Embedded Digital Audio Amplification Device

System Requirements Specification

ECSE421: Embedded Systems, Winter 2010

Team 1: Walid Hussein (260215780), Samuel Cormier-Iijima (260174995), Amin Mirzaee (260209556), Benjamin Nahill (260174420), Jules Farajallah (260188388), Bassel Khaddam (260215343), Christopher Bedirian (260260217), Dong Kwak (260244875)

2/9/2010

Introduction

Upon completion of this project, we will have developed an audio amplification system with an interactive user interface that provides the ability to drive an external audio output device. The complete system will consist of both software and hardware components that work together to produce the desired output. The system will have a variety of features, enabling the user to control various aspects of the output such as volume and equalization settings. Also, the system must provide an interactive menu system which allows the user to perform various tasks and edit settings. An onboard microcontroller will be programmed to control various settings, and communicate with the other hardware devices. The system will also incorporate a high power digitally controlled amplifier which to process the analog signals.

The primary limitations which the group will face while designing and developing the board are due to time and budgetary constraints. Nevertheless, we plan to create a full embedded system which will be able to perform all of the desired tasks in within the implicit deadlines imposed by the interactive nature of the control system.

Functional Requirements

1. *The system must be able to amplify an input stereo audio signal.*

An audio amplifier is a device which amplifies low-power audio signals to a level suitable for driving loudspeakers or headphones. This device is usually the last stage in an audio processing pipeline, and its input will come either from an electric musical instrument or an effect filter.

The system will have standard jacks for an input audio signal and the amplified output signal, which will be either 3.5mm or ¼" TRS connectors. It must be able to sample the stereo input signal at 48kHz, and the output must be able to provide up to 30W of power to the speakers. As an additional feature, the system may provide a digital output for further digital audio processing.

2. *The system must provide an LCD display showing appropriate feedback and menu options to the user.*

LCDs (Liquid Crystal Displays) are thin flat panels used for displaying graphical information such as text and images. LCDs are lightweight, portable, and have low electrical power consumption, making them ideal for use in embedded systems. The amplifier device will use a 16x2 backlit character-based LCD display. While running, the LCD will by default display a spectrum analyzer showing the power spectrum of the audio signal. This will switch to displaying menu options and other configuration parameters when the user chooses to see them.

3. *The system must provide buttons allowing the user to navigate through the menus.*

The system must provide 3 buttons (which may be capacitative), each providing a specific context-dependent function. One of these buttons will provide a quick way of muting or unmuting the output audio. The mute button will also act as a power button: if the system is muted for a certain amount of time, it should switch to a low-power mode which can be left by pressing the mute button again. The second button will be the menu button, which when pressed will enter the main menu. In submenus, the button will also function as an accept button to confirm changes to system settings. The third button will be a cancel button, letting the user exit from submenus.

The amplifier device will also have a continuous knob, which will let the user select different menu items and change parameter values inside submenus.

4. *The system must allow the user to change the output volume using the continuous knob.*

When the system is not displaying a menu, the continuous knob will allow the user to change the output level of the amplifier. Turning it clockwise or counterclockwise will respectively increase or decrease the volume, and the current volume will be displayed on the LCD to provide feedback. The amplifier will also implement soft-ramp and zero-crossing to detect abrupt changes and to provide noise-free level transitions.

5. *The system must be able to save and restore user settings to and from flash memory.*

The system must provide for a mechanism through which all settings are stored, so that restoring a specific set of settings after an interruption in the power supply (both planned and unplanned) is made possible. The system must provide at least three save slots which the user can use to save the current settings. The menu system must provide an option that the user can use to store the current set of settings into one of these preset slots, as well as another option which reloads all the settings stored in the specified slot. Also, as each setting is changed it must be written back to an additional slot, so that the settings can be easily recovered if the system is turned off. When the system is powered on, the settings stored in this slot must be automatically reloaded. All of these save slots must correspond to a location on non-volatile flash memory.

6. *The system must provide an equalizer (EQ) which can be configured through the menu.*

In audio engineering, equalization is the process of enhancing sound by altering the attenuation or boost of different frequency bands. Equalizers can help reduce excessive resonance, noise, or feedback, as well as enhance tones and compensate for room acoustics. A parametric EQ is a type of EQ which applies a band-pass filter to the audio signal and allows the user to control the filter's amplitude, center frequency, and bandwidth (as opposed to a graphic EQ, which allows the user to control the frequency response at specific frequencies.)

The speaker system will provide a menu option letting the user control the EQ being applied to the audio signal, as well as disabling it completely (bypass). In this menu, the user will be presented with either a graphical or parametric EQ and will be able change each of the EQ parameters by using the knob. The digital amplifier IC that we are using supports up to 5 parametric EQs which are specified with the coefficients for a bi-quadratic filter, and these will need to be calculated from the user's settings.

