mindTunes
W4840 - Project Description

Jonathan Chen jtc2119@columbia.edu UNI:jtc2119
Po-Han Huang ph2252@columbia.edu UNI:ph2252
Michael Kempf mjk2154@columbia.edu UNI:mjk2154
Yen-Liang Tung yt2230@columbia.edu UNI:yt2230
Christos Vezyrtzis chris@cisl.columbia.edu UNI:cv2176

Abstract—This project involves the combination of software and hardware design techniques for the implementation of a voice recording system. The particular system will involve both the interface to record voice clips and to store them in MP3 format in an external storage (USB flash drive). Browsing through and playing existing voice clips will also be a feature of the designed system. In order, therefore, to organize the implementation of this project, we provide an analytic description of its functions and building blocks, as well as a set of milestones which will serve as a guideline for the design.

1 Introduction

The purpose of the system to be designed is to serve as a complete voice recording system, exploiting the existing Altera DE2 FPGA boar's characteristics. The system will offer the follwoing capabilities:

- 16-bit quality voice recording
- Storing of recorded voice cpips to an external USB flash drive in an MP3 format
- Browsing the contents of the external storing medium and playback of the existing voice clips

The choice of the MP3 format offers the flexibility of a wolrdwide accepted standard, as well as a significant reduction in storage space. The loss of quality due to this compression is insignificant, considering the purpose of use that the system is intended for, i.e. voice recording. The latter will also enable us to occupy the least amount of system (Altera DE2 board) resources, since the sampling rate and processing required can be minimized.

The choise of an external USB flash drive will offer the possibility of a large storing space, in the **G**igabyte **B**yte (**GB**) region, which can extend the storing capacity to hundreds of hours of voice clips. Futhermore, the USB flash drive storage has a well known interface, which can be implemented through hardware.

The authors are with Columbia University, New York, NY. This project is associated with Embedded Systems Design 4840.

2 Floorplan of the System

The block diagram of the system is illustrated in Figure 1. The nature of each entity (i.e. if its implementation is through hardware or software) is also mentioned Figure 1 and in Table 2.

Required Block	Implementation
ADC/DAC	Hardware
USB Flash Drive and Interface	Hardware
CPU and OS	Software
MP3 Encoder and Decoder	Software
Memory Buffer	Hardware
Speaker, Microphone and Monitor	Hardware

TABLE 1
Required Blocks for mindTunes

We also note the interactions between these building blocks, which denote their way of cooperation for the operation of this system.

3 Analytic Description of the System Building Blocks

We now proceed by analytically describing each building block and its function, along which some key parameters through which it influences the system's performance.

3.1 ADC/DAC

In this voice recording and playback scheme, the ADC and DAC (namely **A**nalog to **D**igital **C**onverter and **D**igital to **A**nalog **C**onverter respectively) are the interfaces of the system's communication with the "outside world". They provide the conversion of the analog inputs to the system (voice) to a digital format (word) so that they can be processed (ADC) and the back-conversion of the words to a clean (in terms of its noise performance) analog signal to be used for playback.

The Altera DE2 board offers a top-level quality ADC and DAC (combined they are named as **"Codec"**), which can be used in numerous (software controlled) settings, depending on the application in hand. These settings include analog and digital pre-and-pro processing filters, along with the sampling rates and the analysis (number of bits) which the ADC and DAC encorporate. These parameters are involved in trade-offs that characterize the system's performance. As a simple example, we refer to the obvious trade-off between speed (and thus power) and audio quality (usually characterized by SQNR or SNR[1]), according to the number of used bits, more bits will lead to more use of resources and power, along with a degradation in speed (due to processing), but improvement in audio quality[2].

The use of the system for voice processing and storing (with a large tolerance in quality degradation and main concern toward storage) indicates the obvious choise for the Mixed Signal (ADC/DAC) blocks: they should be operated in minimum sampling rate and analysis, which is $8kHz$ and 16 bits respectively. A sampling rate of $8kHz$ is enough to handle a voice signal, which is band-limited to less than $4kHz$ (thus the Nyquist criterion is satisfied) and a 16-bit analysis is more than satisfactory for this purpose. This choice will be beneficiary (as mentioned before) in terms of occupied system resources, speed and power consumption and will allow the use of maximum resources in the encoding process (which is clearly more demanding).

3.2 USB Flash Drive and Interface

An external Universal Serial Bus (USB) flash drive will be used as the storage medium for recorded voice clip files. The flash memory will also act as the file repository for voice clip playback. It will have a capacity in the GB region at low cost. The DE2 board provides both USB host and device interfaces using the Philips ISP1362 single-chip USB controller. Both host and device controllers comply with the USB Specification Rev. 2.0. This supports data transfer at full-speed (12 Mb/s) and low-speed (1.5 Mb/s). Typically, the challenge of implementing a USB component is the requirement to design a software driver. Fortunately, uClinux provides a USB driver that we will utilize for writing and reading MP3 formatted voice clips to the USB flash memory.

