Design Document Version 1

Low-Cost High Accuracy Spectral Test System

Iowa State University

Senior Design: 2015-2016

Team May1623

Advisor/Client: Degang Chen

Group Members: Tao Chen, Scott Poder, Yifan Jiang

Table of Contents

List of Figures and Tables

Project Statement

Accurate spectral test is widely used in science and engineering. In particular, accurate spectral test is critical to high performance data converters used base stations, medical instruments, seismic signal detection, military applications, and so on. IEEE standards and prevalent industry solutions impose several stringent requirements on the linearity of input signals, on exact coherency in sampling, and on tolerable jitter in the clock signal. All of these requirements make accurate spectral testing expensive and time consuming, and make the test setup difficult to maintain and calibrate.

The goal of this project is to develop a prototype test system for Extremely Cost-Effective Spectral Test. Recently published spectral test algorithms will be implemented for dramatically relaxing the stringent requirements. The concrete objective is to demonstrate a PCB test system for very low-cost, high-accuracy full spectrum test for high performance ADCs from Texas Instruments. The PCB will include a low cost sine wave generator, a clock generator, a low-order RC filter block, an ADC input driver, a socket for an ADC under test, and ADC output collection. The collected data will be transferred to a computer for analysis and display. Spectral test results will be compared with results obtained from Audio Precision instruments.

System Level Design

The system level block diagram can be seen below in Figure 1. For this test board, we will be using a DAC to generate an impure sine wave. This sine wave will be passed through a buffer for isolation and will be exposed to one of two filters. One of the filters will be an RC Filter while the other will be an RR Filter or voltage divider. Only one filter will be selected at a time, and our board's logic will switch between the two filters as needed. This output will be buffered and sent to the ADC under test. In real time, the digital output data of the ADC will be sent back to our FPGA in real time. From here, we will use spectral test algorithms on the PC to interpret the results of the ADC under test.

