UNIVERSITY OF VAASA

FACULTY OF TECHNOLOGY

TELECOMMUNICATION ENGINEERING

Reza Abbaspour

SPECTRUM WATCHING IMPLEMENTATION ON COGNITIVE RADIO

Master´s thesis for the degree of Master of Science in Technology submitted for inspection, Vaasa, 23th of August, 2011.

Supervisor Prof. Mohammed Elmusrati

Instructor Reino Virrankoski

ACKNOWLEDGMENTS

Firstly, thanks God for his generosity, supporting me incredibly in my life. Then I should sincerely be thankful to my thesis supervisor, Professor Mohammed Elmusrati for his immense support in both my coursework fulfillment and thesis preparation process which cannot be described in words. I would like also to especially thank Reino Virrankoski for his great and kind help.

I would like also give big thank to Texas Instrument Inc. for its support in terms of providing all required semiconductor components, made the project implementation possible.

I am profoundly thankful to my father and mother who are my best teachers as I am particularly indebted to them for everything.

Finally, I am extremely grateful to my dear wife, Mahdie for her eternal love, encouragement, understanding and supporting, making it possible for me to complete my Master's thesis.

Reza Abbaspour

Salo, Finland, 23th of August, 2011

Table of Contents

LIST OF ABBREVATIONS

AC	Alternate Current
ADC	Analog to Digital Converter
API	Application Program Interface
AUC	Advanced Ultra-Low-voltage CMOS
BOM	Bill of Materials
BPF	Band Pass Filer
CDC	Clock Domain Crossing
CM	Common Mode
CMIR	Input Common-mode Range
CMOS	Complementary Metal Oxide Semiconductor
COM	Contrast Parallel Port
CPLD	Complex Programmable Logic Device
DAC	Digital to Analog Converter
DC	Direct Current
DFC	Data Flow Controller
DSP	Digital Signal Processor
EEPROM	Electrically Erasable Programmable Read-Only Memory
ESD	Electrostatic Discharge
FDA	Fully Differential Amplifier
FIFO	First Input First Output
FPGA	Field Programmable Gate Array
GND	Ground
HBM	Human Body Model

IC	Integrated Circuit
IDC	Insulation Displacement Connector
ISM	Industrial Scientific Medical
LDO	Low Dropout Regulator
LPF	Low Pass Filter
LVCMOS	Low Voltage CMOS
LVPECL	Low Voltage Positive Referenced Emitter Coupled Logic
MPSSE	Multi Protocol Synchronous Serial Engine
OEM	Original Equipment Manufacturer
OPAMP	Operational Amplifier
OS	Operating System
PC	Personal Computer
PCB	Printed Circuit Board
PID	Product ID
QoS	Quality of Service
RF	Radio Frequency
SRAM	Static Random Access Memory
TI	Texas Instruments
UART	Universal Asynchronous Receiver/Transmitter
USB	Universal Serial Bus
VCO	Voltage Controlled Oscillator
VID	Vendor ID

TABLE OF FIGURES

LIST OF TABLES

UNIVERSITY OF VAASA
Faculty of Technology
Author: Reza Abbaspour
Topic of the Thesis: Spectrum Watching Implementation on Cognitive Radio

Supervisor: Prof. Mohammed Elmusrati
Instructor: Dr. Reino Virrankoski
Degree: Master of Science in Technology
Department: Department of Computer Science
Degree Program: Degree Program in Telecommunication Engineering
Major of Subject: Telecommunication Engineering
Year of Entering the University: 2009
Year of Completing the Thesis: 2011 **Pages: 110**

Abstract

Modern wireless communication constantly faces RF band shortage especially in commercial applications while all useful bands are already licensed and occupied. Thus, the initiation of new wireless services is not possible because of the critical shortage of free bands. On the other hand, there are only small band potions (ISM band) left for free where the interference level is very high. That is why new technologies are utilized to use the frequency bands efficiently. Cognitive radio is one important and promising technique to reuse the licensed bands, which means that the usage permission of a certain frequency band must not make any harmful interference to the primary (licensed) user and this is accomplished by an accurate spectrum-watching device to decide whether a certain band is free or not. In this project, high-speed spectrum-watching hardware is implemented to be employed in cognitive-radio. However, spectrum-watching device is a generic hardware for different purposes but, the main application is to be used for cognitive radio. Thus the device input is customized to accept RF analog signal from a digital receiver module. The system architecture is designed and implemented in fully hierarchical manner, performing modular approach by allowing the end-user to work with functional blocks in form of daughter boards. In other words, sub-systems are prototyped as distinct modules which can be plugged in together in order to handle the spectrum-watching process. The whole system converts analog signals with maximum 8MHz to digital bit stream and transferring the information via a USB 2.0 connection to PC side for further analysis, which means that the input 8MHz baseband signal can be monitored by a PC. In this regard, RF down-converter module should be configured in a manner that to be compatible with designed spectrum-watching device by properly setting the internal local-oscillator and sweeping circuit. In sum, the thesis highlights the design challenges, system architecture and implementation procedure as well as prototypes' functionality in order to produce knowledge which might be only maintained and protected by commercial manufacturers that produce high-speed ADCs and data acquisition blocks. In this regard, the project **homepage** at the following address represents all information as well as source codes and executable files, supporting the project: http://www.abbreza.com/en/spectrum.htm

KEYWORDS: Spectrum-watching, High-speed ADC, PC-based Data Transfer, Cognitive Radio

1. INTRODUCTION

Wireless data communication industries meet growing demand for higher data-rate while, communications service providers are faced with bandwidth constraint. This forces commercial users to subscribe their required bandwidth for particular applications in high price and thus rethinking how the bandwidth should be allocated to remain high QoS. In this situation new mobile communication technologies are under develop and cognitive radio is considered as a promising and effective technique to improve radio spectrum utilization. This is a form of wireless communication in which a transceiver can intelligently detect which communication channels are in use and which are not, and instantly move into vacant channels while avoiding occupied ones. This optimizes the use of available RF spectrum and prevents wasting resources while simultaneously minimizing interference to other users. To achieve this condition, an accurate spectrum-watching device is needed to decide whether a certain band is already occupied or not. On the other hand, in its most basic form, cognitive-radio is a hybrid technology involving software based controlling. That's why new algorithms should be developed in order to be applied in software side and thus hardware-based experiments must be performed in research level. This reveals the importance of spectrum-watching hardware implementation.

1.1. Scope of Research

The scope of research involves pure hardware implementation of high speed spectrum-watching. The hardware is designed from scratch, implemented and tested successfully. The commercial high-speed data acquisition including ADC unit are available in market in high price. With respect to this issue, a research has been conducted in order to design and implement a cost effective hardware which can flexibly be customized for any application. Indeed, a commercial PC based high-speed data acquisition is completely a black-box expensive device. In other words, manufacturers protect their design documents of end-products against the users and that's why no modification or even customization is possible. Moreover, the key points and design secrets are always

considered confidential by manufacturers and thus not much documentation is available to study the concept deeper. Therefore this project is started from scratch in order to find all the design tips. However, algorithm development and RF receiver hardware implementation are not within the scope of this thesis. But, the project is well-documented and prepared to be fully compatible with RF blocks especially used in cognitive-radio concept.

1.2. Project Open Issues

Figure 1.1 shows the rough system block diagram of spectrum-watching device. The PC interface is considered as a bottleneck, limiting data transmission speed which is the most critical part in this project, thus designing a high-speed PC interface is the main concentration. On the other hand, there are different types of high-speed PC port available to connect an external hardware to a computer but a successful selection is the most popular one and easy-to-use that can effectively be integrated in a design. With respect to those issues, the professional PC interfaces such as PCI-EXPRESS and Fire-Wire (IEEE 1394) do not meet the project requirements and thus they can be omitted from the selection phase because they are expensive to use and requiring additional internal supporting hardware. Hence, the USB 2.0 is considered as a potential candidate, supporting high-speed data transfer up to 480 Mbps and it is popular among the computer users. Additionally, the USB 2.0 is widely used in many PC-based devices, extremely easy-to-use for end-users and thus it is well suited for this project.

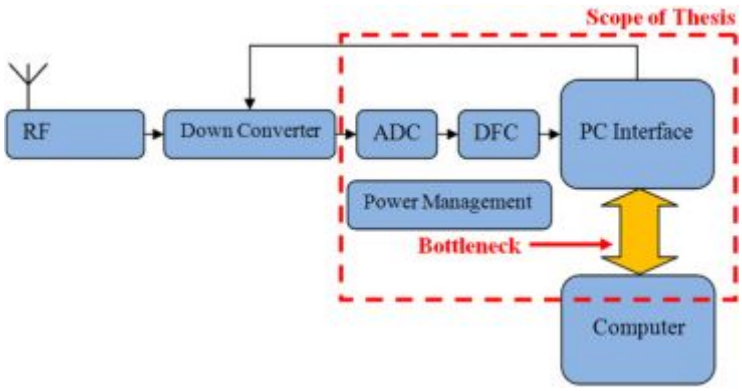


Figure 1.1. The block diagram of spectrum-watching system

In sharp contrast, the USB 2.0 features non-real-time data transmission profile, facing the hardware architecture several design challenges. Firstly, the OS may interrupt data transferring process in a time moment, causing data loss in real-time applications. In fact, typical USB-based devices (e.g. memory sticks) do not have any difficulty with this issue while data stream can be stopped at any time. On the other hand, the elapsed time of data transfer interrupt causing by OS is not even comparable with write access time of external low data-rate hardware and although the device may demand real-time data transfer (e.g. web-camera), there is no need to implement a congestion control sub-system. But spectrum-watching is considered as a high speed real-time application and therefore accurate data flow controller as well as data buffering and data aggregation handler should be designed and implemented. All those functions together require complex hardware architecture consist of expensive FIFOs, SRAM memories, DSP processor and FPGA-based data bus controller. Conversely, a cost-effective solution should be less complex than the aforementioned architecture so an extensive research has been conducted to simplify the project low-level requirements. Furthermore, cost-reduction study has been performed to moderate the project BOM cost. From this point of view a trade-off between system performance and end-product expenses is considered; therefore the costly components such as memories, FPGA and processor have been ignored in final system architecture and eventually a cost-effective, memory-less, processor-less high speed spectrum-watching device has been implemented.

1.3. Project Resources

This project is pure hardware implementation, and therefore a lot of trials in designing and component selection have been done. Thus, a reliable semiconductor manufacturer was needed to prop the design trials with wide variety of components and that’s why all the project electronic components are sourced from TI whose manufactures best-in-class analog and digital semiconductor components. Moreover, extensive design literatures, application notes, technical datasheets and design examples are exclusively available for TI-based designs. Furthermore, many hobbyists as well as students and interested engineers exploit TI free samples and thus, a lot of open-source projects can publicly be

found by googling. All these opportunities facilitate implementation process while all design aspects are unknown at beginning.

PCB layout implementation of Prototypes, component assembly and hardware test are performed in *PCB manufacturing lab* located in ***Technobothnia Research Center*** of University of Vaasa. Moreover the test and measuring facilities of *automation and power electronic lab* are employed. In this regard, waveform monitoring and recording is mainly accomplished by *YokoGawa DLM2000 Series* oscilloscope (which is recently calibrated by factory) in this project.

1.4. Brief Project Overview

The system block diagram illustrated in Figure 1.1, generally describe the system architecture by subsystems as follows:

- High-speed data conversion unit
- USB 2.0 controller circuit
- Power management unit
- Data flow controller
- Analog signal conditioning

In principle, those functional blocks are selected to achieve ***high-level*** *system requirements* that are listed below:

- 8 MHz sample rate
- At least 8-bit resolution
- Maximum 10^{-2} error rate
- Compatible with MATLAB
- Open-source
- Cost-effective
- Easy-to-use

In addition, the final design cost should be as low as possible which means that a cost-effective approach is the most successful one, while the required performance is met. As it described before, the system architecture is based on memory-less and processor-less

design. This means that data stream in form of parallel bits are directly transferred from the 8-bit ADC IC's output at 64 Mbps data-rate to USB controller unit. Since the system does not operate under control of a high-speed processor, DFC unit is utilized to prevent data aggregation.

In sum, this project is considered as a public open-source design and thus it is well-documented, including every single deign detail. In this regard, the project **homepage** at the following address represents all information as well as source codes and executable files, supporting the project:

<URL: http://www.abbreza.com/en/spectrum.htm>

1.5. Thesis Structure

The subsequent project report is organized into the following chapters:

Chapter 2:

Provides a guideline to use the device in couple of seconds. A brief instruction is also included to show how the device can be connected to an external module.

Chapter 3:

Describes the idea behind designing a preliminary demonstration board. In this chapter a rough sketch on selecting the components based on their performance has been drawn.

Chapter 4:

The ADC unit design procedure has been discussed in details. Moreover, required configurations to drive the ADC IC are well-defined in this chapter.

Chapter 5:

Proposes and discusses the USB 2.0 controller and required sort of programming tips in order to read information from the spectrum-watching device connected to USB port.

Chapter 6:

Clock management unit and its importance have been discussed. Furthermore, hardware design details and prototypes are also elaborated.

Chapter 7:

An effective solution for data flow controller is proposed and prior architectures are discussed in details.

Chapter 8:

Low distortion signal conditioning circuit requires to drive the ADC input is discussed in this chapter in hardware level.

Chapter 9:

A brief overview on power management unit is provided to show how the hardware is designed to power up the spectrum-watching circuitries.

Chapter 10:

Draws a conclusion of this thesis investigating how well the project could satisfy the objectives and how future works might strengthen the device architecture.

2. USER GUIDE

The primary goal of this architecture is to provide as modular design as possible, to fulfill the system requirements and to gain future proof for functionality extensibility. The whole system is implemented on five distinct daughter boards that should be plugged in together to be operational, which means that the daughter boards playing complementary role and cannot be functional as a stand-alone system (except power supplier block). In this section, a brief guideline in using the device has been provided, showing how an end-user can setup the system in couple of minutes. Figure 2.1 shows the assembled daughter boards, which are ready to be connected to a PC via USB 2.0 port.

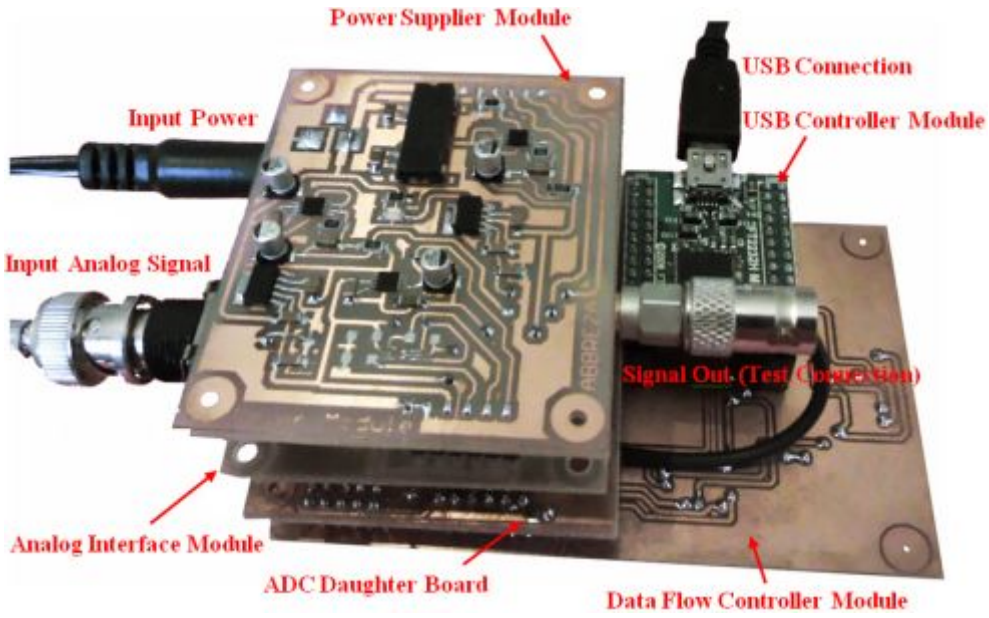


Figure 2.1. Complete assembled spectrum-watching device

It is extremely important to attach the daughter boards in the right order where is shown in Figure 2.1. Furthermore, the right place for each module from bottom to top is listed in Table 2.1 in ascent order. The entire daughter boards are connected to each other through pin-header connectors, thus it is easy to replace a module with a new one or even add a new daughter board, provides a new functionality. Moreover, modular

design approach brings a flexibility to test the hardware easier than highly integrated one. Because the outputs and inputs are well-defined on modules connectors and thus the test and measuring procedure can be conducted in a shorter time.

Table 2.1. The place of modules for correct system assembly

Place	Board	Function
1	DFC	Control the data flow from the ADC unit to USB port.
2	ADC	Heart of the device! Providing analog conversion and parallel data bus driver.
3	Analog Interface	ADC input drive circuit
4	Power supplier	Providing seven different DC voltage-levels for circuitries
5	USB	Handling the USB 2.0 port for data transmission.

The spectrum-watching device has two inputs and a single output where are defined in Table 2.2. Based on the device specifications, the input DC power can be provided by a standard low-cost AC-DC switching adaptor. The device supports a typical plug in 5.5mm power jack with ***positive polarity on internal shaft***.

Table 2.2. System inputs and output characteristics

Function	Type	Characteristic
Input DC power	Input	Typically requires 400mA, minimum 7v DC and maximum 15v DC.
Input analog signal	Input	50Ω input impedance
PC data connection	Output	According to USB 2.0 compliant

The device input is characterized as a 50Ω signal sink, which means that the external RF receiver block or down-converter must feature 50Ω output impedance in order to be matched with ADC drive circuit, otherwise some passive components can be changed on analog interface daughter board to adjust a desirable input impedance (see chapter 8).

The USB connection on spectrum-watching device is ESD-protected based on HBM standard. It is should be mentioned that the spectrum-watching data transferring rate is only compatible with USB 2.0, which means that the older technology (USB 1.0) is no longer employed for those applications demand for PC-based high data rate transmission. Once a PC gets connected to spectrum-watching device, Microsoft Windows asks for driver which can be downloaded from the project **homepage**.

In order to start data capturing, the operator should firstly download the device PC-based application software form the project **homepage** and then run it on Windows machine. This enables the end-user to activate the spectrum-watching hardware via the USB port and starting the data acquisition process.

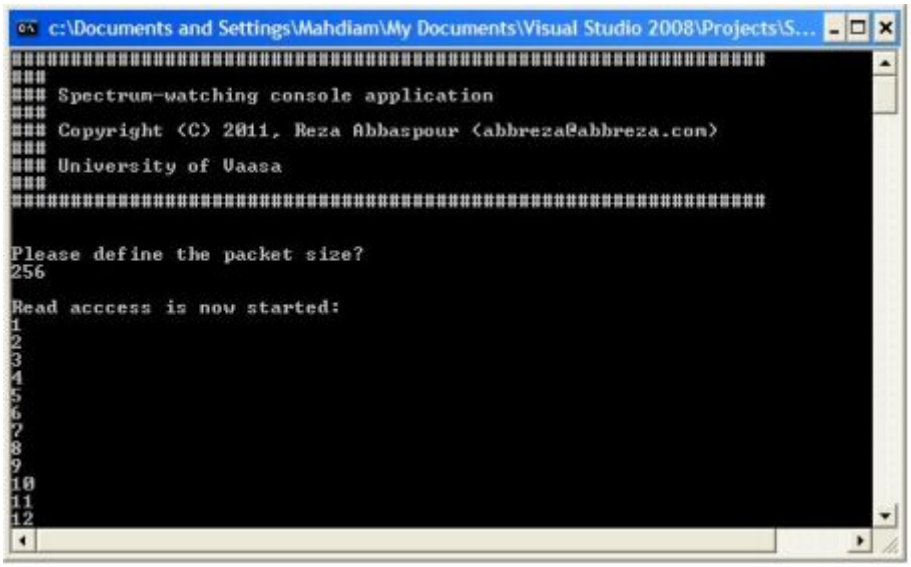


Figure 2.2. Screen shot from the PC-based application software

As the Figure 2.2 shows the Win32 Console application screenshot, the user should determine the number of data Bytes which would be read from the hardware. In other

words, the number of digital samples is determinable in the software in order to limit the data stream in a time moment (see section 5). For instance, if one defines 255 at beginning, the PC will read 255 subsequent samples from the input analog signal and shows on display. Moreover the captured data will be stored in text file (spectrum.txt) on the main root directory of the PC (e.g. C: \) for further analysis. Thus, the end-user can simply use the text file in MATLAB in order to test and develop the communication algorithm.

3. DEMONSTRATION BOARD

Component selection could be considered as a crucial task in implementation phase, while a wide range of components are well-suited to the application, thus a preliminary list of components has been prepared based on the high-level project requirements. The selection is performed based on required functionality and component performance. As a consequence, the demonstration board (see Figure 3.1) has been designed in order to evaluate the selectees such as ADC IC, clock distributor IC, power management, signal conditioning and analog circuitry.

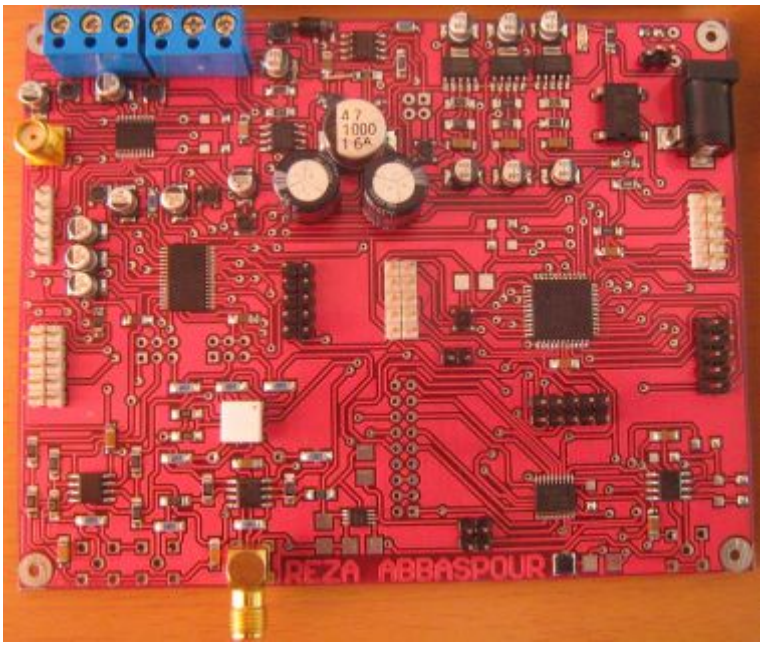


Figure 3.1. Demonstration board

From the other point of view, the prototype has been implemented in order to examine hardware performance, while the components are served to be operated together on single PCB layout.

3.1. Component Selection and Hardware Design

The ADC IC characteristics could even direct the design procedure so it should carefully be selected. For instant, 10 MSPS, 10-bit resolution ADC IC demands for 100 Mbps digital interface while 8 MSPS, 8-bit resolution ADC IC generates 64 Mbps data stream. This makes big difference in designing the PC interface part. Thus, relatively high performance, low-cost, 8-bit resolution, 8 MSPS programmable ADC IC should be selected. With respect to those specifications, the THS10082 has been chosen which contains extra integrated blocks such as internal FIFO, power independent output buffers and so forth (see section 0).

In principle, the DAC IC should be compatible with the THS10082 in order to be qualified, which means that its bandwidth should be matched with the ADC input bandwidth. Moreover, the DAC output update-rate as well as write access time must be also taken into account as the DAC selection criteria. That's why the DAC7821 is chosen with update-rate 20.4 MSPS which is almost twice of the THS10082's sampling rate (8 MSPS). On the other hand, the DAC7821 supports parallel data bus with fast write access (17 ns), making it suitable to be connected directly to the THS10082.

In designing the DAC unit an external voltage reference is required to support the DAC7821 and this is accomplished by a voltage reference IC (REF5025). On the other hand, the THS10082 should be initiated at startup via an external controller which is an AVR microcontroller (ATMEGA32). The ATMEGA32 can apply the user settings on the THS10082 in order to initiate it with right configurations that provide more flexibility for designer to exploit all the capabilities of a programmable ADC IC.

Basically, driving the THS10082's input through a reliable analog interface is an essential issue, guarantees system operation with minimum signal distortion. Thus, the active drive circuit is built around high-speed, high-slew-rate and very low noise OPAMP (THS4052). Additionally, a RF transformer (Z9280-AL) is employed to isolate the analog input circuitry from the ADC part. On the other hand, since the THS4052 has proper technical features in terms of bandwidth and slew-rate, it is also considered to be used as the bipolar output driver circuit, buffering the DAC7821's output analog signal.

Particularly, all the digital units demand for synchronous clock signals with different frequencies, generating from a single clock source and thus, this significantly reduces

clock jitter and latency problems. In order to achieve the objective, high performance clock management IC (CDCE937L) is employed to generate the system clock from a single crystal by programmable PLLs and clock dividers (see section 6.1).
Essentially, all the aforementioned hardware blocks should be powered up with specific DC voltage-levels based on their power requirements. Thus, the power management unit has been utilized in order to provide different supply voltage-levels through four LDOs and a negative voltage generator. The system block diagram is illustrated in Figure 3.2, provides a complementary description on hardware design.

