequency distortions, latency and design time. A fully programmable analog array IC methodology is presented which combines the benefits of DSP programmability with analog signal processing by way of a case study demonstrating software design tools and custom software configuration models. This single chip Subwoofer Conditioner solution implements a sub-sonic filter, adjustable audio compressor, Linkwitz transform filter and low pass output filter with full software control. Performance measurements of this implementation as well as further enhancements to the software models are also discussed.

1. SUBWOOFER CONDITIONER DESIGN CONSIDERATIONS

1.1. Background

The modern popularity of active subwoofers has been driven by automotive sound, Dolby Surround formats and more recently DVD and home theatre systems in which a more controlled bass is desirable for accurate reproduction of high quality audio. The small cabinet provides convenience whether in the home or tucked within an automotive interior but generates design difficulties due to cabinet resonances.

Original subwoofers employed acoustic suspension designs and then later James Novak developed equations [1] to enable the control of bass reflex speakers. Other designs used excursion subwoofers including the Infinity Servo-Static, introduced over 30 years ago. In later years more sophisticated control of closed cabinet design has been employed using the Siegfried Linkwitz transform published in 1980 [2]. This technique along with band and multiband compression are widely used in home, studio and automotive designs. More recently subwoofer designers have pursued active auto-tuning techniques to compensate interior acoustics and user adjustable designs allowing the choices between controlled subtle bass or a dramatic sound for movie watching.

From an analog electronics point of view subwoofer design has moved slowly from being a fixed, passive acoustically tuned design to an active, equalized and manually tunable one. Modern subwoofer design requires preamplifier level control (limiting or compression), steep low pass filtering, DC removal and cabinet tuning filtering.

1.2. A Need for Tunability

The Linkwitz transform allows the tuning of Fc and Qtc of closed cabinet designs and compression generates predictable output levels from increasingly unpredictable broadcast source levels. Plus, in the car and in the home the desire for a remote controllable system via preset or user definition does not fit well with conventional analog discrete circuitry.

The ability to design, e.g., a single active analog loudspeaker crossover circuit, deploy it in a range of products and for each one simply reprogram it to apply the specific acoustic crossover, EQ and compression characteristics has not been possible previously. To date, digital tuning of analog circuits has been restricted to the use of digital potentiometers alongside other discrete components, where overall precision and tolerances can vary between 5% and 20%. Such systems are complex, unreliable and expensive. Matching is not tightly defined and the designer must supply control registers and addressing mechanisms. All this can be prohibitive.

What is required is an integrated digitally controlled programmable solution that can be integrated with modern user interface standards. Field Programmable Analog Array (FPAA) or DSP solutions provide the obvious route to such solutions.

A DSP solution would provide all the control required and the possibility of experimental room equalization algorithms [3][4] which offer excellent promise but so far due to processing cost and considerable design difficulties have eluded the market place in any numbers. Their effect is so far too subtle to fully justify the processing resource required. DSP provides excellent temporal processing, programmability and repeatability but have latency issues that are particularly problematic for real-time playback of broadcast material. For all but the most expensive systems even simple DSP filtering a level control solutions are not employed in subwoofer design due to high cost & complexity in a traditionally analog industry.

An alternative solution is available. The Field Programmable Analog Array (FPAA) can provide a cost effective programmable analog solution that not only has traditional processing techniques available to it but could conceivably be used in an auto-tune capacity given that the technology has real-time programmability and signal generation capability. Alternatively, it can be twinned with DSP to offload from the DSP those processor-intensive tasks such as high-end filtering and compression that are more suitable to highly parallel analog processing.

2. THE PROGRAMMABLE ANALOG APPROACH

2.1. The Enabling Device - the Field Programmable Analog Array (FPAA)

Within an FPAA, a wholly analog CMOS-based circuit network is combined with a programming layer to allow designers to implement an extremely wide variety of signal processing functions, applied using digital configuration data. Circuit function is dependent only on the matching of components, thus the technology delivers very high precision (between 0.1% and 1%), significantly better than discrete analog components. It also offers excellent linearity (typically exceeding 94dB or 0.002% in a filter block, for example) and typical bandwidths range from DC to 2MHz – ample for audio. A price-competitive 16-bit fixed point DSP/CODEC solution implementing a similar function might deliver 80dB (0.01%) at best due to noise and rounding errors.

The nature of the FPAA is that it is *field* programmable. Moreover, it is dynamically and partially re-configurable. This means that by applying the appropriate configuration data, the FPAA can:

- perform a single function when powered-up
- perform a sequence of different functions over time
- perform a dynamic function which can partially change over time, or can have its attributes adjusted freely without interrupting operation.

The next section outlines the process of designing an FPAA application. The result of this process is typically either a set of configuration data, a collection of data sets, or a set of software instructions.

