LOW BIT RATE SPEECH CODING USING TMS320C6416 DSP PROCESSOR

Oleh

Mohamad Habib Bin Mahamad Haniffah

Disertasi ini dikemukakan kepada

UNIVERSITI SAINS MALAYSIA

Sebagai memenuhi sebahagian daripada syarat keperluan

untuk ijazah dengan kepujian

SARJANA MUDA KEJURUTERAAN (KEJURUTERAAN ELEKTRONIK)

Pusat Pengajian Kejuruteraan
Elektrik dan Elektronik
Universiti Sains Malaysia

Mac 2005

ABSTRACT

The title of the project is Low Bit Rate Speech Coding Using TMS320C6416 DSP Processor. The scope of this project is divided into two main parts. Part one involves the study of the TMS320C6416 DSP processor. My task was to understand the architecture of this board and complete the tutorials in Code Composer Studio (CCS). The second part is concerned with the sampling of speech signal (analog signal) at different sampling frequencies and to study its effects on the quality of the reconstructed speech signal. Initially MATLAB and SIMULINK were used to sample the speech file and to study the effect of variation in sampling frequency on the quality of the speech signal and its waveform. Later, the sampling process is implemented in real time using the TMS320C6416 DSP Processor. Three sampling frequencies were chosen which are 8000 Hz, 4000 Hz and 2000 Hz. The results were divided into two sections; before real-time implementation and after real-time implementation. The comparison of the quality of the sampled audio signal was carried out for the three sampling frequencies as mentioned earlier. Two methods were used to measure the quality of the reconstructed audio signal. First, fifteen students were chosen to rate their score for the quality of the reconstructed signal. The score range was from 1(bad) to 5(excellent). Secondly, scope was used to display the waveform of the original and reconstructed signal. The results showed that the quality of the sound degrades from 8000 Hz to 2000 Hz.

ABSTRAK

Tajuk projek ini adalah Pengekod Suara Bit Rendah Menggunakan Pemproses Isyarat Digit TMS320C6416 (Low Bit Rate Speech Coding Using TMS320C6416 DSP Processor). Skop projek ini boleh di bahagikan kepada dua bahagian. Bahagian pertama melibatkan pengenalan kepada pemproses TMS320C6416. Bahagian ke dua projek ini melibatkan pensampelan isyarat suara pada frekuensi pensampelan yang berlainan dan mengenalpasti perbezaan di antara kualiti suara yang dihasilkan selepas pensampelan. Pada mulanya, Matlab dan Simulink digunakan untuk proses pensampelan dan mengenalpasti kesan frekuensi pensampelan terhadap kualiti isyarat yang dihasilkan. Kemudian, proses pensampelan diteruskan dengan menggunakan pemproses TMS320C6416. Tiga frekuensi pensampel di pilih iaitu 8000 Hz, 4000 Hz dan 2000 Hz. Bahagian keputusan di bahagikan kepada dua bahagian iaitu sebelum masa nyata dan selepas masa nyata (menggunakan TMS320C6416). Perbandingan di buat di antara kualiti isyarat suara yang dihasilkan pada frekuensi pensampelan yang berlainan. Dua kaedah telah digunakan untuk mengukur kualiti isyarat yang dihasilkan. Kaedah pertama ialah dengan meminta 15 orang pelajar memberikan skor mereka terhadap kualiti isyarat yang dihasilkan. Julat skor yang dipilih ialah di antara 1(sangat tidak memuaskan) hingga 5(sangat bagus). Kaedah ke dua ialah dengan menggunakan osiloskop untuk memaparkan bentuk isyarat yang dihasilkan dan membandingkannya dengan isyarat asal. Keputusan memunjukkan bahawa kualiti isyarat berkurangan dari 8000 Hz ke 2000 Hz.

ACKNOWLEDGEMENT

I would like to express my sincere thanks to my supervisor, Professor Farid Ghani, for his guidance and support in completing my final year project successfully at University Science Malaysia. Also, I am thankful to Mr. Suardi for his fruitful help in learning the Code Composer Studio software in a short period of time. This research could not have been possible without their technical expertise, critical insight and enlightening suggestions.

Moreover, I acknowledged all my fellow undergraduate students in the Digital Signal Processing Laboratory for their encouragement and companionship. Special thanks go to Digital Signal Processing Laboratory technicians, Mr. Hisham and Mr. Amir who constantly gave me both technical and non-technical advice.

I am thankful to my friends Siva Kumar,Pramila, Ronald Reegan, Sofian, Awie, Nuril Syahira, Amri, Nitiabalan, Yeap Teoh Shyen, Subashini, Prakash Babu, Anita, Pragash, Rozi and Rasyid who participated in the listening tests for this project.

My deepest gratitude goes to my respective family for their continuous support and encouragement.

Please feel free to write or send me e-mail:

Mohamad Habib B. Mahamad Haniffah
Science University of Malaysia
habib_usm@hotmail.com

CONTENTS **PAGE**

LIST OF FIGURES AND TABLES

CHAPTER 1
INTRODUCTION

1.1 Digital Signal Processing

Digital signal processing (DSP) involves the manipulation of digital signals in order to extract useful information from them. A digital signal is formed by sampling and quantizing an analog signal. The digitization process is achieved via an analog-to-digital (A/D) converter. An A/D converter converts analog voltage values into discrete voltage values. Many transducers and sensors generate analog signals. Therefore, some sort of an A/ D converter is normally needed at the front-end of a DSP system, as is shown in Figure 1.1.

