UNIVERSAL AUDIO SIGNAL PROCESSOR BOARD DEVELOPMENT

Kristóf VARGA*, Bertalan CSANÁDI**
*Óbuda University, Budapest, varga.kristof@t-online.hu
**Óbuda University, Budapest, csanadi.bertalan@kvk.uni-obuda.hu

***Abstract*— This paper shows the development of an audio signal processing device with audio inputs and outputs, modern digital interfaces, capability to use external memories, interconnections to a digital studio and capable of high modularity. With our device we are capable of realizing universal filters. We can overwrite our filter's multiplying constants, this way we can realize a new filter without the need of changing a device.**

I. INTRODUCTION

Audio signal processing is an always developing area. With the new processors and codecs coming available, we have the possibility to increase the effectiveness of our existing instruments or create new ones. Using new chips allow us to make a low price and more flexible panel. In our project we choose Microchip's new hardware elements to develop a new board for universal usage and laboratory measurement. Our goal was to create a board that can be used freely for audio signal processing, is easy to handle, compatible with other boards and could be built in a larger system.

II. PLANNING THE BOARD

In the first step of our development, we listed what functions we will need. The main solution was that the device will take place on 3 boards. The top board contains the Pic microprocessor[1] and the user interfaces. The middle board holds the codec IC and the analog connections. The bottom board, which is not presented in the paper, contains symmetric studio interfaces, if it is necessary *{Figure 1.}*.

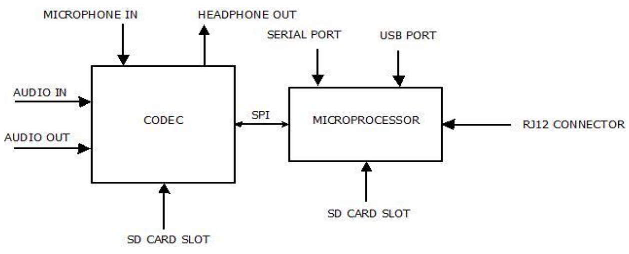

Figure 1: Logical diagram of the device

[1] Microchip Technology Inc. dsPIC33EPXXX(GP/MC/MU)806/810/814 Datasheet Available: http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en554301

III. THE MICROPROCESSOR BOARD

A. The dspic33ep256mu806

The heart of the microprocessor board is the DSPIC33EP256MU806 Pic microcontroller, from Microchip Technology Inc. The microprocessor has twenty-four analog channels, which makes it capable for the audio signal processing board, with the 70 MIPS speed. This is not a real DSP processor, it is more like a DSC (digital signal controller). The DSP engine consists of a high-speed 17 bit x 17 bit multiplier, a 40 bit barrel shifter and a 40 bit adder or subtracter. The pin remapping is greatly increasing our possibilities with making a highly universal panel

B. Connectors on the board

Making the programming and the handling easier, we designed the board to have as much interfaces accessible as possible. These interface connections (with the exception of the serial port) are on the microprocessor board. The mini-USB connector grants us easy access for our PC or with a bootloader we can use it for programming the device. The RJ12 connector allows us to connect with an ICD[2], for easy programming and debugging possibility. We placed an SD card slot on the board, which can function as memory or we can use it to load our programs from it *{Figure 2.}*.

C. Power supply and overpower indication

The power supply of the board is provided with a DC JACK connector, using a TS1117CW-3.3 IC which regulates the 3.3V for our boards from the 5V provided. With decoupling capacitors we reduce the noises and disturbances in our power supply. There are two leds placed to indicate if we have the necessary voltage levels. The overpower indication is realized using a zener diode circuit with a led to show us if the power level exceeds the allowed limit.