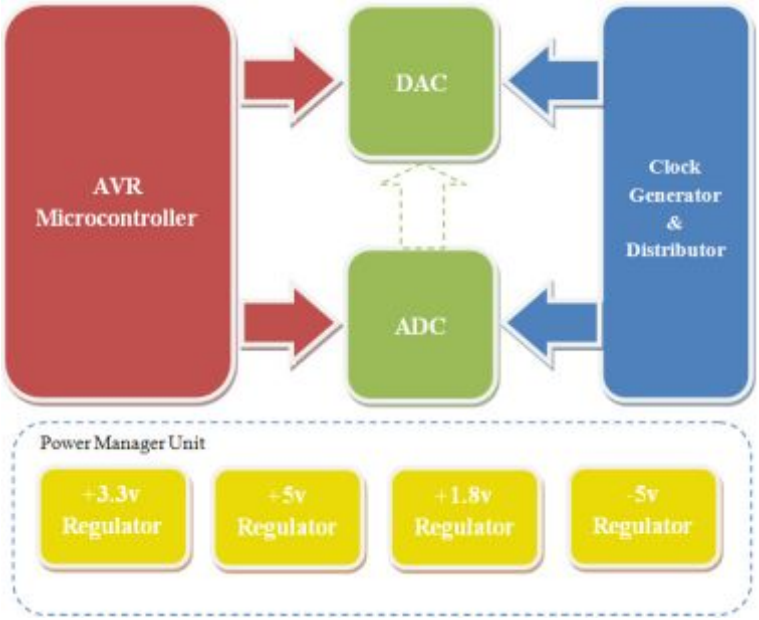


Figure 3.2. System block diagram of the demonstration board

3.2. System Architecture

In prototype level, the designed system as well as implemented PCB layout should be flexible as much as possible. Thus, firstly convenient access to every I/O pins should be provided via pin-header connectors. Secondly, the data bus connection between the THS10082, the ATMEGA32 and the DAC7821 is not fixed anymore, providing more convenient test and measurement, which means the parallel data bits on the THS10082

output can be transferred to the DAC7821 via flat-cables and connectors (see Figure 3.1) and therefore the system evaluation can be performed easier by connecting the hardware units through the IDC cables. Since, the THS10082 internal registers should be setup, the microcontroller always should be connected to it via the IDC cable at the startup time. Thus, the THS10082 will be initiated in this way and then it is ready for signal conversion. Furthermore, the ADC parallel data bus could be linked to the DAC7821 via the pin-header connector in order to start recovering the analog input signal from the digital bit stream. All the input/output analog signals are passed through SMA connectors in order to prevent signal degradation while, an external circuitry is injecting or capturing analog signals to demonstration board and thus more accurate measuring, test and evaluation can be performed.

3.3. Hardware Evaluation and Practical Result

According to the project plan, the first prototype must be implemented in order to demonstrate how well the chosen electrical components operate. Indeed, the ADC part is the most critical part, thus the demonstration board is designed especially to examine the ADC IC (THS10082) performance. Moreover, the DAC IC (DAC7821) is also utilized to operate in conjugate with the THS10082, because in this way nonlinearity of the digitization process can be measured. This means that, the input analog signal would firstly be converted by the THS10082 to digital bit stream and then the DAC7821 will reconvert the digital data to the equivalent analog signal. Hence, a comparison between the input and output analog signals can reveal the conversion nonlinearity as it shown in Figure 3.3. Based on the captured signals, some distortions can be observed. This could be happened by nonlinearity of conversion process both in ADC and DAC ICs or/and additive noise in OPAMP-based analog drive circuits. Anyhow, the result is somehow satisfactory, demonstrates sufficient functionality of the THS10082 for using in final implementation of the spectrum-watching device.

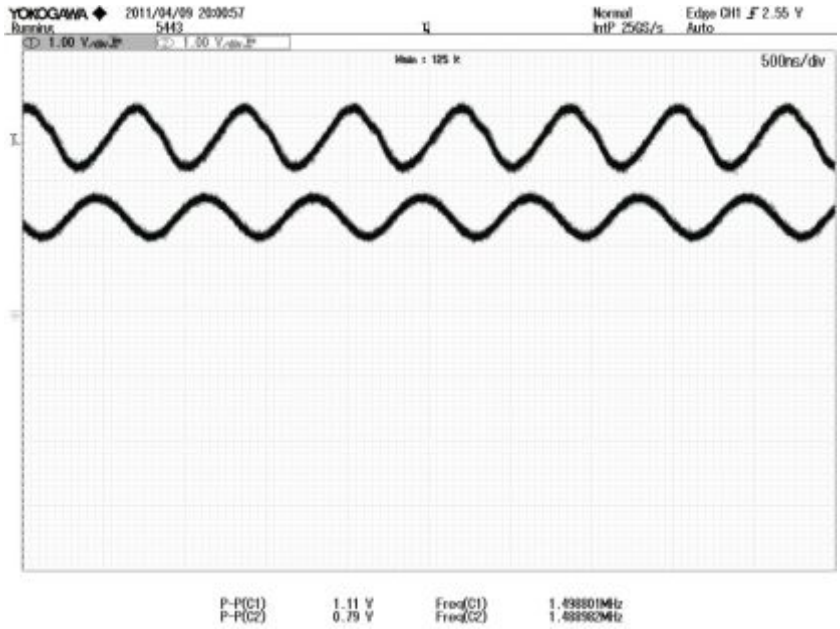


Figure 3.3. ADC input analog signal in compare with the DAC output signal

4. HIGH SPEED ADC

Spectrum-watching mainly relies on digital-to-analog converter in order to transform the analog world to corresponding discrete values. This enables complex analysis on relatively low-cost powerful digital processors or the conventional PCs, leading to less processing cost in compare with analog computation. On the other hand, wide bandwidth ADC devices demand for high-speed interface in order to transfer information to a PC. Commercial interfaces (e.g. USB 2.0 and UART) could be considered as a bottle neck with respect to their operating speed and non-real-time profile. In this section, the implemented ADC hardware (see Figure 4.1) will be described in details.

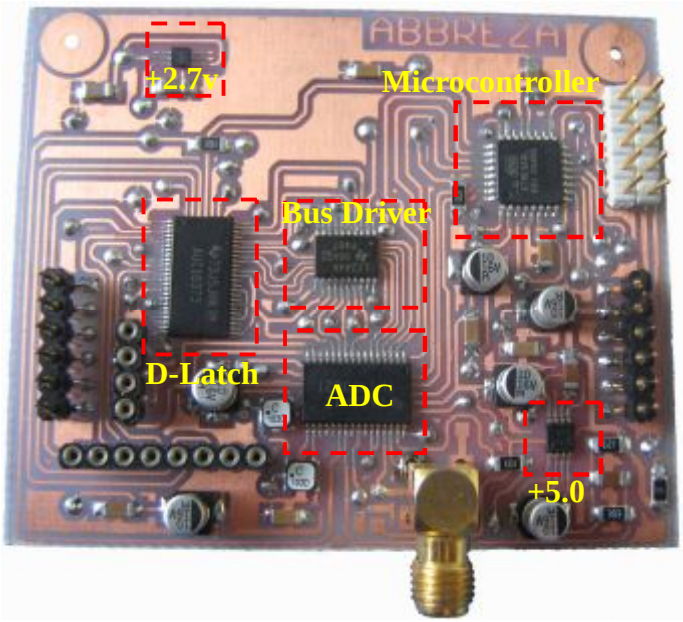


Figure 4.1. The ADC unit daughter board

4.1. ADC IC Specifications

According to the project outline the spectrum-watching hardware should be capable to convert the analog signals at 8 MHz sampling frequency to 8-bit digital symbols. The

ADC IC running at 8 MHz, consider as the high-speed one which is more expensive than the traditional ADCs work in matters of hundreds KHz. That's why the ADC part in this project has a big effect on the BOM cost, thus it is taken into account as one of the most critical parts in component selection phase. With respect to this issue, the *thesis supervisor* suggested to go over some irregular solutions such as mixed signal FPGAs. Based on the project pre-study, ADC module is fully integrated in special types of FPGA which are called mixed signal FPGAs and therefore, the system blocks such as USB controller, bus driver, multiplexer, SRAM interface and so forth can be implemented on the FPGA. The integration of all functions on FPGA-based architecture may significantly reduce the project cost but later, according to the project feasibility study the mixed signal FPGAs could not meet the project requirement in terms of high sampling frequency, because the integrated ADC module at most would be operational in sub-MHZ. Hence, employing a discrete high-speed ADC IC has been adopted in component selection phase. On the other hand, while TI is considered as the project component supplier, the ADC IC has been selected from this company components' pool. Accordingly, the THS10082 is chosen as the most proper one that its features are too close to the determined specifications for the ADC IC. The THS10082 is a CMOS, 10-bit, ADC IC with the following key features:

- Maximum 8 MHz sampling frequency
- Integrated 16-Word FIFO
- Simultaneous sampling of two single-ended signals or one differential signal
- 59 dB SNR
- 1.5v and 3.5v internal voltage reference
- Pipeline CMOS architecture
- Parallel microcontroller/DSP interface
- Programmable internal control registers

Since the THS10082 is a *single-supply* ADC IC with high bandwidth, it is well suited for ***high-speed data acquisition and telecommunication*** applications. It has an internal error correction stage providing no missing codes over a wide temperature variation.

Moreover, the internal programmable registers make the device operation flexible in many hardware architectures. The connected processor can take advantage of using load-off time while the integrated FIFO performs data storage task. The ADC IC can operate either with the internal clock oscillator or the external clock source which makes it a flexible device in clocking issue. The THS10082 internal block diagram is illustrated in Figure 4.2.

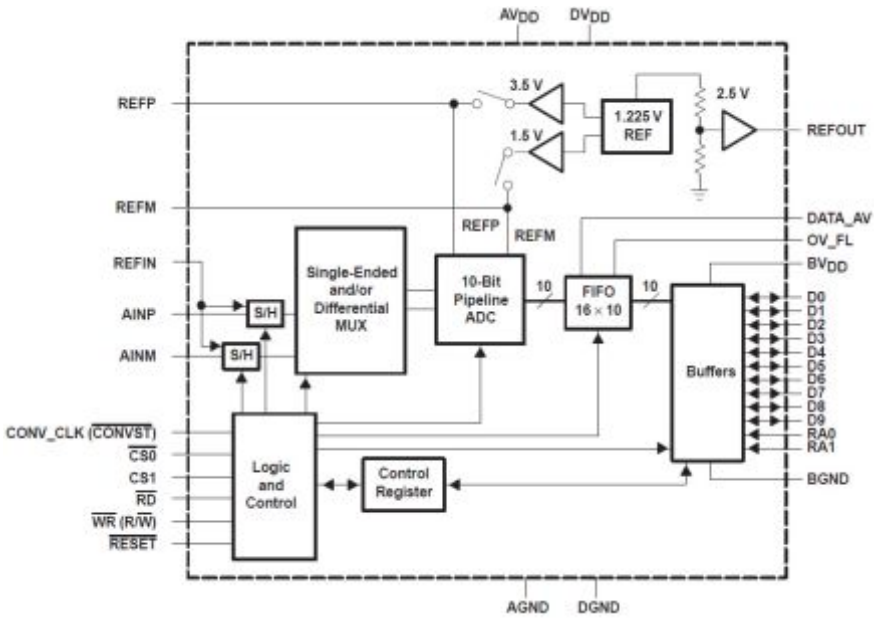


Figure 4.2. The THS10082 internal block diagram (Datasheet)

There are two internal 12-bit control registers (CR0, CR1) which can be programmed with desirable values in order to setup the ADC IC operation. The control registers make the THS10082, a programmable device. On the other hand, no internal EEPROM memory exists inside the IC to store user's settings. That's why, when the IC is powered up all predefined configurations should be reapplied once again by an external circuitry. In this project 8-bit AVR microcontroller (ATMEGA8) is employed to accomplished the task.

4.2. Data Bus Controller

Bi-directional parallel data bus has been internally shared between the FIFO output and the internal control registers, thus a multiplexer circuit is required to switch the ADC data bus either on microcontroller or USB interface. In other words, when the ADC internal control registers are initiating, the ADC parallel data bus should be connect to microcontroller's I/O pins but when the THS10082 is converting the analog input signal to digital symbols the data bus should be connected to USB interface. This means that the data flow direction at initialization time is from the ATMEGA8 to the THS10082 and in running condition is from the THS10082 to the USB interface. As the schematic in Figure 4.3 shows, the switch circuit is simply based on an *octal buffer IC* (SN74LVC244) with 3-state outputs. This allows the microcontroller to connect/disconnect its I/O pins to/from the ADC parallel data bus through SN74LVC244 control pins (1OE and 2OE). When OE is low, the SN74LVC244 conducts digital signal from inputs A to the outputs Y and when OE is pulled high, the outputs are in the high-impedance state (disable mode). Since the SN74LVC244 is an octal buffer, it cannot drive all the ten bits of parallel data bus so the first two bits (Bit0 and Bit1) are directly connected to the microcontroller. Because the 10-bit data bus is already reduced to 8-bit data bus by eliminating ***the first two bits*** of the ADC output (LSB bits), and thus the first two bits never be read by USB interface. In this case there is no need to consider the first two bits as bi-directional data bits because they would never be required to switch on USB interface.

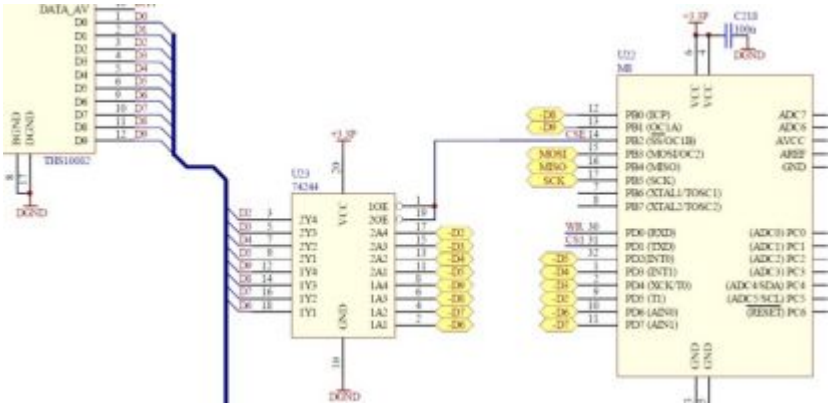


Figure 4.3. The data bus switch circuit

Pin mapping between the microcontroller I/O pins and the ADC output has been provided in Table 4.1. It could be considered as a guideline, showing how the embedded application software should be configured.

Table 4.1. Mapping between microcontroller I/O pins and the ADC output

Bit#	ADC Bit Function	Microcontroller Bit Function
0 (LSB)	D0 (DATA)	Port C#4 (I/O)
1	D1 (DATA)	Port C#3 (I/O)
2	D2 (DATA)	Port D#5 (I/O)
3	D3 (DATA)	Port D#4 (I/O)
4	D4 (DATA)	Port D#3 (I/O)
5	D5 (DATA)	Port D#2 (I/O)
6	D6 (DATA)	Port D#6 (I/O)
7	D7 (DATA)	Port D#7 (I/O)
8	D8 (DATA)	Port B#0 (I/O)
9 (MSB)	D9 (DATA)	Port B#1 (I/O)

Moreover, the microcontroller should firstly select the target control register (CR0 or CR1) via the addressing control bits (RA0 and RA1) where are directly connected to the ATMEGA8. During the write process, the data bits (D0 to D9) carry the information which should be written on CR0 and CR1. Table 4.2 shows the AVR I/Os' connections to RA0 and RA1.

Table 4.2. Addressing modes via control bits

Addressing Control Register	RA0	RA1
CR0 is selected	0	0
CR1 is selected	1	0
Reserved	0	1
Reserved	1	1

Additionally, the ATMEGA8 is responsible to setup the read or write access by switching on right combinations of chip select inputs (CS0 and CS1), write input (WR) and read input (RD). The WR control pin is configurable input which can be used to handle the both read and write functions simultaneously because the connected processor may consist of a combined read/write output signal. Based on the internal logic circuit of the ADC (see Figure 4.4), reading from the THS10082 or writing to the THS10082 are taken place by an internal RD_{int} and WR_{int} signals, generating from the logical combination of the external signals CS0, CS1 and RD.

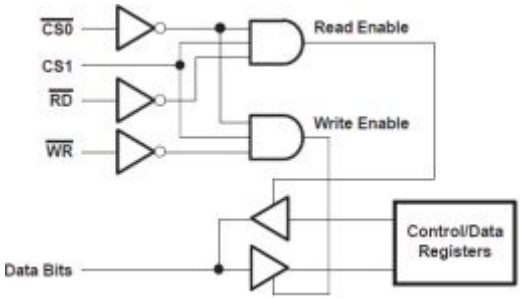


Figure 4.4. Internal logic circuit of the THS10082 (Datasheet)

The last external signal (either WR, CS1 or CS0) to become valid generates WR_{int} interrupt signal while the RD pin is inactive. On the other hand, the last external signal (either RD, CS1, or CS0) to become valid generates RD_{int} interrupt signal while the WR pin is inactive so this property can be used to control the read and write process accurately through four distinct logical combinations which are available to designer in

order to interface the external controller on the basis of timing diagram. Table 4.3 shows how the control and signaling pins of the THS10082 are connected to the ATMEGA8 I/O pins. Since the ADC output buffers are supplied with 3.3v, the AVR microcontroller should also support the same voltage-level. From this point of view, the “L” type AVR microcontroller is employed which could be operated from 2.7v up to 5.5v.

Table 4.3. Mapping between microcontroller I/O pins and the ADC control pins

ADC Control Bits	**AVR Port**
RA0 (Register Address)	Port C#1 (I/O)
RA1 (Register Address)	Port C#0 (I/O)
CS0 (Chip Select)	Port C#2 (I/O)
RD (Read)	Port C#5 (I/O)
WR (Write)	Port D#0 (I/O)
CS1 (Chip Select)	Port D#1 (I/O)

A D-type data latch IC with 3-state output (SN74AUC16373) is considered in designing the parallel data bus controller in order to hold the ADC output data, then transfer them to the parallel data bus as the schematic shows in Figure 4.5. This enables the DFC providing right data sequences for the USB interface that ultimately facilitates data transfer procedure by minimizing operation dependency of the USB interface with ADC data read process and therefore more flexible timing diagram is resulted. From this point view, the DFC can hold the data on parallel bus until the USB interface transfers the information to PC side, so in this case the THS10082 sets free for the next conversion.

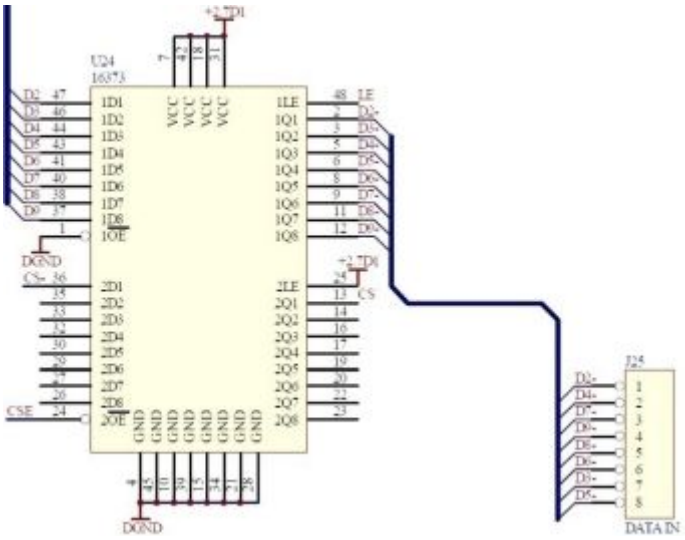


Figure 4.5. Schematic of data latch circuit

Indeed, the data bus is delay sensitive thus, the D-latch IC's propagation delay should be as low as possible. With respect to this issue, the AUC logic family has been selected to minimize the gate propagation delay effect. On the other hand the AUC logic family gates are optimized to be operational from 0.8v up to 2.7v, while the bus voltage-level is adjusted on 3.3v. To solve the incompatibility issue, the gate input voltage threshold should be also taken into account in order to prevent any random gates' behavior by receiving the input signal at USB interface below the high switching threshold (minimum 2.0v). That's why the SN74AUC16373 is powered up at its maximum tolerable voltage supply 2.7v. In this case, the SN74AUC16373 outputs' logic-high level (2.2v) is above high switching threshold at USB interface side (2.0v).

Latch-enable (LE) pin provides one more control bit for the DFC which can be used to freeze input data on the USB interface and this helps to control the data transfer procedure independently from ADC data output bus. In sum, the control bits connected to the DFC are listed in Table 4.4.

Table 4.4. Data bus control bits connected to the DFC

Pin Name	Device	Function
CS0	THS10082	Control bit read process from ADC
DATA_AV	THS10082	Data available strobe signal
LE	SN74AUC16373	Latch enable

4.3. ADC Unit Power Requirements

The ADC board contains four ICs that their operation depends on reliable and low noise power supply. Based on ADC board power requirements, two voltage regulators are locally included providing ultra-low noise, precise +5.0v and 2.7v. Since the SN74AUC16373 demands for 2.7v, it is locally provided because the power supply unit does not generate this voltage level. On the other hand the internal ADC analog circuitry should be supplied with precise and ultra-low noise +5.0v, thus the same voltage regulator (TPS7A4901) which is used for analog interface circuit, has been employed locally for this purpose.

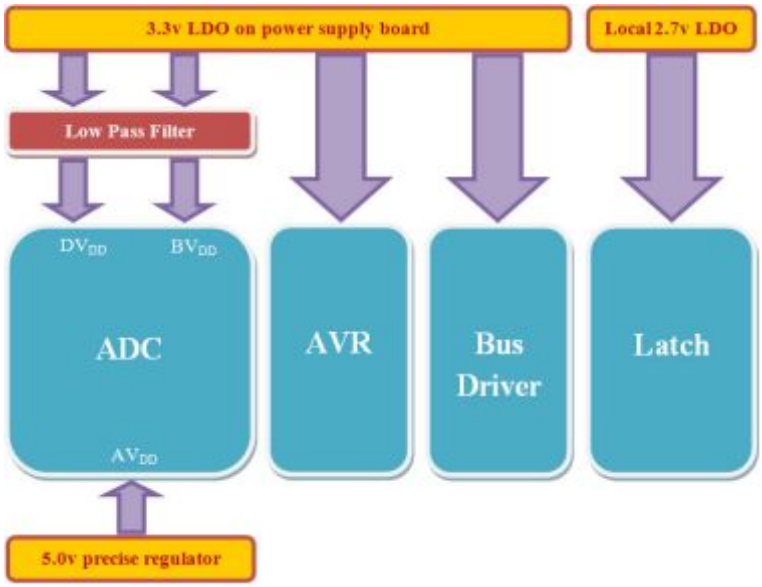


Figure 4.6. Block diagram of ADC unit power requirements

The block diagram in Figure 4.6 shows that the internal circuitry of the THS10082 demands for three distinct supply voltages. The power supply for the ADC output buffers (BV_{DD}) is separated from the internal digital circuitry (DV_{DD}) in order to make the ADC IC compatible with different voltage levels. This means that if 5.0v is dominant operating voltage in a circuit, the THS10082 can be easily integrated into the circuit without any voltage translation stage. In this project, 3.3v is considered as the main operating voltage in digital units, thus there is no need to feed the BV_{DD} with different voltage amplitude. That's why DV_{DD}, BV_{DD}, AVR and the SN74LVC244 are powered up from a single voltage regulator as it illustrated in Figure 4.6. Additionally, the LC low pass filters on DV_{DD} and BV_{DD} reduce generated noise on the shared power supply line. The ADC internal analog circuitry should be supplied with +5v, providing by high performance LDO (TPS7A4901) in order to maintain the accuracy of the conversion process. The TPS7A4901 is embedded on the ADC PCB layout and placed as close as possible to the THS10082, reducing the additive noise on power line.

4.4. ADC Initialization

Each single bit in control registers defines a specific function where are shortly described in Table 4.5 and Table 4.6. The THS10082 is highly integrated ADC IC with all required functions to perform analog to digital conversion. The THS10082 can be configured to operate as a differential ADC with internal voltage reference makes is suitable for using in spectrum-watching hardware. The integrated FIFO memory is fully configurable allows to ADC would be easily connected to a low-cost DSP processor for embedded digital signal processing. In addition, the ADC output could be initiated to represent the conversion result either in form of Binary or two's complement.

Table 4.5. The CR0 bit functions

Bit#	Function	Value	Selected Function
Bit 0	Internal/External voltage reference selector	0	Internal voltage reference
Bit 1	Continuous/Single conversion mode	0	Continuous conversion mode
Bit 2	Active/Power down mode	0	The ADC is active
Bit 3	Analog input channel selection	0	Both channels are selected
Bit 4	Analog input channel selection	0	Both channels are selected
Bit 5	Number of differential channels	1	Single differential channel
Bit 6	Number of differential channels	0	Single differential channel
Bit 7	Autoscan function	0	Disabled
Bit 8	Test mode selection	0	Normal operation
Bit 9	Test mode selection	0	Normal operation

Table 4.6. The CR1 bit functions

Bit#	Function	Value	Selected Function
Bit 0	Device reset	0	Normal Operation
Bit 1	FIFO reset	0	Normal Operation
Bit 2	FIFO trigger level	0	No buffering
Bit 3	FIFO trigger level	0	No buffering
Bit 4	DATA_AV signal mode	0	Static mode
Bit 5	DATA_AV signal polarity	1	Active high
Bit 6	Logic circuit mode: R/W or RD/WR	1	WR pin=R/W input, RD=disable
Bit 7	Output type: Complement/Binary	1	Binary format
Bit 8	Offset cancellation mode	0	Disabled
Bit 9	RESERVED	0	Always write 0

All the preceded functions should be initiated through the control registers prior to starting ADC operation. From this point of view, clarification of the design requirements and expected functionality from the THS10082 are the first steps in exploiting the IC performance. Before any further discussion on device performance and possible configurations, bit definitions of the CR0 and the CR1 are shown in Figure 4.7 along with desirable bit values.

Control Register 0

BIT 11	BIT 10	BIT 9	BIT 8	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
0	0	TEST1	TEST0	SCAN	DIFF1	DIFF0	CHSEL1	CHSEL0	PD	MODE	VREF
0	0	0	0	0	0	1	0	0	0	0	0

Control Register 1

BIT 11	BIT 10	BIT 9	BIT 8	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
0	1	RESERVED	OFFSET	BIN/2s	$R/\overline{W}$	DATA_P	DATA_T	TRIG1	TRIG0	FRST	RESET
0	1	0	0	1	1	1	0	0	0	0	0

Figure 4.7. Bit definitions of the CR0 and the CR1

The CR0 and the CR1 would be programmed by the ATMEGA8 at startup. Thus, the write process on the THS10082 should be also under control of the microcontroller. The ATMEGA8 sets the control bits (CS0, CS1, RD and W/R) in order to perform the write access, when the whole device is powered up. Then the initialization steps which are shown in flowchart of Figure 4.8 will be executed. After the setup phase, the ATMEGA8 reconfigures the control bits in order to prepare the THS10082 for starting the read access and data transfer process to the USB interface and then DFC unit takes the control of read access procedure.