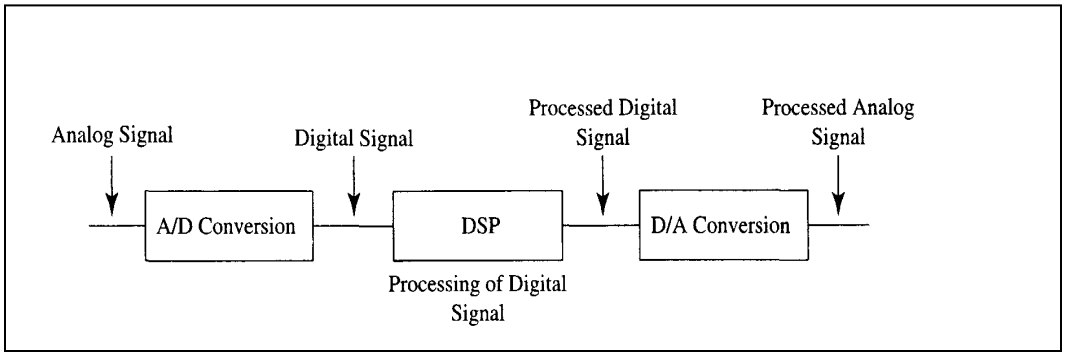


Figure 1.1: DSP system main components.

There are many reasons why one would want to process an analog signal in a digital fashion by converting it into a digital signal. The main reason is that digital processing allows programming flexibility. The same DSP hardware can be used for many different applications by simply changing the code residing in memory. Another reason is that digital circuits provide a more stable and tolerant output as compared with analog circuits, for example, when subjected to temperature changes.

In addition, the advantage of operating in the digital domain could be intrinsic. For example, a linear-phase filter can only be designed by using digital signal processing techniques, and many adaptive systems are achievable only via digital manipulation of signals. In essence, digital representation (0's and 1's) allows voice, audio, and video

data to be treated the same for transmission and storage purposes. As a result, digital processing, and hence digital signal processors (also called DSPs) are expected to play a major role in the next generation of high-speed communication links, including cable (cable modem) and telephone lines (Digital Subscriber Lines, or DSLs). [Simon Haykin, 1999]

1.2 Speech Coding

Speech communication is at present the most dominant and common service in telecommunication networks. Analogue telecommunication of speech is a cumbersome and inflexible process when transmission power and spectral utilization, the foremost resources in any communication system, are considered. Digital transmission of speech is more versatile which provides the opportunity of achieving lower costs, consistent quality, security and spectral efficiency in the systems that exploit it.

The first stage in the digitalization of speech involves sampling and quantization. The minimum sampling frequency is limited by the Nyquist criterion. The emergence of new mobile telecommunication systems for which spectral efficiency is a prime mover has encouraged the need for more and more reductions in the pulse code modulation(PCM) bit rate of speech signals. [John Bellamy, 1990]

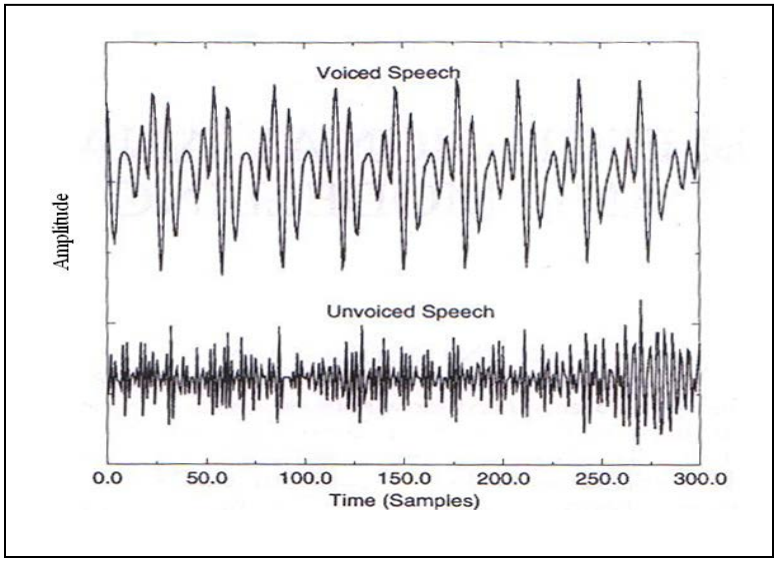


Figure 1.2: Voiced and unvoiced speech waveform. [A.M.Kondoz, 2000]

1.3 The Need For Speech to be Digitized

Before considering methods of speech coding at low bit rates, it is important to know why speech needs to be digitized and why low bit rates are preferable to high bit rates. The reasons for requiring speech to be in digital form are as follows:

a) **Multiplexing** – when a number of signals must be transmitted along a common circuit, the most common methods of multiplexing use digital techniques to provide negligible inter-channel interference and minimum distortion of data signals.

b) **Switching** – when a signal is in digital form it can be passed through a large number of switching nodes without incurring any degradation or distortion. Circuit switches (telephone exchanges) can be made very compact when the signals are digital.

c) **Storage** – In general, digital information is much easier to store than equivalent analogue information due to the wide range of memories and other digital storage devices that are now available.

d) **Packet Transmission** – Modern packet transmission systems combine storage, multiplexing and switching techniques to achieve very efficient use of transmission resources. All these techniques are most easily implemented with digitized signals.

e) **Encryption** – In order to prevent unauthorized interception of messages, the only really secure techniques of encryption involve numerical randomization of a digitized version of the message.

f) **Spread Spectrum Transmission** – spreading of the spectrum of a signal can provide valuable protection against a wide range of interference and electronic countermeasures. However, the signal modifications introduced to spread a signal

spectrum at the transmitter, can most easily be removed at the receiver without introducing additional distortion if the signal is in digital form. [Frank Fallside]

1.4 Reasons For Low Bit Rate Speech Coding:

Speech coding at low bit rates has become a very important research due to its advantages. These low rates allow digitized speech to be passed over narrow bandwidth channels such as 3 kHz analogue telephone circuits and HF radio circuits. Low bit rates provide a great ability to overcome channel distortions and means that simpler modems can be used. High bit rates usually require high transmitter power. So, the use of low bit rates can reduce this problem. Moreover, they allow more channels to be passed through a given capacity of multiplexed circuit or through a given circuit switch or packet network. In addition to that, the use of low bit rates can reduce the memory which is important to store a given quantity of speech, such as required in automated switch announcement systems. Furthermore, they provide greater resistance to noise, interference and jamming when combined with error correction and/or spread spectrum coding schemes. [A.M.Kondoz, 2000]

1.5 Problems in Speech Coding at Low Bit Rates

Digital coding of speech and the bit rate reduction process has thus emerged as an important area of research, largely addressing the following problems:

- It is very attractive to decrease the PCM (Pulse Code Modulation) bit rate as much as possible, but it becomes increasingly difficult to maintain acceptable speech quality as the bit rate falls.

