EECS 452 – Winter 2003
Project Plan – A^2REq

Due Feb 19th

Name of Project: Adaptive Audio Room Equalizer (A^2REq) [*Preliminary*]

Team Members:

Bill Stewart wbs (at) umich (dot) edu
Eric Hatty ehatty (at) umich (dot) edu
James Glettler jglettle (at) umich (dot) edu
John Bemesderfer jbemes (at) umich (dot) edu
Landry Tientcheu ltientch (at) umich (dot) edu
Oz Pearlman opearlma (at) umich (dot) edu

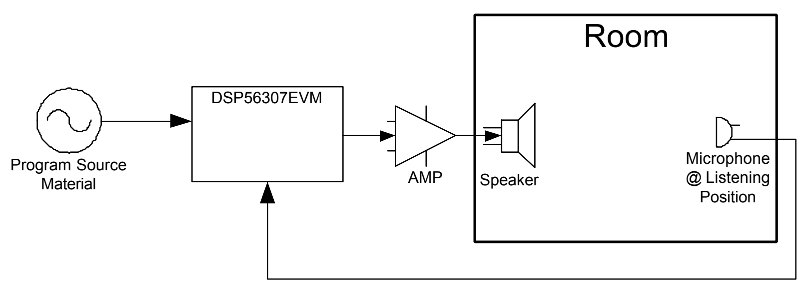
Team Email: 452eq (at) umich (dot) edu

Brief Description of project:

This project aims to develop an audio equalizer that can automatically adapt its filter coefficients in order to flatten the frequency response of a room and associated sound reproduction system at a listening position. The end result of this project has multiple uses in the professional audio field and some products like it are already on the market at prices varying between hundreds and thousands of dollars.

The basic premise is to use a filter implemented on a digital signal processor to do the frequency equalization in real-time. The equalization is needed to correct for non-uniform magnitude (and possibly phase) response of the amplifier, loudspeakers and acoustic environment or room preceding the listener. The other function of the DSP is to inject a test signal into the output and measure the response from the microphone. It uses this measurement to calculate updated filter coefficients for the equalization filter.

This is a difficult project because there are multiple ways to implement the filtering algorithm, the measurement algorithm, and the control backend. Each way offers a different set of constraints and benefits which will need to be researched and simulated. The final project will be implemented on the DSP hardware with some other support hardware. A block diagram is shown below.



Goal to be reached by the end of the semester:

Have working Matlab prototype/simulation and working DSP implementation of:

- Real-time frequency equalization on one channel (monaural) over limited audio bandwidth
- Frequency response measurement
- Use of measurement to generate new equalization coefficients (possibly off-board)
- Insertion of new coefficients into equalizer.

Extensions (if time allows):

- Make frequency response measurement and update equalization coefficients in the background of the equalization functionality while program material is playing. This requires that method for frequency response measurement is inaudible.
- Implement a graphical user interface (GUI) for the project over RS232 serial using Matlab (or other software such as LabView/LabWindows)
- Use the GUI to display the pre-corrected and post-corrected room response curves
- Expand system to support two channels (stereo)
- Implement stand-alone operation using Flash-memory.

Required resources:

DSP: One(1) Motorola **DSP56307EVM**

We desire the 307 because of its advanced feature set. We need the to have the capability of higher processor speeds in order to perform a great deal of operations but also to reduce the possibility of timing conflicts. The 307 also has a good deal more memory than the 303 which should allow for larger and more precise filters. Two major functions the 307 has we wish to use is the Enhanced Filter Co-processor (EFCOP) for faster computation and the RS-232 serial port for computer control or possibly GUI functions. In the end we may not need all the extras that the 307 has to offer but we feel that it is better to go with a bigger system now than to risk getting stuck later.

Hardware:

- Motorola DSP56307EVM (provided)
- A number of different loudspeakers various quality (owned by team members)
- An audio power amplifier (owned by team members)
- A computer with at least two RS-232 ports (provided)
- A high quality omni-directional microphone with known and good frequency response. (*must be purchased*)

A high quality microphone is a must for this project. By using a microphone with a known and calibrated frequency response over the entire audio bandwidth (using the current model of a swept sine or broadband measurements) a separate filter does not need to be implemented to correct for the microphone's own frequency response flaws. The cost of such a microphone would most likely lie in the \$50 to \$100 range. The hardware exists in lab if a microphone preamplifier is needed. One option is to get a Sound Pressure Level meter that has signal output. This would be more multipurpose and perhaps be useful for future EECS 452 project groups.

Software:

The software requirements are minimal and include Debug-56K, a terminal emulator, a text editor, and Matlab. Another interface software such as LabView may be used as well.

Other:

We will also try to use the anechoic chamber in the RadLab to make measurements, so the DSP hardware must be mobile in some way.