Lab 2
Data Acquisition

September 6, 2012

The purpose of this lab is to get more familiar with the DSK board and to understand the operation of the Codec. By the end of the lab you will be able to input and output signals through the Codec chip as well as see the effects of aliasing.

1 Introduction

The DSK board includes a C6713 floating-point digital signal processor and a 16-bit stereo codec (AIC23 which contains a coder (ADC) that converts analog waveforms to 16-bit signed integer sequences, and a decoder (DAC) that reconstructs an analog signal from a sequence of 16-bit signed integer samples.

Communication with the AIC23 is carried over two multichannel buffered serial ports (MCBSPs): the MCBSP0 is used as unidirectional channel to send a 16-bit control word to the AIC23, the MCBSP1 is used a bidirectional channel to send and receive audio data.

2 Lab

2.1 Part 1

Create a program that accepts a signal through the LINE IN of the AIC23 codec, and outputs it through the LINE OUT.

1. Start a new project.
2. Write a program that will allow you to read/write from/to the codec. You can use same code as the one you used in the previous lab to read and write.
3. Set the function generator to output a 500 mV amplitude and 1 kHz sinusoidal signal, and connect the function generator to the LINE IN of the board.
4. Set the sampling frequency to 96 kHz.
5. Connect an oscilloscope probe to the LINE OUT of the board.
6. Record and plot the magnitude of the output signal as you vary the frequency from 1 kHz to 96 kHz. Explain what you are recording.
7. Change the sampling frequency to 8 kHz. Vary the frequency and record your observations. Comment on your results.

2.2 Part 2

As mentioned previously, the read function for the AIC23 returns only one channel at a time. By changing the configuration of the MCBSP we can read both channels at the same time. To do that you need to use dsk6713config.h and modify your code to use the new configuration as shown in the sample file, sample.c to read and write the data instead of the standard function. All you have to do is save the file from the link and then include it into your project, also under `Build options/compiler/advanced/` select memory model `far`. Now you are reading both channels at the same time where the first 16 bits are the left channel and the second 16 bits are the right channel.

1. Change the sampling frequency to 96 kHz and modify your code to output only every third sample. Vary the frequency and record your observations.

2. Modify your code to read the signal from the function generator and output this signal to both channels.