In the simplest case, a design may be fixed having completed a prototype, and a single set of data is available to configure the device. The FPAA is volatile, i.e. like a DSP it is programmed on power-up. The simplest way to do this is to couple it to an EEPROM containing the data. When power is applied, the FPAA(s) will download configuration data from the EEPROM (see Figure 1).

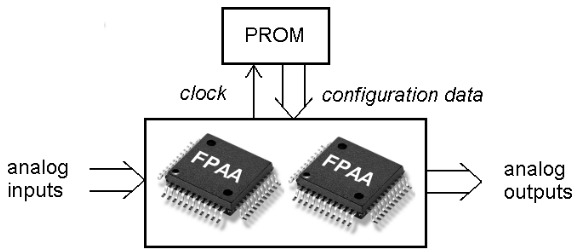


Figure 1 - Statically programmed FPAA(s)

More flexible operation is afforded with the use of some form of logic controller, which can be selective of the data to be loaded into the device. For example a micro-controller, which has been programmed to perform reconfiguration of all or part of the FPAA, and applies configuration data selectively based on controls or system conditions (see Figure 2).

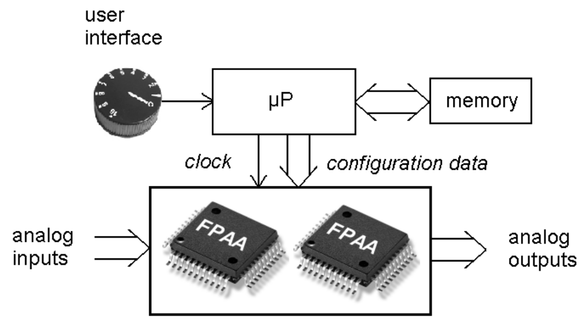


Figure 2 - Re-configuring FPAA(s)

The 'user interface' might be anything from a simple EQ "tilt" select switch at the back of a loudspeaker, to a sophisticated graphical user interface in a digital audio workstation controlling complex analog signal processing. For both, the circuit design and reconfiguration control development process are the same, and are very straightforward. These are described in the next section.

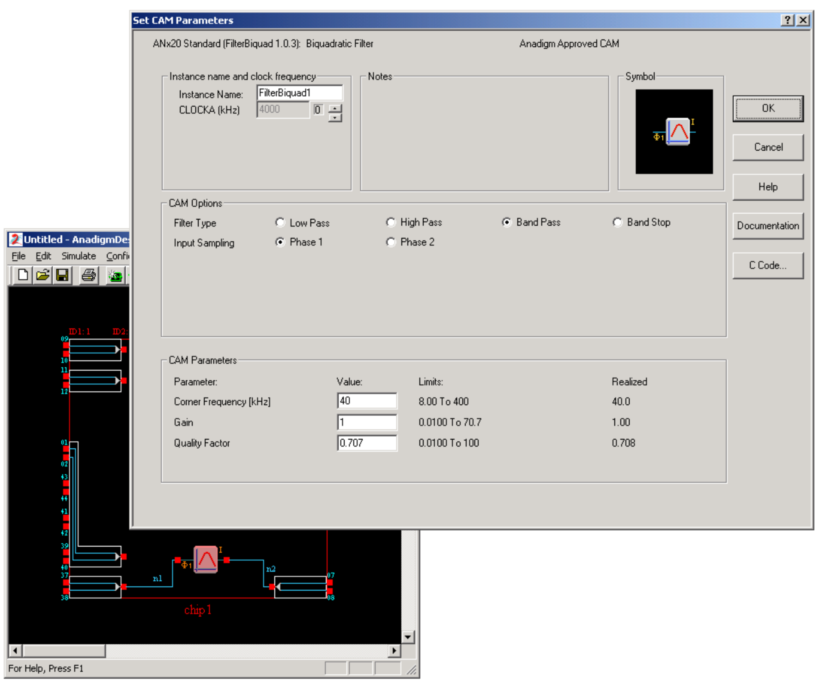


Figure 3 - FPAA Design tool and CAM Parameter setting.

2.2. Designing an FPAA Circuit

2.2.1. Initial and Single-state Design Built with CAMs

Two of the strengths of the FPAA design approach are the abstraction of design to a functional level, and the ease of integration with other system components. This facility is substantially provided via the use of circuit building-blocks called Configurable Analog Modules (CAMs). Examples of CAMs are filter stages, gain stages, summing/difference stages, voltage multiplication, phase/voltage comparators, rectifiers, oscillators, peak detectors and references. In all, some 50-60 such building blocks are available.

The FPAA also supports non-linear processing functions including modulators, arbitrary transfer functions and programmable soft-knee compression. This is illustrated further later in this paper.