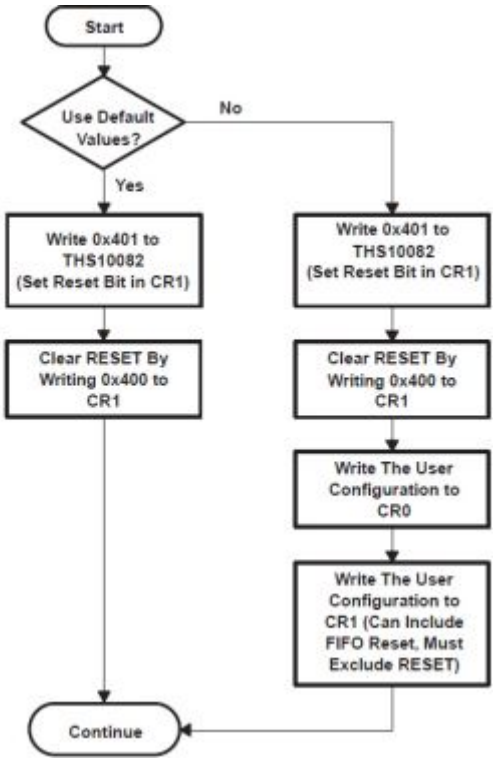


Figure 4.8. ADC initialization flowchart (Datasheet)

4.5. Internal Reference Voltage

The analog signal applying to the ADC input is swing limited, which means that peak-to-peak voltage range of input signal must be between the maximum value (V_{REFP}) and the minimum one (V_{REFM}). In other word, the analog signal amplitude equals to V_{REFP} and V_{REFM} would be interpreted as the maximum digital value (1023) and the minimum digital value (0) respectively. The reference voltage is used to define the V_{REFP} and V_{REFM} either from the internal source or an external reference voltage generator. Since the internal reference voltage is well-stable against temperature fluctuations (50 PPM/°C), it is considered in hardware design as the source of reference voltage which provides the voltage V_{REFM} of 1.5v and the voltage V_{REFP} of 3.5v so in this case the analog signal swing is 2.0v. The swing range might be increased by applying different reference voltage on V_{REFP} and V_{REFM} through an external source. In addition, using the internal reference voltage source requires a signal conditioning stage before the

THS10082, shifting the voltage level to 1.5v and also limiting the maximum signal amplitude. As it is already mentioned the THS10082 internally provides the required reference voltage, thus it can be activated by setting the first bit of CR0 to 0, and therefore the internal source would be automatically connected to the internal ADC block through the internal switches (see Figure 4.2). If everything goes fine, the respective reference voltage is measurable by a simple voltmeter on pins 25 and 26 of IC. Moreover, bypass capacitors should be externally attached to pins 25 and 26 in order to achieve stable reference voltage.

4.6. Analog Inputs Configuration

The THS10082 is suitable to convert either a differential analog signal or two single-end signals. This means that the two channels on THS10082 could be separately configured in order to feature a differential channel as it shown in Figure 4.9.

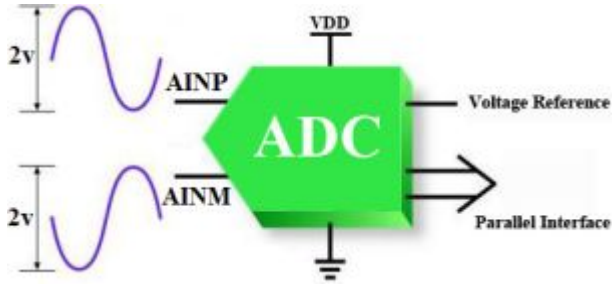


Figure 4.9. Differential ADC IC symbol

Best performance could also be achieved in differential mode in comparison to the single-ended configuration. In differential configuration the voltage which is applied at the input of the ADC (V_{ADC}) is the difference between the inputs AINP and AINM which can be calculated from Equation 3.1.

$$V_{ADC} = |AINP - AINM| \quad (3.1)$$

$$V_{CM} = \frac{AINP + AINM}{2} \quad (3.2)$$

In differential configuration the common-mode voltage (see Equation 3.2) can be rejected if the following two conditions are met, and this is the major advantage of using differential analog signals.

1) $AINM \geq GND$
2) $AINP \leq V_{DD}$

DC-offset rejection is another advantage of using differential signals in analog design. On the other hand in single-ended mode, 4 MSPS is the maximum achievable throughput rate, while in differential configuration this could be increased up to 8 MSPS. With respect to the all those preceded advantages, differential analog channel has been selected via bits 3 to 7 of CR0. On the other hand, autoscan function should be disabled by setting Bit 6 to zero, because a single differential channel is in use and there is no need to scan periodically two single-ended channels.

4.7. FIFO Trigger Level

The THS10082 exploits an integrated 16-word FIFO, facilitates an efficient connection to modern processors by temporary data storage in circular type memory. This means that the circular buffer could store up to 16 conversion values in order to significantly reduce the amount of interrupts to be served by the processor. In order to control the read process from FIFO or write process to FIFO, three pointers have been used. The first two pointers (read and write) are completely transparent to designer while the trigger pointer is fully configurable via control registers. The read pointer indicates the location where is read next and the write pointer always show the location where contains the last written sample. Whenever the buffer (FIFO) reaches to amount of data, defined through trigger level setting, an interrupt signal that is called DATA_AV (data

available) would be generated, showing the processor can start read access process. In other words, the trigger condition is satisfied when as many values as chosen for the trigger level are written into the FIFO. A proper configuration for trigger level could also be chosen which best fits the application. Since the PC interface has enough capacity to transfer information, there is no need to store data in a buffer thus, the trigger level should be set to one via bits 2 and 3 of CR1, which means that the THS10082 never uses the internal FIFO and therefore the information will directly be written on USB interface.

4.8. Data Available Interrupt Signal

The DFC is informed via DATA_AV interrupt signal that a block of data is ready to be served. The block size is always equal to trigger level. The DATA_AV signal is activated with rising edge of conversion clock (CONV_CLK) if the trigger pointer has been reached to the trigger level. The polarity of the DATA_AV interrupt signal is configurable via bit 5 of CR1, which means that the signal could be either active low or active high. In this project the signal is set to be active high. Moreover, the DATA_AV could be configured either in static mode or pulse mode. In pulse mode, the DATA_AV is activated with rising edge of the CONV_CLK, if trigger condition is satisfied. It again becomes inactive with next falling edge of the CONV_CLK. Inactivation timing diagram could be changed if the controller starts the first read before falling edge of the CONV_CLK, and thus the DATA_AV would be forced to become inactive earlier. On the other hand, the next DATA_AV activation will not occur again until the DFC reads the whole data block. In static mode, the DATA_AV becomes active with rising edge of the CONV_CLK and never becomes active again until the whole data block has been read from the FIFO. In this mode, the DATA_AV switches back to the inactive state as soon as the DFC starts the first read access as it shown in Figure 4.10. Since the static mode is simpler than the pulse mode, less-complex DFC requires to be implemented, hence the THS10082 is configured to operate in static mode via bit 4 of the CR1. Moreover, in static mode, the data block can rapidly be read from the FIFO and then more time will be left until next data availability.

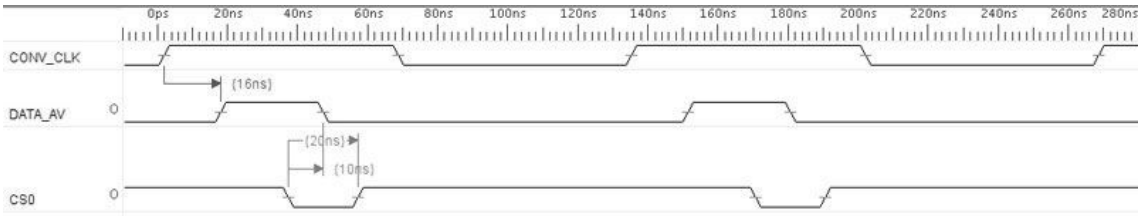


Figure 4.10. Read access timing diagram

4.9. ADC Test Function

Complex circuits in prototype level carries design bugs and implementation errors such as short circuits or unconnected PCB tracks where could be extremely difficult to detect. Since, electrical connections in parallel data bus are more complex than serial bus, higher possibility of failure should be expected. Therefore, in design assessment steps we have exploited an especial feature of the THS10082 providing test condition to verify the implementation. The ADC test function depicts three different known test conditions, allowing the designer to check all hardware connections as well as ADC operation. The ADC test modes are selectable via bits 8 and 9 of CR0 as listed below in Table 4.7.

Table 4.7. ADC test modes

Test Mode	Bit #9	Bit #8	10-Bit Digital Output	Equivalent 8-Bit Digital Output
Full	0	1	3FF	FF
Half	1	0	1FF	7F
Zero	1	1	0	0

4.10. ADC Emulator

The ADC unit itself along with its sub-systems has relatively a complex structure. This complexity may bring many uncertainties when it is connected to another complex system. Since, both the USB interface and the DFC in this project are designed from

scratch, it is extremely important to have a reliable ADC board which provides right data values; so any error or design bug can easier be detected in hardware side. On the other hand, the ADC part is designed and implemented in prototype level and its operation depends upon the USB interface. Since the ADC data throughput rate is 64 Mbps, it is difficult to monitor it as a standalone board. That's why as a test strategy, one block should be fixed and then the verification process would be applied on the rest. Hence, ADC emulator board has been designed in order to provide similar timing diagram and generating consistent responses to the DFC and USB interface commands similar to the actual ADC unit. This approach helps to verify the operation of the PC interface hardware. The ADC emulator is designed in two major parts. First, the timing circuitry (see Figure 4.11) that is responsible to generate the DATA_AV interrupt signal based on the CONV_CLK (8M in Figure 4.11) and CS0 signals. The Flip-Flop generates the DATA_AV on rising edge of the CONV_CLK, and then the DFC resets the DATA_AV signal to its initiate state by pulling down the CS0 control bit that is connect to the CLR pin of the Flip-Flop.

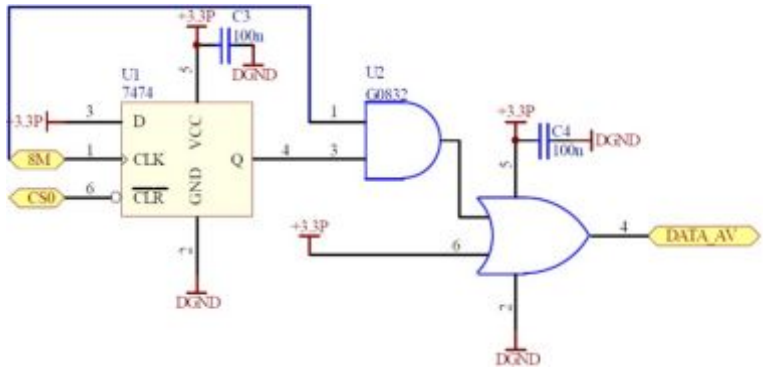


Figure 4.11. Timing circuit schematic of ADC emulator

Second block of the ADC emulator should generate a sort of numbers in ascending order so it is called sequence generator in this context. If the USB interface and DFC operate error-free, 256 numbers in ascending order should be displayed on computer screen. As the schematic in Figure 4.12 shows, two 4-bit synchronous binary counters (SN74LV163A) are employed in cascode configuration in order to generate 8 parallel

bits. Since the actual ADC IC converts the analog signal on falling edge of the CONV_CLK, the sequence generator should exactly perform the same function. Thus the NOT gate converts the falling edges of conversion clock to rising edges, and then the SN74LV163A ascends its output on falling edge of the CONV_CLK.

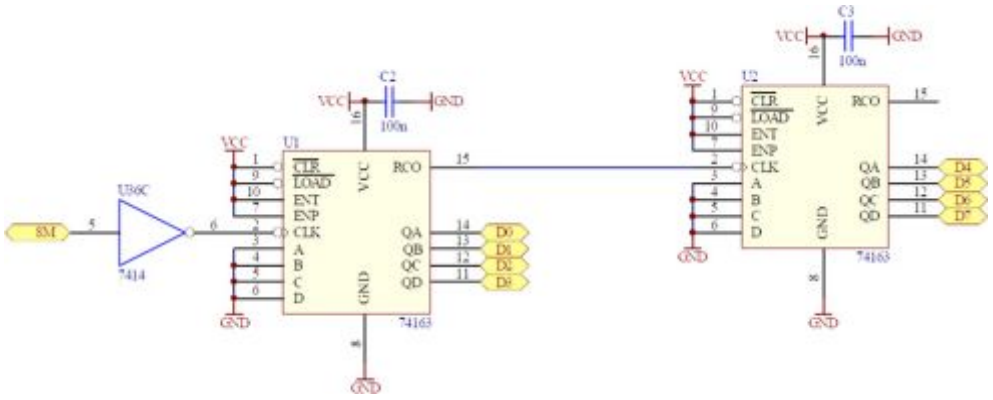


Figure 4.12. Sequence generator schematic of ADC emulator

The rising edge of input clock at CLK pin of the first SN74LV163A, leading to an increment at the IC's output in binary format and the operation will continue by RCO signal as the input clock of the second counter. The sequence generator and timing circuit implementation are shown in Figure 3.14.

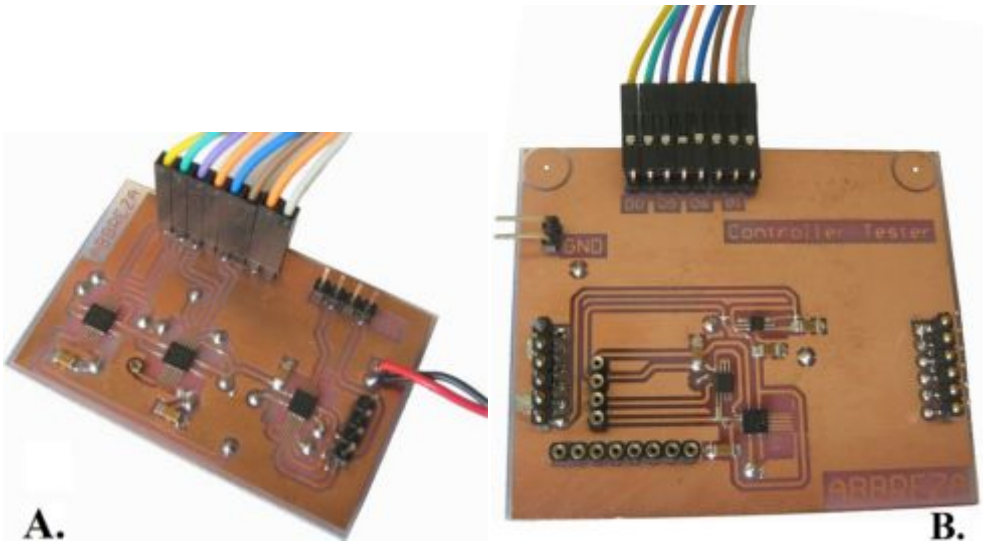


Figure 4.13. ADC emulator, A. Sequence generator, B. Timing module

5. USB INTERFACE

Data communication between a PC and external hardware requires a proper interface both in software and hardware sides, providing sufficient data bandwidth for a reliable data transfer. Moreover the PC communication port should be easy to use for end-user and being popular in variety of applications. The USB 2.0 has all those qualifications, requiring for a low cost spectrum-watching device. On the other hand, many IC manufacturers offer USB to serial/parallel-peripheral converters with different features and functionalities. Based on the system requirements and pre-study information, FT2232H is selected as a well-known USB controller IC which is commonly used in many implementations with good documentations and software support. This multipurpose USB controller has the capability of being configured in variety of industry standard serial or parallel interfaces. This enables the designer to include rapidly USB port to system architecture with minimum effort. Moreover, different library examples for PC side help developers to implement the software applications without any extra knowledge from hardware side. Furthermore, FT2232H has internal FIFO memory, fully integrated PLL and complete USB protocol stack in order to provide embedded hardware support for data transfer in High-speed mode (480 Mbps).

5.1. Direct Access to USB Port

Parallel to serial converter is firstly integrated in the FT245 USB controller chip. The IC accepts incoming data from the external device in parallel form and then converts them by shift registers and integrated FIFOs to serial form in order to make the data transferable via USB port. Since the internal FIFO provides a great technical specification for high-speed data transmission, the FT245 features high data rate up to 8 Mbps that is higher data throughput than conventional virtual COM port mode (2.4 Mbps). In other words, the USB driver of the IC conventionally opens a virtual serial port (COM port) for data communication in PC side with maximum 2.4 Mbps throughput, while the demand for higher data rate requires direct access to USB port. That’s why the new USB driver has been developed in order to exploit the full

capability of USB port. This is offered for the first time in FT245 chip, while the integrated FIFO and shift registers can support high data rate in hardware side. Thus, this mode of operation is named "*FT245 style synchronous FIFO mode*". The computer driver in this mode provides direct access to USB stack and therefore, full data rate could be expected to achieve from the USB port. The Figure 5.1 illustrates the direct access driver architecture in hierarchical layers, showing the great capability of using API in USB-based data communication. Since the FTD2xx dynamic library provides API access to USB stack, there is no need to have low level programming effort which is extremely time consuming process. Furthermore different programming languages such as Delphi, VB, VC++, VC# and Borland C++ are supported for developers in order to use the API in their applications.

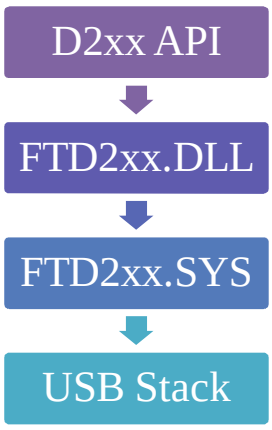


Figure 5.1. Direct access to USB stack via API

The new generation of the low cost USB controller **FT2232H** features high speed USB 2.0 data rate up to 480 Mbps in the *FT245 style synchronous FIFO mode*. In order to enter the mode, the application should open a USB port via the D2xx API and configure the USB IC in this mode. The USB IC should already be prepared for this mode of operation via the connected EEPROM which is designed to keep the user configurations (see section 5.4). If a device is configured to be operational in D2xx mode then no virtual COM port would be exposed for the device. Moreover, the FT2232H's *channel-A* is the only one which is configurable in the FT245 style synchronous FIFO mode;

therefore the *channel-B* is not available as all resources have been allocated to the channel-A.

5.2. Application Development

The flowchart in Figure 5.2 shows the sequences which should be followed by the PC application software in order to initiate the USB port via D2xx API and proceed to read operation. After opening a port the application software should firstly check for the device I/O *bit mode*, indicating the current mode of operation. Then the current user configurations will be read from the EEPROM and it should be reset to the right mode (Synchronous Bit Bang Mode). Once the FT2232 enters to this mode the 60MHz reference clock will be appeared on the CLKOUT pin.

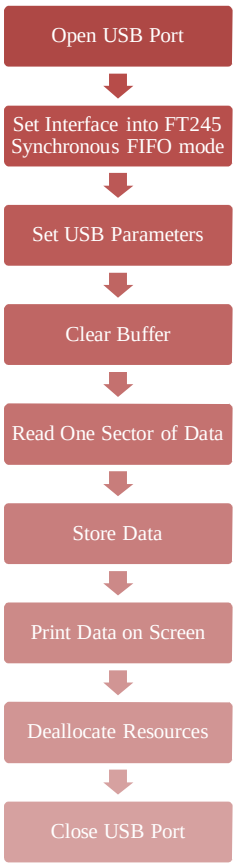


Figure 5.2. Access to a USB port in FT245 synchronous FIFO mode

As the next step in developing the application software, required USB buffer size should be set to a multiple of 64 Bytes between 64 Byte and 64 KB (see section 5.6) that the best transfer size (USB buffer size) depends on application. Since spectrum-watching should handle streaming data, large transfer size is more efficient. When the USB port is opened, the read request will be sent to the hardware and if the hardware would not be free to serve the request it will be rejected, thus the PC application should try again in a conditional loop until the hardware responds to the read request. Once the hardware starts to serve the request, one data sector will be read from hardware. The user should define the sector size prior to read operation at beginning. The codes are implemented in *Visual C++ 2008 Express Edition* by **VC++** programming language as a *Win32 Console Application*. Moreover, the executable file as well as source codes and USB driver files could be downloaded from the **project homepage**. It is wise to mention that, the whole software side is successfully implemented and tested on XP-Windows machine.

5.3. USB Timing Requirements

Since the USB 2.0 can support data rate up to 480 Mbps in high-speed mode, it needs an accurate timing and clocking sub-systems which meet the following timing diagram in Figure 5.3.

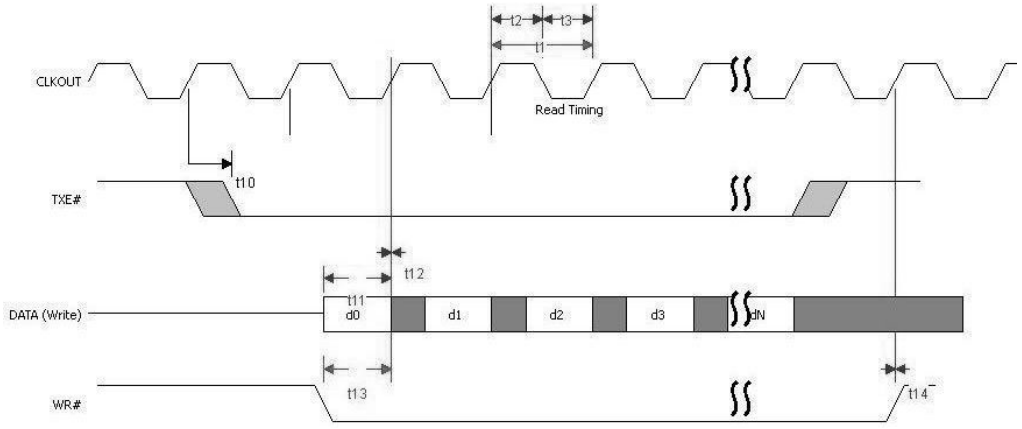


Figure 5.3. Read access to the FT2232 timing diagram (Datasheet)

When the FT2232H is configured through the external EEPROM to operate in single channel synchronous FIFO mode, the IC's pins start working in predefined functions as listed in Table 5.1. In this mode, when the external system is ready to transmit data, it should first monitor the TXE# pin and wait until the TXE# goes low, indicating the USB port is ready for transmission and then the information should be written into the internal FIFO on the rising edge of the CLKOUT—60 MHz reference clock. In this case, the WR# pin should then be pulled down before proceeding to send data bytes on rising edges of the reference clock. If the TXE# is pulled up, the DFC should immediately stop data transferring and pulls the WR# pin high.

Table 5.1. The FT2232 pins' functions in FT245 synchronous mode (Datasheet)

Channel A Pin No.	Name	Type	FT245 Configuration Description
24,23,22,21, 19,18,17,16	ADBUS[7:0]	I/O	D7 to D0 bidirectional FIFO data. This bus is normally input unless OE# is low.
26	RXF#	OUTPUT	When high, do not read data from the FIFO. When low, there is data available in the FIFO which can be read by driving RD# low. When in synchronous mode, data is transferred on every clock that RXF# and RD# are both low. Note that the OE# pin must be driven low at least 1 clock period before asserting RD# low.
27	TXE#	OUTPUT	When high, do not write data into the FIFO. When low, data can be written into the FIFO by driving WR# low. When in synchronous mode, data is transferred on every clock that TXE# and WR# are both low.
28	RD#	INPUT	Enables the current FIFO data byte to be driven onto D0...D7 when RD# goes low. The next FIFO data byte (if available) is fetched from the receive FIFO buffer each CLKOUT cycle until RD# goes high.
29	WR#	INPUT	Enables the data byte on the D0...D7 pins to be written into the transmit FIFO buffer when WR# is low. The next FIFO data byte is written to the transmit FIFO buffer each CLKOUT cycle until WR# goes high.
32	CLKOUT	OUTPUT	60 MHz Clock driven from the chip. All signals should be synchronized to this clock.
33	OE#	INPUT	Output enable when low to drive data onto D0-7. This should be driven low at least 1 clock period before driving RD# low to allow for data buffer turn-around.
30	SIWU	INPUT	The Send Immediate / WakeUp signal combines two functions on a single pin. If USB is in suspend mode (PWREN# = 1) and remote wakeup is enabled in the EEPROM , strobing this pin low will cause the device to request a resume on the USB Bus. Normally, this can be used to wake up the Host PC. During normal operation (PWREN# = 0), if this pin is strobed low any data in the device TX buffer will be sent out over USB on the next Bulk-IN request from the drivers regardless of the pending packet size. This can be used to optimize USB transfer speed for some applications. Tie this pin to VCCIO if not used.

5.4. Setting External EEPROM

Once the ready-to-use FT2232H-based USB hardware is connected to the USB port, the EEPROM settings can be programmed via either FTDIs MPROG or FT_PROG utilities (can be downloaded from the manufacturer's **website**). Thus, these utilities can be used to set the channel-A in the "*245 FIFO*" synchronous mode as it shown in Figure 5.4. Moreover, the item "*D2XX Direct*" should be selected as the USB driver for high-speed data communication. Hence, this changes data communication path from virtual serial port to direct access, which means that the highest data rate can be achieved by direct access to the USB port via D2XX driver. On the other hand, USB VID and PID codes might be changed through this utility, while several numbers of spectrum-watching USB devices are connected to a single PC. Furthermore, the product description and manufacturer name could be also changed for OEM production. In this project, the field manufacturer name and product description are filled with "*UWASA*" and "*Spectrum Watching*" respectively.