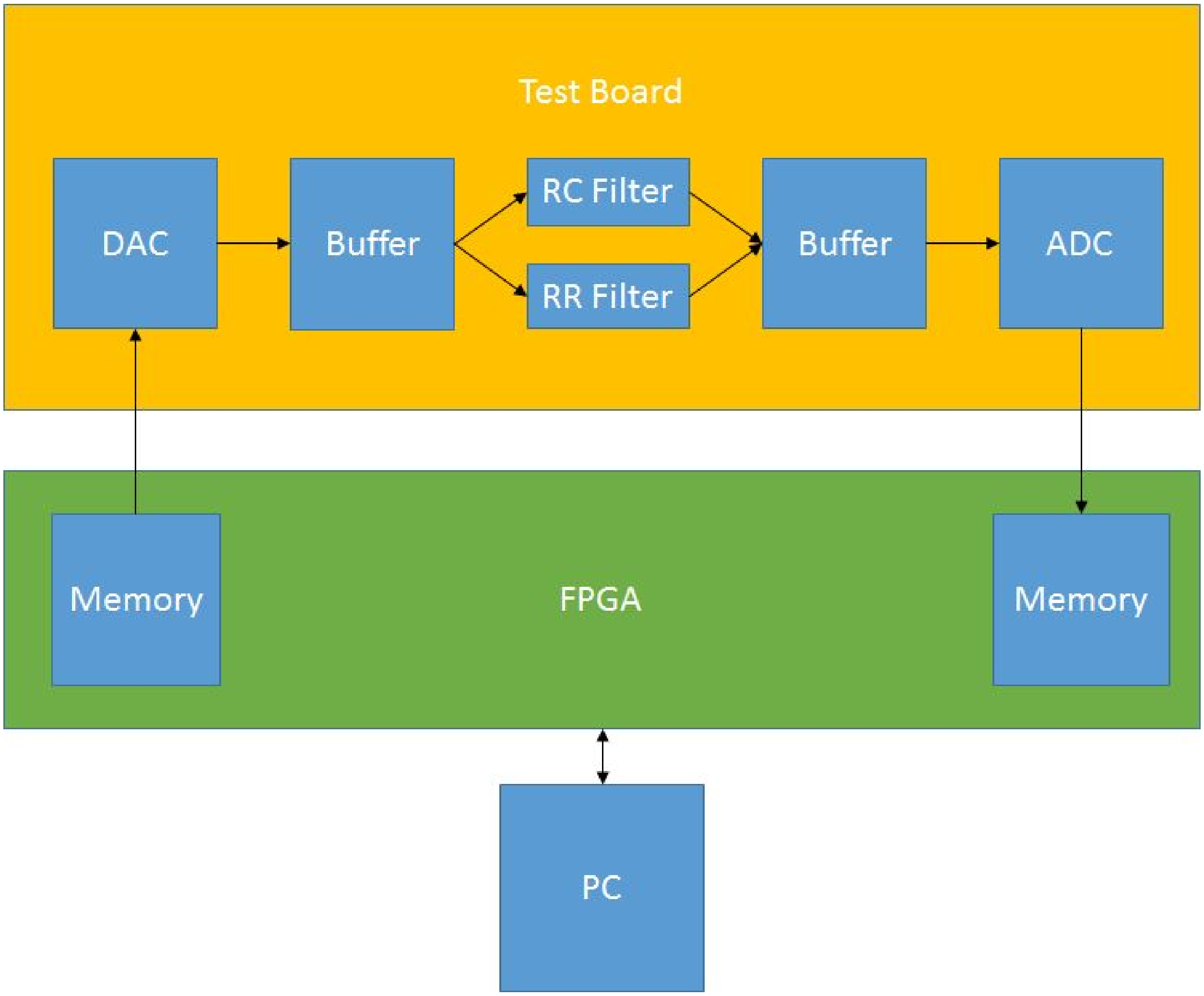

Figure 1: Block Diagram of the Low-Cost High Accuracy Spectral Test System

Detailed Description

ADC

The Analog to Digital Converter (ADC) under test will be the ADS8881 from Texas Instruments. This ADC is an 18-bit, 1-MSPS, SAR (successive approximation), precision ADC that operates on very low power. The ADC is also takes in a dual-differential input, which we have to account for in our input driver, which is described below. On the actual test board, a socket will be in place of the ADC so that we can run the spectral test on a large quantity of ADCs. The ADC contains some supporting circuitry around it, including a reference driver and an input driver which will be shown below once the design is completed in future versions of this document.

It is also important to note that the ADC will be operating in 3-wire mode, which we will go into detail in future versions when we create timing diagrams. The entire test board is designed around this ADC, so its selection was the first step in our design process.