3.3 MP3 Encoder-Decoder

The large use of MP3-formatted clips worldwide, as long as the smaller space that they occupy, makes MP3 encoding and decoding an attractive choice for this voice-recording system. However, this encoding and decoding scheme requires some system resources both for computation (manipulation of the raw audio samples) and the temporary storage of the intermediate data. This dictates the use of a Buffer Block (see corresponding section) as an intermediate storage, which needs to be "on-board" to offer the advantage of large speed and easy access interface.

As far as the coding process is concerned, the increased complexity of the procedure indicates software encoding as the optimal method. The wide availability of (open-source) existing libraries will enable us to easily use them through an operating system that we can install on the Altera DE2 board (see next section), which will result in minimal extra-system-resource occupation.

The open-source libraries will handle both the encoding and the decoding of MP3 tracks, which leads to the re-use of the existing hardware (buffer memory) instead of implementing two different blocks, one for encoding and one for the inverse process. We will pursue further possible enhancement of the processing speed of the coding process by using hardware devoted to this process (FFT-handling hardware to improve the coding speed).

3.4 Operating System

The need to access (browsing through the files and storing) the USB flash drive data indicates the neccesity to use an operating system, to take advantage of the existing libraries to handle the file system needs. Furthermore (see previous section), we chose to use software to implement the MP3 coding, which again can be found as open-source code for an existing operating system.

A wide-used choice for such applications is **uClinux**. Programming a uClinux OS to the board is a procedure which does is not resouce-hungry and enables the use of a wide variety of libraries for any use, thus wide programmability and function range. A careful design of the Operating System will save any such design of significant hardware complexity, with the single use of some (prefedined) resources (memory and gates).

Finally, the Operating System can handle many aspects of the file system used in this project.

3.5 Avalon Bus and Nios II processor

Controlling of some extra peripherals of the Altera DE2 board, such as a keyboard and the VGA display, on which the browsing of the flash drive's contents will appear, can be more easily done by means of Nios II software. Through this procedure, the implementation of the user interfaces for browsing flash drive entries and storing recorded clips can be enhanced.

The Nios II and Avalon Bus offer an easy way to

1. **SQNR** means **S**ignal to **Q**uantization and **N**oise **R**atio while **SNR** means **S**ignal to **N**oise **R**atio

2. The SNR or SQNR will approximately be $6.02n+1.76$ in dB, where n is the number of employed bitsif no post-processing filter is used.

implement these user interfaces and will be used to design a **G**raphical **U**ser **I**nterface (**GUI**) (to work on the Altera DE2 external VGA screen), which will display the contents of the extrnal storage medium. The user will, furthermore, have an option to name the files that will be stored on the external medium and choose which ones from the existing files will be played back.

3.6 Buffer Memory

The increased complexity of the calculations and modifications required to convert the ADC ”raw” output samples (after their conversion from serial to parallel form of course) to an MP3 stream indicates the need to use a rather fast memory block as an intermediate result storage and as a buffer for the general coding process.

Of all the available on-chip memory options, the best choice here would be the one offering the maximum storing capacity, which leads to the choise of SRAM. Despite some increase in the interface complexity, the particular memory is sufficient to handle the temporary storage of a reasonable amount of data, considering the application in hand.

4 Milestones

A careful planning of the system's milestones is necessary to ensure concstant progress and success in meeting the system's specifications in time. When planning for the design of a system, caution must be exercised in the careful and reasonable placement of milestones within the entire time frame devoted for the project, to ensure proper margins for testing and final corrections, as well as time to handle individual blocks' interfaces and the proper interaction of different building blocks.

We chose to divide the design of **mindTunes** by placement of three (3) milestones. The first will be devoted to the design of the Codec (ADC/DAC) and the ensurement of the proper sampling of voice clpis and playback of existing ”raw data” waveforms, using the minimum amount of complexity and resources (see ADC/DAC subsection). The time devoted to it will be approximately four weeks and, during that period, sufficent time will also be devoted into the development and gathering of all software required for the other steps.

The second milestone will include the MP3 coding (encoding plus decoding). It will handle the use of the corresponding software (i.e. the Operating System setup) and the interaction between it and the resources devoted to the coding (i.e. the data transfer between the buffer memory block and the resources devoted to the Operating System). The time devoted to it will be three weeks.