Access to the CAM is via a high-level design tool. The CAMs are contained within a design library, and can be dragged-and-dropped onto a schematic 'canvas' within the design tool, on which the FPAA device(s) that will realize the final design are depicted (Figure 3). The designer sets various attributes for that CAM – e.g. filter corner frequency – via a parameter dialog box, and the system algorithmically calculates the 'best fit' component values and circuit structure to meet requirements (e.g. see CAM sub-circuit of Figure 11). Thus the designer need never be concerned with low-level circuit design.

The FPAA software assembles the configuration data set(s) for the entire design, and delivers it in a form ready for delivery from EEPROM or a host microprocessor.

2.2.2. Variable-State and Tunable Design

The FPAA can be re-configured in full or in part while it is running. This is a critical feature for analog signal processing, where the ability to tune or modify the circuits offers major advantage over classic descrete implementations. Once a design is assembled in the FPAA design tool, the tool provides the ability to extract instructions to change parameters of, or replace, any circuit feature under digital control.

The designer may wish to have alternative programming states presented as sets of raw configuration data, which is one approach supported by the tools.

Alternatively it may be desirable to generate a software-based Application Programming Interface (API) which allows control over the FPAA from system software. This API information is delivered in the form of C-code, automatically generated by the FPAA design tools to suit the specific design [5].

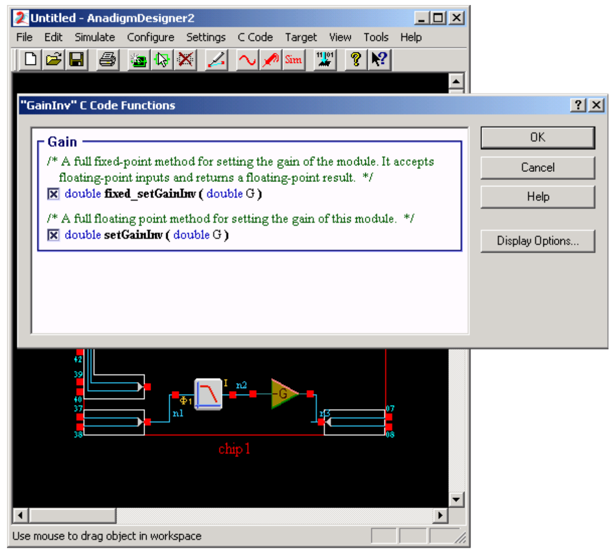


Figure 4 Constructing a custom API for an FPAA design

A simple dynamically re-configurable FPAA design might comprise a filter and a gain stage in which the gain is to be varied under processor control (see Figure 4). Associated with the gain stage is at least one software procedure available for calculating new circuit configurations from a high-level parameter – in this case gain, G. This can be done for each element in the circuit which is to be dynamically controlled..

The FPAA design tool then exports these algorithms in the form of an ANSI-C code library. This would then be ready to be incorporated into the software for the in-system micro-processor of Figure 2. In this way, simple function calls (e.g. "setGainInv()" to reset inverter gain in Figure 4) are all that are needed to reconfigure the FPAA from a system micro. The designer need not be concerned about memory maps, algorithm details, or bitstream segment construction – these are built into the automatically generated code.