[2] MPLAB ICD In-Circuit Debugger , Available: http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en537580&redirects=icd3

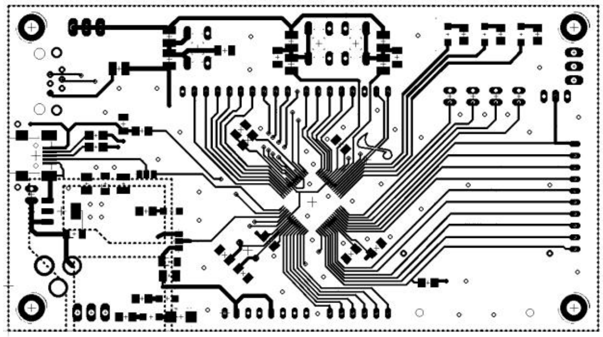

Figure 2: The top layer of the microprocessor board

IV. THE CODEC BOARD

A. The tlv320aic23b

For the codec board, we used the TLV320AIC23B[3] codec, from Texas Instruments Inc. This is a codec from the industrial standard series. The TLV320AIC23B use multibit sigma-delta technology with integrated oversampling digital interpolation filters. It supports 8-96 kHz sampling rate with 16, 20, 24 and 32 bit data-transfer word length. The ADC module can provide 90 dBA signal-to-noise (SNR) ratio with 96 kHz sampling rate. The DAC module can provide 100 dBA SNR ratio at audio sampling rates up to 96 kHz, while consuming less than 23 mW during playback only. The codec has an integrated and programmable microphone amplifier and integrated anti-aliasing filters built in, allowing us to save more space on our board, while making it cost-effective.

B. Connectors on the codec board

Near the codec chip we have the analog input and output connectors, with the necessary analog interfaces. We have two RCA connectors for the left and right analog channels. With 2.5 mm JACK connectors we can connect our microphone or headphone to the board. Using the 4N33 optocoupler we designed a circuit for receiving world clock signal which allows us to connect the board in a studio system. The serial port connection is also on this board, the serial data is lead up to the microprocessor holding board. Using the MAX3232 IC we are translating the 3.3V logic level to the RS232 level and reverse for the necessary communication. There is a second SD card slot placed on this board, which is connected to the microprocessor *{Figure 2.}*.

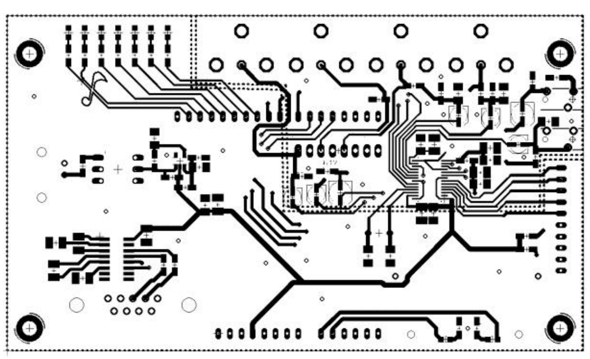

Figure 3: The top layer of the codec board

[3] [6.] TLV230AIC23B datasheet, Texas Insturments Inc. (2004) Available: http://www.ti.com/lit/ds/symlink/tlv320aic23b.pdf

V. UNIVERSAL RECONFIGURABLE, BIQUAD FILTER STRUCTURE

In the software we use an array for storing the multiplying constants. In every signal processing cycle we use a subroutine that receives this array, with the multiplying constants and does the processing from the received data. This method allows us, to overwrite a value in the array, simply recreating our filter. Using serial port connected to our computer is easy to send the new values to our device. We can use our keyboard if we want to add new values manually or (this method needs more programming) run a Matlab® [4]script for example, which calculate our new values, convert them into the form, that the microprocessor expects and sending it through the serial port. If we plan a filter using the Filter Design and Analysis Tool[5] (FDA tool) we can just export the transfer function for our program, updating our filter as fast as it possible. This only takes the calculation and sending time, allowing us to change filters, unnoticeable by human hearing.

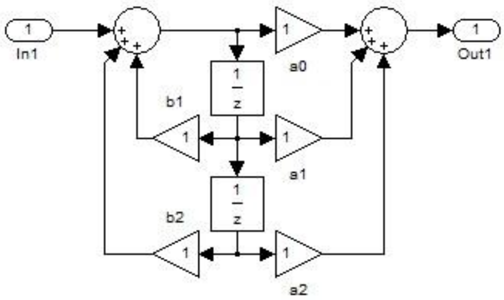

Figure 4: The one stage BiQuad structure

$$H(z) = \prod_{i=1}^{n} \frac{a_0(i)+a_1(i)Z^{-1}+a_2(i)Z^{-2}}{1-b_1(i)Z^{-1}-b_2(i)Z^{-2}} \qquad \{5.1\},$$

Most of the times we used a BiQuad structure for realizing filters, because this is the most easy, still effective structure. It is possible to realize a high order filter from only second order BiQuad structures, connecting these filters into a cascade.