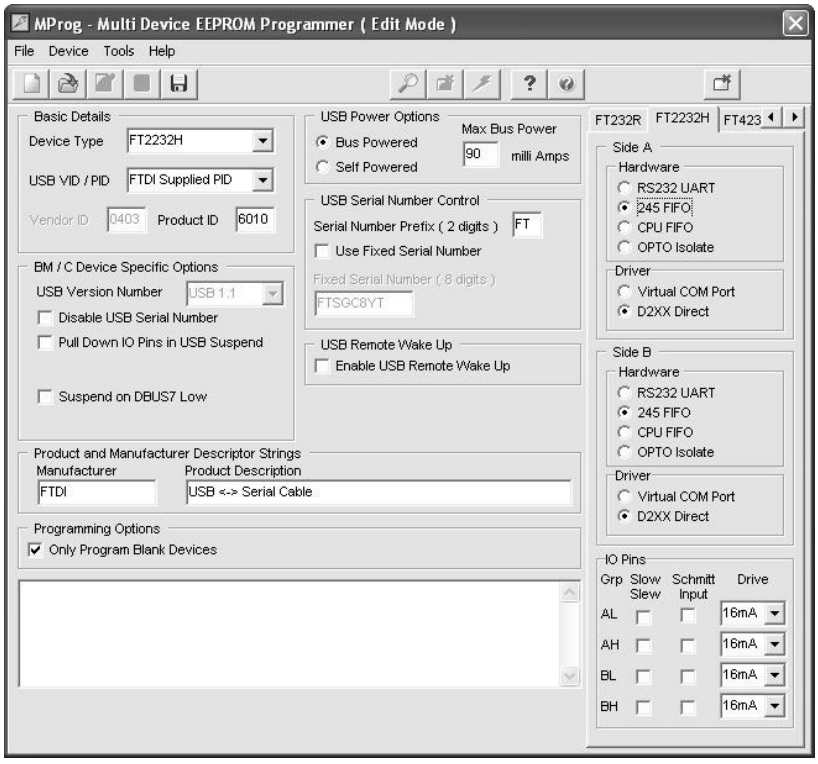


Figure 5.4. Required setting for EEPROM on MProg screenshot

FT2232H could be powered up either by an external power supplier or from the USB bus. In this project the USB IC is supplied from the USB port and its current consumption is limited to 100mA.

5.5. USB Data Latency

Data transfer over USB port is not interrupts-oriented whereas schedule-based system is used and consequently, there can be periods when the USB request does not get scheduled and, if handshaking is not employed, data loss will be experienced. Accordingly, the USB device transfers data in packets to PC, then a packet of data is received by USB scheduler who puts the request onto the list of tasks for the USB host controller to perform and then finally the device driver interprets received information for PC application software. This would typically take minimum one millisecond to execute because it will not pick up the new request until the next USB frame (the frame period is one millisecond). Indeed, there is a sizable overhead —depending on required throughput—associated with moving the data from the USB device to the application in order to mitigate delay problem otherwise if data were sent in solid frames in terms of certain bytes in a certain time by the application, this would severely limit the entire system throughput. Moreover, a latency timer is designed in order to avoid timing out condition. This along with USB buffer size affects on data throughput, thus it is important to set optimum values for latency timer as well as USB buffer.

5.6. Data Packet Size

Polling method is used in order to receive data from the USB port to PC. This means that a certain amount of data is requested by the device driver from the USB scheduler. This is done in multiples of 64 Bytes, while the packet size should be at most 64 bytes and always should be determined by user at software startup (see chapter 2). The USB controller will read data from the device until either the requested data length is reached or a packet shorter than 64 Bytes is received. This causes read access termination. On

the other hand the requested packet size from the device driver is limited from 64 Bytes up to 4 Kbytes because the size of the packet is proportional with system performance which is particularly dependent upon the data rate thus, the largest packet size means very high speed data transfer. For real-time applications such as voice transmission at e.g. 115200 Baud, the possible smallest packet size is desirable, otherwise the device will be holding up 4k of data at a time.

5.7. Practical Test Result of Data Acquisition Process

RF signal existence in a certain frequency band is the main objective of spectrum-watching implementation on cognitive radio. This means that the system should be able to detect only the existence of baseband signal in scanned RF range and thus there is no need to recover the signal. In this regard, the Nyquist–Shannon sampling theorem is not applicable anymore, but in order to assess the system functionality the Nyquist theorem is taken into account by selecting a pilot 50 KHz sinusoidal waveform as the input analog signal with 1v peak-to-peak amplitude as it shown in Figure 5.5. Since the frequency of the assumed input signal is 150 lower than sampling frequency (7.5 MHz), it can manifestly be recovered in software side and therefore it can demonstrate the system functionality by comparing the conversion result and input signal.

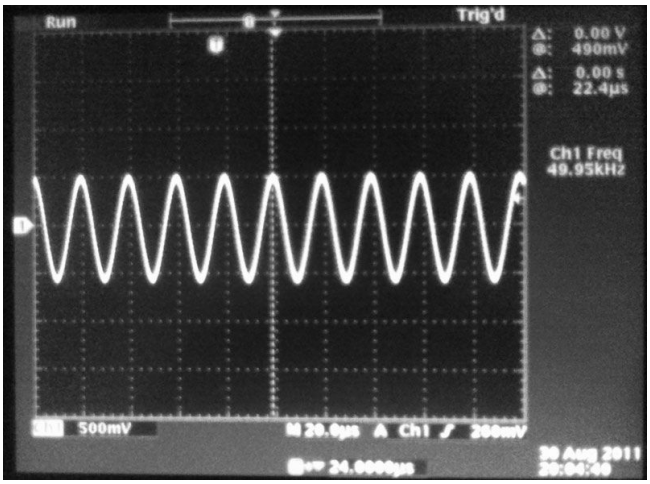


Figure 5.5. Pilot 50 KHz sinusoidal waveform captured by oscilloscope

As the Figure 5.6 shows, the converted input analog signal is reconstructed in MATLAB from the recorded digital bits. The hardware side has taken 800 samples from the input sinusoidal signal and the conversion result is stored in a dedicated text file (C:\spectrum.txt) where can be downloaded from the project **homepage**. The PC-based Win32 application software stores the information in this text file in form of decimal numbers (0-255), representing the digital samples.

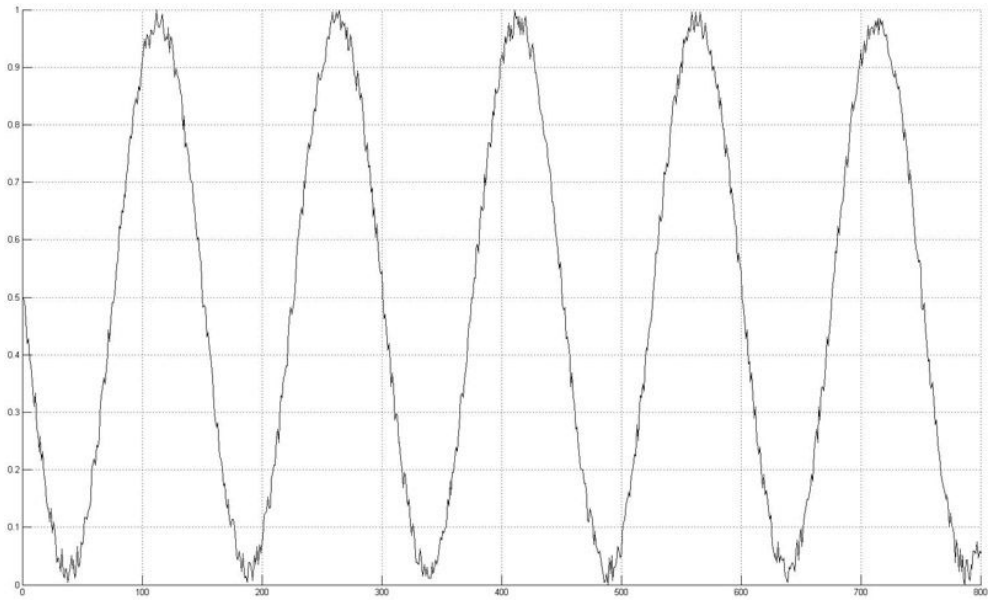


Figure 5.6. Reconstructed input signal from digital samples

The Comparison between the actual captured waveform and the result of conversion reconstructed by MATLAB, indicates successful and consistent process. This means that the designed hardware along with software side perform front-end spectrum watching process.

6. CLOCK MANAGEMENT AND DISTRIBUTION

Clock management and distribution is the crucial part in this project while ADC unit and USB interface require clock source for their operation in different frequencies. Clock source selection, distribution strategy and clock conversion will be under concentration in the following discussion.

6.1. Preliminary Clock Management Unit

Figure 6.1 shows the existing blocks in conventional high speed data acquisition system, demand for clock source. Although all these units require different clock speeds but they must be synchronized with a single clock line, that's why the clock management unit is considered to take this responsibility. On the other hand, the final design in this project has been simplified for cost reduction and thus the FIFO, SRAM and DSP modules which are illustrated in Figure 6.1 are dismissed. Hence the ADC and USB interface are the only ones demand clock source.

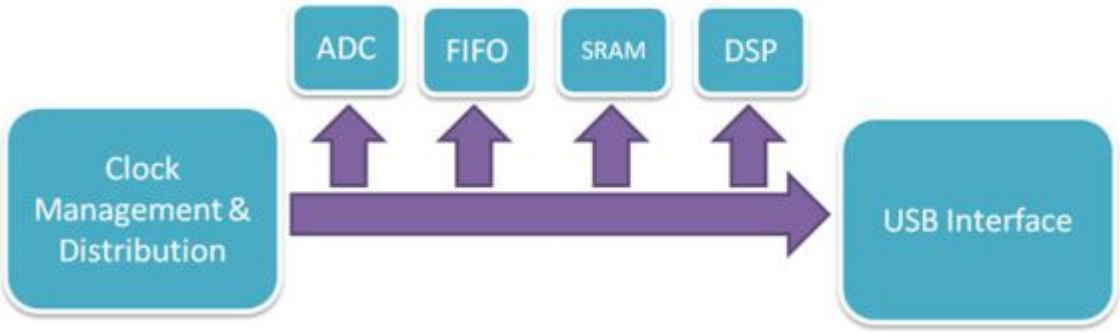


Figure 6.1. Clock requirements for internal blocks of a spectrum-watching hardware

As the first design approach, the CDCE937L which is a programmable 3-PLL clock synthesizer is employed in demonstration board (see chapter 3) in order to evaluate its

functionality for this application. The key features of CDCE937L are taken into account as follows:

- Three independent PLLs and 7 independent clock outputs.
- In-system programmability and EEPROM.
- On-chip VCXO: Pull Range ±150 ppm.
- Selectable output frequency up to 230 MHz.
- Low-noise PLL core.
- Integrated PLL loop filter components.

Indeed, the programmability characteristic of CDCE937L is the main reason for this selection, because all its outputs can be independently programmed in different frequencies via I2C bus either by embedded processor or PC-based graphical application software. This enables the designer to have more degree-of-freedoms for the rest of design process while the big part of the project outline is still uncertain. In order to exploit the CDCE937L programmability which is so essential for integration of this block into the whole design, implementation of the PC-based I2C interface is the first step to be preceded; therefore a well-known USB to I2C converter module (see Figure 6.2) is used in order to accelerate the design procedure. Since, the converter module is perfectly compatible with Devantech sensor modules such as ultrasonic proximity detectors (SRF08) and digital compass (CMPS03), it comes with open-source monitoring software which translates the user's commands to I2C format and then send it through virtual COM port.

Figure 6.2. USB to I2C converter module (Datasheet)

While the CDCE937L supports Byte Write/Read operations, the PC can individually get access to addressed bytes. This requires data framing and complex computation in software side in order to correctly set the device registers, thus the easiest way to achieve this objective is using the standard graphical user interface software which is designed by Texas-instruments for this purpose. Figure 6.3 shows a screenshot from TI ClockProg's main menu. This application can be easily used for configuring the device internal registers by end-user. The internal block diagram of the CDCE937L which is illustrated in Figure 6.4, shows many internal functions due to the device capability in forming independently a desirable clock frequency on each output pin. The internal VCXO can generate a base-clock frequency from a single external crystal and feeds the rest of circuitries including the internal PLLs as well as I2C bus driver. As it shown in Figure 6.4 user configurable control terminals (S0, S1 and S2) can be used either for serial data communication or handling the user predefined functions such as spread spectrum clocking selection, frequency selection and output state selection. Hence an external host controller can immediately change the output frequency by setting different binary values on the control terminals.

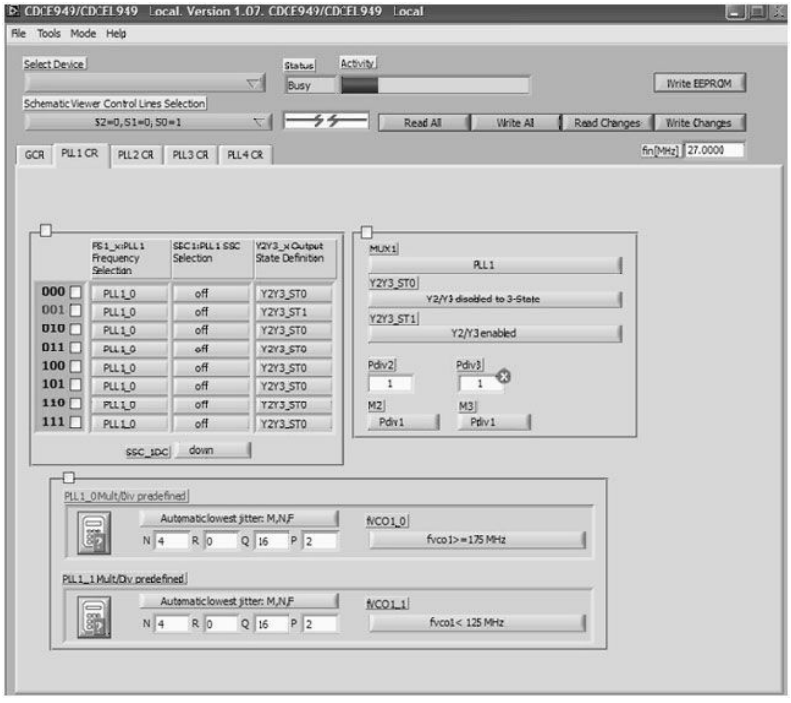


Figure 6.3. Screenshot from the TI ClockProg main menu

Control terminals have desirable functionality for spectrum-watching application, while data reading/writing frequency on FIFO memory should be variable in different conditions, (e.g. in congestion situation that writing access frequency on PC interface must be reduced and more room should be occupied from buffer to store data). On the other hand, any clock frequency can be generated by the combination of PLLs and frequency dividers at the device outputs based on user's configurations which are digitally set through ClockProg application software and then the integrated CDCE937L's EEPROM saves those configurations in order to prevent losing data in power-down situation, thus there is no need to reconfigure the IC again at startup again.

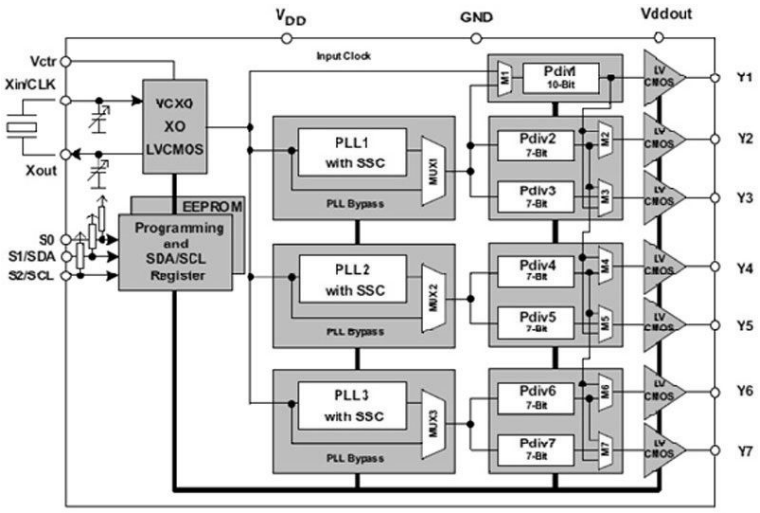


Figure 6.4. The CDCE937L internal block diagram (Datasheet)

Moreover, the internal LVCMOS buffers are utilized on device outputs which are driven by separate power pin (Vdd_{out}). This provides voltage-level compatibility between the parts connected to the CDCE937L with different operating voltage-level. In other words, since the CDCE937L powers up with 1.8v, it is not compatible with 2.7/3.3v standards so Vdd_{out} allows the designer to apply a right voltage-level to the LVCMOS buffers in order to eliminate the need for using voltage-level translator ICs.

Based on the factory default settings of the device EEPROM, the crystal frequency would be appeared on device outputs once it powers up, which means that the internal

PLLs and frequency dividers' ratio are set on one. This allows the CDCE937L operates normally without any programming effort. It also provides testing environment in prototype level when the circuit should be evaluated and thus a known clock frequency on the device outputs equal to the crystal frequency verifies the design and implementation.

Enabling the output buffers by setting the S0 control pin on high logic level is a key point while the CDCE937L is in test condition, because this activates the device outputs and clock signal can be measured by instruments (e.g. oscilloscope or frequency meter) otherwise the device outputs remain in high-impedance state.

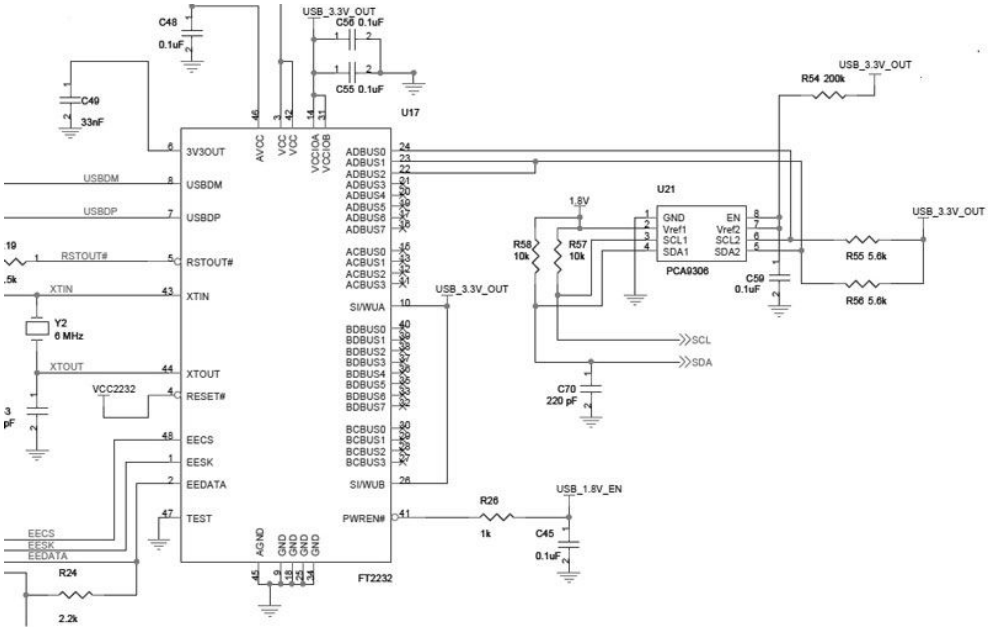


Figure 6.5. USB programmer for CDCE937L clock synthesizer

The CDCE937L's programmer is the crucial part that should be fully functional in order to exploit whole benefits of using the CDCE937L. As it mentioned before a USB-to-I2C converter module had been used for this purpose and while it could not be integrated easily with the whole design another approach is adopted. Based on the information provided in the technical datasheet of CDCE9xx evaluation board, the FT2232 USB interface IC might be used as a PC interface in the device programmer.

Because the FT2232 can be configured in *MPSSE mode* in order to provide a USB-based I2C bus controller. On the other hand, the ClockProg is fully compatible with MPSSE-based driver, thus there is no need to modify the software side, but since the I2C interface of the CDCE937L operates with 1.8v and the FT2232 supports only 3.3v standard, a voltage-level translator circuit should be employed. This is a challenging issue because it cannot be simply solved by using a resistive voltage divider while the SDA line is a bi-directional data line. This means that the voltage-level could easily be reduced from 3.3v to 1.8v by a resistive voltage-divider when data flow direction is from the FT2232 to the CDCE937L but in reverse direction the 1.8v level could not simultaneously be increased up to 3.3v while the signal is in the same path. In sum, resistive voltage-divider is not an applicable solution to handle data flow from the CDCE937L to the FT2232 and vice versa. Hence, an active circuit (PCA9306) should be used as it shown in Figure 6.5. The PCA9306 is a bidirectional I2C and SMBus voltage-level translator, which is well-suited for this purpose.

6.2. I2C Timing Problem

The CDCE937L's EEPROM programming experiments revealed, some practical matters that should be taken into account. Firstly, the combination of the FT2232 and the PCA9306 should be placed as near as possible to the CDCE937L because delay issues may cause problem in synchronously receiving the data packets. In other words, there is a possibility that delay between the SCL and SDA is not compliant or too marginal due to different signals' paths on the circuit board. In this case, unwanted STOP or START bits can be happened during a message transfer, causing the client device (slave) to misleadingly interpret the message or resulting in a missing acknowledgment (ACK). Indeed, this problem has been occurred in project implementation due to the far distance (20cm wiring) between the PCA9306 and the CDCE937L, and consequently a lot of efforts have been made in order to find the source of error. With respect to this issue

6.3. Pull-up Resistor Selection

The output of the I2C interface is designed open-drain/collector in order to prevent data collusion by performing a wire-AND operation. This smart way allows the master device to control the bus status by pulling it down. In other words, when the SDA line is low, it would be interpreted as busy status by slave devices and also, this configuration allows the master device to check the bus status. If the master pulls the SDA line high but the line stays low (while some clients may pull the line low) then it indicates that the bus is still busy and if it goes high, then it means that the bus is free. On the other hand, the open-drain/collector outputs should be pulled up by resistors and thus the value of those resistors must be carefully defined, depending on several factors as follows:

- Bus speed (standard/fast)
- Maximum current load (I_{max})
- Bus voltage level (V_{dd})
- Bus capacitance

The minimum resistor value (R_{min}) could be calculated by Equation 6.1, where V_{olmax} represents the maximum logical low voltage-level.

$$R_{\min} = \left(\frac{V_{dd} - V_{ol\max}}{I_{\max}} \right) \qquad (6.1)$$

The pull-up resistor for the SDA and SCL is down-limited by the I_{max} and the static load specifications which are definable based on the I2C specifications. In both *standard and fast mode*, I_{max} should be limited to 3mA while 20mA is recommended for the *fast mode plus*. Since the bus speed in this project is 100 KHz (based on the standard mode), the I_{max} should be 3mA. Moreover, the V_{olmax} is considered 0.4v with respect to the PCA9306 and the CDCE937L switching level specifications. Furthermore, the R_{min} should distinctly be calculated for different bus voltage-levels. Hence, 466Ω and 966Ω are defined as the minimum values of pull-up resistors for 1.8v and 3.3v buses respectively.

The resistor is also up-limited by the rising and falling edge specification of signal. From this point of view, the bus capacitance (C) has the main effect on selecting maximum value. Equation 6.2 is used to find the time spent (rise time) while the voltage level goes from 30% up to 70% of $V_{dd.}$

$$t = t_2 - t_1$$

$$\therefore V(t_1) = 0.3 \times V_{dd} = V_{dd} \times \left(1 - e^{\left(\frac{-t_1}{R_{Max}C}\right)}\right) \qquad (6.2)$$

$$\therefore V(t_2) = 0.7 \times V_{dd} = V_{dd} \times \left(1 - e^{\left(\frac{-t_2}{R_{Max}C}\right)}\right)$$

By further calculation and simplifications the following result in Equation 6.3 has been achieved, showing a linear dependency between rise time (t), bus capacitance and maximum resistor value.

$$t = 0.8473 \times R_{\max} \times C \qquad (6.3)$$

In I2C standard mode rise time is considered 1us, and thus:

$$R_{\max} \times C = 1180ns$$

Moreover, the rise time in fast-mode is 300ns, resulting $R_{max}C$=354 ns. Hence the R_{max} for the standard-mode and the fast-mode is 2.95 KΩ and 885Ω respectively, while the bus capacitance is assumed 400pF. In sum, the pull-up resistor value for a 1.8v I2C bus in standard-mode could be chosen from 466Ω up to 2.95 KΩ, while the minimum value for 3.3v I2C bus is 966Ω and the maximum value is as same as the value for 1.8v bus. Hence, 1.2 KΩ is selected to be used in both sides (1.8v and 3.3v) and practically it works with no problem. Conventionally, 10 KΩ pull-up resistors are employed in I2C buses while it is mostly applicable in 5v bus and using it for 3.3v/1.8v buses would be

led to occurrence of unwanted STOP or START bits during data transmission. Indeed, this is happened in the first prototype of the CDCE937L programmer due to using of 10 KΩ pull-up resistors, but the I2C communication problem has effectively been solved by replacing the resistors with the right value (1.2 KΩ).

6.4. ClockPro Compatibility

Two distinct versions of *ClockPro* have been released by TI. One is designed based on *NI Labview* graphical programming software and the other version is designed based on.*NET* programming environment. Currently TI supports Labview-based ClockPro, that's why the same version is preferred in this project. On the other hand, the new version "*Rev. B*" is just compatible with TI USB controller IC (TUSB3410) which is not used in this project, but the older one "*Rev. A*" perfectly works with the FT2232 USB interface, and therefore the version Rev. A has been selected.

The USB devices are identified by their PID and VID codes which are subject to change in order to support multiple USB connections to a single host controller. Indeed, the USB-based programmer of the CDCE937L should have a specific PID code to be detectable by the ClockPro and this is a key point that should be taken into account otherwise the programmer does not work anymore. Actually, a lot of effort has been made in order to understand where the problem is and what the right PID code should be defined for the programmer. Finally, the PID code (C5A8) of the *TI ADS5545 evaluation board* (ADC-EVM) is applied and then the ClockPro could recognize the programmer.