3. AN FPAA-BASED SUBWOOFER CONDITIONER

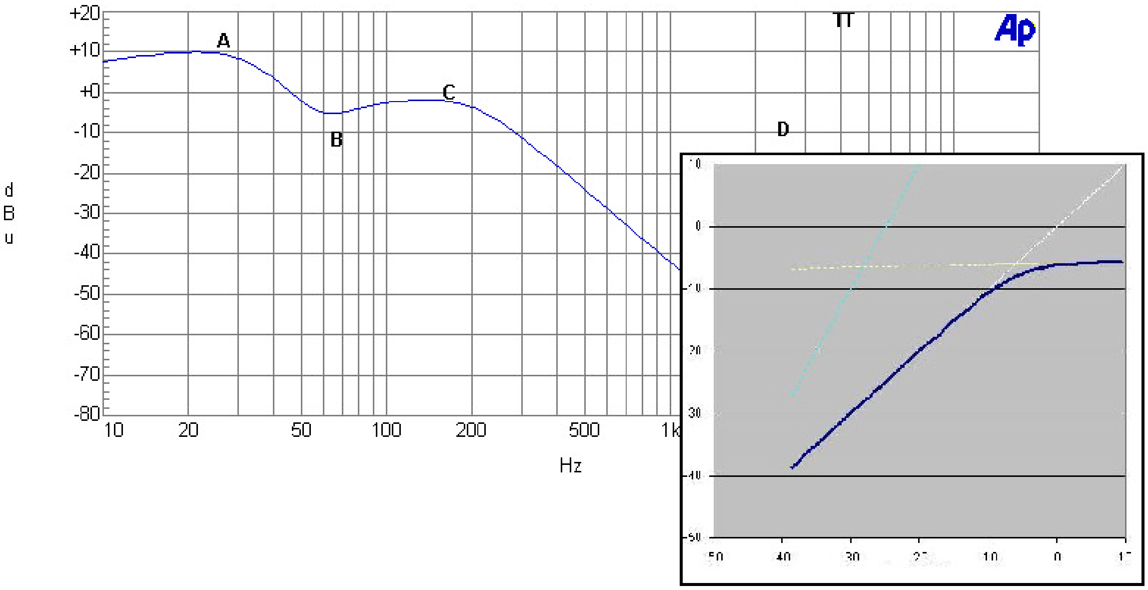


Figure 5 - Subwoofer conditioner response targets

3.1. System Objective & Design

The target system is a family of home entertainment or studio monitor subwoofers for which independent control over crossover and EQ characteristics is desired on a single reference design.

The functions supported by this Subwoofer Signal Conditioning circuit are (with reference to Figure 5):

- **Sub-sonic filter** to remove low frequency and DC components, with user-selectable corner frequency to the left of point 'A'.
- **Linkwitz Transform equalizer** circuit with user-programmable:
  - pole frequency and Q factor at point 'A'
  - zero frequency and Q factor at point 'B'
- **3rd order low-pass filter** with dynamically programmable cut-off frequency at point 'C'
- **Amplitude compression circuit** with user-specified compression characteristic ('D').

It is also a design objective that the complete system be supported by a single FPAA device. The resulting design is shown in Figure 6.

It can be seen that 5 CAMs are used to implement the four functions above – these are described fully in the following sections.

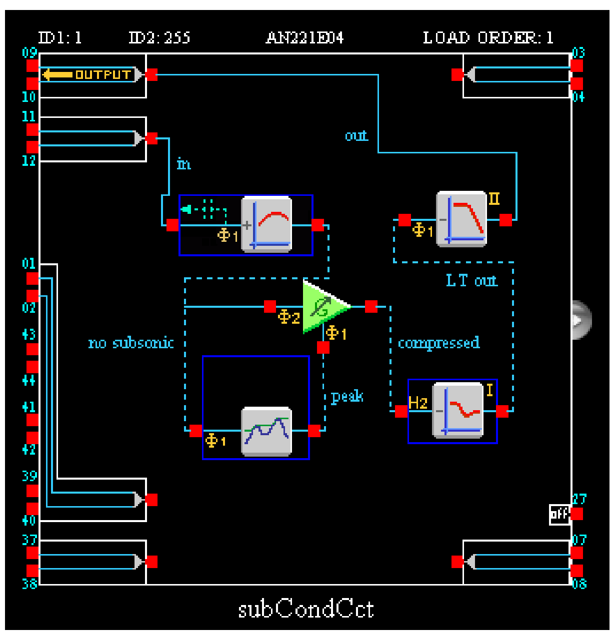


Figure 6 – Subwoofer conditioner FPAA design

3.1.1. Sub-Sonic Filter

The primary task of this block is the suppression of DC and very low frequency components, assuming that the input signal is not AC-coupled, it must be performed by the FPAA. Inputs can be differential or single-ended. Figure 7 shows a single-ended summing input, where left and right audio channels are combined.

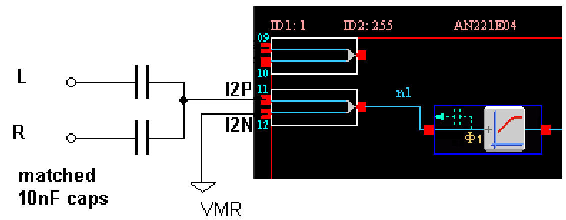


Figure 7 - Subsonic filter

The CAM used for this function is a dual bilinear filter, with two widely-spaced poles, one high-pass the other low-pass. The high-pass pole is extremely low frequency, and combines the external capacitor(s) C_{ext} with input capacitor C1 (see Figure 8).

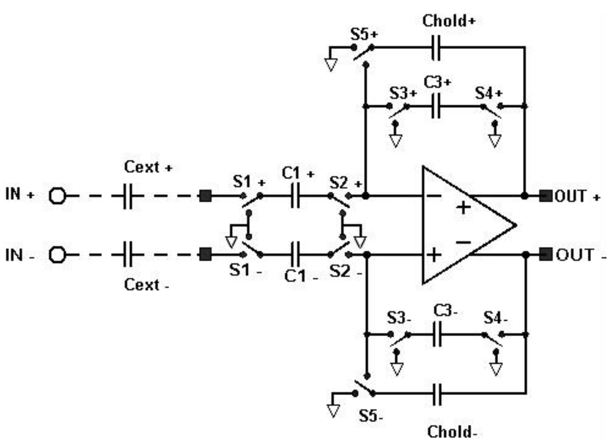


Figure 8 - Subsonic filter circuit schematic

If the recommended **10nF** capacitors are used, this CAM will deliver a 1st order subsonic filter with –3dB corner frequency of **11.6Hz.**