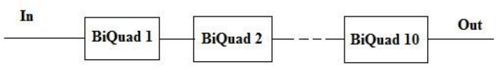

Figure 5: The ten stage biquad structure, using second order filters

[4] Matlab homepage Available: http://www.mathworks.com/products/matlab/

[5] Matlab, FDA tool homepage Available: http://www.mathworks.com/products/signal/examples.html?file=/products/demos/shipping/signal/introfdatooldemo.html

VI. A BIQUAD EXAMPLE

In this example I will show an example of realizing an eighth order filter. We start with the well-known biquad structure. In the Matlab's FDA tool we can design our filter. We created a second order lowpass filter. The filter has the following transfer function {6.1.}.

$$H(z) = \frac{1 + Z^{-1} - 0{,}0039Z^{-2}}{1 - 0{,}0625\, Z^{-1} + 0{,}0875\, Z^{-2}} \qquad \{6.1.\}$$

To attain an eighth order filter from the designed second order we need to connect four of it in a cascade. Using the formula {5.1} we can calculate the transfer function of the final eighth order filter. Since using the same filter four times, it is less complicated to calculate this. The final transfer function is in the {6.2.} formula.

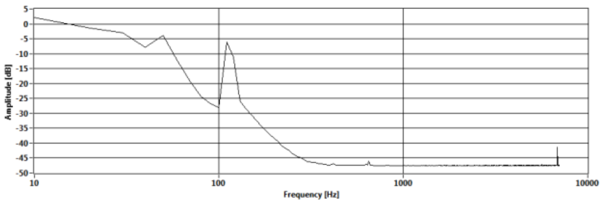

Figure 7: The measured transfer function of the eighth order biquad structure

After listening to the realized filter the difference between the second order and the eighth order comes clear. We modulated music with the realized filter, which fit our expectations. After listening to the second order filter we felt like listening the music through water. With the eighth order structure the situation was completely different. The middle and high frequencies disappeared completely, only the bass guitar and the bass drum left from the whole band.

$$\frac{1 + 4\, z^{-1} + 5{,}98\, z^{-2} + 3{,}95\, z^{-3} + 0{,}95\, z^{-4} - 0{,}015\, z^{-5} + 9{,}1e^{-5}\, z^{-6} - 2{,}37e^{-7}\, z^{-7} + 2{,}31e^{-10}\, z^{-8}}{1 - 0{,}25\, z^{-1} + 3{,}52\, z^{-2} - 0{,}65\, z^{-3} + 4{,}63\, z^{-4} - 0{,}57\, z^{-5} + 2{,}69\, z^{-6} - 0{,}16\, z^{-7} + 0{,}58\, z^{-8}} \qquad \{6.2.\}$$

After calculating the transfer functions we can draw the Bode diagrams using Matlab. As we expect there will be only difference between the amplification values, in other words, the eighth order filter has better selectivity.

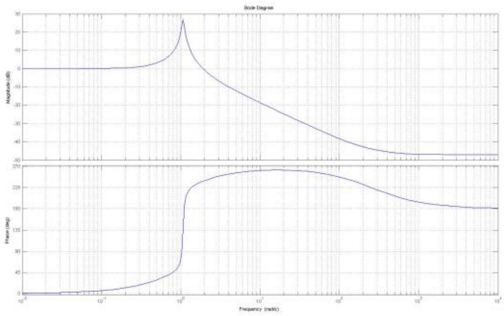

Figure 6: The Bode diagram of the second order biquad structure

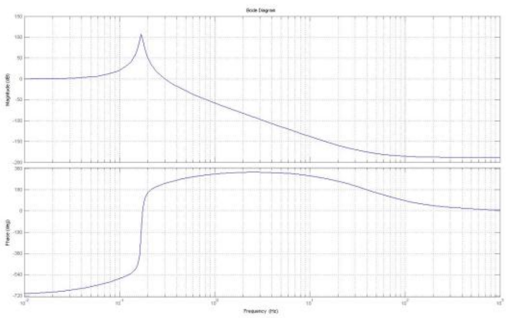

Figure 7: The Bode diagram of the eighth order biquad structure

After realizing the structure we made a measurement to see the real transfer function of our system.