- As the bit rate falls, acceptable speech quality can only be maintained by:
    - Employing very complex algorithms which are difficult to implement in real-time even with the new fast processors with their associated high cost and power consumption,

- Incurring excessive delay which might induce echo control problems elsewhere in the system.

➢ For low bit rate, parameters of a speech production and / or perception model are encoded and transmitted. These parameters, however are extremely sensitive to corruption. On the other hand, the system in which these coders are needed typically operate on highly degraded channels, raising the acute problem of maintaining acceptable speech quality from sensitive speech parameters, even in good channel conditions. [Sadaoki Furui,2001]

1.6 Method of Coding

Voice digitization techniques can be broadly categorized into two classes: those digitally encoding analog waveforms as faithfully as possible and those processing waveforms to encode only the perceptually significant aspects of speech and hearing processes. The first category is representative of the general problem of analog-to-digital and digital-to-analog conversions and is not restricted to speech digitization. The three most common techniques used to encode a voice waveform are as follows:

- Pulse code modulation(PCM),
- Differential PCM (DPCM), and
- Delta modulation (DM).

Basically, when talking about speech encoding techniques we are, in fact, investigating the more general realm of analog-to-digital conversion. The second category of speech digitization is concerned primarily with producing very low data rate speech encoders and decoders for narrowband transmission systems or digital storage devices with limited capacity. A device from this special class of techniques is commonly referred to as a "vocoder" (voice coder). [John Bellamy, 1990]

1.7 Scope of the Project:

The scope of this project can be divided into two main parts. Part one involves the study of the TMS320VC6416 DSP processor. The second part is concerned with the sampling

of speech signal (analog signal) at different sampling frequencies and to study its effects on the quality of the reconstructed speech signal. The sampling frequency must be kept below the Nyquist rate in order to reduce the bit rate of the coded speech signal. Initially MATLAB and SIMULINK are used to sample the speech file and to study the effect of variation in sampling frequency on the quality of the speech signal and its waveform. Later, the sampling process is implemented in real time using the DSP Processor. This project concerned on the understanding of analog signals, digital signals, various type of filters and as well as the quality measurement of sampled audio signal.

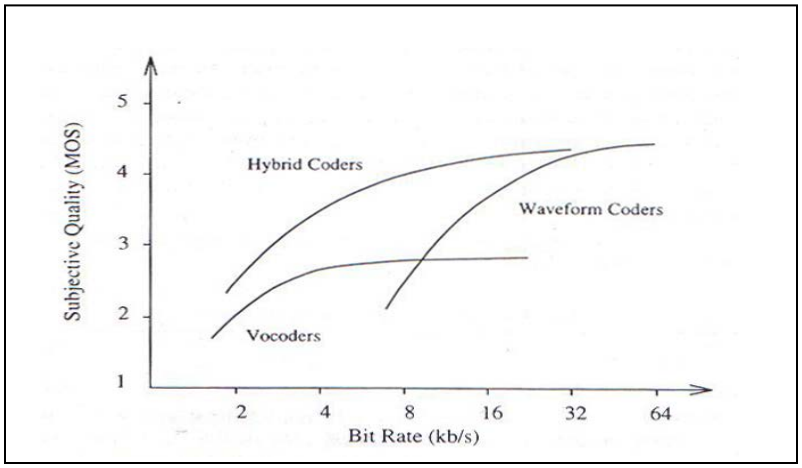


Figure 1.3 : Quality comparison of speech coding schemes. [A.M.Kondoz,2000]

1.8 Methodology

The project will be carried under the following main steps:

a) *Study of the TMS320VC6416 DSP Processor and its architecture.*

The TMS320VC6416 DSP Starter Kit is available at the DSP Lab of the Electrical & Electronic Department. Actually, there are three types/versions of the DSP processor. First, for application that focuses on digital control, the TMS320C2000 platform offers the most controlled optimized DSPs in the world. For application that requires low power consumption the TMS320C5000 platform offers good power efficient performance. For high performance applications, the MS320C6000 platform of DSPs is considered the highest performance DSPs in the world.

b) *Study of Pulse Code Modulation for coding speech signals.*

Pulse Code Modulation (PCM) is a common technique used in coding of analog signals because it provides excellent quality for all types of input signals (e.g., voice or data) at a moderate data rate (64 kb/s) and a moderate cost. PCM uses discrete sample times with analogue sample amplitudes to extract the information in a continuously varying analogue signal. PCM system is inherently capable of encoding an arbitrarily random waveform whose maximum frequency component does not exceed one-half the sampling rate. Furthermore, the coded speech from a 8 bit PCM system with a sampling frequency of 8 kHz is generally used as the reference for comparing lower rate speech coders, as its performance is considered to be toll quality.

c) *Sampling of speech files using MATLAB and SIMULINK.*

In this case, speech files were sampled at different frequencies using Matlab and Simulink. Chosen frequencies were at 8 kHz, 4 kHz and 2 kHz. 8 kHz is the standard sampling frequency for speech coding in digital telephony but due to high demand in low bit rate speech coding, 4 kHz and 2 kHz were sampled as well to see their performance.

d) *Study of the effect of variation of sampling frequency on the quality and waveform of reconstructed speech.*

As mentioned previously, three sampling frequencies were selected which are 8 kHz, 6 kHz and 4 kHz. These frequencies were tested independently to see their performance/quality. The importance of this testing was to verify whether the quality of the sampled speech is maintained while reducing the sampling frequency. The testing was done by hearing the output speech created and then comparing the shape of the reconstructed waveform to the original one.

e) *Installation and testing of the DSP Processor*

Installation and testing of the DSP Processor was carried out at the DSP Lab of the Electrical & Electronic Department.

f) Real time realization of the sampling and reconstruction process on the DSP Processor.

