ISSN : 2231-0711
P.V. Ingole et al IJCSET | March 2011 | Vol 1, Issue 2,95-98

Implementation of Signal Generator (DSP) Using TMS 320 C 6713 DSK

P.V. Ingole, A. K. Sapkal , G.G. Sarate , S. R. Hirekhan

Instrumentation and DSP Laboratory, Electronics and Telecommunication Engineering Department,
Sant Gadge Baba Amravati University,
Government College of Engineering Amravati. (M. S.), India.

ABSTRACT

TMS 320 C 6713 DSK is used for the implementation of signal generator (DSP). The DSP processor TMS320C6713DSK with Code Composer Studio has been used to generate the signal waveforms and is used for interfacing user and the DSP board. A look-up table method has been used to generate signal waveforms and frequencies as well as amplitude of waveforms are independently adjustable. The user can program required waveform, amplitude, and frequency of the signal. The DSP processor has also been programmed to produce amplitude modulated signal. Simultaneously we can produce two different waveforms by using stereo audio codec of the kit.

Keywords: DSP starter kit (DSK), Texas Instrument (TI), Digital Signal Processor (DSP), Code Composer Studio (CCS).

I. INTRODUCTION

In measurement and instrumentation system signal generators play a significant role where analogue and digital systems are characterized by different types of waveforms. This signal generator is based on a digital signal processor to generate signals between 0.002 Hz and 15 kHz. In the proposed signal generator the lower frequency range is not limited by the external components but the upper frequency range is limited due to bandwidth of stereo codec AIC23.

In this paper we try to discuss implementation of signal generator which could function for various signal generators, DSP board TMS 320 C 6713, process of code building, methodology and signal generator applications.

The TMS320C6713DSK has main features such as fast data access, fast computation, numerical fidelity, fast execution control which make it suitable for various applications.[9][10] Texas Instrument (TI) DSP starter kit TMS320C6713DSK, which consists of a TMS320C6713 DSP chip and stereo codec TLV320AIC23, is used in design. The design algorithms are processed by DSP and the stereo codec TLV320AIC23 consisting of an analogue-to-digital converter (ADC) and a digital-to-analogue converter (DAC). [1] The processing of the signal is simple, the analogue signal coming from port mike in or line in then, it is sampled to digital signal by the codec. Then the sampled signal will be processed digitally by the DSP. This signal processing depends on how the user programs the DSP board. Then, the output signal will be release through line out or headphone. As mentioned before, signal processing by DSP depends on program algorithm given from user to the DSP processor. The process of writing a program in the DSP processor is as same as writing a program in other processors, which is using the machine language or assembler. As an alternative, we can also use C language to program the DSP processor. To development an application with DSK, Texas Instrument provides software called Code Composer Studio (CCS). CCS is software that is used for interfacing user and the DSP board. CCS provides a work environment that contains all of the tools needed to program the DSP processor [2]. The tools are integrated with the CCS. Code Composer Studio is flexible software that can work together with popular software, such as MATLAB, Lab View, Visual Basic, etc.

II. THE TMS320C6713 DSK

The TMS320C6x are the first processors to use velociTI architecture, having implemented the VLIW (Very Long Instruction Word) architecture. The TMS320C62x is a 16-bit fixed point processor and the ‘67x is a floating point processor, with 32-bit integer support. The C6713 DSK is a low-cost standalone development platform that enables users to evaluate and develop applications for the TI C67xx DSP family. [3]

The simplified architecture of TMS320C6713 is shown in the Figure.1 below. The processor consists of three main parts: CPU, peripherals and memory.

The CPU contains program fetch unit, Instruction dispatch unit, instruction decode unit. The CPU fetches advanced very-long instruction words (VLIW) to supply up to eight 32-bit instructions to the eight functional units during every clock cycle. The first bit of every 32-bit instruction determines if the next instruction belongs to the same execute packet as the previous instruction, or whether it should be executed in the following clock as a part of the next execute packet. The CPU contains two general purpose register files A and B. These can be used for data or as data address pointers. Each file contains sixteen 32-bit registers. [11]

The CPU features two sets of functional units. Each set contains four units and a register file. Each functional unit has two 32-bit read ports for source operands and one 32-bit write port into a general purpose register file. The memory system of the TMS320C671x series processor implements a modified

Harvard architecture, providing separate address spaces for instruction and data memory. [10] The development tool used for TMS320C6713DSK is TI Code Composer Studio (CCS).