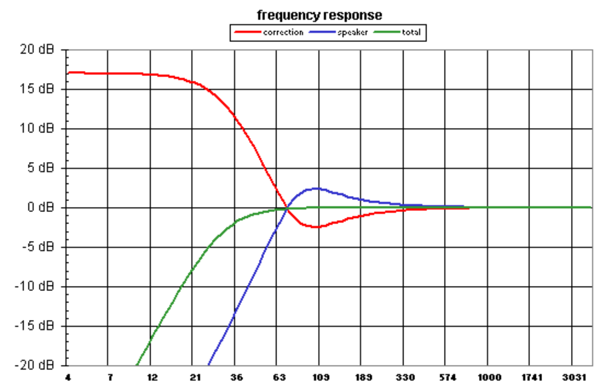
This CAM also performs two other functions:

Gain Prescaling: if the differential input signal level is other than 0dBu nominal (with **maximum** peak level of 8.75dBu – see Section 3.1.4) then this CAM should be used to pre-scale the gain accordingly. This is necessary for the optimal operation of the compressor.

Low-Pass Filter: this uses the second, higher frequency low-pass pole of the circuit. This filter is used in conjunction with the 2nd order output low-pass filter to create an overall 3rd order (18dB/8ve) low-pass function. See also Section 3.1.3.

3.1.2. Linkwitz Transform Equalizer

The Linkwitz transform circuit is a hugely flexible way to equalize the bottom end of a sealed loudspeaker enclosure. A speaker that is corrected using this method is has flat response from below resonance to the upper limit of the selected driver. The low frequency boost is determined by the parameters of the transform circuit. Should the enclosure size be too small and cause a peak in the response, this is also corrected with a notch – or 'zero' – in the transfer function (see Figure 9).



source: True Audio

Figure 9 - Linkwitz Transform response

This filter response is achieved using a biquadratic filter with independent control over frequency and Q for both the pole and the zero.

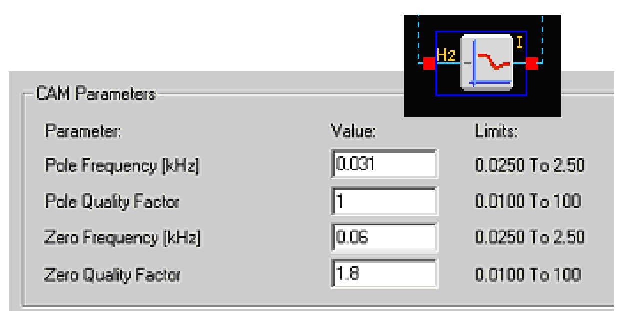


Figure 10 – Symbol and settings for biquad with independent Pole/Zero control

The design parameters for the subwoofer conditioner are shown in Figure 10, and the schematic in Figure 11.

The circuit is tuned to ensure that high-frequency gain is always unity, and so low-frequency gain boost can be varied by adjusting the pole frequency and Q factor.

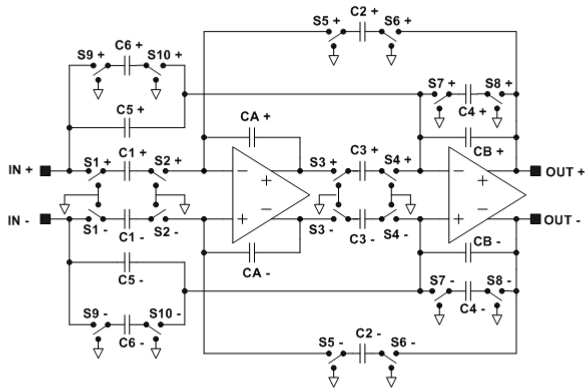


Figure 11 - Pole/Zero biquad schematic

3.1.3. 3rd Order Low-Pass Filter

This function is realized through combination of a standard low-pass biquadratic filter (Figure 12) with the 1st order filter implemented within the subsonic filter CAM (see Section 3.1.1).

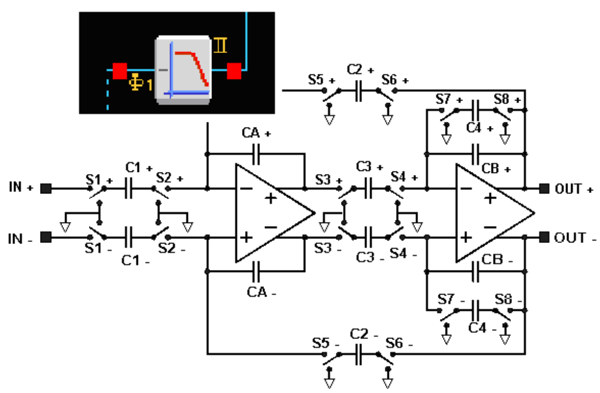


Figure 12 - 2nd order low-pass filter

By setting the corner frequencies of both to the same value, and setting the 'Q' factor of the biquad to 1.0, the combined result is a 3rd order Butterworth low-pass filter.