A real-time process is a task which needs to be performed within a specified time limit. In digital speech coding at 8 kHz sampling rate, the real-time processing needs to be performed within 125 μs for coders involving the sample-by-sample coding process. Real-time implementation is very important from a cost point of view. Any speech coding algorithm can be implemented using available digital signal processor (DSP) chip technology, but the cost of that implementation will increase rapidly with the increase in the number of DSP chip used. The other important consideration in real-time implementation is the power consumption of the final product.

1.9 Relevance of the Study:

Digital Signal Processing (DSP) has experienced an enormous growth in the last 20 years. Nowadays DSP processors are used in a wide variety of products from cellular phones to motor drives. These processors are expected to play a major role in the next generation of high-speed communication networks. Moreover, Speech coding at low bit rates has become a very important research due to its advantages. These low rates allow digitized speech to be passed over narrow bandwidth channels such as 3 kHz analogue telephone circuits and HF radio circuits. Low bit rates provide a great ability to overcome channel distortions and means that simpler modems can be used. High bit rates usually require high transmitter power. So, the use of low bit rates can reduce this problem. Moreover, they allow more channels to be passed through a given capacity of multiplexed circuit or through a given circuit switch or packet network. In addition to that, the use of low bit rates can reduce the memory which is important to store a given quantity of speech, such as required in automated switch announcement systems. Hence the proposed study is quite relevant to the present day applications.

CHAPTER 2
ARCHITECTURE OF THE TMS320C6416 DSP PROCESSOR

2.1 Overview

The TMS320C6416 DSP Starter Kit (DSK) developed jointly with Spectrum Digital is a low-cost development platform designed to speed the development of high performance applications based on TI´s TMS320C64x DSP generation. The kit uses USB communications for true plug-and-play functionality. Both experienced and novice designers can get started immediately with innovative product designs with the DSKs full featured Code Composer Studio v2.2 IDE and eXpressDSP Software which includes DSP/BIOS and Reference Frameworks.

All users will benefit from the eXpressDSP for Dummies textbook featured for the first time in this DSK. The C6416 DSK tools includes the latest fast simulators from TI and access to the Analysis Toolkit via Update Advisor which features the Cache Analysis tool and Multi-Event Profiler. Using Cache Analysis developers improve the performance of their application by optimizing cache usage. By providing a graphical view of the on-chip cache activity over time the user can quickly determine if their code is using the on-chip cache to get peak performance.

The C6416 DSK allows us to download and step through code quickly and uses Real Time Data Exchange (RTDX™) for improved Host and Target communications. The DSK utilities include Flashburn to program flash, Update Advisor to download tools, utilities and software and a power on self test and diagnostic utility to ensure the DSK is operating correctly. [TMS320C6416 DSK Technical Reference,2000]

2.1.1 DSP Starter Kit (DSK) for the TMS320VC6416.

The TMS320VC6416 Starter Kit contains:

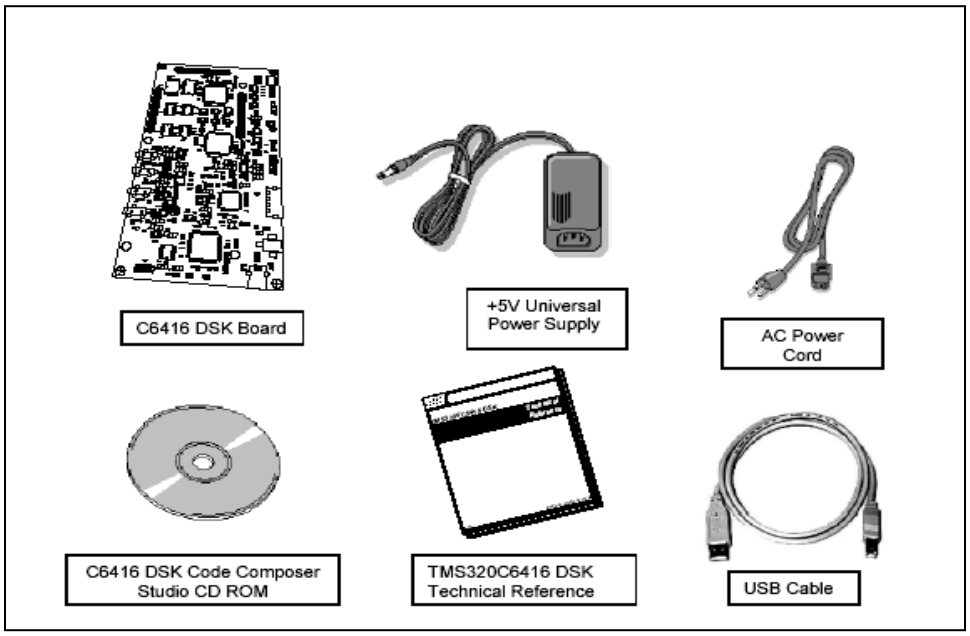


Figure 2.1: TMS320VC6416 Starter Kit [TMS320C6416 DSK Technical Reference]

2.1.2 Compatibility

System Hardware and Software Requirements:

These operating platform requirements are necessary to install the code composer studio (CCS) Integrated Development and support the USB port. The requirements for the operating platform are:

- **Minimum hardware configuration:**
  - 233 MHz or faster Pentium or compatible,
  - 600 MB of free hard disk space,
  - Microsoft Windows 98SE, 2000(SP1 or higher), or XP,
  - 64 MB of RAM,
  - SVGA (640 x 480) color display.

- **Recommended Hardware Configuration:**
  - 128 MB of RAM,
  - SVGA (1024 X 768) color display,
  - 500 MHz or faster Pentium or compatible.