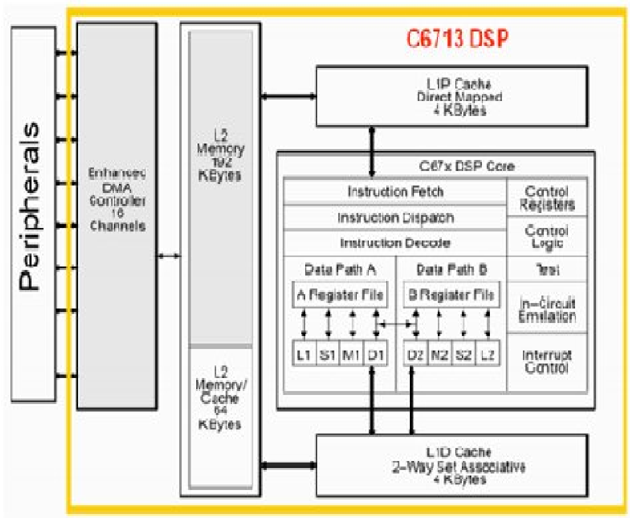

Fig.1. Simplified block diagram of TMS320C67xx core architecture

It offers robust core functions with easy-to-use configuration and graphical visualization tools for system design. Programming in C is used to write for the application. It is then compiled, linked and executed by the CCS. With the efficiency of the C programming code, the desired applications are to be made programmable and user-friendly. [6]

III. CODE-BUILDING PROCESS

The DSP code-building process relies on a set of software development tools, typically provided by DSP manufacturers. Figure2. Shows the main elements and tools needed for the code-building process

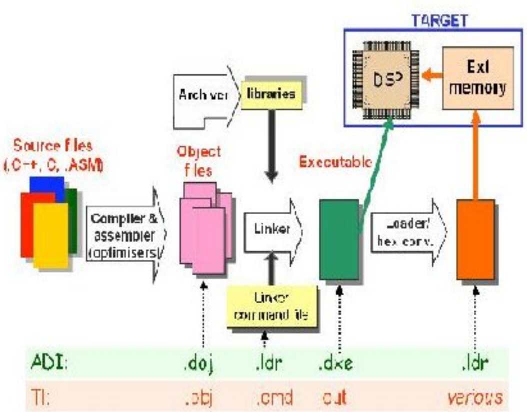

Fig2. Main elements of the code building process.

Source files are converted to object files by the compiler and the assembler. Archiver tools allow the creation of libraries from object files; these libraries can then be linked to object files to create an executable. The executable can be directly downloaded from the IDE to the target DSP via a JTAG interface; as an alternative, the executable can be converted to a special form and loaded to a memory external to the DSP, from which the DSP itself will boot. The first approach is typically used during the DSP development phase, while the second approach is more convenient during system exploitation. Three tools, namely compiler, assembler, and linker, are used to generate executable code from C/C++ or assembly source code. Figure 3 shows their use in the code-building process on TI DSPs. [5]

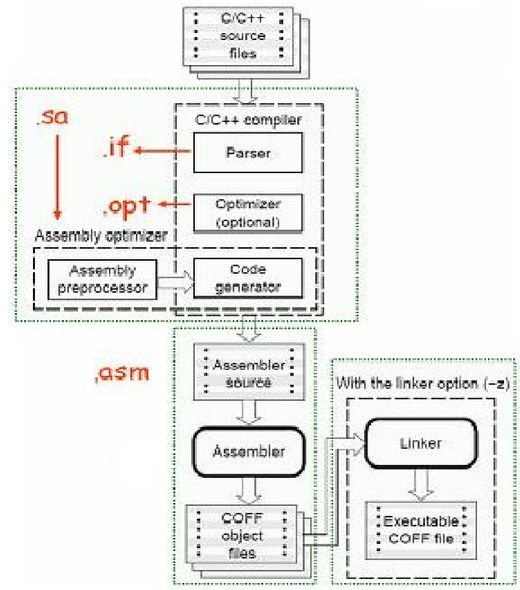

Fig.3. Generic code-building processing

IV. METHODOLOGY

Beside a source code to program a TMS 320 C 6713 processor, BIOS configuration file and linker command file need to be added into the project. The Board Support Library is a software library that provides a C-language interface for configuring and controlling all on-board devices. Individual Application Programming Interface (API) is used to represent the devices on board. Hence, appropriate API to be included on source program as per the hardware features to be used.

For generation of a variety of signal a look-up table method has been used to generate. The samples of the signal to be generated are pre-calculated using a known algorithm over one cycle and stored in the digital memory. The preferred signals are retrieved from the look–up table stored in the memory using the proper number of samples within one cycle. The analogue signal is generated by passing the digital output through an inbuilt digital-to-analogue converter (TLV320AIC23) in the DSP starter kit. Different types of signals that can be generated are sine wave, square wave, triangle wave, periodic arbitrary waveforms, pseudo random noise sequence and amplitude modulated signal. When generating waveform, three aspects are considered; namely type of waveform, amplitude and frequency of waveform. These waveforms are generated using different algorithms.