For real-time variable control over this filter, the corner frequencies of both blocks must be adjusted – see Section 4). If the clock frequencies remain unchanged, the resulting filter cut-off frequency can vary between 80Hz and 5kHz. This range can be further extended by changing clock frequencies.

3.1.4. Amplitude Compression Circuit

This circuit comprises a peak detector with independently programmable attack/decay and a variable gain stage whose gain is arbitrarily dependent on a control voltage (see Figure 13).

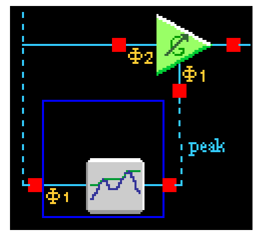


Figure 13 - Amplitude compressor circuit

By converting the amplitude of the input signal to a voltage – in this case using a peak detector – and then using that voltage to control the gain in the signal path in a pre-determined manner, a compressor (or soft limiter) can be implemented.

Alternative amplitude-sensing functions can be deployed, e.g. rectifying filters or RMS-to-DC converters.

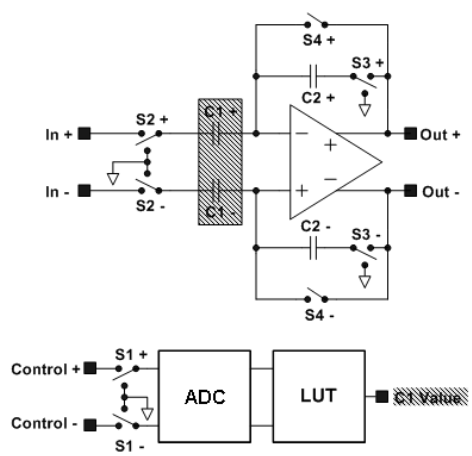


Figure 14 - Voltage controlled gain circuit

The circuit of the voltage-controlled gain stage is shown in Figure 14. The notable feature of this voltage-controlled-gain stage is that the control-voltage-to-gain characteristic is arbitrary. This is achieved through the use of a digital look-up table (LUT), built into the

FPAA. The forward signal path gain is variable by changing the value of the input capacitor C1. This capacitor is just one of the many dynamically re-configurable resources of the FPAA. The control voltage (in this case signal amplitude level) is converted to a digital code which is used as an address for the LUT. The data at that address is applied to C1 to set the gain of the analog signal path. The LUT is simply a memory block, which can contain any value for the capacitor. Hence any arbitrary amplitude-to-gain characteristic can be applied to limit, compress or expand the signal amplitude.

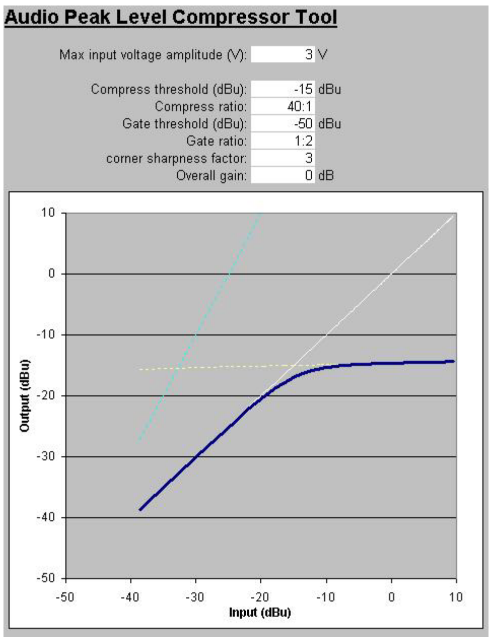


Figure 15 - Peak level compressor configuration tool

A simple graphical tool is used to generate the LUT data sets required to achieve different compression characteristics (Figure 15) and these are loaded into the voltage-controlled-gain CAM.

The resulting compression is very smooth, with the primary signal path being fully analog. Section 5 shows measured compression characteristics and signal FFT plots.The response times of the compressor are set by the attack/decay of the peak detector.

4. DYNAMIC CONTROL OF THE SUBWOOFER CONDITIONER

As described in Section 2.2.2, having designed an FPAA circuit, its real advantage can come to the fore if its re-programmability is exercised in the field. In the case of speaker systems, this could be to compensate for room resonances, to apply different compression chacacteristics dependent on the signal source, or simply to partner the subwoofer with different monitor speakers (needing a different crossover frequency).

Simple control systems could be written to achieve this to suit a very low-cost microcontroller, which could be housed within the speaker cabinet itself.