6.5. Clock Source

Based on the design outline, everything should be synchronized with the 60 MHz reference clock, generating by USB interface. The reference clock will be appeared on the pin 32 of the FT2232 while, it has been entered into the synchronous FIFO mode.

This requires EEPROM configuration prior to any access from PC side via the application software. Observing the waveform with an oscilloscope has been performed in order to verify the signal quality as it shown in Figure 6.6.

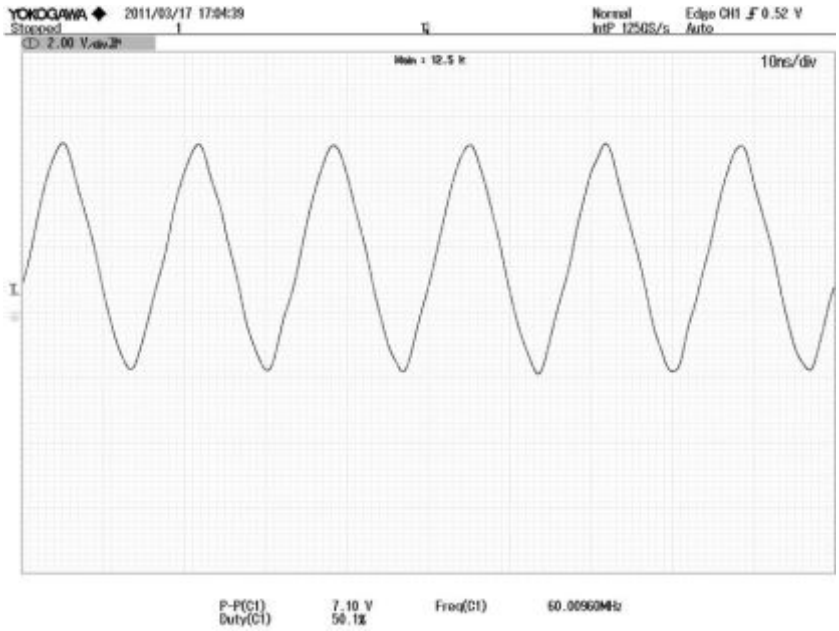


Figure 6.6. The 60 MHz reference clock waveform

The signal edges are not sharp enough thus it is almost similar to a sinusoidal signal while it should be rectangle type. With respect to this issue, current source has been increased on the respective IO pin (pin 32) by altering the FT2232 configurations via MProg utility, leading to sharper waveform as the result is shown in Figure 6.7. Since the reference clock frequency is higher than hundreds of KHz, coaxial cable should be used as transmission line, that's why SMA connectors and coaxial cables are utilized on the clock management board to transfer the clock signal to the target boards in order to prevent signal degradation.

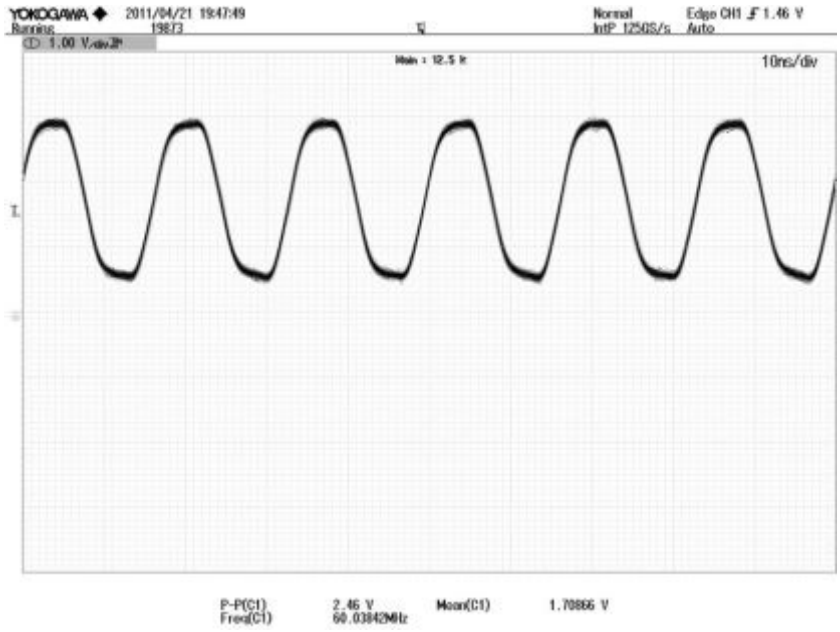


Figure 6.7. Enhanced 60 MHz reference clock waveform

6.6. Clock Divider

Sub-systems of a spectrum-watching device would demand different clock frequencies for their operation. This means that, the reference clock should be adjusted based on the clock requirements by clock management and distribution unit. As it discussed in previous sections, the CDCE937L is well-suited as the clock management of spectrum-watching in terms of flexibility and programmability, but its output should be synchronized by the reference clock. Since the CDCE937L does not meet zero-delay property and is not designed to be synchronized with an external clock, thus as a new approach the required clock frequencies are generated by discrete units with minimum propagation delay. Accordingly, clock divider ICs are the main concentration in clock adaptation procedure while the required clock frequencies for the ADC part and DFC unit are lower than 60 MHz. On the other hand if any digital unit operates in higher frequency than reference clock, using a PLL is essential.

After design simplification and cost reduction, there is no digital unit operates with higher frequency than 60 MHz, thus the frequency dividers are considered as the main

part of clock management. From this point of view employing a high performance, low-jitter, low propagation delay frequency divider IC is the main objective. Thus, programmable clock divider (CDCM1802 and CDCM1804) with very low propagation delay (600-900 ps) has been selected for evaluation. The CDCM1802 distributes the differential clock input to a single LVPECL and LVCMOS output, while the CDCM1804 provides one LVCMOS output and three LVPECL outputs. Both the CDCM1802 and CDCM1804 have almost the same performance and are characterized for industrial applications as their internal block diagrams are comparable in Figure 6.8. The clock dividers accept differential clock input up to 800 MHz and generate outputs in form of LVCMOS and LVPECL.

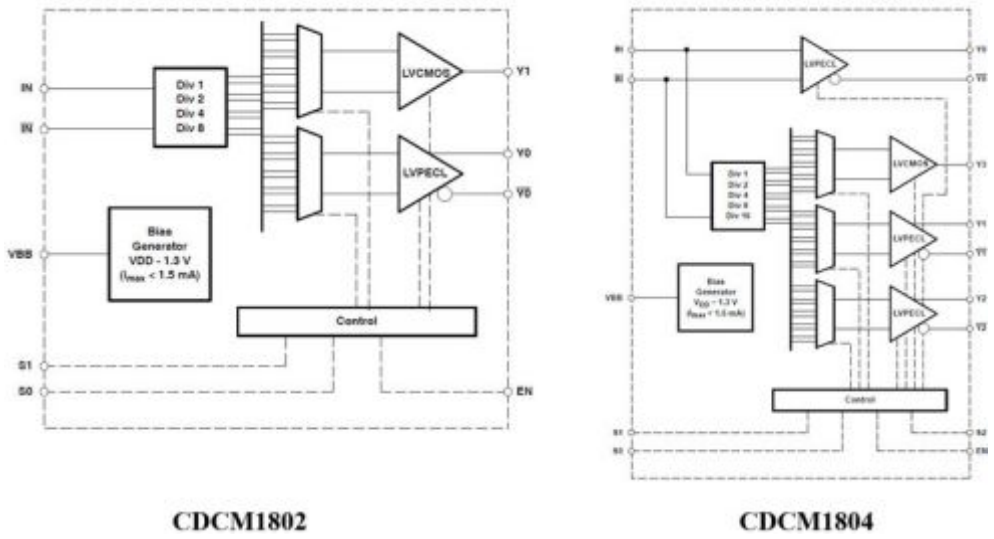


Figure 6.8. Internal block diagrams of CDCM1802 and CDCM1804 (Datasheet)

As the first design step, an evaluation module is implemented for testing the CDCM1802 performance in order to employ it in clock management unit. The module which is shown in Figure 6.9 receives single-ended clock signal and divides it by ratio from 1 up to 8. This means that 60 MHz reference clock could be converted to 7.5 MHz clock frequency that is suitable for clocking the ADC part.

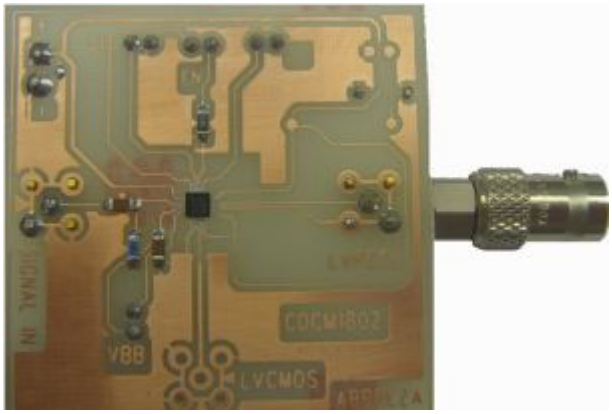


Figure 6.9. The evaluation module of the CDCM1802 clock divider

The input of the CDCM1802 is characterized with very large CM voltage range and has high impedance. It is originally designed to receive differential signals but since the reference clock from USB interface is single-ended, the CDCM1802's input should be adapted to accept it. Therefore, the unused input of the CDCM1802 is biased with CM voltage and thus it is connected directly to the internal voltage reference (2V) as it shown in Figure 6.10. This sets a threshold for inverting input of internal differential amplifier and consequently it operates as a typical voltage comparator, which means that the CDCM1802 is now compatible to accept single-ended clock at its input.

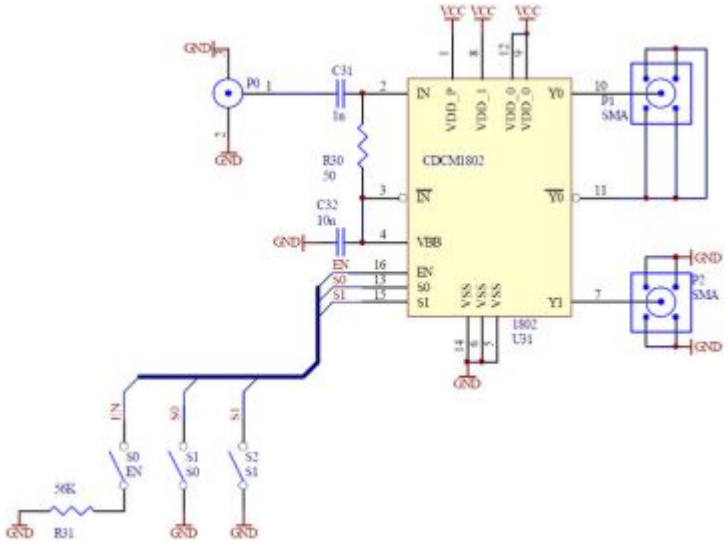


Figure 6.10. Schematic of clock divider evaluation module based on CDCM1802

Moreover the signal duty cycle must be remained on 50%, thus the internal DC biased signal (VBB) is employed to pull up the non-inverting input through a 50Ω resistor (R30). It is important to mention that 10nF ceramic capacitor should be externally connected to voltage reference for stabilization issue.

The CDCM1802 has three control pins (EN, S0 and S1) that can be set on 3-levels in order to configure the IC on proper division ratio. All those control pins are internally pulled up by 60 KΩ resistors, thus if they would be left unconnected, logical high will be assumed. Since we are interested to generate 8 MHz sampling clock from reference clock for ADC IC, division ratio should be set to 8. In this case the output waveform on the LVCMOS output is shown in Figure 6.11.

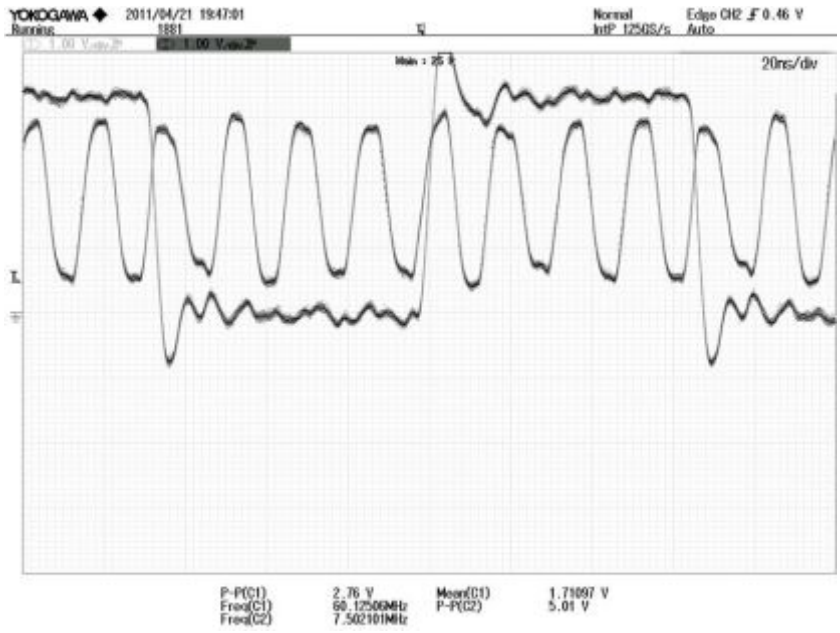


Figure 6.11. The LVCMOS output of the CDCM1802

The generated clock frequency for the ADC at the CDCM1802 output is 7.5 MHz and it is slightly lower than the required maximum conversion clock frequency which is 8 MHz. This is not a problem anymore while the ADC IC could be operational at even 1 MHz. In some applications, lower sampling frequency requires, thus CDCM1804 could be a better choice in terms of division ratio. By using the CDCM1804, the input signal

can be divided by 16, which means that lower sampling frequency 3.25 MHz can be derived from the 60 MHz reference clock. With respect to this issue an evaluation module has been prototyped (see Figure 6.12) to assess the CDCM1804 performance.

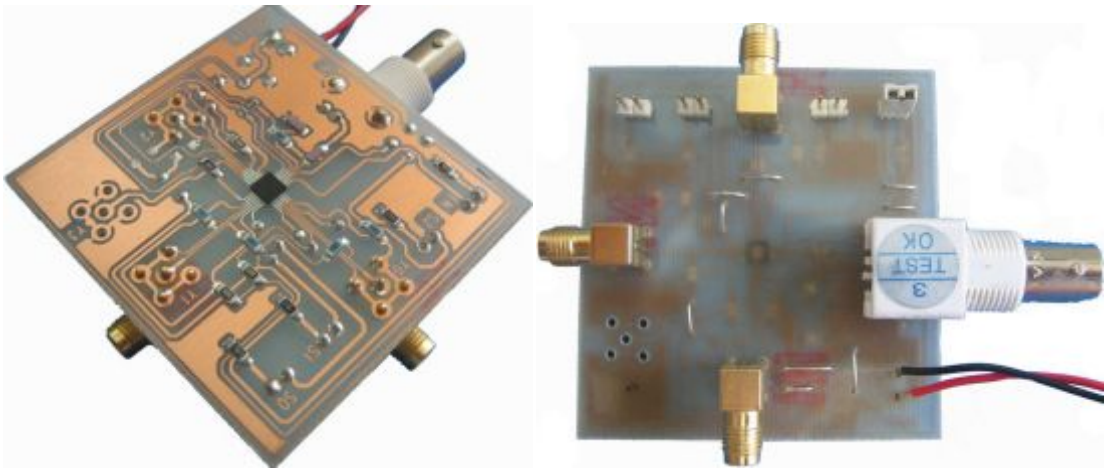


Figure 6.12. The evaluation module of the CDCM1802 clock divider

The reference clock is injected from a BNC connector to the module input and the divided clock is captured from the SMA connectors where are connected to an oscilloscope. Since delay propagation issue is too critical in timing diagram, it should be measured and compared with the values that are declared in the CDCM1804's technical datasheet. It is mentioned that the front-end propagation delay from the device input to the LVPECL output is 600-900ps, while the LVCMOS output is purposefully delayed by 1.6 ns over the three LVPECL output stages to reduce noise impact during signal transitions, which means that the overall propagation delay from the input to the LVCMOS output is almost 2.5 ns. This is measured by oscilloscope and the result is illustrated in Figure 6.13 which demonstrates almost 2 ns propagation delay.

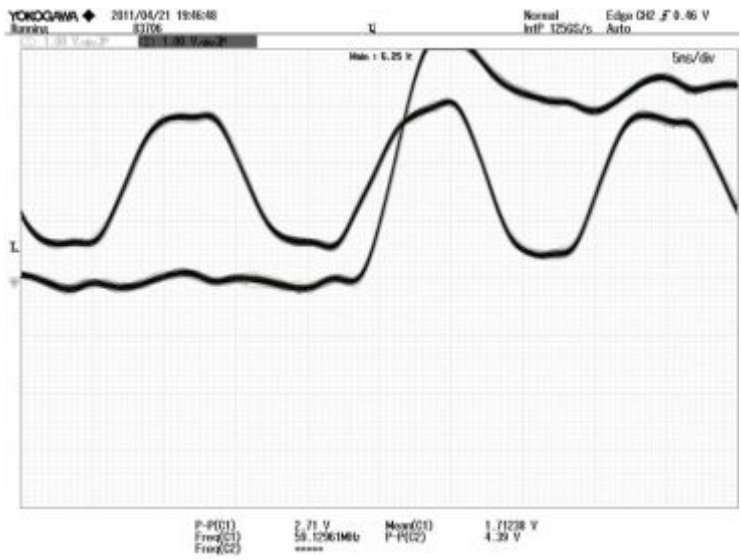


Figure 6.13. The LVCMOS output of CDCM1804 and 60MHz reference clock

Achieving minimum propagation delay is the first objective in designing the clock divider circuitry so the LVPECL outputs with 600ps delay are more attractive in compare with the LVCMOS output in terms of delay. But the ADC IC is only compatible with the single-ended clock signal, that's why LVPECL to LVCMOS translator circuit should be considered hence, the SN65EPT21 is selected as a PECL-to-TTL translator IC with maximum 1 ns propagation delay and therefore, the overall propagation delay should be at most 1.9 ns. The design schematic illustrated in Figure 6.14 is implemented in order to measure the propagation delay by an accurate oscilloscope.

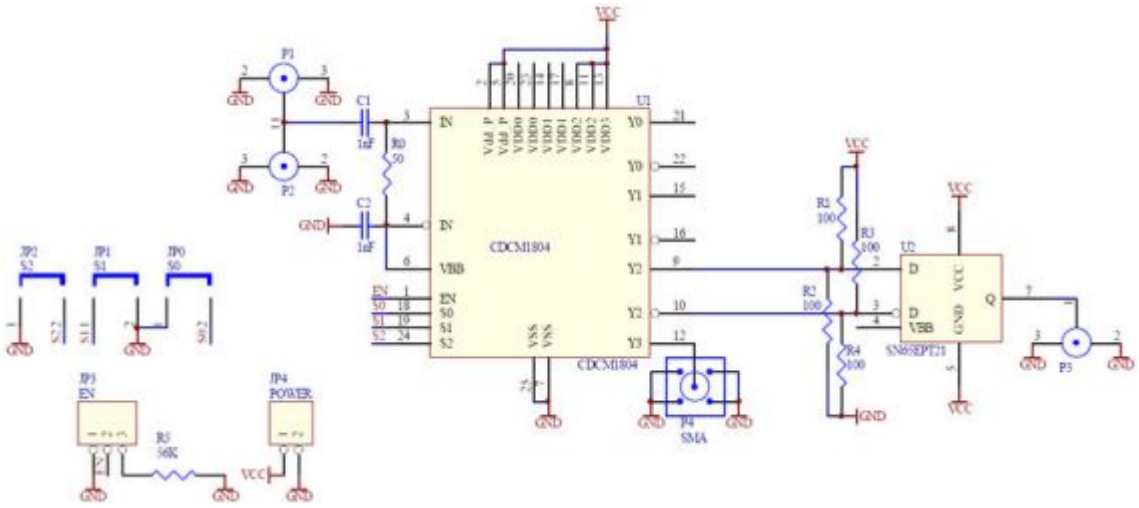


Figure 6.14. Schematic of CDCM1804 along with voltage-level translator circuit

Since the CDCM1804 is connected to the SN65EPT21 via a differential line, DC-coupling should be taken into account, so the ICs could be connected together without any AC coupling capacitors. DC-coupling on differential lines has several advantages such as no dc-wander issues and simple board design. The 50-Ω LVPECL communication lines are typically terminated to *VCC-2v*, but mostly the electrical circuits have difficulty to support dual power supplies of 3.3 V and 1.3 V, that's why using a simple voltage divider, is another alternative method to terminate the LVPECL by establishing the CM voltage of 2v at the receiver. In other words, the pull-up and pull-down combination of resistors (see R1, R2, R3 and R4 in Figure 6.14) terminates the 50-Ω transmission line. Despite the design complexity is increased due to using the LVPECL-to-LVCMOS translator circuit, no improvement is observed in the waveform which is shown in Figure 5.16. That's why the CDCM1802/CDCM1804's LVCMOS output is finally considered as the output signal for clocking the ADC IC, leading less expensive and less complex hardware. This means that the LVCMOS output would be directly connected to ADC clock input without any interface so in this case, 2.2 ns propagation delay is added to the ADC conversion clock which must be compensated by data flow controller.

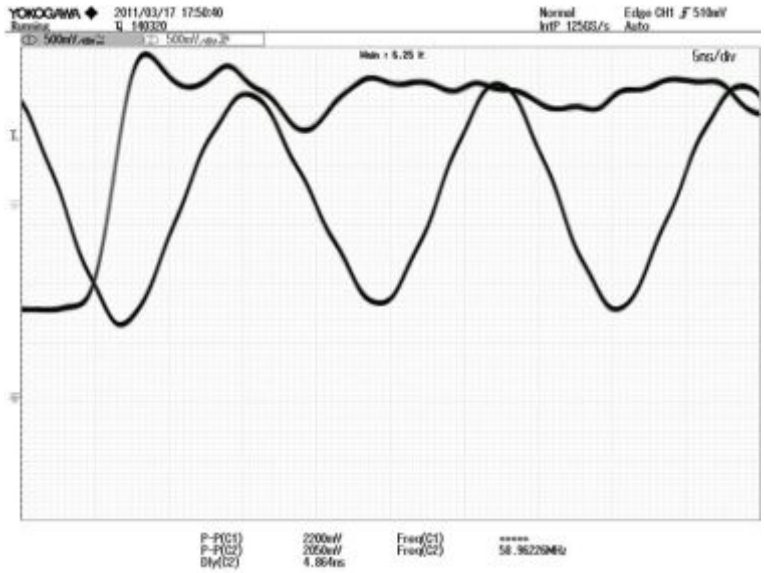


Figure 6.15. 7.5MHz output signal of SN65EPT21 and 60MHz reference clock

Apart from selecting the CDCM1802/1804 clock dividers as the final choice, in component selection phase many suitable frequency dividers as well as clock multipliers were taken into account, but eventually a few of them has been qualified for further study based on their performance and specifications, thus it is worthwhile to have all those candidates for further system development in future as they are listed in Table 6.1.

Table 6.1. Clock management ICs

#	Part Number	Description	Key Features
1	CDC2582	3.3V PLL Clock Divider	Zero-Phase Delay
2	CDCE421A	Low-Jitter Crystal Oscillator Clock Generator	Programmable PLL+Clock Divider
3	CDFC5801	Clock Multiplier with Delay control	Phase Alignment Contol
4	CDCM61001	Integrated VCO, Low-Jitter Clock Generator	Programmable Clock Divider+PLL

7. DATA FLOW CONTROLLER

High speed data acquisition concept mainly involved with data aggregation issue, thus data storage and management unit should be taken into consideration, avoiding any data congestion. This requires high speed DSP and costly FPGA-based logic circuit as well as expensive FIFO and SRAM memories adding more system complexity and higher BOM cost, thus the major project objective that is a cost-effective design, could not be achieved. Accordingly, after many trials and simplifying various designs, the final design of DFC unit is considered as the most cost effective one which is implemented in gate level without employing any digital processor or expensive SRAM/FIFO memories.

7.1. Design Challenges

The DFC unit design for high-speed data acquisition involves two critical issues as follows:

- Clock synchronization: every single part in spectrum-watching circuitry must be accurately synchronized with the USB reference clock, that's why clock synchronization must be applied carefully in all system architecture with respect to jitter, propagation delay and latency problems in electrical components.
- Congestion control: Without any congestion control even a millisecond delay could result in losing 64 Kb of information (if the sampling rate is assumed 8MSPS with 8–bit resolution), that's why the system reliability highly depends upon the performance of congestion control sub-system.

7.2. Preliminary Study

The PC interface should support at least 64-Mbps data transfer rate in order to be applicable in data conversion process performing by 8MSPS ADC with 8-bit resolution.

This may need a real-time DFC unit including high speed data buffers, but it mainly depends upon the type of PC port and hardware characteristics. Since the USB 2.0 is widely used in many commercial and industrial devices and supports up to 480 Mbps, it is preselected for this application. On the other hand, based on the USB standard data communication process may experience short delays due to multi-task feature of WINDOWS OS, that's why employing a data buffering sub-system is taken into account in order to prevent losing information. Figure 7.1 shows the basic blocks in designing a processor-based high speed DFC unit.