Frequency and amplitude of the waveforms can be varied by changing the algorithms too. It is possible to obtain two synchronized signals of different types simultaneously. The operation of the signal generator is very simple. The user has to enter the three essential parameters, namely type, frequency and amplitude of the desired signal at the start of the operation. [8]

Figure4. Consider typical periodic arbitrary signal waveforms as shown in figure4.

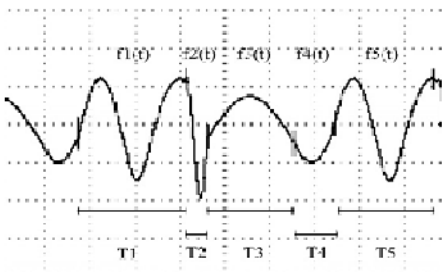

Fig 4. Arbitrary waveforms.

The above signal waveform is a mixture of sine waves with a range of frequencies. These can be obtained by defining the waveform through the following equation:

$$\begin{aligned} Y &= f_1(t) && 0<t<T_1 \\ Y &= f_2(t) && T_1<t<T_2 \\ Y &= f_3(t) && T_2<t<T_3 \\ Y &= f_4(t) && T_3<t<T_4 \end{aligned} \quad (1)$$

.
.
.

The digital samples of functions $f_1(t)$, $f_2(t)$, $f_3(t)$, $f_4(t)$……...are stored in a RAM. The clock signal is used to read samples stored in to the RAM. The output of the memory is passed through a digital to analogue converter to get the analogue output signal. This is illustrated in the block diagram shown in Figure 5. Changing the clock frequency can vary the frequency of the arbitrary waveform. [8]

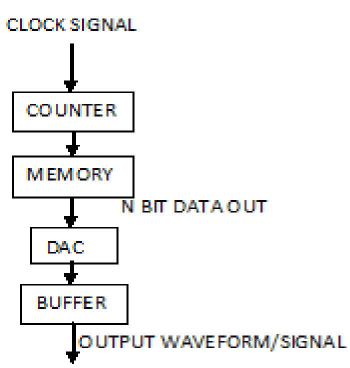

Fig 5. Block diagram of periodic arbitrary waveform generation.

The DSP processor generates signal waveforms with variable frequency. The frequency is obtained by the following equation:

$$f_o = \frac{fs}{sample/cycle} \quad (2)$$

Where f_o = output frequency
fs = sampling frequency
Sample/cycle = number of sample in one cycle

The sampling frequency is the sampling rate of the codec of DSP processor, which can be varied using the appropriate API. The sampling rate used here is 8 kHz. For sine wave, as shown in Figure 6, can be generated by *sin ( )* function.
The algorithm used is as below:

Sine wave = a*sin (2* PI * (sample/cycle)) (3)

Where a = amplitude of waveform
Sample/cycle = number of sample in one cycle

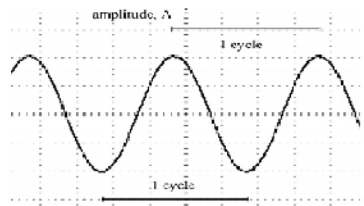

Fig.6. Sine wave generated by function generator

For square wave, as shown in Figure 7, is generated by the following definition:

$$\text{Square wave} = \begin{cases} 1, & T1 \\ -1, & T2 \end{cases} \quad (4)$$

For the first half cycle, the amplitude is 1 and for the second cycle the amplitude is -1, or vice versa.

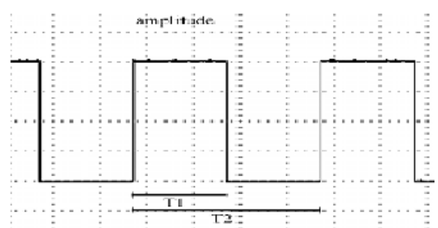

Fig.7. Square wave generated by function generator

Whereas, a simple mathematic algorithm for straight line is used for triangle wave generation:

$$\begin{aligned} y &= mt \\ y &= -V + mt && 0 \le t \le T1 \\ y &= V + m(t-T1) && T1 \le t \le T2 \end{aligned} \quad (5)$$

Where y = amplitude of waveform
m = slope, t = time

Along with these waveforms saw tooth, multitone waveforms, Pseudo Random Noise Sequence Generation, Digital Modulation Schemes like Pulse Amplitude Modulation and Phase Shift Keying can be implemented.[4]