The final milestone will include the flash memory access and the development of the GUI to display and playback the existing audio files on the flash drive, plus of course intgrating the system to combine all previous mentioned features. By the time of the third milestone the result obtained will be the complete voice recording system mindTunes.

- Milestone 1
  03/28 Complete the audio interface
- Milestone 2
  04/04 Complete the encoding
- Milestone 3
  04/25 Complete the file system and prototype

5 Conclusion

In this project we wish to fully develop a complete voice recording system, with the feature to store voice clips in the wide-used MP3 format. The storage will be done in an external memory card and the user will be able to browse through the external memory's contents and listen to existing recordings, in $16 - bit$ quality sound. This report goes through all the key points involved in our design, as well as a careful floorplan of the system and of the time diagram that we chose to follow.

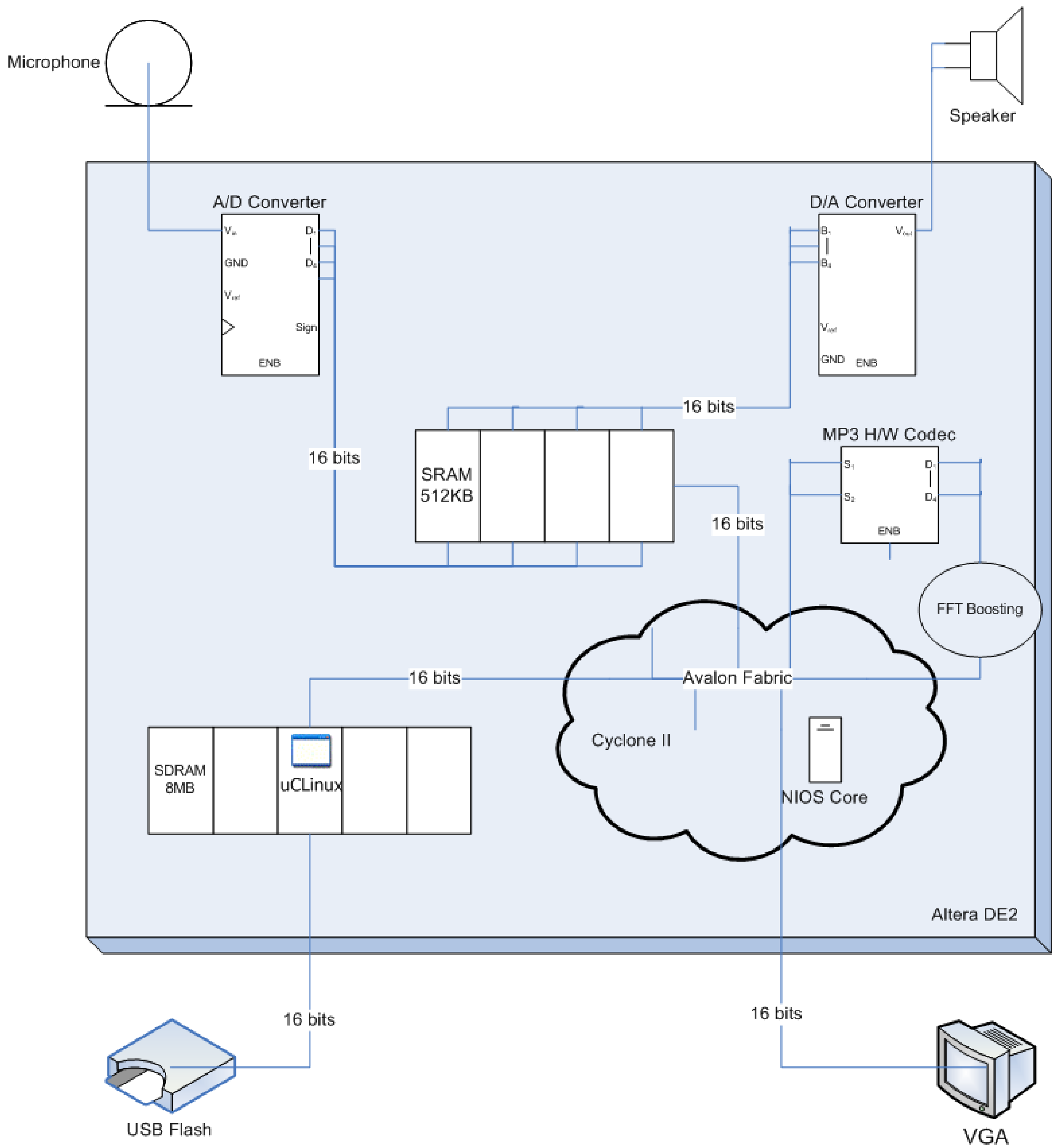

Fig. 1. Top Level Design of mindTunes