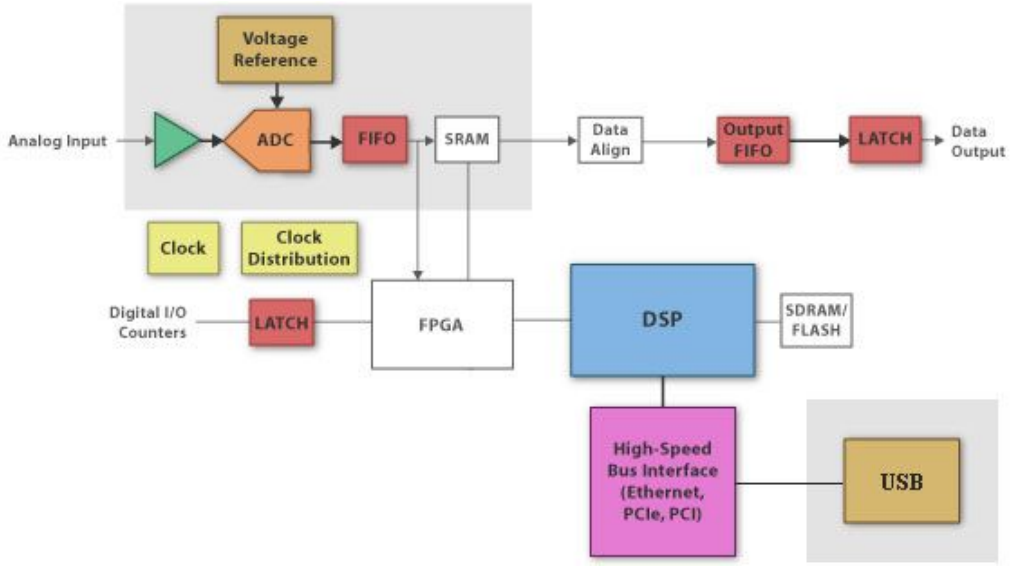


Figure 7.1. Complete block diagram of DFC design (TI, 2011)

With respect to the aforementioned block diagram the logic-circuits as well as memory interface should be implemented on the FPGA-based platform, otherwise using of discrete components demands more PCB space, thus Altera NIOS development board is considered in pre-evaluation efforts. Moreover, TMS320LF240 eZdsp DSP evaluation board is used along with NIOS board in order to test the functionality of the demonstration board (see chapter3) which is shown in Figure 7.2.

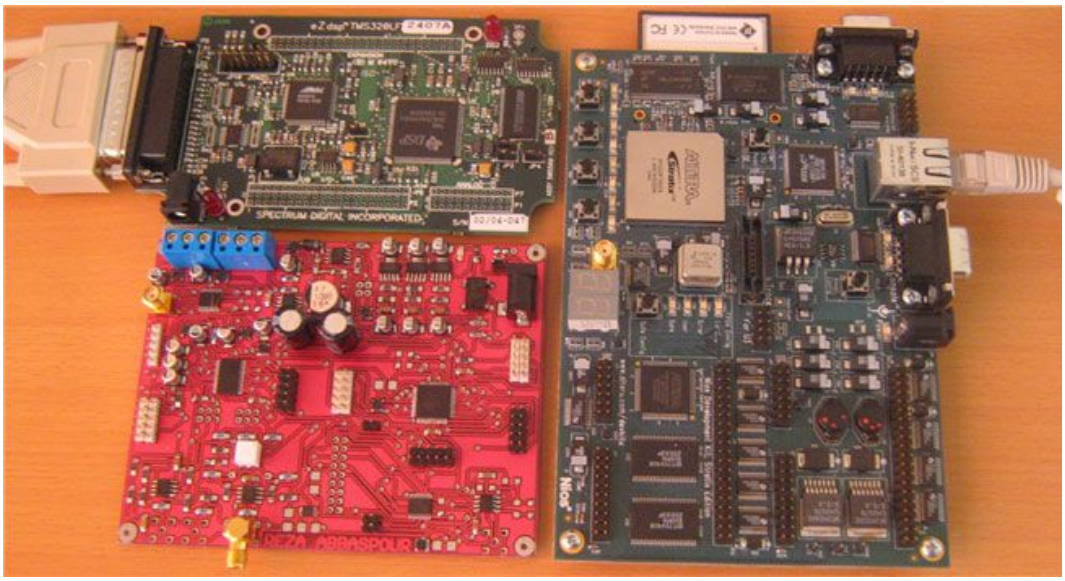


Figure 7.2. Demonstration board along with DSP and FPGA modules

As it will be explained in chapter 5, a research is conducted on selecting a proper USB controller IC and finally the well-known low cost FTDI USB 2.0 controller has been chosen based on its functionality as PC interface transferring data from the spectrum-watching hardware to the computer side. In principle, data transfer procedure requires a local embedded processor in order to monitor USB port status and synchronously triggering the hardware in right time moment based on device timing diagram. But we have aimed to eliminate the preceding processor in data transfer procedure and control the USB port via a processor-less DFC unit that will be described in the following sections.

7.3. Read Access on the THS10082 Using RD

The read timing diagram is illustrated in Figure 7.3 when the WR input is programmed as write input. Moreover, in this configuration the input RD is initiated as the read-input. Since the RD control pin is the last signal becomes valid, the timing is called RD-controlled.

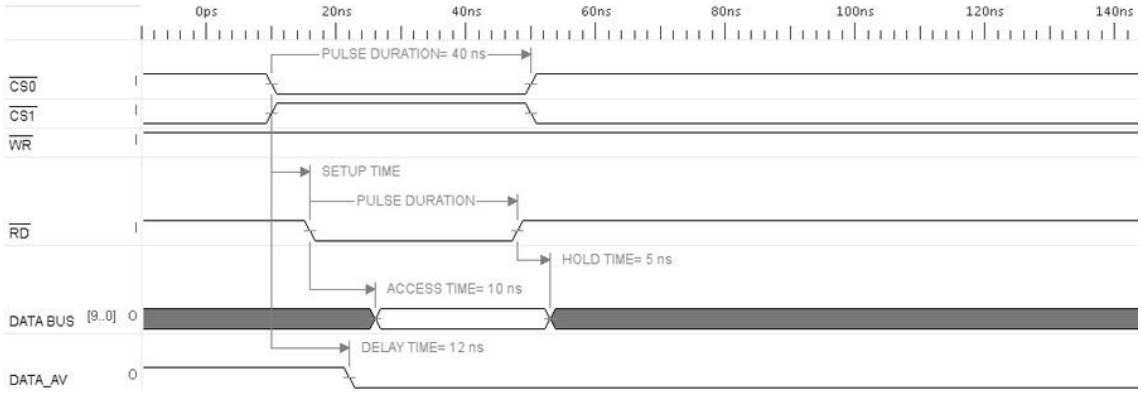


Figure 7.3. The THS10082 read access timing diagram controlled by RD

Accordingly, the CS0 should become low and then the CS1 should be pulled high in order to prepare the initial condition for read access controlled by RD input. Once the *RD input becomes low,* the read operation will be started and when the read process is completed the ADC output will be latched for 5 ns.

7.4. Read Access on the THS10082 Using R/W, CS0-Controlled

The read access procedure is configurable via control registers of the THS10082 and it can be performed by two control bits. In this configuration, the RD control pin is disabled via bit 6 of CR1, then it should be tied to high-level and hence the read access would be controlled through CS0 and R/W pins. The timing diagram in Figure 7.4 shows the control sequences which must be followed by DFC. This means that the CS1, R/W and RD should firstly be switched to logic-high and once the CS0 becomes logic-low the read access would be started.

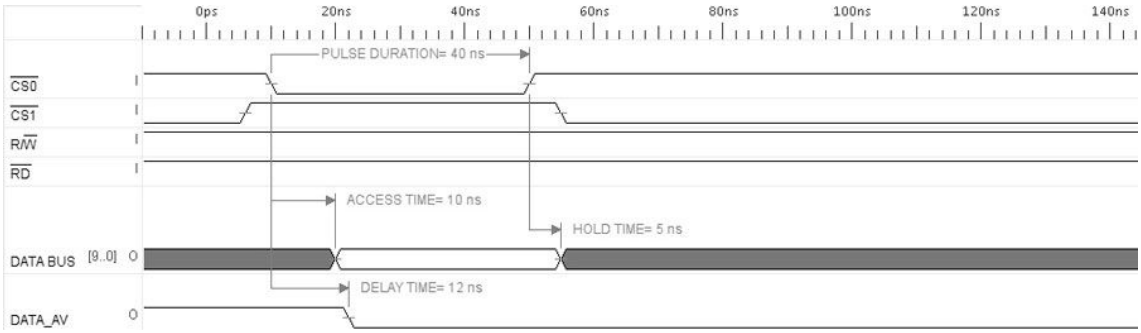


Figure 7.4. The THS10082 read access timing diagram controlled by CS0

The DATA_AV signal becomes high just after the trigger condition is satisfied. Whenever the CS0 input is pulled down and the read access begins, the DATA_AV signal would immediately be reset after 12 ns delay. In this configuration the read access could be controlled only by a single bit (CS0), while the rest of control signals (R/W, CS1 and RD) are tied to high-level, thus this mode of operation could be considered as a simple read-access configuration on the THS10082.

7.5. Write Access Using R/W, CS0-Controlled

In principle, write access process is almost similar to the read access, while the CS0 is configured to be the main control bit. The R/W control bit is the only one which toggles the process from read access to write access. By pulling the R/W pin down, the external controller could start to write on internal registers of the THS10082. Figure 7.5 shows the write access timing diagram, controlled by the CS0.

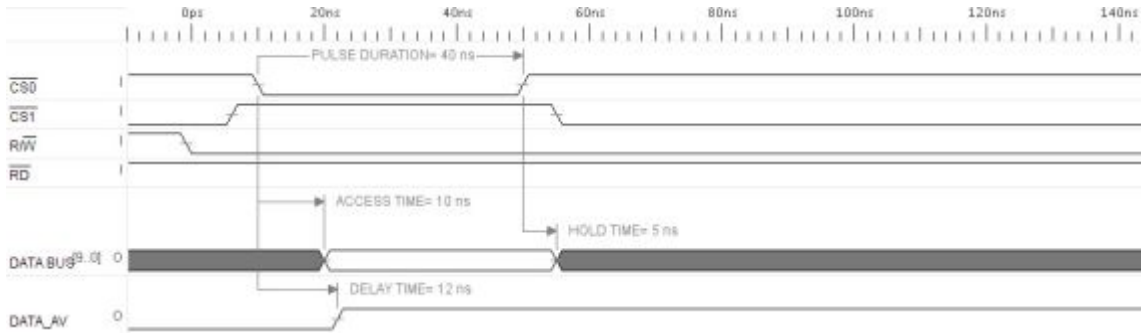


Figure 7.5. The THS10082 write access timing diagram controlled by CS0

7.6. Theory of Operation

The entire read operation is simultaneously served by the THS10082 internal logic circuit, USB interface and data flow controller. The flowchart in Figure 7.6 shows the read process and data transfer to PC side. The PC application software starts data transfer by opening the USB port (see chapter 5). Packet size is also set by the application software, which means that the USB controller IC knows how many conversions should be read from the ADC unit. If the end-user defines 256 as the packet size, then 256 sequential conversions (8-bit data block) from the THS10082 will be read. Once the dataflow controller detects the DATA_AV signal on the THS10082, it enables the SN74AUC16373 in order to hold the data on parallel data bus. Finally, the data flow controller checks whether the USB interface is available and then proceeds to write the information on the FT2232's internal FIFO. If the USB port is busy data loss would be occurred and then an error signal will be generated by the DFC which is applicable to calculate error rate. The whole process would be iterated until the PC application receives the whole data blocks equal to the user-defined packet size.

The DFC unit plays major role in process control. It synchronizes the control signals (CS0, WR and LE), trigger the USB interface and orchestrating the read access process on the THS10082 as well as data writing on the FT2232's FIFO. Since the whole design is intended to be processor-less, the DFC is implemented in gate level. This means that the combination of NAND gates along with high speed Flip-Flops, accomplishing the process control. On the other hand, using digital gates significantly reduce the BOM

cost while the traditional discrete logic ICs are cheaper than costly high speed embedded processors. Alternatively, as a future works those discrete ICs and logic circuit can be integrated on a low-cost CPLD chip.

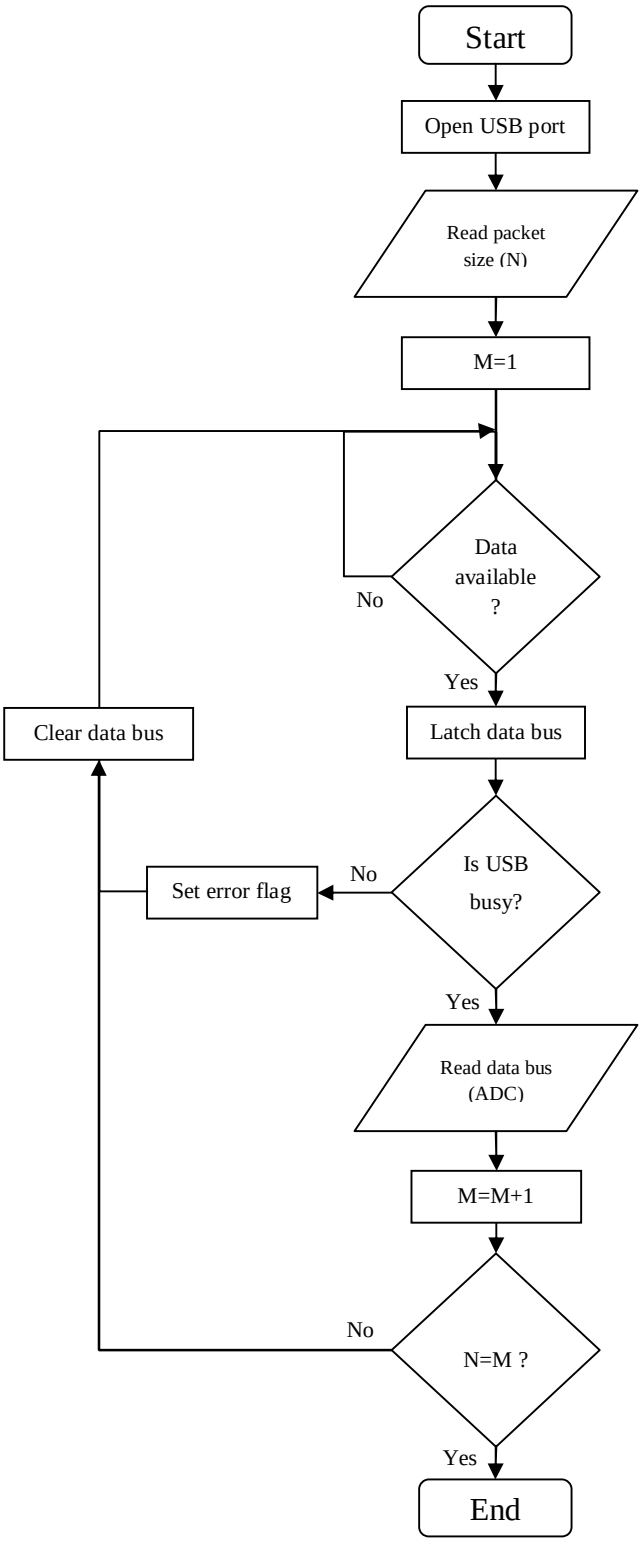


Figure 7.6. Flowchart of the entire read access process

7.7. Various Designs

Processor-less design style is the main concentration in the DFC unit implementation, which means that a logic circuit that is built around digital gates, takes the control of data communication between USB interface and ADC part.

Partially, the typical failure in digital design, in addition to clock-domain issues, includes design bugs mainly caused by timing errors. Furthermore, clock domain crossing errors are a confluence of bad implementation, logic and timing which are very difficult to diagnose and detect in the lab or via simulation, that's why the design should be verified in prototype level. Thus, four distinct prototypes are being tested in order to improve the hardware functionality. Finally the third prototype demonstrated the expected performance, requiring for high speed data transfer via USB port.

7.7.1. DFC First Design

The first DFC implementation is shown in Figure 7.7 which is designed based on AHC logic family gates. In order to trigger the FT2232 according to the USB timing diagram (see Figure 5.3), all the control signals should be synchronized with minimum latency. That's why the D-type Flip-Flops have been employed to synchronize the control signals with the reference clock. In this regard, the Flip-Flop output (where is connected to a control signal) takes the state of its input D (where is connected to DFC command line) on the rising edge of the reference clock that is connected to the Flip-Flop's CLK pin. Since Flip-Flops have relatively a short propagation delay, they can be employed to synchronize the signal on the D input with the clock signal on the CLK pin. On the other hand, two ICs in the DFC circuit are fed by the reference clock, where the ICs' input capacitance adds more distortion on the reference clock. Moreover, the additive capacitance causes electrical mismatching between the clock management circuit and USB reference clock output, that's why the first Flip-Flop (U32A) is employed to duplicate the reference clock signal in order to feed the rest of circuitry—Edge1 signal.

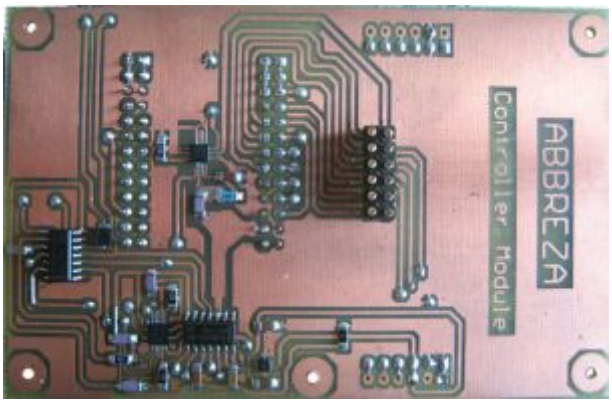


Figure 7.7. The first implementation of DFC unit

As it shown in Figure 7.8, the second Flip-Flop (U32B) checks the DATA_AV signal status on the rising edge of the Edge1 signal and then it generates the control signals (LE and CS0). Once the CS0 signal is generated, the read access on the THS10082 will be started and thus the DATA_AV signal will be forced to become inactive. On the other hand, the latch-enable input of the SN74AUC16373 is simultaneously taken high via the LE signal. This prepares the data bus to accept new data by unlatching the SN74AUC16373's outputs. At this moment the DATA_AV signal is low and then with the first rising edge on the U32B's CLK pin, the control signals will be toggled, hence the latch IC holds the data on the parallel data bus.

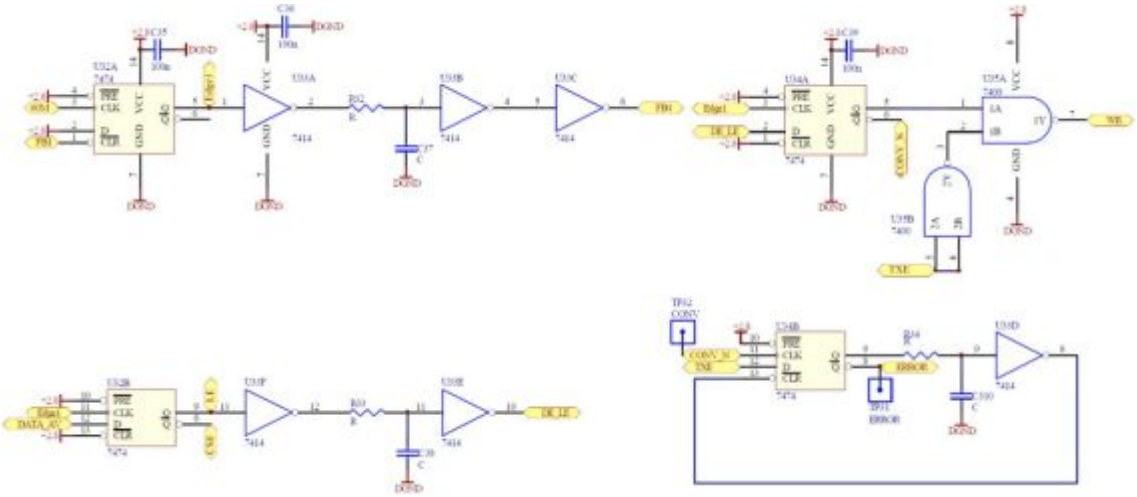


Figure 7.8. The schematic of first DFC design

Additionally, the RC delay circuit connected to the U32B's output creates the delayed version of the LE signal (DE_LE) to support the third Flip-Flop (U34A). The DE_LE signal indicates the data is read from the THS10082 and is available on the parallel data bus. Then the U34A synchronously generates the WR signal in order to trigger the USB interface based on the DE_LE status. In other words, the U34A generates the WR signal on rising edge of Edge1 only when the DE_LE signal is high on the D input of the U34A, indicating the read access time (10 ns) is already passed. The CONV_N signal is reverse of the WR signal, showing a read request is generated for USB interface. If the USB port is not ready to transfer information to PC side, it pulls up TXE pin of the FT2232. This ignores the read request because the WR signal could not be generated anymore (see the schematic in Figure 7.8), and thus the data block would permanently be lost. The forth Flip-Flop (U34B) is designed to generate error signal with respect to CONV_N and TXE status. While a rising edge CONV_N on CLK input of U34B indicates data should be read by USB interface, TXE signal in high-level condition simultaneously shows the port is busy, and thus the Flip-Flop sets its output, exhibiting data loss. The clocking and timing plan of the DFC is analyzed by TimingTool software and the result is illustrated in Figure 7.9. The timing diagram helps to verify the design before prototyping, reducing implementation cost. It is also applicable to take into consideration some practical design issues such as propagation delay time (t_{pd}), CDC glitch and clock jitter.

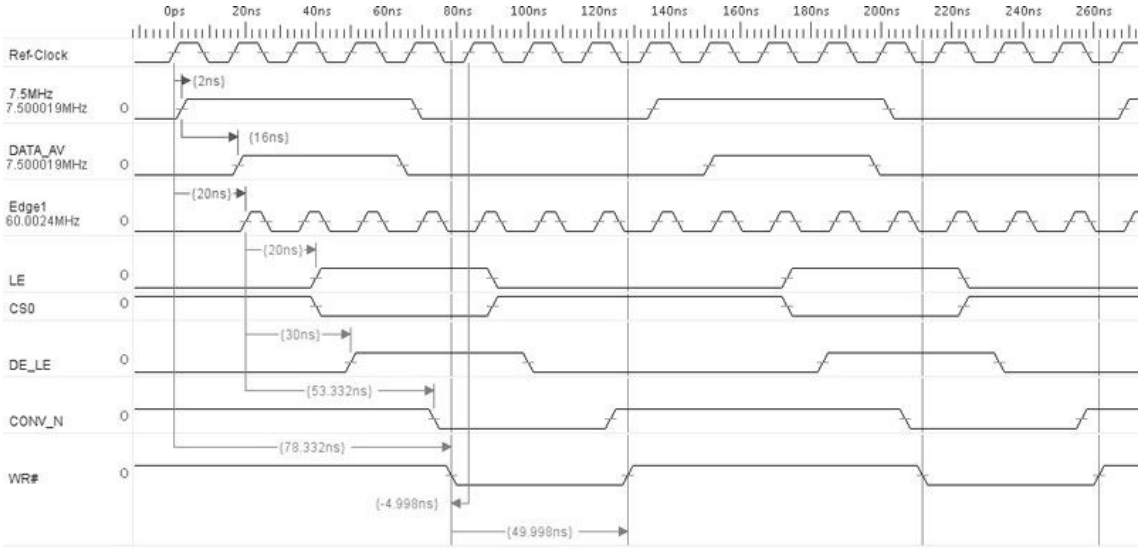


Figure 7.9. Timing diagram of first DFC unit design

While the logic circuits are excessively sensitive to unsought propagation delay which may cause CDC glitch between asynchronous clock domains, t_{pd} is taken into account in timing diagram for all components. On the other hand t_{pd} is not exactly mentioned for some components (e.g. SN74AHC74Q-Q1) in their technical datasheet, thus an estimated value for t_{pd} is considered in timing analysis.

In practice, the first design could not demonstrate proper functionality. Because its timing analysis particularly depends on propagation delay assumptions, while real values vary from component to component and even could be changed by ambient temperature fluctuations. In other words, the component profile critically affects overall system performance and thus this first design could not meet the expected reliability and robustness.

7.7.2. DFC Second Design

As it is explained in chapter 5, the USB controller IC should be triggered on the rising edge of 60 MHz reference clock by the DFC sub-system, thus it is extremely important to synchronize the whole digital signals with the reference clock. From this point of view, jitter and propagation delay of logic circuits may result in unwanted behaviors as the first DFC design has such a problematic issue. To mitigate the problem those critical logic nodes which their operation are closely correlated to the synchronization issue are identified and then replaced with AUC logic family gates in order to minimize the propagation delay. The AUC technology meets the demand for faster switching speeds by offering operation speed at 1.5ns to 2.0ns propagation delay time. Moreover, the AUC is the first logic family optimized for operation at the 1.8v node and this requires voltage-level translation circuit while the USB interface is 3.3v type. With respect to this issue, using voltage-level translation circuit between the DFC unit and USB interface is avoidable and consequently it adds more delay, resulting in asynchrony as well. Figure 7.10 shows the second implementation of the DFC daughter board which has some minor differences in circuit design in compare with the first design. Since the AUC digital gates are employed, the additive capacitance on reference clock line

causing by effects of the gates' input is reduced, thus the two Flip-Flops (U32A and U34A) are directly connected to the reference clock in this architecture.

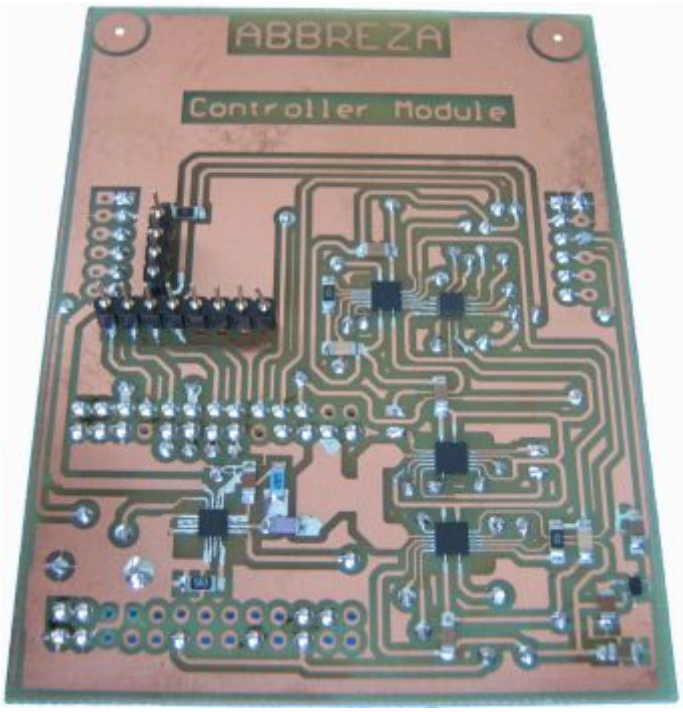


Figure 7.10. Second design of the DFC daughter board

In order to avoid using of a voltage-translator IC, the AUC type NAND gate connected to the WR pin of the FT2232 is powered with 2.7v. This provides a bridge compatible with both 3.3 and 1.8 voltage-levels.