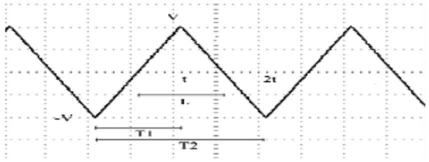

Fig.8. Triangle wave generated by function generator

Other than single waveforms, amplitude modulation can be produced using DSP processor as well. Amplitude modulation is a process of shifting a signal frequency to another frequency, called carrier frequency, which is usually higher. The information, or better the content of the signal (modulating) signal is transferred to the carrier frequency. Frequency shifting is done by multiplication of two signals in the time domain. Therefore, amplitude modulation produced by the DSP processor has used the algorithm as below:

Mod=carrier signal*modulating signal (6)

Where Mod = amplitude modulated signal

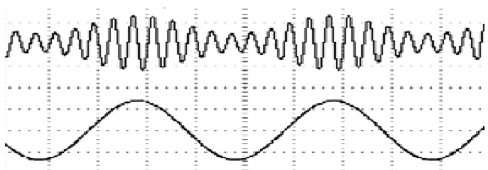

Fig.9. Amplitude modulated output carrier frequency sine wave.

Synchronization is a very important issue when more than one waveform is involved. It can be solved using phase-locked loop (PLL) system. Moreover, the initial point of the waveforms can be specified for synchronization purpose. Using the look-up table, the desired initial point can be defined by calculation. [4]

V. APPLICATIONS OF SIGNAL GENERATOR

The generation of signals is an important development in the troubleshooting and development of electronic design. The signal generator is used to provide known test conditions for the performance evaluation of electronic system design and for replacing signals that are missing in system during repairing work. While signal generators are widely used to test and maintain a wide range of RF equipment, such as radio receivers and transmitters, they are also extremely useful for testing digital, clock driven systems, especially high speed serial storage and serial communication ICs, circuit boards and devices. In general signal generators are used in designing, testing, troubleshooting, and repairing electronic devices specially related to communication. [7]

VI. CONCLUSION

There are many applications for which the Digital Signal Processor becomes an ideal choice as they provide the best possible combination of performance, power and cost. By using digital signal processing flexible signal generator that are often used in measuring and testing of communication systems is implemented. The output frequency range can be increased by using an external high-speed digital-to-analogue converter. The DSP processor TMS320C6713DSK with Code Composer Studio and C programming language has been used to generate the desired waveforms. Two different types of waveforms can be generated simultaneously using the stereo codec of TMS320C6713DSK. Because of the high performance of DSP processor and the efficiency of the C language, signal generation has become user-friendly and more programmable.

REFERENCES

[1] TMS320C6713 DSK Technical Reference, Spectrum Digital, Inc., May 2003.

[2] Code Composer Studio Help, Texas Instrument, 2003

[3] Rajean Arseneau, Michelle E. Sutherland, and J.J. Zelle,'A Test System for Calibrating Flicker meters', IEEE Transactions Instrumentation and Measurement, Vol. 51, No. 4, August 2002.

[4] Sia Lih Huoy, S.S. Jamuar, Roslina Mohd. Sidek, and Mohd.Hamiruce Marhaban, 'Digital Signal Processing Based Waveform Generator for Flicker meter Calibration Test System'4th Student Conference on Research and Development (SCOReD 2006), Shah Alam, Selangor, MALAYSIA, 27-28 June, 2006, IEEE 2006

[5] TMS320C6000 Optimizing Compiler – User's Guide, Texas Instruments Literature Number SPRU187L, May 2004.

[6] TMS320C6713, TMS320C6713B Floating-Point Digital Signal Processors, Texas Instruments Manual SPRS186I, December 2001, Revised May 2004.

[7] http://www.electronicsprojectdesign.com/SignalGenerator

[8] H. Sia, S.S. Jamuar, Roslina Mohd Sidek and Mohd Hamiruce Marhaban 'Digital-signal-processor-based waveform generator' IOP PUBLISHING, Meas. Sci. Technol. 18 (2007) N35–N40

[9] E.A. Lee, Programmable DSP Architectures: Part I, *IEEE ASSP Mag*., October 1988.

[10] TMS320C621x/C671x DSP Two-Level Internal Memory Reference Guide, Texas Instruments Literature Number SPRU609A, November 2003.

[11] D. Talla, L.K. John, V. Lapinskii and B.L. Evans, Evaluating Signal Processing and Multimedia Applications on SIMD, VLIW and Superscalar Architectures, Proceedings of the International Conference on Computer Design, September 2000.