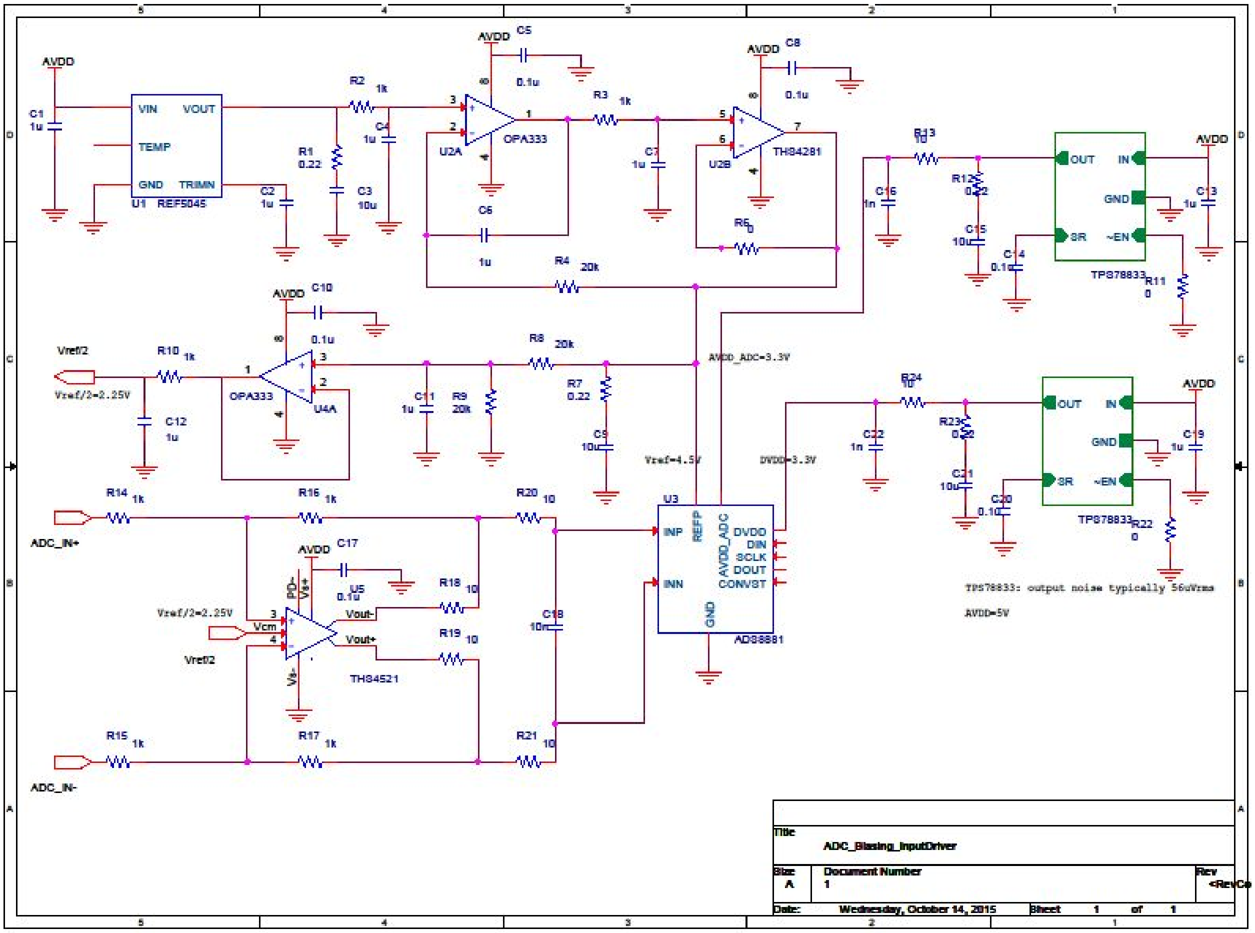

Figure 2. ADC biasing circuitry

Power Supplies

In order to provide power to the test board, we will be using a 9V battery to supply power to the board. Obviously, not all parts run at the same voltage, so voltage regulations in the form of LDOs will be used to obtain 5V, 4.5V, and 3.3V where appropriate in the system. The outputs of these LDOs will be considered clean, which is important for the precision of the analog portions of the schematic.

In addition, analog and digital supply voltages must be separated from one another, therefore, we will need to use separate LDOs for the digital/analog power supplies on the board.

DAC

The Digital to Analog Converter (DAC) that we selected for the board is the DAC8831 from Texas Instruments. This DAC is a 16-bit, low power, voltage output DAC whose architecture is of the R-2R structure. This DAC was chosen due to its sample rate (2MSPS) being larger than the sampling rate of the ADC (1MSPS). This is important for precision of our spectral tests.

We are using the DAC to generate an impure sine wave. The design that we will be using will generate a sine wave with an amplitude of 2.5V (which will be added in future versions when completed.) Due to the spectral tests we will be performing, we need multiple sine waves with varying frequencies. Our center frequency for testing will be 50kHz. We got this number through the following calculations:

(ADC Sample Rate)/2 = Nyquist Rate
(Nyquist Rate) * 10% = Center Frequency

(1M)/2 = 500k
500k * 10% = **50k**

We chose the maximum frequency to be slightly under our Nyquist value (~450kHz) and our minimum to be (~5k). From here we will use the geometric mean to find the intermediate frequencies, which leads us to our filter design.