7.7.3. Final DFC Design

In practical experiments, the second DFC architecture could not achieve 10^{-2} error rate, so a new design has been implemented as the third one. Since the CONV_CLK should be synchronized with the reference clock, sort of dependencies in clock distribution sub-system must be taken into account, thus in a successful design architecture, clock dependency in asynchronous clock domains must be minimized as much as possible. On

the other hand, the second DFC design is based on an open-loop architecture which does not receive any feedback from its output, that's why a new design structure is adopted in order to solve the aforementioned problems. The schematic in Figure 7.11 shows the closed-loop design of the DFC unit. In this new design, it is tried to overcome the clock dependency issue, which means that the ADC unit should not necessarily be fed by the reference clock. This provides more freedom in design procedure while the ADC part can even have its own local oscillator.

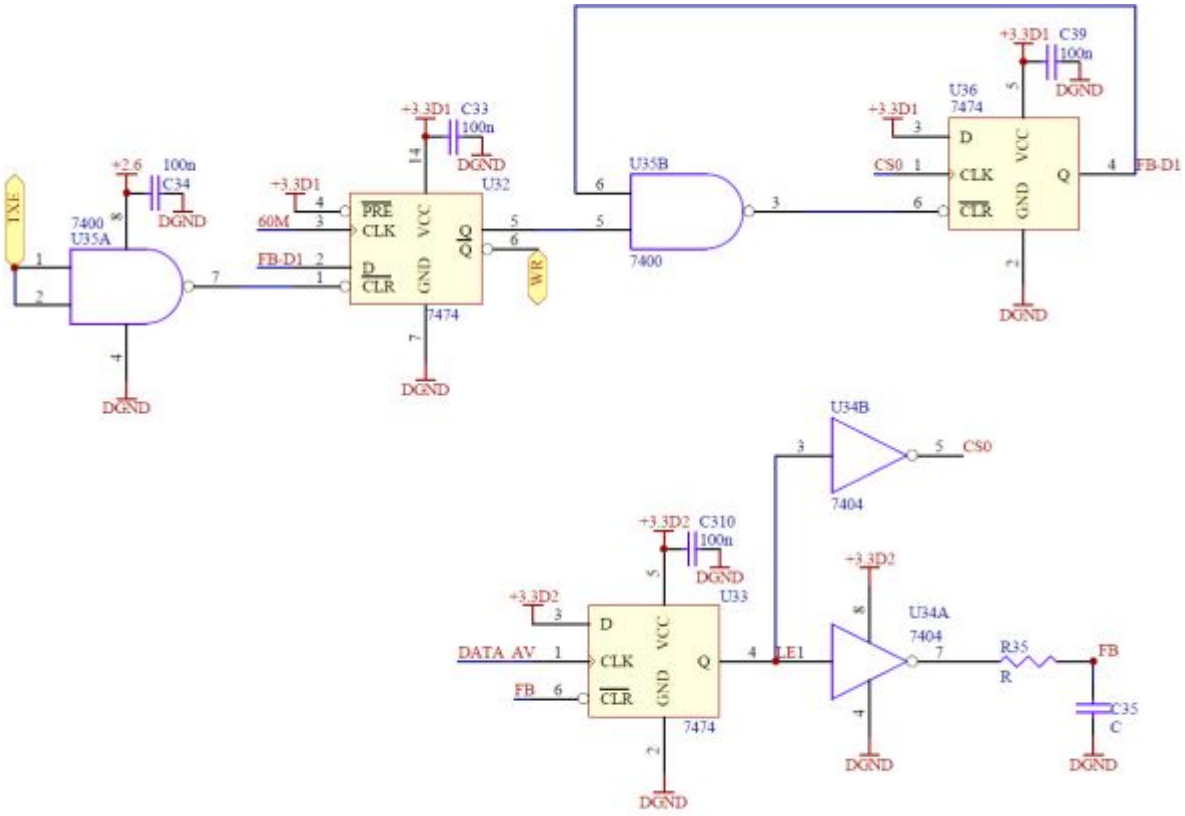


Figure 7.11. Schematic of the final DFC design

The first Flip-Flop (U32) generates the WR control signal which is the most critical one. Based on the FT2232 specifications, the WR control signal should be synchronized with the 60 MHz reference clock. That's why this node is the only one that is synchronized with reference clock. The WR signal status is controlled on the basis of the feedback signal (FB-D1) condition. This means that the U32 generates the trigger signal WR,

indirectly from the CS0 signal through the U36. In other words, the rising edge of the CS0, triggering the U36 shows that the read access procedure is completed and data is available on the parallel data bus. Then the FB-D1 signal will be generated only if the CLR pin of the U36 is set high-level, indicating that the USB interface is in steady-state situation. Once the FB-D1 signal level changes from low to high, the U32 waits for the first rising edge on the reference clock, thus the rising edge sets the U32 output high with respect to high-level of the FB-D1 signal. At this point, the NAND gate pulls the CLR pin of the U36 down, resetting its output (FB-D1) to low-level. Hence, the U32 does not generate the WR signal on the rising edges of the reference clock while the FB-D1 signal is logic-low. This mechanism is designed to ensure that, once the data is ready on the bus, it will be sent to the PC side only for one time and thus this prevents multiple transmissions. The timing diagram in Figure 7.12 shows that the LE and CS0 signals are generated by the U33 just after occurrence of the DATA_AV rising edge.

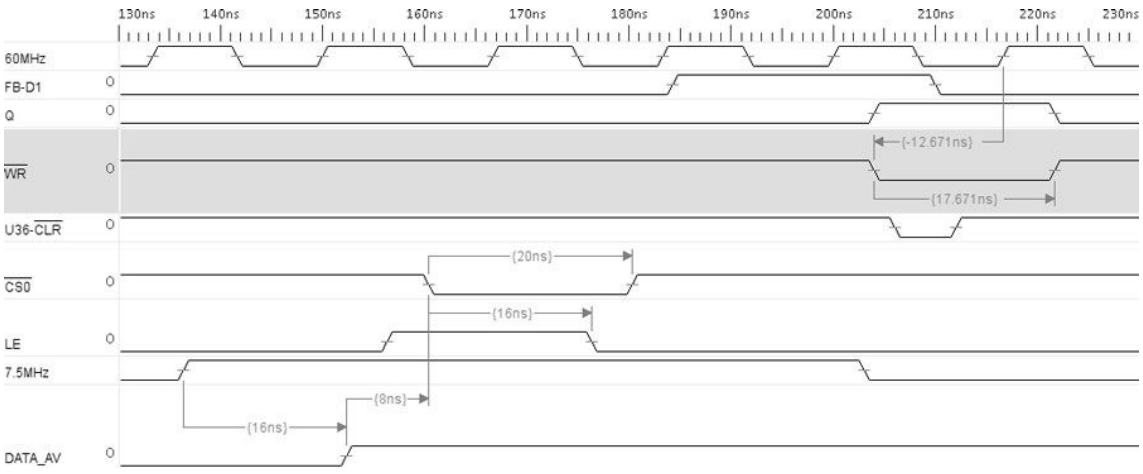


Figure 7.12. The DFC timing diagram along with control signals

As it shown in Figure 7.11, the CS0 signal is simply inverse of the LE signal, resulting from the NOT gate, thus the gate propagation delay practically provides data hold time for the latch IC. In other words, when the LE signal becomes active, the read access is already completed and the data could safely be latched on the data bus. From the other point of view, 4 ns guard-time is reserved by keeping the CS0 signal high in order to

guarantee that the data latch procedure would be completed just before CS0 signal becomes high again. It is wise to mention that the high-level duration of the LE signal which should be at least 14ns is adjustable via the RC delay circuit (R35 and C35) that is shown in schematic.

7.8. How to Test the Design Architecture

The hardware test should be accomplished under constant controlled condition in order to provide valid, reliable and comparable results, that's why ADC emulator is designed and implemented in hardware level (see section 4.10). This simulates the ADC responses based on incoming control signals (CS0 and LE) that are generated by the DFC unit, which means that the emulator circuit receives 7.5 MHz conversion clock as well as CS0 and LE control signals, and then provides the DATA_AV signal and the parallel data bus according to the THS10082 timing diagram. At every falling edge of the conversion clock, the emulator increases the data value by one from zero up to 0xFF. Then it is read by the USB interface under control of the DFC module once the emulator generates the DATA_AV signal, indicating the data is increased by one. With respect to the procedure, it is expected to receive 256 Bytes in computer side starting from zero up to 0xFF. Thus, any disorderliness in received Bytes could be detected while the numbers should be displayed in ascend order. This reveals countable errors that could be divided by 256 and the result represents the error rate.

8. ACTIVE INTERFACE CIRCUIT

Passive type analog interface is commonly used in high-speed analog to digital convertors while input signal has enough power to drive the ADC IC input and its amplitude can manually be adjusted. On the other hand, if the noise and distortion contributions of passive components could be considered negligible, low frequency response, pass-band ripple, cost and matching issue can limit the usefulness of passive interfaces. That's why, active interface circuits which are based on high-speed OPAMPs or FDAs are taken into account as alternative. The OPAMP-based drive circuits are low-cost and flexible solutions for those applications demand gain adjustment, high input impedance, level shifts, flat pass-band and low ripple. In principle, designing an interface circuit mainly depends upon the ADC input specifications which can be buffered or unbuffered type. The drive circuit should mainly feature the following characteristics for unbuffered ADC inputs:

- Low power loss over the input frequency range
- Sink (ADC input) and source (signal generator) impedance matching

With respect to the preceding reasons, the ADC input impedance must be known as a design parameter. In order to meet the signal chain requirements such as single-ended signal conversion to differential signal path and signal source buffering, a driver amplifier could be employed. This helps the designer to take advantage of using differential transmission line that features a good AC performance in high input frequencies. Thus the analog inputs are preferred to be driven differentially, for optimum performance. In this case, CM noise immunity as well as even-order harmonic rejection would be improved.

The block diagram of a band-limited driver circuit is illustrated in Figure 8.1 for a typical high-speed ADC. In this project the signal source is considered single-ended while mostly the RF receiver blocks feeding the spectrum-watching device input, feature single-ended output; therefore the interface circuit should convert it to differential type. This would be accomplished by using whether a high performance FDA or dual OPAMP-based active interface.

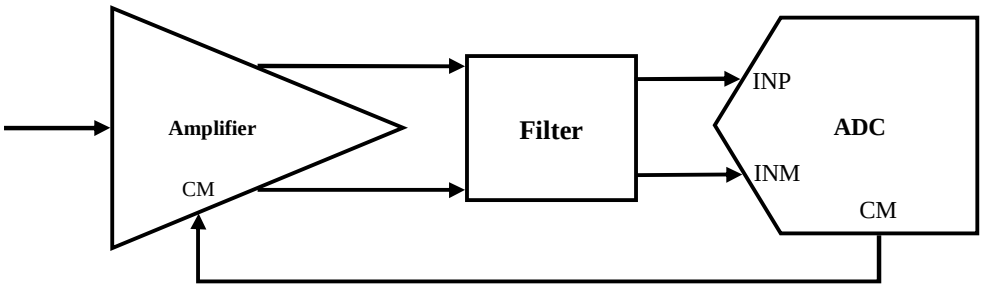


Figure 8.1. Block diagram of a typical ADC drive circuit

The circuit diagram in Figure 8.2 is designed in ADC-Pro software environment based on the THS4052 which is a dual, voltage-feedback OPAMP. It is very low-distortion, high-speed and cost effective solution for driving high-speed ADCs in noninverting buffer configuration.

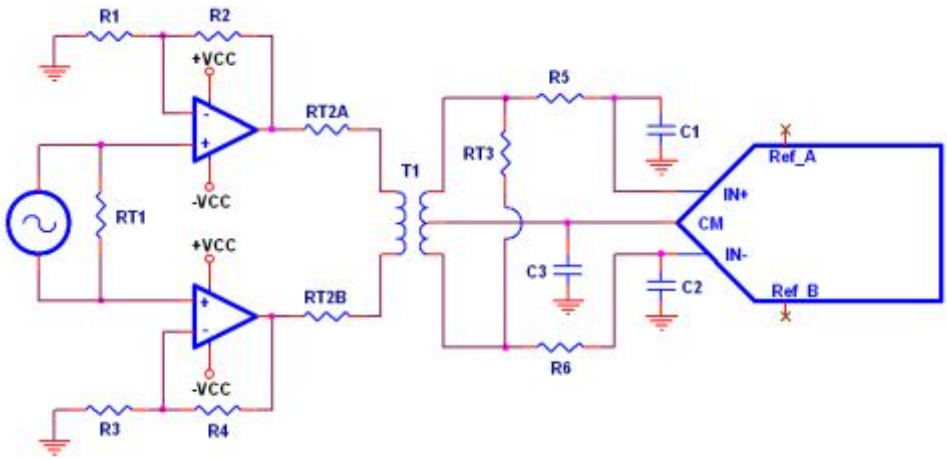


Figure 8.2. Schematic of dual OPAMP active interface

The design exploits a RF transformer to isolate the ADC IC from the signal source. In addition, the transformer is also used for noise filtering, impedance matching and biasing purposes. The signal source should have a DC CM voltage equal to the ADC reference CM voltage which is one of the general requirements for effective amplifier performance down to DC. This is accomplished by the RF transformer since, the

transformer center-tap biases the input signal (V_{SIG}) with the reference CM voltage of ADC; therefore as ADC IC selection criteria, the component should support the amplifier CMIR and vice versa. This means that, The INP and INM inputs of the ADC IC should externally be biased around the CM voltage where is generated as a reference voltage on the ADC V_{CM} pin. Hence, for a full-scale differential input, the input pins (INP and INM) must symmetrically swing between CM $\pm V_{PEAK}$.

The interface filter (see Figure 8.1) is generally a LPF or BPF to reduce the out-of-band noise. In the preceded schematic, the filter circuit has been designed around the RF transformer along with resistors and capacitors, thus the RLC circuit is served as a noise-limiting LPF and to provide simultaneously some isolation between the OPAMP output and the ADC CM kick-back. The RT3 resistor connecting the transformer outputs serves as a matching component and current path for the incoming signal. It is wise to mention that using a higher value for RT3 than 200Ω may result in degraded performance.

8.1. Active Driver Design Considerations Based on FDA

If a driver amplifier is needed as it is required in this project, its parameters can have a great impact on the overall system performance. In this section, the main FDA-based driver circuit metrics and how they characterize the entire system performance will be described. Using FDA in ADC drive circuit provides easy and efficient DC-coupled interface with gain adjustment. A proper selection of driver amplifier often simplifies the design procedure and improves the signal chain that is well-suited with the application, but the selection criteria could be considered on a case by case basis.

The FDA integrates differential inputs and outputs in a single chip serving as a proper and compact ADC driver while conversion of a single-ended input to differential outputs requires. The proposed circuit converting single-ended bipolar input signal to differential type is shown in Figure 8.3, whose parameters are defined as follows:

- V_S+ and $V_S–$: The power supplies to the FDA.
- R_S+ and R_T+: The resistors that provide input signal attenuation.

- R_G and R_F: The amplifier gain-setting resistors.

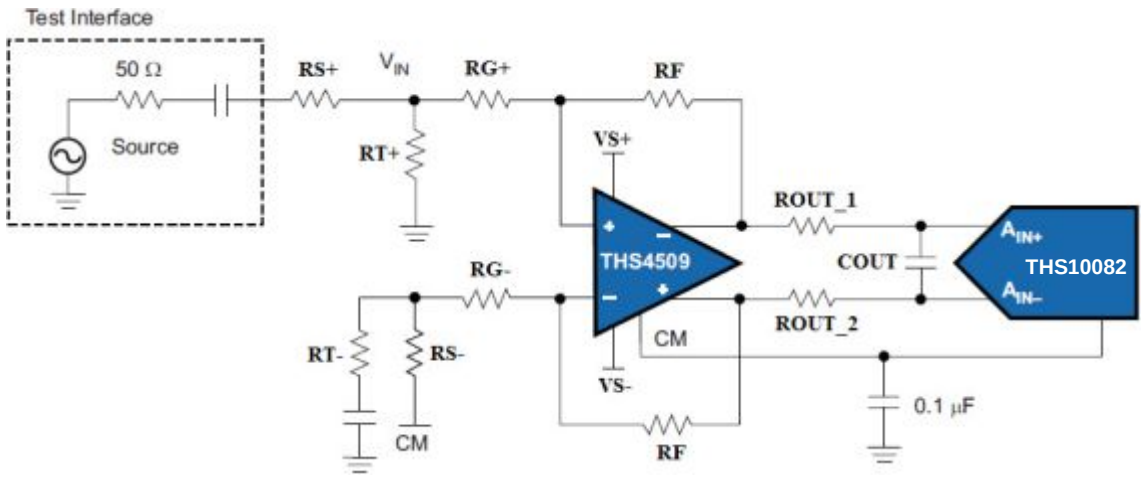


Figure 8.3. Schematic of ADC drive circuit based on FDA

Since, the input signal is considered as a low-amplitude single-ended bipolar, it should be amplified and level-shifted to the appropriate levels to fittingly drive the input ADC. Thus, the first step is to design the amplifier circuit in order to accept a single-ended bipolar signal. This is firstly accomplished by input signal attenuation via a voltage divider circuit consisting of RT+ and RS+. Secondly, allotting a same value for RT+ and RT- resistors is the key point in design steps, setting the amplifier gain on each side of the signal balanced so that no offsets are generated.

The RS– and RT– have been added to the circuit in a manner that uses V_{CM} to provide biasing on the alternate input. At this point in order to derive the equations, AC-analysis should be performed by considering the DC bias signals grounded, short signal source and the capacitor in series with RT-, and thus the equivalent resistor (R_{Eq}) can be defined by the Equation 8.1.

$$R_{Eq}+ = RG_+ + RS_+ \,||\, RT_+$$
$$R_{Eq}- = RG_- + RS_- \,||\, RT_- \qquad (8.1)$$

The simplified schematic which is illustrated in Figure 8.4, shows a typical FDA-based amplifier circuit consisting of gain adjustment and feedback resistors. This facilitates finding the gain equation.

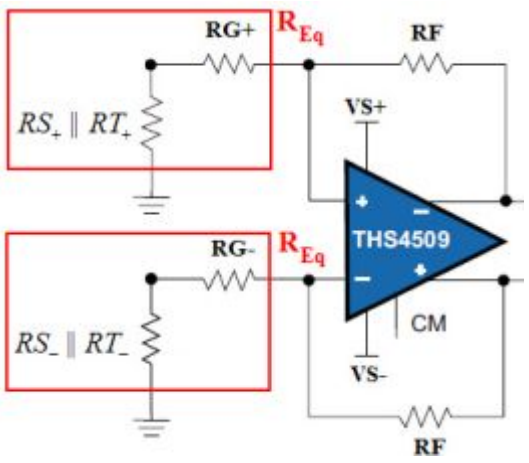


Figure 8.4. Simplified schematic of amplifier circuit

Balanced amplification should be performed on single-ended input signal while it is converting through the THS4509 to differential type in order to minimize distortion. This is accomplished by setting the amplifier gain equally on both signal side, which means that the bipolar alternate input signal should equally be amplified and thus the positive and negative gains of the FDA as the Equation 8.2 shows must be equal.

$$V_{OUT\pm} \Big/ V_{SIG} = \left(\frac{RT_{\pm}}{RT_{\pm} + RS_{\pm}} \right) \times \left(\frac{R_F}{R_{Eq} \pm} \right) \qquad (8.2)$$

The other parameter which should be taken into account is noise gain, uses to define the stability criteria of OPAMP and could be calculated as the gain from the input terminal to the output. In General, OPAMP stability could be affected by considering a value below the minimum noise gain, leading to instability or oscillation. A stable OPAMP with unity-gain is compensated for a noise gain of 1, while a stable FDA would

typically be compensated for a noise gain of 2. Thus, the noise gain of the FDA should be set to 2 in the gain equation by making the second half of the Equation 8.2 equal to 1. Hence, the R_F value is chosen on this basis as it defined in Equation 8.3.

$$\begin{aligned} R_F &= RG_+ + RS_+ \,||\, RT_+ \\ R_F &= RG_- + RS_- \,||\, RT_- \end{aligned} \tag{8.3}$$

With respect to the Equations 8.2 and 8.3, the amplifier gain is simplified further as it shown in Equation 8.4. As a consequence, it seems that the gain would not be adjusted by the feedback resistors (R_F). But form the other point of view, the Equation 8.3 demonstrates that value selection of the RT and RS resistors is dependent to R_F. Accordingly, the amplifier gain is indirectly affected by R_F.

$$gain = \frac{RT_-}{RT_- + RS_-} = \frac{RT_+}{RT_+ + RS_+} \tag{8.4}$$

In order to balance additive offset due to the CM voltage of the input signal, the Equation 8.5 has to be satisfied. This means that the CM voltage of V_{SIG} is attenuated by the input voltage divider (RT+ and RS+), and then the resulted signal amplitude should be equal to provided reference voltage on RG–. Since there is no need to implement one more reference voltage for this purpose, thus this can be achieved through the CM voltage of the ADC IC (V_{CM}) and the voltage divider on the alternate input, adjusting V_{CM} level on the RG– (the R_S– and R_T–).

$$V_{CM} \times \left(\frac{RT_-}{RT_- + RS_-} \right) = V_{sig\_com} \times \left(\frac{RT_+}{RT_+ + RS_+} \right) \tag{8.5}$$

Since maximum power should be transferred from the signal source to amplifier, impedance matching circuit required. Amplifier input impedance (Z_{in}) can simply be adjusted by the input resistors (RS_{+}, RT_{+} and RG_{+}) and the Equation 8.6 approximately shows the relationship.

$$Z_{in} \approx RS_{+} + RT_{+} \parallel RG_{+} \tag{8.6}$$

The THS4509 is utilized in the proposed architecture, simplifying the necessary interface filter. Thus a typical LPF RC circuit which is illustrated in Figure 8.3 has been employed in order to reduce the out-of-band noise. Based on the Equation 8.7, the filter cut-off frequency is set via the R_{OUT} and C_{OUT}.

$$f_{-3dB} = \frac{1}{4\pi \times R_{OUT} C_{OUT}} \tag{8.7}$$

8.2. Practical Results

The THS10082 accepts the analog input signal in the range 1.5v to 3.5v, centered at 2.5v. With respect to the signal conditioning requirements and the aforementioned calculations, the TH4509 should shift the input signal in the range -1v to 1v with minimum distortion to the range 1.5v to 3.5v; therefore the voltage reference generated by the THS10082 is used to center the input signal at 2.5v. Hence, the CM voltage pin of the TH4509 is directly connected to the THS10082 voltage reference. Since the ADC maximum sampling rate is 8Msps, the input signal frequency could be at most 8 MHz, that's why the interface LPF cut-off frequency is considered 8 MHz. The values 56Ω and 150pF are selected for R_{OUT} and C_{OUT}, and thus the filter -3-dB bandwidth is approximately 9.5 MHz, that covers the first Nyquist zone of the converter. Moreover, the interface input impedance should be characterized to be driven by 50Ω signal source thus, the input resistors value must be selected in a manner that features 50Ω

transmission line characteristic and simultaneously provides desirable amplifier gain. Table 8.1 contains information regarding the interface circuit input impedance (Z_{in}), the power supplies to the amplifier (V_S+ and V_S–), current consumption (I_c), input and output signals characteristics. These are measured in seven different experiments and the results reveal that the last test demonstrates the expected functionality.

Table 8.1. Practical test results along with analog interface characteristics

Test Number	**T1**	**T2**	**T3**	**T4**	**T5**	**T6**	**T7**
Frequency [MHz]	3	3	3	3	5.8	5.8	5.8
ZIN [Ω]	40.5	40.5	40.5	50.5	40.5	40.5	40.5
OPAMP Gain	1	1.17	1.08	0.9	1.16	1.33	2.02
ROUT [Ω]	56	56	56	56	56	56	56
COUT [pF]	150	150	150	150	150	150	150
VS+ [v]	3.66	3.66	3.66	3.66	3.66	3.66	3.66
VS- [v]	-1.34	-1.34	-1.34	-1.34	-1.34	-1.34	-1.34
VOUT-Max [v]	3.12	3.8	3.7	2.9	3.5	3.5	3.46
VOUT-Min [v]	2.12	1.45	1.55	1.1	1.75	1.5	1.52
VOUT-CM [v]	2.63	2.6	2.6	2.5	2.6	2.5	2.49
Ic [mA]	83	101	74	130	88	100	102
Vin-PP [v]	1	2	2	2	1.5	1.5	0.96
Vin-CM [v]	0.5	1	0	0	0.75	0	0.05

As a consequence, the analog input signal must be bipolar in the range -0.5v to 0.5v, centered at 0. Hence, the interface circuit can accurately amplify the input signal and then shift it to the input CM voltage of the THS10082. **Error! Reference source not found.** shows the final test result (T7).