For demonstration purposes, a PC-based software controller was developed for this circuit. In this case the target processor is running the Windows® operating system, and the control interface is a graphical user interface (Figure 15).

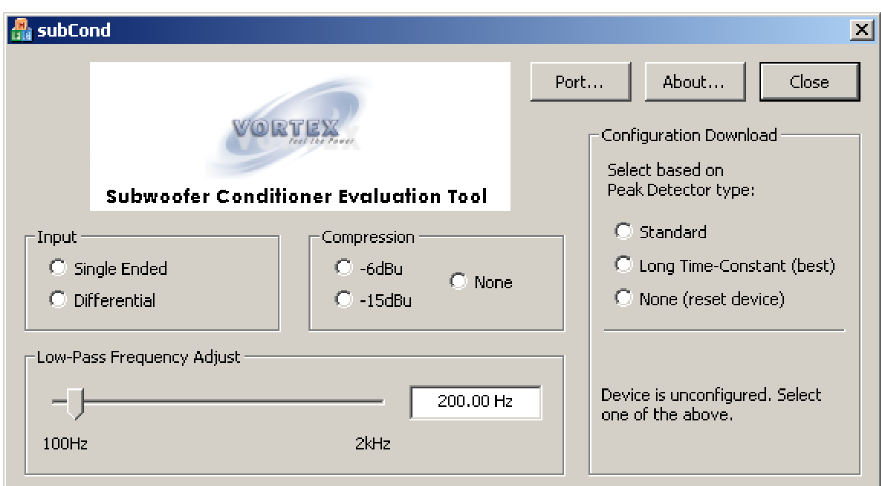


Figure 16 - Example PC-based control interface for real-time parameter adjustment

It can be seen that although the signal conditioner circuit is analog from front-to-back, sophisticated control can be applied real-time to the system. In this case there are controls for the crossover low-pass filter corner, and for applying or removing different compression curves. The 3rd order lowpass crossover characteristic combines the responses of two blocks, as described in Section 3.1.3. The slider control, therefore, calls API functions for both of these to deliver the correct response (see results in Section 5.2).

Of course, any of the functions described in 3.1 can be adjusted real-time by deploying the relevant automatically-generated API call.

5. SUBWOOFER CONDITIONER MEASUREMENTS

5.1. Linkwitz Transform Adjustment

The biquad filter with independent pole and zero control (implementing the Linkwitz Transform equalizer) is a very versatile building block. Figure 17 to Figure 19 illustrate various aspects of its parametric control.

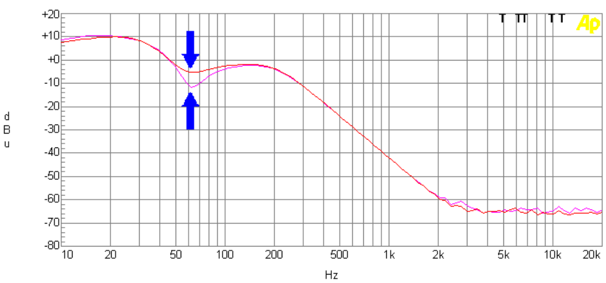


Figure 17 - Varying Zero 'Q'

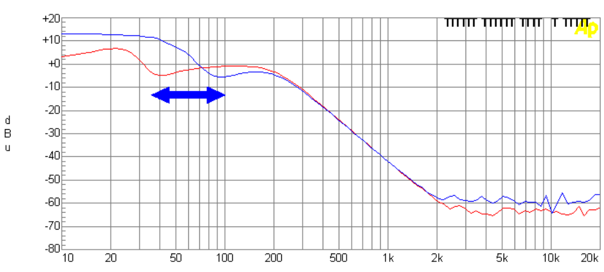


Figure 18 - Varying Zero frequency

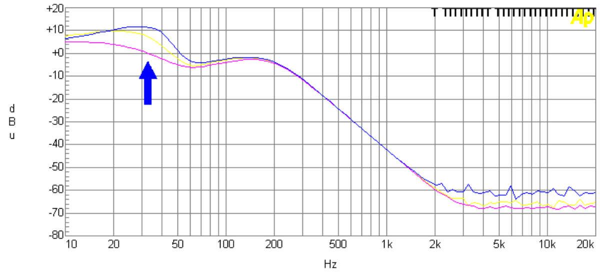


Figure 19 - Varying low frequency boost

5.2. Crossover Frequency Adjustment

Crossover frequency is variable between 80Hz and 5kHz for the clock frequencies deployed in the subwoofer conditioner design. Figure 20 illustrates curves where the crossover –3dB corner frequency is set to 100Hz, 200Hz and 400Hz respectively.