Buffers and Filters

For the test board, we will have two filters present. One will be a low pass RC Filter while the other will be a RR Filter or voltage divider. Only one filter will be selected at a time. This selection will be handled digitally with the use of a low signal relay which will be controlled by a program. The RR Filter will be designed so that it contains a constant attenuation of 3dB. The RC Filter will be designed so that the corner frequency will be at -3dB. Due to spectral test requiring multiple frequencies, we will include a capacitor bank of various values and use the relay to obtain a wide range of RC Filters with the corner frequencies matching our input frequencies (see DAC).
In order to provide isolation between the Filter and the ADC/DAC, we must include buffers around it. For these, we will design one inverting amplifier circuit at the filter input and one at the filter output. The gain of the inverting amplifier at the filter input will be sqrt(2) in order to compensate for the 3dB loss of the filters. The inverting amplifier at the output of the filter will have a gain that ensures that the signal we feed the ADC matches its input driver circuit (still to be determined, and will be included in future version of this document).

Clock

We will use the clock drive from FPGA board. Because FPGA provide maximum of 100 MHz clock frequency. that is enough for both ADC and DAC operate clock frequency.
As it mentioned in previous paragraph, our sine wave frequency is roughly between 5 KHz - 450 KHz. In order to get sine wave frequency number from FPGA main clock's frequency, we can use Johnson Ring Counter, this type of shift register can divide a digital signal by an even integer multiple.

As for ADC, we will run it at the maximum clock frequency, 80MHz, in our case. Because our goal is to test the accuracy of ADC. the maximum error will occur at the maximum frequency. if the ADC at 80 MHz is good enough, we can safely say the ADC we test is qualified.

FPGA/Memory

We will choose FPGA - Cyclone V from ALTERA in our design. Because to generate sine wave, we need write a lookup table with sine wave value into a ROM. it is easy to achieve from FPGA. Also, FPGA is easy for use to design a clock divider, and feed the desired clock frequency into DAC and ADC. FPGA can reduce the difficult level of our project and it is easy to get from CPRE 281 LAB.

After ADC, we will store the data in FPGA's memory part. Than we can gather the data either by wireless communication or transfer the data via USB drive.

Serial Peripheral Interface

The chip to chip communication protocol we are going to use is called *Serial Peripheral Interface.* Basically, this protocol uses master to slave architecture to communicate between chips. Basically, there are four pins we concerned about, SCLK (serial clock), MOSI (master output slave input, output from master), MISO (master input slave output, output from slave), SS (slave select, output from master). In some situation, there might be only three pins, for example, DAC8831. We use 3 pins to control it, SCLK (serial clock), SDI (serial data input, same as MOSI, digital codes are going to be sent into this pin sequentially), ~CS (chip select, same as SS).

PC Interface

There are two softwares will be used to interface between the PCB and FPGA, Quartus and MATLAB. Quartus is used to control the FPGA, and MATLAB is used to generate the input wave information for the DAC. Take generating sine wave for example, MATLAB will be used to generate a file which contains the information of sine wave, like data points, frequency and amplitude. And then, this file will be used on the FPGA with the assistance of Quartus to control the DAC.

Algorithms

The algorithms will provided by Dr.chen next semester. we will use this new algorithms to manipulate our collected data.

Test Procedures

We will compare the dynamic spectral response from our “low cost, non-linear” input signal with the audio precision input signal test results. we can calculate the erroneous between the standard test method to our designed method.

Conclusion

Our project in system level at this point is pretty well layouted, what we are going to do next is assemble these function blocks or interfaces together, and integrate them into the printed circuit board. The chips for the PCB are chosen already, and their biasing circuitries are given. The filter designing is coming along; the FPGA controlling mechanism is in the process of researching and proofing. The ADC needs to be simulated in the Tina with the pure sine wave and the filters connected. The output data of ADC is going to be analyzed by the MATLAB with the given algorithm in the next semester. In a nutshell, the final goal of this semester is have schematics done and fabricate PCBs.

Modified date: October 27, 2015