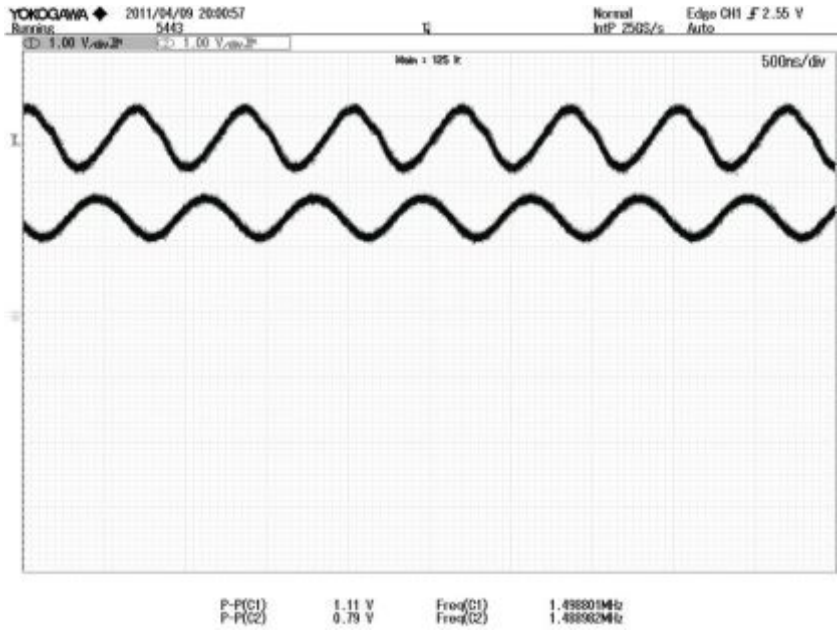


Figure 8.5. Differential output of FDA-based drive circuit and bipolar input signal

9. POWER MANAGEMENT UNIT

Achieving optimum performance from a high-speed ADC depends upon many factors such as power supply decoupling, good implementation and PCB layout grounding. All these issues are the essentials to preserving ADC accuracy. Traditionally, low-noise LDOs are employed to clean up the ADC power line noise. Anyhow, the minimum dropout voltage on LDO is typically 0.2v up to 0.5v which should be taken into account as the selection criteria. Moreover, the maximum tolerable input voltage of LDO should also be considered as a critical design parameter because in some cases step-down regulator must be employed right before a low-voltage LDO in order to reduce the input voltage. With respect to this issue, those LDOs which accept relatively high-voltage are chosen in order to dismiss using of step-down regulators and thus the project BOM cost will be decreased as well.

Power management unit is implemented in the first design phase, while the system power requirement was still an open issue, that’s why it is considered as a general purpose power supplier. Hence, it is designed in a manner that efficiently provides various applicable DC voltage-levels in order to power up connected daughter boards via pin-header connectors. This enables the designer to implement the analog and digital circuits independently on daughter boards without reengineering the entire power management unit; therefore the implementation phase can be completed in a shorter time in this strategy. On the other hand all the circuits should be implemented and tested in order to achieve acceptable performance, thus it is extremely important to provide the required resources keeping the system reliably up and running. Hence, having a general purpose, low-noise power management unit is the first objective in component selection and board layout design.

9.1. Design Architecture

The power supply board is designed to power up all the hardware units such as DFC, ADC, USB interface and analog driver. All these units demand their own specific power

requirements, while the LDOs and voltage regulators supplying the whole system should be integrated on a single board. The block diagram in Figure 9.1 shows the generated DC voltage-levels along with regulator connections to the external AC-DC switching converter. The power management unit distinctly provides seven DC voltage-levels, including negative and positive ones.

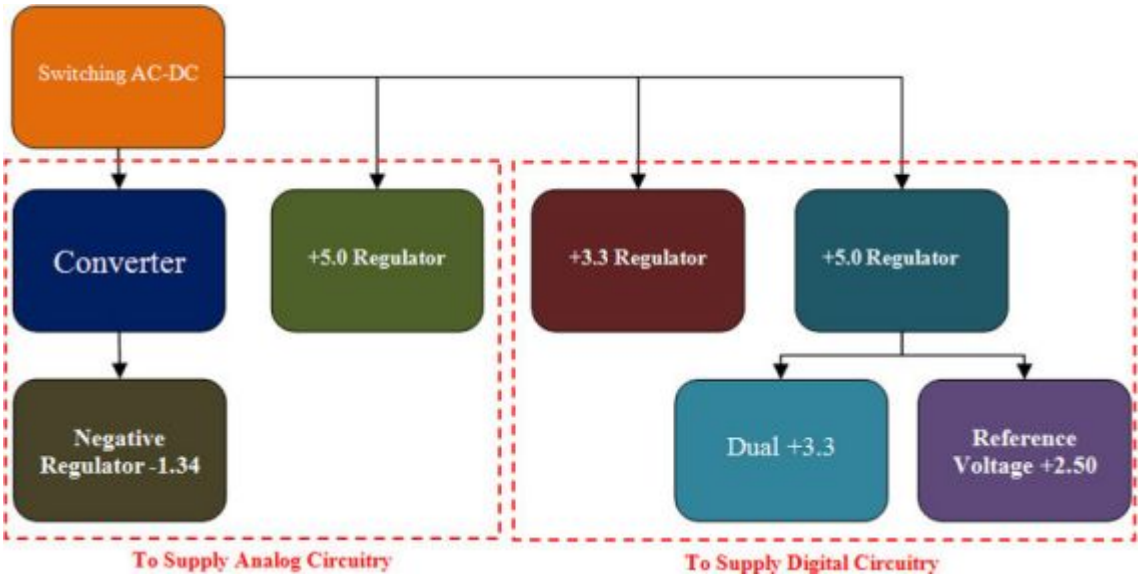


Figure 9.1. Power management unit block diagram

All required voltage-levels are generated from a single power supply line, provides by a standard AC-DC switching converter as the external DC power supply, which means that the negative rail (-1.34) should be also derived from the positive DC Voltage supply. This should typically be accomplished by a negative voltage converter, while it increases power management design complexity. In order to adopt less complex design strategy as an effective alternative, DC-DC isolated voltage regulator, yielding +5v output with separate GND connection is used. In this case, the positive polarity is tied to circuit GND and therefore the GND of the isolated regulator provides negative DC voltage with respect to circuit GND as Figure 9.2 describes the configuration in form of block diagram.

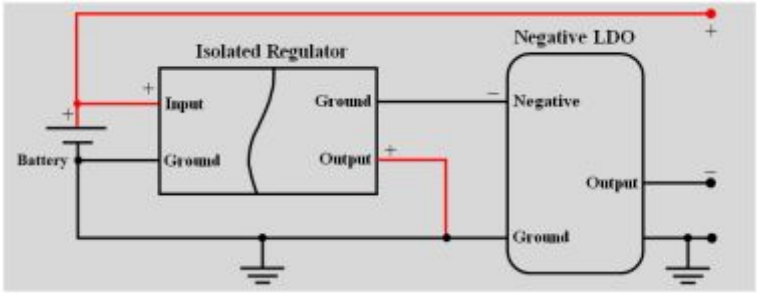


Figure 9.2. Block diagram of negative voltage generator

Since the THS4509 is powered in dual supply mode, the power management unit should generate low-noise negative DC voltage, thus a high-performance negative LDO (TPS7A3001) has been served for this purpose, cleaning up the negative OPAMP power supply line effectively. The Figure 9.3 shows the position of DC regulators and LDOs on the power management daughter board.

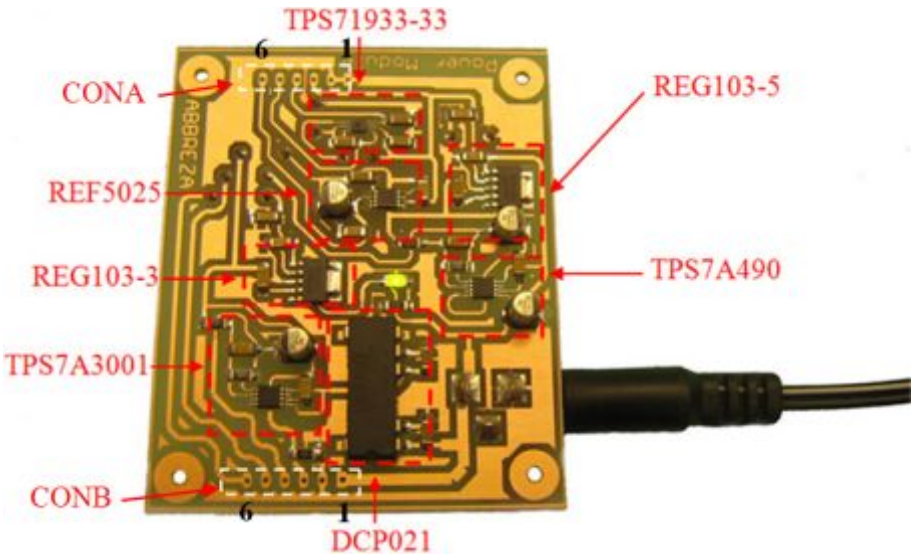


Figure 9.3. Implemented power management unit as a daughter board

A high performance ultra low noise positive LDO (TPS7A490) is also considered to generate positive DC supply demands for powering positive side of the THS4509. Indeed, the LDO is well-suited for high precise analog applications where clean voltage

rails are crucial to maximize overall system performance. Since the TPS7A490 requires a few passive components around to be operational, it can easily be integrated in circuit design. Moreover, to supply the digital circuitry (USB controller, DFC unit and data bus controller) dedicated low-noise and low-dropout 0.5A LDOs (REG103-3 and REG103-5) are served, providing +5v and +3.3v. Furthermore, a voltage reference IC (REF5025) is considered to yield +2.50v precise reference voltage to support analog interface circuit and ADC unit. The REF5025 receives its input power from the REG103-5 which is considered to power up the digital circuitry. In this way, the REF5025 is protected from input peak voltage. As it shown in Figure 9.1, the REG103-5 output is also connected to a low-voltage dual LDO (TPS71933-33) that its input does not tolerate higher than 6.5v. The TPS71933-33 provides two independent +3.3v to be used in digital circuitry. The regulators' specifications are listed briefly in Table 9.1.

Table 9.1. The employed LDOs' specifications in power management unit

Part-Number	Output Voltage	Output Current	Maximum tolerable input
REG103-5	+5V	500mA	+16V
REG103-3	+3.3V	500mA	+16V
TPS7A490	+3.66V	150mA	+36V
TPS7A3001	-1.34V	150mA	-36V
REF5025	+2.5V	30mA	18V
TPS71933-33	+3.3V	200mA	7V
DCP021	-5V	400mA	15V

Since the daughter boards should be connected to the power management board, the pin-header connectors are utilized. Table 9.2 represents the pin assignments on the power management daughter board.

Table 9.2. Pin assignments on the power management daughter board

Pin#	Name	Direction	Description
CONA1	PGND	Output	Power GND connection
CONA2	PGND	Output	Power GND connection
CONA3	+5	Output	Low noise +5v, 500mA DC power supply
CONA4	+3.3D2	Output	+3.3v, 200mA DC power supply for digital circuits
CONA5	+3.3D1	Output	+3.3v, 200mA DC power supply for digital circuits
CONA6	+2.5	Output	+2.5v, voltage reference
CONB1	Vin	Output	External DC voltage from the switching power supply
CONB2	PGND	Output	Power GND connection
CONB3	AGND	Output	Analog circuitry GND connection
CONB4	NV	Output	High-precise negative DC supply for analog circuitry
CONB5	PV	Output	High-precise positive DC supply for analog circuitry
CONB6	+3.3P	Output	+3.3v, 500mA DC power supply

10. CONCLUSION AND FUTURE WORKS

Spectrum monitoring is utilized in wide range of applications from space science to tactical data communication. Recently, using of intelligent spectrum management is greatly discoursed in commercial mobile communication while mobile data traffic growth rate is higher than anticipated, leading to seriously bandwidth shortage. Respecting this, new technologies are underdeveloped to change the traditional bandwidth licensing regime in order to increase spectrum efficiency and minimize the need for centralized spectrum management. In this regard, cognitive radio is introduced as a promising technique based on spectrum monitoring and minimizing interference. Since cognitive radio is aimed to efficiently manage available spectrum, communication algorithm is considered as the crucial research issue that is mainly developed in simulation software (e.g. MATLAB). In order to test and verify simulation results and developed algorithm an external hardware should be utilized, monitoring RF spectrum. This is accomplished via a combination of PC interface and high-speed ADC block. In this project spectrum-watching hardware successfully is implemented, tested and being used. Since PC-based high speed data acquisition modules are commonly available in market in very high price as black-box systems, the project is being inspired to provide well and open-source documentation. Furthermore, during system design procedure, cost reduction phase is performed several times to achieve a cost effective design. The implemented processor-less, memory-less spectrum-watching hardware is capable to capture maximum 8MHz input analog signal and then transfer the information to computer side via popular USB 2.0 PC port with low error rate (below 10^{-2}). The whole device costs 48 U.S. Dollar in prototype level which can significantly be decreased in mass production level. This is less expensive than current commercial data acquisition cards with the same sampling rate (e.g. typical commercial price is over 400 U.S. Dollar). On the other hand, the project is well-documented, describing every single detail in order to produce applicable knowledge for academic research projects. This means that the design can be modified, customized and even easily integrated into other designs. All the project information as well as PC side programming codes and executable files are downloadable from the project **homepage** at the following address: http://www.abbreza.com/en/spectrum.htm

The primary goal of the project architecture is to fulfill the system requirements with a modular hardware design. Consequently, identified parts of the architecture are made extensible with a possibility to extend the functionality with additional hardware modules in form of daughter boards, without a need to reengineer the whole system. The approach in this architecture is to analyze the system requirements and distribute the required functionality in hardware modules and blocks with respect to dependencies. The other purpose is to ease further system develop in future, which means that additional daughter boards can simply be attached to the existing spectrum-watching hardware. Accordingly, an automatic level controller (ALC) unit can be considered in project development phase to provide flexible input on device analog interface because in current system design, the input signal amplitude should not exceed from certain level due to compatibility issue. On the other hand, the RF receiver blocks are excluded from the scope of this project and therefore they can be implemented in project extension phase. In other words, RF amplifier and down-converter modules could be engineered and locally added to the device as a future work in order to create fully functional, RF front-end spectrum-watching device. Consequently, a wide range down-converter can be implemented in order to sweep wider RF range and thus, 8MHz baseband signal is detectable with this system architecture. For example, an arbitrary multi-band (e.g. 3 to 7 bands) down-converter, providing conversion from 1.7 up to 2.55 GHz can sweep the conversion range ideally in steps of 8MHz. The sweeping process can be controlled by the spectrum management algorithm on PC side via the available USB interface.

Finally, we believe that the cognitive radio technology is getting expanded as a cutting-edge RF spectrum licensing technique which will primarily be adopted by authorities in near future in order to mitigate bandwidth shortage in high-speed mobile data communication both in commercial and ISM-based applications. Thus, rapid development in research level is a crucial issue in order to accelerate the technology commercialization process and therefore hardware implementation is truly a big step to inject new technologies in public applications.

REFERENCES

Baker, Bonnie (2006). *A Glossary of Analog-to-Digital Specifications and Performance Characteristics.* Rev. 2008, Texas Instruments Inc.

Baker, Bonnie C. (2006). *Matching the noise performance of the operational amplifier to the ADC.* Analog Applications Journal, Texas Instruments Inc.

Bartolome, Eduardo, Vineet Mishra, Goutam Dutta & David Smith (2005). *Clocking high-speed data converters*. Analog Applications Journal, Texas Instruments Inc., pp. 20-26

Beckemeyer, Heinz-Peter (2000). *Designing With the THS1206 High-Speed Data Converter*. Application Report, Texas Instruments Inc.

Buccini, Mark (2002). *Using direct data transfer to maximize data acquisition throughput*. Analog Applications Journal, Texas Instruments Inc., pp. 14-20

Burr-Brown Co. (1994). *Principles of Data Acquisition and Conversion.* Application Bulletin, May 1994, U.S.A.

Ferrando, M. Perez (2009). *Troubleshooting I2C Bus Protocol*. Application Report, Texas Instruments Inc.

Ferrando, M. Perez (2009). *Usage of I2C™ for CDCE(L)949, CDCE(L)937, CDCE(L)925, CDCE(L)913*. Application Report, Texas Instruments Inc.

Future Technology Devices International Ltd. (2006). Title: Bit Mode Functions for the FT2232. Application Note, AN2232-02.

Future Technology Devices International Ltd. (2009). Title: Instructions On Including The FTD2xx DLL In A VS2008 Project. Ver. 1.0, Technical Note, TN_109.

Future Technology Devices International Ltd. (2010). Title: FT2232H Used In An FT245 Style Synchronous FIFO Mode. Ver. 1.2, Application Note, AN_130.

Future Technology Devices International Ltd. (2010). Title: Software Application Development D2XX Programmer's Guide. Ver. 1.03

Gentile, Ken (2003). *Introduction to Zero-Delay Clock Timing Techniques.* Rev. 0, Norwood U.S.A, Analog Device Inc.

Hendrick, Tom (2002). *Reading the Configuration Registers of the 10-Bit THS10064, THS1007, THS10082, and THS100.* Texas Instruments Inc.

Hsia, Kang (2011). *Input Impedance Measurement Using ADC FFT Data.* Application Report, Texas Instruments Inc.

Jackson, Tom (1996). *Advanced Bus-Matching/Byte-Swapping Features for Internetworking FIFO Applications.* Adv. Sys. Logic-Semiconductor Group, Texas Instruments Inc.

Karki, James (2002). *Fully-Differential Amplifiers.* Application Report, Texas Instruments Inc.

Karki, Jim (2008). *Input impedance matching with fully differential amplifiers.* Analog Applications Journal, Texas Instruments Inc.

Karki, Jim (2009). *Using fully differential op amps as attenuators, Part 2: Single-ended bipolar input signals.* Analog Application Journal, Texas Instruments Inc.

Karki, Jim (2009). *Using fully differential op amps as attenuators, Part 3: Single-ended unipolar input signals.* Analog Application Journal, Texas Instruments Inc.

Karki, Jim (2009). *Using fully differential op amps as attenuators, Part 1: Differential bipolar input signals.* Analog Application Journal, Texas Instruments Inc.

Kester, Walt (1991). *Data Converter Codes—Can You Decode Them.* Rev. A, Tutorial MT-009, Analog Device Inc.

Kester, Walt (1991). *Intermodulation Distortion Considerations for ADCs.* Rev. A, Tutorial MT-012, Analog Device Inc.

Kester, Walt (1991). *What the Nyquist Criterion Means to Your Sampled Data System Design.* Rev. A, Tutorial MT-002, Analog Device Inc.

Kester, Walt (1992). *Differential Drivers for High Speed ADCs Overview.* Rev. 0, Tutorial MT-075, Analog Device Inc.

Kester, Walt (1992). *Differential Drivers for Precision ADCs.* Rev. 0, Tutorial MT-074, Analog Device Inc.

Kester, Walt (1992). *ADC Architectures V: Pipelined Sub-ranging ADCs.* Rev. A, Tutorial MT-024, Analog Device Inc.

Kester, Walt, James Bryant & Mike Byrne (1992). *Grounding Data Converters and Solving the Mystery of "AGND" and "DGND".* Rev. A, Tutorial MT-031, Analog Device Inc.

Kester, Walt (1993). *Decoupling Techniques.* Rev. 0. Tutorial MT-101. Analog Device Inc.

Koen, Mike (1991). *High-speed Data Conversion.* Application Bulletin, June 1991, U.S.A, Burr-Brown Co.

McClendon, Heather & Kal Mustafa (2004). *Small Package, Big Performance.* Application Report, Texas Instruments Inc.

Miller, John (2009). *Driving High-Speed Analog-to-Digital Converters: Circuit Topologies and System-Level Parameters*. Rev. Sep. 2010, Application Report, Texas Instruments Inc.

Miller, John (2009). *Driving High-Speed, Analog-to-Digital Converters - Part I, Circuit Topologies and System-Level Parameters*. Application Report, Texas Instruments Inc.

Mustafa, Kal & Chris Sterzik (2003). *DC-Coupling Between Differential LVPECL, LVDS, HSTL, and CM*. Application Report, Texas Instruments Inc.

Neu, Thomas (2010). *Power-supply design for high-speed ADCs*. Analog Applications Journal, Texas Instruments Inc.

Ramus, Xavier (2004). PCB Layout for Low Distortion High-Speed ADC Drivers. Application Report, Texas Instruments Inc.

Rawat, M. Singh (2006). *Dependency Sequences and Hierarchical Clocks: Efficient alternatives to Vector Clocks for Mobile Computing Systems*. India, Indian Institute of Technology Bombay.

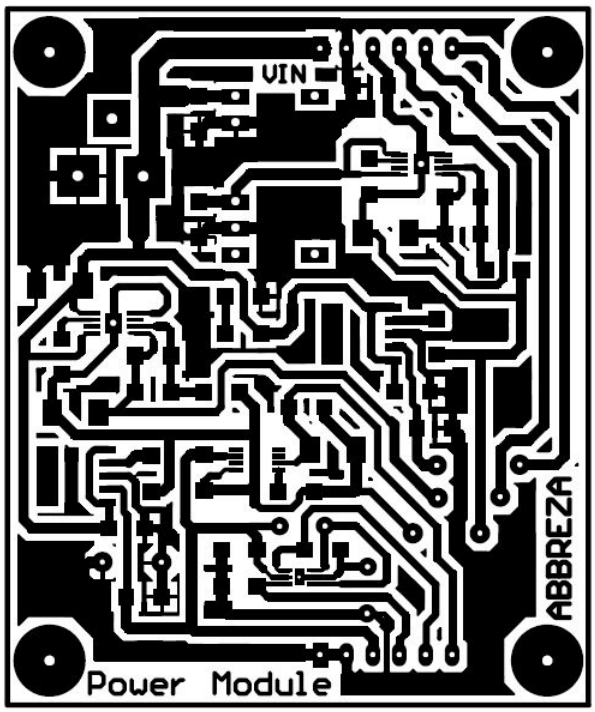
Steffes, Michael (2003). *RLC Filter Design for ADC Interface Applications*. Rev. Jan. 2005. Application Report, Texas Instruments Inc.

Steffes, Michael & Xavier Ramus (2005). *Low-power, high-intercept interface to the ADS5424 14-bit, 105-MSPS converter for undersampling applications*. Analog Applications Journal, Texas Instruments Inc.

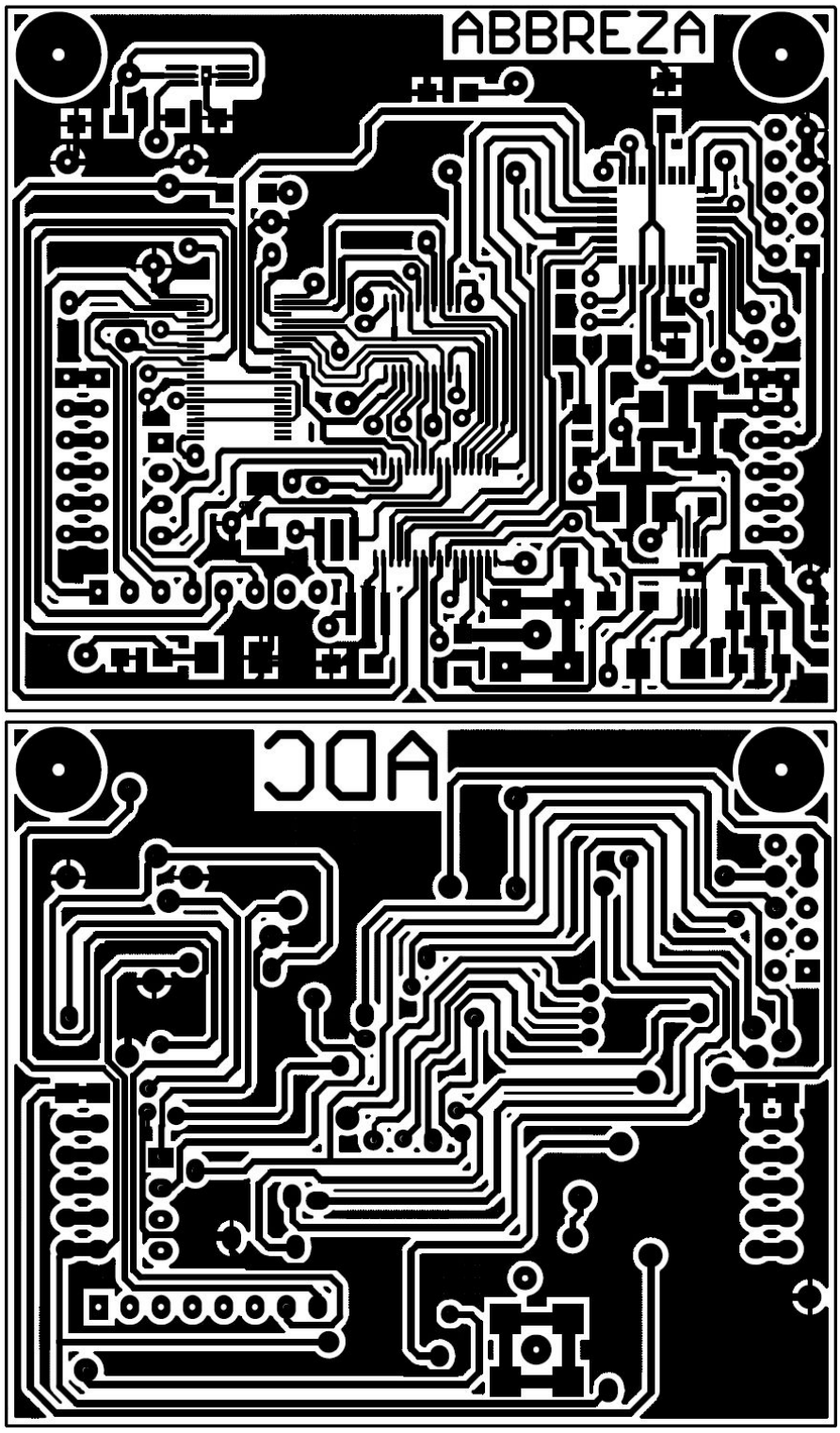
Weiler, Alexander & Alexander Pakosta (2006). *High-Speed Layout Guidelines*. Texas Instruments Inc.

APPENDICES

APPENDIX 1. Power supplier module PCB top layer, 57mm*67.6mm



APPENDIX 2. ADC module top and bottom PCB layout, 67.5mm*57.2mm



APPENDIX 1. DFC module top and bottom PCB layouts, 67.8mm*93.2mm