7. *The system must allow the user to enable or disable Adaptive Loudness Compensation.*

Adaptive Loudness Compensation allows the volume of the output to be normalized to a standard level. A time-averaged power is computed on the input and the output is scaled to improve audibility of low-level audio. All time-varying gain adjustments must be tapered to eliminate distortion in transitions. During a movie, for example, this would allow quiet speech to be clearly audible without increasing the maximum volume of the system. This is a useful feature in tight living quarters, where it is preferable to reduce volume without sacrificing audibility. The system menu will provide a way of toggling Adaptive Loudness Compensation by pressing the menu button while it is selected.

8. *When idle, the system must display a spectrum analyzer on the LCD.*

A spectrum analyzer will be provided as a novelty on the LCD, displaying 16 frequency components of the output signal. Using a short-time Fourier transform on the output, the system will compute approximate bin values for each component, displaying them with custom characters on the 16-character-wide LCD. As there are two channels of audio present, the system will mix the stereo channels into a single mono channel before computing. To compute 16 bins, we require a length-16 real Goertzel DFT, which is equivalent to only an 8-bin complex DFT due to the conjugate symmetry of a real DFT. The LCD will display the magnitude of each of these bins to represent the high, mid, and low frequency components of the filtered audio signal, and will be updated around 10 times per second.

9. *(Error condition) The system must detect overheating and prevent hardware damage.*

High-power amplifier circuits have a tendency to overheat, especially if operating in an environment with poor air circulation. As such, the system must take certain measures to ensure that the hardware components do not suffer any damage in the event that hardware temperatures go above a certain threshold (to be determined based on system design). The system must incorporate a temperature sensor, with which the microprocessor can interact (either via interrupts or polling). When the temperature is determined to be too high, the system must act immediately to protect damage, by temporarily lowering the amplification to a safe level (to be determined based on system design). The system must also provide feedback to the user via the onboard LCD display indicating that an error has occurred, and the amplification levels will return to normal once the temperature is normalized.

10. *(Error condition) The system will detect and appropriately respond to clipping at the output.*

Clipping is a (usually) undesirable effect which occurs when an audio signal's peak levels exceed a maximum threshold and the amplifier attempts to deliver an output voltage beyond its maximum power capability. This leads to distortion and noise (although it can also be used for artistic effect with e.g. distortion guitars.)

The system will consider clipping to be an error condition and handle it appropriately. The digital amplifier IC provides a peak signal limiter which detects when the output level exceeds a programmable threshold. If this happens, the system will lower the volume at a user-configurable attack rate until it drops below the threshold, at which point the volume will gradually return to its original level at a configurable release rate.

Technical Requirements

Hardware

The system will consist of a Microchip PIC32MX695F512H 32-bit MIPS4K microcontroller, providing 128K of RAM and 512K of Flash programming memory (see Table 1 for additional details). Timing for the microcontroller will be provided by an 8MHz external crystal (boosted to an 80MHz internal clock via a phase-locked-loop system) for added thermal stability and protection from jitter in the varying temperature conditions of a high-power amplifier. Also, a Cirrus CS4525 class-D (digital) audio amplifier IC will be used. This amplifier will communicate with the microcontroller through the I2S protocol, and will require a 24.576MHz crystal to provide a reference for its analog conversion. These components will be placed on a custom-printed circuit board. The board will have additional pins to allow the microcontroller to communicate with the 16x2 LCD display through the HD44780 protocol. The buttons, continuous knob, and status LEDs will be connected to the extra I/O pins of the microcontroller.

Family	PIC32MX6xx
Max Speed MHz	80
Program Memory Size (KB)	512
RAM (KB)	128
DMA Channels	8
SPI	3
I2C Compatible	4
A/D channels	16
Max A/D Sample Rate	1MHz
Input Capture	5
Output Compare/Std. PWM	5
16-bit Digital Timers	5
Parallel Port	PMP
Comparators	2
Internal Oscillator	8 MHz, 32 kHz
I/O Pins	53
Pin Count	64

Table 1: Overview of the PIC32MX6xx series microprocessor

Software

The system will mainly be written in C (possibly assembly for system initialization) and compiled using the MPLAB C32 compiler (based off of GCC). This compiler is cross-platform and can be used from either Linux or Windows. C is an appropriate choice for the system due to being both flexible and efficient. The microcontroller will be programmed using the PICKit2 USB debugger, which will connect to pin headers on the circuit board and is also accessible from both Linux and Windows.

An RTOS (such as FreeRTOS) will be used to manage tasks and resources on the microcontroller, improving the reliability and organization of the system.