This filter is an appropriate way to realize the low pass part of a crossover system. The same method can be applied to create the other parts of the system, with great selectivity.

VII. SUMMARY

Making the device universal and easy to handle, while keeping high performance and cost-efficiency is a hard task. The new chipsets allows this possibility. This panel can be built for about 100EUR. With the DSPIC33EP256MU806 and the pin remapping, the device can be realized to be really flexible, since we determine the pins functions. Connecting the two boards with the Arduino board connectors allows us to switch the microprocessor board for our Arduino device. With this opportunity we can work with a system that we already have and already know how to use. We can easily program the device with the RJ12, mini-USB and the serial port accessible. Having a device with a codec and microprocessor not only allows us to realize DSP functions, but to make a recorder device, an effect processing unit or realize a crossover system. Using biquad structures for our filters allows us to make high-order cascade structures for high selectivity without hard mathematical calculations. Realizing a universal biquad structure allows us to change our filter's transfer function whenever we need. Using our keyboard or running a Matlab code makes it very easy to overwrite the existing filter, so we don't have to change devices.

The realized eighth order filter shown in the example is a suitable way to realize a studio application like a crossover system. With little mathematical calculations we can create low order filters and connect them in a cascade. This way we save ourselves from the time long calculations, still can achieve filters with high selectivity.

REFERENCES

[1] Gyányi Sándor, Wührl Tibor, *Digitális jelfeldolgozó hálózatok gyakorlati megvalósítása*, 2009, BMF-KVK 2072, Budapest

[2] F. Shipley, F. Arnold, M. Tsecouros, R. King (2002, february) *Evaluation Platform for the TLV320 AIC23 Stereo Audio CODEC and TLV320DAC23 Stereo DAC*. Available: http://www.ti.com/lit/ug/sleu016/sleu016.pdf

[3] Horváth Elek, *Méréstechnika* változatlan utánnyomás 2010., Budapest

[4] Eagle user's manual 2010., Available: http://www.cadsoftusa.com/

[5] MAX3232 datasheet, Texas Instruments Inc (2000 January) Available: http://www.ti.com/lit/ds/symlink/max3232.pdf

[6] TLV230AIC23B datasheet, Texas Insturments Inc. (2004) Available: http://www.ti.com/lit/ds/symlink/tlv320aic23b.pdf

[7] SD Socket datasheet, Attend Technology Inc., Available: http://www.hestore.hu/files/mcc-sd_2.pdf

[8] dsPIC33EPXXX(GP/MC/MU)806/810/814 Datasheet, Microchip Technology Inc., Available: http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en554301 (2012. 03. 21.)

[9.] MPLAB ICD In-Circuit Debugger , Available: http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en537580&redirects=icd3

[9] R. Lyons, *Using Mason's Rule to Analyze DSP Networks,* Available: http://www.dsprelated.com/showarticle/76.php

[10] Tae Hong Park, *Introduction to Digital Signal processing Computer Musically Speaking,* 2010, Singapore

[11] J. Goette , *An IIR-Filter Example: A Butterworth Filter*, 2012, Biel

[12] Mitra, Sanjit K., *Digital Signal Processing: A Computer Based Approach, SecondEdition*, McGraw Hill,2001.

[13] Julios O. Smith *Physical Audio Signal Processing: For Virtual Musical Instrumentsand Audio Effects* August 2007 Edition, 2007

[14] D. Schlichthärle, *Digital Filters*, Basics and Design second edition, 2011

[15] Dr. Simán István, *Digitális Jelfeldolgozás*, 2008, Budapest

[16] Dr. Simonyi Ernő, *Digitális Szűrők,* 1984, Budapest