2.2 Board Tour

The figure below illustrates the main sections of the board:

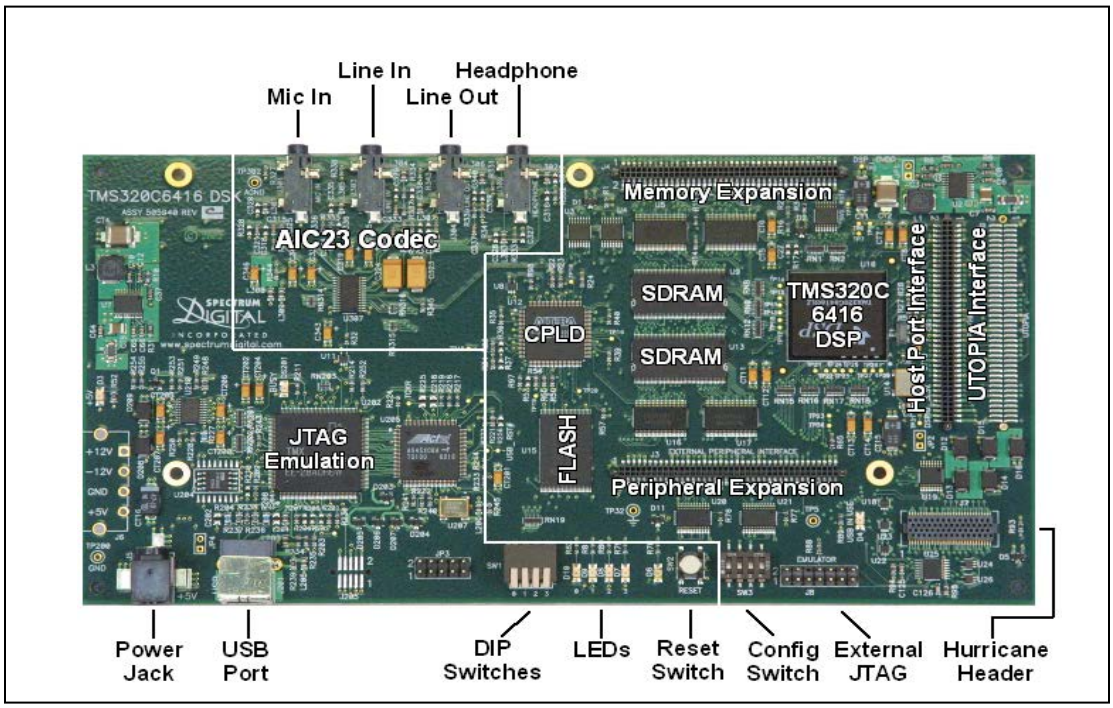


Figure 2.2: Main sections of the C6416 board.

The TMS320C6416 DSP is the heart of the system. It is a core member of Texas Instruments' C64X line of fixed point DSPs whose distinguishing features are an extremely high performance 600MHz VLIW DSP core and a large amount of fast on-chip memory (1Mbyte). On-chip peripherals include two independent external memory interfaces (EMIFs), 3 multi-channel buffered serial ports (McBSPs), three on-board timers and an enhanced DMA controller (EDMA). The 6416 represents the high end of TI's C6000 integer DSP line both in terms of computational performance and on-chip resources.

The C6416 has a significant amount of internal memory so typical applications will have all code and data on-chip. External accesses are done through one of the EMIFs, either the 64-bit wide EMIFA or the 16-bit EMIFB. EMIFA is used for high bandwidth memories such as the SDRAM while EMIFB is used for non-performance critical devices such as the Flash memory that is loaded at boot time. A 32-bit subset of EMIFA is brought out to standard TI expansion bus connectors so additional functionality can be added on daughter card modules.

DSPs are frequently used in audio processing applications so the DSK includes an on-board codec called the AIC23. Codec stands for coder/decoder, the job of the AIC23 is to code analog input samples into a digital format for the DSP to process, and then decode data coming out of the DSP to generate the processed analog output. Digital data is sent to and from the codec on McBSP2.

The DSK has *4 light emitting diodes (LEDs)* and 4 DIP switches that allow users to interact with programs through simple LED displays and user input on the switches. Many of the included examples make use of these user interface options.

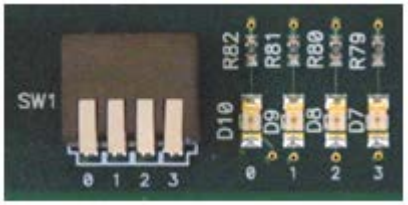


Figure 2.3: Four light emitting diodes (LED's) of C6416 DSK.

The DSK implements the logic necessary to tie board components together in a programmable logic device called a CPLD. In addition to random glue logic, the CPLD implements a set of 4 software programmable registers that can be used to access the on-board LEDs and DIP switches as well as control the daughter card interface.

2.2.1 Development Environment

Code Composer Studio is TI's flagship development tool. It consists of an assembler, a C compiler, an integrated development environment (IDE, the graphical interface to the tools) and numerous support utilities like a hex format conversion tool. The DSK

includes a special version of Code Composer specially tailored to features on the 6416 DSK board.

The Code Composer IDE consists of an editor for creating source code, a project manager to identify the source files and options necessary for the programs and an integrated source level debugger that lets us examine the behavior of our program while it is running. The IDE is responsible for calling other components such as the compiler and assembler so developers don't have to deal with the hassle of running each tool manually.

The 6416 DSK includes a special device called a JTAG emulator on-board that can directly access the register and memory state of the 6416 chip through a standardized JTAG interface port. When a user wants to monitor the progress of his program, Code Composer sends commands to the emulator through its USB host interface to check on any data the user is interested in.

This debugging method is extremely powerful because programs can be debugged unobtrusively on real hardware targets without making any special provisions for debug like external probes, software monitors or simulated hardware. When designing our own hardware around the 6416 we can debug our application with the same rich functionality of the DSK simply by using Code Composer with an external emulator and including a header for the JTAG interface signals.

We should always be aware that the DSK as a different system than our PC, when we recompile a program in Code Composer on our PC we must specifically load it onto the 6416 on the DSK. Other things to be aware of are:

- When we tell Code Composer to run, it simply starts executing at the current program counter. If we want to restart the program, we must reset the program counter by using Debug → Restart or re-loading the program which sets the program counter implicitly.

- After we start a program running it continues running on the DSP indefinitely. To stop it we need to halt it with Debug → Halt.

2.2.2 Key Features

The C6416 is a low-cost standalone development platform that enables users to evaluate and develop applications for the TI C64xx DSP family. The DSK also serves as a hardware reference design for the TMS320C6416 DSP.