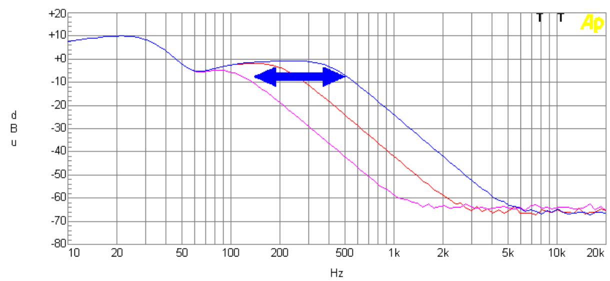


Figure 20 - Varying crossover frequency

5.3. Changing Compression Characteristic

The compressor was configured with three different characteristics, one having no compression (gain = 1.0 for all amplitudes).

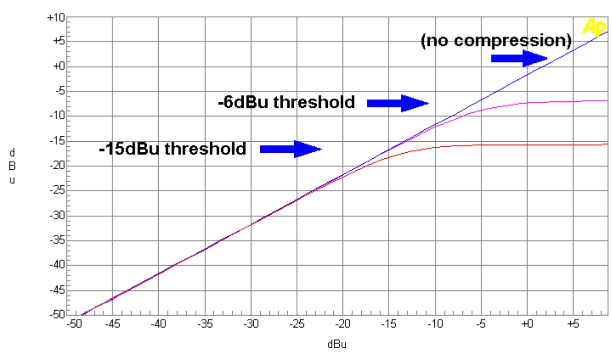


Figure 21 - Three compression curves

Other compression characteristics used a 'soft-knee' characteristic to a high, positive compression ratio (40:1), yielding a near-flat gain response for high amplitude gains. Two thresholds were selected for these gains: -6dBu and -15dBu.

Measurement curves were taken with a 200Hz sinusoidal signal, and so additional slight attenuation is visible due to the crossover filter.

5.4. Harmonic Distortion

Figure 22 shows an FFT of the complete subwoofer conditioner circuit with –6dB compressor active.

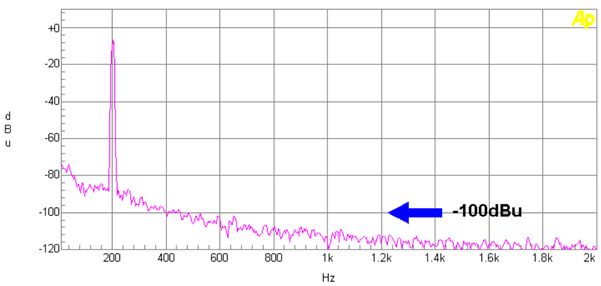


Figure 22 – FFT with –6dBu compression

It should be noted that any compressor which detects signal amplitude using RMS-to-DC, rectifying filters or peak detectors will inevitably inject some harmonic content into the signal path, irrespective of whether analog or digital techniques are used, and can be up to –40dB relative to the fundamental frequency. This can be minimized by selecting a large decay constant for the peak detector (or filter time constant), which allows the true system linearity to emerge.

In our example, the decay constant is a very practical 0.4V/ms and the spur-free dynamic range (dB to the highest harmonic peak) is better than 94dB, or 0.002%. Note that the fundamental in this case is suppressed to –6dB by the compressor.

6. CONCLUSIONS

Discrete Analog circuitry does not easily allow for digital programmability and dynamic control. DSP can fill this need, though the digital nature of the signal processing can add complexities and costs that can be unattractive.

The FPAA solution provides a cost-effective complement to the DSP which allows the designer to remain in the analog domain and yet still support designs which are highly configurable and adaptive under digital control. Thus the ability differentiate, "future-proof" and tune product by exploiting software-based configuration, and to maintain security of design by use of an integrated solution is maintained, whilst retaining true analog signaling.

The software design tools facilitate development of both the circuit and control interface. The Subwoofer conditioner design has been shown to perform well under test conditions.

7. REFERENCES

[1] "Performance of Enclosures for Low Resonance High Compliance Loudspeakers", J. Novak, IRE Transactions on Audio, Volume 7 Number 1 pp. 29-37; January 1959

[2] "A Three-Enclosure Loudspeaker System with Active Delay and Crossover", Parts 1, 2, 3, S. Linkwitz, Speaker Builder 2/80, 3/80, 4/80.

[3] "Digital Equalization of Room Acoustics", Mourjopoulos J. J. Audio Eng. Soc. Vol. 42, No. 11, November 1994

[4] "Multiple-Point Equalization in a Room Using Adaptive Digital Filters", Elliott SJ, Nelson PA, J. Audio Eng. Soc. Vol. 37, No. 11, November 1989

[5] "New FPAAs: Putting Analog Signal Processing Under Software Control", I Macbeth, International Signal Processing Conference 2003.

[6] "Designing a Linkwitz Transform Circuit", J. Murphy, True Audio Website