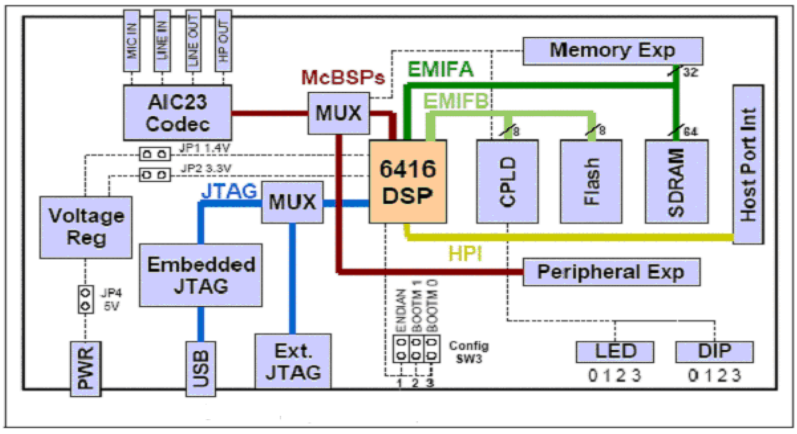


Figure 2.4: Block diagram of C6416 DSK. [TMS320C6416 DSK Technical Reference]

The DSK comes with a full compliment of on-board devices that suite a wide variety of application environments. Key features include:

- A Texas Instruments TMS320C6416 operating at 600 MHz.
- An AIC23 stereo codec
- 16 Mbytes of synchronous DRAM
- 512 Kbytes of non-volatile Flash memory
- 4 user accessible LEDs and DIP switches
- Software board configurations through registers implemented in CPLD
- Configurable boot options
- Standard expansion connectors for daughter card use
- JTAG emulation through on-board JTAG emulator with USB host interface or external emulator

- Single voltage power supply (+5V)

2.3 Functional Overview of the TMS320C6416

The DSP on the 6416 DSK interfaces to on-board peripherals through one of two busses, the 64-bit wide EMIFA and the 8-bit wide EMIFB. The SDRAM, Flash and CPLD are each connected to one of the busses. EMIFA is also connected to the daughter card expansion connectors which are used for third party add-in boards. An on-board AIC23 codec allows the DSP to transmit and receive analog signals. McBSP1 is used for the codec control interface and McBSP2 is used for data. Analog I/O is done through four 3.5mm audio jacks that correspond to microphone input, line input, line output and headphone output. The codec can select the microphone or the line input as the active input. The analog output is driven to both the line out (fixed gain) and headphone (adjustable gain) connectors. McBSP1 and McBSP2 can be re-routed to the expansion connectors in software.

A programmable logic device called a CPLD is used to implement glue logic that ties the board components together. The CPLD also has a register based user interface that lets the user configure the board by reading and writing to the CPLD registers. The DSK includes 4 LEDs and 4 position DIP switch as a simple way to provide the user with interactive feedback. Both are accessed by reading and writing to the CPLD registers.

An included 5V external power supply is used to power the board. On-board switching voltage regulators provide the 1.4V DSP core voltage and 3.3V I/O supplies. The board is held in reset until these supplies are within operating specifications. A separate regulator powers the 3.3V lines on the expansion interface. Code Composer communicates with the DSK through an embedded JTAG emulator with a USB host interface. The DSK can also be used with an external emulator through the external JTAG connector. [TMS320C6416 DSK Technical Reference]

2.4 Basic Operation

The DSK is designed to work with TI's Code Composer Studio development environment and ships with a version specifically tailored to work with the board. Code Composer communicates with the board through the on-board JTAG emulator. To start, follow the instructions in the Quick Start Guide to install Code Composer. This process will install all of the necessary development tools, documentation and drivers. After the install is complete, these steps are followed to run Code Composer. The DSK must be fully connected to launch the DSK version of Code Composer.

1) Included power supply is connected to the DSK.
2) DSK is connected to the PC with a standard USB cable (also included).
3) Code Composer is launched from its icon on the desktop.

Detailed information about the DSK including a tutorial, examples and reference material is available in the DSK's help file. It is accessible through Code Composer's help menu.

2.4.1 Memory Map

The C64xx family of DSPs has a large byte addressable address space. Program code and data can be placed anywhere in the unified address space. Addresses are always 32-bits wide. The memory map shows the address space of a generic 6416 processor on the left with specific details of how each region is used on the right. By default, the internal memory sits at the beginning of the address space. Portions of memory can be remapped in software as L2 cache rather than fixed RAM.

Each EMIF (External Memory Interface) has 4 separate addressable regions called chip enable spaces (CE0-CE3). The SDRAM occupies CE0 of EMIFA while the CPLD and Flash are mapped to CE0 and CE1 of EMIFB respectively. Daughter cards use CE2 and CE3 of EMIFA. [TMS320C6416 DSK Technical Reference,2003]

Refer Attachment D for memory map of C6416 diagram.

2.4.2 Configuration Switch Settings

The DSK has 3 configuration switches that allow users to control the operational state of the DSP when it is released from reset. The configuration switch block is labeled SW3 on the DSK board, next to the reset switch.

Configuration switch 1 controls the endianness of the DSP while switches 2 and 3 configure the boot mode that will be used when the DSP starts executing. By default all switches are off which corresponds to EMIFB boot (out of 8-bit Flash) in little endian mode. The figure below shows these settings.

Table 2.1: Configuration Switch Settings.

Switch 3	Switch 2	Switch 1	Configuration Description
Off	Off		EMIF boot from 8-bit Flash
Off	On		No Boot
On	Off		Reserved
On	On		HPI boot
		Off	Little endian
		On	Big endian

2.4.3 Power Supply

The DSK operates from a single +5V external power supply connected to the main power input (J5). Internally, the +5V input is converted into +1.4V and +3.3V using a dual voltage regulator. The +1.4V supply is used for the DSP core while the +3.3V supply is used for the DSP's I/O buffers and all other chips on the board. The power connector is a 2.5mm barrel-type plug.

2.5 Board Components

2.5.1 CPLD (Programmable Logic)

The C6416 DSK uses an Altera EPM3128TC100-10 Complex Programmable Logic Device (CPLD) device to implement:

- 4 Memory-mapped control/status registers that allow software control of various board features.
- Address decode and memory access logic.
- Control of the daughter card interface and signals.
- Assorted "glue" logic that ties the board components together.

2.5.2 CPLD Overview

The CPLD logic is used to implement functionality specific to the DSK. Our own hardware designs will likely implement a completely different set of functions or take advantage of the DSPs high level of integration for system design and avoid the use of external logic completely.

The CPLD implements simple random logic functions that eliminate the need for additional discrete devices. In particular, the CPLD aggregates the various reset signals coming from the reset button and power supervisors and generates a global reset. [TMS320C6416 DSK Technical Reference,2003]

2.5.3 CPLD Registers

The 4 CPLD memory-mapped registers allows users to control CPLD functions in software. On the 6416 DSK the registers are primarily used to access the LEDs and DIP switches and control the daughter card interface. The registers are mapped into EMIFB data space at address 0x60000000. They appear as 8-bit registers with a simple asynchronous memory interface.

2.5.4 USER_REG Register

USER_REG is used to read the state of the 4 DIP switches and turn the 4 LEDs on or off to allow the user to interact with the DSK. The DIP switches are read by reading the top 4

bits of the register and the LEDs are set by writing to the low 4 bits.

2.5.5 DC_REG Register

DC_REG is used to monitor and control the daughter card interface. DC_DET detects the presence of a daughter card. DC_STAT and DC_CNTL provide simple communications with the daughter card through readable status lines and writable control lines.

2.5.6 MISC Register

The MISC register is used to provide software control for miscellaneous board functions. On the 6416 DSK, the MISC register controls how auxiliary signals are brought out to the daughter-card connectors.

McBSP1 and McBSP2 are usually used as the control and data ports of the on-board AIC23 codec. The power-on state of these bits (both 0s) represents that configuration. Set MCBSP1SEL or MCBSP2SEL to route the McBSPs to the daughter card connectors rather than the codec.

The Flash and CPLD share CE1 which means that the highest address bit (A21) is used to differentiate between the two. In this configuration 512Kbytes of 8-bit Flash are visible at the beginning of CE1 which matches the chip on the production board. If the Flash is replaced with a 1Mbyte chip, only 512Kbytes of Flash will still be visible but FLASH_PAGE can be used to select between the top and bottom halves.

FLASH_PAGE replaces the address bit (A21) that is lost sharing CE1 with the CPLD. The 6416's PCI interface and McBSP2 share some pins. The McBSP2_EN signal is used to disable McBSP2 when the PCI interface is active. McBSP2_EN is generated on the board when an appropriate daughter card that uses PCI is plugged in, it can be read through this CPLD bit. The scratch bits are unused. They can be set to any value.

2.5.7 AIC23 Codec

The DSK uses a Texas Instruments AIC23 (part #TLV320AIC23) stereo codec for input and output of audio signals. The codec samples analog signals on the microphone or line

inputs and converts them into digital data so it can be processed by the DSP. When the DSP is finished with the data it uses the codec to convert the samples back into analog signals on the line and headphone outputs so the user can hear the output.

The codec communicates using two serial channels, one to control the codec's internal configuration registers and one to send and receive digital audio samples. McBSP1 is used as the unidirectional control channel. It should be programmed to send a 16-bit control word to the AIC23 in SPI format. The top 7 bits of the control word should specify the register to be modified and the lower 9 should contain the register value. The control channel is only used when configuring the codec, it is generally idle when audio data is being transmitted.

McBSP2 is used as the bi-directional data channel. All audio data flows through the data channel. Many data formats are supported based on the three variables of sample width, clock signal source and serial data format. The DSK examples generally use a 16-bit sample width with the codec in master mode so it generates the frame sync and bit clocks at the correct sample rate without effort on the DSP side. The preferred serial format is DSP mode which is designed specifically to operate with the McBSP ports on TI DSPs.

The codec has a 12MHz system clock. The 12MHz system clock corresponds to USB sample rate mode, named because many USB systems use a 12MHz clock and can use the same clock for both the codec and USB controller. The internal sample rate generate subdivides the 12MHz clock to generate common frequencies such as 48kHz, 44.1kHz and 8kHz. The sample rate is set by the codec's SAMPLERATE register. The figure below shows the codec interface on the C6416 DSK.

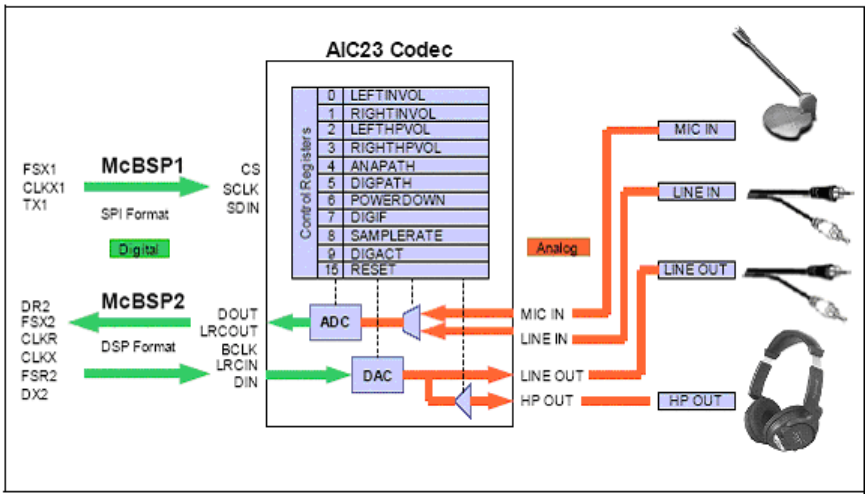


Figure 2.5: TMS320C6416 Codec Interface [TMS320C6416 DSK Technical Reference,2003]

2.5.8 Synchronous DRAM

The DSK uses a pair of industry standard 64 megabit SDRAMs in CE0 of EMIFA. The two devices are used in parallel to create a 64-bit wide interface. Total available memory is 16 megabytes. The DSK uses an EMIFA clock of 100MHz. The integrated SDRAM controller is started by configuring the EMIF in software. Timings can be found in the SDRAM datasheet and the DSK help file. When using the SDRAM, note that one row of the memory array must be refreshed at least every 15.6 microseconds to maintain the integrity of its contents.

2.5.9 Flash Memory

The DSK uses a 512Kbyte external Flash as a boot option. It is connected to CE1 of EMIFB with an 8-bit interface. Flash is a type of memory which does not lose its contents when the power is turned off. When read it looks like a simple asynchronous read-only memory (ROM). Flash can be erased in large blocks commonly referred to as sectors or pages. Once a block has been erased each word can be programmed once through a special command sequence. After that the entire block must be erased again to change the contents. The Flash requires 70ns for both reads and writes. The general settings used with the DSK use 8 cycles for both read and write strobes (80ns) to leave a little extra margin.

2.5.10 LEDs and DIP Switches

The DSK includes 4 software accessible LEDs (D7-D10) and DIP switches (SW1) that provide the user a simple form of input/output. Both are accessed through the CPLD USER_REG register.

2.5.11 Daughter Card Interface

The DSK provides three expansion connectors that can be used to accept plug-in daughter cards. The daughter card allows users to build on their DSK platform to extend its capabilities and provide customer and application specific I/O. The expansion connectors are for memory, peripherals, and the Host Port Interface (HPI).

The memory connector provides access to the DSP's asynchronous EMIF signals to interface with memories and memory mapped devices. It supports byte addressing on 32 bit boundaries. The peripheral connector brings out the DSP's peripheral signals like McBSPs, timers, and clocks. Both connectors provide power and ground to the daughter card.

The HPI is a high speed interface that can be used to allow multiple DSPs to communicate and cooperate on a given task. The HPI connector brings out the HPI specific control signals as well as McBSP2.

Most of the expansion connector signals are buffered so that the daughter card cannot directly influence the operation of the DSK board. The use of TI low voltage, 5V tolerant buffers, and CBT interface devices allows the use of either +5V or +3.3V devices to be used on the daughter card.

Other than the buffering, most daughter card signals are not modified on the board. However, a few daughter card specific control signals like DC_RESET and DC_DET exist and are accessible through the CPLD DC_REG register. The DSK also multiplexes the McBSP1 and McBSP2 of on-board or external use. This function is controlled through the CPLD MISC register.

CHAPTER 3

VOICE SAMPLING BEFORE REAL-TIME IMPLEMENTATION

3.1 Simulink Model:

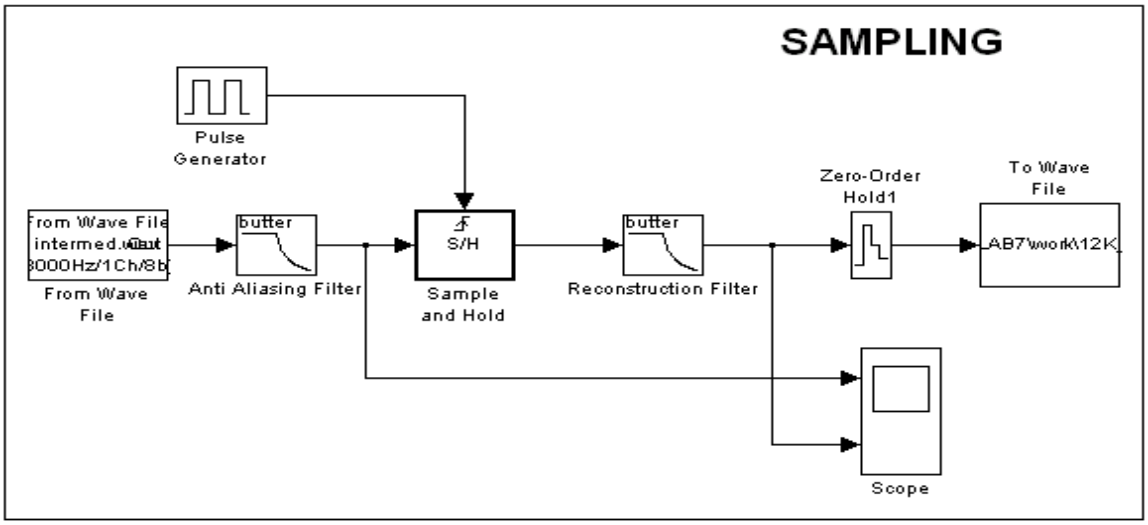


Figure 3.1: Simulink Model of Voice Sampling.

3.1.1 Model Description

A wave file from a human's voice saying 'HELLO' is sampled at three different frequencies which are 2000 Hz, 4000 Hz and 8000 Hz. The sampled wave file is saved at a desired folder so that the file can be accessed and the quality of the sound created after sampling can be measured by hearing it. Even tough, a scope is used to display the shape of the waveforms before and after sampling, the difference between before and after sampling is more identical and easily noticed by simply hearing to it.

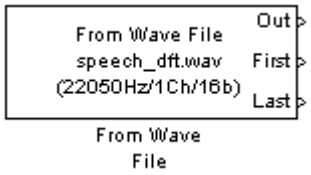
3.2 Block Description

Refer description for Pulse Generator, Sample and Hold and Zero Order Hold at page 39 -45.

3.2.1 From Wave File.

Purpose: Read audio data from a Microsoft Wave (.wav) file.

Symbol:



Library: Platform-specific I/O / Windows (WIN32)

Description:

The From Wave File block reads audio data from a Microsoft Wave (.wav) file and generates a signal with one of the data types and amplitude ranges in the following table:

Table 3.1: Output data type and its amplitude range.

Output Data Type	Output Amplitude Range
double	±1
single	±1
int16	-32768 to 32767 (-2^{15} to 2^{15} - 1)
uint8	0 to 255

The audio data must be in uncompressed pulse code modulation (PCM) format. The block supports 8-, 16-, 24-, and 32-bit Microsoft Wave (.wav) files. [MATLAB 7.0]